การจดจำวงจรโลจิกเกทที่เขียนด้วยมือ โดยใช้หลักการแบ่งแยกลายเส้น

HANDWRITTEN LOGIC CIRCUIT RECOGNITION BASED ON LINE SEGMENTATION

สมคิด อุนการุณวงษ์

SOMKID UNKAROONWONG

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2545

ISBN 974-648-588-1

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การจดจำวงจรโลจิกเกทที่เขียนด้วยมือ
โดยใช้หลักการแบ่งแยกลายเส้น

HANDWRITTEN LOGIC CIRCUIT RECOGNITION BESED ON LINE SEGMENTATION

สมคิด อุนการุณวงษ์
SOMKID UNKAROONWONG

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมไฟฟ้า
บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ. 2545

ISBN 974-648-588-1

HANDWRITTEN LOGIC CIRCUIT RECOGNITION BASED ON LINE SEGMENTATION

SOMKID UNKAROONWONG

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

MASTER OF ENGINEERING IN ELECTRICAL ENGINEERING

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2002

ISBN 947-648-588-1

COPYRIGHT 2002

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

หัวข้อวิทยานิพนธ์	การจดจำวงจรโลจิกเกทที่เขียนด้วยมือโดยใช้หลักการแบ่งแยกลายเส้น
นักศึกษา	นายสมคิด อุนการุณวงษ์
รหัสประจำตัว	38061250
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมไฟฟ้า
พ.ศ.	2545
อาจารย์ผู้ควบคุมวิทยานิพนธ์	รศ. เกษตร์ ศิริสันติสัมฤทธิ์

บทคัดย่อ

วิทยานิพนธ์ฉบับนี้จะนำเสนอการจดจำวงจรโลจิกเกทที่เขียนด้วยมือโดยใช้หลักการแบ่งแยกลายเส้น วงจรโลจิกเกทที่เขียนด้วยมือจะถูกสแกนด้วยเครื่องสแกนเนอร์และถูกแปลงเป็นข้อมูลภาพไบนารี่ ข้อมูลภาพดังกล่าวจะถูกนำมาปรับปรุงคุณภาพของลายเส้น ด้วยการลดและเพิ่มขนาดของลายเส้น จากนั้นทำการหาเส้นขอบของลายเส้น ซึ่งเส้นขอบวงปิดแต่ละเส้นจะถูกกรองสัญญาณด้วยตัวกรองเกาซ์เซียนและหาจุดมุมแบ่งแยกลายเส้นด้วยการหาค่าความโค้ง เส้นขอบวงปิดใดมีจุดมุมแบ่งแยกลายเส้นมากกว่า 2 จุดแต่น้อยกว่า 5 จุด จะถูกพิจารณาว่าเป็นตัวโลจิกเกท จากนั้นวิเคราะห์องค์ประกอบของลายเส้นสำหรับเส้นขอบวงปิดที่เป็นโลจิกเกท โดยการทำโค้งกระชับด้วยวิธีกำลังสองน้อยที่สุด ซึ่งจะทำให้สามารถจดจำและจำแนกประเภทได้ตามโครงสร้างหลัก 4 ชนิดบนพื้นฐานของหลักการแบ่งแยกลายเส้น โลจิกเกทที่ตรวจพบจะนำมาวิเคราะห์หาจำนวนอินพุทและเอาท์พุทด้วยวิธีการติดตามลายเส้นและโทโปโลยี ผลลัพธ์ภายหลังการวิเคราะห์หาอินพุทและเอาท์พุทจะทำให้การจดจำชนิดของโลจิกเกทถูกต้องแม่นยำมากขึ้น จากนั้นวิเคราะห์การต่อของลายวงจรระหว่างตัวโลจิกเกท และตัวโลจิกเกทกับจุดต่อร่วม และหาทิศทางหลักของวงจรโลจิกเกท จากนั้นทำการเขียนวงจรโลจิกเกทขึ้นมาใหม่ด้วยการย่อขยายตัวโลจิกเกทให้เป็นไปตามมาตราฐานที่กำหนด ประสิทธิภาพการทำงานในการรู้จำวงจรโลจิกเกทที่เขียนด้วยมือของวิธีการที่นำเสนอได้แสดงไว้ในผลการทดลอง

Thesis Title	Handwritten Logic Circuit Recognition Based on Line Segmentation
Student	Mr. Somkid Unkaroonwong
Student ID.	38061250
Degree	Master of Engineering
Programme	Electrical Engineering
Year	2002
Thesis Advisor	Assoc.Prof. Kaset Sirisantisamrid

ABSTRACT

This thesis presents handwritten logic circuit recognition based on line segmentation. A handwritten logic circuit is scanned by scanner and transformed to binary image. The image is enhanced the line quality by thinning and blurring the line. Then, the edges of line are detected. The closed loop edges are reduced the corrupted noise by using Gaussian filter and detected the corner points by curvature calculation. The closed loop edges that have more than one and less than five corners points are considered as the logic gates. After that, the characteristics of all lines that are component of logic gate are analyzed by using curve fitting with method of least square. The logic gates are recognized and classified to 4-types of main structure based on line segmentation. The logic gates to be found are analyzed the number of input-output by using line tracing and topology. The results of input-output analyzing will accurate recognize the logic gate type. Then, the circuit connection is analyzed as logic gate with logic gate, logic gate with junction point etc. After that, the main direction of circuit is determined and the size of logic gates is resized by scaling follow up to decided standard. Finally, the new logic circuit is redrawn. The performances of the proposed method shown at experimental result section.

กิตติกรรมประกาศ

ในการจัดทำวิทยานิพนธ์ฉบับนี้จนสำเร็จลุล่วงไปด้วยดีตามวัตถุประสงค์นั้น เริ่มต้นจากแนวความคิดของ รศ. เกษตร์ ศิริสันติสัมฤทธิ์ อาจารย์ที่ปรึกษาและเป็นผู้ควบคุมวิทยานิพนธ์ ซึ่งให้ความกรุณาอนุเคราะห์แนะนำเกี่ยวกับงานวิจัยนี้ ตลอดจนกระตุ้นเตือนและสอบถามถึงความคืบหน้าของงานวิจัยนี้อย่างต่อเนื่องสม่ำเสมอ ผู้วิจัยรู้สึกซาบซึ้งและขอกราบขอบพระคุณท่านอาจารย์เป็นอย่างสูง

ขอขอบพระคุณมารดา และพี่น้องทุกคนของผู้วิจัยที่คอยช่วยเหลือและให้กำลังใจจนงานวิจัยสำเร็จ

ขอขอบพระคุณอาจารย์และเจ้าหน้าที่ของภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรมทุกท่านที่กรุณาให้ความเอื้อเฟื้อสถานที่ในการทำวิจัยจนสำเร็จลุล่วงไปด้วยดี

สมคิด อุนการุณวงษ์

สารบัญ

หน้า

บทคัดย่อภาษาไทย I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญตาราง VII

สารบัญภาพ VIII

บทที่ 1 บทนำ 1

1.1 ความเป็นมาและความสำคัญของปัญหา 1

1.2 จุดมุ่งหมายและวัตถุประสงค์ 1

1.3 ขอบเขตของการศึกษา 1

1.4 หลักการที่นำเสนอ 2

1.5 รายละเอียดของวิทยานิพนธ์ 3

บทที่ 2 การศึกษาและวิจัยที่ผ่านมา 4

2.1 แนวความคิดการจดจำโลจิกเกทโดยใช้หลักการแบ่งแยกลายเส้น 4

2.2 ขั้นตอนการประมวลผลของการจดจำวงจรโลจิกเกทที่เขียนด้วยมือ 5

2.2.1 การแยกโลจิกเกทออกจากวงจร 6

2.2.2 การหาจุดแบ่งแยกลายเส้น 7

2.2.2.1 การหาจุดมุมของเส้นขอบวงปิด 7

2.2.2.2 การหาจุดเริ่มต้นเส้นโค้งของเส้นขอบวงปิด 9

2.2.3 การวิเคราะห์ลักษณะลายเส้น 10

2.2.4 การหาอินพุทและเอาท์พุทของโลจิกเกท 11

2.2.4.1 การหาเอาท์พุท 12

2.2.4.2 การหาอินพุท 13

2.2.5 การวิเคราะห์การต่อวงจร 14

2.2.6 การปรับทิศทางและการวาดภาพ 14

2.3 บทสรุป 16

สารบัญ (ต่อ)

หน้า

บทที่ 3 การวิเคราะห์โครงสร้างของโลจิกเกท 18

3.1 การปรับปรุงคุณภาพของลายเส้น 19

3.1.1 การทำลายเส้นให้บาง (Thinning algorithm) 21

3.1.2 การทำลายเส้นให้เบลอ (Blur algorithm) 23

3.2 การแยกโลจิกเกทออกจากวงจร 24

3.3 การหาจุดแบ่งแยกลายเส้น 26

3.3.1 การประสานเส้นขอบวงปิดด้วยตัวกรองเกาส์เซียน 26

3.3.2 การหาค่าความโค้งบนเส้นขอบวงปิด 29

3.4 การวิเคราะห์ลักษณะลายเส้น 32

3.5 การจดจำและจำแนกโลจิกเกทตามโครงสร้างหลัก 35

บทที่ 4 การหาอินพุท-เอาท์พุทและการสร้างภาพวงจรขึ้นใหม่ 37

4.1 การหาโครงร่างของภาพ 38

4.2 การหาอินพุท-เอาท์พุทของโลจิกเกท 38

4.2.1 การหาเอาท์พุท 40

4.2.1.1 การค้นหาเส้นโครงร่างของตัวโลจิกเกท 40

4.2.1.2 การตรวจสอบจุดแยกบนเส้นโครงร่างของโลจิกเกท 42

4.2.1.3 การตรวจสอบวงกลม NOT 43

4.2.2 การหาอินพุท 45

4.3 การวิเคราะห์การต่อวงจร 47

4.4 การสร้างภาพวงจรโลจิกเกทขึ้นใหม่ 48

4.4.1 การปรับปรุงข้อมูล 48

4.4.2 การวาดภาพ 52

บทที่ 5 ผลการทดลอง 54

5.1 การทดลองการวิเคราะห์โครงสร้างของโลจิกเกท 54

5.1.1 การเก็บภาพวงจรโลจิกเกทเข้าคอมพิวเตอร์ 54

5.1.2 ผลการวิเคราะห์โครงสร้างของโลจิกเกท 54

สารบัญ (ต่อ)

หน้า

5.1.3 สรุปผลการวิเคราะห์โครงสร้างของโลจิกเกท 64
5.2 การทดลองการหาอินพุท-เอาท์พุท และการสร้างภาพวงจรขึ้นใหม่ 64
5.2.1 ผลการหาอินพุท-เอาท์พุท และการสร้างภาพวงจรขึ้นใหม่ 64
5.2.2 สรุปผลการหาอินพุท-เอาท์พุท และการสร้างภาพวงจรโลจิกเกทขึ้นใหม่ 72

บทที่ 6 บทสรุปและข้อเสนอแนะ 73

เอกสารอ้างอิง 75
ภาคผนวก ผลงานวิจัยที่ได้รับการตีพิมพ์ 77
ประวัติผู้เขียน 96

สารบัญตาราง

ตารางที่ หน้า

1.1 สัญลักษณ์ของโลจิกเกท 2

2.1 องค์ประกอบทางโครงสร้างของโลจิกเกท 4

4.1 การเลือกใช้รูปแบบการลากเส้นต่อวงจร 52

5.1 ผลการทดลองการวิเคราะห์โครงสร้างของโลจิกเกท 63

สารบัญรูป

รูปที่ หน้า

2.1 ขั้นตอนของการประมวลผล 5

2.2 Mask operator ขนาด 3x3 จุดภาพ 6

2.3 ภาพก่อนและหลังการหาขอบของภาพ 7

2.4 การหาค่าความโค้งของจุดใดๆ บนลายเส้น 8

2.5 การเปรียบเทียบค่าความโค้งของสมการที่ (2-1) กับ (2-2) 8

2.6 จุดมุมและจุดเริ่มต้นเส้นโค้งของ AND gate 9

2.7 จุดมุมและจุดเริ่มต้นเส้นโค้ง 9

2.8 กราฟค่าความโค้งใหม่ 10

2.9 ระยะทางต่างๆ ระหว่างจุดบนลายเส้นกับเส้นตรง (P_jP_k) 10

2.10 ภาพก่อนและหลังการหาลายเส้นโครงร่างของภาพ 12

2.11 การกำหนดตำแหน่งจุดอ้างอิง และการหาเอาท์พุท 13

2.12 การกำหนดตำแหน่งจุดอ้างอิง และการหาอินพุท 14

2.13 ตำแหน่งอินพุทและเอาท์พุทที่หาได้ 14

2.14 รูปแบบการลากเส้นต่อวงจร 15

2.15 วงจรโลจิเกทที่สร้างขึ้นใหม่ 16

2.16 ลักษณะของ AND gate และ NAND gate 16

3.1 ชนิดและโครงสร้างหลักของโลจิกเกท 18

3.2 ขั้นตอนการวิเคราะห์โครงสร้างของโลจิกเกท 19

3.3 ข้อผิดพลาดที่เกิดขึ้นจากลายเส้นไม่สมบูรณ์ 20

3.4 ภาพก่อนและหลังการทำลายเส้นให้บาง 1 รอบ 23

3.5 ภาพก่อนและหลังการทำลายเส้นให้เบลอ 24

3.6 รูปแบบของจุดภาพที่เกิดขึ้นหลังการปรับปรุงลายเส้น 24

3.7 ภาพก่อนและหลังการหาขอบของภาพ 25

3.8 เส้นกราฟของสมการตัวกรองเกาซ์เซียน 26

3.9 วิธีการเก็บข้อมูลและการกำหนดเส้นทางเดินของจุดภาพบนเส้นขอบวงปิด 27

3.10 ขนาดหน้าต่างสำหรับเก็บค่าสัมประสิทธ์ของตัวกรองเกาซ์เซียน
ที่ค่าสเกล-สเปซต่างๆ กัน 28

สารบัญรูป (ต่อ)

รูปที่ หน้า

3.11 คุณสมบัติของตัวกรองเกาซ์เซียนในการกรองเส้นขอบวงปิด
โดยมีค่าสกล-สเปซต่างๆ กัน 29

3.12 ผลของการหาจุดมุมเมื่อเปลี่ยนแปลงค่าสเกล-สเปซ 31

3.13 ภาพเปรียบเทียบกลุ่มจุดภาพกับเส้นตรงที่ได้จากสมการ 32

3.14 การเปรียบเทียบลายเส้นและค่า R_{av} 35

3.15 ภาพข้อมูลอินพุทและผลของการวิเคราะห์โครงสร้าง 36

4.1 ขั้นตอนการหาอินพุท-เอาท์พุทและการสร้างภาพวงจรขึ้นใหม่ 37

4.2 ภาพก่อนและหลังการหาลายเส้นโครงร่างของภาพ 38

4.3 การกำหนดตำแหน่งจุดเริ่มต้นของการหาอินพุท-เอาท์พุท 39

4.4 การค้นหาเส้นโครงร่างของตัวโลจิกเกท 41

4.5 การตรวจสอบจุดแยกบนลายเส้นโครงร่าง 42

4.6 การตรวจหาวงกลม NOT 43

4.7 การหาอินพุทของโลจิกเกทโครงสร้างต่างๆ 45

4.8 การตรวจหาอินพุทของโลจิกเกทชนิด Exclusive 46

4.9 ภาพหลังการหาอินพุท–เอาท์พุทของโลจิกเกท 47

4.10 การแบ่งกลุ่มและการปรับทิศทางของโลจิกเกท 50

4.11 รูปแบบการลากเส้นต่อวงจร 52

4.12 ภาพวงจรโลจิกเกทที่สร้างขึ้นใหม่ 53

5.1 ภาพข้อมูลอินพุทขนาด 256x256 จุดภาพ 55

5.2 ผลของการปรับปรุงลายเส้น 55

5.3 ผลของการแยกตัวโลจิกเกทออกจากวงจร 55

5.4 ผลของการหาจุดแบ่งแยกลายเส้น 56

5.5 ผลของการวิเคราะห์โครงสร้าง 56

5.6 ภาพข้อมูลอินพุทขนาด 478x195 จุดภาพ 56

5.7 ผลของการปรับปรุงลายเส้น 57

5.8 ผลของการแยกตัวโลจิกเกทออกจากวงจร 57

5.9 ผลของการหาจุดแบ่งแยกลายเส้น 57

5.10 ผลของการวิเคราะห์โครงสร้าง 58

สารบัญรูป (ต่อ)

รูปที่ หน้า

5.11 ภาพข้อมูลอินพุทขนาด 453x373 จุดภาพ 58

5.12 ผลของการปรับปรุงลายเส้น 59

5.13 ผลของการแยกตัวโลจิกเกทออกจากวงจร 59

5.14 ผลของการหาจุดแบ่งแยกลายเส้น 60

5.15 ผลของการวิเคราะห์โครงสร้าง 60

5.16 ภาพข้อมูลอินพุทขนาด 272x234 จุดภาพ 61

5.17 ผลของการปรับปรุงลายเส้น 61

5.18 ผลของการแยกตัวโลจิกเกทออกจากวงจร 62

5.19 ผลของการหาจุดแบ่งแยกลายเส้น 62

5.20 ผลของการวิเคราะห์โครงสร้าง 63

5.21 ผลของการหาลายเส้นโครงร่าง 65

5.22 ผลของการหาอินพุท-เอาท์พุท 65

5.23 ผลของการวิเคราะห์การต่อวงจร 65

5.24 ผลของการสร้างภาพวงจรโลจิกเกทขึ้นใหม่ 66

5.25 ผลของการหาลายเส้นโครงร่าง 66

5.26 ผลของการหาอินพุท-เอาท์พุท 66

5.27 ผลของการวิเคราะห์การต่อวงจร 67

5.28 ผลของการสร้างภาพวงจรโลจิกเกทขึ้นใหม่ 67

5.29 ผลของการหาลายเส้นโครงร่าง 68

5.30 ผลของการหาอินพุท-เอาท์พุท 68

5.31 ผลของการวิเคราะห์การต่อวงจร 69

5.32 ผลของการสร้างภาพวงจรโลจิกเกทขึ้นใหม่ 69

5.33 ผลของการหาลายเส้นโครงร่าง 70

5.34 ผลของการหาอินพุท-เอาท์พุท 70

5.35 ผลของการวิเคราะห์วงจร 71

5.36 ผลของการสร้างภาพวงจรโลจิกเกทขึ้นใหม่ 71

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ปัจจุบันคอมพิวเตอร์เข้ามามีบทบาทในชีวิตประจำวันมากยิ่งขึ้น ประกอบกับวิทยาการด้านคอมพิวเตอร์มีการพัฒนาอย่างรวดเร็ว โดยเฉพาะอย่างยิ่งการนำคอมพิวเตอร์ไปใช้งานด้านการประมวลผลภาพดิจิตอล (Digital image processing) เพื่อการรู้จำวัตถุในภาพ เช่น การรู้จำภาพพิมพ์ลายนิ้วมือ (Fingerprint recognition) การรู้จำตัวอักษร (Character recognition) และระบบการมองเห็นของหุ่นยนต์ (Robot vision) เป็นต้น จุดมุ่งหมายของการนำคอมพิวเตอร์มาใช้งานก็เพื่ออำนวยความสะดวกต่อผู้ใช้ ลดเวลาในการทำงาน และประหยัดค่าใช้จ่าย

งานด้านการออกแบบวงจรโลจิกเกทในปัจจุบันนี้ ผู้ออกแบบวงจรหรือวิศวกรไฟฟ้าจะออกแบบโดยใช้ดินสอหรือปากกาเขียนภาพวงจรอย่างหยาบๆ เมื่อทำการทดลองและแก้ไขปรับปรุงแล้วก็มักจะส่งต่อให้ช่างเทคนิคทำการเขียนวงจรให้ใหม่ ซึ่งเป็นงานที่เสียเวลามาก ดังนั้นเราจึงมีแนวความคิดว่า ถ้าเราสามารถนำวงจรที่ออกแบบไว้อย่างหยาบๆ ซึ่งสามารถใช้งานได้แล้วมาสแกนด้วยเครื่องสแกนเนอร์ (Scanner) จากนั้นพัฒนาโปรแกรมที่สามารถจดจำลายเส้นต่างๆ ที่อยู่ในภาพว่าเป็นโลจิกเกทชนิดใด มีการต่อวงจรอย่างไร จากนั้นโปรแกรมก็จะจัดการเขียนลายวงจรขึ้นมาใหม่ตามมาตรฐานของโลจิกเกทที่กำหนดไว้โดยแสดงผลได้ที่หน้าจอ และสามารถพิมพ์ออกทางเครื่องพิมพ์ได้ ก็จะเป็นการประหยัดเวลา และแรงงานทั้งของวิศวกรผู้ออกแบบ และช่างเทคนิคที่ทำหน้าที่เขียนแบบได้ ทำให้เวลาที่เหลือนำไปใช้เพื่อการประดิษฐ์คิดค้นต่อไปได้

1.2 จุดมุ่งหมายและวัตถุประสงค์

1) เพื่อศึกษา วิจัย และประยุกต์ใช้การประมวลผลภาพ ในการวิเคราะห์และจดจำวงจรโลจิกเกทที่เขียนด้วยมือ
2) เพื่อพัฒนาซอฟท์แวร์ (Software) และอัลกอริทึม (Algorithms) สำหรับการจดจำวงจรโลจิกเกทที่เขียนด้วยมือ
3) เพื่ออำนวยความสะดวกกับผู้ออกแบบวงจรโลจิกเกท

1.3 ขอบเขตของการศึกษา

ขอบเขตหลักของการศึกษาและวิจัยการจดจำวงจรโลจิกเกทที่เขียนด้วยมือ มีดังนี้

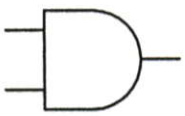
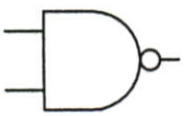
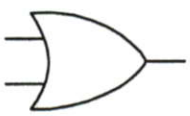
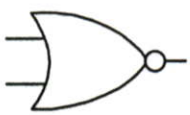
1) สามารถจดจำโลจิกเกทที่เขียนด้วยมือ ที่เป็นสัญลักษณ์แบบ Distinctive-shape Graphic Symbol ทั้ง 8 ชนิด คือ AND gate, NAND gate, OR gate, NOR gate, XOR gate, XNOR

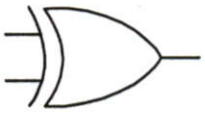
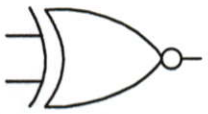
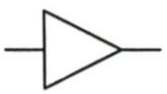
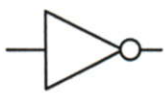
gate, BUFFER gate และ NOT gate รวมทั้งสัญลักษณ์ที่มีลักษณะเป็น BLOCK เช่น Flipflop และวงจรรวม เป็นต้น ซึ่งแสดงไว้ในตารางที่ 1.1 โดยไม่จำกัดทิศทางของโลจิกเกท และเส้นรอบรูปของตัวโลจิกเกทจะต้องมีขนาดมากกว่า 30 จุดภาพ (Pixel)

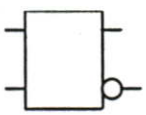
2) สามารถจดจำโลจิกเกทชนิดต่างๆ ที่มีวงกลม NOT ที่อินพุทของโลจิกเกทได้

3) สามารถวิเคราะห์การต่อวงจรของโลจิกเกทในภาพ และสร้างภาพวงจรโลจิกเกทขึ้นมาใหม่ในรูปแบบที่สวยงาม

ตารางที่ 1.1 สัญลักษณ์ของโลจิกเกท

Name	Symbol
AND gate	
NAND gate	
OR gate	
NOR gate	

Name	Symbol
XOR gate	
XNOR gate	
BUFFER gate	
NOT gate	

BLOCK	

1.4 หลักการที่นำเสนอ

การจดจำสัญลักษณ์โลจิกเกทที่เขียนด้วยมือจะใช้หลักการแบ่งแยกลายเส้น โดยการวิเคราะห์โครงสร้างของโลจิกเกทแต่ละตัว กล่าวคือจะพิจารณาองค์ประกอบของโลจิกเกทแต่ละตัวว่าประกอบด้วยลายเส้นตรงและลายเส้นโค้งเป็นจำนวนเท่าใด ซึ่งทำให้เราสามารถจดจำสัญลักษณ์

โลจิกเกทได้และสามารถจำแนกประเภทโลจิกเกทตามโครงสร้างหลักได้สี่ประเภท เมื่อทำการวิเคราะห์หาจำนวนอินพุทและเอาท์พุทของโลจิกเกทที่ถูกจำแนกประเภทแล้ว และนำมาใช้ร่วมกันจะทำให้สามารถจดจำชนิดของโลจิกเกทได้ถูกต้องแม่นยำมากยิ่งขึ้น

จากนั้น เราจะทำการวิเคราะห์การต่อลายวงจรระหว่างอินพุทกับอินพุท อินพุทกับเอาท์พุท อินพุทกับจุดต่อร่วม และเอาท์พุทกับจุดต่อร่วม ผลลัทธ์ที่ได้จากการวิเคราะห์การต่อลายวงจรจะถูกเก็บไว้เป็นฐานข้อมูล ในขั้นตอนนี้ทำให้เราสามารถจดจำวงจรโลจิกเกทที่เขียนด้วยมือได้ จากนั้นทำการย่อขยายขนาดของโลจิกเกท หมุนภาพโลจิกเกท และทำการเขียนวงจรโลจิกเกทขึ้นมาใหม่

1.5 รายละเอียดของวิทยานิพนธ์

ในวิทยานิพนธ์ฉบับนี้ได้แบ่งเนื้อหาที่ใช้ในการดำเนินการวิจัยไว้เป็นบทต่างๆ ด้วยกันทั้งหมด 6 บท ซึ่งพอจะกล่าวโดยสังเขปได้ดังนี้

1) บทที่ 1 บทนำ จะกล่าวถึงความเป็นมา และจุดมุ่งหมายของงานวิจัย ตลอดจนหลักการและขอบเขตของงานวิจัย
2) บทที่ 2 การศึกษาและวิจัยที่ผ่านมา จะกล่าวถึงวิธีการและขั้นตอนต่างๆ ที่เคยทำการศึกษาและทดลอง รวมทั้งจุดบกพร่องที่พบ
3) บทที่ 3 การวิเคราะห์โครงสร้างของโลจิกเกท จะนำเสนอรายละเอียดของขั้นตอน วิธีการและคณิตศาสตร์เกี่ยวกับการวิเคราะห์โครงสร้างหลักของโลจิกเกทที่ใช้ในวิทยานิพนธ์ฉบับนี้
4) บทที่ 4 การหาอินพุท-เอาท์พุท และการสร้างภาพวงจรขึ้นใหม่ จะนำเสนอรายละเอียดของวิธีการวิเคราะห์หาอินพุทและเอาท์พุทของโลจิกเกท ตลอดจนถึงการสร้างภาพวงจรโลจิกเกทขึ้นมาใหม่
5) บทที่ 5 ผลการทดลอง เป็นการแสดงให้เห็นผลลัพธ์ของการประมวลผลในขั้นตอนต่างๆ
6) บทที่ 6 บทสรุป จะกล่าวถึงปัญหาต่างๆ ที่เกิดขึ้นในงานวิจัย และนำเสนอแนวทางในการพัฒนาต่อไป

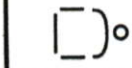
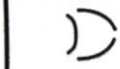
บทที่ 2
การศึกษาและวิจัยที่ผ่านมา

เนื้อหาที่นำเสนอในบทนี้ เป็นรายละเอียดเกี่ยวกับการจดจำวงจรโลจิกเกทที่เขียนด้วยมือโดยใช้หลักการแบ่งแยกลายเส้นที่เคยทำการศึกษาและวิจัยผ่านมาโดยสังเขป ทั้งนี้เพื่อให้เห็นข้อบกพร่องในขั้นตอนต่างๆ และมีการพัฒนาปรับปรุงเพิ่มประสิทธิภาพและขีดความสามารถของกระบวนการในบทต่อไป

2.1 แนวความคิดการจดจำโลจิกเกทโดยใช้หลักการแบ่งแยกลายเส้น

สัญลักษณ์ของโลจิกเกทมาตรฐานแต่ละชนิดจะมีความแตกต่างกันที่รูปลักษณ์หรือโครงสร้างและในการจดจำชนิดของโลจิกเกทถ้าพิจารณาจดจำที่รูปลักษณ์ของโลจิกเกทจะทำให้การจดจำมีความยุ่งยากและซับซ้อนมาก แต่ถ้าพิจารณาถึงองค์ประกอบทางโครงสร้างของสัญลักษณ์โลจิกเกทจะพบว่าสัญลักษณ์โลจิกเกทแต่ละชนิดมีองค์ประกอบที่เป็นลายเส้นตรงและลายเส้นโค้งเป็นจำนวนที่แตกต่างกัน ซึ่งสามารถแบ่งแยกโครงสร้างของโลจิกเกทแต่ละชนิดตามองค์ประกอบของลายเส้นได้ดังตารางที่ 2.1

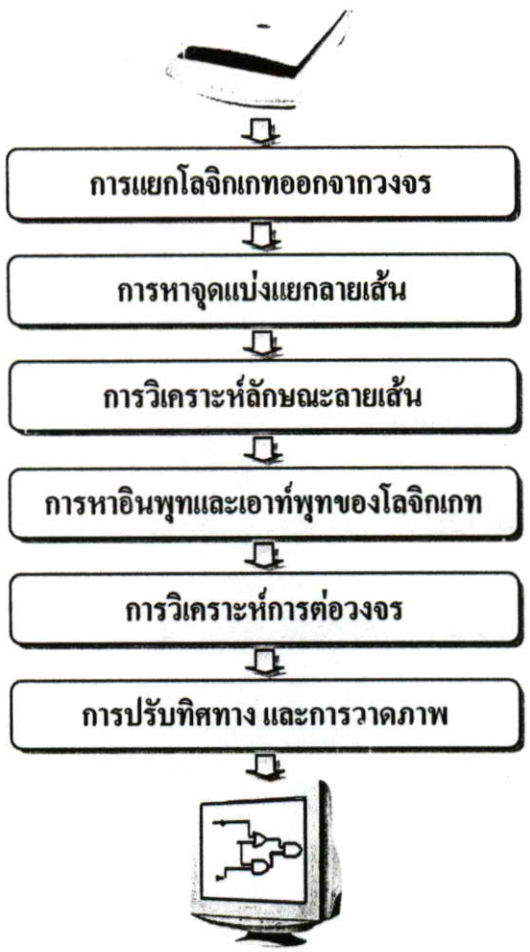
ตารางที่ 2.1 องค์ประกอบทางโครงสร้างของโลจิกเกท

โครงสร้าง / ชนิด	โครงสร้าง	เส้นตรง (เส้น)	เส้นโค้ง (เส้น)	วงกลมหน้า (วง)	เส้นหลัง (เส้น)
AND gate		3	1	ไม่มี	ไม่มี
NAND gate		3	1	1	ไม่มี
OR gate		ไม่มี	3	ไม่มี	ไม่มี
NOR gate		ไม่มี	3	1	ไม่มี
XOR gate		ไม่มี	3	ไม่มี	1
XNOR gate		ไม่มี	3	1	1
BUFFER gate		3	ไม่มี	ไม่มี	ไม่มี
NOT gate		3	ไม่มี	1	ไม่มี

การพิจารณาองค์ประกอบทางโครงสร้างของโลจิกเกทในลักษณะเช่นนี้ จะช่วยให้การวิเคราะห์ การรู้จำ และจำแนกชนิดของโลจิกเกทมีความซับซ้อนน้อยลง และเมื่อใช้จำนวนและตำแหน่งของอินพุท-เอาท์พุทของโลจิกเกทแต่ละตัวมาประกอบการตัดสินใจจะทำให้การจดจำชนิดของโลจิกเกทมีความถูกต้องมากยิ่งขึ้น

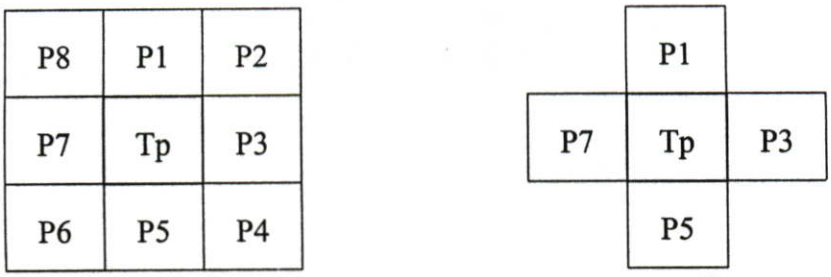
2.2 ขั้นตอนการประมวลผลของการจดจำวงจรโลจิกเกทที่เขียนด้วยมือ

การประมวลผลเพื่อจดจำวงจรโลจิกเกทที่เขียนด้วยมือนี้จะประกอบด้วยขั้นตอนต่างๆ ทั้งหมด 6 ขั้นตอน ดังแสดงไว้ในรูปที่ 2.1 โดยรายละเอียดของแต่ละขั้นตอนได้อธิบายในหัวข้อถัดไป



รูปที่ 2.1 ขั้นตอนของการประมวลผล

ในการประมวลผลภาพสำหรับการวิเคราะห์ในขั้นตอนต่างๆ นั้น จะใช้การประมวลผลภาพจากจุดรอบข้าง (Neighborhood points) เป็นหลัก โดยการนำจุดภาพ (Tp) พร้อมจุดรอบข้างมาพิจารณาเรียกรวมกันว่า Mask หรือ Window ซึ่งมีขนาด 3x3 จุด ดังแสดงในรูปที่ 2.2 และมี 8 จุดรอบข้าง (8-Neighbors) คือ P1, P2, P3, ..., P8 ส่วนจุด P1, P3, P5, P7 ถูกเรียกว่า จุดรอบข้าง 4 จุด (4-Neighbors)



(ก) จุดรอบข้าง 8 จุด (ข) จุดรอบข้าง 4 จุด

รูปที่ 2.2 Mask operator ขนาด 3x3 จุดภาพ

2.2.1 การแยกโลจิกเกทออกจากวงจร

ข้อมูลภาพวงจรที่ใช้ในการวิเคราะห์เป็นภาพวงจรโลจิกเกทที่เขียนด้วยมือ ซึ่งมีขนาด 256x256 จุดภาพ (Pixel) และเป็นภาพ 2 ระดับ (Binary image) การแยกตัวโลจิกเกทออกจากวงจรจะใช้วิธีการหาขอบของภาพ (Edge detection) ทั้งนี้เพื่อจุดประสงค์สองประการคือ ประการแรกแยกตัวโลจิกเกทออกจากวงจร ดังแสดงในรูปที่ 2.3 ประการที่สองลดความซับซ้อนของการวิเคราะห์ลายเส้นในขั้นตอนต่างๆ เนื่องจากลายเส้นขอบของภาพจะมีความหนาเพียง 1 จุดภาพ และไม่มีจุดแยกบนลายเส้น ซึ่งวิธีการหาขอบของภาพจะใช้ Mask operator ขนาด 3x3 จุดภาพ ดังแสดงไว้ในรูปที่ 2.2 เมื่อ T_p เป็นจุดภาพตรวจสอบที่เป็นลายเส้น (จุดภาพที่มีค่าเป็น "1") และมีเงื่อนไขจุดขอบของภาพดังนี้

<u>เงื่อนไขการตรวจสอบจุดขอบของภาพ</u>

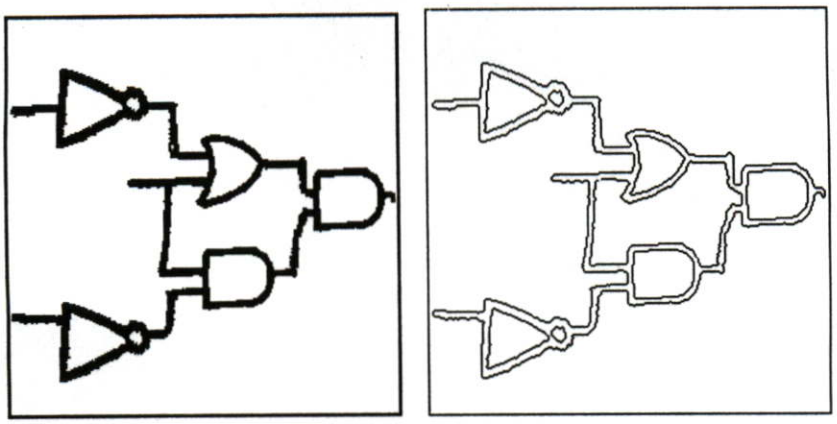
1) $\overline{(P_1 \cdot P_3 \cdot P_5 \cdot P_7)} = 1$

2) $(P_2 + P_4 + P_6 + P_8) \geq 2$

3) $\sum_{i=0}^{7} P_i > 2$

เมื่อ (.) เป็น ตรรกะ AND

จุดภาพลายเส้นจุดใดเป็นไปตามเงื่อนไขทั้ง 3 ข้อ จะถูกทำเครื่องหมายไว้ (flag) และเมื่อตรวจสอบจนครบทั้งภาพแล้ว จุดที่ถูกทำเครื่องหมายไว้จะถูกเปลี่ยนเป็นลายเส้นขอบของภาพดังแสดงในรูปที่ 2.3 ลายเส้นขอบของภาพเหล่านี้จะมีลักษณะเป็นวงปิด (Closed loop) ดังนั้นในที่นี้จะเรียกว่า "เส้นขอบวงปิด" ซึ่งเส้นขอบวงปิดทั้งหมดจะถูกนำไปวิเคราะห์โครงสร้างในขั้นตอนต่อไป



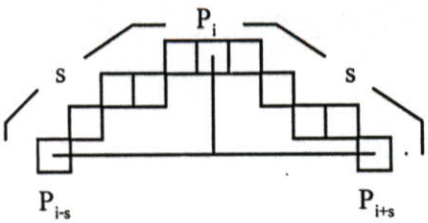
รูปที่ 2.3 ภาพก่อนและหลังการหาขอบของภาพ

2.2.2 การหาจุดแบ่งแยกลายเส้น

ในการแบ่งแยกเส้นขอบวงปิดออกเป็นลายเส้น เพื่อวิเคราะห์ลักษณะและจำนวนของลายเส้นจะต้องหาจุดแบ่งแยกบนลายเส้นก่อน ซึ่งการหาจุดแบ่งแยกบนลายเส้นนี้เป็นสิ่งสำคัญมาก เพราะถ้าหากการหาจุดแบ่งแยกลายเส้นผิดพลาดจะทำให้การแบ่งแยกลายเส้น การวิเคราะห์ลักษณะของลายเส้น รวมถึงการจดจำชนิดของโลจิกเกทผิดพลาดไปด้วย จากการจดจำวงจรโลจิกเกทที่เขียนด้วยมือใน [1] การหาจุดแบ่งแยกลายเส้นจะใช้วิธีการติดตามจุดภาพบนเส้นขอบวงปิดโดยใช้ 4-Connectivity พร้อมทั้งตรวจสอบการเปลี่ยนแปลงทิศทางของจุดภาพ ซึ่งวิธีการนี้สามารถหาจุดแบ่งแยกลายเส้น ได้เฉพาะกรณีที่เกทมีทิศทางตามแนวแกน X หรือแกน Y เท่านั้น ซึ่งทำให้มีข้อจำกัดในการใช้งานมาก ดังนั้นเมื่อพิจารณาจากรูปแบบของโลจิกเกทในตารางที่ 2.1 จะเห็นว่าโลจิกเกททุกชนิดจะใช้จุดมุมเป็นจุดแบ่งแยกลายเส้น ยกเว้นกรณีของ AND gate และNAND gate จะต้องหาจุดแบ่งแยกลายเส้นเพิ่มอีก 2 จุด คือบริเวณจุดต่อของเส้นตรงกับเส้นโค้ง หรือจุดเริ่มต้นเส้นโค้ง ด้วยเหตุนี้การหาจุดแบ่งแยกลายเส้นจึงถูกแบ่งออกเป็น 2 ขั้นตอน คือการหาจุดมุมและการหาจุดเริ่มต้นเส้นโค้ง โดยขั้นตอนทั้งสองนี้สามารถหาจุดแบ่งแยกลายเส้นได้ไม่ว่าเกทจะมีทิศทางใดก็ตาม ซึ่งรายละเอียดมีดังต่อไปนี้

2.2.2.1 การหาจุดมุมของเส้นขอบวงปิด

จุดมุมของเส้นขอบวงปิดหมายถึง จุดที่มีการเปลี่ยนแปลงทิศทางของลายเส้นอย่างรวดเร็ว หรือจุดที่มีค่าความโค้ง (K) มากๆ ซึ่งการหาจุดมุมหรือการหาค่าความโค้งที่นำเสนอใน [10] คือ กำหนดช่วงของจุดภาพที่ติดต่อกัน โดยมีจุดที่ต้องการหาค่าความโค้งเป็นจุดกึ่งกลางของช่วง (P_i) และคำนวณหาค่าความโค้งได้จากระยะทางจากจุดกึ่งกลาง (P_i) ถึงเส้นตรงที่ลากจากจุดเริ่มต้นของช่วง (P_{i-s}) ไปยังจุดสุดท้ายของช่วง (P_{i+s}) ในแนวตั้งฉากดังรูปที่ 2.4



รูปที่ 2.4 การหาค่าความโค้งของจุดใดๆ บนลายเส้น

สมการหาค่าความโค้งของจุดใดๆ บนลายเส้นจะเป็นดังนี้

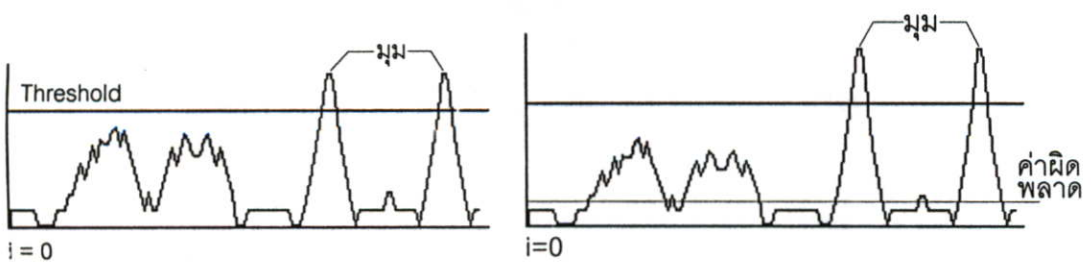
$$K_i = \left| \frac{(y_{i+s} - y_{i-s})x_i - (x_{i+s} - x_{i-s})y_i + x_{i+s}y_{i-s} - y_{i+s}x_{i-s}}{\sqrt{(y_{i+s} - y_{i-s})^2 + (x_{i+s} - x_{i-s})^2}} \right| \quad (2\text{-}1)$$

เมื่อ i คือ จุดตรวจสอบใดๆ บนเส้นขอบวงปิด (i = 0, 1, 2, …, n)

n คือ จำนวนจุดบนลายเส้นขอบวงปิด

จากสมการที่ (2-1) สามารถหาค่าความโค้งของทุกๆ จุดบนเส้นขอบวงปิดได้ดังรูปที่ 2.5(ก) ซึ่งค่าความโค้งที่ได้นี้จะมีค่ามากหรือน้อยขึ้นอยู่กับการกำหนดช่วง (s) ถ้ากำหนดช่วง (s) มากค่าความโค้งก็จะมาก ในทางกลับกันถ้ากำหนดช่วง (s) น้อยค่าความโค้งก็จะน้อย ซึ่งมีผลให้การกำหนดค่า Threshold ทำได้ยากหรือต้องกำหนดค่า Threshold หลายค่า ดังนั้นเพื่อให้การกำหนดค่า Threshold หนึ่งค่า สามารถครอบคุมการกำหนดช่วง (s) ได้หลายๆ ค่าสามารถทำได้โดยการทำให้ค่าความโค้งเป็นอัตราส่วนกับระยะห่างจากจุดเริ่มต้นของช่วง (Pi-s) กับจุดสุดท้ายของช่วง (Pi+s) ดังสมาการที่ (2-2)

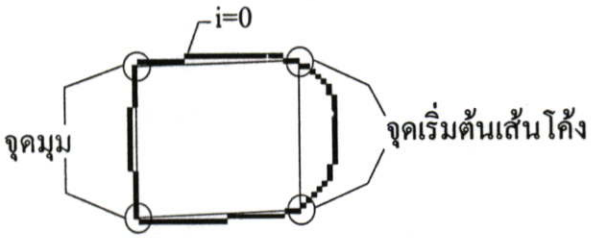
$$K_i = \left| \frac{(y_{i+s} - y_{i-s})x_i - (x_{i+s} - x_{i-s})y_i + x_{i+s}y_{i-s} - y_{i+s}x_{i-s}}{(y_{i+s} - y_{i-s})^2 + (x_{i+s} - x_{i-s})^2} \right| \quad (2\text{-}2)$$



(ก) กราฟค่าความโค้งที่ได้จากสมการที่ (2-1) (ข) กราฟค่าความโค้งที่ได้จากสมการที่ (2-2)

รูปที่ 2.5 การเปรียบเทียบค่าความโค้งของสมการที่ (2-1) กับ (2-2)

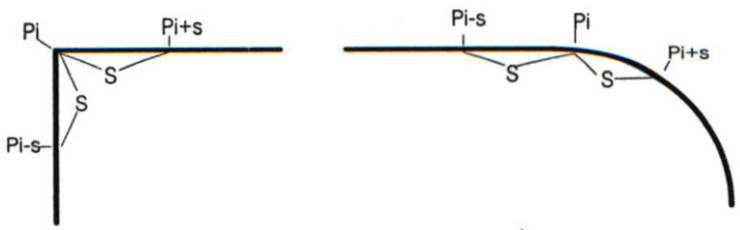
ค่าความโค้งที่หาได้จากสมการที่ (2-2) แสดงในรูปที่ 2.5 (ข) ซึ่งจะเห็นว่ารูปแบบของกราฟจะเหมือนเดิมแต่ค่าความโค้งบริเวณที่เป็นจุดมุมจะสูงเด่นขึ้น เมื่อผ่านขั้นตอนนี้ถ้าเส้นขอบวงปิดใดสามารถหาจุดมุมได้ 3 จุด จะพิจารณาว่าเส้นขอบวงปิดนั้นอาจจะเป็นโลจิกเกทชนิด OR gate, NOR gate, XOR gate, XNOR gate, NOR gate หรือ BUFFER gate และสามารถนำจุดมุมที่ได้นี้ไปแบ่งแยกลายเส้นเพื่อวิเคราะห์ลักษณะของลายเส้นในขั้นตอนต่อไป แต่ถ้าเส้นขอบวงปิดนั้นหาจุดมุมได้ 2 จุด จะพิจารณาว่าอาจจะเป็นโลจิกเกทชนิด AND gate หรือ NAND gate และจะต้องหาจุดแบ่งแยกลายเส้นเพิ่มอีก 2 จุด ในขั้นตอนการหาจุดเริ่มต้นเส้นโค้ง ส่วนกรณีที่เส้นขอบวงปิดมีจุดมุมมากกว่า 3 มุม หรือน้อยกว่า 2 มุม จะพิจารณาว่าเส้นขอบวงปิดนั้นไม่ใช่โลจิกเกท



รูปที่ 2.6 จุดมุมและจุดเริ่มต้นเส้นโค้งของ AND gate

2.2.2.2 การหาจุดเริ่มต้นเส้นโค้งของเส้นขอบวงปิด

จุดเริ่มต้นเส้นโค้ง หรือจุดต่อระหว่างเส้นตรงกับเส้นโค้งสามารถหาได้ โดยนำค่าความโค้งที่ได้ในขั้นตอนการหาจุดมุมมาพิจารณา จากรูปที่ 2.7 (ก) จุด P_i คือจุดที่มีค่าความโค้งมาก (จุดมุม) และจุดที่ถัดจากจุด P_i ออกมาจะมีค่าความโค้งลดลงจนถึงจุด P_{i-s} และ จุด P_{i+s} จะมีค่าความโค้งเป็นศูนย์ (หรือน้อยกว่าค่าผิดพลาดที่กำหนด) ดังรูปที่ 2.5 (ข) และจุดที่ถัดจากจุด P_{i-s} และ จุด P_{i+s} ก็จะมีค่าความโค้งเป็นศูนย์ เนื่องจากเป็นเส้นตรง แต่ถ้าจุด P_i เป็นจุดเริ่มต้นเส้นโค้งดังรูปที่ 2.7 (ข) ค่าความโค้งที่จุด P_{i+s} และจุดถัดไปจะไม่เป็นศูนย์เพราะเป็นเส้นโค้ง ในขณะที่ค่าความโค้งที่จุด P_{i-s} และจุดถัดไปเป็นศูนย์ ดังนั้นเราสามารถนำจำนวนจุดที่มีค่าความโค้งเป็นศูนย์นี้ (หรือน้อยกว่าค่าผิดพลาดที่กำหนด) มาพิจารณาร่วมกับค่าความโค้งที่จุด P_i ได้ ดังสมาการที่ (2-3)



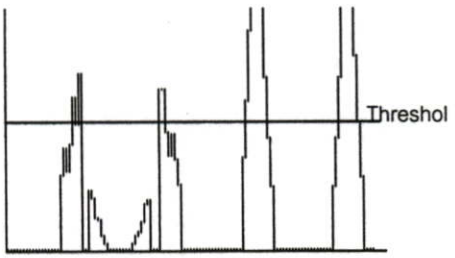
(ก) จุดมุม (ข) จุดเริ่มต้นเส้นโค้ง

รูปที่ 2.7 จุดมุมและจุดเริ่มต้นเส้นโค้ง

$$K'_i = K_i\sqrt{t_1^2+t_2^2} \tag{2-3}$$

เมื่อ t_1 คือ จำนวนจุดที่มีค่าความโค้งน้อยกว่าค่าผิดพลาดที่กำหนด
โดยเริ่มนับจากจุด P_{i-s} ในทิศทางลดลง

t_2 คือ จำนวนจุดที่มีค่าความโค้งน้อยกว่าค่าผิดพลาดที่กำหนด
โดยเริ่มนับจากจุด P_{i+s} ในทิศทางเพิ่มขึ้น

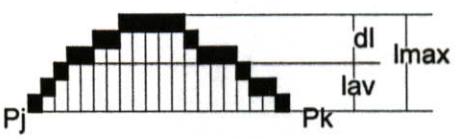


รูปที่ 2.8 กราฟค่าความโค้งใหม่

จากสมการที่ (2-3) จะทำให้บริเวณจุดต่อระหว่างเส้นตรงกับเส้นโค้งมีค่าความโค้งมากขึ้น (ดังรูปที่ 2.8) และสามารถนำจุดที่ได้นี้ไปแบ่งแยกลายเส้น เพื่อใช้ในขั้นตอนต่อไป

2.2.3 การวิเคราะห์ลักษณะลายเส้น

การวิเคราะห์ลักษณะลายเส้นว่าเป็นเส้นตรงหรือเส้นโค้ง ในที่นี้จะพิจารณาจากระยะทางต่างๆ ภายใต้ลายเส้นที่อยู่ระหว่างจุดแบ่งแยกลายเส้น 2 จุดกับเส้นตรง (P_jP_k) ดังรูปที่ 2.9 ซึ่งระยะทางดังกล่าวสามารถหาได้จากสมการดังต่อไปนี้



รูปที่ 2.9 ระยะทางต่างๆ ระหว่างจุดบนลายเส้นกับเส้นตรง (P_jP_k)

ระยะทางระหว่างจุดใดๆ บนลายเส้น (l_i) กับเส้นตรง (P_jP_k) คำนวณได้ดังนี้

$$l_i = \frac{(y_k - y_j)x_i - (x_k - x_j)y_i + x_k y_j - y_k x_j}{\sqrt{(y_k - y_j)^2 + (x_k - x_j)^2}} \tag{2-4}$$

เมื่อ x_k,y_k และ x_j,y_j คือ พิกัดในแนวแกน x และ y ของจุดแบ่งแยกลายเส้น
x_i,y_i คือ พิกัดในแนวแกน x และ y ของจุดใดๆ บนลายเส้น

สมการหาระยะทางเฉลี่ยภายใต้เส้นโค้ง (l_{av}) จะเป็นดังนี้

$$l_{av} = \frac{\sum_{i=j}^{k} l_i}{\sqrt{(y_k - y_j)^2 + (x_k - x_j)^2}} \qquad (2\text{-}5)$$

และผลต่างระหว่างระยะทางสูงสุดกับระยะทางเฉลี่ย (dl) หาได้จาก

$$dl = l_{\max} - l_{av} \qquad (2\text{-}6)$$

เมื่อ

$$l_{\max} = \max\{|l_i|\} \qquad (2\text{-}7)$$

การนำระยะทางต่างๆ ภายใต้ลายเส้นมาพิจารณาวิเคราะห์ลักษณะความเป็นเส้นตรงและเส้นโค้งของลายเส้นนั้นสามารถทำได้โดยใช้เงื่อนไขดังต่อไปนี้

<u>เงื่อนไขการวิเคราะห์ลักษณะลายเส้น</u>

1) $(l_{av} < 1) \mid ((l_{av} < 1.5) . (dl < 1))$ เป็นเส้นตรง

2) $(l_{av} > 2) \mid (dl > 2)$ เป็นเส้นโค้ง

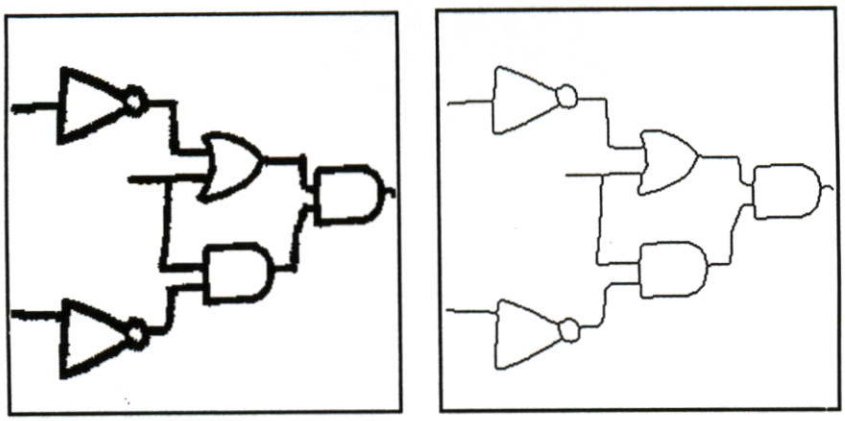
เมื่อ (.) คือ ตรรกะ AND

(|) คือ ตรรกะ OR

ในกรณีที่ไม่เป็นไปตามเงื่อนไขทั้ง 2 ข้อจะถือว่าลายเส้นนั้นเป็นได้ทั้งเส้นตรง และเส้นโค้ง เมื่อผ่านขั้นตอนนี้เส้นขอบวงปิดจะถูกวิเคราะห์เป็นโลจิกเกท 3 ชนิด คือ OR gate, AND gate และ BUFFER gate

2.2.4 การหาอินพุทและเอาท์พุทของโลจิกเกท

การวิเคราะห์ในขั้นตอนที่ผ่านมาจะวิเคราะห์กับลายเส้นขอบของภาพ แต่ในการวิเคราะห์หาอินพุทและเอาท์พุทของโลจิกเกทจะวิเคราะห์ที่ลายเส้นโครงร่างของภาพ ดังนั้นจะต้องนำภาพข้อมูลอินพุทไปหาลายเส้นโครงร่าง (Skeleton) โดยวิธีการทำลายเส้นให้บาง (จะกล่าวถึงต่อไปในหัวข้อที่ 3.1.1) และมีผลแสดงในรูปที่ 2.10

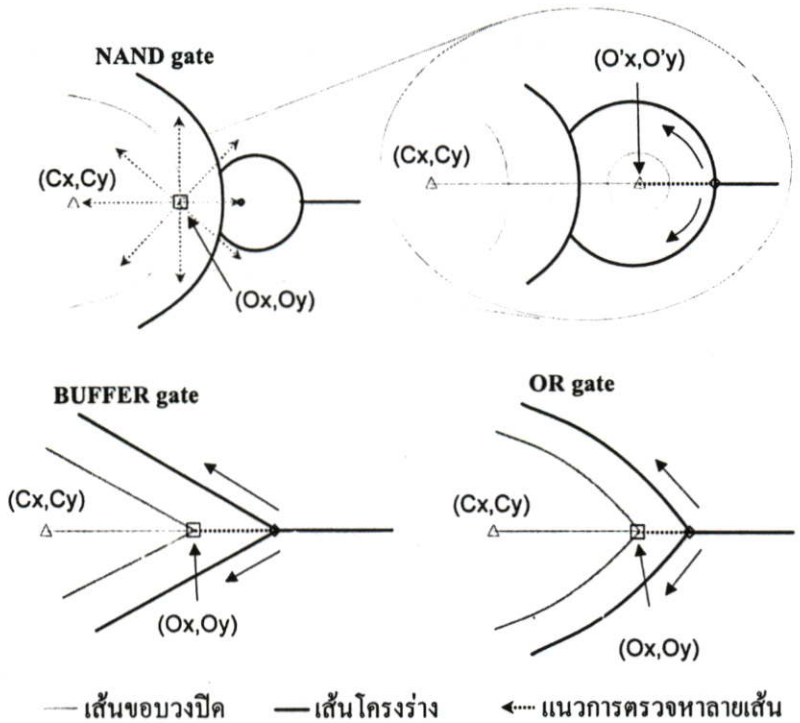


รูปที่ 2.10 ภาพก่อนและหลังการหาลายเส้นโครงร่างของภาพ

เนื่องจากโลจิกเกทมาตรฐานทั้ง 8 ชนิดจะมีจำนวนเอาท์พุทเพียง 1 เอาท์พุท และมีตำแหน่งที่แน่นอน ขณะที่อินพุทจะมีจำนวนมากกว่าหรือเท่ากับ 1 อินพุท ทั้งนี้ขึ้นอยู่กับชนิดของโลจิกเกท ดังนั้นการหาเอาท์พุทจึงทำได้ง่ายกว่า และถ้าหาเอาท์พุทไม่พบจะไม่ทำการหาอินพุทอีก

2.2.4.1 การหาเอาท์พุท

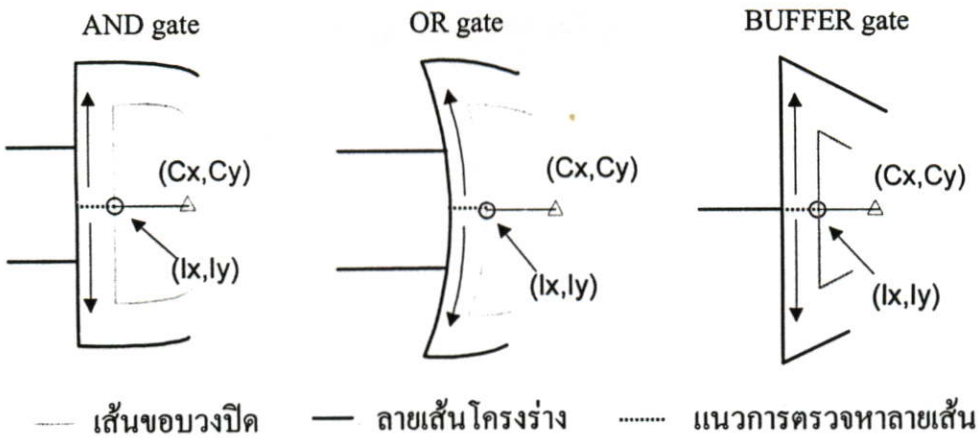
ในการหาเอาท์พุทของโลจิกเกทจะต้องกำหนดจุดอ้างอิงที่ใกล้กับจุดเอาท์พุท (หรือบริเวณที่ควรจะมีจุดแยกเอาท์พุท) ซึ่งการกำหนดแบ่งออกเป็น 2 กรณี คือ กรณีของ OR gate หรือ BUFFER gate จะใช้จุดมุมทั้งสามจุดเป็นจุดอ้างอิง ส่วนกรณี AND gate จะใช้จุดกึ่งกลางลายเส้นที่เป็นเส้นโค้งเป็นจุดอ้างอิงดังแสดงในรูปที่ 2.11 จากนั้นจะตรวจหาวงกลม NOT รอบๆ จุดอ้างอิง (Ox,Oy) โดยการตรวจสอบเป็นเส้นตรงออกจากจุดอ้างอิง (Ox,Oy) 8 ทิศทาง ถ้าไม่พบวงกลม NOT (BUFFER และ OR gate ในรูปที่ 2.11) จะตรวจหาลายเส้นโครงร่างโดยตรวจสอบเป็นเส้นตรงออกจากจุดศูนย์กลางของโลจิกเกท (Cx,Cy) ผ่านจุดอ้างอิง (Ox,Oy) ไปจนพบเส้นโครงร่างและจะติดตามลายเส้นหาจุดแยกเอาท์พุทบนเส้นโครงร่างนั้น แต่ถ้าพบวงกลม NOT (NAND gate ในรูปที่ 2.11) จะหาตำแหน่งศูนย์กลางของวงกลม NOT นั้นเพื่อนำไปคำนวณทิศทางของเกท และกำหนดเป็นจุดอ้างอิงใหม่ (O'x,O'y) จากนั้นจะตรวจหาลายเส้นโครงร่างที่ล้อมรอบวงกลม NOT ด้วยวิธีการเช่นเดียวกันคือ จะตรวจสอบเป็นเส้นตรงออกจากจุดศูนย์กลางของโลจิกเกท (Cx,Cy) ผ่านจุดอ้างอิง (O'x,O'y) ไปจนพบเส้นโครงร่าง และติดตามลายเส้นหาจุดแยกเอาท์พุทบนเส้นโครงร่างดังกล่าว



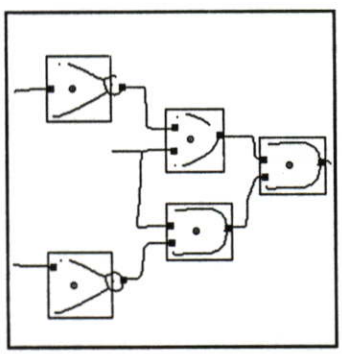
รูปที่ 2.11 การกำหนดตำแหน่งจุดอ้างอิง และการหาเอาท์พุท

2.2.4.2 การหาอินพุท

เมื่อรู้ตำแหน่งของเอาท์พุท ก็สามารถกำหนดจุดอ้างอิงสำหรับการหาอินพุท (Ix,Iy) เช่นเดียวกับการหาเอาท์พุท แต่กำหนดที่ลายเส้นด้านตรงข้ามกับตำแหน่งเอาท์พุทดังรูปที่ 2.12 จากนั้นจะตรวจหาลายเส้นโครงร่าง โดยการตรวจสอบเป็นเส้นตรงออกจากจุดศูนย์กลางของโลจิกเกท (Cx,Cy) ผ่านจุดอ้างอิง (Ix,Iy) ไปจนกระทั่งพบเส้นโครงร่างและติดตามลายเส้นหาจุดแยกอินพุทบนเส้นโครงร่างนั้น แต่ถ้าเป็นกรณีของ XOR gate และ XNOR gate จะไม่พบจุดแยกอินพุทบนลายเส้นด้านหลังของตัวโลจิกเกท ดังนั้นจะต้องหาลายเส้นด้านหลังใหม่ โดยการตรวจหาเป็นเส้นตรงออกจากลายเส้นหลังของโลจิกเกท เมื่อพบเส้นหลังใหม่ก็จะติดตามลายเพื่อหาจุดแยกอินพุทเหมือนครั้งแรก ผลของการหาอินพุทและเอาท์พุทแสดงไว้ในรูปที่ 2.13 เมื่อถึงขั้นตอนนี้จากข้อมูลที่ทำการวิเคราะห์มาทั้งหมดจะสามารถจดจำชนิดของโลจิกเกทได้



รูปที่ 2.12 การกำหนดตำแหน่งจุดอ้างอิง และการหาอินพุท



รูปที่ 2.13 ตำแหน่งอินพุทและเอาท์พุทที่หาได้

2.2.5 การวิเคราะห์การต่อวงจร

หลังจากการหาอินพุท-เอาท์พุทของโลจิกเกทแล้ว ลายเส้นที่เป็นตัวโลจิกเกทจะถูกตัดขาดจากอินพุท-เอาท์พุทดังรูปที่ 2.13 ดังนั้นการวิเคราะห์การต่อวงจรของโลจิกเกทแต่ละตัวจึงไม่ซับซ้อน เพียงแต่ติดตามลายเส้นจากอินพุท-เอาท์พุทของโลจิกเกททีละตัว เมื่อพบจุดแยก (จุดต่อร่วม) ตำแหน่งของจุดแยกจะถูกเก็บไว้ แต่ถ้าไม่พบจุดแยกติดตามลายเส้นจนสุดลายเส้น (จุดปลาย) จะนำตำแหน่งจุดปลายไปเปรียบเทียบกับตำแหน่งอินพุท-เอาท์พุทของโลจิกเกททุกตัว เพื่อตรวจสอบการต่อวงจรระหว่างโลจิกเกทกับโลจิกเกท เมื่อตรวจสอบอินพุท-เอาท์พุทของโลจิกเกทครบทุกตัวจะนำตำแหน่งจุดแยกที่เก็บไว้มาตรวจสอบโดยการติดตามลายเส้นเช่นเดียวกัน เมื่อตรวจสอบครบทุกจุดแยกแล้วจะทำให้เรารู้การต่อวงจรของโลจิกเกททุกตัวและจุดแยกทุกจุด

2.2.6 การปรับทิศทางและการวาดภาพ

เมื่อสามารถจดจำชนิดและตำแหน่งของโลจิกเกทจนครบทั้งภาพแล้ว จะต้องคำนวณหาทิศทางของโลจิกเกทแต่ละตัวเพื่อปรับทิศทางให้เหมาะสมก่อนการสร้างภาพวงจรใหม่ การคำนวณทิศทางของโลจิกเกทสามารถทำได้โดยใช้จุดศูนย์กลางของโลจิกเกท และจุดศูนย์กลางของวงกลม

NOT หรือตำแหน่งเอาท์พุทของโลจิกเกท (กรณีที่ไม่มีวงกลม NOT) และคำนวณโดยใช้สมการหาทิศทางทั่วไปดังนี้

$$\beta = \tan^{-1}\left(\frac{Uy - Cy}{Ux - Cx}\right) \tag{2-8}$$

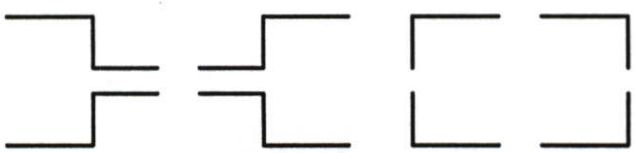
เมื่อ β คือ ทิศทางของโลจิกเกท

(Cx,Cy) คือ พิกัดของจุดศูนย์กลาง

(Ux,Uy) คือ พิกัดของจุดศูนย์กลางวงกลม NOT หรือเอาท์พุท

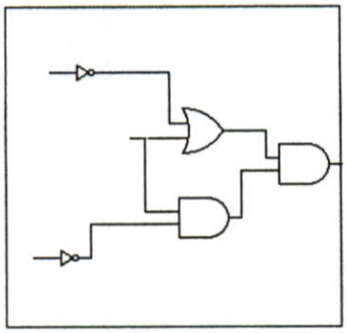
การปรับตำแหน่งและทิศทางของโลจิกเกทสามารถทำได้โดยพิจารณาทิศทางของโลจิกเกทแต่ละตัวและแยกโลจิกเกทออกเป็นกลุ่มๆ ซึ่งแต่ละกลุ่มจะมีทิศทางของโลจิกเกทต่างกันมากกว่า ±45 องศา เนื่องจากรูปแบบของโลจิกเกทมาตรฐานจะมีทิศทางตามแนวแกน x และแกน y เท่านั้น จากนั้นจะหาทิศทางหลักของโลจิกเกทเพื่อหมุนหรือปรับตำแหน่งเกททุกตัวรวมทั้งจุดแยกให้เหมาะสม โดยคำนวณทิศทางเฉลี่ยจากทิศทางของโลจิกเกทในกลุ่มที่มีจำนวนโลจิกเกทมากที่สุด (จะกล่าวถึงโดยละเอียดในหัวข้อที่ 4.4.1)

เมื่อปรับตำแหน่งโลจิกเกทและจุดแยกแล้วจะวาดภาพตัวโลจิกเกทชนิดต่างๆ ตามตำแหน่งและทิศทางใหม่จนครบทุกตัว ขั้นตอนต่อไปคือ การลากเส้นต่อวงจร ซึ่งรูปแบบการลากเส้นจะเป็นเส้นตรงที่ลากในแนวแกน x และแกน y ดังรูปที่ 2.14



รูปที่ 2.14 รูปแบบการลากเส้นต่อวงจร

การลากเส้นต่อวงจรจะเริ่มลากจากอินพุท-เอาท์พุทของโลจิกเกททีละตัวและใช้รูปแบบการลากเส้นทีละแบบจนกว่าจะสำเร็จ (เนื่องจากการลากเส้นมีการตรวจสอบการชนกันของลายเส้น) เมื่อต่อวงจรของโลจิกเกทครบทุกตัวจะทำการลากเส้นต่อวงจรที่จุดแยกเช่นเดียวกันจนครบ (โลจิกเกทหรือจุดแยกที่ถูกต่อวงจรสมบูรณ์แล้วจะไม่มีการลากเส้นต่อวงจรซ้ำอีก)

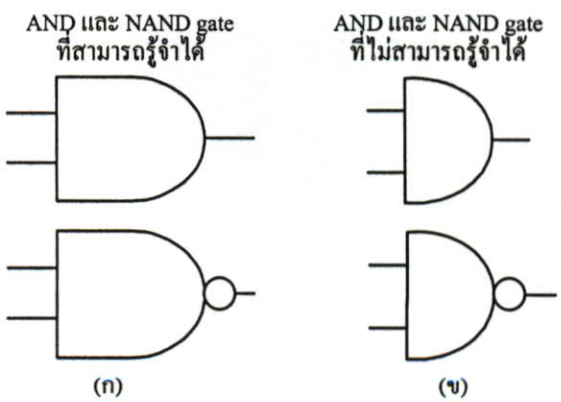


รูปที่ 2.15 วงจรโลจิเกทที่สร้างขึ้นใหม่

2.3 บทสรุป

ในการทดลองวิธีการและขั้นตอนข้างต้น ทำให้พบข้อบกพร่องและปัญหาในขั้นตอนต่างๆ ซึ่งสรุปได้ดังนี้

1) ภาพวงจรโลจิกเกทที่มีความหนาของลายเส้นน้อยกว่า 3 จุดภาพ จะไม่สามารถแยกตัวโลจิกเกทออกจากวงจรได้ เนื่องจากลายเส้นขอบของภาพแต่ละเส้นไม่แยกออกจากกัน และลายเส้นที่มีช่องว่าง (Hold) บนลายเส้น เมื่อหาลายเส้นโครงร่างจะเกิดจุดแยกที่ไม่ถูกต้องขึ้นบนลายเส้นจึงทำให้การวิเคราะห์การต่อวงจรผิดพลาด
2) การกำหนดโครงสร้างมาตรฐานของ AND gate และ NAND gate สร้างความซับซ้อนในการหาจุดแบ่งแยกลายเส้น (จุดมุม 2 จุด และจุดเริ่มต้นเส้นโค้ง 2 จุด) ทำให้เสียเวลาในการประมวลผลและเกิดข้อผิดพลาดมากขึ้น รวมทั้งเป็นการจำกัดขอบเขตการจดจำ AND gate และ NAND gate ให้สามารถจดจำได้เฉพาะโลจิกเกทที่มีลักษณะดังรูปที่ 2.16 (ก) แต่ไม่สามารถจดจำรูปที่ 2.16 (ข) ได้



รูปที่ 2.16 ลักษณะของ AND gate และ NAND gate

3) ขั้นตอนการหาจุดแบ่งแยกลายเส้น การกำหนดระยะตรวจสอบ (s) กับค่า Threshold จะต้องเปลี่ยนแปลงให้เหมาะสมกับขนาดของโลจิกเกท โดยโลจิกเกทที่มีขนาดใหญ่ระยะ

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ตรวจสอบ (s) และค่า Threshold จะต้องมากจึงจะสามารถหาจุดมุมได้ถูกต้อง และในทางกลับกันโลจิกเกทที่มีขนาดเล็กระยะตรวจสอบ (s) และค่าค่า Threshold จะต้องน้อยจึงจะสามารถหาจุดมุมได้ถูกต้อง แต่ความสัมพันธ์ระหว่างขนาดของโลจิกเกทกับระยะตรวจสอบ (s) และค่า Threshold ไม่เป็นเชิงเส้น

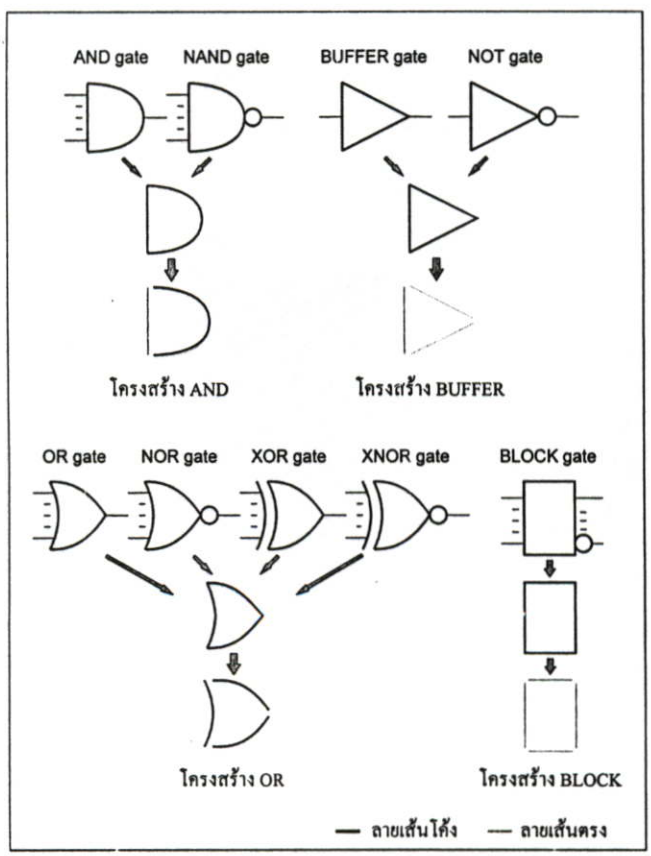
4) ขั้นตอนการวิเคราะห์ลักษณะลายเส้น การวิเคราะห์จะใช้ระยะทางใต้ลายเส้น (ดังแสดงในรูปที่ 2.9) เป็นตัวกำหนดเงื่อนไข (ความเป็นเส้นตรงและเส้นโค้งของลายเส้น) จึงทำให้การวิเคราะห์ลายเส้นตรงและลายเส้นโค้งไม่ชัดเจน และมีเงื่อนไขที่ซับซ้อน
5) ขั้นตอนการหาอินพุทและเอาท์พุท กระบวนการหาเอาท์พุทและอินพุทมีกระบวนการที่แตกต่างกัน และกระบวนการหาอินพุทไม่สามารถหาวงกลม NOT ที่อินพุทได้
6) ขั้นตอนการลากเส้นต่อลายวงจรในการสร้างภาพใหม่มีรูปแบบของการลากเส้นน้อย และไม่มีการเลือกใช้รูปแบบการลากเส้นที่เหมาะสมกับทิศทางของโลจิกเกท

ข้อบกพร่องและปัญหาที่กล่าวมาทั้งหมดนี้ ได้มีการศึกษาค้นคว้าและหาวิธีการใหม่รวมทั้งมีการปรับปรุงกระบวนการให้เหมาะสมเพื่อแก้ไขข้อบกพร่องและปัญหาดังกล่าว ซึ่งจะได้นำเสนอในบทต่อไป

42561

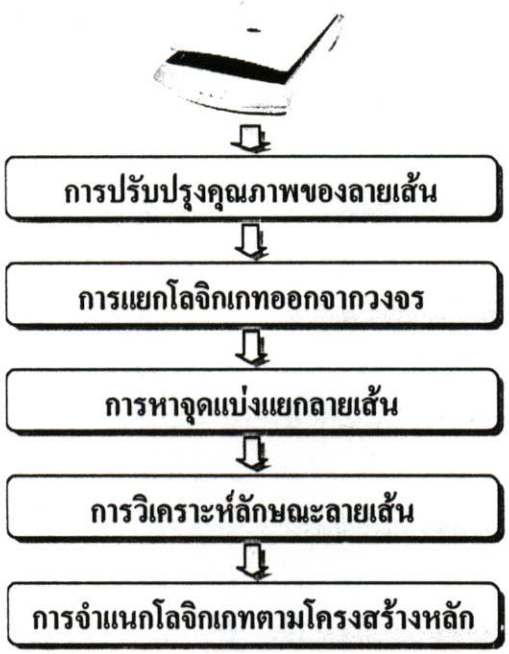
บทที่ 3
การวิเคราะห์โครงสร้างของโลจิกเกท

ขั้นตอนการวิเคราะห์โครงสร้างของโลจิกเกทเป็นขั้นตอนการหาตำแหน่งและวิเคราะห์โครงสร้างของโลจิกเกทแต่ละตัวในภาพวงจร เปรียบเทียบกับโครงสร้างหลักของโลจิกเกท ซึ่งเป็นการวิเคราะห์โครงสร้างโดยพิจารณาจากองค์ประกอบทางโครงสร้างของโลจิกเกทแต่ละชนิดที่ประกอบด้วยเส้นตรงและเส้นโค้งในจำนวนแตกต่างกัน เช่น OR gate ประกอบด้วยเส้นโค้ง 3 เส้น และ BUFFER gate ประกอบด้วยเส้นตรง 3 เส้น เป็นต้น โลจิกเกทบางชนิดมีโครงสร้างเหมือนกันแต่แตกต่างกันที่องค์ประกอบของอินพุท-เอาท์พุทกล่าวคือ มีวงกลม NOT ที่เอาท์พุทหรือมีลายเส้นหลังที่อินพุท เช่น กรณีที่โลจิกเกทเป็น Exclusive สัญลักษณ์ของโลจิกเกทที่นิยมใช้กันทั่วไป (แบบ Distinctive-shape) มีอยู่ 8 ชนิด คือ AND gate, NAND gate, OR gate, NOR gate, XOR gate, XNOR gate, BUFFER gate และ NOT gate รวมทั้งสัญลักษณ์แบบ BLOCK เช่น Flipflop โดยทั้งหมดนี้สามารถแบ่งตามโครงสร้างหลักได้เป็น 4 ชนิด คือ โครงสร้าง AND, OR, BUFFER และ BLOCK ดังรูปที่ 3.1 ซึ่งหลักการวิเคราะห์โครงสร้างของโลจิกเกทโดยแบ่งแยกโครงสร้างของโลจิกเกทออกเป็นลายเส้นนี้เราเรียกว่า "หลักการแบ่งแยกลายเส้น (Line segmentation)"



รูปที่ 3.1 ชนิดและโครงสร้างหลักของโลจิกเกท

ขั้นตอนในการวิเคราะห์โครงสร้างของโลจิกเกทประกอบด้วย 5 ขั้นตอน ดังแสดงในรูปที่ 3.2 ขั้นตอนแรกจะเป็นการปรับปรุงคุณภาพของลายเส้นในภาพวงจรให้สมบูรณ์และมีความหนามากกว่าหรือเท่ากับ 3 จุดภาพ โดยประยุกต์ใช้วิธีการทำลายเส้นให้บาง (Thinning algorithm) และการทำลายเส้นให้เบลอ (Blur algorithm) ทั้งนี้เพื่อป้องกันความผิดพลาดที่จะเกิดขึ้นในการวิเคราะห์โครงสร้าง จากนั้นจะทำการแยกโลจิกเกทออกจากวงจรด้วยการหาขอบภาพ (Edge detection) เพื่อลดความซับซ้อนของการวิเคราะห์โครงสร้าง เนื่องจากลายเส้นขอบที่ได้มีความหนาเพียง 1 จุดภาพ และไม่มีจุดแยกบนลายเส้น ซึ่งภาพที่ได้นี้จะเป็นภาพลายเส้นขอบวงปิดของตัวโลจิกเกทและส่วนประกอบของลายวงจร ต่อจากนั้นเส้นขอบวงปิดแต่ละเส้นจะถูกหาจุดแบ่งแยกลายเส้นหรือจุดมุมโดยการประสาน (Convolution) เส้นขอบวงปิดด้วยตัวกรองเกาซ์เซียน (Gaussian filter) และหาค่าความโค้ง (Curvature) บนลายเส้น เมื่อได้จุดแบ่งแยกลายเส้นจะทำการวิเคราะห์ลักษณะลายเส้นแต่ละเส้นที่เป็นองค์ประกอบของเส้นขอบวงปิดว่าเป็นเส้นตรงหรือเส้นโค้ง โดยการทำโค้งกระชับ (Curve fitting) ขั้นตอนสุดท้ายคือการจำแนกชนิดของโลจิกเกทตามโครงสร้างหลัก 4 ชนิด ดังแสดงในรูปที่ 3.1 เมื่อผ่านการวิเคราะห์โครงสร้างของโลจิกเกทแล้วจะทำให้รู้ตำแหน่งและชนิดของโลจิกเกท (ตามโครงสร้างหลัก) ทุกตัวในภาพวงจร รายละเอียดของแต่ละขั้นตอนมีดังต่อไปนี้



รูปที่ 3.2 ขั้นตอนการวิเคราะห์โครงสร้างของโลจิกเกท

3.1 การปรับปรุงคุณภาพของลายเส้น

ภาพวงจรโลจิกเกทที่ต้องการจดจำเป็นภาพวงจรที่เกิดขึ้นจากการเขียนด้วยมือ โดยใช้อุปกรณ์เครื่องเขียนที่มีใช้อยู่ทั่วไป เช่น ปากกา และดินสอ เป็นต้น ดังนั้นลายเส้นในภาพแต่ละภาพ

หรือแม้แต่ภาพเดียวกันจะมีความหนาของลายเส้นแตกต่างกัน โดยขึ้นอยู่กับขนาดของปลายปากกาหรือดินสอที่ใช้ และการลงน้ำหนักขณะลากเส้นในการเขียนภาพวงจร รวมทั้งกระบวนการรับภาพเข้าสู่คอมพิวเตอร์โดยใช้สแกนเนอร์ (Scanner) จะมีการเปลี่ยนแปลงข้อมูลภาพ เช่น การย่อหรือขยายภาพ และการแปลงข้อมูลภาพเป็นภาพสองระดับ (Binary image) เป็นต้น ซึ่งอาจจะเกิดสัญญาณรบกวนในภาพได้ นอกจากนี้ภาพวงจรที่ได้ถ้าลายเส้นมีความหนามากกว่าหรือเท่ากับ 3 จุดภาพ และมีลายเส้นที่สมบูรณ์ไม่มีช่องว่าง (Hold) เกิดขึ้นบนลายเส้นการประมวลผลเพื่อแยกตัวโลจิกเกทออกจากวงจรและการวิเคราะห์จะไม่เกิดปัญหา อย่างไรก็ตามถ้าลายเส้นมีความหนา 1 หรือ 2 จุดภาพ หรือลายเส้นไม่สมบูรณ์มีช่องว่างบนลายเส้นจะไม่สามารถแยกตัวโลจิกเกทออกจากวงจรได้ เนื่องจากเมื่อทำการหาลายเส้นโครงร่างของภาพจะพบจุดแยกหรือจุดต่อร่วมที่ไม่ถูกต้องเกิดขึ้นบนลายเส้นโครงร่างดังรูปที่ 3.3 ซึ่งมีผลทำให้การวิเคราะห์โครงสร้างและการต่อวงจรของโลจิกเกทผิดพลาด ดังนั้นจึงจำเป็นต้องมีการปรับปรุงความสมบูรณ์ของลายเส้นเพื่อป้องกันปัญหาดังกล่าวข้างต้น

(ก) ข้อมูลอินพุท (ข) การแยกโลจิกเกทออกจากวงจร (ค) ลายเส้นโครงร่างของภาพ

รูปที่ 3.3 ข้อผิดพลาดที่เกิดขึ้นจากลายเส้นไม่สมบูรณ์

การเลือกใช้วิธีในการปรับปรุงลายเส้นเพื่อแก้ปัญหาข้างต้นจำเป็นต้องคำนึงถึงการสูญเสียข้อมูลและคุณสมบัติโทโปโลยี (Topology) ของภาพหลังการปรับปรุงลายเส้นด้วย ดังนั้นในงานวิจัยนี้จึงประยุกต์ใช้วิธีการทำลายเส้นให้บาง 1 รอบ และการทำลายเส้นให้เบลอ 1 รอบ เนื่องจากการทำลายเส้นให้บาง 1 รอบ เป็นการลดความหนาของลายเส้นลงน้อยกว่าหรือเท่ากับ 2 จุดภาพ และยังเป็นวิธีการที่จำเป็นต้องใช้ในขั้นตอนการหาโครงร่างของภาพด้วย ส่วนการทำลายเส้นให้เบลอเป็นการเพิ่มความหนาของลายเส้นขึ้น 2 จุดภาพ ดังนั้นภาพวงจรที่มีความหนาของลายเส้นมากกว่าหรือเท่ากับ 3 จุดภาพ เมื่อผ่านขั้นตอนทั้งสองนี้ข้อมูลภาพจะถูกเปลี่ยนแปลงน้อยมาก แต่ถ้าภาพวงจรมีความหนาของลายเส้นน้อยกว่า 3 จุดภาพ เมื่อผ่านขั้นตอนนี้ลายเส้นจะมีความหนา 3 จุดภาพเสมอ นอกจากนี้การทำลายเส้นให้เบลอยังช่วยลดรูปร่างปลีกย่อยที่เกิดจากสัญญาณรบกวน

บริเวณขอบของลายเส้น และลดช่องว่างบนลายเส้นที่มีขนาดน้อยกว่า 9 จุดภาพได้ จึงทำให้ลายเส้นในภาพมีความสมบูรณ์มากขึ้น ดังแสดงไว้ในรูปที่ 3.6 รายละเอียดของวิธีการทั้งสองมีดังต่อไปนี้

3.1.1 การทำลายเส้นให้บาง (Thinning algorithm)

ในงานทางด้านการประมวลผลภาพ (Digital image processing) การทำลายเส้นให้บางเป็นขั้นตอนพื้นฐานเริ่มแรกวิธีหนึ่งที่ถูกใช้สำหรับการวิเคราะห์รูปแบบหรือโครงสร้าง เช่น การรู้จำลายนิ้วมือ (Fingerprint recognition) การรู้จำตัวอักษร (Optical character recognition หรือ OCR) และการจดจำภาพวัตถุในระบบการมองเห็นของหุ่นยนต์ เป็นต้น ภาพลายเส้นที่ได้หลังจากการทำลายเส้นให้บางกับภาพสองระดับ (Binary image) จะมีความหนาของลายเส้น 1 จุดภาพ และถูกเรียกว่าลายเส้นโครงร่าง (Skeleton) ประโยชน์อย่างหนึ่งของการทำลายเส้นให้บางคือ การลดขนาดของหน่วยความจำที่ใช้ในการเก็บข้อมูลโครงสร้างของรูปแบบ และลดความซับซ้อนในการวิเคราะห์รูปแบบนั้น การทำลายเส้นให้บางมีหลายวิธี เช่น วิธีของ Zhang and Suen วิธีของ A.DATTA and S.K.PARUI และวิธีของ BEN-KWEI JAND and ROLAND T.CHIN ซึ่งแต่ละวิธีจะมีข้อดีข้อด้อยแตกต่างกัน ในงานวิจัยนี้เลือกใช้วิธีการทำลายเส้นให้บางของ Zhang and Suen [14] เนื่องจากมีขั้นตอนไม่ซับซ้อน และใช้เวลาในการประมวลผลน้อย โดยยังคงรักษาคุณสมบัติโทโปโลยี (Topology) ไว้ [18] รายละเอียดของวิธีการมีดังนี้

ขั้นตอนการทำลายเส้นให้บางประกอบด้วย 2 ขั้นตอนที่กระทำกับจุดขอบ (Edge point) ของลายเส้นในภาพโดยจุดขอบของลายเส้นเป็นจุดที่มีค่าเป็น "1" และมีจุดรอบข้างอย่างน้อย 1 จุดที่มีค่าเป็น "0" เมื่อกำหนดให้ Tp เป็นจุดบนลายเส้นที่ทำการตรวจสอบและมีจุดรอบข้างดังแสดงในรูปที่ 2.2 (ก) เงื่อนไขการกำจัดจุดตรวจสอบ Tp ของแต่ละขั้นตอนมีดังนี้

<u>ขั้นตอนที่ 1 เงื่อนไขการกำจัดจุดขอบด้านบนและด้านซ้ายมือ</u>

1) $2 \le N(Tp) \le 6$
2) $S(Tp) = 1$
3) $\overline{(P1 \,.\, P3 \,.\, P7)} = 1$
4) $\overline{(P1 \,.\, P5 \,.\, P7)} = 1$

<u>ขั้นตอนที่ 2 เงื่อนไขการกำจัดจุดขอบด้านล่างและด้านขวามือ</u>

1) $2 \le N(Tp) \le 6$
2) $S(Tp) = 1$
3) $\overline{(P1 \,.\, P3 \,.\, P5)} = 1$
4) $\overline{(P3 \,.\, P5 \,.\, P7)} = 1$

เมื่อ N(Tp) คือจำนวนจุดรอบข้าง Tp ที่มีค่าเป็น 1 นั่นคือ

$$N(Tp) = \sum_{i=1}^{8} Pi \qquad (3\text{-}1)$$

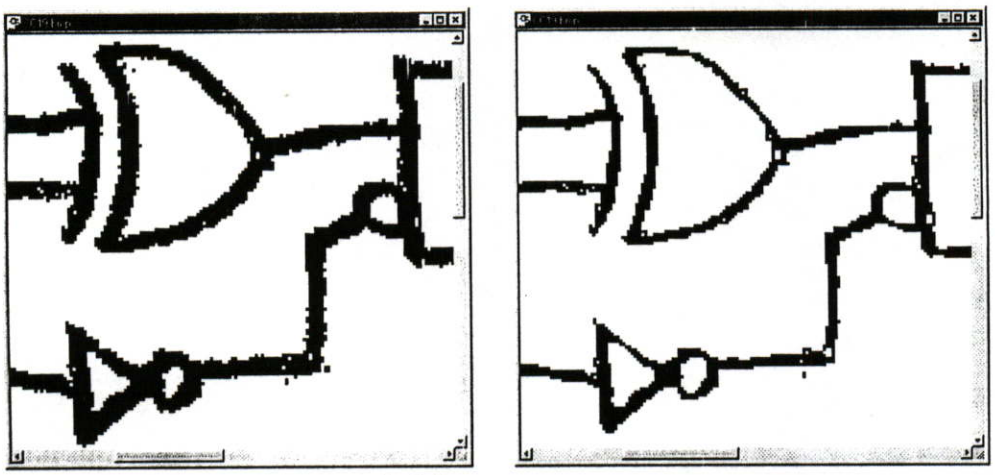
S(Tp) คือ จำนวนครั้งของการเปลี่ยนแปลงจากค่า "0" เป็น "1" ของจุดรอบข้าง โดยพิจารณาเริ่มจาก P1, P2, P3, ..., P8, P1 ตามลำดับ

และ (.) คือ ตรรกะ AND

ในการประมวลผลจะกระทำทีละขั้นตอนโดยพิจารณาจุดภาพ Tp ที่เป็นลายเส้นทีละจุด และตรวจสอบตามเงื่อนไขแต่ละขั้นตอน ถ้าจุดภาพ Tp ตรงตามเงื่อนไขของขั้นตอนที่ 1 หรือขั้นตอนที่ 2 ทุกข้อ จุดภาพ Tp จะถูกทำเครื่องหมาย (flag) และเมื่อตรวจสอบครบทั้งภาพจุดภาพที่ถูกทำเครื่องหมายไว้จะถูกลบออก (เปลี่ยนเป็นค่า "0") การประมวลผลขั้นตอนที่ 1 และขั้นตอนที่ 2 หนึ่งรอบจะกำจัดจุดขอบของลายเส้นทั้ง 4 ด้าน คือ ขอบด้านบน, ล่าง, ซ้ายมือ และขวามือ เมื่อประมวลผลซ้ำหลายๆ รอบจนกระทั่งไม่มีการจำกัดจุดขอบของลายเส้นก็จะได้ลายเส้นโครงร่างของภาพ อัลกอริทึ่มของการทำลายเส้นให้บางแสดงในอัลกอริทึ่มที่ 3.1 ภาพก่อนและหลังการทำลายเส้นให้บาง 1 รอบ แสดงดังรูปที่ 3.4

```
For y = 0 to Image Height
    For x = 0 to Image Width
        If (Pixel(x,y) = 1)
            If ((p1&p3&p7) = 0)&((p1&p5&p7) = 0)
                NTp = p1+p2+...+p8
                STp = (p1^p2)+(p2^p3)+...+(p8^p1)
                If (2 ≤ NTp ≤ 6)&(STp = 2)
                    Flag the Pixel(x,y).
For y = 0 to Image Height
    For x = 0 to Image Width
        If (Pixel(x,y) = flag)
            Pixel(x,y) = 0
For y = 0 to Image Height
    For x = 0 to Image Width
        If (Pixel(x,y) = 1)
            If ((p1&p3&p5) = 0)&((p3&p5&p7) = 0)
                NTp = p1+p2+...+p8
                STp = (p1^p2)+(p2^p3)+...+(p8^p1)
                If (2 ≤ NTp ≤ 6)&(STp = 2)
                    Flag the Pixel(x,y).
For y = 0 to Image Height
    For x = 0 to Image Width
        If (Pixel(x,y) = flag)
            Pixel(x,y) = 0
```

อัลกอริทึมที่ 3.1 การทำลายเส้นให้บาง



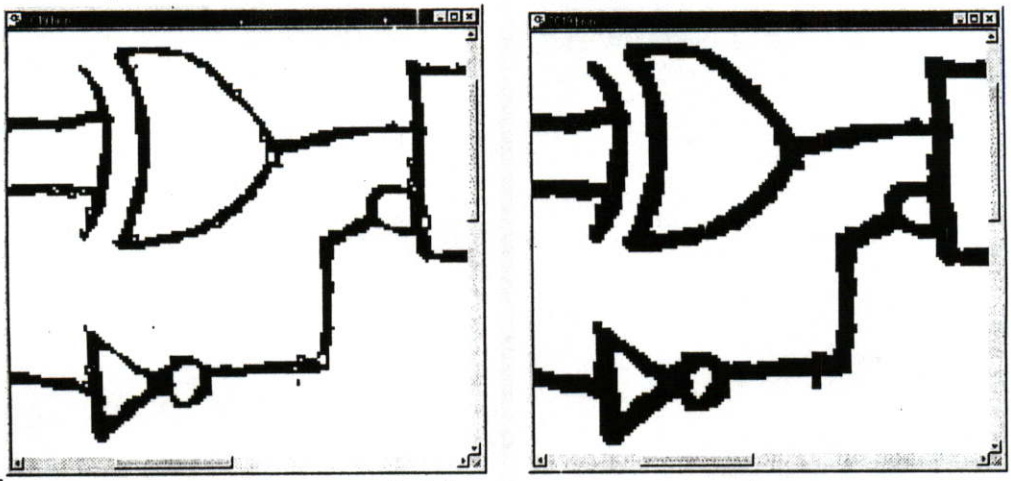
รูปที่ 3.4 ภาพก่อนและหลังการทำลายเส้นให้บาง 1 รอบ

3.1.2 การทำลายเส้นให้เบลอ (Blur algorithm)

การทำลายเส้นให้เบลอโดยทั่วไปถูกใช้ในการเพิ่มขนาดของวัตถุในภาพ และปรับปรุงข้อมูลภาพเนื่องจากเป็น Software filter [15] การทำลายเส้นให้เบลอสำหรับภาพ 2 ระดับ คือ การทำให้ความหนาของลายเส้นเพิ่มขึ้นโดยการเพิ่มจุดภาพ (ทำให้จุดภาพมีค่าเป็น "1") บริเวณขอบของลายเส้น ขั้นตอนการทำลายเส้นให้เบลอจะประมวลผลกับจุดภาพรอบข้างของจุดภาพที่เป็นลายเส้น (จุดภาพที่มีค่าเป็น "1") ทุกจุดในภาพ กล่าวคือ จะตรวจหาจุดภาพที่เป็นลายเส้นและทำเครื่องหมายกับจุดภาพรอบข้างที่มีค่าเป็น "0" จนกระทั่งครบทั้งภาพจากนั้นเปลี่ยนจุดที่ทำเครื่องไว้ทั้งหมดให้เป็นจุดภาพลายเส้น (เปลี่ยนจุดภาพเป็น "1") อัลกอริทึมของการทำลายเส้นให้เบลอแสดงในอัลกอริทึมที่ 3.2 ในกระบวนการทำลายเส้นให้เบลอที่เรียกว่า "เบลอระดับที่ 1" หมายถึงการทำลายเส้นให้เบลอตลอดทั้งภาพ 1 รอบ และ "เบลอระดับที่ 2" หมายถึงการทำลายเส้นให้เบลอตลอดทั้งภาพ 2 รอบ เป็นต้น ดังนั้นเมื่อระดับการเบลอเพิ่มขึ้นรูปร่างปลีกย่อยจะค่อยๆ หายไปมากขึ้น ตัวอย่างของผลการทำลายให้เบลอระดับที่ 1 แสดงในรูปที่ 3.5

```
For y = 0 to Image Height
    For x = 0 to Image Width
        If (Pixel(x,y) = 1)
            If (p1 = 0)
                Flag the Pixel(p1)
            If (p2 = 0)
            .           .
            .           .
            .           .
            If (p8 = 0)
                Flag the Pixel(p8)
For y = 0 to Image Height
    For x = 0 to Image Width
        If (Pixel(x,y) = flag)
            Pixel(x,y) = 1
```

อัลกอริทึมที่ 3.2 การทำลายเส้นให้เบลอ

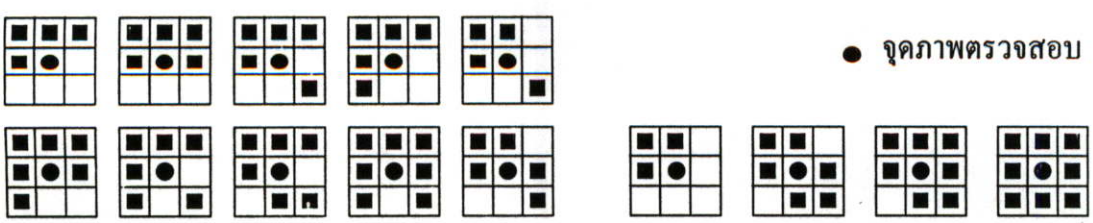


รูปที่ 3.5 ภาพก่อนและหลังการทำลายเส้นให้เบลอ

เมื่อนำภาพวงจรโลจิกเกทที่เขียนด้วยมือมาผ่านขั้นตอนการปรับปรุงลายเส้นแล้ว ลายเส้นที่ได้จะมีความหนามากกว่าหรือเท่ากับ 3 จุดภาพเสมอ ดังรูปที่ 3.5 ซึ่งจะทำให้สามารถแยกตัวโลจิกเกทออกจากวงจรในขั้นตอนต่อไปได้อย่างสมบูรณ์

3.2 การแยกโลจิกเกทออกจากวงจร

ในการประมวลผลภาพวิธีการพื้นฐานในการลดข้อมูลเพื่อการวิเคราะห์โครงสร้างและการรู้จำวัตถุในภาพ วิธีการหนึ่งที่นิยมใช้กันทั่วไปคือ การหาขอบของภาพ ซึ่งนอกจากจะเป็นการลดข้อมูลของการประมวลผลแล้ว ยังช่วยลดความซับซ้อนในการวิเคราะห์ เนื่องจากลายเส้นขอบของภาพจะมีความหนาเพียง 1 จุดภาพ และถ้าลายเส้นในภาพมีความหนามากกว่าหรือเท่ากับ 3 จุดภาพ ลายเส้นขอบที่ได้จะไม่มีจุดต่อจุดแยกเกิดขึ้น ดังนั้นจึงสามารถนำมาใช้แยกตัวโลจิกเกทออกจากวงจรเพื่อวิเคราะห์โครงสร้างได้ ภาพวงจรโลจิกเกทที่ใช้เป็นข้อมูลภาพสองระดับและผ่านการปรับปรุงลายเส้นด้วยวิธีการดังกล่าวข้างต้นแล้ว จะมีรูปแบบของจุดภาพลายเส้นและจุดภาพรอบข้างเกิดขึ้นในภาพทั้งหมด 14 รูปแบบดังแสดงในรูปที่ 3.6 ซึ่งรูปแบบจุดภาพที่เป็นจุดขอบของภาพมี 10 รูปแบบแสดงในรูปที่ 3.6 (ก) ส่วนรูปที่ 3.6 (ข) เป็นจุดภาพบนลายเส้นทั่วไปที่ไม่ใช่จุดขอบภาพ ดังนั้นเราสามารถกำหนดเงื่อนไขเพื่อแยกจุดขอบของภาพได้ดังนี้



(ก) รูปแบบจุดขอบภาพ (ข) รูปแบบจุดภาพลายเส้นทั่วไป

รูปที่ 3.6 รูปแบบของจุดภาพที่เกิดขึ้นหลังการปรับปรุงคุณภาพของลายเส้น

เงื่อนไขจุดขอบของภาพ

1) $1 < (P1 + P3 + P5 + P7) < 4$

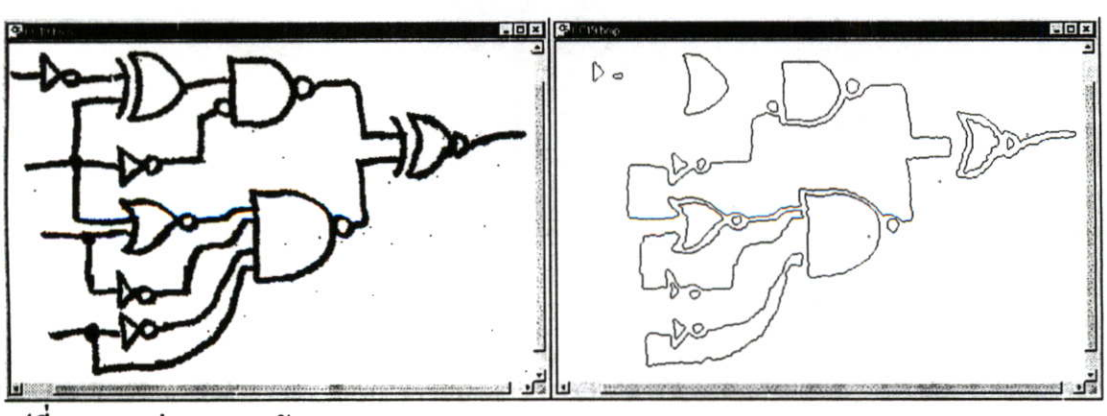
2) $1 < (P2 + P4 + P6 + P8)$

ขั้นตอนการหาขอบของภาพ

ขั้นตอนการหาขอบของภาพสามารถทำได้โดยค้นหาจุดภาพที่เป็นลายเส้น (จุดภาพที่มีค่าเป็น "1") และตรวจสอบเงื่อนไขข้างต้น ถ้าตรงตามเงื่อนไขทั้งสองข้อ ให้ทำเครื่องหมายที่จุดภาพนั้นและตรวจสอบจุดภาพอื่นๆ ต่อไปจนครบทั้งภาพ จากนั้นเปลี่ยนจุดภาพที่ทำเครื่องหมายไว้เป็นจุดขอบของภาพ (โดยกำหนดให้มีค่าเป็น "2" หรือค่าอื่นๆ ที่ไม่ใช่ค่า "1" และค่า "0" ทั้งนี้เพื่อไม่ให้เกิดความสับสนในการประมวลผล) อัลกอริทึมสำหรับการหาขอบของภาพแสดงไว้ในอัลกอริทึมที่ 3.3 ส่วนรูปที่ 3.7 เป็นผลของการหาขอบของภาพ ซึ่งจะเป็นลายเส้นขอบภาพที่มีลักษณะเป็นวงปิดที่เรียกว่าเส้นขอบวงปิด โดยมีทั้งเส้นขอบวงปิดของตัวโลจิกเกทและเส้นขอบวงปิดของลายวงจร เส้นขอบวงปิดที่ได้ทั้งหมดนี้จะถูกนำไปวิเคราะห์หาจุดแบ่งแยกลายเส้นในขั้นตอนต่อไป ข้อดีของอัลกอริทึมการหาขอบของภาพด้วยวิธีนี้คือจะให้ลายเส้นขอบที่มีความหนาเพียงหนึ่งจุดภาพ

```
For y = 0 to Image Height
    For x = 0 to Image Width
        If (Pixel(x,y) = 1)
            If (1 < (p1+p3+p5+p7) < 4)&((p2+p4+p6+p8) > 1)
                Flag the Pixel(x,y).
For y = 0 to Image Height
    For x = 0 to Image Width
        If (Pixel(x,y) = flag)
            Pixel(x,y) = Edge
```

อัลกอริทึมที่ 3.3 การหาขอบของภาพ



รูปที่ 3.7 ภาพก่อนและหลังการหาขอบของภาพ

3.3 การหาจุดแบ่งแยกลายเส้น

การหาจุดแบ่งแยกลายเส้นหรือจุดมุม (Corner) มีด้วยกันหลายวิธี เช่น การหาจุดมุมจาก Chain-code [9] การหาจุดมุมจากสมการหาค่าความโค้งทั่วไป (นำเสนอในหัวข้อที่ 2.2.2) และการหาจุดเด่นบนความโค้งด้วยตัวกรองเกาซ์เซียน [13] เป็นต้น จากการศึกษาและทดลอง การหาจุดมุมด้วยตัวกรองเกาซ์เซียนเป็นวิธีหนึ่งที่สามารถหาจุดมุมได้อย่างมีประสิทธิภาพ ดังนั้นในวิทยานิพนธ์ฉบับนี้จึงใช้วิธีการหาจุดมุมด้วยตัวกรองเกาซ์เซียน ซึ่งสามารถทำได้โดยประสานลายเส้นด้วยตัวกรองเกาซ์เซียน ทำให้ลายเส้นมีความราบเรียบ (Smooth) กำจัดสัญญาณรบกวนบนลายเส้น จากนั้นจึงหาค่าความโค้งบนลายเส้นและหาจุดมุม ณ ตำแหน่งที่มีค่าความโค้งสูงสุด รายละเอียดของการหาจุดมุมด้วยตัวกรองเกาซ์เซียนมีดังนี้

3.3.1 การประสานเส้นขอบวงปิดด้วยตัวกรองเกาซ์เซียน

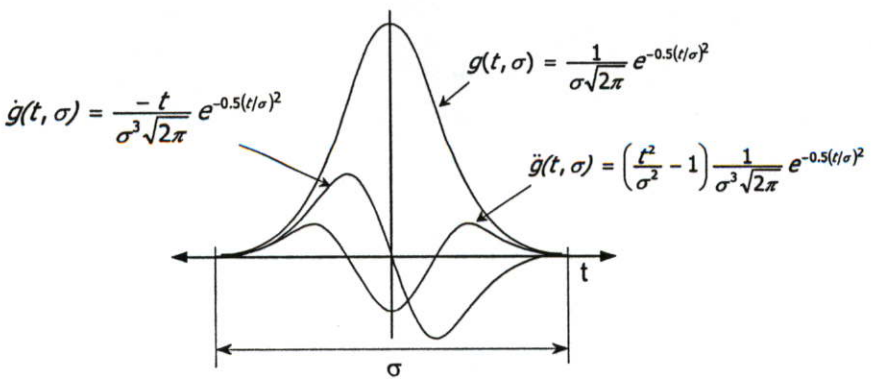
สมการเกาซ์เซียนเป็นสมการที่มีคุณสมบัติเป็นตัวกรองความถี่โดยมีค่าสเกล-สเปซ (Scale-Space) เป็นตัวควบคุมคุณสมบัติของการกรองความถี่ กล่าวคือ ถ้าค่าสเกล-สเปซมีค่าน้อยตัวกรองจะมีคุณสมบัติยอมให้ความถี่สูงผ่านไปได้ แต่ในทางตรงกันข้ามค่าสเกล-สเปซมีค่ามากตัวกรองก็จะมีคุณสมบัติเป็นตัวกดความถี่สูง โดยสมการของตัวกรองเกาซ์เซียนถูกนิยามไว้ดังนี้

$$g(t,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-0.5(t/\sigma)^2} \qquad (3\text{-}2)$$

เมื่อ σ คือ ค่าสเกล-สเปซของตัวกรองเกาซ์เซียน หรือค่าเบี่ยงเบนมาตราฐาน (Standard deviation)

t คือ ตัวแปรอิสระ

อนุพันธ์ย่อยอันดับที่หนึ่งและอันดับที่สองของสมการเกาซ์เซียน สามารถหาได้จากสมการที่ (3-3) และ (3-4) ตามลำดับ และมีเส้นกราฟของสมการแสดงไว้ในรูปที่ 3.8



รูปที่ 3.8 เส้นกราฟของสมการตัวกรองเกาซ์เซียน

$$\dot{g}(t,\sigma) = \frac{\partial g(t,\sigma)}{\partial t} = \frac{-t}{\sigma^3\sqrt{2\pi}} e^{-0.5(t/\sigma)^2} \quad (3\text{-}3)$$

$$\ddot{g}(t,\sigma) = \frac{\partial^2 g(t,\sigma)}{\partial t^2} = \left(\frac{t^2}{\sigma^2} - 1\right)\frac{1}{\sigma^3\sqrt{2\pi}} e^{-0.5(t/\sigma)^2} \quad (3\text{-}4)$$

เนื่องจากเส้นขอบวงปิดที่ปรากฏบนหน้าจอคอมพิวเตอร์เป็นจุดภาพหลายๆ จุดวางเรียงต่อกัน ซึ่งจุดภาพเหล่านี้จะถูกเก็บตำแหน่งด้วยเลขจำนวนเต็ม ดังนั้นการบอกตำแหน่งของจุดภาพบนเส้นขอบวงปิดจึงสามารถบอกแทนได้ด้วยสมการแบบไม่ต่อเนื่อง (Discrete function) ดังนี้

$$C = f(x, y) \quad (3\text{-}5)$$

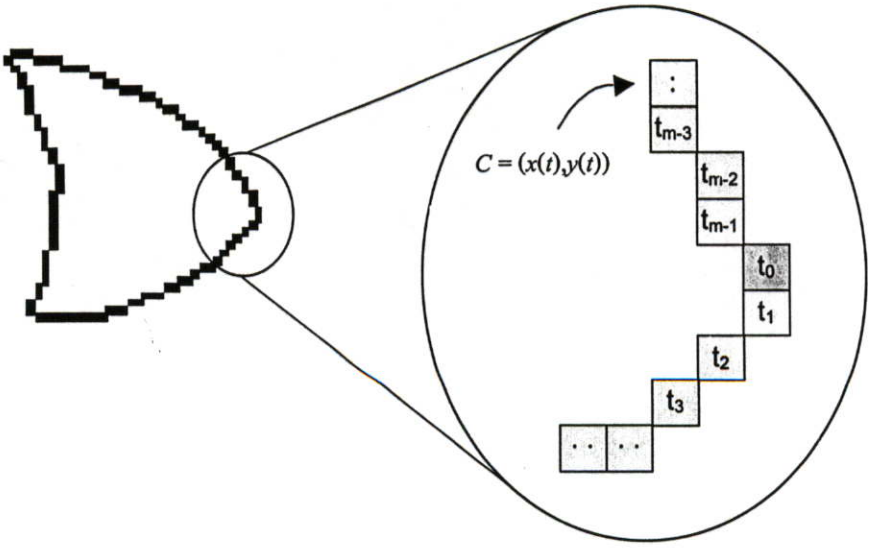
เมื่อ C คือเส้นขอบวงปิด

x,y คือพิกัดในแนวแกน x และ y ของตำแหน่งจุดภาพ

เพื่อให้สอดคล้องต่อหลักเกณฑ์การคำนวณทางคณิตศาสตร์และเหมาะสมต่อการใช้งาน สมการที่ (3-5) สามารถเขียนใหม่ให้อยู่ในรูปของสมการพาราเมตริกได้ดังนี้

$$C(t) = \big(x(t), y(t)\big) \quad (3\text{-}6)$$

เมื่อ t คือเส้นทางเดินของจุดภาพบนเส้นขอบวงปิด (แสดงในรูปที่ 3.8)



รูปที่ 3.9 วิธีการเก็บข้อมูลและการกำหนดเส้นทางเดินของจุดภาพบนเส้นขอบวงปิด

เมื่อนำตัวกรองเกาซ์เซียนมาประสาน (Convolution) กับเส้นขอบวงปิดจะได้สมการดังนี้

$$X(t,\sigma) = x(t) * g(t,\sigma)$$
$$Y(t,\sigma) = y(t) * g(t,\sigma) \quad (3\text{-}7)$$

เมื่อ x(t) และ y(t) คือ สมการพาราเมตริกของเส้นขอบวงปิด

สัญลักษณ์ * คือ การประสาน (Convolution)

เนื่องจากสมการของเส้นขอบวงปิดเป็นสมการไม่ต่อเนื่องดังที่กล่าวมาแล้ว แต่ตัวกรองเกาซ์เซียนที่แสดงในสมการที่ (3-2) เป็นสมการต่อเนื่อง ดังนั้นเพื่อให้สอดคล้องกับการประมวลผลตัวกรองเกาซ์เซียนจึงต้องถูกแซมพลิง (Sampling) ให้เป็นสมการแบบไม่ต่อเนื่องด้วย ดังแสดงในรูปที่ 3.10 ซึ่งโดยทั่วไปแล้วขนาดของหน้าต่างสำหรับเก็บค่าสัมประสิทธ์ของตัวกรองเกาซ์เซียนจะถูกกำหนดด้วยจำนวนแซมพลิง 2N+1 โดยที่ N สามารถคำนวณได้ดังนี้

$$N = Min\{N_1, N_2\} \quad (3\text{-}8)$$

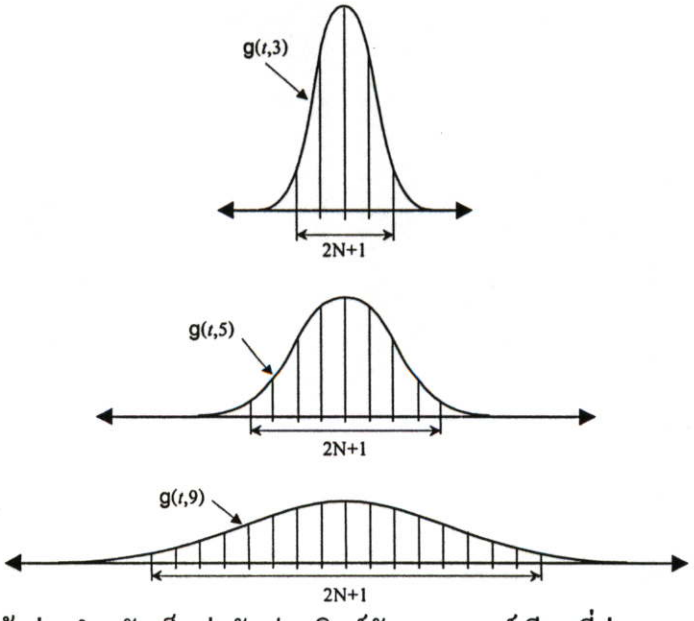
เมื่อ

$$N_1 = \left\{ n : Max \sum_{t=-n}^{n} \frac{1}{\sigma\sqrt{2\pi}} e^{-0.5(t/\sigma)^2} \le 0.99 \right\} \quad (3\text{-}8a)$$

และ

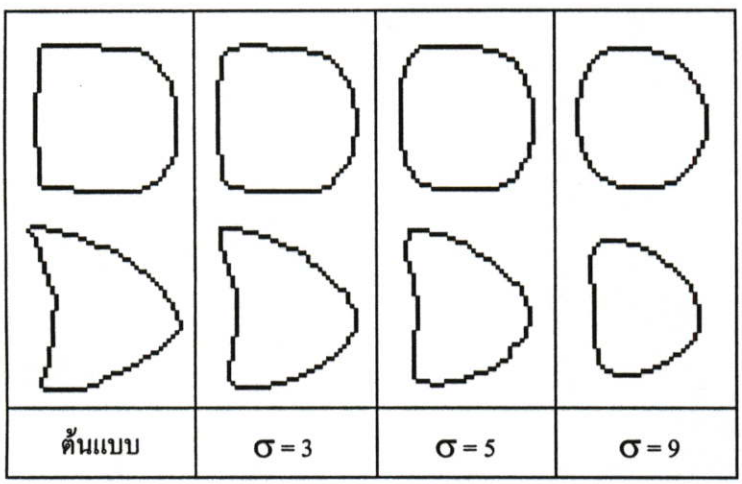
$$N_2 = \lfloor 5\sigma \rfloor \quad (3\text{-}8b)$$

เมื่อ $\lfloor a \rfloor$ หมายถึง เลขจำนวนเต็มที่มากที่สุดแต่น้อยกว่าหรือเท่ากับ a



รูปที่ 3.10 ขนาดหน้าต่างสำหรับเก็บค่าสัมประสิทธ์ตัวกรองเกาซ์เซียนที่ค่าสเกล-สเปซต่างๆ กัน

การประสานตัวกรองเกาซ์เซียนกับเส้นขอบวงปิด เมื่อเปลี่ยนแปลงค่าสเกล-สเปซจะมีผลทำให้ความราบเรียบ (Smooth) ของเส้นขอบวงปิดเปลี่ยนแปลงดังแสดงในรูปที่ 3.11 ซึ่งส่วนโค้งและส่วนเว้าที่เกิดจากสัญญาณรบกวนก็จะถูกกำจัดออก ขณะที่ส่วนโค้งและส่วนเว้าที่แท้จริงจะยังปรากฏอยู่ และสามารถคำนวณเป็นค่าตัวเลขได้ด้วยสมการหาค่าความโค้ง ซึ่งจะเป็นสิ่งกำหนดคุณลักษณะของ "จุดมุม" ส่วนโค้งและส่วนเว้าของเส้นขอบวงปิดที่สามารถนิยามให้เป็นจุดมุมได้นั้นจะต้องเป็นส่วนโค้งและส่วนเว้าที่มีค่าความโค้งสูงเมื่อเทียบกับบริเวณข้างเคียง โดยวิธีการคำนวณจะได้นำเสนอในหัวข้อถัดไป รูปที่ 3.11 แสดงคุณสมบัติของตัวกรองเกาซ์เซียน ซึ่งมีผลทำให้เกิดความราบเรียบของเส้นขอบแตกต่างกันที่ค่าสเกล-สเปซต่างกัน



รูปที่ 3.11 คุณสมบัติของตัวกรองเกาซ์เซียนในการกรองเส้นขอบวงปิดโดยมีค่าสกล-สเปซต่างๆ กัน

3.3.2 การหาค่าความโค้งบนเส้นขอบวงปิด

การคำนวณค่าความโค้งบนลายเส้นที่มีสมการเป็น $y = f(x)$ สามารถหาได้จากสมการดังนี้

$$K = \frac{\left|\frac{d^2y}{dx^2}\right|}{\left(1 + \left(\frac{dy}{dx}\right)^2\right)^{3/2}} \tag{3-9}$$

ในทำนองเดียวกันลายเส้นที่มีสมการอยู่ในรูปแบบสมการพาราเมตริก (สมการที่ (3-6)) ก็สามารถคำนวณได้เช่นกัน เมื่อกำหนดให้

$$\begin{aligned} x &= x(t) \\ y &= y(t) \end{aligned} \tag{3-10}$$

และอนุพันธ์ของ C(t) จะมีสมการดังนี้

$$\dot{C}(t) = \begin{pmatrix} \dot{x}(t) & \dot{y}(t) \end{pmatrix} \tag{3-11a}$$

ดังนั้นค่าความชันของลายเส้นสามารถหาได้จาก

$$y' = \frac{dy}{dx} = \frac{\frac{dy}{dt}}{\frac{dx}{dt}} = \frac{\dot{y}(t)}{\dot{x}(t)} \tag{3-11b}$$

และในทำนองเดียวกันอนุพันธ์อันดับที่สองของลายเส้น C(t) ก็สามารถคำนวณได้จาก

$$\ddot{C}(t) = [\ddot{x}(t) \quad \ddot{y}(t)] \tag{3-12a}$$

หรือ

$$\frac{d^2y}{dx^2} = \frac{\frac{dy'}{dt}}{\frac{dx}{dt}} = \frac{\dot{x}(t)\ddot{y}(t) - \dot{y}(t)\ddot{x}(t)}{(\dot{x}(t))^3} \tag{3-12b}$$

เพราะฉะนั้นสมการสำหรับหาค่าความโค้งของลายเส้นที่อยู่ในรูปแบบสมการพาราเมตริกสามารถเขียนใหม่ได้ดังนี้

$$K = \frac{\frac{\dot{x}(t)\ddot{y}(t) - \dot{y}(t)\ddot{x}(t)}{(\dot{x}(t))^3}}{\left(1 + \left(\frac{\dot{y}(t)}{\dot{x}(t)}\right)^2\right)^{3/2}} = \frac{\dot{x}(t)\ddot{y}(t) - \dot{y}(t)\ddot{x}(t)}{\left((\dot{x}(t))^2 + (\dot{y}(t))^2\right)^{3/2}} \tag{3-13}$$

เมื่อพิจารณาสมการที่ (3-13) จะพบว่าสมการดังกล่าวเป็นสมการหาค่าความโค้งของลายเส้นที่ยังไม่มีการประสานด้วยตัวกรองเกาซ์เซียน ดังนั้นเมื่อนำลายเส้นที่ถูกประสานด้วยตัวกรองเกาซ์เซียนไปแทนในสมการหาค่าความโค้งดังกล่าว จะสามารถเขียนสมการหาค่าความโค้งใหม่ได้ดังนี้

$$K(t,\sigma) = \frac{\dot{X}(t,\sigma)\ddot{Y}(t,\sigma) - \dot{Y}(t,\sigma)\ddot{X}(t,\sigma)}{\left((\dot{X}(t,\sigma))^2 + (\dot{Y}(t,\sigma))^2\right)^{3/2}} \tag{3-14}$$

เมื่อ

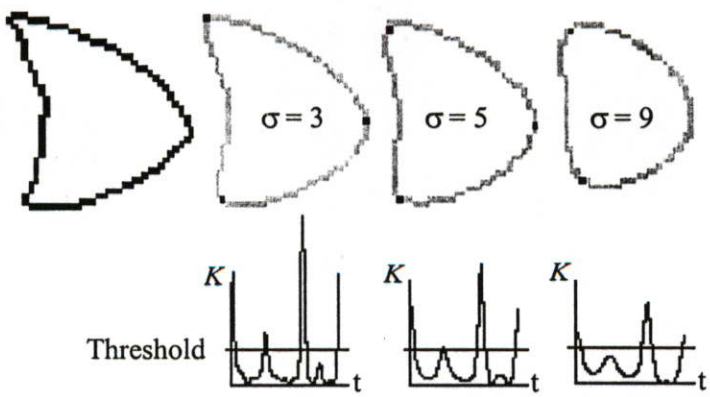
$$\dot{X}(t,\sigma) = \frac{\partial X(t,\sigma)}{\partial t} \qquad \dot{Y}(t,\sigma) = \frac{\partial Y(t,\sigma)}{\partial t}$$
$$= x(t) * \frac{\partial g(t,\sigma)}{\partial t} \qquad = y(t) * \frac{\partial g(t,\sigma)}{\partial t} \qquad \text{(3-15a)}$$
$$= x(t) * \dot{g}(t,\sigma) \qquad = y(t) * \dot{g}(t,\sigma)$$

$$\ddot{X}(t,\sigma) = \frac{\partial \dot{X}(t,\sigma)}{\partial t} \qquad \ddot{Y}(t,\sigma) = \frac{\partial \dot{Y}(t,\sigma)}{\partial t}$$
$$= x(t) * \frac{\partial \dot{g}(t,\sigma)}{\partial t} \qquad = y(t) * \frac{\partial \dot{g}(t,\sigma)}{\partial t} \qquad \text{(3-15b)}$$
$$= x(t) * \ddot{g}(t,\sigma) \qquad = y(t) * \ddot{g}(t,\sigma)$$

ค่าความโค้งที่ได้จากการคำนวณนี้จะเป็นสิ่งกำหนด "จุดมุม" ที่ปรากฏบนเส้นขอบวงปิด โดยจุดใดที่มีค่าความโค้งสูงเท่านั้นจึงจะสามารถกำหนดให้เป็นจุดมุมได้ ลำดับขั้นของการประมวลผลเพื่อหาจุดมุมบนเส้นขอบวงปิดมีดังนี้

<u>ขั้นตอนการประมวลผล</u>

1) กำหนดค่าสเกล-สเปซ
2) คำนวณขนาดหน้าต่างของตัวกรองเกาซ์เซียนสำหรับเก็บค่าสัมประสิทธ์ ซึ่งสามารถคำนวณได้จาก 2N+1 ในสมการที่ (3-8)
3) คำนวณหาค่า $\dot{X}(t,\sigma)$, $\ddot{X}(t,\sigma)$, $\dot{Y}(t,\sigma)$ และ $\ddot{Y}(t,\sigma)$ ของจุดภาพทุกจุดบนเส้นขอบวงปิดจากสมการที่ (3-15a) และสมการที่ (3-15b)
4) คำนวณหาค่าความโค้ง $K(t,\sigma)$ ของจุดภาพทุกจุดบนเส้นขอบวงปิดจากสมการที่ (3-14)
5) กำหนดจุดมุม ณ จุดที่มีค่าความโค้งสูงกว่าค่า Threshold ที่กำหนดและมีค่าสูงที่สุดในช่วง



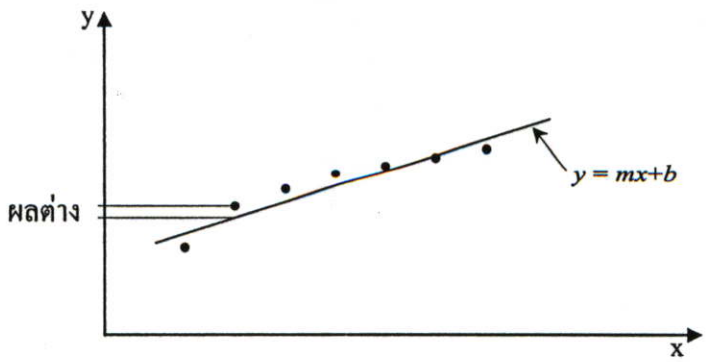
รูปที่ 3.12 ผลของการหาจุดมุมเมื่อเปลี่ยนแปลงค่าสเกล-สเปซ

รูปที่ 3.12 เป็นผลการทดลองหาจุดมุมบนเส้นขอบวงปิดที่ผ่านการประสานด้วยตัวกรองเกาซ์เซียนที่มีค่าสเกล-สเปซต่างๆ กัน ซึ่งการใช้งานจริงจะกำหนดค่าสเกล-สเปซคงที่และคำนวณหาจุดมุมของเส้นขอบวงปิดทุกเส้นในภาพ โดยเส้นขอบวงปิดใดที่มีจุดมุม 2, 3 และ 4 จุดจะถูกพิจารณาว่าเป็นโลจิกเกท และจะถูกนำไปวิเคราะห์ลักษณะลายเส้น เพื่อแบ่งแยกออกเป็นลายเส้นตรงหรือเส้นโค้งในขั้นตอนต่อไป นอกเหนือจากนี้จะถูกพิจารณาว่าเป็นส่วนประกอบของอินพุท-เอาท์พุท (วงกลม NOT) หรือส่วนประกอบของลายวงจร ซึ่งจะเก็บไว้ตรวจสอบในขั้นตอนการหาอินพุท-เอาท์พุทต่อไป

3.4 การวิเคราะห์ลักษณะลายเส้น

การวิเคราะห์ลักษณะลายเส้น เป็นการวิเคราะห์ลักษณะเส้นตรงหรือเส้นโค้งของลายเส้นที่อยู่ระหว่างจุดมุม 2 จุดที่เป็นองค์ประกอบของเส้นขอบวงปิด ลายเส้นดังกล่าวก็คือ กลุ่มของจุดภาพที่สามารถหาสมการทางคณิตศาสตร์มาแทนได้ โดยวิธีการที่เรียกว่า การทำโค้งกระชับ (Curve fitting) ซึ่งมีหลายวิธีและมีสมการหลายรูปแบบ ในการประยุกต์ใช้การทำโค้งกระชับมาวิเคราะห์ลักษณะลายเส้นว่าเป็นเส้นตรงหรือไม่นั้นสามารถทำได้โดย วิธีกำลังสองน้อยสุด (Least squares method) และใช้รูปแบบสมการเส้นตรง วิธีการดังกล่าวจะเป็นการหาสมการเส้นตรงจากกลุ่มของจุดภาพระหว่างจุดมุมสองจุด จากนั้นคำนวณหาค่าความผิดพลาดระหว่างลายเส้นที่คำนวณได้จากสมการกับกลุ่มจุดภาพ ถ้าค่าความผิดพลาดนี้เป็นศูนย์หรือมีค่าน้อยมาก กลุ่มของจุดภาพนั้นจะเป็นเส้นตรง ในทางตรงกันข้ามค่าความผิดพลาดมีค่ามากกลุ่มจุดภาพจะเป็นเส้นโค้ง ด้วยเหตุนี้จึงสามารถนำค่าความผิดพลาดดังกล่าวมาวิเคราะห์ลักษณะเส้นตรงและเส้นโค้งได้

การทำโค้งกระชับด้วยวิธีกำลังสองน้อยสุดโดยใช้รูปแบบสมการเส้นตรง เป็นการสร้างสมการเส้นตรงที่ดีที่สุด (Best fit) สำหรับกลุ่มของจุดภาพดังรูปที่ 3.13 โดยพิจารณาผลต่างระหว่างค่า y ของกลุ่มจุดภาพกับค่า y ที่คำนวณจากสมการ $y = mx+b$ ที่สร้างขึ้น ซึ่งผลต่างนี้เรียกว่า ค่าคงเหลือ (Residual)



รูปที่ 3.13 ภาพเปรียบเทียบกลุ่มจุดภาพกับเส้นตรงที่ได้จากสมการ

เมื่อจุดภาพมีตำแหน่ง (x_i,y_i) ค่าของ $y_1, y_2, y_3, \ldots, y_n$ สามารถหาได้ดังนี้

$$\begin{aligned} y_1 &= mx_1 + b \\ y_2 &= mx_2 + b \\ y_3 &= mx_3 + b \\ &\vdots \\ &\vdots \\ y_n &= mx_n + b \end{aligned} \tag{3-16}$$

ดังนั้นค่าคงเหลือของจุดภาพที่ตำแหน่ง $(x_1,y_1), (x_2,y_2), \ldots, (x_n,y_n)$ จะเป็น

$$y_1 - (mx_1 + b), y_2 - (mx_2 + b), \ldots, y_n - (mx_n + b) \tag{3-17}$$

ค่าคงเหลือที่ได้แต่ละตำแหน่งจะมีทั้งค่าบวกและค่าลบ จึงทำให้ผลรวมของค่าคงเหลือที่มีค่าน้อยไม่สามารถบ่งบอกได้ว่าสมการที่หาได้จะดีที่สุดสำหรับกลุ่มของจุดภาพเหล่านั้น ดังนั้นจะต้องทำค่าคงเหลือของแต่ละจุดให้เป็นบวกก่อนโดยการยกกำลังสองแล้วจึงหาผลรวม สมการหาผลรวมของค่าคงเหลือยกกำลังสองมีดังนี้

$$R = (y_1 - (mx_1 + b))^2 + (y_2 - (mx_2 + b))^2 + \ldots + (y_n - (mx_n + b))^2 \tag{3-18}$$

สมการที่ดีที่สุดสำหรับกลุ่มของจุดภาพจะต้องมีค่าผลรวมของค่าคงเหลือยกกำลังสองน้อยที่สุด จากสมการที่ (3-18) ค่าผลรวม R ประกอบด้วยตัวไม่รู้ค่า b และ m ซึ่งสามารถหาค่าได้ โดยวิธีการหาค่าต่ำสุด (Minimization) ของผลรวม R ที่สัมพันธ์กับตัวไม่รู้ค่าทั้งสองดังนี้

$$\begin{aligned} \frac{\partial R}{\partial b} &= 0 \\ \frac{\partial R}{\partial b} &= -2(y_1 - (mx_1 + b)) - 2(y_2 - (mx_2 + b)) - \ldots - 2(y_n - (mx_n + b)) = 0 \end{aligned} \tag{3-19}$$

$$\begin{aligned} \frac{\partial R}{\partial m} &= 0 \\ \frac{\partial R}{\partial m} &= -2x_1(y_1 - (mx_1 + b)) - 2x_2(y_2 - (mx_2 + b)) - \ldots - 2x_n(y_n - (mx_n + b)) = 0 \end{aligned} \tag{3-20}$$

เมื่อนำสมการที่ (3-19) และสมการที่ (3-20) มาเขียนใหม่จะได้

$$nb + m\left(\sum_{i=1}^{n} x_i\right) = \sum_{i=1}^{n} y_i \tag{3-21}$$

$$b\left(\sum_{i=1}^{n} x_i\right) + m\left(\sum_{i=1}^{n} x_i^2\right) = \sum_{i=1}^{n} x_i y_i \qquad (3\text{-}22)$$

สมการที่ (3-21) และสมการที่ (3-22) นี้สามารถเขียนให้อยู่ในรูปแบบของเมตริกซ์ได้ดังนี้

$$\begin{bmatrix} n & \sum_{i=1}^{n} x_i \\ \sum_{i=1}^{n} x_i & \sum_{i=1}^{n} x_i^2 \end{bmatrix} \begin{Bmatrix} b \\ m \end{Bmatrix} = \begin{Bmatrix} \sum_{i=1}^{n} x_i \\ \sum_{i=1}^{n} x_i y_i \end{Bmatrix} \qquad (3\text{-}23)$$

ซึ่งสามารถใช้กฎของคราเมอร์แก้ระบบสมการนี้เพื่อหาค่าคงตัว b และ m ได้ดังนี้

$$b = \frac{\left(\sum_{i=1}^{n} y_i\right)\left(\sum_{i=1}^{n} x_i^2\right) - \left(\sum_{i=1}^{n} x_i y_i\right)\left(\sum_{i=1}^{n} x_i\right)}{n\left(\sum_{i=1}^{n} x_i^2\right) - \left(\sum_{i=1}^{n} x_i\right)^2} \qquad (3\text{-}24)$$

$$m = \frac{n\left(\sum_{i=1}^{n} x_i y_i\right) - \left(\sum_{i=1}^{n} x_i\right)\left(\sum_{i=1}^{n} y_i\right)}{n\left(\sum_{i=1}^{n} x_i^2\right) - \left(\sum_{i=1}^{n} x_i\right)^2} \qquad (3\text{-}25)$$

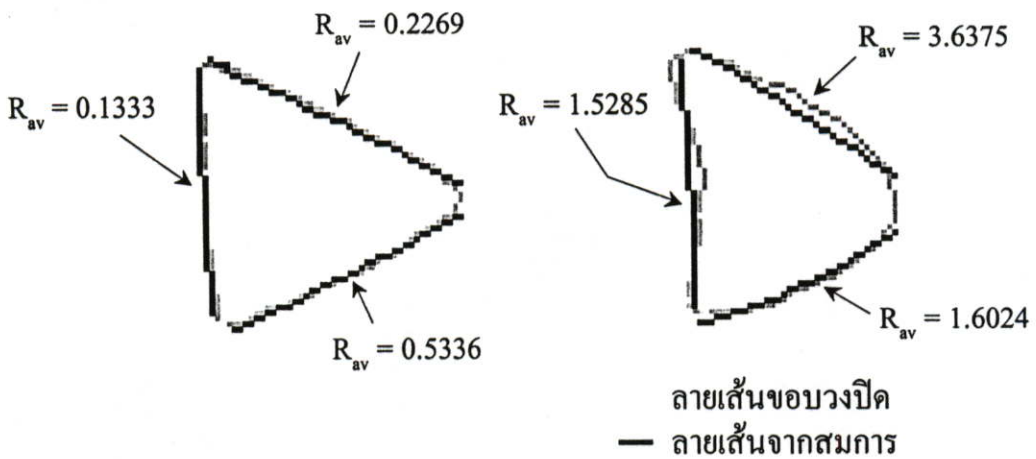
ค่าของ b และ m ที่หาได้นี้เมื่อนำไปแทนในสมการที่ (3-18) จะสามารถหาค่าผลรวม R ได้ ดังที่กล่าวมาแล้วค่าผลรวม R จะมีค่าน้อยหรือเป็นศูนย์เมื่อกลุ่มจุดภาพเป็นเส้นตรง และจะมีค่ามาก เมื่อกลุ่มจุดภาพเป็นเส้นโค้ง แต่จากการทดลองความยาวของลายเส้น (จำนวนจุดภาพ n) มีผลกับค่าผลรวม R จึงทำให้การกำหนดเงื่อนไขในการวิเคราะห์เส้นตรงหรือเส้นโค้งไม่ถูกต้อง เมื่อพิจารณาจากสมการที่ (3-18) ค่าผลรวม R เกิดจากค่าความผิดพลาดของทุกจุดภาพบนลายเส้น ดังนั้นเราสามารถหาค่าความผิดพลาดของแต่ละจุดภาพได้ดังนี้

$$R_{av} = \frac{R}{n} \qquad (3\text{-}26)$$

ค่าของ R_{av} ที่ได้นี้ทำให้เราสามารถกำหนดเงื่อนไขในการวิเคราะห์เส้นตรงหรือเส้นโค้งได้ โดยลายเส้นตรงจะมีค่า R_{av} น้อยกว่าหรือเท่ากับ 1 ส่วนลายเส้นโค้งนั้นจะมีค่า R_{av} มากกว่า 1 ภาพผลลัพธ์แสดงการเปรียบเทียบลายเส้นที่คำนวณได้จากสมการกับค่า R_{av} แสดงดังรูปที่ 3.14 และมีขั้นตอนการประมวลผลดังนี้

ขั้นตอนการประมวลผล

1) คำนวณค่า b และค่า m จากสมการที่ (3-24) และสมการที่ (3-25)
2) คำนวณค่าผลรวม R จากสมการที่ (3-18) โดยแทนค่า b และ m ที่ได้จากข้อ 1.
3) คำนวณค่าผลรวม R_{av} จากสมการที่ (3-26)
4) ตรวจสอบค่า Rav กับเงื่อนไขดังนี้
   - $R_{av} \leq 1$ เป็นเส้นตรง
   - $R_{av} > 1$ เป็นเส้นโค้ง



รูปที่ 3.14 การเปรียบเทียบลายเส้นและค่า R_{av}

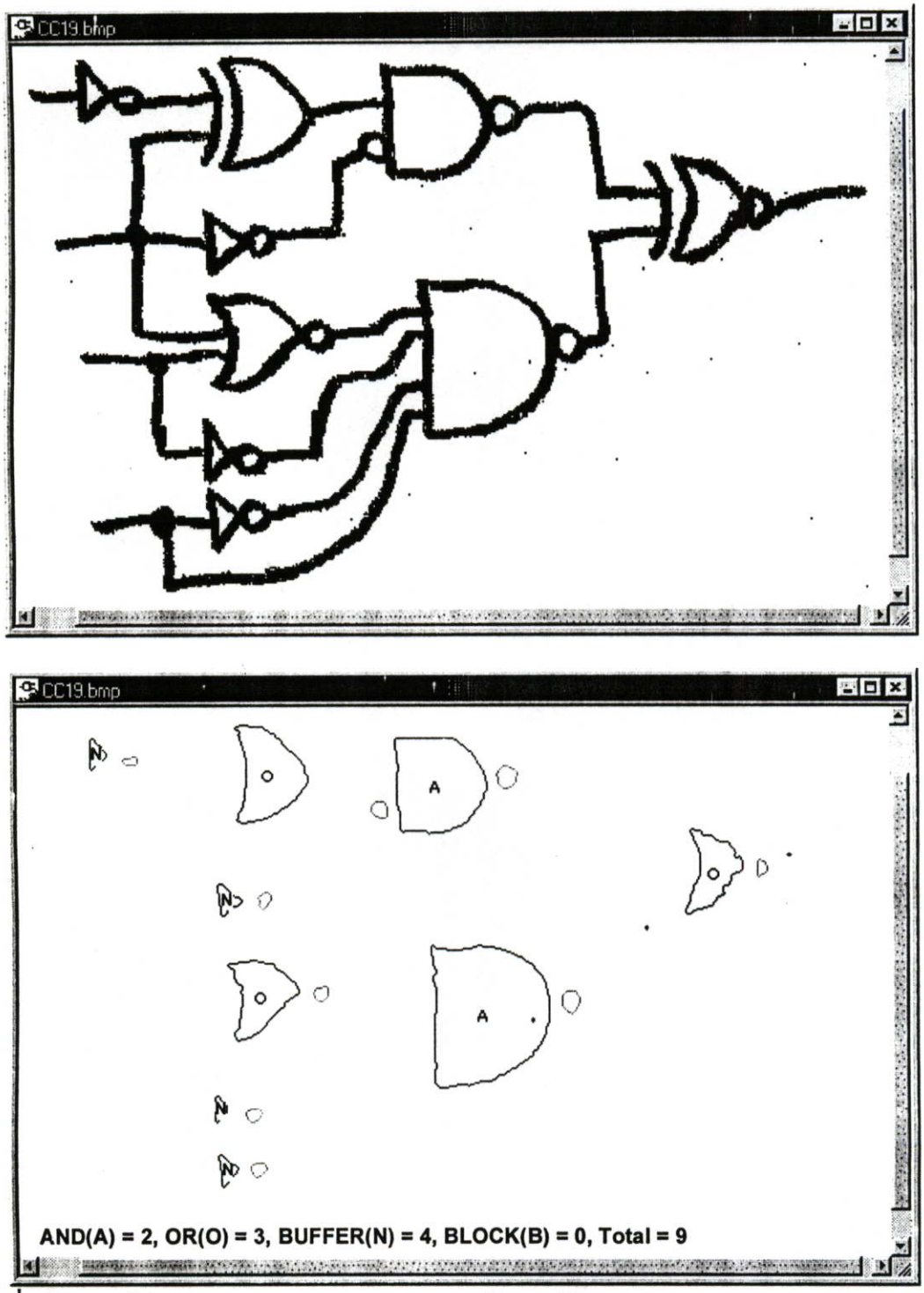
3.5 การจดจำและจำแนกโลจิกเกทตามโครงสร้างหลัก

เมื่อวิเคราะห์ลักษณะลายเส้นที่เป็นองค์ประกอบของเส้นขอบวงปิดครบทุกเส้นแล้วจะสามารถจำแนกโครงสร้างของเส้นขอบวงปิดนั้นได้ว่าเป็นโลจิกเกทโครงสร้าง AND, OR, BUFFER หรือ BLOCK โดยพิจารณาตามเงื่อนไขดังแสดงข้างล่างนี้ อย่างไรก็ตามเงื่อนไขนี้จะกำหนดให้มีความยืดหยุ่นเพียงพอ เพื่อการจดจำและจำแนกเส้นขอบวงปิดทุกๆ เส้นที่ถูกพิจารณาว่าเป็นโลจิกเกท ซึ่งการจดจำชนิดของโลจิกเกททั้งแปดชนิดจะพิจารณาอีกครั้งหลังจากการหาอินพุท-เอาท์พุทเสร็จสิ้น

เงื่อนไขการจดจำและจำแนกโลจิกเกทตามโครงสร้างหลัก

1) โครงสร้าง AND ประกอบด้วยลายเส้น 2 เส้น เป็นเส้นตรง 1 เส้น และเส้นโค้ง 1 เส้น
2) โครงสร้าง OR ประกอบด้วยลายเส้น 3 เส้น เป็นเส้นโค้งอย่างน้อย 2 เส้น
3) โครงสร้าง BUFFER ประกอบด้วยลายเส้น 3 เส้น เป็นเส้นตรงอย่างน้อย 2 เส้น
4) โครงสร้าง BLOCK ประกอบด้วยลายเส้น 4 เส้น เป็นเส้นตรงอย่างน้อย 3 เส้น

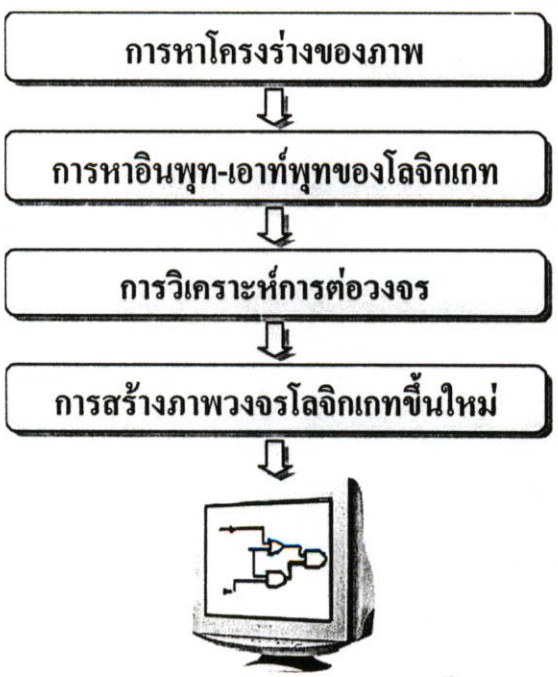
ภาพวงจรโลจิกเกทที่เขียนด้วยมือผ่านขั้นตอนการวิเคราะห์โครงสร้างแล้ว จะทำให้เรารู้โครงสร้างและตำแหน่งของโลจิกเกทแต่ละตัวในภาพวงจร ตัวอย่างของภาพวงจรโลจิกเกทที่เขียนด้วยมือเมื่อผ่านการวิเคราะห์โครงสร้างแสดงไว้ในรูปที่ 3.15 เมื่อ A หมายถึงโครงสร้าง AND, N หมายถึงโครงสร้าง BUFFER และ O หมายถึงโครงสร้าง OR



รูปที่ 3.15 ภาพข้อมูลอินพุทและผลของการวิเคราะห์โครงสร้าง

บทที่ 4
การหาอินพุท-เอาท์พุทและการสร้างภาพวงจรขึ้นใหม่

หลังจากการวิเคราะห์โครงสร้างของโลจิกเกทในขั้นตอนที่ผ่านมา จะทำให้เราสามารถจดจำชนิดและตำแหน่งของโลจิกเกทตามโครงสร้างหลักคือ โครงสร้าง AND, OR, BUFFER และ BLOCK ในบทนี้จะกล่าวถึงขั้นตอนการวิเคราะห์อินพุท-เอาท์พุทไปจนถึงการสร้างภาพวงจรขึ้นใหม่ ซึ่งแบ่งออกเป็น 4 ขั้นตอน ดังแสดงในรูปที่ 4.1 ขั้นตอนแรกเป็นการหาลายเส้นโครงร่างของภาพ (Skeleton) เนื่องจากการวิเคราะห์หาอินพุท-เอาท์พุทของโลจิกเกท และการต่อวงจรจะต้องวิเคราะห์ที่ลายเส้นโครงร่างของภาพ ขั้นตอนที่สองเป็นการวิเคราะห์หาตำแหน่งและจำนวนของอินพุท-เอาท์พุทรวมทั้งตรวจสอบวงกลม NOT ที่อินพุท-เอาท์พุทของโลจิกเกทแต่ละตัว ภายหลังเสร็จสิ้นขั้นตอนนี้ จะทำให้เราสามารถจดจำหรือรู้จำชนิดของโลจิกเกททั้ง 9 ชนิดดังตารางที่ 1.1 ได้ถูกต้องแม่นยำมากขึ้น จากนั้นจะทำการวิเคราะห์การต่อลายวงจรของโลจิกเกททั้งหมดในขั้นตอนที่สาม โดยมีการตรวจสอบการต่อลายวงจรระหว่างโลจิกเกทกับโลจิกเกท โลจิกเกทกับจุดต่อร่วม และจุดต่อร่วมกับจุดต่อร่วม ขั้นตอนสุดท้ายจะนำข้อมูลจากการวิเคราะห์ทั้งหมดมาสร้างภาพวงจรโลจิกเกทขึ้นใหม่ โดยมีการปรับขนาด ทิศทาง และตำแหน่งของโลจิกเกทให้เหมาะสม ซึ่งรายละเอียดของแต่ละขั้นตอนได้นำเสนอไว้ในหัวข้อที่ 4.1 ถึง 4.4



รูปที่ 4.1 ขั้นตอนการหาอินพุท-เอาท์พุทและการสร้างภาพวงจรขึ้นใหม่

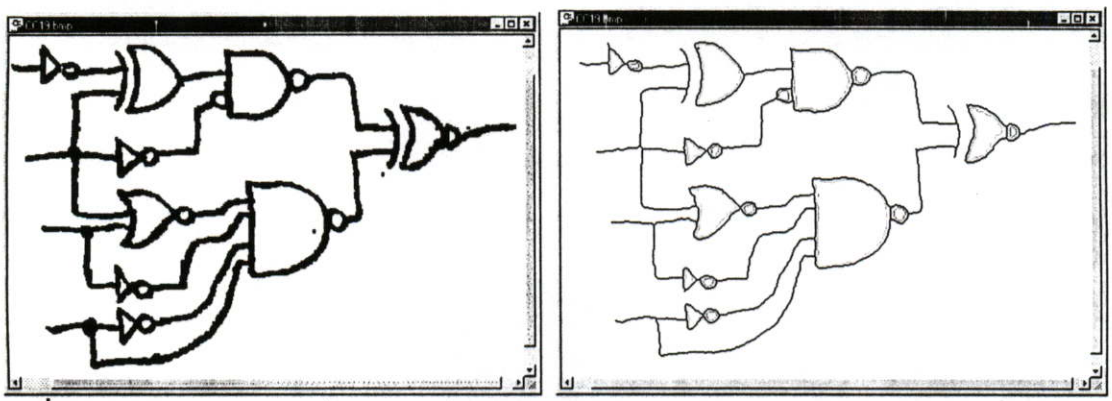
4.1 การหาโครงร่างของภาพ

ในการวิเคราะห์หาอินพุท-เอาท์พุทและการต่อลายวงจรของโลจิกเกทจะต้องทำการวิเคราะห์ที่เส้นโครงร่างของภาพ เนื่องจากเส้นโครงร่างจะมีตำแหน่งอินพุท-เอาท์พุทของโลจิกเกท (จุดแยกบนลายเส้น) อยู่ และมีการต่อเชื่อมเป็นลายวงจร รวมทั้งมีความหนาของลายเส้นเพียง 1 จุดภาพ ซึ่งการหาลายเส้นโครงร่างของภาพสามารถทำได้ด้วยวิธีการทำลายเส้นให้บาง (Thinning algorithm) ที่นำเสนอไว้ในหัวข้อที่ 3.1.1 เมื่อนำภาพวงจรโลจิกเกทไปผ่านการทำลายเส้นให้บางก็จะได้ลายเส้นโครงร่างของภาพดังแสดงในรูปที่ 4.2 และจากการหาลายเส้นโครงร่างนี้ทำให้เราสามารถคำนวณความหนาโดยประมาณจากลายเส้นในภาพ (T_L) ได้ดังสมการที่ (4-1) โดยความหนาของลายเส้นนี้จะนำไปใช้เป็นขอบเขตและเงื่อนไขในขั้นตอนต่างๆ ต่อไป

$$T_L = \frac{N_L}{N_S} \tag{4-1}$$

เมื่อ N_L คือ จำนวนจุดบนลายเส้นทั้งหมดก่อนการหาเส้นโครงร่าง

N_S คือ จำนวนจุดบนลายเส้นโครงร่าง



รูปที่ 4.2 ภาพก่อนและหลังการหาลายเส้นโครงร่างของภาพ

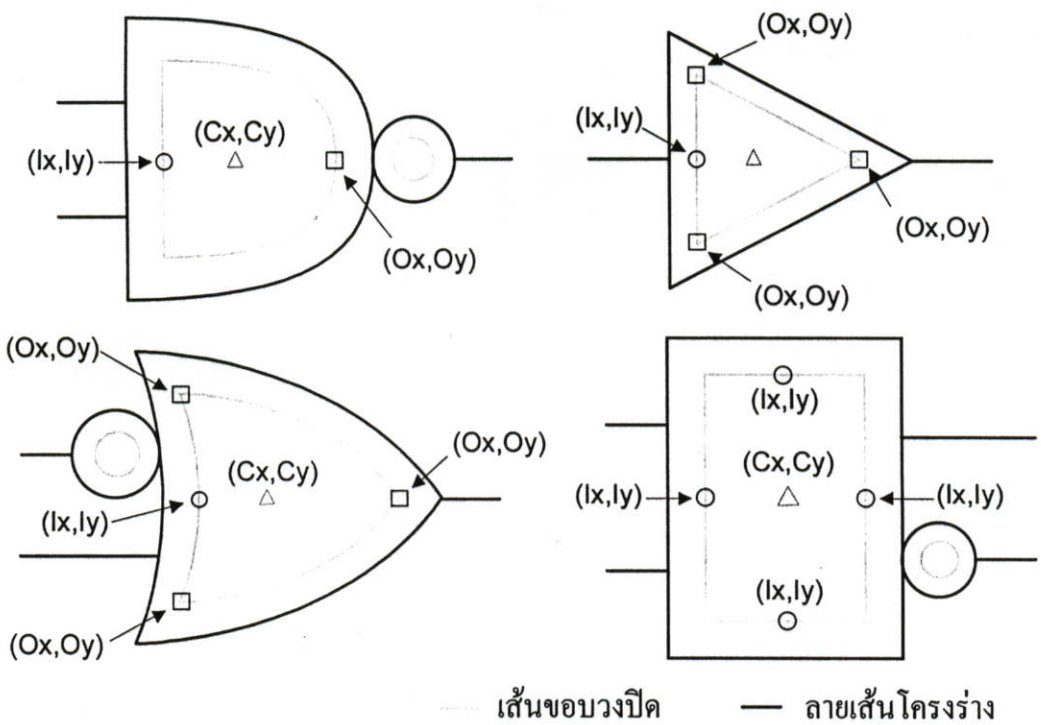
4.2 การหาอินพุท-เอาท์พุทของโลจิกเกท

ในขั้นตอนการวิเคราะห์โครงสร้างที่ผ่านมาจะรู้ตำแหน่งของโลจิกเกทโครงสร้างต่างๆ จากการวิเคราะห์ที่เส้นขอบวงปิด แต่ในการหาอินพุท-เอาท์พุทของโลจิกเกทแต่ละตัว จะหาที่เส้นโครงร่างของตัวโลจิกเกท ดังนั้นการค้นหาเส้นโครงร่างของตัวโลจิกเกทจึงต้องอาศัยจุดบนเส้นขอบวงปิดเป็นจุดเริ่มต้น ซึ่งการกำหนดจุดเริ่มต้นนี้จะกำหนด ณ ตำแหน่งที่คาดว่าจะมีอินพุท-เอาท์พุทอยู่ หรือเป็นตำแหน่งที่ใกล้อินพุท-เอาท์พุทมากที่สุด เมื่อพิจารณาจากสัญลักษณ์ของโลจิกเกทมาตรฐานจะพบว่าเอาท์พุทของโลจิกเกทเกือบทุกชนิดจะมีเพียง 1 เอาท์พุท และมีตำแหน่งที่แน่นอนตามชนิดของโลจิกเกท ดังแสดงในรูปที่ 4.3 ยกเว้นกรณีของ BLOCK ที่ไม่สามารถจำแนก

อินพุท-เอาท์พุทได้ (ในที่นี้จะพิจารณาเป็นอินพุททั้งหมด) การกำหนดจุดเริ่มต้นของการหาเอาท์พุท (Ox,Oy) ของโลจิกเกทโครงสร้างต่างๆ สามารถกำหนดได้ดังนี้

การกำหนดจุดเริ่มต้นของการหาเอาท์พุท

1) โลจิกเกทโครงสร้าง AND จะกำหนดจุดเริ่มต้นของการหาเอาท์พุท (Ox,Oy) ที่จุดกึ่งกลางเส้นโค้งของเส้นขอบวงปิด
2) โลจิกเกทโครงสร้าง OR และ BUFFER จะกำหนดจุดเริ่มต้นของการหาเอาท์พุท (Ox,Oy) ที่จุดมุมทั้งสามของเส้นขอบวงปิด



รูปที่ 4.3 การกำหนดตำแหน่งจุดเริ่มต้นของการหาอินพุท-เอาท์พุท

ส่วนอินพุทของโลจิกเกทแต่ละชนิดจะมีจำนวนที่ไม่แน่นอนโดยมีตั้งแต่ 1 อินพุทขึ้นไป ซึ่งตำแหน่งอินพุทของโลจิกเกทเกือบทุกชนิดจะอยู่ด้านตรงข้ามกับตำแหน่งเอาท์พุทเสมอ ยกเว้นกรณีของ BLOCK ดังรูปที่ 4.3 ดังนั้นการตรวจหาเอาท์พุทก่อนจะทำให้การกำหนดจุดเริ่มต้นของการหาอินพุท (Ix,Iy) ไม่ยุ่งยาก และสามารถกำหนดตามโครงสร้างของโลจิกเกทได้ดังนี้

<u>การกำหนดจุดเริ่มต้นของการหาอินพุท</u>

1) โลจิกเกทโครงสร้าง AND และ BUFFER จะกำหนดจุดเริ่มต้นของการหาอินพุท (Ix,Iy) ที่จุดกึ่งกลางเส้นตรงของเส้นขอบวงปิดด้านตรงข้ามกับตำแหน่งเอาท์พุท

2) โลจิกเกทโครงสร้าง OR จะกำหนดจุดเริ่มต้นของการหาอินพุท (Ix,Iy) ที่จุดกึ่งกลางลายเส้นโค้งของเส้นขอบวงปิดด้านตรงข้ามกับตำแหน่งเอาท์พุท
3) โลจิกเกทโครงสร้าง BLOCK จะกำหนดจุดเริ่มต้นของการหาอินพุท (Ix,Iy) ที่จุดกึ่งกลางลายเส้นตรงของเส้นขอบวงปิดทั้ง 4 ด้าน

4.2.1 การหาเอาท์พุท

กระบวนการหาอินพุท และหาเอาท์พุทของโลจิกเกทจะใช้กระบวนการเดียวกัน โดยจะทำการหาเอาท์พุทก่อน เนื่องจากสามารถทำได้ง่ายกว่าการหาอินพุทดังที่กล่าวมาแล้ว ซึ่งกระบวนการจะแบ่งออกเป็น 3 ขั้นตอนดังต่อไปนี้

4.2.1.1 การค้นหาเส้นโครงร่างของตัวโลจิกเกท

4.2.1.2 การหาจุดแยกบนเส้นโครงร่างของตัวโลจิกเกท

4.2.1.3 การตรวจสอบวงกลม NOT

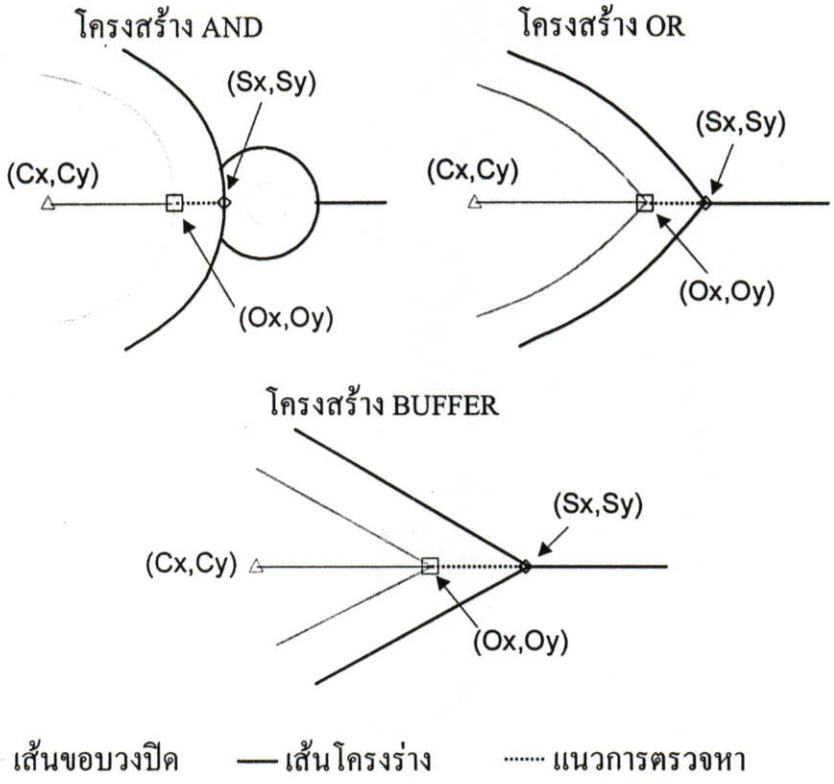
4.2.1.1 การค้นหาเส้นโครงร่างของตัวโลจิกเกท

การค้นหาเส้นโครงร่างของตัวโลจิกเกทจะใช้จุดเริ่มต้นของการหาเอาท์พุท (Ox,Oy) ที่กำหนดขึ้นร่วมกับจุดศูนย์กลางของเส้นขอบวงปิด (Cx,Cy) ที่คำนวณได้จากสมการที่ (4-2) โดยมีการค้นหาเป็นเส้นตรงออกจากจุดศูนย์กลาง (Cx,Cy) ผ่านจุดเริ่มต้น (Ox,Oy) ไปจนถึงเส้นโครงร่างดังแสดงในรูปที่ 4.4 ขอบเขตการหาเส้นโครงร่างถูกกำหนดด้วยความหนาของลายเส้น (T_L) ที่คำนวณได้จากสมการที่ (4-1) ถ้าระยะการค้นหาเท่ากับความหนาของลายแล้วไม่พบเส้นโครงร่างจะยกเลิกการวิเคราะห์หาเอาท์พุท-อินพุทของเส้นขอบวงปิดนั้น และจะทำการวิเคราะห์ในเส้นขอบวงปิดถัดไป ในกรณีที่พบเส้นโครงร่างจะกำหนดให้เป็นจุด (Sx,Sy) เพื่อทำการค้นหาจุดแยกบนเส้นโครงร่างในขั้นตอนต่อไป อัลกอริทึมของการค้นหาลายเส้นแสดงไว้ในอัลกอริทึมที่ 4.1 ซึ่งประยุกต์มาจากอัลกอริทึมการลากเส้นตรงของ Bensanham

$$Cx = \frac{1}{n}\sum_{i=0}^{n} x_i$$
$$Cy = \frac{1}{n}\sum_{i=0}^{n} y_i \qquad (4\text{-}2)$$

เมื่อ x_i, y_i คือ พิกัดในแนวแกน x และ y ของจุด i ใดๆ บนเส้นขอบวงปิด

n คือ จำนวนจุดบนเส้นขอบวงปิด



รูปที่ 4.4 การค้นหาเส้นโครงร่างของตัวโลจิกเกท

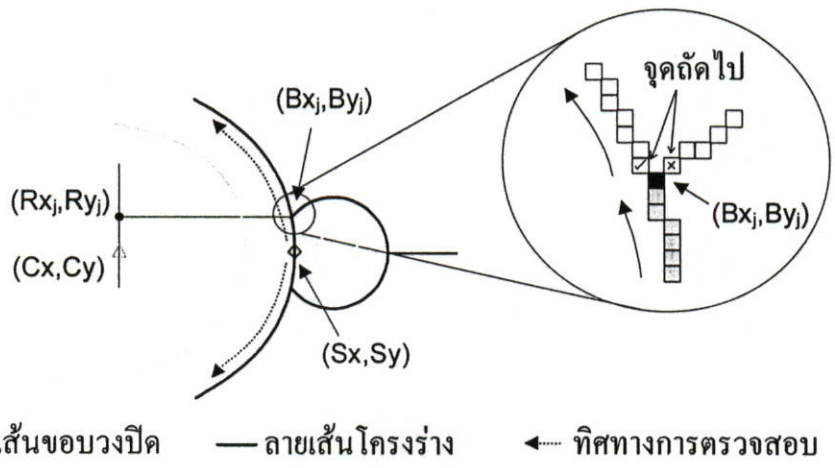
```
Ox,Oy = Start Point.
Cx,Cy = Reference Point.
Npoints = Number of points to scan.
Line = The line must be found.

dx = Ox – Cx ,  dy = Oy – Cy
ix = |dx| ,  iy = |dy|
inc = Max( ix, iy )
pixelx = Ox ,  pixely = Oy
x = 0 ,  y = 0
Found = false

FOR ( ( i = 0 to Npoints ) & ( Found = false ) )
    x = x + ix
    IF ( x > inc )
        IF ( dx > 0)
            pixelx = pixelx + 1
        ELSE
            pixelx = pixelx – 1
        IF ( GetPixel ( pixelx, pixely ) = Line )
            Found = true
    IF ( Found = false )
        y = y + iy
        IF ( y > inc )
            IF ( dy > 0)
                pixely = pixely + 1
            ELSE
                pixely = pixely – 1
            IF ( GetPixel ( pixelx, pixely ) = Line)
                Found = true
```

อัลกอริทึมที่ 4.1 การค้นหาเส้นโครงร่าง

4.2.1.2 การตรวจสอบจุดแยกบนเส้นโครงร่างของโลจิกเกท

เมื่อได้จุด (Sx,Sy) แล้วจะทำการตรวจหาจุดแยกบนเส้นโครงร่างทั้งสองด้านของจุด (Sx,Sy) ดังแสดงในรูปที่ 4.5 โดยจะทำการตรวจหาทีละจุด เมื่อพบจุดแยกจะเก็บตำแหน่งจุดแยกนั้นไว้และกำหนดให้เป็น (Bx_j,By_j) และตรวจหาจุดถัดไป แต่เนื่องจากถัดจากจุดแยกไปจะมีจุดมากกว่าหรือเท่ากับ 2 จุด ดังนั้นเพื่อป้องกันการตรวจสอบผิดลายเส้น (ไม่อยู่บนเส้นโครงร่างของตัวโลจิกเกท) การเลือกจุดถัดไปจะต้องสร้างจุดอ้างอิง (Rx_j,Ry_j) ขึ้นมาเพื่อใช้เปรียบเทียบระยะทาง โดยจุดถัดไปที่จะถูกเลือก จะต้องมีระยะทางจากจุดอ้างอิง (Rx_j,Ry_j) ถึงจุดถัดไปน้อยที่สุด ซึ่งจุดอ้างอิง (Rx_j,Ry_j) นี้สามารถคำนวณได้จากสมการที่ (4-3) ส่วนอัลกอริทึมสำหรับการตรวจหาจุดแยกบนเส้นโครงร่างแสดงไว้ในอัลกอริทึมที่ 4.2

$$\begin{aligned} Rx_j &= (Bx_j - Sx) + Cx \\ Ry_j &= (By_j - Sy) + Cy \end{aligned} \qquad (4\text{-}3)$$



รูปที่ 4.5 การตรวจสอบจุดแยกบนลายเส้นโครงร่าง

การกำหนดระยะหรือความยาวในการตรวจหาจุดแยกบนเส้นโครงร่างจะขึ้นอยู่กับโครงสร้างของโลจิกเกท ซึ่งสามารถกำหนดได้ดังต่อไปนี้

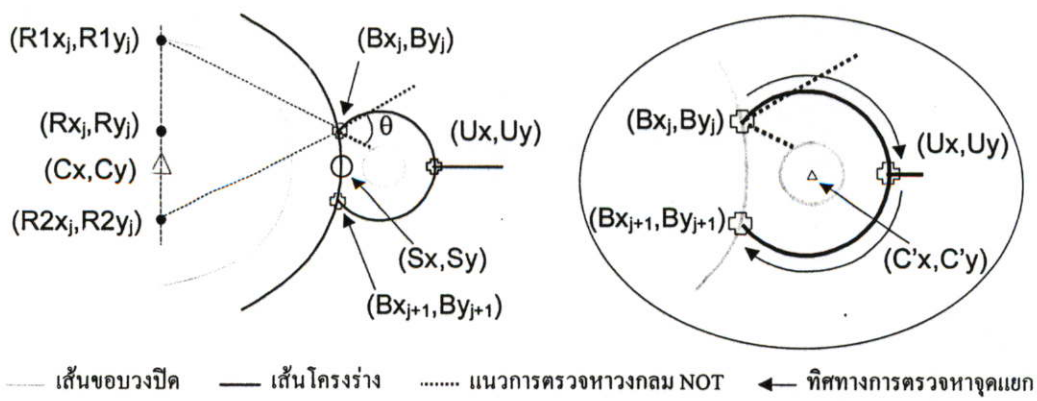
1) โลจิกเกทโครงสร้าง AND จะตรวจหาเป็นระยะทางรวม (ทั้งสองด้านของจุด (Sx,Sy)) เท่ากับ 1/2 ของความยาวลายเส้นโค้ง (ของเส้นขอบวงปิด)
2) โลจิกเกทโครงสร้าง OR จะตรวจหาเป็นระยะทางรวมเท่ากับ 1/9 ของขนาดของโลจิกเกท (จำนวนจุดบนเส้นขอบวงปิดที่เป็นตัวโลจิกเกท)
3) โลจิกเกทโครงสร้าง BUFFER จะตรวจหาเป็นระยะทางรวมเท่ากับ 1/6 ของขนาดของโลจิกเกท

```
Sx,Sy = Start point.
Cx, Cy = Center point.
Npoints = Number of points to scan.

FOR ( i = 0 to 1 )
   px = Sx ,  py = Sy
   FOR ( j = 0 to Npoints/2 )
      Np = p1+p2+p3...+p8 (Neighbour points of (px,py))
      Sp = (p1^p2) + (p2^p3) + ... + (p8^p1)   ;( ^ is Xor)
      IF ( ( Np > 2 ) & ( Sp > 4 ) )
         Keep bifurcation position ( px, py ).
         (p1x, p1y) = NextPointFinding( px, py )
         (p2x, p2y) = NextPointFinding( px, py )
         Rx = Cx + (px – Sx),  Ry = Cy + (py – Sy)
         I1 = Distance( (p1x, p1y), (Rx, Ry) )
         I2 = Distance( (p2x, p2y), (Rx, Ry) )
         IF ( I1 < I2 )
            px = p1x ,  py = p1y
         ELSE
            px = p2x ,  py = p2y
      ELSE
         (px, py) = NextPointFinding(px, py)
```

อัลกอริทึมที่ 4.2 การตรวจสอบจุดแยกบนเส้นโครงร่าง

4.2.1.3 การตรวจสอบวงกลม NOT

ขั้นตอนนี้เป็นการตรวจสอบวงกลม NOT บริเวณจุดแยกทุกจุดที่พบในขั้นตอนที่ผ่านมา ซึ่งวงกลม NOT หมายถึง เส้นขอบวงปิดที่มีขนาดเล็กกว่าตัวโลจิกเกท ในการค้นหาวงกลม NOT จะค้นหาเป็นเส้นตรงสองทิศทางออกจากจุดแยกมีมุมต่างกัน θ องศา ดังแสดงไว้ในรูปที่ 4.6 โดยจะค้นหาทีละเส้นทาง และใช้วิธีการเช่นเดียวกับการค้นหาเส้นโครงร่างของตัวโลจิกเกท (ขั้นตอนที่ 4.2.1.1) แต่จะใช้จุดอ้างอิงต่างกัน เนื่องจากการค้นหามีสองทิศทางจึงต้องใช้จุดอ้างอิงสองจุดคือ $(R1x_j,R1y_j)$ และ $(R2x_j,R2y_j)$ ซึ่งสามารถคำนวณได้จากสมการที่ (4-4a) และ (4-4b) และคำนวณมุมต่าง θ ของเส้นทางทั้งสองจากสมการที่ (4-5) อัลกอริทึมของการค้นหาลายเส้นแสดงไว้ในอัลกอริทึมที่ 4.1



รูปที่ 4.6 การตรวจหาวงกลม NOT

<u>จุดอ้างอิงที่ 1</u>

$$R1x_j = Rx_j + (Sy - Cy)/2$$
$$R1y_j = Ry_j + (Cx - Sx)/2 \qquad (4\text{-}4a)$$

<u>จุดอ้างอิงที่ 2</u>

$$R2x_j = Rx_j - (Sy - Cy)/2$$
$$R2y_j = Ry_j - (Cx - Sx)/2 \qquad (4\text{-}4b)$$

$$\theta = \tan^{-1}\left(\frac{\sqrt{(Rx - R1x_j)^2 + (Ry - R1y_j)^2}}{\sqrt{(Rx - Bx)^2 + (Ry - By)^2}}\right) + \tan^{-1}\left(\frac{\sqrt{(Rx - R2x_j)^2 + (Ry - R2y_j)^2}}{\sqrt{(Rx - Bx)^2 + (Ry - By)^2}}\right)$$

$$\theta = 2\tan^{-1}\left(\sqrt{\frac{\left(\frac{(Cy - Sy)}{2}\right)^2 + \left(\frac{(Sx - Cx)}{2}\right)^2}{(Cx - Sx)^2 + (Cy - Sy)^2}}\right) = 2\tan^{-1}\left(\frac{1}{2}\right) = 53.13° \qquad (4\text{-}5)$$

การค้นหาวงกลม NOT ถ้าไม่พบเส้นขอบวงปิดของวงกลม NOT ด้วยระยะทางเท่ากับความหนาของลายเส้นจะพิจารณาว่าจุดแยกที่ตรวจสอบ (Bx_j,By_j) เป็นตำแหน่งเอาท์พุท (Ux,Uy) ในทางตรงกันข้าม ถ้าการค้นหาพบเส้นขอบวงปิดของวงกลม NOT จะต้องตรวจสอบขนาดของวงกลม NOT นั้นว่าเหมาะสมกับตัวโลจิกเกทหรือไม่ โดยมีหลักเกณฑ์ดังนี้

1) โลจิกเกทโครงสร้าง AND, OR และ BLOCK ควรมีขนาดของวงกลม NOT เล็กกว่า 1/3 ของขนาดตัวโลจิกเกท
2) โลจิกเกทโครงสร้าง BUFFER ควรมีขนาดของวงกลม NOT เล็กกว่าขนาดของตัวโลจิกเกท

ถ้าวงกลม NOT ที่พบไม่ตรงตามหลักเกณฑ์จะพิจารณาว่าจุดแยกที่ตรวจสอบ (Bx_j,By_j) เป็นตำแหน่งเอาท์พุท (Ux,Uy) แต่ถ้าวงกลม NOT ตรงตามหลักเกณฑ์จะตรวจหาจุดแยก (ตำแหน่งเอาท์พุทที่ถูกต้อง) บนเส้นโครงร่างที่ล้อมรอบวงกลม NOT โดยใช้วิธีการเช่นเดียวกับการหาจุดแยกบนเส้นโครงร่างในขั้นตอนที่ 4.2.1.2 โดยเริ่มตรวจสอบที่จุดแยก (Bx_j,By_j) ดังแสดงในรูปที่ 4.6 และใช้จุด (C'x,C'y) ของเส้นขอบวงปิดวงกลม NOT ที่คำนวณจากสมการที่ (4-2) เป็นจุดอ้างอิงการตรวจสอบเมื่อพบจุดแยกจะกำหนดให้เป็นตำแหน่งเอาท์พุท (Ux,Uy) และตรวจสอบต่อไปจนสุดลายเส้นหรือบรรจบกับจุดแยก (Bx_{j+1},By_{j+1}) ซึ่งจุดแยกนี้จะถูกยกเลิก (ไม่ใช่ตำแหน่งเอาท์พุท) และไม่นำมาตรวจสอบวงกลม NOT อีก

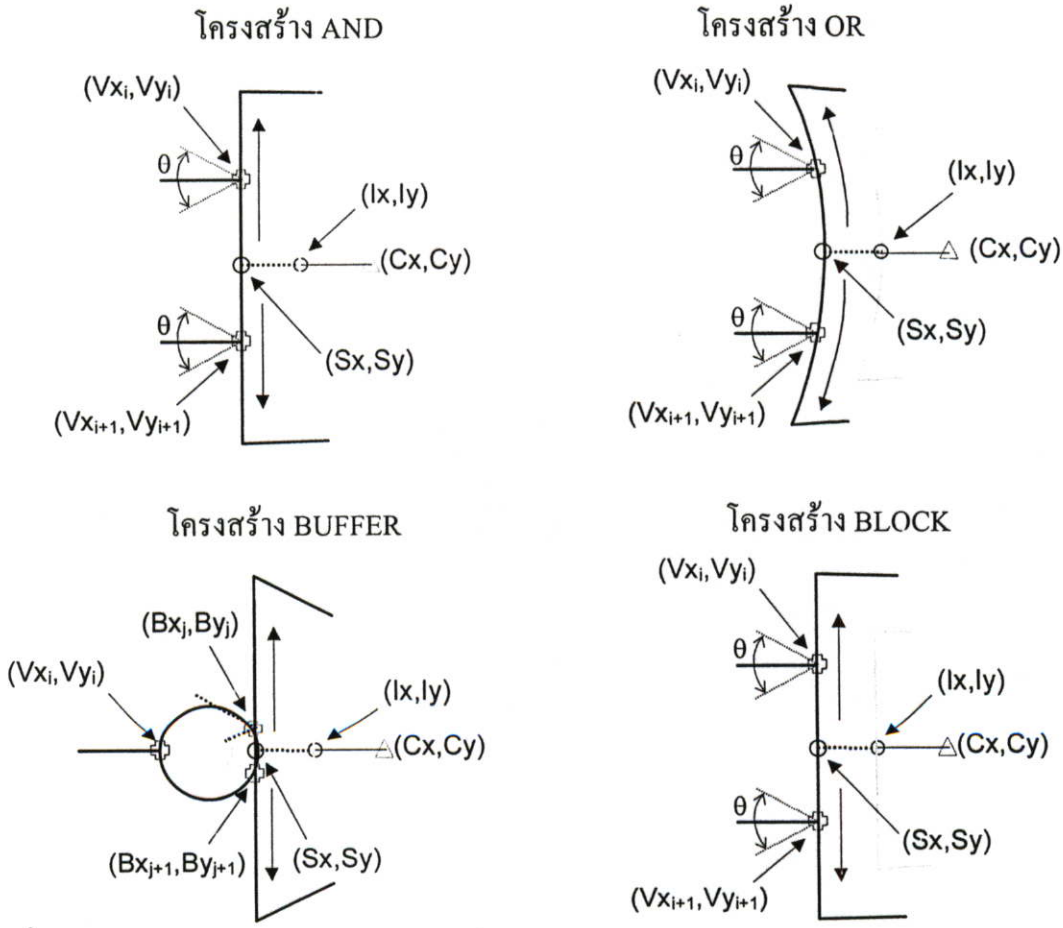
ในการประมวลผล โลจิกเกทโครงสร้าง AND จะหาเอาท์พุทที่จุดเริ่มต้น (Ox,Oy) 1 จุด แต่สำหรับโลจิกเกทโครงสร้าง OR และ BUFFER จะหาเอาท์พุทที่จุดเริ่มต้น (Ox,Oy) ทีละจุดจนกว่า

จะพบเอาท์พุทหรือครบทั้ง 3 จุด ซึ่งการหาเอาท์พุทนี้ ถ้าไม่พบเอาท์พุทหรือพบเอาท์พุทมากกว่า 1 เอาท์พุท จะพิจารณาว่าไม่ใช่โลจิกเกท และจะไม่ทำการหาอินพุทในขั้นตอนต่อไป ในทางกลับกัน ถ้าพบ 1 เอาท์พุท ก็จะทำการตรวจหาอินพุทต่อไป

4.2.2 การหาอินพุท

การหาอินพุทสามารถทำได้ด้วยวิธีการเช่นเดียวกับการหาเอาท์พุทดังแสดงในรูปที่ 4.7 โดยใช้จุดเริ่มต้น (Ix,Iy) แทนจุด (Ox,Oy) และกำหนดระยะการตรวจสอบจุดแยกบนเส้นโครงร่างของตัวโลจิกเกทในขั้นตอนที่ 4.2.1.2 ตามโครงสร้างดังนี้

1) โลจิกเกทโครงสร้าง AND และ BUFFER จะตรวจสอบเป็นระยะทางรวมเท่ากับความยาวลายเส้นตรง (ของเส้นขอบวงปิด) ด้านตรงข้ามกับตำแหน่งเอาท์พุท
2) โลจิกเกทโครงสร้าง OR จะตรวจสอบเป็นระยะทางรวมเท่ากับความยาวลายเส้นโค้งด้านตรงข้ามกับตำแหน่งเอาท์พุท
3) โลจิกเกทโครงสร้าง BLOCK จะตรวจสอบเป็นระยะทางรวมเท่ากับความยาวลายเส้นตรงแต่ละด้านที่ทำการตรวจหาอินพุท

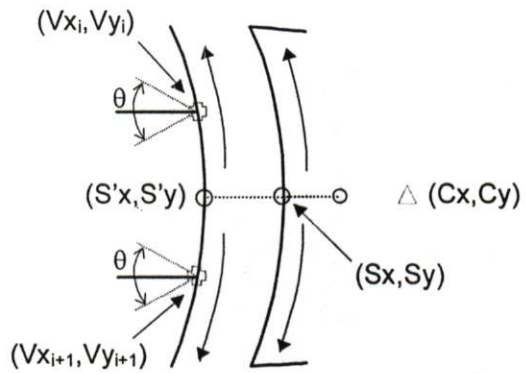


รูปที่ 4.7 การหาอินพุทของโลจิกเกทโครงสร้างต่างๆ

เนื่องจากในการวิเคราะห์โครงสร้าง ขั้นตอนการวิเคราะห์ลักษณะลายเส้นอาจมีความผิดพลาด และทำให้เกิดความสับสนของการวิเคราะห์โลจิกเกทโครงสร้าง OR กับ BUFFER ดังนั้นในการประมวลผลสำหรับการหาอินพุทของโลจิกเกท เราจะใช้จำนวนของอินพุทที่ตรวจพบช่วยในการจดจำชนิดของโลจิกเกทด้วย โดยนำจำนวนอินพุทที่ตรวจพบมาพิจารณาร่วมกับโครงสร้างของโลจิกเกทอีกครั้ง โดยมีหลักเกณฑ์ดังต่อไปนี้

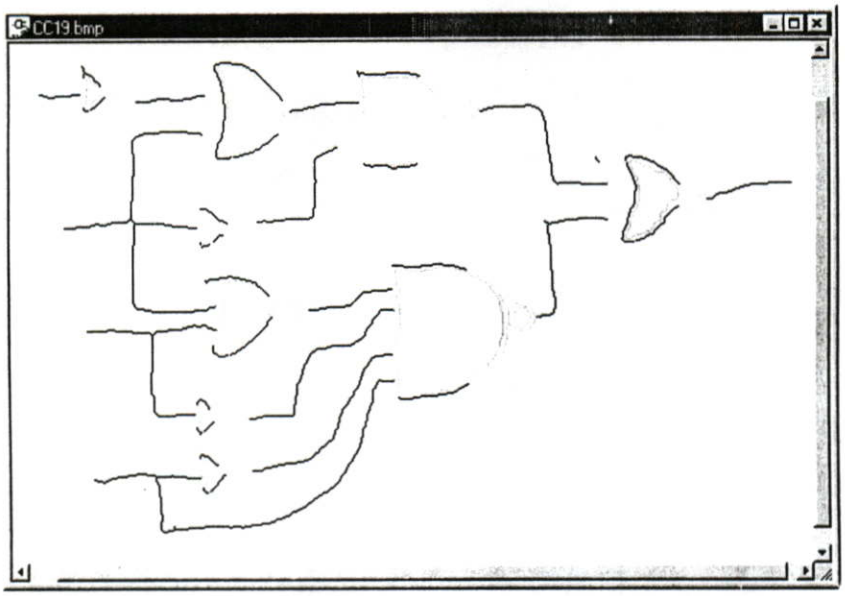
1) โลจิกเกทโครงสร้าง AND จะต้องหาอินพุทได้มากกว่าหรือเท่ากับ 2 อินพุท
2) โลจิกเกทโครงสร้าง BLOCK เมื่อทำการหาอินพุทครบทั้ง 4 ด้านจะต้องมีอินพุทมากกว่าหรือเท่ากับ 1 อินพุท
3) โลจิกเกทโครงสร้าง BUFFER จะต้องหาอินพุทได้ 1 อินพุท แต่ถ้าสามารถหาได้มากกว่า 1 อินพุท จะพิจารณาและเปลี่ยนเป็นโลจิกเกทโครงสร้าง OR แทน
4) โลจิกเกทโครงสร้าง OR จะต้องหาอินพุทได้มากกว่าหรือเท่ากับ 2 อินพุท แต่ถ้าหาได้เพียง 1 อินพุท จะพิจารณาและเปลี่ยนเป็นโลจิกเกทโครงสร้าง BUFFER แทน

ส่วนกรณีของโลจิกเกทชนิด Exclusive การตรวจหาอินพุทที่เส้นโครงร่าง ซึ่งเป็นเส้นโค้งตรงข้ามกับเอาท์พุทจะตรวจหาไม่พบ ดังนั้นจะต้องค้นหาเส้นโค้งที่อยู่ด้านหลังถัดจากโลจิกเกทออกไปดังรูปที่ 4.8 โดยการลากเส้นตรงจากจุด (Cx,Cy) ผ่านจุด (Sx,Sy) จนกระทั่งพบเส้นโครงร่างที่อยู่ด้านหลัง โดยกำหนดระยะทางในการค้นหาเส้นดังกล่าวเท่ากับระยะทางจากจุด (Sx,Sy) ถึงจุดศูนย์กลาง (Cx,Cy) จากนั้นทำการค้นหาอินพุทตามวิธีการที่ได้อธิบายไปแล้วในขั้นตอนที่ 4.2.1.1 ถึง 4.2.1.3 เมื่อประมวลผลครบ 3 ขั้นตอน จะต้องมีอินพุทมากกว่าหรือเท่ากับ 2 อินพุท



รูปที่ 4.8 การตรวจหาอินพุทของโลจิกเกทชนิด Exclusive

ภายหลังจากการตรวจหาอินพุท–เอาท์พุทของโลจิกเกท จะทำให้เราจดจำนวนและตำแหน่งของอินพุท–เอาท์พุทรวมทั้งวงกลม NOT ที่อินพุท-เอาท์พุทด้วย และสามารถจดจำชนิดของโลจิกเกททั้ง 9 ชนิดได้ ส่วนภาพวงจรโลจิกเกทเมื่อผ่านการตรวจหาอินพุท–เอาท์พุทแล้ว ลายเส้นโครงร่างของตัวโลจิกเกทจะถูกลบออกจากลายวงจรดังแสดงในรูปที่ 4.9 ทั้งนี้เพื่อลดความซับซ้อนของการวิเคราะห์การต่อลายวงจรในขั้นตอนต่อไป



รูปที่ 4.9 ภาพหลังการหาอินพุท–เอาท์พุทของโลจิกเกท

4.3 การวิเคราะห์การต่อวงจร

การวิเคราะห์การต่อวงจรสามารถทำได้โดยใช้กระบวนการติดตามลายเส้น (Line tracing) ตามอัลกอริทึมที่ 4.3 โดยเริ่มต้นติดตามลายเส้นจากตำแหน่งอินพุท–เอาท์พุทของโลจิกเกททีละตัวไปจนกระทั่งพบจุดแยก (จุดต่อร่วม) หรือสิ้นสุดลายเส้น ซึ่งการติดตามลายเส้นจากตำแหน่งอินพุท–เอาท์พุทของโลจิกเกทนี้จะทำให้สามารถวิเคราะห์การต่อวงจรได้ดังนี้

1) เมื่อการติดตามลายเส้นพบจุดแยก (จุดต่อร่วม) จะวิเคราะห์การต่อวงจรเป็น "โลจิกเกทต่อกับจุดต่อร่วม"
2) เมื่อการติดตามลายเส้นไปพบจุดปลายของลายเส้นจะนำตำแหน่งสุดท้ายไปเปรียบเทียบกับตำแหน่งอินพุท–เอาท์พุทของโลจิกเกททุกตัวและสามารถวิเคราะห์การต่อลายวงจรได้สองรูปแบบคือ
   - ตำแหน่งสุดท้ายตรงกับตำแหน่งอินพุทหรือเอาท์พุทของโลจิกเกทจะวิเคราะห์การต่อวงจรเป็น "โลจิกเกทต่อกับโลจิกเกท"
   - ตำแหน่งสุดท้ายไม่ตรงกับตำแหน่งอินพุทหรือเอาท์พุทของโลจิกเกทจะวิเคราะห์การต่อวงจรเป็น "โลจิกเกทต่อกับจุดปลาย" (ไม่ต่อวงจร)

เมื่อตรวจสอบการต่อวงจรที่อินพุท-เอาท์พุทของโลจิกเกทครบทุกตัวแล้ว ตำแหน่งจุดแยก (จุดต่อร่วม) ที่พบทั้งหมดจะนำมาตรวจสอบการต่อวงจรอีกครั้ง โดยการติดตามลายเส้นจากตำแหน่งจุดแยกทีละจุด ซึ่งสามารถวิเคราะห์การต่อวงจรได้ดังนี้

1) เมื่อการติดตามลายเส้นพบจุดแยกจะวิเคราะห์การต่อวงจรเป็น "จุดต่อร่วมต่อกับจุดต่อร่วม" และนำตำแหน่งจุดแยกไปเปรียบเทียบกับตำแหน่งจุดแยกที่เก็บไว้ทั้งหมด ถ้าไม่ตรง

กับตำแหน่งจุดแยกที่เก็บไว้ก็จะเก็บตำแหน่งจุดแยกนี้เพิ่มเพื่อทำการตรวจสอบการต่อวงจรต่อไป

2) เมื่อการติดตามลายเส้นไปพบจุดปลายของลายเส้นจะวิเคราะห์การต่อวงจรเป็น "จุดต่อร่วมต่อกับจุดปลาย"

```
Sx, Sy = Start point.

FoundBifurcation = false
Stop = false
DO
    Nx, Ny = NextPointFinding(Sx, Sy)
    IF ( (Nx != Sx) | (Ny != Sy) )
        Np = p1+p2+p3...+p8 (neighbour points of (px,py))
        Sp = (p1^p2) + (p2^p3) + ... + (p8^p1) ( ^ is Xor)
        IF ( ( Np > 2 ) & ( Sp > 4 ) )
            FoundBifurcation = true
            Stop = true
    ELSE
        Stop = true
WHILE( Stop = false )
```

อัลกอริทึมที่ 4.3 การติดตามลายเส้น

เมื่อวิเคราะห์การต่อวงจรครบทั้งภาพจะทำให้เรารู้การต่อวงจรทั้งหมดในภาพ ถึงตอนนี้ข้อมูลที่วิเคราะห์มาทั้งหมดจะถูกเก็บไว้เป็นฐานข้อมูล และสามารถนำไปสร้างภาพวงจรโลจิกเกทขึ้นใหม่ได้ในขั้นตอนต่อไป

4.4 การสร้างภาพวงจรโลจิกเกทขึ้นใหม่

ขั้นตอนการสร้างภาพวงจรโลจิกเกทขึ้นใหม่นี้เป็นขั้นตอนสุดท้ายที่จะนำข้อมูลจากการวิเคราะห์ในขั้นตอนที่ผ่านมาทั้งหมดมาสร้างภาพวงจรโลจิกเกทขึ้นใหม่ โดยจะมีการปรับปรุงข้อมูลให้เหมาะสมก่อนการวาดภาพเช่น ปรับขนาด ทิศทาง และตำแหน่งของโลจิกเกท เป็นต้น จากนั้นจึงวาดโลจิกเกทชนิดต่างๆ ตามตำแหน่งและทิศทางใหม่ สุดท้ายเป็นการลากเส้นต่อลายวงจรให้สมบูรณ์ รายละเอียดของวิธีการต่างๆ มีดังต่อไปนี้

4.4.1 การปรับปรุงข้อมูล

ภาพวงจรโลจิกเกทที่เขียนด้วยมือโดยทั่วไปโลจิกเกทจะไม่สมมาตร เอียง มีขนาดและทิศทางแตกต่างกัน ดังนั้นก่อนการสร้างภาพวงจรขึ้นใหม่จำเป็นต้องปรับปรุงขนาด ทิศทางและตำแหน่งของโลจิกเกทให้เหมาะสม ในการปรับปรุงทิศทางและตำแหน่งของโลจิกเกทจำเป็นต้องรู้ทิศทางของโลจิกเกท (β) แต่ละตัวและทิศทางหลักของโลจิกเกท (α) ในภาพ ซึ่งทิศทางของโลจิก

เกทแต่ละตัวสามารถคำนวณได้จากตำแหน่งของโลจิกเกท (จุดศูนย์กลาง (Cx,Cy)) และตำแหน่งเอาท์พุท (Ux,Uy) ยกเว้นกรณีของ BLOCK จะใช้ตำแหน่งกึ่งกลางลายเส้นด้านยาว (Ix,Iy) ของเส้นขอบวงปิดด้านใดด้านหนึ่ง (ไม่สามารถจำแนกอินพุทหรือเอาท์พุทได้) และคำนวณทิศทางของโลจิกเกทจากสมการดังนี้

$$\beta = \tan^{-1}\left(\frac{Uy - Cy}{Ux - Cx}\right) \tag{4-6}$$

เมื่อรู้ทิศทางของโลจิกเกทแต่ละตัวก็สามารถหาทิศทางหลักของโลจิกเกท (α) ได้โดยแบ่งโลจิกเกทออกเป็น 4 กลุ่มดังนี้

กลุ่มที่ 1 โลจิกเกทจะต้องมีทิศทางต่างจากแกน 0° น้อยกว่า ±45°

กลุ่มที่ 2 โลจิกเกทจะต้องมีทิศทางต่างจากแกน 90° น้อยกว่า ±45°

กลุ่มที่ 3 โลจิกเกทจะต้องมีทิศทางต่างจากแกน 180° น้อยกว่า ±45°

กลุ่มที่ 4 โลจิกเกทจะต้องมีทิศทางต่างจากแกน 270° น้อยกว่า ±45°

เมื่อจัดกลุ่มของโลจิกเกทแล้วจะเลือกกลุ่มที่มีจำนวนโลจิกเกทมากที่สุด เพื่อคำนวณหาทิศทางเฉลี่ยของกลุ่ม และกำหนดเป็นทิศทางหลัก (α) ของโลจิกเกทในภาพ ซึ่งมีสมการดังนี้

$$\alpha = \frac{1}{n}\sum_{i=1}^{n}\beta_i \tag{4-7}$$

เมื่อ β_i คือ ทิศทางของโลจิกเกทในกลุ่มที่มีจำนวนโลจิกเกทมากที่สุด

n คือ จำนวนโลจิกเกทในกลุ่ม

ทิศทางหลัก (α) ของโลจิกเกทนี้จะถูกนำมาใช้เพื่อปรับหรือหมุนตำแหน่งต่างๆ ที่ได้จากการวิเคราะห์คือ ตำแหน่งโลจิกเกท ตำแหน่งอินพุท-เอาท์พุท และตำแหน่งจุดต่อร่วมให้อยู่ในตำแหน่งที่เหมาะสม โดยสามารถคำนวณตำแหน่งใหม่ได้จากสมการดังนี้

$$\begin{aligned} x' &= (x - ICx)Cos(\alpha) - (y - ICy)Sin(\alpha) + ICx \\ y' &= (y - ICy)Cos(\alpha) - (x - ICx)Sin(\alpha) + ICy \end{aligned} \tag{4-8}$$

เมื่อ (ICx,ICy) คือ จุดศูนย์กลางของภาพ

แต่เนื่องจากการเขียนโลจิกเกทในวงจรโดยทั่วไปจะเขียนในแนวแกน x และแกน y หรือมี 4 ทิศทางคือ 0, 90, 180 และ 270 องศา ดังนั้นวงจรที่สร้างขึ้นใหม่จะต้องสามารถวาดโลจิกเกทได้

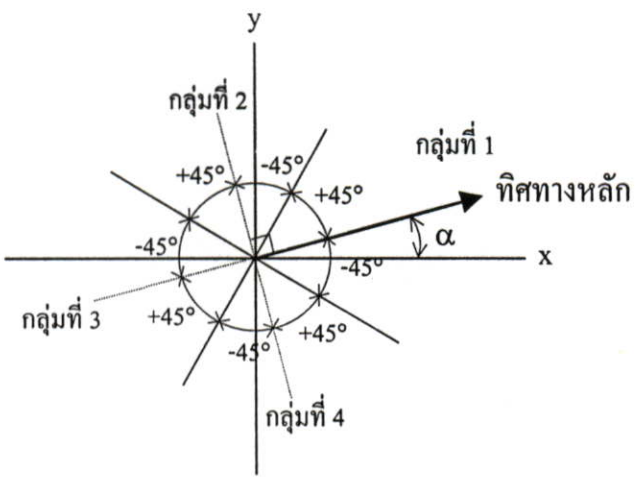
ทั้ง 4 ทิศทางเหมือนกับภาพข้อมูลที่รับเข้ามา ด้วยเหตุนี้ทิศทางหลักของโลจิกเกทจึงถูกนำมาใช้เป็นทิศทางอ้างอิงเพื่อจัดกลุ่มของโลจิกเกทใหม่ตามทิศทาง โดยแบ่งออกเป็น 4 กลุ่ม ดังแสดงในรูปที่ 4.10 และมีทิศทางของแต่ละกลุ่มดังนี้

กลุ่มที่ 1 โลจิกเกทจะต้องมีทิศทางต่างจากทิศทางหลักเท่ากับ $\alpha \pm 45^\circ$

กลุ่มที่ 2 โลจิกเกทจะต้องมีทิศทางต่างจากทิศทางหลักเท่ากับ $\alpha + 90^\circ \pm 45^\circ$

กลุ่มที่ 3 โลจิกเกทจะต้องมีทิศทางต่างจากทิศทางหลักเท่ากับ $\alpha + 180^\circ \pm 45^\circ$

กลุ่มที่ 4 โลจิกเกทจะต้องมีทิศทางต่างจากทิศทางหลักเท่ากับ $\alpha + 270^\circ \pm 45^\circ$



รูปที่ 4.10 การแบ่งกลุ่มและการปรับทิศทางของโลจิกเกท

หลังจากการจัดกลุ่มแล้วจะทำการหมุนทิศทางหลักไปที่ 0 องศา ทำให้ทิศทางของโลจิกเกทในกลุ่มต่างๆ จะถูกเปลี่ยนไปเป็นดังต่อไปนี้

โลจิกเกทกลุ่มที่ 1 จะกำหนดให้มีทิศทางเป็น 0 องศา

โลจิกเกทกลุ่มที่ 2 จะกำหนดให้มีทิศทางเป็น 90 องศา

โลจิกเกทกลุ่มที่ 3 จะกำหนดให้มีทิศทางเป็น 180 องศา

โลจิกเกทกลุ่มที่ 4 จะกำหนดให้มีทิศทางเป็น 270 องศา

เนื่องจากในการสร้างภาพวงจรขึ้นใหม่ โลจิกเกทจะมีขนาดมาตรฐานคงที่ แต่ภาพวงจรที่รับเข้ามาจะมีขนาดของโลจิกเกทแตกต่างกัน ดังนั้นจำเป็นต้องหาอัตราการย่อขยายขนาดของโลจิกเกท (Scale) เพื่อปรับตำแหน่งต่างๆ ใหม่ ซึ่งสามารถทำได้โดยการคำนวณหาอัตราการย่อขยายเฉลี่ยของโลจิกเกทโครงสร้าง AND, OR และ BUFFER (ยกเว้นโครงสร้าง BLOCK ไม่นำมาพิจารณาเพราะมีขนาดความกว้างและความยาวไม่แน่นอน) อัตราการย่อขยายขนาดของโลจิกเกทสามารถคำนวณได้จากสมการดังนี้

$$Scale = \frac{1}{3}\left[\frac{\sum_{i=1}^{m} A_i}{mS_A} + \frac{\sum_{j=1}^{n} O_j}{nS_O} + \frac{\sum_{k=1}^{p} B_k}{pS_B}\right] \quad (4\text{-}9)$$

เมื่อ A_i คือ ขนาดของโลจิกเกทโครงสร้าง AND

O_j คือ ขนาดของโลจิกเกทโครงสร้าง OR

B_k คือ ขนาดของโลจิกเกทโครงสร้าง BUFFER

S_A คือ ขนาดมาตรฐานของโลจิกเกทโครงสร้าง AND

S_O คือ ขนาดมาตรฐานของโลจิกเกทโครงสร้าง OR

S_B คือ ขนาดมาตรฐานของโลจิกเกทโครงสร้าง BUFFER

m คือ จำนวนโลจิกเกทโครงสร้าง AND ทั้งหมดในภาพวงจร

n คือ จำนวนโลจิกเกทโครงสร้าง OR ทั้งหมดในภาพวงจร

p คือ จำนวนโลจิกเกทโครงสร้าง BUFFER ทั้งหมดในภาพวงจร

เมื่อได้อัตราการย่อขยาย (Scale) ก็สามารถคำนวณหาตำแหน่งของโลจิกเกท ตำแหน่งอินพุท-เอาท์พุท และตำแหน่งจุดต่อร่วมใหม่ได้ด้วยสมการดังต่อไปนี้

$$x'' = \frac{x'}{Scale}$$
$$y'' = \frac{y'}{Scale} \quad (4\text{-}10)$$

ในการจัดเรียงโลจิกเกทและจุดต่อร่วมให้อยู่ในแนวเดียวกันนั้นสามารถทำได้ โดยการกำหนดแนวตาราง (Grid) ขึ้นมา และปรับตำแหน่งต่างๆ ให้ตรงกับแนวตารางที่กำหนดไว้ ซึ่งการปรับตำแหน่งต่างๆ ให้ตรงกับแนวตารางสามารถคำนวณได้จากสมการดังนี้

$$x''' = (x'' Mod\ G_H)G_H$$
$$y''' = (y'' Mod\ G_V)G_V \quad (4\text{-}11)$$

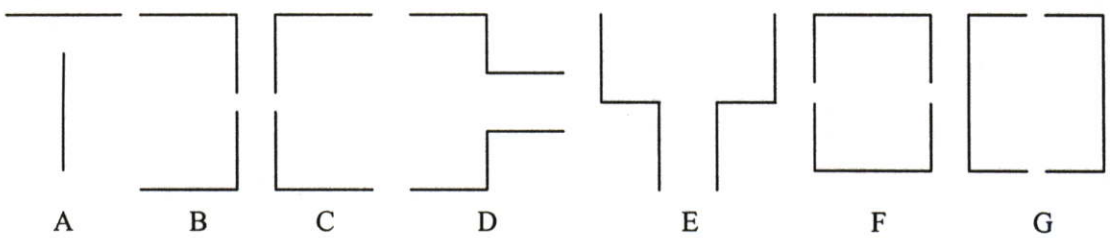
เมื่อ Mod คือ การหารเป็นเลขจำนวนเต็ม

G_H คือ ความกว้างของตารางในแนวนอน

G_V คือ ความกว้างของตารางในแนวตั้ง

4.4.2 การวาดภาพ

เมื่อปรับตำแหน่งและทิศทางของโลจิกเกทแล้วจะทำการวาดภาพโลจิกเกทมาตรฐานที่กำหนดไว้ตรงตำแหน่งและทิศทางใหม่จนครบทุกตัว จากนั้นจึงทำการลากเส้นต่อวงจร โดยจะลากเส้นในแนวแกน x และแกน y เท่านั้น ซึ่งมีรูปแบบการลากเส้นหลายรูปแบบดังแสดงในรูปที่ 4.11 ทั้งนี้เพื่อให้การลากเส้นต่อวงจรมีความเหมาะสม การลากเส้นแต่ละรูปแบบจะมีการตรวจสอบการทับกันของลายเส้นและตัวโลจิกเกทด้วย ดังนั้นการใช้รูปแบบการลากเส้นเพียงรูปแบบเดียวอาจต่อวงจรได้ไม่สำเร็จ ขณะเดียวกันการเลือกใช้รูปแบบการลากเส้นแต่ละรูปแบบจะต้องคำนึงถึงทิศทางของโลจิกเกทที่จะทำการเชื่อมต่อด้วย ดังนั้นเราจึงกำหนดการเลือกใช้รูปแบบการลากเส้นที่สัมพันธ์กับทิศทางของโลจิกเกทดังตารางที่ 4.1 ผลของการวาดภาพโลจิกเกทและการลากเส้นต่อวงจรที่สมบูรณ์แสดงไว้ในรูปที่ 4.12

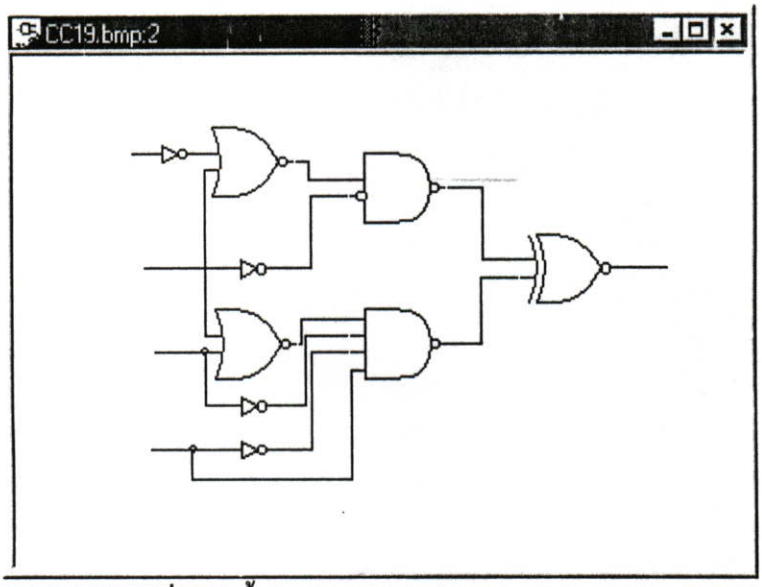


รูปที่ 4.11 รูปแบบการลากเส้นต่อวงจร

ตารางที่ 4.1 การเลือกใช้รูปแบบการลากเส้นต่อวงจร

โลจิกเกททิศทาง	โลจิกเกททิศทาง	ใช้รูปแบบลายเส้น
0	0	A,D,B,C,E,F,G
0	90	A,B,C,E,D,F,G
0	180	A,B,C,E,D,F,G
0	270	A,B,C,E,D,F,G
90	90	A,E,C,B,D,G,F
90	180	A,B,C,E,D,F,G
90	270	A,C,B,D,E,G,F
180	180	A,D,B,C,E,F,G
180	270	A,B,C,E,D,F,G
270	270	A,E,C,B,D,G,F

โลจิกเกททิศทางใดๆ	จุดต่อร่วม	A,B,C,D,E,F,G
จุดต่อร่วม	จุดต่อร่วม	A,B,C,D,E,F,G



รูปที่ 4.12 ภาพวงจรโลจิกเกทที่สร้างขึ้นใหม่

บทที่ 5

ผลการทดลอง

เนื่องจากกระบวนการจดจำวงจรโลจิกเกทเกทที่เขียนด้วยมือโดยใช้หลักการแบ่งแยกลายเส้นที่นำเสนอในวิทยานิพนธ์ฉบับนี้ประกอบด้วยขั้นตอนต่างๆ หลายขั้นตอน ซึ่งแต่ละขั้นตอนมีความสำคัญและมีผลต่อการจดจำวงจรโลจิก ดังนั้นในการทดลองเพื่อตรวจสอบความถูกต้องของขั้นตอนต่างๆ จึงแบ่งการทดลองออกเป็น 2 ส่วนคือ ส่วนแรกจะเป็นการทดลองการวิเคราะห์โครงสร้างของโลจิกเกท ส่วนที่สองเป็นการทดลองการหาอินพุท-เอาท์พุท และการสร้างภาพวงจรขึ้นใหม่ ซึ่งจะเป็นผลการทดลองการจดจำวงจรโลจิกเกทที่เขียนด้วยมือทั้งระบบ รายละเอียดของแต่ละส่วนมีดังนี้

5.1 การทดลองการวิเคราะห์โครงสร้างของโลจิกเกท

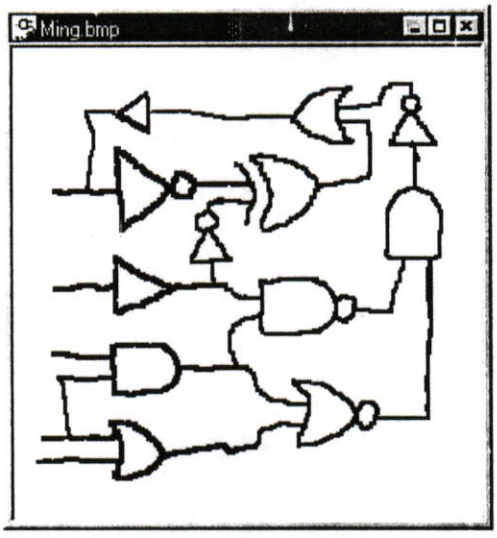
การทดลองการวิเคราะห์โครงสร้างของโลจิกเกทนี้ เพื่อตรวจสอบผลของแต่ละขั้นตอนในการวิเคราะห์โครงสร้างว่ามีความถูกต้องหรือผิดพลาดอย่างไร ซึ่งมีรายละเอียดของการทดลองดังต่อไปนี้

5.1.1 การเก็บภาพวงจรโลจิกเกทเข้าคอมพิวเตอร์

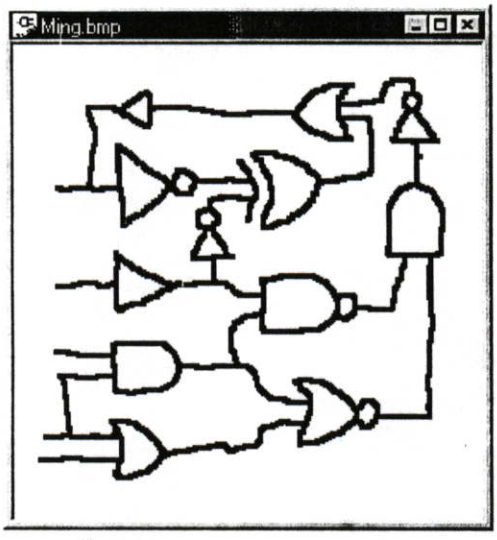
ภาพวงจรโลจิกเกทที่ใช้ในการทดลองเป็นภาพที่เขียนขึ้นด้วยดินสอและปากกาที่ใช้กันอยู่ทั่วไป โดยมีรูปแบบของวงจรหลายรูปแบบ มีขนาดและทิศทางของโลจิกเกทต่างๆ กัน และเก็บข้อมูลเข้าเครื่องคอมพิวเตอร์ด้วยสแกนเนอร์ (Scanner) โดยสแกนเป็นภาพชนิดภาพวาดขาว-ดำ (Back-White drawing) ที่ความละเอียดต่างๆ กัน และบันทึกเป็นไฟล์รูปภาพแบบ Bitmap (ไฟล์นามสกุล BMP)

5.1.2 ผลการวิเคราะห์โครงสร้างของโลจิกเกท

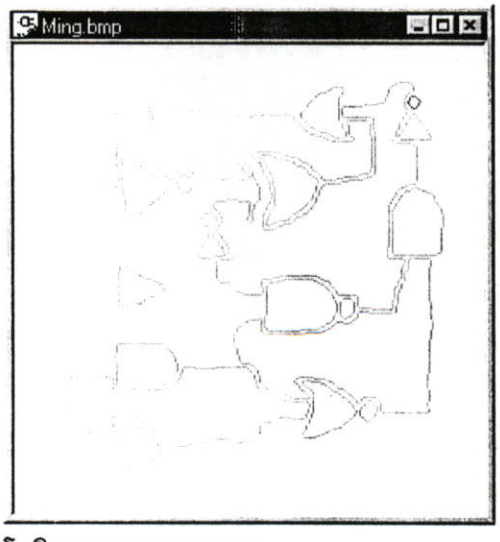
ในการทดลองการจดจำวงจรโลจิกเกทจะใช้โปรแกรม Microsoft Visaul C++ เขียนโปรแกรมเพื่อทดสอบอัลกอริทึมต่างๆ ทั้งหมด ซึ่งภาพอินพุทที่เก็บไว้จะนำมาประมวลผลในขั้นตอนต่างๆ ของการวิเคราะห์โครงสร้าง โดยกำหนดค่าสเกล-สเปซและค่า Threshold ในขั้นตอนการหาจุดแบ่งแยกลายเส้นเท่ากับ 3 และ 0.035 ตามลำดับ และมีผลการทดลองของขั้นตอนต่างๆ ดังแสดงไว้ในรูปที่ 5.1 ถึง 5.20



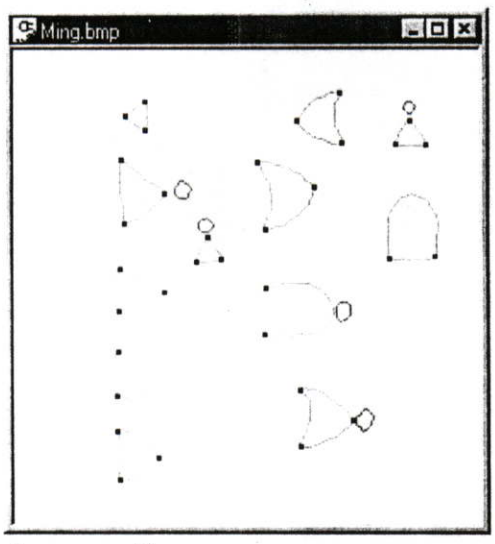
รูปที่ 5.1 ภาพข้อมูลอินพุทขนาด 256x256 จุดภาพ



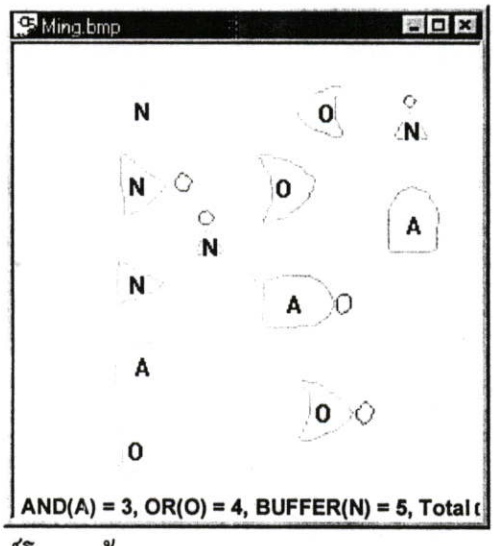
รูปที่ 5.2 ผลของการปรับปรุงลายเส้น



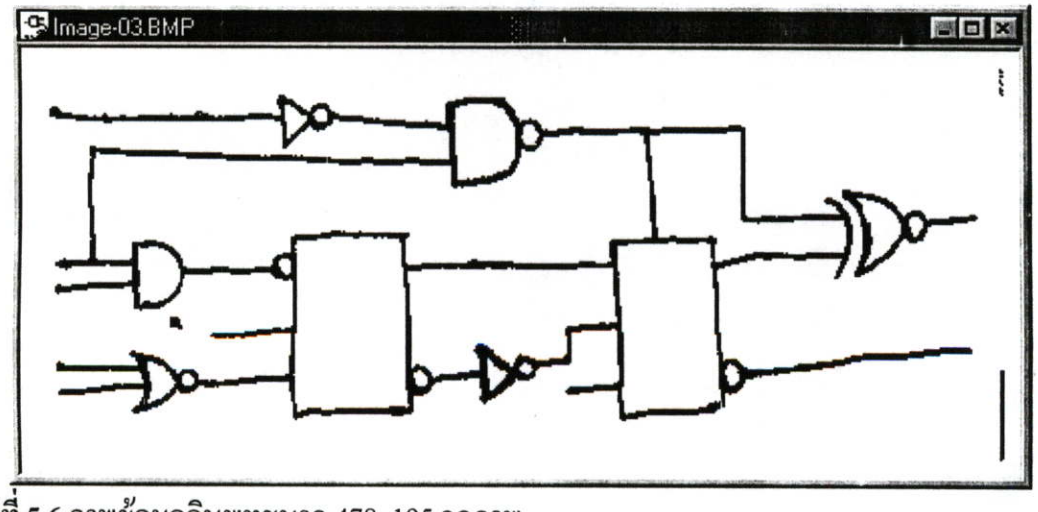
รูปที่ 5.3 ผลของการแยกตัวโลจิกเกทออกจากวงจร



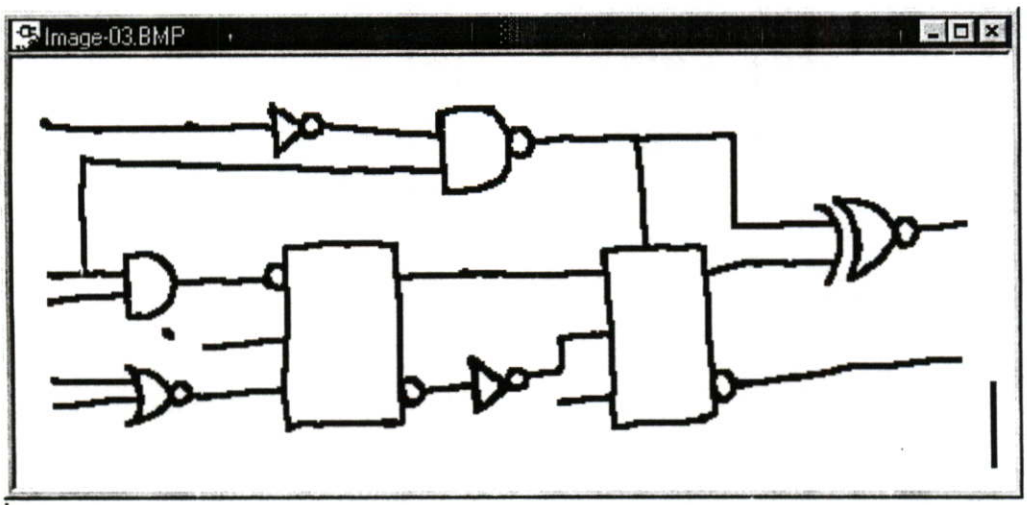
รูปที่ 5.4 ผลของการหาจุดแบ่งแยกลายเส้น



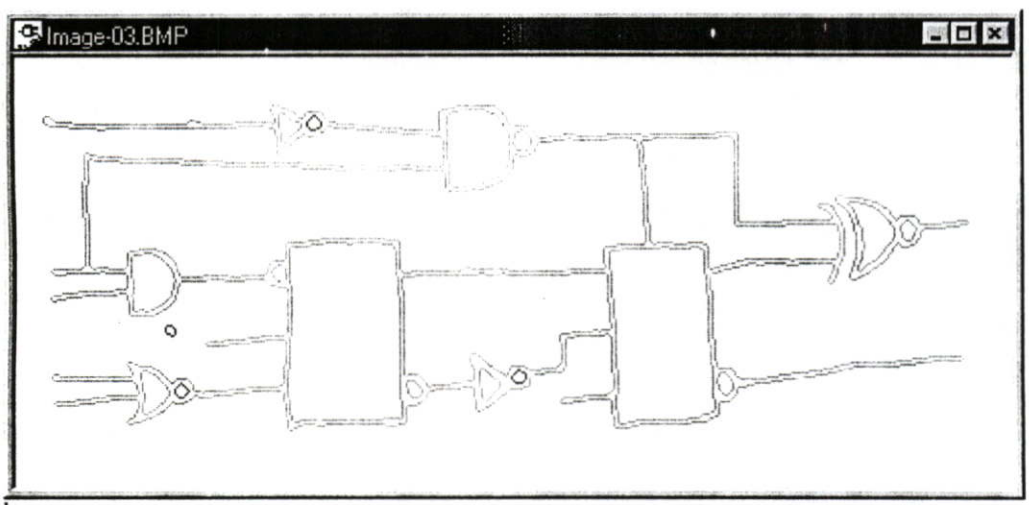
รูปที่ 5.5 ผลของการวิเคราะห์โครงสร้าง



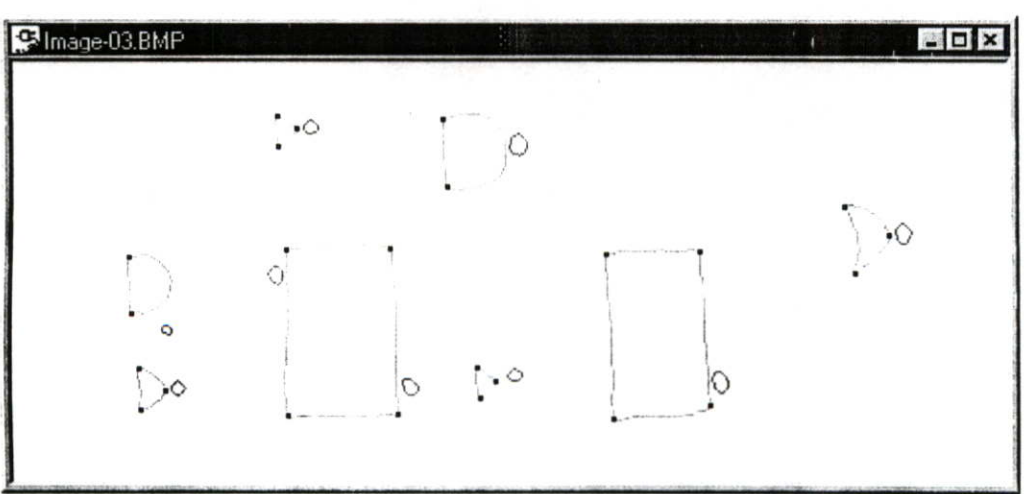
รูปที่ 5.6 ภาพข้อมูลอินพุทขนาด 478x195 จุดภาพ



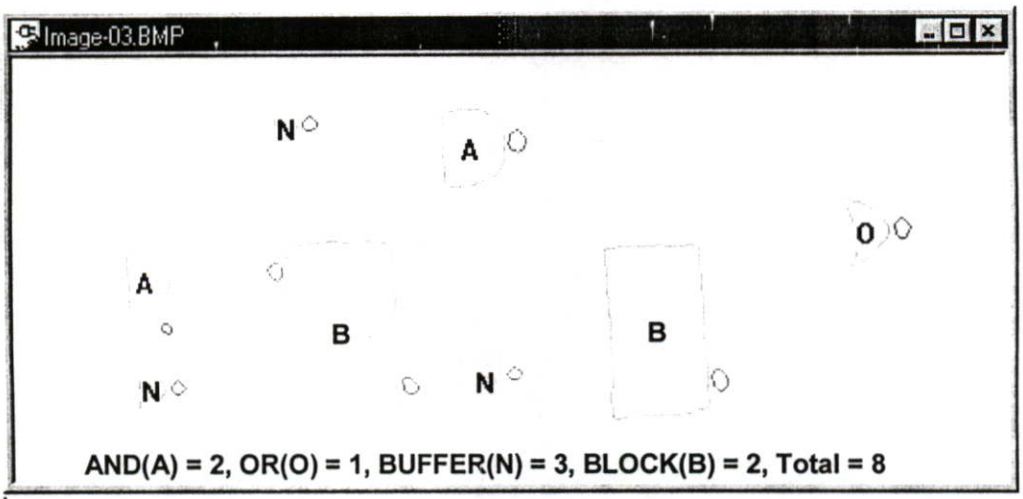
รูปที่ 5.7 ผลของการปรับปรุงลายเส้น



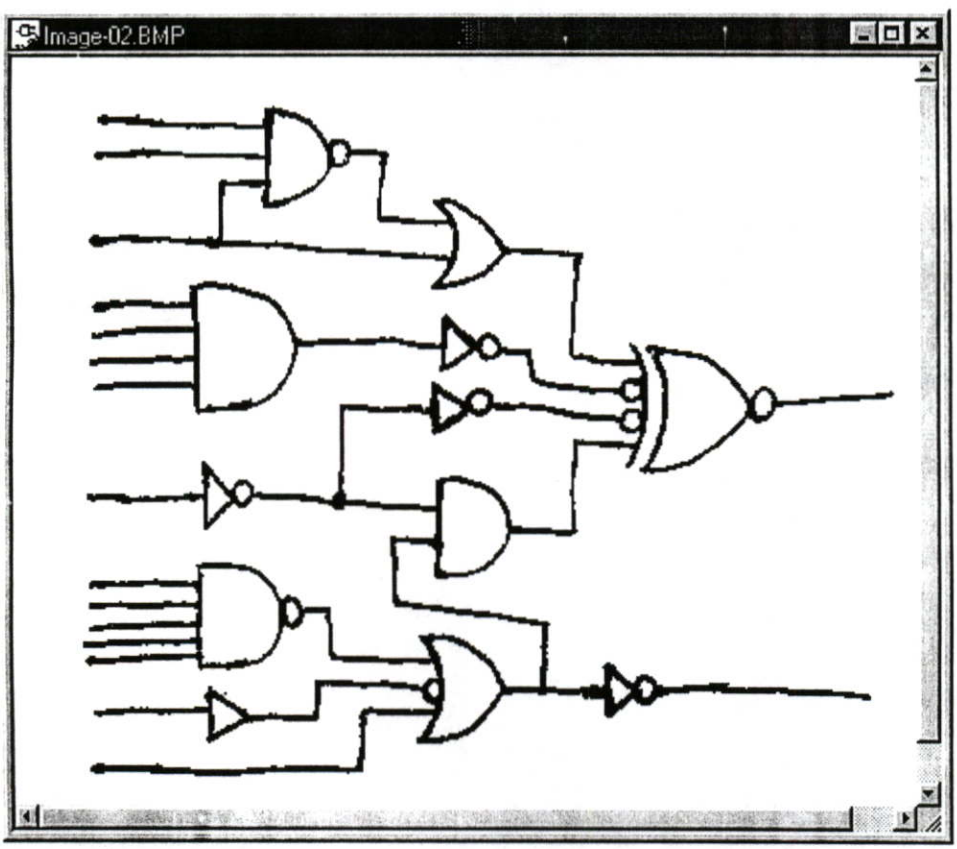
รูปที่ 5.8 ผลของการแยกตัวโลจิกเกทออกจากวงจร



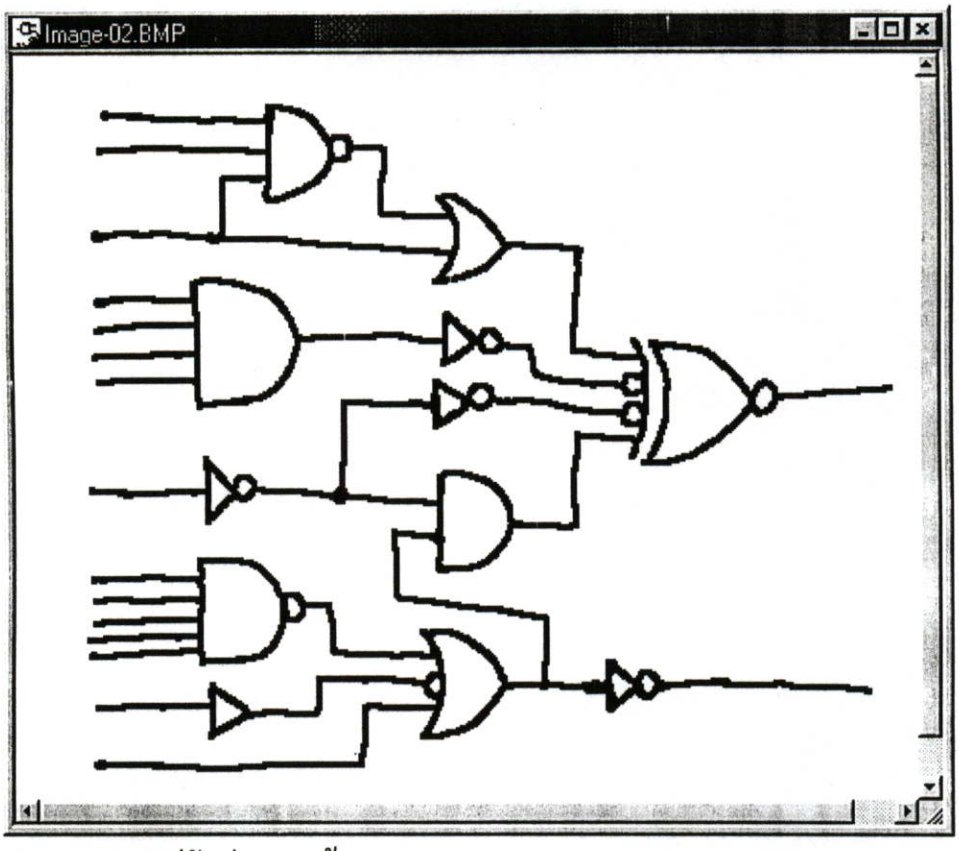
รูปที่ 5.9 ผลของการหาจุดแบ่งแยกลายเส้น



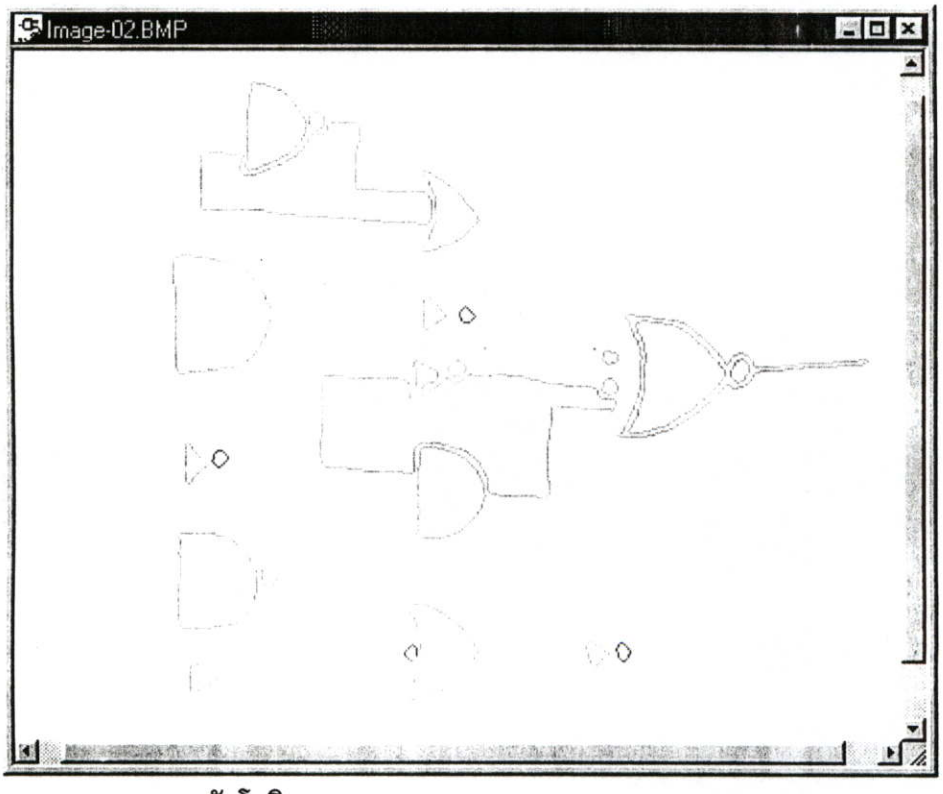
รูปที่ 5.10 ผลของการวิเคราะห์โครงสร้าง



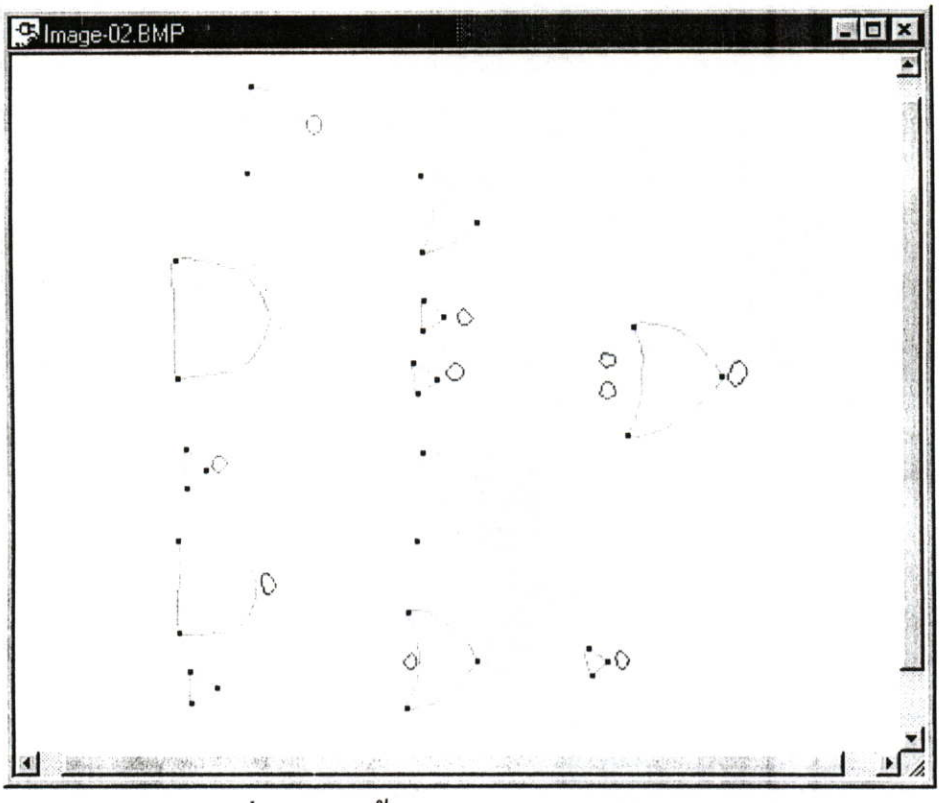
รูปที่ 5.11 ภาพข้อมูลอินพุทขนาด 453x373 จุดภาพ



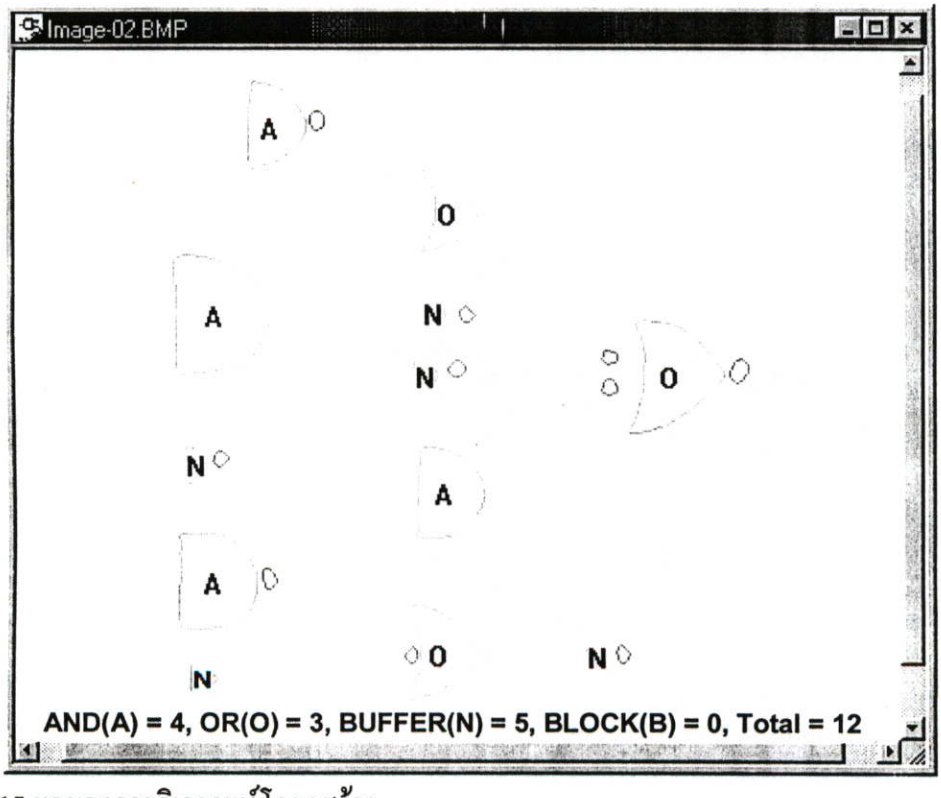
รูปที่ 5.12 ผลของการปรับปรุงลายเส้น



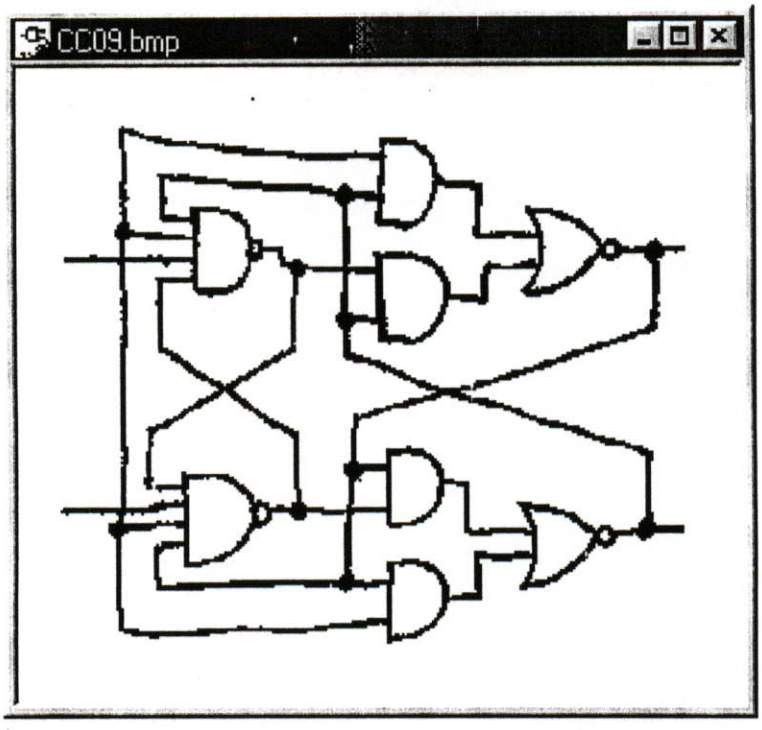
รูปที่ 5.13 ผลของการแยกตัวโลจิกเกทออกจากวงจร



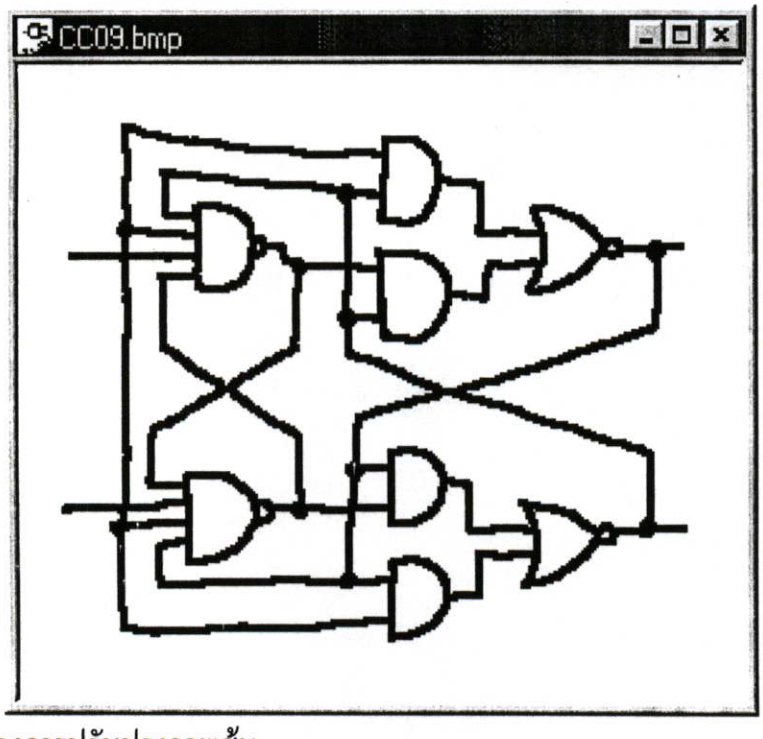
รูปที่ 5.14 ผลของการหาจุดแบ่งแยกลายเส้น



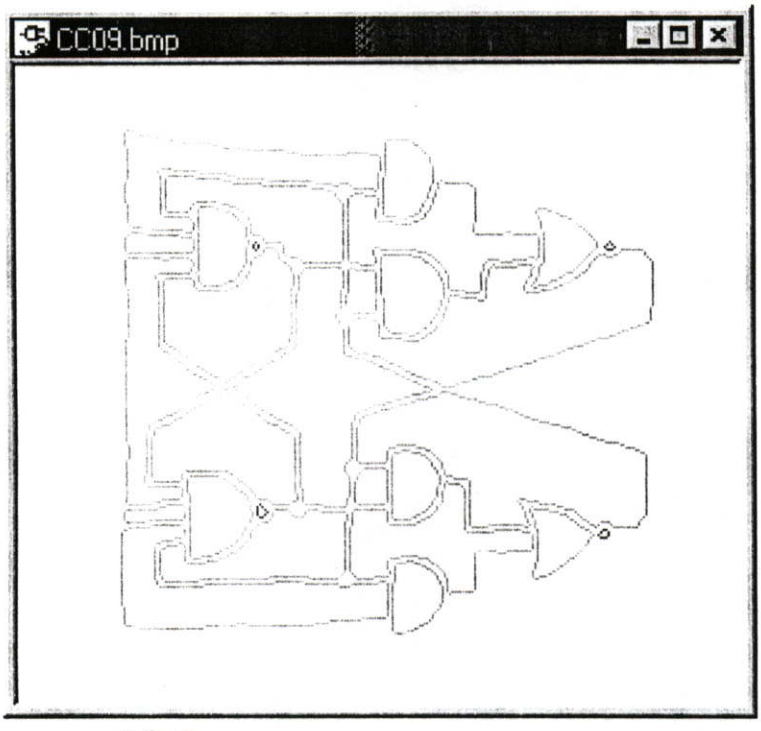
รูปที่ 5.15 ผลของการวิเคราะห์โครงสร้าง



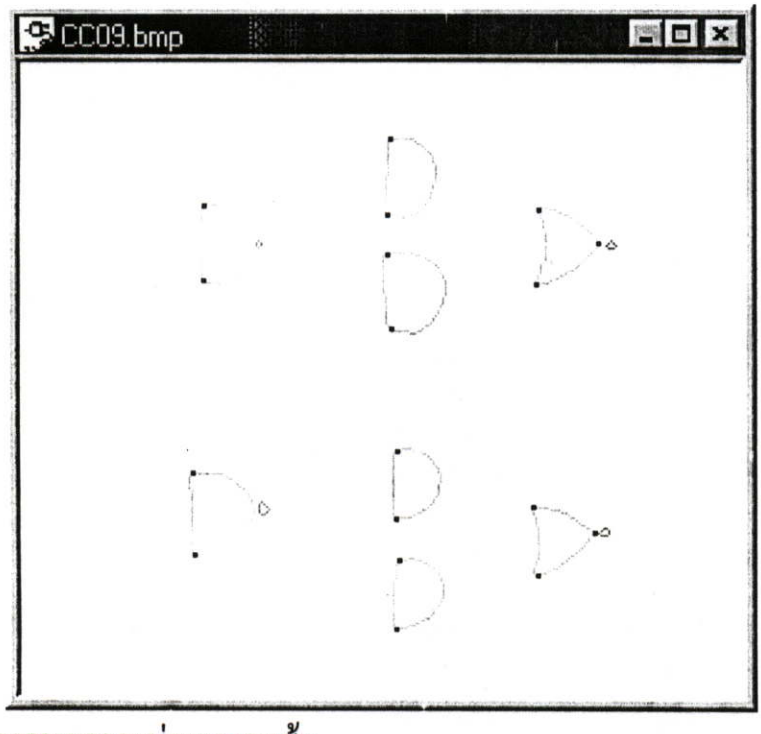
รูปที่ 5.16 ภาพข้อมูลอินพุทขนาด 272x234 จุดภาพ



รูปที่ 5.17 ผลของการปรับปรุงลายเส้น



รูปที่ 5.18 ผลของการแยกตัวโลจิกเกทออกจากวงจร



รูปที่ 5.19 ผลของการหาจุดแบ่งแยกลายเส้น

รูปที่ 5.20 ผลของการวิเคราะห์โครงสร้าง

การทดลองการวิเคราะห์โครงสร้างของโลจิกเกทนี้ ได้ทดลองกับภาพวงจรโลจิกเกท จำนวน 163 ภาพ โดยมีจำนวนตัวโลจิกเกทชนิดต่างๆ รวม 837 ตัว และมีผลการทดลองดังแสดงในตารางที่ 5.1

ตารางที่ 5.1 ผลการทดลองการวิเคราะห์โครงสร้างของโลจิกเกท

โลจิกเกทโครงสร้าง	ถูกต้อง		ผิดพลาด		รวม
	ตัว	%	ตัว	%	ตัว
AND	233	97.0	7	3.0	240
OR	227	90.8	23	9.2	250
BUFFER	199	90.9	20	9.1	219
BLOCK	115	89.9	13	10.1	128
รวม	774	92.5	63	7.5	837

ความผิดพลาดของการวิเคราะห์โครงสร้างของโลจิกเกทที่พบคือ การหาจุดแบ่งแยกลายเส้นไม่ถูกต้องบริเวณเอาท์พุทของโลจิกเกทโครงสร้าง OR มีมุมไม่เด่นชัด หรือมีลักษณะเป็นเส้นโค้งจึงหาจุดมุมไม่พบ การวิเคราะห์ลักษณะลายเส้นไม่ถูกต้องเนื่องจากจำนวนจุดบนลายเส้นมีน้อย (โลจิกเกทมีขนาดเล็ก) และการเขียนเส้นตรงและเส้นโค้งของตัวโลจิกเกทไม่เด่นชัดทำให้การ

วิเคราะห์โครงสร้างผิดพลาดดังรูปที่ 5.20 วิเคราะห์โลจิกเกทโครงสร้าง OR เป็นโครงสร้าง BUFFER (ข้อผิดพลาดนี้จะถูกแก้ในขั้นตอนการหาอินพุท) เป็นต้น

5.1.3 สรุปผลการวิเคราะห์โครงสร้างของโลจิกเกท

การทดลองขั้นตอนต่างๆ ในการวิเคราะห์โครงสร้างของโลจิกเกทสามารถสรุปผลของการทดลองได้ดังนี้

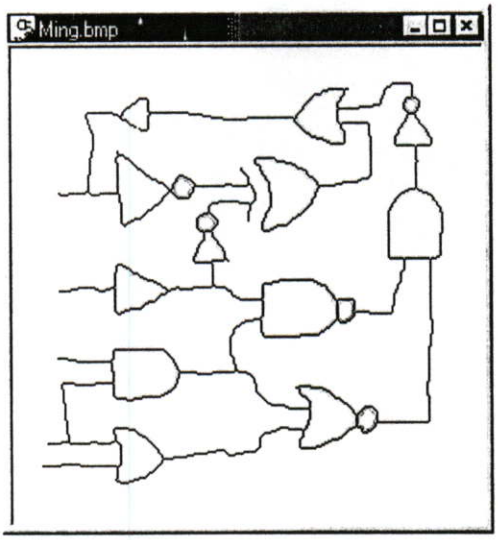
1) ขั้นตอนการปรับปรุงลายเส้น สามารถปรับปรุงคุณภาพของลายเส้นได้ผลดี โดยลายเส้นเมื่อผ่านการปรับปรุงจะมีความหนามากกว่า 3 จุดภาพขึ้นไป การขาดของลายเส้นเล็กๆ น้อยๆ และหลุมบนลายเส้นที่มีขนาดน้อยกว่า 3 จุดภาพจะถูกต่อและเติมให้เต็ม ซึ่งมีผลทำให้การแยกตัวโลจิกเกทออกจากวงจรสำเร็จ แม้ว่าภาพข้อมูลอินพุทจะมีความหนาของลายเส้นน้อยกว่า 3 จุดภาพ มีการขาดของลายเส้นเล็กน้อยก็ตาม และเป็นการกำจัดจุดแยกที่ไม่ถูกต้องบนเส้นโครงร่างที่เกิดจากหลุมบนลายเส้น ซึ่งวิธีการที่นำเสนอจะช่วยลดความผิดพลาดในการวิเคราะห์การต่อวงจรได้
2) ขั้นตอนการหาจุดมุมแบ่งแยกลายเส้น สามารถหาจุดมุมแบ่งแยกลายเส้นได้อย่างมีประสิทธิภาพ โดยสามารถหาจุดมุมของโลจิกเกทที่ขนาดแตกต่างกันมาก (ในภาพเดียวกัน) ได้ถูกต้องดังจะเห็นได้จากผลการทดลอง (โลจิกเกทโครงสร้าง BLOCK กับ โลจิกเกทโครงสร้าง BUFFER) เนื่องจากการกำหนดค่าสเกล-สเปซและค่า Threshold มีความยืดหยุ่น (การกำหนดค่าสเกล-สเปซ และค่า Threshold หนึ่งค่าสามารถใช้กับโลจิกเกทได้หลายขนาด)
3) ขั้นตอนการวิเคราะห์ลักษณะลายเส้น สามารถวิเคราะห์ลักษณะเส้นตรงและเส้นโค้งได้ผลดี และมีเงื่อนไขการวิเคราะห์ที่ชัดเจน

5.2 การทดลองการหาอินพุท-เอาท์พุท และการสร้างภาพวงจรขึ้นใหม่

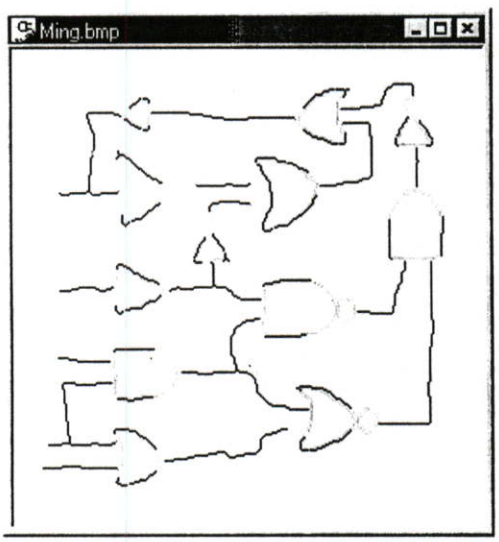
การทดลองนี้เป็นการตรวจสอบผลของการหาอินพุท-เอาท์พุท การวิเคราะห์การต่อวงจร และการสร้างภาพวงจรขึ้นใหม่ โดยมีรายละเอียดของการทดลองดังนี้

5.2.1 ผลการหาอินพุท-เอาท์พุท และการสร้างภาพวงจรขึ้นใหม่

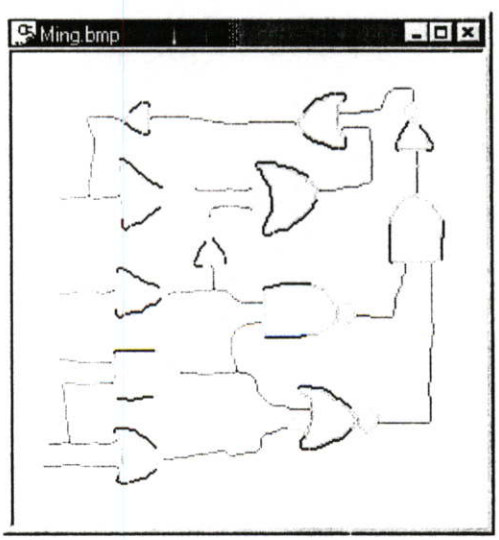
ภาพวงจรโลจิกเกทที่ใช้ในขั้นตอนนี้เป็นภาพที่ผ่านการวิเคราะห์โครงสร้างมาแล้ว ซึ่งเป็นการทดลองต่อจากส่วนแรก โดยจะประมวลผลทีละขั้นตอนและมีผลของการทดลองแต่ละขั้นตอนดังแสดงในรูปที่ 5.21 ถึง 5.36



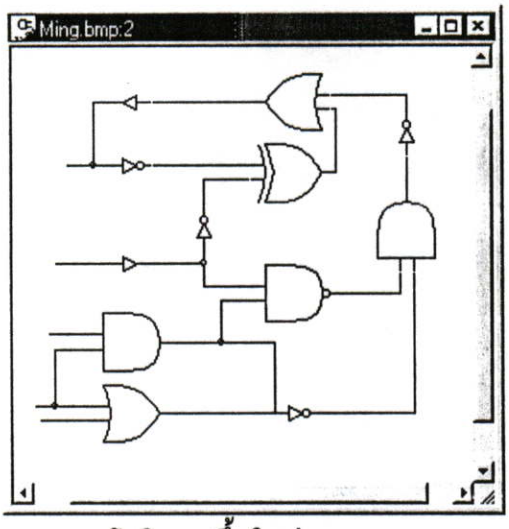
รูปที่ 5.21 ผลของการหาลายเส้นโครงร่าง



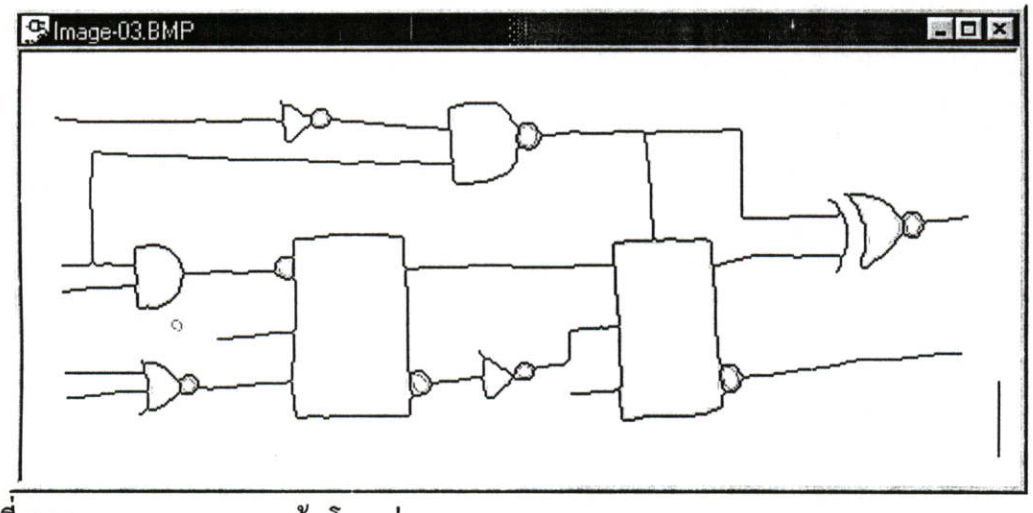
รูปที่ 5.22 ผลของการหาอินพุท-เอาท์พุท



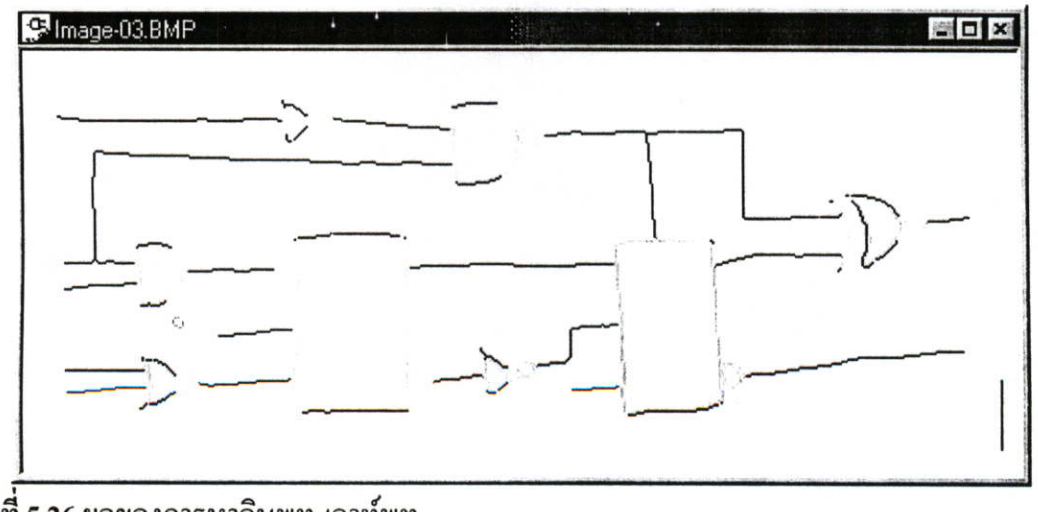
รูปที่ 5.23 ผลของการวิเคราะห์การต่อวงจร



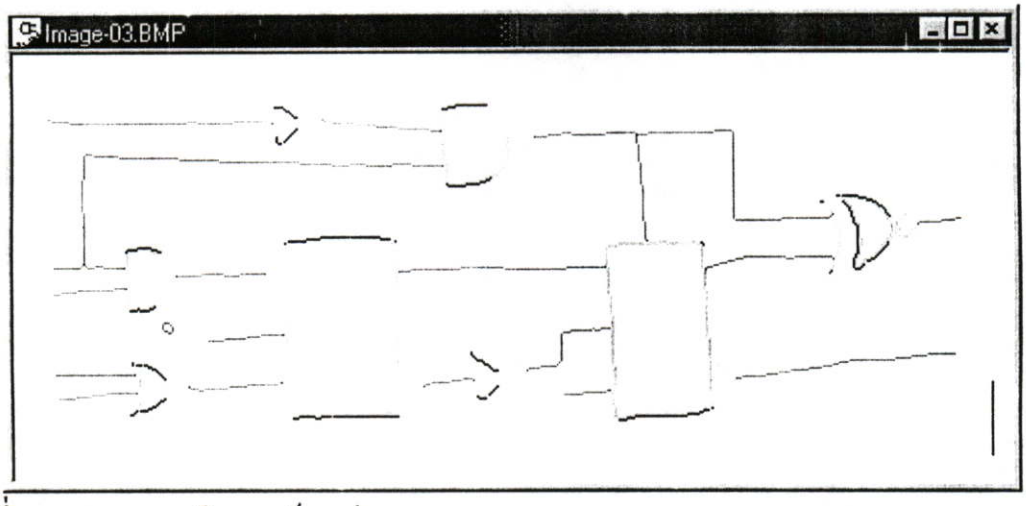
รูปที่ 5.24 ผลของการสร้างภาพวงจรโลจิกเกทขึ้นใหม่



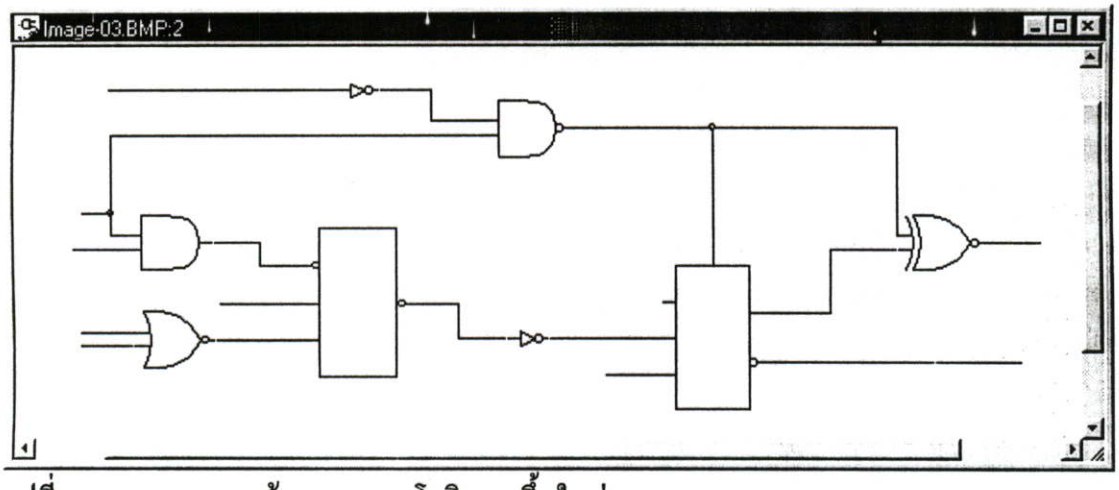
รูปที่ 5.25 ผลของการหาลายเส้นโครงร่าง



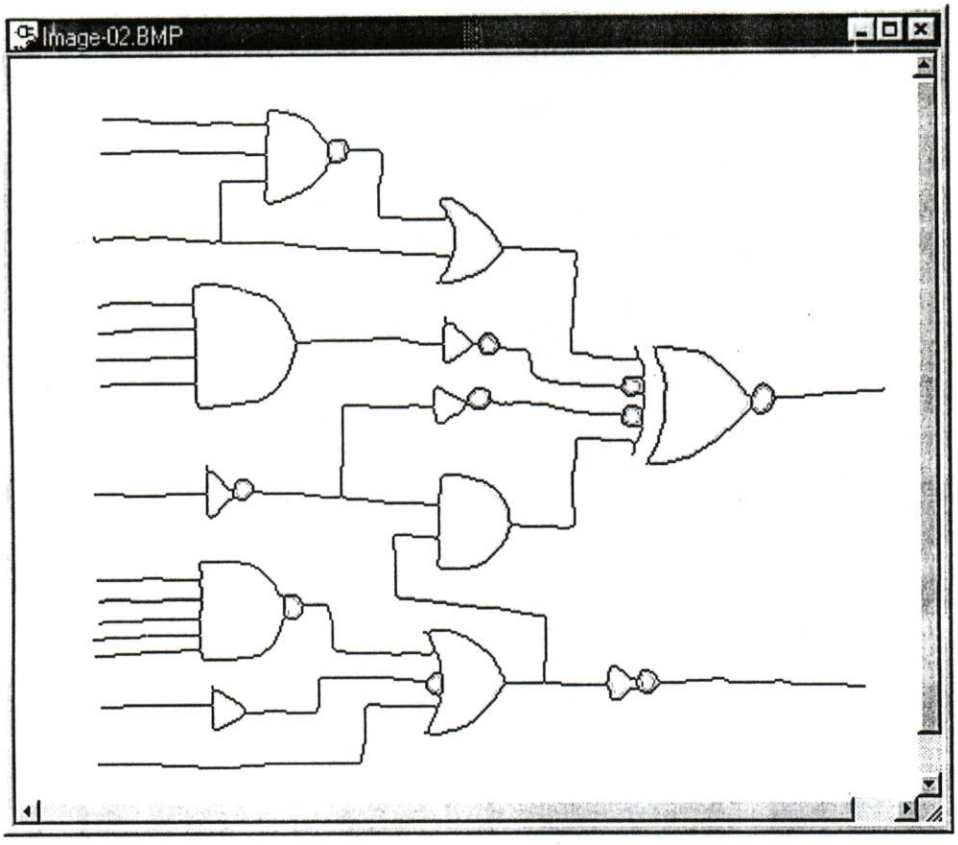
รูปที่ 5.26 ผลของการหาอินพุท-เอาท์พุท



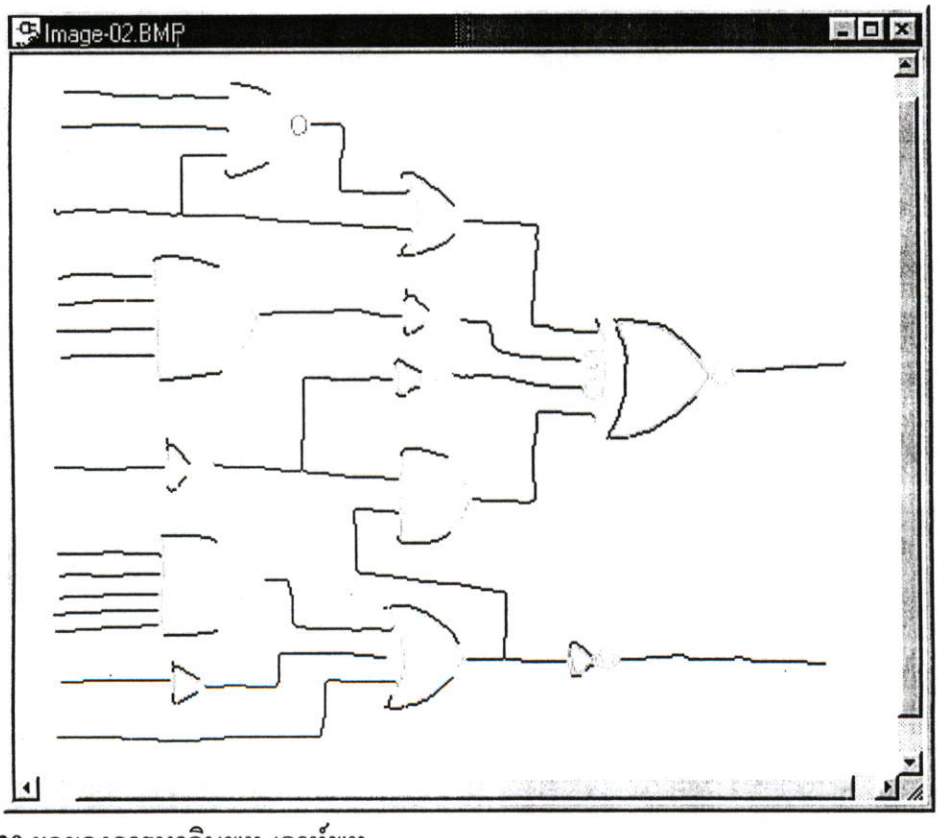
รูปที่ 5.27 ผลของการวิเคราะห์การต่อวงจร



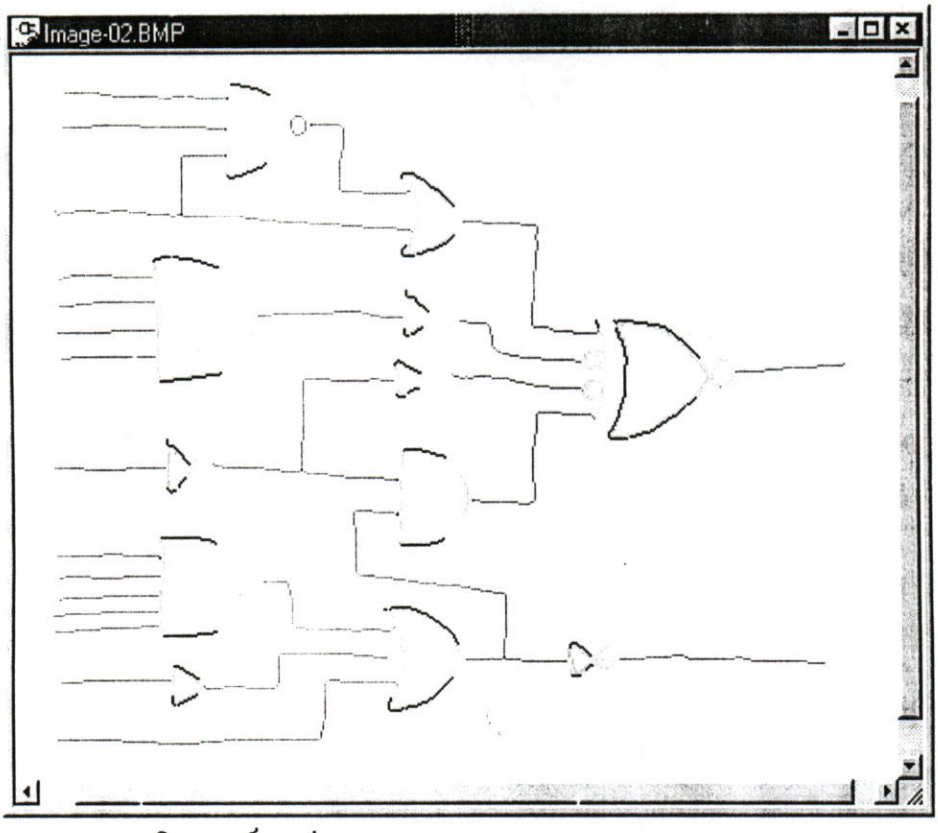
รูปที่ 5.28 ผลของการสร้างภาพวงจรโลจิกเกทขึ้นใหม่



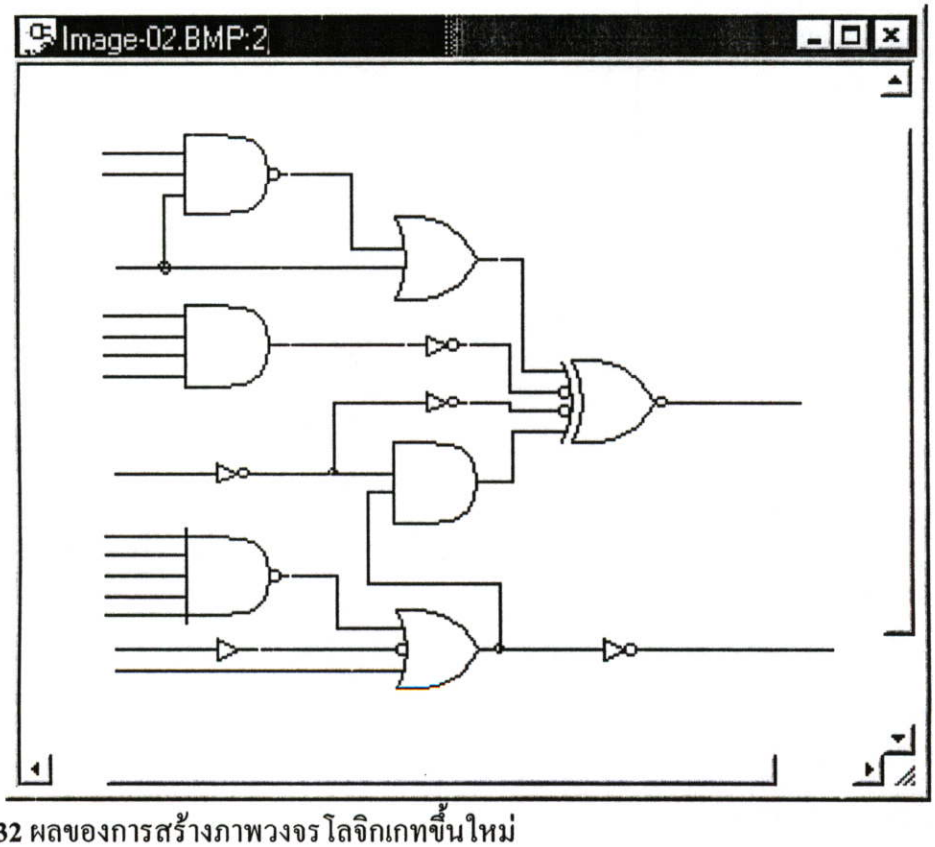
รูปที่ 5.29 ผลของการหาลายเส้นโครงร่าง



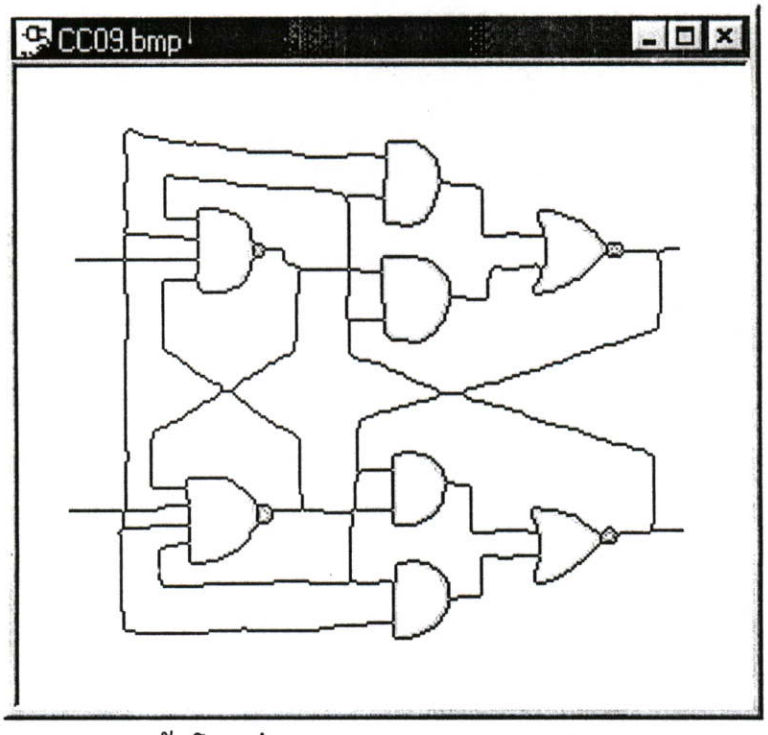
รูปที่ 5.30 ผลของการหาอินพุท-เอาท์พุท



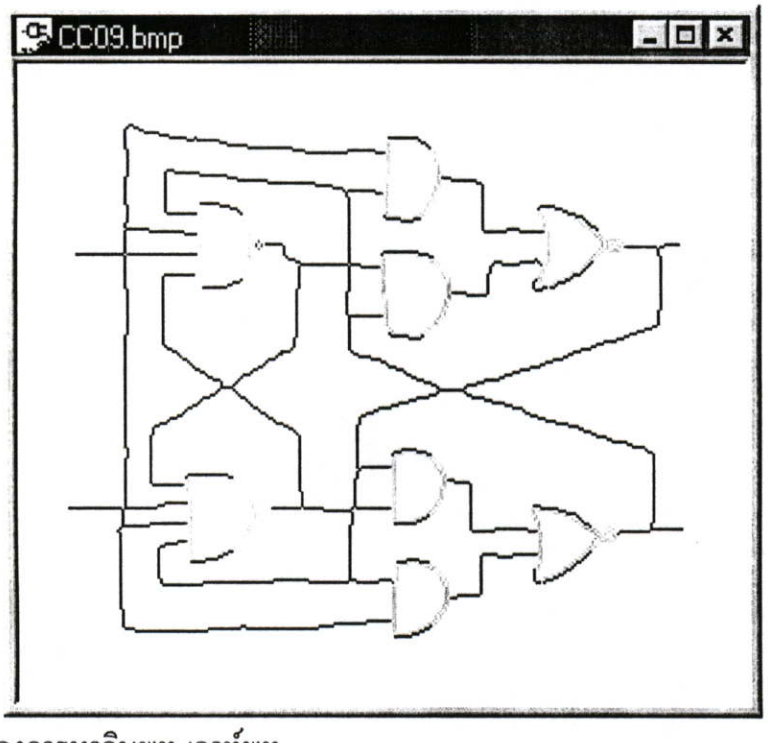
รูปที่ 5.31 ผลของการวิเคราะห์การต่อวงจร



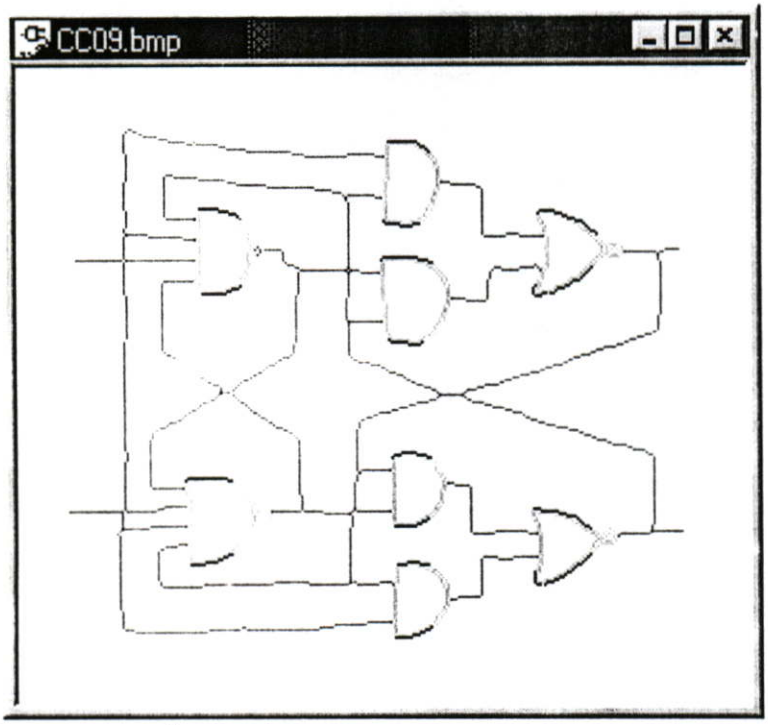
รูปที่ 5.32 ผลของการสร้างภาพวงจรโลจิกเกทขึ้นใหม่



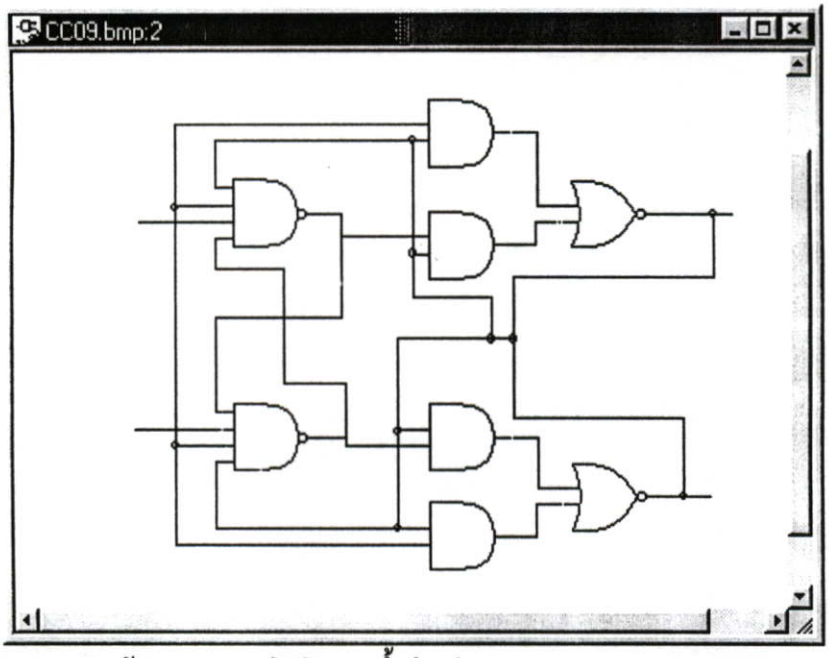
รูปที่ 5.33 ผลของการหาลายเส้นโครงร่าง



รูปที่ 5.34 ผลของการหาอินพุท-เอาท์พุท



รูปที่ 5.35 ผลของการวิเคราะห์วงจร



รูปที่ 5.36 ผลของการสร้างภาพวงจรโลจิกเกทขึ้นใหม่

เมื่อทดลองกับภาพทั้ง 163 ภาพแล้ว ให้ผลลัพธ์ในการหาอินพุท-เอาท์พุท การวิเคราะห์การต่อวงจร และการสร้างภาพวงจรโลจิกเกทขึ้นใหม่ได้ถูกต้อง 141 ภาพ หรือ 86.5 % ผิดพลาด 22 ภาพ หรือ 13.5 % ซึ่งความผิดพลาดที่พบคือ อินพุทของโลจิกเกทโครงสร้าง OR อยู่ใกล้จุดมุมทำให้ตรวจพบว่าเป็นเอาท์พุท โลจิกเกทมีวงกลม NOT ที่อินพุทอยู่ติดกันทำให้หาอินพุทผิดพลาด และการต่อวงจรไม่สำเร็จ ณ จุดต่อร่วมที่มี 4 สาขา เป็นต้น

5.2.2 สรุปผลการหาอินพุท-เอาท์พุท และการสร้างภาพวงจรโลจิกเกทขึ้นใหม่

การทดลองผลของขั้นตอนต่างๆ ตั้งแต่การหาอินพุท-เอาท์พุทของโลจิกเกท ไปจนถึงการสร้างภาพวงจรโลจิกเกทขึ้นใหม่ สามารถสรุปผลการทดลองได้ดังต่อไปนี้

1) ขั้นตอนการหาอินพุท-เอาท์พุท การหาเอาท์พุทของโลจิกเกทโครงสร้าง AND และ BUFFER จะมีความยืดหยุ่นและความถูกต้องสูง ขณะที่โลจิกเกทโครงสร้าง OR จะมีความยืดหยุ่นน้อย เพราะการกำหนดขอบเขตของการหาเอาท์พุทจะต้องคำนึงถึงการเขียนอินพุทใกล้จุดมุมด้วย เนื่องจากการหาเอาท์พุทจะหาที่จุดมุมทั้งสามจุด (ทีละจุด) ในส่วนของการหาอินพุทจะมีผลกระทบโดยตรงจากการหาเอาท์พุท (การกำหนดจุดเริ่มต้นของการหาอินพุทขึ้นอยู่กับตำแหน่งเอาท์พุท) ดังนั้นถ้าการหาเอาท์พุทถูกต้องการหาอินพุทจะมีความถูกต้องสูงเช่นกัน และกรณีของโลจิกเกทโครงสร้าง OR และ BUFFER การหาอินพุทสามารถตรวจสอบความผิดพลาดของการหาเอาท์พุทได้ เนื่องจากการหาอินพุทไม่พบจะทำการหาเอาท์พุทใหม่ที่จุดมุมอื่นๆ นอกจากนี้จำนวนอินพุทที่พบจะช่วยในการจดจำชนิดที่ถูกต้องของโลจิกเกทได้ดังตัวอย่างในรูปที่ 5.20 และรูปที่ 5.36
2) ขั้นตอนการวิเคราะห์การต่อวงจร เป็นการตรวจสอบการต่อวงจรโดยใช้อัลกอริทึมในการติดตามลายเส้นที่ไม่ซับซ้อนความผิดพลาดจึงน้อยมาก
3) ขั้นตอนการสร้างภาพวงจรโลจิกเกทขึ้นใหม่เป็นขั้นตอนสุดท้าย ซึ่งจากการทดลองการวาดภาพโลจิกเกทในตำแหน่งและทิศทางใหม่มีความถูกต้องมาก แต่การลากเส้นต่อวงจรยังมีข้อจำกัด ซึ่งบางภาพต่อวงจรไม่เหมาะสมดังตัวอย่างในรูปที่ 5.40 เนื่องจากจุดต่อร่วมไม่มีทิศทางจึงไม่สามารถกำหนดรูปแบบการลากเส้นต่อวงจรที่เหมาะสมได้

บทที่ 6
บทสรุปและข้อเสนอแนะ

วิทยานิพนธ์ฉบับนี้ได้นำเสนอ กระบวนการจดจำวงจรโลจิกเกทที่เขียนด้วยมือโดยใช้หลักการแบ่งแยกลายเส้น เป็นหลักการวิเคราะห์โครงสร้างของวงจรโลจิกเกท โดยพิจารณาจากองค์ประกอบของลายเส้นหลายๆ เส้นที่ประกอบกันเป็นตัวโลจิกเกท ซึ่งเป็นเพียงแนวความคิดหนึ่งเท่านั้น เพื่อให้สอดคล้องกับหลักการและแนวความคิดที่กล่าวไว้ข้างต้น ข้อมูลภาพวงจรโลจิกเกทที่เขียนด้วยมือที่ได้จากเครื่องสแกนเนอร์จะนำมาหาเส้นขอบ เส้นขอบวงปิดแต่ละเส้นจะถูกวิเคราะห์ว่าเป็นตัวโลจิกเกทหรือไม่ ถ้าเป็นตัวโลจิกเกทมันเป็นโลจิกเกทที่จัดอยู่ในประเภทใดตามโครงสร้างหลัก 4 ชนิดที่กำหนดไว้ ซึ่งการวิเคราะห์ดังกล่าวจะใช้วิธีการหาค่าความโค้งและการทำโค้งกระชับ อย่างไรก็ตามเส้นขอบวงปิดแต่ละเส้นจะมีสัญญาณรบกวนปะปนอยู่ และอาจมีความหนาของลายเส้นไม่เท่ากัน มีการขาดของลายเส้น มีหลุมเล็กๆ ภายในลายเส้นเป็นต้น ซึ่งสิ่งเหล่านี้เป็นตัวแปรที่สำคัญอันหนึ่งที่ทำให้การวิเคราะห์ลายเส้นเพื่อการจดจำและแยกประเภทโลจิกเกทตามโครงสร้างหลักผิดพลาด เราได้แก้ไขปัญหาดังกล่าวด้วยการปรับปรุงคุณภาพของลายเส้นให้ดีขึ้นทำการกำจัดสัญญาณรบกวนบนลายเส้นด้วยตัวกรองเกาซ์เซียนและได้กำหนดเงื่อนไขการจดจำตัวโลจิกเกทที่มีความยืดหยุ่นมากขึ้น ซึ่งผลลัพธ์ที่ได้ทำให้เราสามารถวิเคราะห์โครงสร้างของตัวโลจิกเกทถูกต้องมากยิ่งขึ้น อย่างไรก็ตามการวิเคราะห์โครงสร้างหลักนี้อาจมีความผิดพลาดได้เช่นกัน ดังแสดงไว้ในบทที่ 5 หลังจากการวิเคราะห์โครงสร้างหลักเสร็จสิ้น เราได้ดำเนินการหาเส้นโครงร่างของภาพ เพื่อวิเคราะห์หาจำนวนอินพุท-เอาท์พุทของโลจิกเกทแต่ละตัวด้วยวิธีการติดตามลายเส้นและใช้คุณสมบัติโทโปโลยี สำหรับการวิเคราะห์หาจำนวนอินพุท-เอาท์พุทนี้เราได้นำเสนออัลกอริทึมที่ไม่ซับซ้อนสามารถตรวจหาวงกลม NOT ที่อินพุทและเอาท์พุทได้ด้วยอัลกอริทึมเดียวกัน ภายหลังการวิเคราะห์หาอินพุท-เอาท์พุทเสร็จสิ้น เราได้ใช้จำนวนของอินพุท-เอาท์พุทที่ตรวจพบมาพิจารณาร่วมกับโครงสร้างของโลจิกเกทอีกครั้ง ทำให้เราสามารถรู้จำชนิดของโลจิกเกททั้ง 9 ชนิดได้ถูกต้องแม่นยำมากยิ่งขึ้น จากนั้นทำการวิเคราะห์การต่อลายวงจรระหว่างโลจิกเกทกับโลจิกเกทและโลจิกเกทกับจุดต่อร่วม ทำการเขียนวงจรโลจิกเกทขึ้นมาใหม่ด้วยการหาทิศทางหลักของวงจร ทำการย่อขยายตัวโลจิกเกทให้มีขนาดตามมาตราฐานที่ได้กำหนดไว้ จากผลการทดลองที่แสดงไว้ในบทที่ 5 ได้แสดงให้เห็นว่าอัลกอริทึมที่นำเสนอสามารถจดจำวงจรโลจิกเกทที่เขียนด้วยมือได้

ข้อเสนอแนะ

เนื่องจากการจดจำวงจรโลจิกเกทที่เขียนด้วยมือนี้จะใช้เฉพาะวงจรโลจิกเกทที่มีสัญลักษณ์เหมือนกับที่นำเสนอเท่านั้น แต่ถ้าเป็นสัญลักษณ์มาตราฐานอื่นยังไม่สามารถจดจำได้ รวมทั้งอัลกอ

ริทึมที่นำเสนอยังมีข้อจำกัด จึงควรจะมีการศึกษาวิจัยและพัฒนาต่อไป เพื่อให้การจดจำวงจรโลจิกเกทที่เขียนด้วยมือมีความสมบูรณ์มากยิ่งขึ้น โดยมีแนวทางการพัฒนาดังต่อไปนี้

1) ลดความซับซ้อนในการต่อลายวงจร เพื่อให้การต่อลายวงจรมีความสวยงามและเหมาะสมมากยิ่งขึ้น
2) พัฒนาให้สามารถจำลองการทำงาน (Simulation) ของวงจรโลจิกเกทที่จดจำได้
3) เพิ่มความสามารถในการจดจำตัวอักษรที่เขียนด้วยมือที่มีอยู่ในวงจร
4) เพิ่มความสามารถในการจดจำอุปกรณ์อิเลคโทรนิคส์อื่นๆ เช่น ตัวต้านทาน (Resister), ตัวเก็บประจุ (Capacitor) และทรานซิสเตอร์ (Transistor) เป็นต้น

เอกสารอ้างอิง

[1] เกษตร์ ศิริสันติสัมฤทธิ์ และ อรุณ กำเหนิดนนท์, "การจดจำโลจิกเกทที่เขียนด้วยมือ โดยใช้หลักการแบ่งแยกลายเส้น", การประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 16, หน้า 297-301, พฤศจิกายน 2536

[2] เกษตร์ ศิริสันติสัมฤทธิ์, อโณทัย ปฐมมาณิศ และ สมคิด อุนการุณวงษ์. "การจดจำโลจิกเกทที่เขียนด้วยมือ โดยใช้หลักการแบ่งแยกลายเส้น." การประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 17. พฤศจิกายน 2537. หน้า 309-313.

[3] เดชา พัฒนประสิทธิ์ชัย, สมคิด อุนการุณวงษ์, ธีรชาติ แสนชัย และ เกษตร์ ศิริสันติสัมฤทธิ์. "การวิเคราะห์โครงสร้างของวงปิดสำหรับการจดจำวงจรโลจิกเกทที่เขียนด้วยมือ." การประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 23. พฤศจิกายน 2543. หน้า 473-476.

[4] สมคิด อุนการุณวงษ์ และเกษตร์ ศิริสันติสัมฤทธิ์. "กระบวนการหลังของการรู้จำวงจรโลจิกเกทที่เขียนด้วยมือ." การประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 24. พฤศจิกายน 2544.

[5] Rafael C. Gonzalez and Richard E. Woods. **Digital Image Processing**. Addison-Wesley Publishing Company. 1992

[6] Wayne Niblack. **An Untroduction to Digital Image Processing**. Prentice Hall International. 1986

[7] Richard S. Sandige. **Modern Digital Design**. McGraw-Hill Publishing Company. 1990

[8] David F. Rogers and J. Alan Adams. **Mathematical Elements for Computer Graphics**. McGraw-Hill Publishing Company. 1990

[9] Herbert Freeman and Larry s. Davis. "A Corner-Finding Algorithm For Chain-Coded Curves." IEEE Transaction On Computers, March 1977. pp. 297-303.

[10] Jia-Guu Leu and Hok-Lai Yau. "Detecting The Dislocations In Metal Crystals From Microscopic Images." Pattern Recognition, vol. 24, no. 1, 1991. pp. 41-56.

[11] Nirwan Ansari and Edward J.Depl. "On Detecting Dominant Points." Pattern Recognition, vol. 24, no. 5, 1991. pp. 441-451.

[12] S.C.Pei and C.N.Lin. "The Detection of Dominant Points on Digital Curves by Scale-Space Filtering." Pattern Recognition, vol. 25, 1992. pp. 130

[13] พารา ลิมมะณีประเสริฐ และ เกษตร์ ศิริสันติสัมฤทธิ์. "การตรวจหาจุดเด่นบนเส้นโค้งของขอบภาพโดยการเปลี่ยนค่าสเกล-สเปซของตัวกรอง." การประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 17. ธันวาคม 2537. หน้า 289-295.

[14] Zhang T. Y. and Suen C. Y. "A Fast Parallel Algorithm for Thinning Digital Pattren." Comm. ACM, vol. 27, no. 3. 1984. pp. 236-239.

[15] ชม กิ้มปาน. "**การรู้จำรูปแบบอักษรพิมพ์ภาษาไทยของคอมพิวเตอร์**." วิทยานิพนธ์วิศวกรรมศาสตรมหาบัณฑิต สาขาวิศวกรรมไฟฟ้า บัณฑิตวิทยาลัย, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง. 2529

[16] M. L. James, G. M. Smith and J. C. Wolford. **Applied Numerical Methods for Digital Computation**. Harper Collins College Publishers. 1993

[17] ปราโมทย์ เดชะอำไพ. **ระเบียบวิธีเชิงตัวเลขในงานวิศวกรรม**. สำนักพิมพ์แห่งจุฬาลงกรณ์มหาวิทยาลัย

[18] เอกรัตน์ จุลวรรณ. "การประเมินผลของขั้นตอนวิธีการทำลายเส้นให้บางเพื่อนำไปใช้กับภาพพิมพ์ลายนิ้วมือ." วิทยานิพนธ์วิศวกรรมศาสตรมหาบัณฑิต สาขาวิศวกรรมไฟฟ้า บัณฑิตวิทยาลัย, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง. 2541

ภาคผนวก

ผลงานวิจัยที่ได้รับการตีพิมพ์

การประชุมวิชาการทาง วิศวกรรมไฟฟ้า

ครั้งที่ ๑๗ ๑๐ - ๑๒ ธันวาคม ๒๕๓๗

ณ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ

การจดจำวงจรโลจิกเกทที่เขียนด้วยมือ โดยใช้หลักการแบ่งแยกลายเส้น

Hand Writing Logic Circuit Recognition Based on Line Segmentation

อโณทัย ปฐมมาณิศ* สมคิด อุนการุณวงษ์* เกษตร์ ศิริสันติสัมฤทธิ์**

* นักศึกษาทดลองวิจัย คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

** อาจารย์ประจำภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

บทคัดย่อ บทความนี้ เป็นการนำเสนอวิธีการที่จะให้ไมโครคอมพิวเตอร์สามารถจดจำภาพวงจรโลจิกเกท ที่เขียนด้วยมือ โดยใช้หลักการแบ่งแยกลายเส้น ข้อมูลภาพจะถูกเก็บเข้าเครื่องไมโครคอมพิวเตอร์ ผ่านทางสแกนเนอร์ ไมโครคอมพิวเตอร์จะประมวลผล โดยการหาขอบของภาพเพื่อแยกตัวเกท และลายวงจรออกเป็นหลาย ๆ ลูป ซึ่งลูปแต่ละลูปจะถูก นำมาวิเคราะห์จำนวนของลายเส้น และลักษณะของลายเส้นที่เป็นองค์ประกอบของลูป แล้วนำมาเปรียบเทียบกับเกทมาตรฐาน ถ้าลูปนั้นเป็นลูปของตัวเกท จะจดจำชนิดของเกทนั้นไว้จนครบทุกลูป จากนั้นก็จะวิเคราะห์การต่อวงจรของเกทแต่ละตัว สุดท้าย คือการนำข้อมูลที่ได้จากการวิเคราะห์มาสร้างวงจร โลจิกเกท ขึ้นมาใหม่ตามรูปแบบมาตรฐาน ทั้งนี้เพื่อให้ผู้ออกแบบวงจรโลจิกเกทสามารถออกแบบได้อย่างรวดเร็ว

Abstract. The article presents the method of hand writing logic gate circuit recognition by using microcomputer to recognize hand writing. This method, is called line-segment method. With this method, firstly the image will be converted to binary data by scanner; Then microcomputer will separate the logic gate from line by considering from out-line. Number and Character of line will categorize the type of gate; after that microcomputer will analyze circuit connecting between each gate. Finally, new standardized circuit will be rebuilt. This process is easy and quickly for designer to design logic circuit.

1 บทนำ

การจดจำวงจรโลจิกเกทที่เขียนด้วยมือนี้ เป็นการนำวิธีการประมวลผลภาพ (Image Processing) มาประยุกต์ใช้ร่วมกับไมโครคอมพิวเตอร์ เพื่อให้ผู้ออกแบบวงจรโลจิกเกทสามารถออกแบบวงจรได้รวดเร็ว เพียงแต่เขียนวงจรหยาบ ๆ (Draft Writing) และสแกนเข้าไมโครคอมพิวเตอร์ หลังจากนั้นไมโครคอมพิวเตอร์จะแปลงวงจรนี้ ใหม่ให้อยู่ในรูปแบบมาตรฐาน

บทความนี้ เป็นการพัฒนา การจดจำวงจรโลจิกเกทที่เขียนด้วยมือ ที่ใช้หลักการแบ่งแยกลายเส้น[1] เพื่อให้สามารถจดจำ ชนิดของเกท ขนาดของเกท และทิศทางของเกท ได้มากขึ้น และปรับปรุงวิธีการแยกลายเส้น จากการติดตามลายเส้นของลูป (4-connectivity) มาเป็นการหามุม (corner) สำหรับวงจรโลจิกเกทที่สร้างขึ้นใหม่จะมีขนาดและการต่อวงจรเหมือนเดิม แต่ทิศทางของเกทจะถูกปรับตามทิศทางมาตรฐาน

2 โครงสร้างเกทมาตรฐาน และการหาขอบของภาพ

ในการวิเคราะห์เพื่อแยกชนิดของเกทนั้น จะพิจารณาโครงสร้างพื้นฐานของเกทมาตรฐาน เนื่องจากตัวเกทแต่ละชนิดประกอบด้วยเส้นตรงและเส้นโค้งในจำนวนไม่เท่ากัน ซึ่งสามารถวิเคราะห์ และแยกเกท แต่ละชนิดได้ดังตารางที่ 1

โครงสร้างของเกทมาตรฐานที่ได้นี้ จะช่วยให้การวิเคราะห์แยกชนิดของเกทเพื่อจดจำได้ง่ายขึ้น แต่ในการตัดสินใจว่าเป็นเกทหรือไม่ และชนิดใดนั้นต้องอาศัยองค์ประกอบอื่นๆ ช่วยเช่น อินพุทและเอาท์พุทของเกท

ตารางที่ 1 โครงสร้างของเกทมาตรฐาน

ลักษณะของเส้น / ชนิดเกท	เส้นตรง (เส้น)	เส้นโค้ง (เส้น)	วงกลมหน้า (วง)	เส้นหลัง (เส้น)
AND	3	1	ไม่มี	ไม่มี
NAND	3	1	1	ไม่มี
OR	ไม่มี	3	ไม่มี	ไม่มี
NOR	ไม่มี	3	1	ไม่มี
XOR	ไม่มี	3	ไม่มี	1
XNOR	ไม่มี	3	1	1
BUFFER	3	ไม่มี	ไม่มี	ไม่มี
NOT	3	ไม่มี	1	ไม่มี

ภาพวงจรที่ใช้ในการวิเคราะห์นี้ มีขนาด 256x256 จุดภาพ (pixel) 2 ระดับ (Binary Image) ก่อนการวิเคราะห์ลายเส้นในขั้นตอนต่าง ๆ จะต้องนำภาพผ่านขั้นตอนการหาขอบของภาพ (Edge Detect) เพื่อจุดประสงค์สองประการคือ ประการแรก แยกตัวเกทออกจากวงจร ดังแสดงในรูปที่ 3 ประการที่สองทำให้การวิเคราะห์ลายเส้นต่าง ๆ ไม่ยุ่งยาก เนื่องจากการหาขอบของภาพจะได้ลายเส้นที่มีความหนาเพียง 1 จุดภาพ และไม่มีจุดแยกเกิดขึ้นบนลายเส้น ซึ่งวิธีการหาขอบของภาพในที่นี้จะพิจารณาจุดภาพรอบข้าง (Neighbourhood) ของจุดภาพตรวจสอบ (P) ดังรูปที่ 1 ที่มีคุณสมบัติตามเงื่อนไขดังนี้

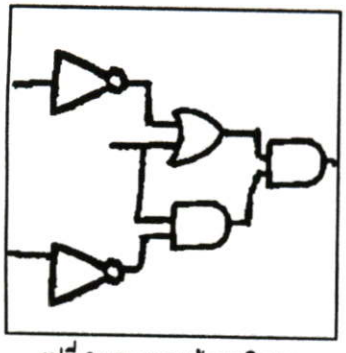
P_7	P_0	P_1
P_6	P	P_2
P_5	P_4	P_3

รูปที่ 1 การพิจารณาจุดรอบข้าง

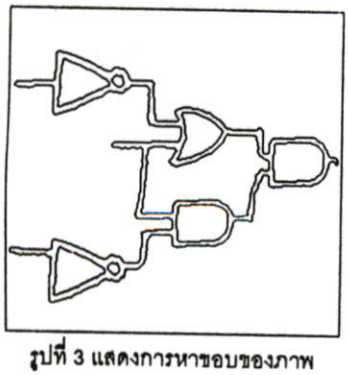
<u>เงื่อนไขการตรวจสอบ</u>

1. $P_0 \& P_2 \& P_4 \& P_6 = 0$
   (เมื่อ & = AND)
2. $P_1+P_3+P_5+P_7 >= 2$
3. $\sum_{i=0}^{7} Pi > 2$

จุดภาพตรวจสอบใด (P) เป็นตามเงื่อนไขทั้งสามข้อข้างต้น จะถูกทำเครื่องหมายไว้ และเมื่อตรวจสอบจนครบทุกจุดภาพก็จะได้ขอบของภาพตามต้องการดังรูปที่ 3 ซึ่งวิธีการนี้สามารถหาขอบของภาพได้เร็วกว่าวิธีการติดตามขอบที่ใช้ Turtle Algorithm[2] ในการวิเคราะห์ขั้นตอนต่อไปจะพิจารณาเฉพาะลายเส้นที่เป็นลูปเท่านั้น



รูปที่ 2 แสดงภาพข้อมูลอินพุท



รูปที่ 3 แสดงการหาขอบของภาพ

3 การแบ่งแยกลายเส้นและการจดจำ

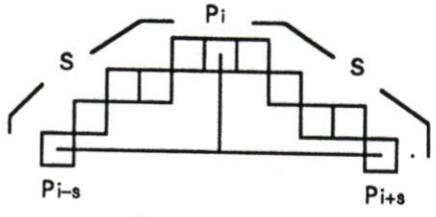
ในงานวิจัยการจดจำโลจิกเกทที่เขียนด้วยมือ โดยใช้หลักการแบ่งแยกลายเส้น[1] สามารถจดจำวงจรโลจิกเกทที่จำกัด ขนาด ทิศทาง และ จำนวนอินพุท ของเกท ดังนั้นการนำไปใช้งานจะมีข้อจำกัดมาก ด้วยเหตุนี้การพัฒนาเพื่อให้สามารถจดจำวงจรโลจิกเกทที่ไม่จำกัดขนาด ทิศทาง และจำนวนอินพุทของเกทได้นั้น จะต้องมีการปรับปรุงวิธีการ และขั้นตอน บางอย่าง เช่น การหาจุดแบ่งแยกลายเส้น การวิเคราะห์ลักษณะลายเส้น และการหาจำนวนอินพุทของเกท ซึ่งรายละเอียดของขั้นตอนต่าง ๆ มีดังนี้

<u>3.1 การหาจุดแบ่งแยกลายเส้น</u>

จากรูปแบบของเกทมาตรฐาน OR Gate, NOR Gate, NOT Gate และ BUFFER สามารถใช้จุดมุมเป็นจุดแบ่งแยกลายเส้นได้ แต่ในกรณี AND Gate และ NAND Gate จะต้องหาจุดแบ่งแยกลายเส้นเพิ่มอีก 2 จุด คือบริเวณ จุดต่อของเส้นตรงกับเส้นโค้งหรือ จุดเริ่มต้นเส้นโค้ง ซึ่งวิธีการหาจุดแบ่งแยกลายเส้นมีดังนี้

<u>3.1.1 การหามุมของลูป (CORNER)</u>

มุมของลูปคือ จุดที่มีการเปลี่ยนแปลงทิศทางของลายเส้นอย่างรวดเร็ว หรือจุดที่มีค่าความโค้ง (K) มาก ๆ ซึ่งค่าความโค้งในที่นี้หาได้จาก ระยะทางจากจุดตรวจสอบ (P_i) ไปยังเส้นตรง $P_{i-s}P_{i+s}$ ในแนวตั้งฉาก หารด้วยความยาวของเส้นตรง $P_{i-s}P_{i+s}$ โดยที่ P_{i-s} และ P_{i+s} มีระยะห่างจากจุดตรวจสอบ (P_i) เท่ากัน (S) ดังแสดงในรูปที่ 3



รูปที่ 4 การหาค่าความโค้ง

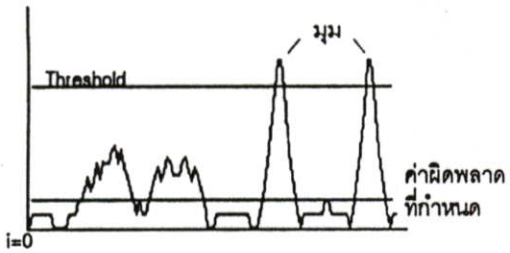
สมการหาค่าความโค้งเป็นดังนี้

$$K_i = \left| \frac{(y_{i+s} - y_{i-s})x_i - (x_{i+s} - x_{i-s})y_i + x_{i+s}y_{i-s} - y_{i+s}x_{i-s}}{(y_{i+s} - y_{i-s})^2 + (x_{i+s} - x_{i-s})^2} \right| \quad \ldots (1)$$

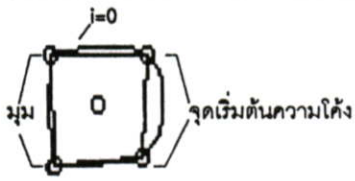
กำหนดให้

i = จุดใดๆ ของลูป (i = 0,1,2,..,n)

n = จำนวนจุดในลูป



รูปที่ 5 กราฟแสดงค่าความโค้งของ AND Gate



รูปที่ 6 แสดงมุม และจุดเริ่มต้นความโค้งของ AND Gate

ดังนั้นมุมจะเป็นจุดที่มีค่าความโค้งมากกว่าค่าที่กำหนด (threshold) และเป็นจุดที่มีค่ามากที่สุด (รูปที่ 6) ถ้าลูปนั้นสามารถหามุมได้ 3 มุม แสดงว่าอาจจะเป็นเกทชนิด OR Gate, NOR Gate ,XOR Gate, XNOR Gate, BUFFER หรือ NOT Gate ซึ่งสามารถนำมุมที่ได้นี้ไปแบ่งแยกลายเส้นเพื่อวิเคราะห์ในขั้นตอนต่อไปได้ แต่ถ้าลูปนั้นหามุมได้ 2 มุม (รูปที่ 6) แสดงว่าอาจจะเป็นเกทชนิด AND Gate หรือ NAND Gate ซึ่งต้องหาจุดแบ่งแยกลายเส้นเพิ่มในขั้นตอน 3.1.2 ส่วนกรณีที่หามุมได้มากกว่า 3 มุม หรือน้อยกว่า 2 มุม จะไม่ถือว่าเป็น Gate และไม่นำไปพิจารณา

3.1.2 การหาจุดต่อระหว่างเส้นตรงกับเส้นโค้งของลูป

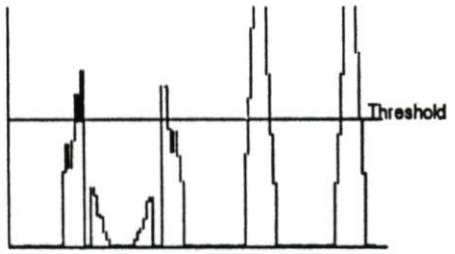
จากค่าความโค้งที่ได้ในขั้นตอนการหามุมของลูป ดังรูปที่ 5 จุดที่เป็นส่วนหนึ่งของเส้นตรง จะมีค่าความโค้งเป็น 0 หรือ น้อยมาก (น้อยกว่าค่าผิดพลาดที่กำหนด) แต่จุดที่เป็นส่วนของเส้นโค้งจะมีค่าความโค้งมากกว่าเส้นตรง ดังนั้นเราสามารถนำจำนวนจุดที่มีค่าความโค้งน้อยกว่าค่าผิดพลาด (t_1,t_2) มาพิจารณาร่วมกับค่าความโค้งที่จุดตรวจสอบได้[3] โดย t_1 จะเริ่มนับจากจุด P_{i-s} ไปในทิศทางลดลง และ t_2 จะเริ่มนับจากจุด P_{i+s} ไปในทิศทางเพิ่มขึ้น จนกว่าจะพบจุดที่มีค่าความโค้งมากกว่า หรือเท่ากับค่าผิดพลาดที่กำหนด ก็จะหยุดนับ และนำมาพิจารณาดังสมการ

$$Knew_i = \sqrt{t_1^2 + t_2^2} \times Kold_i \quad \ldots (2)$$

เมื่อ

n = จำนวนจุดในลูป

i = จุดใดๆ ของลูป

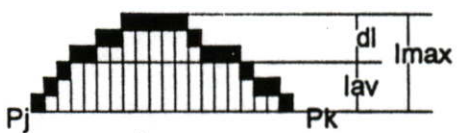


รูปที่ 7 กราฟแสดงค่าความโค้งที่นำค่า t_1 และ t_2 มาพิจารณา

จากวิธีการนี้จะทำให้บริเวณจุดต่อระหว่างเส้นตรงกับเส้นโค้งมีค่าความโค้งมากขึ้น (ดังรูปที่ 7) และสามารถนำจุดที่ได้นี้ไปแบ่งแยกลายเส้น เพื่อใช้ในขั้นตอนต่อไป

3.2 การวิเคราะห์ลักษณะลายเส้น

การวิเคราะห์ลักษณะลายเส้นว่าเป็นเส้นตรง หรือเส้นโค้ง ในที่นี้พิจารณาระยะทางเฉลี่ย หรือพื้นที่เฉลี่ย (l_{av}) ภายใต้ลายเส้นกับเส้นตรง P_jP_k ดังรูปที่ 8 และผลต่าง (dl) ของระยะทางสูงสุด (l_{max}) กับระยะทางเฉลี่ย (l_{av}) ซึ่งระยะทางสามารถหาได้ดังนี้



รูปที่ 8 แสดงการหาระยะทาง

$$l_{av} = \frac{\sum_{i=j}^{k} l_i}{\sqrt{(y_k - y_j)^2 + (x_k - x_j)^2}} \quad \ldots (3)$$

$$l_{max} = \max\left\{\left|l_i\right|\right\} \quad \ldots (4)$$

$$l_i = \frac{(y_k - y_j)x_i - (x_k - x_j)y_i + x_k y_j - y_k x_j}{\sqrt{(y_k - y_j)^2 + (x_k - x_j)^2}} \quad \ldots (5)^{[4]}$$

$$dl = l_{max} - l_{av} \quad \ldots (6)$$

เงื่อนไขการวิเคราะห์ลักษณะลายเส้นมีดังนี้

1. (l_{av}<1) || ((l_{av}<1.5) & (dl<1)) เป็นเส้นตรง
2. (l_{av}>2) || (dl>2) เป็นเส้นโค้ง

(เมื่อ & = AND และ || = OR)

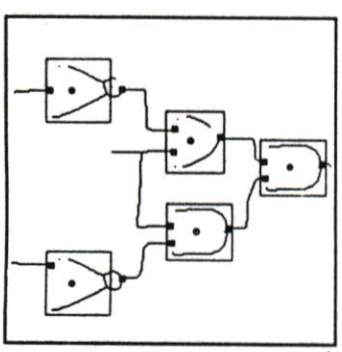
ในกรณีที่ไม่เป็นไปตามเงื่อนไขทั้ง 2 ข้อจะถือว่าลายเส้นนั้นเป็นได้ทั้งเส้นตรง และเส้นโค้ง เมื่อผ่านขั้นตอนนี้ลูปจะถูกวิเคราะห์เป็นเกทได้ 3 ชนิด คือ OR Gate, AND Gate และ BUFFER

3.3 การหาอินพุท และเอาท์พุท

เนื่องจากการวิเคราะห์ในขั้นตอนที่ผ่านมา จะวิเคราะห์กับลายเส้นที่เป็นขอบของภาพ แต่ในการวิเคราะห์หาอินพุท และเอาท์พุท จะวิเคราะห์กับลายเส้นที่เป็นโครงร่างของภาพ ดังนั้นจะต้องนำภาพข้อมูลอินพุทไปผ่านขั้นตอน การหาโครงร่าง (Thinning) ก่อน

3.3.1 การหาเอาท์พุท

ในการหาเอาท์พุทของเกทจะต้องกำหนดจุดอ้างอิงที่ใกล้กับจุดเอาท์พุท (หรือบริเวณที่ควรจะมีจุดแยกเอาท์พุท) ซึ่งการกำหนดแบ่งออกเป็น 2 กรณี คือ กรณีของ OR Gate หรือ BUFFER จะใช้จุดมุมเป็นจุดอ้างอิง ส่วนกรณี AND Gate จะใช้จุดกึ่งกลางลายเส้นที่เป็นเส้นโค้งเป็นจุดอ้างอิง แต่ก่อนที่จะทำการหาจุดแยกเอาท์พุทนั้น จะต้องตรวจสอบรอบ ๆ จุดอ้างอิงว่ามีลูปเล็ก ๆ (วงกลม NOT ที่มีขนาดน้อยกว่า 1/3 เท่าของเกท) อยู่ใกล้หรือไม่ ถ้ามีจะหาตำแหน่งศูนย์กลางของลูปเล็กนั้น เพื่อนำไปคำนวณทิศทางของเกท และกำหนดเป็นจุดอ้างอิงใหม่ หลังจากนั้นจะสแกนหาลายเส้นที่เป็นโครงร่างของเกท เพื่อหาจุดแยกบริเวณจุดอ้างอิงดังกล่าว



รูปที่ 9 แสดงตำแหน่งอินพุท และเอาท์พุทที่หาได้

3.3.2 การหาอินพุท

เมื่อรู้ตำแหน่งของเอาท์พุท ก็สามารถหาลายเส้นที่เป็นเส้นหลังของตัวเกท และสแกนตามลายหาจุดแยกอินพุทได้ แต่ถ้าเป็นกรณีของ XOR Gate และ XNOR Gate จะไม่พบจุดแยกอินพุทที่เส้นหลังของเกท ดังนั้นจะต้องหาเส้นหลังใหม่ โดยการสแกนออกจากเส้นหลังของตัวเกทในทิศทางตั้งฉาก เมื่อพบเส้นหลังใหม่ก็จะสแกนตามลายหาจุดแยกอินพุทเหมือนครั้งแรก ซึ่งก็จะได้ตำแหน่ง และจำนวน อินพุทของเกท เมื่อถึงขั้นตอนนี้จากข้อมูลที่ทำการวิเคราะห์มาทั้งหมดจะสามารถจดจำชนิดของเกทได้

3.4 การหาทิศทางของเกท

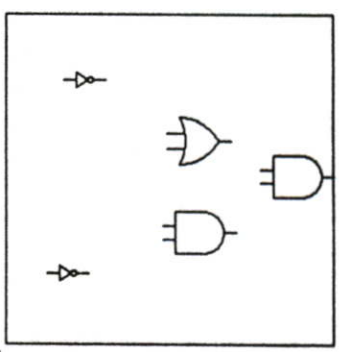
การคำนวณหาทิศทางของเกทในที่นี้อาศัยจุด 2 จุด คือจุดศูนย์กลางของตัวเกท และจุดศูนย์กลางของวงกลมหน้าเกท (กรณีมีวงกลม NOT Gate) หรือตำแหน่งเอาท์พุท (กรณีไม่ NOT Gate)

3.5 การวิเคราะห์การต่อวงจร

หลังจากการหาอินพุทและเอาท์พุทของเกท ลายเส้นที่เป็นตัวเกท จะถูกตัดขาดจากอินพุท และเอาท์พุท (ดังรูปที่ 9) ดังนั้นการวิเคราะห์การต่อลายวงจรจะไม่ยุ่งยาก เพียงแต่สแกนตามลายเส้นจากอินพุท และเอาท์พุทของเกทที่ละตัว ในกรณีที่เกทต่อกับจุดแยกตำแหน่งของจุดแยกจะถูกจดจำไว้

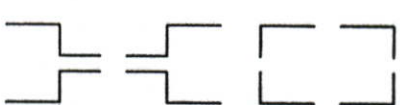
4 การปรับทิศทางและการวาดภาพ

เมื่อสามารถจดจำชนิด ตำแหน่ง และทิศทางของเกทจนครบทั้งภาพแล้ว ก่อนการวาดภาพเกทมาตรฐานลงไปแทนที่เกทตัวเก่าจะต้องปรับตำแหน่งของเกททุกตัว โดยจะพิจารณาทิศทางของเกทแต่ละตัว และแยกเกทออกเป็นกลุ่ม ๆ ซึ่งแต่ละกลุ่มจะมีทิศทางของเกทต่างกันมากกว่า 45 องศา เนื่องจากรูปแบบของเกทมาตรฐานจะมีทิศทางตามแนวแกน x และแกน y เท่านั้น ต่อจากนั้นจะหาทิศทางเฉลี่ยที่เบี่ยงเบนจากแกน x หรือแกน y เพื่อหมุนหรือปรับตำแหน่งเกททุกตัวรวมทั้งจุดแยกให้เหมาะสม หลังจากนั้นจึงวาดเกทมาตรฐานลงไป ดังรูปที่ 10



รูปที่ 10 แสดงการวางตำแหน่งของเกทมาตรฐาน

เมื่อวาดเกทมาตรฐานครบทุกตัว ขั้นตอนต่อไปคือ การลากเส้นลายวงจร ซึ่งรูปแบบการลากเส้นจะเป็นเส้นตรงที่ลากในแนวแกน x และแกน y เท่านั้น ดังรูป 11



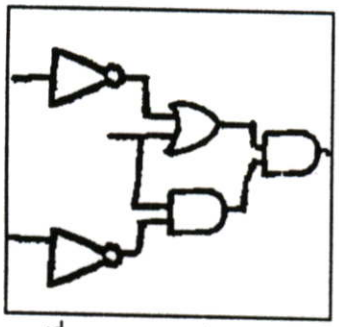
รูปที่ 11 ลักษณะการลากเส้นต่อลายวงจร

ในกรณีที่อินพุทหรือเอาท์พุทของเกทต่อกับจุดแยก จุดแยกจะถูกปรับให้ตรงกับขาของเกทนั้น โดยมีขอบเขตในการปรับขึ้นอยู่กับความหนาของลายเส้น ของภาพข้อมูลอินพุท

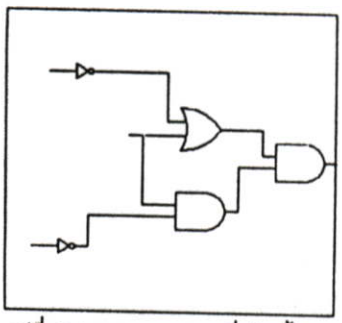
5 ผลการทดลอง

ในการทดลอง โปรแกรมเขียนขึ้นด้วยภาษา C และทดลองบนเครื่องไมโครคอมพิวเตอร์ IBM PC/AT โดยเก็บข้อมูลภาพขนาด

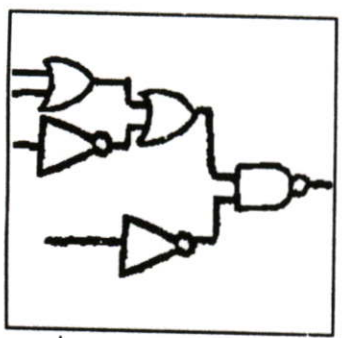
256x256 จุดภาพ ด้วยเครื่องสแกนเนอร์ จำนวน 36 ภาพ เพื่อทดสอบวิธีการข้างต้น ซึ่งผลการทดลองสามารถจดจำ และสร้างวงจรโลจิกเกทขึ้นใหม่ได้ถูกต้อง 94.4% และผิดพลาดเนื่องจากเกทมีขนาดเล็ก 6.6 % โดยในที่นี้ได้นำมาแสดงให้ดู 3 ภาพ ดังรูปที่ 12 ถึง รูปที่ 14



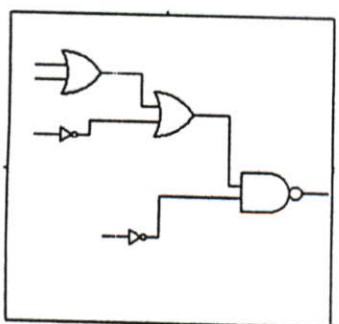
รูปที่ 12 a แสดงภาพข้อมูลอินพุท



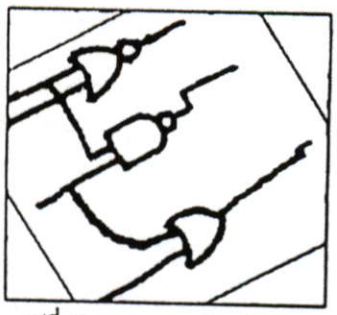
รูปที่ 12 b แสดงภาพวงจรที่สร้างขึ้นใหม่



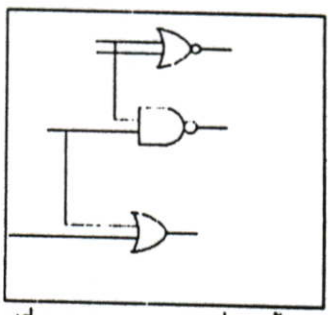
รูปที่ 13 a แสดงภาพข้อมูลอินพุท



รูปที่ 13 b แสดงภาพวงจรที่สร้างขึ้นใหม่



รูปที่ 14 a แสดงภาพข้อมูลอินพุท



รูปที่ 14 b แสดงภาพวงจรที่สร้างขึ้นใหม่

6 สรุปผล

งานวิจัยนี้เป็นการพัฒนาการจดจำวงจรโลจิกเกทที่เขียนด้วยมือ โดยใช้หลักการแบ่งแยกลายเส้น เพื่อเพิ่มความสามารถในการจดจำรูปร่างของเกท ขนาดของเกท และทิศทางการวางตัวเกท ในส่วนของการทดลองได้ทำการทดสอบกับภาพวงจรจำนวน 36 ภาพ ซึ่งสามารถจดจำ และสร้างภาพวงจรใหม่ได้ 94.4% อย่างไรก็ตามงานวิจัยนี้ยังต้องพัฒนาให้สามารถจดจำ Equivalent logic gate, Flip-Flop และอุปกรณ์ที่ใช้ร่วมในวงจรโลจิกเกท เช่น ตัวต้านทาน และตัวเก็บประจุ

7 หนังสืออ้างอิง

[1] อรุณ กำเหนิดนนท์ และ เกษตร์ ศิริสันติสัมฤทธิ์, การจดจำโลจิกเกทที่เขียนด้วยมือ โดยใช้หลักการแบ่งแยกลายเส้น, บทความในการประชุมทางวิชาการ วิศวกรรมไฟฟ้า ครั้งที่ 16

[2] Rafael C. Gonzalez, Digital Image Processing Second Edition, Addison-Wesley,1987

[3] Herbert Freeman and Larry S.Davis, A Corner-Finding Algorithm for Chain-Coded Curves, IEEE Transaction on computers, pp. 297-303, March 1977

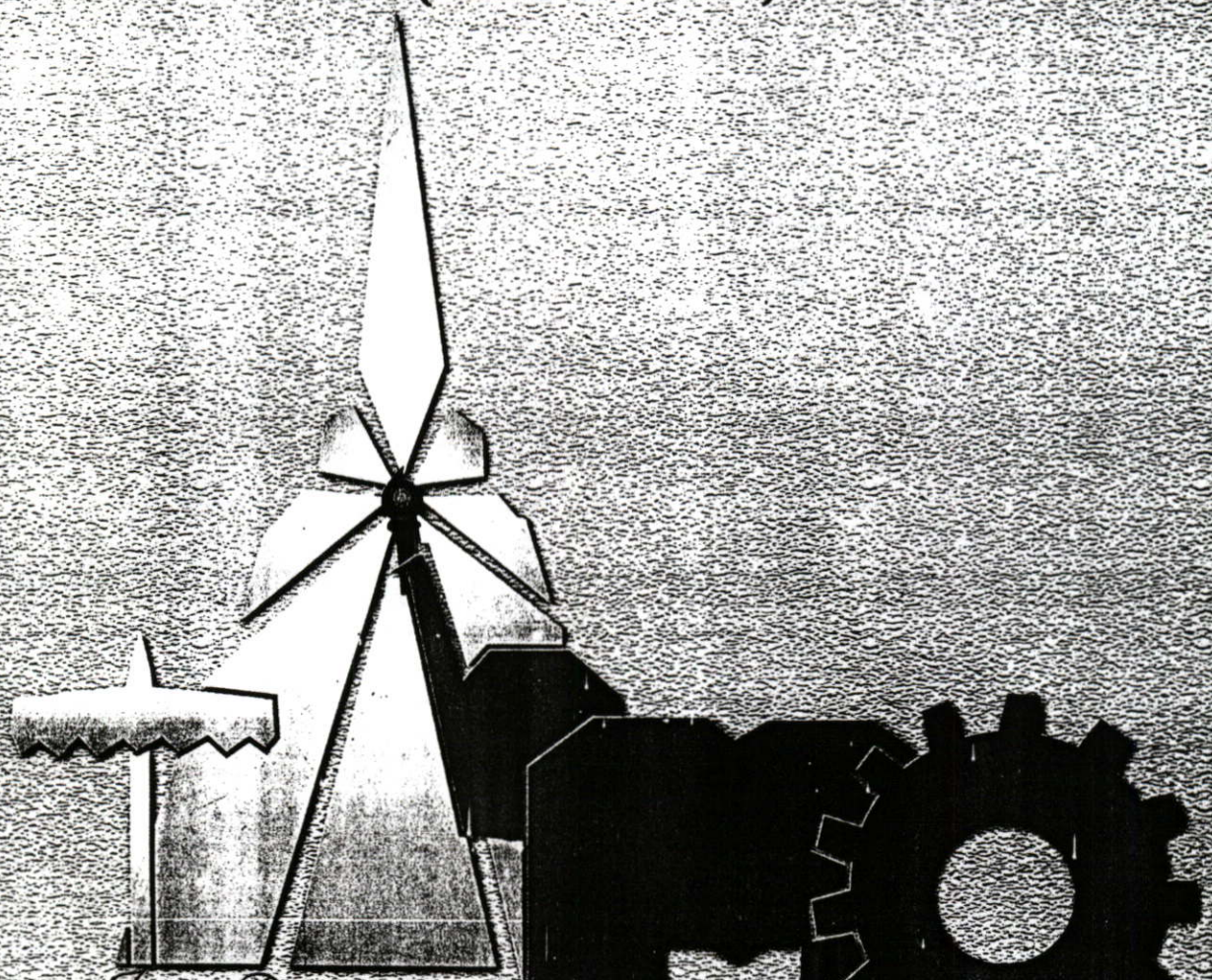
[4] Jia-Guu Leu and Hok-Lai Yau, Detecting The Dislocations In Metal Crystals From Microscopic Images, Patttern Recognition, Vol. 24, No. 1, pp. 41-55, 1991

[5] วีนัส พีชวณิชย, แคลคูลัส ฉบับเตรียมสอบ

การประชุมวิชาการ ทางวิศวกรรมไฟฟ้า ครั้งที่ 23

EECON XXIII

23rd Electrical Engineering Conference (EECON-23)



Jan - Dec 2000
30th Anniversary
Faculty of Engineering, Chiang Mai University

วันที่ 23-24 พฤศจิกายน 2543 โรงแรมดิเอ็มเพรส เชียงใหม่

ดำเนินการจัดประชุมโดย

ภาควิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์

มหาวิทยาลัยเชียงใหม่

การวิเคราะห์โครงสร้างของลูปสำหรับการจดจำวงจรโลจิกเกทที่เขียนด้วยมือ

Loop Structure Analysis for Hand Written Logic Circuit Recognition

เศชา พัฒนประสิทธิ์ชัย สมคิด อุนการุณวงษ์ ธีรชาติ แสนชัย เกษตร์ ศิริสันติสัมฤทธิ์
ภาควิชาวิศวกรรมการวัดคุม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ถนนฉลองกรุง เขตลาดกระบัง กรุงเทพฯ 10520
โทร 3267346-7 ต่อ 104 โทรสาร 3267346-7 ต่อ 103 E-mail: kskaset@kmitl.ac.th

บทคัดย่อ

บทความนี้เป็นการนำเสนอวิธีการวิเคราะห์โครงสร้างของลูปสำหรับการจดจำวงจรโลจิกเกทที่เขียนด้วยมือ ภาพลายเส้นขอบของโลจิกเกทซึ่งมีลักษณะเป็นลูปจะถูกกรองสัญญาณรบกวนและหาจุดมุมภายในลูปโดยตัวกรองเกาซ์เซียนเพื่อแบ่งแยกลูปออกเป็นลายเส้น ต่อจากนั้น ลายเส้นแต่ละเส้นจะถูกวิเคราะห์หาลักษณะของลายเส้นโดยการทำโค้งกระชับ เพื่อจำแนกลายเส้นออกเป็นเส้นตรงและเส้นโค้ง จากวิธีการดังกล่าวนี้ทำให้เราทราบโครงสร้างของลูป และสามารถจดจำชนิดของโลจิกเกทได้ถูกต้องมากยิ่งขึ้น

คำสำคัญ : การแบ่งส่วน, ตัวกรองเกาซ์เซียน, การทำโค้งกระชับ

Abstract

This paper presents the segmentation and analysis of closed loop line structure, which are necessary part of Hand Written Logic Circuit Recognition research. Initially the corner points on the contour of the logic image are calculated by using Gaussian filter, in order to segment the loop into lines. Then, the characteristic of all the lines are classified as straight and curved by using curve fitting with method of least square. These methods can be analyze the structure of loop and recognize the type of logic gate more accurately.

Keywords : segmentation, Gaussian filter, curve fitting

1. บทนำ

ในการศึกษาวิจัยเรื่องการจดจำวงจรโลจิกเกทที่เขียนด้วยมือโดยใช้หลักการแบ่งแยกลายเส้น [1] วงจรโลจิกเกทที่เขียนด้วยมือที่ต้องการให้จดจำ จะถูกสแกนเข้าเครื่องคอมพิวเตอร์ และแปลงข้อมูลภาพให้เป็นภาพสองระดับ จากนั้นทำการหาขอบภาพเพื่อแยกตัวโลจิกเกท และลายวงจรออกเป็นลูปๆ นำลูปที่ได้แต่ละลูปมาคำนวณหาจุดมุมโดยพิจารณาจากค่าความโค้ง เพื่อแบ่งแยกลูปออกเป็นลายเส้น ต่อจากนั้นนำลายเส้นของแต่ละเส้นในลูปมาวิเคราะห์หาโครงสร้างของลูปว่าประกอบด้วยเส้นตรงหรือเส้นโค้ง เมื่อเราทราบโครงสร้างของลูปแต่ละลูปและนำไปเปรียบเทียบกับโครงสร้างของเกทมาตรฐาน จะทำให้เราสามารถจดจำได้ว่าลูปนั้นเป็นโลจิกเกทชนิดใด สุดท้ายจะทำการวิเคราะห์ต่อลายวงจรของโลจิกเกทแต่ละตัว และสร้างวงจรโลจิกเกทขึ้นมาใหม่ในรูปแบบมาตรฐาน

อย่างไรก็ตาม การแบ่งแยกลายเส้นด้วยวิธีการดังกล่าวข้างต้นอาจทำให้เกิดความผิดพลาดในการหาโครงสร้างของลูปว่าเป็นเส้นตรงหรือเส้นโค้ง สืบเนื่องจากการหาจุดมุมของลูปไม่ถูกต้อง นอกจากนี้ยังยุ่งยากในการกำหนดค่า Threshold ความโค้ง และเมื่อลูปของเกทมีขนาดเล็กจะทำให้เกิดความผิดพลาดได้สูง ด้วยเหตุนี้ บทความนี้จะนำเสนอวิธีการวิเคราะห์โครงสร้างของลูปโดยตัวกรองเกาซ์เซียนและการทำโค้งกระชับด้วยวิธีกำลังสองน้อยที่สุด

2. ทฤษฎี

2.1 การหาจุดมุมด้วยตัวกรองเกาซ์เซียน

โดยทั่วไปเส้นขอบของวัตถุในภาพจะมีสัญญาณรบกวนทำให้มีการโค้งเว้าที่ผิดไป (False local concavities and convexities) อยู่เป็นจำนวนมาก ซึ่งความโค้งเว้าที่ผิดไปเหล่านี้จะเป็นสาเหตุทำให้การหาจุดมุมผิดพลาดได้เมื่อใช้วิธีการหาค่าความโค้ง ดังนั้นเพื่อลดจุดผิดพลาดดังกล่าวการกรองสัญญาณรบกวนหรือการทำให้ลายเส้นราบเรียบจึงเป็นสิ่งจำเป็น วิธีการทำลายเส้นให้ราบเรียบที่มีประสิทธิภาพสูงวิธีหนึ่งคือการประสาน (Convolution) ลายเส้นด้วยตัวกรองเกาซ์เซียน (Gaussian filter) [2] โดยสมการของตัวกรองเกาซ์เซียนถูกนิยามไว้ดังนี้

$$g(t,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-0.5(t/\sigma)^2} \qquad (1)$$

เมื่อ σ คือ ค่าสเกล-สเปซของตัวกรองเกาซ์เซียน
t คือ ตัวแปรอิสระ

กลุ่มของจุดภาพที่เป็นลายเส้นขอบสามารถแสดงในรูปแบบสมการพาราเมติกได้ดังนี้

$$(x(t), y(t)) \in R^2 \qquad (2)$$

เมื่อ $x(t)$ คือตำแหน่งพิกัดทางแกน x ที่ t ใดๆ

C16

y(t) คือตำแหน่งพิกัดทางแกน y ที่ t ใดๆ

เมื่อนำตัวกรองเกาซ์เซียนมาประสานกับลายเส้นจะได้สมการ

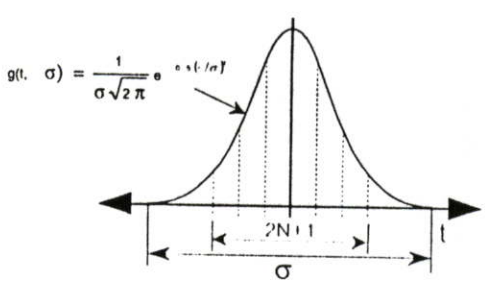
$$X(t,\sigma) = x(t) * g(t,\sigma)$$
$$Y(t,\sigma) = y(t) * g(t,\sigma) \qquad (3)$$

เมื่อ * คือ การประสาน (Convolution)

เนื่องจากตัวกรองเกาซ์เซียนที่แสดงในสมการที่ (1) เป็นฟังก์ชันต่อเนื่อง แต่ข้อมูลที่ใช้ร่วมกับตัวกรองเกาซ์เซียนเป็นลายเส้นซึ่งเป็นข้อมูลที่ไม่ต่อเนื่อง (Discrete) ดังนั้นตัวกรองเกาซ์เซียนจึงต้องถูกแซมพลิง (Sampling) ให้เป็นฟังก์ชันที่ไม่ต่อเนื่องด้วยดังรูปที่ 1 โดยทั่วไปแล้วการเก็บค่าสัมประสิทธิ์ของตัวกรองเกาซ์เซียนจะถูกกำหนดด้วยจำนวนแซมพลิง 2N+1 โดยที่ N สามารถคำนวณได้จากสมการ

$$N = min\{N_1, N_2\}$$
$$N_1 = \left\{ n : Max \sum_{t=-n}^{n} \frac{1}{\sigma\sqrt{2\pi}} e^{-0.5\left(\frac{t}{\sigma}\right)^2} \le 0.99 \right\}$$
$$N_2 = \lfloor 5\sigma \rfloor \qquad (4)$$

เมื่อ $\lfloor b \rfloor$ คือจำนวนเต็มที่มากที่สุด แต่น้อยกว่าหรือเท่ากับ b



รูปที่ 1 เส้นกราฟของสมการตัวกรองเกาซ์เซียนและการแซมพลิง

การหาค่าความโค้งของจุดใดๆ บนลายเส้นที่ผ่านการทำให้ราบเรียบแล้วสามารถหาได้จากสมการ

$$K(t,\sigma) = \left| \frac{\dot{X}\ddot{Y} - \dot{Y}\ddot{X}}{\left(\dot{X}^2 + \dot{Y}^2\right)^{3/2}} \right| \qquad (5)$$

เมื่อ

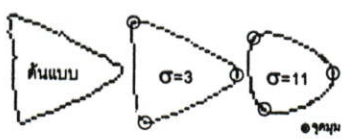
$$\dot{X} = \frac{dX}{dt},\ \ddot{X} = \frac{d^2X}{dt^2},$$
$$\dot{Y} = \frac{dY}{dt},\ \ddot{Y} = \frac{d^2Y}{dt^2} \qquad (6)$$

เนื่องจากลายเส้นขอบมีลักษณะที่ไม่ต่อเนื่อง การหาอนุพันธ์อันดับที่หนึ่งและที่สองของลายเส้นสามารถคำนวณได้จากสมการดังนี้

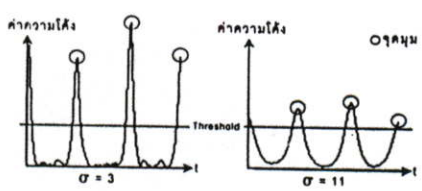
$$\dot{X} = X(t+1) - X(t-1)$$
$$\ddot{X} = X(t+1) + X(t-1) - 2X(t)$$
$$\dot{Y} = Y(t+1) - Y(t-1)$$
$$\ddot{Y} = Y(t+1) + Y(t-1) - 2Y(t) \qquad (7)$$

<u>ลำดับขั้นตอนของการหาจุดมุมมีดังต่อไปนี้</u>

1) ประสานตัวกรองเกาซ์เซียนกับทุกๆ จุดบนลายเส้น
2) คำนวณหาค่าความโค้งของทุกๆ จุดบนลายเส้น
3) หาจุดที่มีค่าความโค้งสูงกว่าค่า Threshold ที่กำหนด และเป็นจุดสูงสุดในช่วงนั้น กำหนดให้เป็นจุดมุม ดังรูปที่ 2 และ 3



รูปที่ 2 การทำลายเส้นให้ราบเรียบโดยกำหนดค่าสเกล-สเปซต่างกัน



รูปที่ 3 กราฟค่าความโค้งที่จุดต่างๆ บนลายเส้น

2.2 การวิเคราะห์ลักษณะลายเส้นโดยใช้โค้งกระชับ

จุดมุมที่ได้ในขั้นตอนที่ผ่านมาเราจะนำมาใช้เป็นจุดแบ่งแยกรูปเป็นลายเส้นเพื่อวิเคราะห์ลักษณะของลายเส้น ลายเส้นดังกล่าวก็คือกลุ่มของจุดภาพที่สามารถหาสมการทางคณิตศาสตร์มาแทนได้ โดยวิธีที่เรียกว่าการทำโค้งกระชับ (Curve fitting) ซึ่งมีวิธีหลายวิธีและหลายรูปแบบ ดังนั้นในการวิเคราะห์ลายเส้นว่าเป็นเส้นตรงหรือไม่ สามารถทำได้โดยการหาสมการเส้นตรงจากลายเส้นและหาค่าผิดพลาดของลายเส้นที่คำนวณได้จากสมการกับลายเส้นจริง โดยค่าผิดพลาดนี้จะเป็นตัวชี้ความเป็นเส้นตรงของลายเส้น ดังนั้นวิธีที่เหมาะสำหรับการวิเคราะห์ดังกล่าวก็คือ วิธีกำลังสองน้อยสุด (Method of least squares)

วิธีกำลังสองน้อยสุด เป็นวิธีการสร้างสมการเส้นตรงที่ดีที่สุดสำหรับกลุ่มของจุดภาพ โดยพิจารณาผลต่างระหว่างค่า y ของจุดภาพกับค่า y ที่คำนวณจากสมการ y = mx+b ที่สร้างขึ้น ซึ่งผลต่างนี้เรียกว่าค่าคงเหลือ (Residual) ถ้าจุดภาพมีตำแหน่ง (x_i,y_i) ค่าของ y_i สามารถหาได้โดยแทน x ด้วย x_i ในสมการ y = mx+b ดังนั้นค่าคงเหลือของจุดภาพตำแหน่ง $(x_1,y_1),(x_2,y_2),\ldots,(x_n,y_n)$ จะได้เป็น

$$y_1-(mx_1+b),\ y_2-(mx_2+b),\ldots,y_n-(mx_n+b) \qquad (8)$$

เนื่องจากค่าคงเหลือที่ได้จะมีทั้งค่าบวก และค่าลบ ดังนั้นผลรวมของค่าคงเหลือแต่ละจุดมีค่าน้อย ไม่สามารถบ่งบอกได้ว่าสมการที่หาได้จะดีที่สุดสำหรับกลุ่มของจุดภาพเหล่านั้น แต่ถ้าทำค่าคงเหลือของแต่ละจุดให้เป็นบวกโดยการยกกำลังสองแล้วหาค่าผลรวม (S_R) สมการที่ดีที่สุดสำหรับกลุ่มของจุดภาพจะต้องมีค่าผลรวม (S_R) ดังกล่าวน้อยที่สุด ซึ่งค่าคงเหลือยกกำลังสองสำหรับจุด (x_i,y_i) หาได้จาก

$$y_i^2 - 2mx_iy_i - 2y_ib + m^2x_i^2 + 2mx_ib + b^2 \qquad (9)$$

ในการหาค่าผลรวมของค่าคงเหลือยกกำลังสองเหล่านี้ เพื่อให้กระชับและดูง่าย กำหนดให้

$$\sum x = x_1 + x_2 + \ldots + x_n,\ \sum y = y_1 + y_2 + \ldots + y_n$$
$$\sum x^2 = x_1^2 + x_2^2 + \ldots + x_n^2,\ \sum y^2 = y_1^2 + y_2^2 + \ldots + y_n^2$$
$$\sum xy = x_1y_1 + x_2y_2 + \ldots + x_ny_n \qquad (10)$$

เมื่อกำหนดให้ S_R เป็นผลรวมของค่าคงเหลือยกกำลังสองจะได้สมการดังนี้

$$S_R = \sum y^2 - 2m\sum xy - 2b\sum y + m^2\sum x^2 + 2mb\sum x + nb^2 \qquad (11)$$

จากสมการที่ (11) ค่า S_R จะประกอบด้วยค่า m และ b ถ้าพิจารณาค่า b เป็นค่าคงที่ และเขียนกราฟในระนาบ mS_R จะได้รูปพาราโบล่าหงายขึ้น เนื่องจากค่า m^2 เป็นบวก และค่า S_R จะน้อยที่สุดที่จุดยอดของพาราโบล่า ซึ่งเราสามารถหาจุดนี้ได้ดังนี้

$$\frac{\partial S_R}{\partial m} = 0$$
$$m\sum x^2 + b\sum x - \sum xy = 0 \qquad (12)$$

ในทำนองเดียวกัน ถ้าเราพิจารณาค่า m เป็นค่าคงที่และเขียนกราฟในระนาบ bS_R เช่นเดียวกันกับวิธีข้างบน เราจะได้ค่าของ S_R น้อยที่สุดเมื่อ

$$\frac{\partial S_R}{\partial b} = 0$$
$$m\sum x - \sum y + nb = 0 \qquad (13)$$

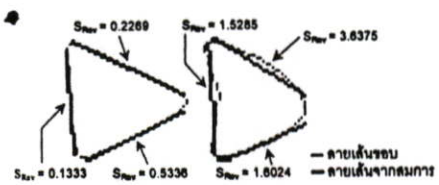
จากสมการที่ (12) และ (13) เราสามารถหาค่าของ m และ b ที่ทำให้ผลรวมของค่าคงเหลือยกกำลังสอง (S_R) น้อยที่สุดได้ดังนี้

$$m = \frac{n\sum xy - \sum x\sum y}{n\sum x^2 - (\sum x)^2}$$
$$b = \frac{\sum x^2\sum y - \sum x\sum xy}{n\sum x^2 - (\sum x)^2} \qquad (14)$$

เมื่อพิจารณาค่าของ S_R จะเห็นว่าจำนวนจุดในลายเส้น (n) มีผลกับค่าของ S_R โดยตรง ดังนั้นการนำค่า S_R มาใช้จำเป็นต้องหาเป็นค่าเฉลี่ยดังนี้

$$S_{Rav} = \frac{S_R}{n} \qquad (15)$$

ค่าของ S_{Rav} ที่ได้นี้จะเป็นตัวบ่งชี้ความเป็นเส้นตรงของลายเส้นได้ดี กล่าวคือ ถ้าลายเส้นเป็นเส้นตรงจะมีค่าของ S_{Rav} เป็น 0 หรือน้อยมาก และถ้าเป็นเส้นโค้งจะมีค่าของ S_{Rav} มาก (รูปที่ 4)



รูปที่ 4 การเปรียบเทียบลายเส้นที่ได้จากสมการกับลายเส้นจริง

ลำดับขั้นตอนการวิเคราะห์ลักษณะของลายเส้น

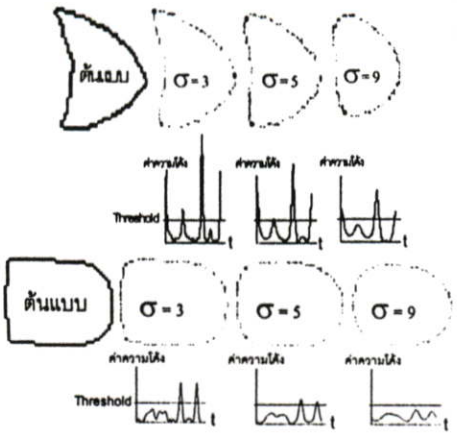
1) นำลายเส้นแต่ละเส้นที่แยกออกมาจากลูปมาหาค่า m และค่า b จากสมการที่ 14
2) หาค่า S_R และ S_{Rav} จากสมการที่ 11 และ 15
3) จำแนกเส้นตรงและเส้นโค้งโดยใช้เงื่อนไขดังนี้
   - ถ้าค่า S_{Rav} น้อยกว่า 1 เป็นเส้นตรง
   - ถ้าค่า S_{Rav} มากกว่า หรือเท่ากับ 1 เป็นเส้นโค้ง

3. ผลการทดลอง

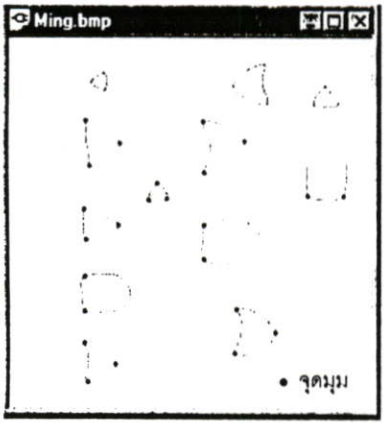
ในการทดลองเรานำวิธีการวิเคราะห์โครงสร้างของลูปนี้ไปใช้ในงานวิจัยเรื่องการจดจำวงจรโลจิกเกทที่เขียนด้วยมือโดยใช้หลักการแบ่งแยกลายเส้น ซึ่งมีผลการทดลองดังต่อไปนี้

ภายหลังจากเสร็จสิ้นขั้นตอนการแยกตัวโลจิกเกทและลายวงจรออกเป็นลูป ๆ แล้ว นำลูปที่ได้มาคำนวณหาจุดมุมด้วยตัวกรองเกาซ์เซียน จากการทดลองเราพบว่าค่าสเกล-สเปซมีผลโดยตรงกับความราบเรียบของลายเส้นและค่าความโค้งของแต่ละจุดบนลายเส้น กล่าวคือค่าสเกล-สเปซน้อย ความราบเรียบของลายเส้นจะน้อย แต่ค่าความโค้งของแต่ละจุดจะสูงขึ้น ในทางตรงกันข้าม ค่าสเกล-สเปซมาก ความราบเรียบของลายเส้นจะมากแต่ค่าความโค้งของแต่ละจุดจะลดลง(ดังรูปที่5) ดังนั้นการกำหนดค่าสเกล-สเปซที่เหมาะสมจะทำให้การหาจุดมุมได้ถูกต้องมากยิ่งขึ้น ซึ่งการทดลองนี้เราใช้ค่าสเกล-สเปซเท่ากับ 3.0 ค่า Threshold สำหรับการหาจุดมุมเท่ากับ 0.035 และได้ผลการทดลองดังรูปที่ 6

C16



รูปที่ 5 ผลของการปรับค่าสเกล-สเปซ



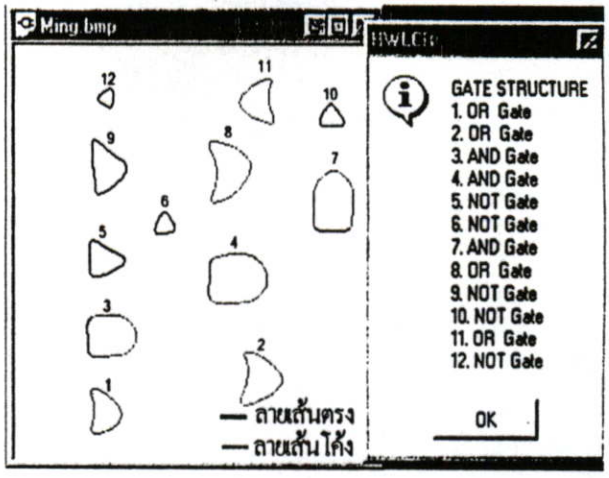
รูปที่ 6 ผลของการหาจุดมุมด้วยตัวกรองเกาซ์เซียน

หลังจากนั้น นำจุดมุมที่ได้มาวิเคราะห์หาลักษณะของลายเส้นโดยใช้โค้งกระชับวิธีกำลังสองน้อยสุด และวิเคราะห์หาโครงสร้างของโลจิกเกท ซึ่งจะได้ผลลัพธ์ดังแสดงในรูปที่ 7

ตารางที่ 1 เป็นผลการทดลองกับโลจิกเกทชนิดต่างๆ ที่มีขนาดต่างกันโดยกำหนดให้ค่าสเกล-สเปซและ Threshold คงที่ ซึ่งให้ความถูกต้องในการวิเคราะห์โครงสร้างเฉลี่ย 92.4% จากโลจิกเกททั้งหมด 642 ตัว

ตารางที่ 1 ผลการทดลอง

โครงสร้าง	ถูกต้อง		ผิดพลาด		รวม
	ตัว	%	ตัว	%	ตัว
AND, NAND	185	97.4	5	2.6	190
OR, NOR, XOR, XNOR	172	90.5	18	9.5	190
NOT, BUFFER	144	90.6	15	9.4	159
SQUARE	92	89.3	11	10.7	103
รวม	593	92.4	49	7.6	642



รูปที่ 7 การจำแนกเส้นตรง-เส้นโค้ง และการวิเคราะห์โครงสร้างของเกท

4. บทสรุป

การหาจุดมุมด้วยตัวกรองเกาซ์เซียน ค่าของสเกล-สเปซ (σ) มีผลโดยตรงกับการทำลายเส้นให้ราบเรียบ (Smoothing) และค่าความโค้งของแต่ละจุดบนลายเส้น ในส่วนของการวิเคราะห์ลักษณะของลายเส้นด้วยโค้งกระชับ โดยหลักการเป็นการสร้างสมการเส้นตรงที่ดีที่สุดสำหรับกลุ่มของจุดภาพที่เป็นเส้นตรง ดังนั้นค่าของ S_{Rav} จึงสามารถเป็นตัวบ่งชี้ความเป็นเส้นตรงและเส้นโค้งของลายเส้นได้ดี และสามารถนำมาใช้วิเคราะห์หาโครงสร้างของวงจรโลจิกเกทที่เขียนด้วยมือโดยใช้หลักการแบ่งแยกลายเส้น ซึ่งผลของการวิเคราะห์โครงสร้างของโลจิกเกทที่เขียนด้วยมือให้ความถูกต้องเฉลี่ย 92.4 % จากโลจิกเกททั้งหมด 642 ตัว

เอกสารอ้างอิง

[1] อโณทัย ปฐมมานิศ, สมคิด อุนการุณวงษ์ และ เกษตร์ ศิริสันติสัมฤทธิ์, " การจดจำวงจรโลจิเกทที่เขียนด้วยมือโดยใช้หลักการแบ่งแยกลายเส้น ", บทความในการประชุมวิชาการทางวิศวกรรมไฟฟ้าครั้งที่ 17

[2] N. Ansari and E.J. Delp, " On Detecting Dominant Points ", Pattern Recognition, vol. 24, No.5, 1991, pp.441-451

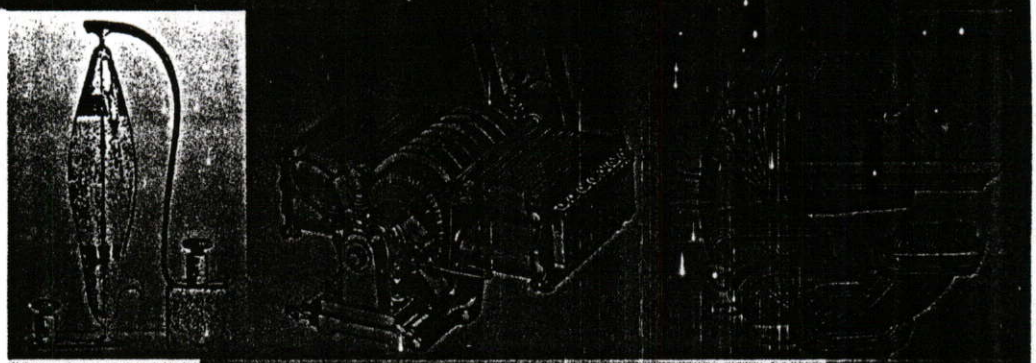
[3] ปราโมทย์ เดชะอำไพ, " ระเบียบวิธีเชิงตัวเลขในงานวิศวกรรม ", สำนักพิมพ์แห่งจุฬาลงกรณ์มหาวิทยาลัย

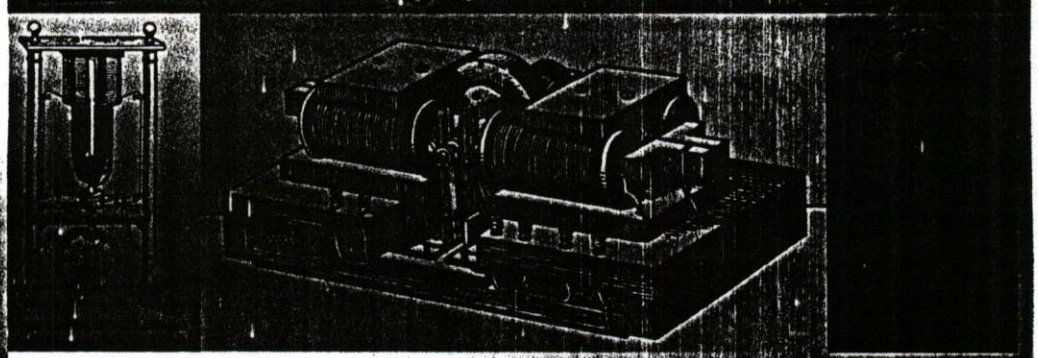
[4] Rafael C.Gonzalez and Richard E.Woods, " Digital Image Processing ", Addison-Wesley Publishing Company, 1992

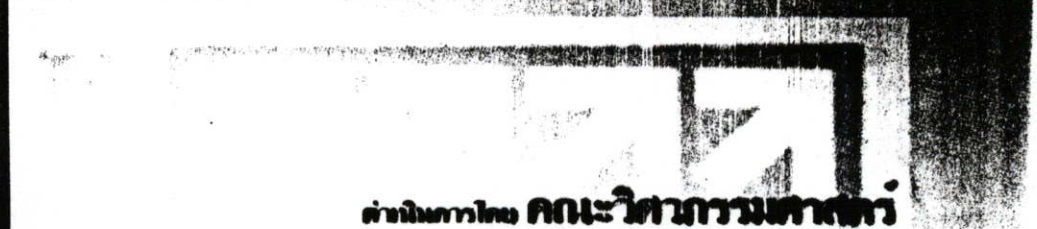
[5] S.C.Pei and C.N.Lin, " The Detection of Dominant Points on Digital Curves by Scale-Space Filtering ", Pattern Recognition, vol.25, 1992, pp.130-136

การประชุมวิชาการทางวิศวกรรมไฟฟ้าครั้งที่ 24

24th Electrical Engineering Conference (EECON-24)







ขอบเขตของบทความ

ไฟฟ้ากำลัง (PW)
อิเล็กทรอนิกส์กำลัง (PE)
ระบบควบคุมและการวัดคุม (CT)
งานวิจัยที่เกี่ยวข้องกับ
วิศวกรรมไฟฟ้า (GN)

ติดต่อสอบถามได้ที่ **คณะวิศวกรรมศาสตร์**
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

22-23 พฤศจิกายน 2544

โทร : 0-2739-2359 โทรสาร : 0-2326-9865
http://www.ladkrabang.com

กระบวนการหลังของการรู้จำวงจรโลจิกเกทที่เขียนด้วยมือ

Post-Processing Step of Hand Written Logic Circuit Recognition

เกษตร์ ศิริสันติสัมฤทธิ์ สมคิด อุนการุณวงษ์
ภาควิชาวิศวกรรมการวัดคุม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ถ.ฉลองกรุง เขตลาดกระบัง กทม. 10520
โทร 3267346-7 ต่อ 104 โทรสาร 3267346-7 ต่อ 103 E-mail: kskaset@kmitl.ac.th

บทคัดย่อ

บทความนี้เป็นการนำเสนอกระบวนการหลังของการรู้จำวงจรโลจิกเกทที่เขียนด้วยมือ ซึ่งเป็นการปรับปรุงอัลกอริทึมที่ได้นำเสนอไว้ใน [2] ลายเส้นขอบภายในวงปิดที่ผ่านการวิเคราะห์ และจำแนกประเภทตามโครงสร้างของโลจิกเกท 4 ชนิด คือ โครงสร้าง AND, OR, BUFFER และ BLOCK ในกระบวนการก่อนหน้าแล้ว จะถูกกำหนดจุดเริ่มต้นอินพุท-เอาท์พุทตามประเภทโครงสร้าง และคำนวณหาจุดศูนย์กลางของโลจิกเกทแต่ละตัว จากนั้นลากเส้นสมมุติจากจุดศูนย์กลางของโลจิกเกทแต่ละตัวผ่านจุดเริ่มต้นอินพุท-เอาท์พุทจนกระทั่งไปตัดกับเส้นโครงร่างของโลจิกเกท และติดตามลายเส้นโครงร่างเพื่อหาวงกลม NOT และจุดอินพุท-เอาท์พุทแท้จริงโดยใช้คุณสมบัติของโทโปโลยี ต่อจากนั้นวิเคราะห์การต่อวงจรระหว่างโลจิกเกทกับโลจิกเกท และโลจิกเกทกับจุดต่อร่วม และเก็บผลการวิเคราะห์การต่อวงจรในฐานข้อมูล สุดท้ายเป็นการสร้างวงจรโลจิกเกทขึ้นมาใหม่ตามมาตรฐานที่กำหนด โดยการย่อขยายขนาดของโลจิกเกทและปรับทิศทางของโลจิกเกทให้เหมาะสม และใช้รูปแบบลายเส้นต่าง ๆ ที่กำหนดไว้ รวมทั้งข้อมูลในฐานข้อมูล เพื่อทำการต่อลายวงจร ในการทดสอบอัลกอริทึมที่นำเสนอ เราได้ทดสอบกับภาพวงจรโลจิกเกทที่เขียนด้วยมือจำนวน 124 ภาพ ให้ผลลัพธ์ในการวิเคราะห์และสร้างภาพวงจรขึ้นมาใหม่ถูกต้อง 89.5 %

Abstract

This paper presents the post-processing step of hand written logic circuit recognition, which is modify the algorithm of proposed literature [2]. The close loop lines are analyzed and classified follow to four types logic structures such as AND, OR, BUFFER and BLOCK structures in pre-processing step. They are fixed the initial input-output points depend on logic structure and the center point of each logic gates are be calculated. Then, draw the virtual line from the center point of each logic gates pass through the initial input-output points, until it cross with the skeleton line of logic gate. The line tracing on the skeleton line is processed for detect the NOT-circle and actual input-output points by using the topology characteristic. After that, the circuit connection between logic gate and another one, and logic gate and junction points is analyses. The result of its analysis is stored in database. Finally, a new one is redrawn based on fixed standard by using scaling logic gate size, adjust logic gate direction and used the various line patterns and the data in database for circuit connection. For testing the proposed algorithm, we tested with 124 pictures of hand written logic circuit image. The result for circuit analysis and redrawing the new one, which its correction is about 89.5%.

Keywords: Pre-processing step, Post-processing step, Hand Written Logic Circuit Recognition

1. คำนำ

การรู้จำวงจรโลจิกเกทที่เขียนด้วยมือเป็นการประยุกต์ใช้งานการประมวลผลภาพ ทั้งนี้เพื่อเพิ่มขีดความสามารถของคอมพิวเตอร์ให้สูงขึ้น อำนวยความสะดวกต่อการทำงานของวิศวกรไฟฟ้า และประหยัดเวลาในการทำงาน การรู้จำวงจรโลจิกเกทที่เขียนด้วยมือโดยใช้หลักการแบ่งแยกลายเส้นถูกพัฒนาขึ้นโดย [1-2] เป็นการรู้จำวงจรโลจิกเกทโดยใช้องค์ประกอบของลายเส้นตรง เส้นโค้งที่ประกอบกันเป็นโลจิกเกทชนิดต่างๆ ตัวอย่างเช่น OR gate มีองค์ประกอบของลายเส้นโค้ง 3 เส้น เป็นต้น อย่างไรก็ตามวิธีการที่นำเสนอโดย [1-2] ในกระบวนการก่อนหน้า การหาจุดแบ่งแยกลายเส้น และการวิเคราะห์ลักษณะลายเส้นอาจไม่ถูกต้องอันเนื่องจากสัญญาณรบกวน (Noise) บนลายเส้น ส่งผลให้การจำแนกประเภทของโลจิกเกทผิดพลาดได้ ดังนั้นจึงมีการปรับปรุงขั้นตอนการจำแนกโครงสร้างของโลจิกเกทขึ้นโดย [3] ส่วนในกระบวนการหลังการหาอินพุท-เอาท์พุทของโลจิกเกทแต่ละตัวมีความซับซ้อนมาก และไม่สามารถหาอินพุทที่มีวงกลม NOT อยู่ด้วยได้ และการหาทิศทางหลักของโลจิกเกทยังมีข้อบกพร่อง รวมทั้งการต่อลายวงจรไม่มีการเลือกรูปแบบที่เหมาะสม และมีรูปแบบการต่อลายวงจรที่จำกัด

ในบทความฉบับนี้เป็นการนำเสนอกระบวนการหลังของการรู้จำวงจรโลจิกเกทที่เขียนด้วยมือ โดยนำข้อมูลภาพลายเส้นที่ผ่านการวิเคราะห์และจำแนกประเภทแล้วมาตรวจสอบหาจำนวนอินพุท-เอาท์พุท

DS23

ทิศทางของโลจิกเกท วิเคราะห์การต่อวงจร และการเขียนวงจรโลจิกเกทขึ้นมาใหม่เป็นต้น รายละเอียดของบทความจะเป็นดังนี้ ในหัวข้อที่ 2 เป็นการอธิบายโดยย่อของการรู้จำวงจรโลจิกเกทที่เขียนด้วยมือ หัวข้อที่ 3 เป็นการหาจำนวนของอินพุท-เอาท์พุทและทิศทางของโลจิกเกทแต่ละตัว และขั้นตอนอื่นๆ จนถึงการเขียนวงจรโลจิกเกทขึ้นมาใหม่ สุดท้ายหัวข้อที่ 4 เป็นผลการทดลอง

2. การรู้จำวงจรโลจิกเกทที่เขียนด้วยมือ

การรู้จำวงจรโลจิกเกทที่เขียนด้วยมือ [1,2] ประกอบด้วย 2 กระบวนการหลักดังนี้

2.1. กระบวนการก่อนหน้า (Pre-processing step)

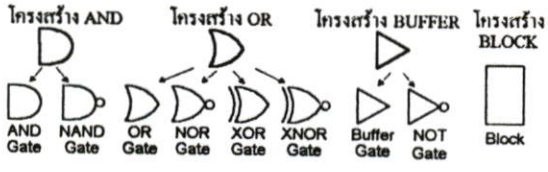
ภาพวงจรโลจิกเกทที่เขียนด้วยมือจะถูกสแกนเข้าเครื่องคอมพิวเตอร์ โดยทำให้เป็นข้อมูลภาพสองระดับ และเข้าสู่กระบวนการก่อนหน้า ซึ่งมีขั้นตอนโดยย่อดังนี้

1) ปรับปรุงคุณภาพของลายเส้น (Line enhancement) ในภาพสองระดับให้ดีขึ้น เพื่อแก้ปัญหาลายเส้นขาดและช่องว่างภายในลายเส้น

2) หาเส้นขอบภาพของลายเส้นโดยใช้อัลกอริทึมตรวจหาเส้นขอบ (Edge detection algorithm) ซึ่งจะได้เส้นขอบภาพ 2 ชนิดคือ เส้นขอบภาพที่อยู่รอบนอกและภายในของลายเส้น

3) เส้นขอบภาพภายในจะถูกกำจัดสัญญาณรบกวนที่ปะปนมาด้วยตัวกรองเกาซ์เซียน (Gaussian filter) โดยการเปลี่ยนแปลงค่าสเกล-สเปซแล้วหาจุดแบ่งลายเส้นของเส้นขอบภาพภายใน

4) เส้นขอบภาพภายในที่มีจุดแบ่งลายเส้น ≤4 จุดจะถูกวิเคราะห์ว่ามีองค์ประกอบลายเส้นใดบ้างโดยใช้วิธีการทำให้คังกระชับ (Curve fitting) และจำแนกประเภทลายเส้นตามโครงสร้างโลจิกเกทได้ 4 ประเภทคือ โลจิกเกทโครงสร้าง AND, OR, BUFFER และ BLOCK ดังรูปที่ 1



รูปที่ 1 โครงสร้างหลักของโลจิกเกท

2.2 กระบวนการหลัง (Post-processing step)

ประกอบด้วย 4 ขั้นตอนคือ การหาอินพุท-เอาท์พุทของโลจิกเกทแต่ละตัว วิเคราะห์การต่อวงจรของโลจิกเกทกับโลจิกเกท และโลจิกเกทกับจุดต่อร่วม หาทิศทางหลักและการสเกลลิ่ง (Scaling) และสุดท้ายเขียนวงจรโลจิกเกทขึ้นมาใหม่

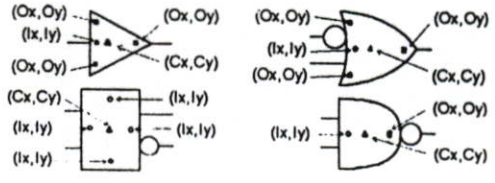
3. กระบวนการหลัง (Post-processing step)

การวิเคราะห์ของกระบวนการก่อนหน้าจะวิเคราะห์ที่ลายเส้นขอบภาพภายในวงปิด ซึ่งต่อไปนี้จะเรียกว่า เส้นขอบวงปิด ส่วนกระบวนการหลังจะวิเคราะห์ที่เส้นโครงร่าง (Skeleton) ของลายเส้น ดังนั้นต้องนำภาพลายวงจรมาหาเส้นโครงร่างโดยวิธีทำให้บาง (Thinning algorithm) [4] และทำการหาความหนาเฉลี่ยของลายเส้นโดยสมการที่ (1)

$$\text{ความหนาลายเส้นเฉลี่ย} = \frac{\text{จำนวนจุดบนลายเส้นทั้งหมด}}{\text{จำนวนจุดบนลายเส้นโครงร่าง}} \qquad (1)$$

3.1. การหาอินพุท-เอาท์พุท

ก่อนอื่นทำการกำหนดจุดเริ่มต้นอินพุท (Ix,Iy) และเอาท์พุท (Ox,Oy) บนลายเส้นขอบวงปิด โดยมีหลักเกณฑ์ในการพิจารณาดังนี้ กรณีเส้นขอบวงปิดมีโครงสร้างเป็น AND จะวางจุด (Ox,Oy) ที่จุดกึ่งกลางของลายเส้นโค้ง และวางจุด (Ix,Iy) ที่จุดกึ่งกลางของลายเส้นตรงที่อยู่ตรงข้ามกับ (Ox,Oy) กรณีโครงสร้างเป็น OR และ BUFFER วางจุด (Ox,Oy) ที่มุมทั้งสามด้าน และวางจุด (Ix,Iy) ที่จุดกึ่งกลางของเส้นโค้งหรือเส้นตรงที่อยู่ตรงข้ามกับจุด (Ox,Oy) ส่วนกรณีของโครงสร้าง BLOCK ไม่สามารถจำแนกอินพุท-เอาท์พุทได้ ดังนั้นจึงกำหนดให้มีเพียงจุดเริ่มต้น (Ix,Iy) เท่านั้นที่จุดกึ่งกลางของลายเส้นตรงแต่ละด้านดังรูปที่ 2



รูปที่ 2 ตำแหน่งจุดต่างๆ สำหรับการหาอินพุท เอาท์พุท

เมื่อได้จุดเริ่มต้นดังกล่าวจะใช้ร่วมกับจุดศูนย์กลางโลจิกเกท (Cx,Cy) ที่คำนวณได้จากสมการที่ (2) เพื่อหาลายเส้นโครงร่างของตัวโลจิกเกทเมื่อพบลายเส้นโครงร่างแล้วจะติดตามลายเส้นเพื่อหาจุดแยกต่างๆ บนลายเส้นโครงร่างแล้วนำจุดแยกเหล่านั้นมาวิเคราะห์ว่าเป็นส่วนประกอบของวงกลม NOT หรือไม่ ถ้าพบว่าเป็นจะติดตามลายเส้นโครงร่างของวงกลม NOT เพื่อหาจุดแยกที่เป็นอินพุท-เอาท์พุทของโลจิกเกทต่อไป แต่ถ้าพบว่าไม่เป็นจะพิจารณาว่าจุดแยกนั้นน่าจะเป็นอินพุท-เอาท์พุทของโลจิกเกท ซึ่งหลักการดังกล่าวข้างต้นทำให้สามารถหาอินพุท และเอาท์พุทที่มีวงกลม NOT ได้ด้วย

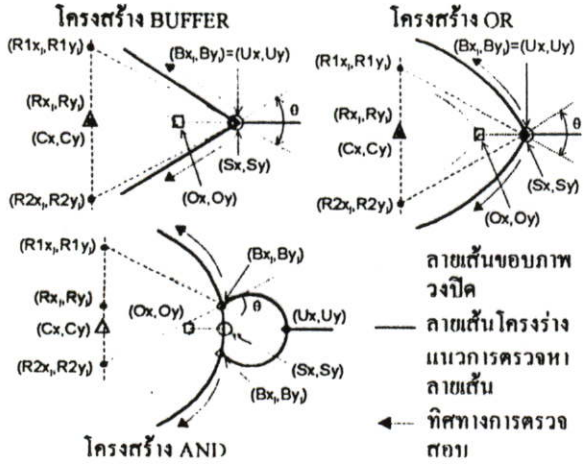
$$Cx = \frac{1}{n}\sum_{i=1}^{n} x_i \,, \qquad Cy = \frac{1}{n}\sum_{i=1}^{n} y_i \qquad (2)$$

เมื่อ x_i, y_i คือ ตำแหน่งจุดบนเส้นขอบวงปิดในแนวแกน X และ Y

n คือ จำนวนจุดทั้งหมดบนเส้นขอบวงปิด

3.1.1. การหาเอาท์พุท

โดยทั่วไปโลจิกเกทจะมีเพียง 1 เอาท์พุท และมีตำแหน่งที่แน่นอน (ยกเว้น BLOCK ที่ไม่สามารถจำแนกอินพุท-เอาท์พุทได้) ต่างจากอินพุทที่มีอย่างน้อย 1 อินพุท ดังนั้นการตรวจหาเอาท์พุทจะทำได้ง่ายกว่า และถ้าการตรวจหาเอาท์พุทไม่พบจะไม่มีการตรวจหาอินพุทอีก



รูปที่ 3 การหาเอาท์พุท

การหาเอาท์พุทจะเริ่มจากการหาลายเส้นโครงร่างของตัวโลจิกเกท โดยตรวจหาเป็นเส้นตรงจากจุดศูนย์กลางของโลจิกเกท (Cx,Cy) ผ่านจุดเริ่มต้น (Ox,Oy) ไปจนกระทั่งพบจุดของลายเส้นโครงร่าง (Sx,Sy) ดังรูปที่ 3 โดยมีกระบวนการตามอัลกอริทึมที่ 1 ซึ่งประยุกต์มาจากอัลกอริทึมการลากเส้นตรงของ Bresanham

```
Ox,Oy = Start Point.
Cx,Cy = Reference Point.
Npoints = Number of points to scan.
Line = The line must be found.

dx = Ox - Cx , dy = Oy - Cy
ix = |dx| , iy = |dy|
inc = Max( ix, iy )
pixelx = Ox , pixely = Oy
x = 0 , y = 0
Found = false

FOR ( ( i = 0 to Npoints ) & ( Found = false ) )
    x = x + ix
    IF ( x > inc )
        IF ( dx > 0)
            pixelx = pixelx + 1
        ELSE
            pixelx = pixelx - 1
        IF ( GetPixel ( pixelx, pixely ) = Line )
            Found = true
    IF ( Found = false )
        y = y + iy
        IF ( y > inc )
            IF ( dy > 0)
                pixely = pixely + 1
            ELSE
                pixely = pixely - 1
            IF ( GetPixel ( pixelx, pixely ) = Line)
                Found = true
```

อัลกอริทึมที่ 1 การตรวจหาลายเส้น (LineFinding)

จากนั้นจะตรวจสอบลายเส้นโครงร่างทั้ง 2 ด้านของจุด (Sx,Sy) เพื่อหาจุดแยกเอาท์พุท โดยมีระยะการตรวจสอบดังนี้ โลจิกเกทโครงสร้าง AND มีระยะการตรวจสอบรวม 1/2 ของความยาวลายเส้นโค้ง ส่วนโลจิกเกทโครงสร้าง OR และ BUFFER มีระยะการตรวจสอบรวม 1/9 ของขนาดโลจิกเกท ซึ่งการกำหนดระยะดังกล่าวเพื่อป้องกันความผิดพลาด เนื่องจากการเขียนเอาท์พุทไม่ตรงตำแหน่ง หรือการหาจุดเริ่มต้น (Ox,Oy) คลาดเคลื่อน เมื่อการตรวจสอบลายเส้นพบจุดแยก (Bx_j,By_j) การหาจุดต่อไปจะต้องเลือกจุดที่มีระยะห่างจากจุดเปรียบเทียบ (Rx_j,Ry_j) น้อยที่สุด โดยจุดเปรียบเทียบคำนวณได้จากสมการที่ (3) เมื่อตรวจสอบลายเส้นจนครบ (ตามอัลกอริทึมที่ 2) ตำแหน่งจุดแยกทุกจุดจะถูกเก็บไว้

$$Rx_j = (Bx_j - Sx) + Cx, \quad Ry_j = (By_j - Sy) + Cy \qquad (3)$$

เมื่อ j = จุดแยกจุดที่ 1, 2, 3, ..., n

```
Sx,Sy = Start point.
Cx, Cy = Center point.
Npoints = Number of points to scan.

FOR ( i = 0 to 1 )
    px = Sx , py = Sy
    FOR ( j = 0 to Npoints/2 )
        Np = p1+p2+p3...+p8 (Neighbour points of (px,py))
        Sp = (p1^p2) + (p2^p3) + ... + (p8^p1) ( ^ is Xor)
        IF ( ( Np > 2 ) & ( Sp > 4 ) )
            Keep bifurcation position ( px, py ).
            (p1x, p1y) = NextPointFinding( px, py )
            (p2x, p2y) = NextPointFinding( px, py )
            Rx = Cx + (px - Sx), Ry = Cy + (py - Sy)
            I1 = Distance( (p1x, p1y), (Rx, Ry) )
            I2 = Distance( (p2x, p2y), (Rx, Ry) )
            IF ( I1 < I2 )
                px = p1x , py = p1y
            ELSE
                px = p2x , py = p2y
        ELSE
            (px, py) = NextPointFinding(px, py)
```

อัลกอริทึมที่ 2 การตรวจสอบจุดแยกบนลายเส้น (BifurcationFinding)

เมื่อได้ตำแหน่งและจำนวนจุดแยกมาจะตรวจหาวงกลม NOT บริเวณจุดแยกแต่ละจุด โดยตรวจหาเป็นเส้นตรงสองทิศทางออกจากจุดแยกมีมุมต่างกัน 53 องศา (θ) ซึ่งใช้อัลกอริทึมที่ 1 เช่นเดียวกับการหาลายเส้นโครงร่าง โดยใช้จุดแยกเป็นจุดเริ่มต้นและมีระยะการหาวงกลม NOT เท่ากับความหนาเฉลี่ยของลายเส้นที่คำนวณได้จากสมการที่ (1) ส่วนจุด $(R1x_j,R1y_j)$ และ $(R2x_j,R2y_j)$ คำนวณได้ดังนี้

จุดอ้างอิงที่ 1

$$R1x_j = Rx_j + (Sy - Cy)/2$$
$$R1y_j = Ry_j + (Sx - Cx)/2 \qquad (4)$$

จุดอ้างอิงที่ 2

$$R2x_j = Rx_j - (Sy - Cy)/2$$
$$R2y_j = Ry_j - (Sx - Cx)/2 \qquad (5)$$

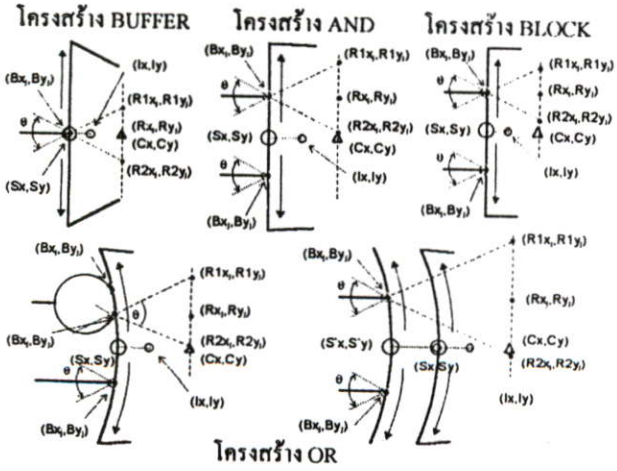
การตรวจหาถ้าพบวงกลม NOT จะต้องตรวจสอบขนาดวงกลม NOT ว่าเหมาะสมกับตัวโลจิกเกทหรือไม่ โดยโลจิกเกทโครงสร้าง AND - OR และ BLOCK ควรมีวงกลม NOT เล็กกว่า 1/3 ของขนาดโลจิกเกท ส่วนโครงสร้าง BUFFER ควรมีวงกลม NOT เล็กกว่าขนาดของโลจิกเกท ถ้าตรงตามเงื่อนไขจะตรวจสอบจุดแยกเอาท์พุทบนลายเส้นโครงร่างที่ล้อมรอบวงกลม NOT ตามอัลกอริทึมที่ 2 โดยใช้จุดศูนย์กลางของวงกลม NOT ที่สามารถคำนวณได้จากสมการที่ (2) แทนจุด (Cx,Cy) และใช้จุดแยกที่ตรวจสอบแทนจุด (Sx,Sy) เมื่อตรวจสอบจนสุดลายเส้นบรรจบกับจุดแยกใด จุดแยกนั้นจะถูกลบออก แต่ถ้าการตรวจหาไม่พบวงกลม NOT หรือพบวงกลม NOT ไม่เหมาะสมจะเก็บจุดแยกนั้นเป็นเอาท์พุท กระบวนการที่กล่าวมาเบื้องต้นแสดงไว้ในอัลกอริทึมที่ 3

```
Bx[], By[] = Bifurcation points.
Nbifurcation = Number of bifurcation points.
Cx , Cy = Center of gate loop.
NOTsize = NOT loop size.
Sx, Sy = Start point on skeleton.

FOR ( i = 0 to Nbifurcation )
    Rx = Cx + (Bx[i] – Sx),  Ry = Cy + (By[i] – Sy)
    R1x = Rx + (Sy – Cy)/2,  R1y = Ry + (Sx – Cx)/2
    Found = LineFinding( (Bx[i],By[i]), (R1x,R1y), loop line,
                         line thickness )
    IF ( Found = false )
        R2x = Rx - (Sy – Cy)/2,  R2y = Ry - (Sx – Cx)/2
        Found = LineFinding( (Bx[i],By[i]), (R2x,R2y), loop
                             line, line thickness )
    IF ( Found = true )
        IF ( NOT loop size = NOTsize )
            BifurcationFinding( (Bx[i], By[i]), (Cx, Cy) )
```

อัลกอริทึมที่ 3 การตรวจหาวงกลม NOT (NOTCircleFinding)

3.1.2. การหาอินพุท

การหาอินพุทของโลจิกเกทจะใช้วิธีการเดียวกันกับการหาเอาท์พุทเพียงแต่มีจุดเริ่มต้นเป็นจุด (Ix,Iy) และมีระยะของการตรวจหาจุดแยกบนลายเส้นโครงร่างเท่ากับความยาวของด้านที่ตรวจสอบ ดังรูป



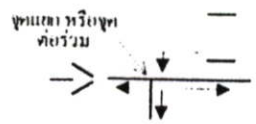
รูปที่ 4 การหาอินพุท

กรณีโลจิกเกทเป็น Exclusive การหาอินพุทจะไม่พบ ดังนั้นจะต้องหาลายเส้น Exclusive และตรวจหาอินพุทบนลายเส้นดังกล่าว ซึ่งสามารถทำได้โดยใช้วิธีการแบบเดียวกับหาเอาท์พุท เพียงแต่กำหนดจุดเริ่มต้นที่จุด (Sx,Sy) และกำหนดระยะการตรวจหาลายเส้น Exclusive เท่ากับระยะทางระหว่างจุดศูนย์กลาง (Cx,Cy) ของโลจิกเกทกับจุดเริ่มต้น (Sx,Sy) เมื่อผ่านกระบวนการตรวจหาอินพุท-เอาท์พุทแล้วลายเส้นของอินพุท-เอาท์พุทจะถูกตัดออกจากตัวโลจิกเกทดังรูปที่ 5 เพื่อลดความซับซ้อนในการวิเคราะห์การต่อวงจรของโลจิกเกทแต่ละตัวในขั้นตอนต่อไป

รูปที่ 5 ก่อนและหลังการหาอินพุท-เอาท์พุทของ XOR gate

3.2. การวิเคราะห์การต่อลายวงจร

การวิเคราะห์การต่อลายวงจรจะเริ่มที่ตำแหน่งอินพุท-เอาท์พุทของโลจิกเกทแต่ละตัว และตรวจสอบตามลายเส้นจนกว่าจะพบจุดแยกหรือสุดลายเส้นตามอัลกอริทึมที่ 4 ซึ่งการตรวจสอบถ้าไม่พบจุดแยกจุดสุดท้ายของลายเส้นจะถูกนำไปเปรียบเทียบกับอินพุท-เอาท์พุทของโลจิกเกททุกตัวเพื่อตรวจสอบการเชื่อมต่อกันระหว่างโลจิกเกท โดยอินพุท-เอาท์พุทของโลจิกเกทที่ถูกเชื่อมต่อจะไม่นำมาตรวจสอบอีก แต่ถ้าการเปรียบเทียบไม่ตรงแสดงว่าอินพุท-เอาท์พุทของโลจิกเกทที่ตรวจสอบไม่มีการต่อ (ปล่อยลอย) กรณีที่การตรวจสอบลายเส้นพบจุดแยก (จุดต่อร่วม) จะเก็บตำแหน่งจุดแยกนั้น และเริ่มตรวจสอบลายเส้นทุกลายเส้นที่เป็นสาขาของจุดแยกด้วยวิธีการเช่นเดิม (อัลกอริทึมที่ 5) ดังรูปที่ 6 เมื่อวิเคราะห์โลจิกเกทครบทุกตัวจะทำให้เรารู้การต่อวงจรทั้งหมด



รูปที่ 6 การตรวจสอบลายเส้นเมื่อพบจุดแยก

```
Sx, Sy = Start point.

Stop = false
DO
    Nx, Ny = NextPointFinding(Sx, Sy)
    IF ( (Nx != Sx) || (Ny != Sy) )
        Np = p1 + p2 + p3... + p8 (neighbour points of (px,py))
        Sp = (p1 ^ p2) + (p2 ^ p3) + ... + (p8^p1) ( ^ is Xor)
        IF ( ( Np >= 2 ) & ( Sp >= 4 ) )
            Stop = true
    ELSE
        Stop = true
    Sx = Nx , Sy = Ny
WHILE( Stop = false )
```

อัลกอริทึมที่ 4 การตรวจสอบตามลายเส้น (LineTracing)

```
Bx, By = Bifurcation point.
Nbranch = Number of Branch.

FOR ( i = 0 to Nbranch )
    FoundBifurcation = LineTracing( Bx, By )
    IF ( FoundBifurcation = true )
        Bifurcation connect to bifurcation.
        BifurcationBranchCheck(New bifurcation point,
                               New bifurcation branch)
    ELSE
        IF(Last point is input-output of gates )
            Bifurcation connect to gate.
```

อัลกอริทึมที่ 5 การตรวจสอบลายเส้นสาขาของจุดแยก (BifurcationBranch Check)

3.3. การปรับทิศทาง และการสเกลลิ่ง

ภาพโลจิกเกทที่เขียนด้วยมือโดยทั่วไปจะไม่สมมาตร เอียง และมีทิศทางแตกต่างกัน ดังนั้นจำเป็นต้องหาทิศทางของโลจิกเกทแต่ละตัวเพื่อจัดกลุ่มและ หาทิศทางหลักของโลจิกเกท (α) ซึ่งจะนำไปใช้ในการปรับทิศทางให้เหมาะสม เพื่อการวาดภาพใหม่ การคำนวณทิศทางของโลจิกเกทจะใช้ตำแหน่งของจุด 2 จุด คือ จุดศูนย์กลาง (Cx,Cy) กับจุดเอาท์พุท (Ux,Uy) ของโลจิกเกท ยกเว้น BLOCK ที่ไม่สามารถจำแนกอินพุท-เอาท์พุทได้ จึงต้องใช้จุดกึ่งกลางลายเส้นด้านยาวด้านใดด้านหนึ่ง (จุดเดียวกันกับการหาอินพุท-เอาท์พุท) แทน และคำนวณโดยใช้สมการหาทิศทางดังนี้

$$\text{ทิศทางของเกท} = tan^{-1}\left(\frac{Uy - Cy}{Ux - Cx}\right) \qquad (6)$$

เมื่อคำนวณทิศทางของโลจิกเกทครบทุกตัวจะจัดโลจิกเกทออกเป็นกลุ่มๆ โดยโลจิกเกทที่มีทิศทางต่างกันน้อยกว่า 45 องศาจะอยู่กลุ่มเดียวกัน จากนั้นจะหาทิศทางหลัก (α) จากการคำนวณทิศทางเฉลี่ยของกลุ่มที่มีจำนวนโลจิกเกทมากที่สุด ซึ่งทิศทางหลักที่ได้นี้จะนำมาใช้เป็นทิศทางอ้างอิงเพื่อจัดกลุ่มของโลจิกเกทใหม่ โดยจัดเป็น 4 กลุ่มซึ่งแต่ละกลุ่มมีทิศทางต่างกัน 90 องศา ซึ่งกลุ่มแรกมีทิศทางต่างจากทิศทางหลักอยู่ในช่วง $\pm$45 องศา ส่วนกลุ่มที่สอง, สาม และสี่ มีทิศทางดังรูปที่ 7 จากนั้นจะปรับหรือหมุนตำแหน่งของจุดต่างๆ เช่น ตำแหน่งโลจิกเกท, อินพุท-เอาท์พุท, จุดตัดหรือจุดเชื่อม และอื่นๆ โดยมีองศาในการหมุนเท่ากับทิศทางหลัก และสามารถคำนวณตำแหน่งใหม่ได้จาก

$$x' = (x - ICx)Cos(\alpha) - (y - ICy)Sin(\alpha) + ICx$$
$$y' = (y - ICy)Cos(\alpha) - (x - ICx)Sin(\alpha) + ICy \qquad (7)$$

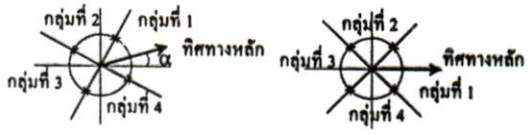
เมื่อ (ICx,ICy) คือจุดศูนย์กลางของภาพ

เนื่องจากตัวโลจิกเกท และภาพวงจรที่รับเข้ามามีขนาดไม่แน่นอน แต่การวาดภาพวงจรใหม่จะมีขนาดของตัวโลจิกเกทคงที่ ดังนั้นจะต้องหาอัตราการสเกลลิ่ง (Scaling) ภาพ โดยจะหาขนาดเฉลี่ยของโลจิกเกทเฉพาะโครงสร้าง AND และ OR เทียบกับขนาดของ AND หรือ OR gate มาตรฐาน (AND และ OR gate มาตรฐานมีขนาดเท่ากัน) ดังนี้

$$Scale = \frac{\text{ขนาดเฉลี่ยของ AND และ OR gate ในภาพ}}{\text{ขนาดของโลจิกเกทมาตรฐาน}} \qquad (8)$$

ตำแหน่งใหม่ของจุดต่างๆ คำนวณดังนี้

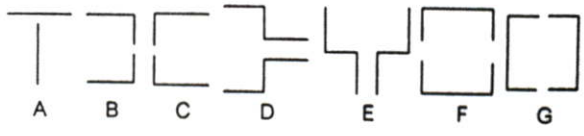
$$x'' = \frac{x'}{Scale}$$
$$y'' = \frac{y'}{Scale} \qquad (9)$$



รูปที่ 7 การจัดกลุ่มก่อน และหลังการปรับทิศทาง

3.4. การวาดภาพ

เมื่อปรับตำแหน่งต่างๆ แล้ว ทิศทางหลักของโลจิกเกทจะเป็น 0 องศา ดังนั้นโลจิกเกทที่อยู่ในกลุ่มต่างๆ จะมีทิศทางดังนี้คือ โลจิกเกทที่อยู่ในกลุ่มที่ 1 จะมีทิศทาง 0 องศา โลจิกเกทที่อยู่ในกลุ่มที่ 2, 3 และ 4 จะมีทิศทาง 90, 180 และ 270 องศาตามลำดับ ส่วนการวาดภาพโลจิกเกทใหม่จะวาดในตำแหน่ง และทิศทางตามข้อมูลที่ได้ โดยลำดับ และตำแหน่งของอินพุทใหม่จะต้องสัมพันธ์กับลำดับ และตำแหน่งอินพุทที่ได้จากการวิเคราะห์ เพื่อป้องกันการต่อลายวงจรผิดพลาด



รูปที่ 8 รูปแบบลายเส้น

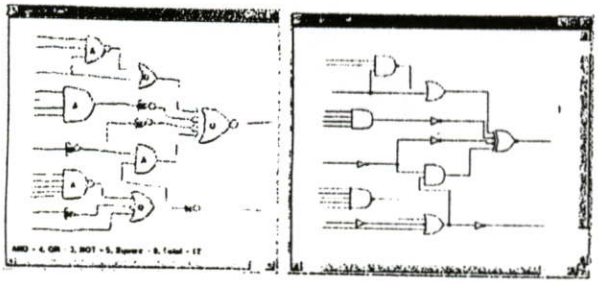
การต่อลายวงจรโดยทั่วไปจะลากเส้นในแนวแกน X และแกน Y เท่านั้น โดยมีรูปแบบของลายเส้นหลายรูปแบบเพื่อความเหมาะสมดังรูปที่ 8 ในการลากเส้นทุกรูปแบบจะมีการตรวจสอบทุกๆ จุดในแนวเส้นที่ลากผ่าน เพื่อป้องกันการลากทับลายเส้นหรือตัวโลจิกเกท ด้วยเหตุนี้การต่อลายวงจรโดยใช้รูปแบบลายเส้นเพียงรูปแบบเดียวอาจต่อลายวงจรไม่สำเร็จ จึงจำเป็นต้องมีรูปแบบลายเส้นสำรองหลายรูปแบบ และทดลองต่อลายวงจรทีละแบบจนกว่าจะต่อลายวงจรสำเร็จ ส่วนการเลือกใช้รูปแบบลายเส้นที่เหมาะสมเพื่อต่อลายวงจรจะต้องพิจารณาทิศทางของโลจิกเกทที่เชื่อมต่อกันดังนี้

ตารางที่ 1 การเลียกรูปแบบต่อลายเส้น

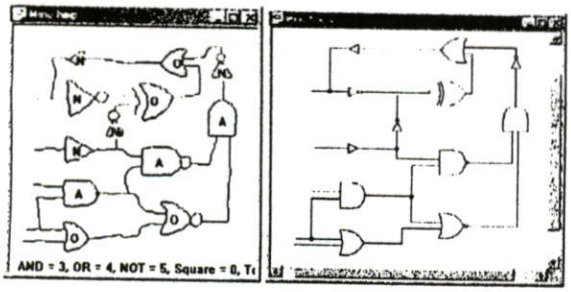
โลจิกเกททิศทาง	โลจิกเกททิศทาง	ใช้รูปแบบลายเส้น
0	0	A,D,B,C,E,F,G
0	90	A,B,C,E,D,F,G
0	180	A,B,C,E,D,F,G
0	270	A,B,C,E,D,F,G
90	90	A,E,C,B,D,G,F
90	180	A,B,C,E,D,F,G
90	270	A,C,B,D,E,G,F
180	180	A,D,B,C,E,F,G
180	270	A,B,C,E,D,F,G
270	270	A,E,C,B,D,G,F
โลจิกเกททิศทางใดๆ	จุดแยก	A,B,C,D,E,F,G
จุดแยก	จุดแยก	A,B,C,D,E,F,G

4. ผลการทดลอง

ในการทดลองเราใช้ภาพวงจรโลจิกเกทที่เขียนด้วยมือ ซึ่งมีขนาดของภาพ, ทิศทางของโลจิกเกท และลายวงจรต่างๆ กัน โดยนำภาพวงจรมาผ่านกระบวนการก่อนหน้า จากนั้นจึงนำมาทดลองกับกระบวนการหลัง และมีผลการทดลองดังรูปที่ 9 ถึง 10 โดยมีขนาดของโลจิกเกทมาตรฐาน 120 จุด



รูปที่ 9 ผลของกระบวนการก่อนหน้าและกระบวนการหลัง



รูปที่ 10 ผลของกระบวนการก่อนหน้าและกระบวนการหลัง

การทดลองได้ทดลองกับภาพวงจรจำนวน 124 ภาพ ให้ผลลัพธ์ในการหาอินพุท-เอาท์พุท การวิเคราะห์วงจร และการสร้างภาพวงจรขึ้นมาใหม่ถูกต้องรวมจำนวน 111 ภาพ หรือ 89.5 เปอร์เซ็นต์ โดยผิดพลาด 13 ภาพ หรือ 10.5 เปอร์เซ็นต์ อันเนื่องมาจากการวาดอินพุทของโลจิกเกทใกล้จุดมุม (ชิดขอบของตัวโลจิกเกทเกท) การวาดวงกลม NOT ติดกันที่อินพุท และการต่อลายเส้นไม่สำเร็จที่จุดแยกที่มีสี่สาขา

5. สรุป

เราได้นำเสนอกระบวนการหลังของการรู้จำวงจรโลจิกเกทที่เขียนด้วยมือ โดยพัฒนาอัลกอริทึมให้มีประสิทธิภาพ และเพิ่มขีดความสามารถในการประมวลผลในขั้นตอนต่างๆ ของกระบวนการหลังให้มากขึ้น สามารถตรวจหาทิศทางของโลจิกเกท จำนวนอินพุท และเอาท์พุทได้โดยอัลกอริทึมเดียวกัน และลดความซับซ้อนในการประมวลผล นอกจากนี้วิธีที่นำเสนอยังช่วยให้การรู้จำวงจรโลจิกเกทถูกต้องมากยิ่งขึ้น

เอกสารอ้างอิง

[1] เกษตร์ ศิริสันติสัมฤทธิ์ และ อรุณ กำเหนิดนนท์ "การจดจำวงจรโลจิกเกทที่เขียนด้วยมือ โดยใช้หลักการแบ่งแยกลายเส้น" การประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 16 พฤศจิกายน 2536 หน้า 297-301

[2] อโณทัย ปฐมมานิศ สมคิด อุนการุณวงษ์ เกษตร์ ศิริสันติสัมฤทธิ์ "การจดจำวงจรโลจิกเกทที่เขียนด้วยมือ โดยใช้หลักการแบ่งแยกลายเส้น" การประชุมวิชาการวิศวกรรมไฟฟ้า ครั้งที่ 17 ธันวาคม 2537 หน้า 309-313

[3] เดชา พัฒนประสิทธิ์ชัย สมคิด อุนการุณวงษ์ ธีรชาติ แสนชัย เกษตร์ ศิริสันติสัมฤทธิ์ "การวิเคราะห์โครงสร้างของวงปิดสำหรับการจดจำวงจรโลจิกเกทที่เขียนด้วยมือ" การประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 23 พฤศจิกายน 2543 หน้า 473-476

[4] ZhangT.Y.and Suen C.Y. "A Fast Parallel Algorithm for Thinning Digital Pattern" Comm.ACM, Vol.27, No.3; pp.236-239; 1984

เกษตร์ ศิริสันติสัมฤทธิ์ ปัจจุบันดำรงตำแหน่งรองศาสตราจารย์ระดับ 8 ภาควิชาวิศวกรรมการวัดคุม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง สนใจงานวิจัยทางด้าน Digital Image Processing, Computer Vision, Pattern Recognition, Speech Coding, Industrial Process Instrumentation.

สมคิด อุนการุณวงษ์ สำเร็จการศึกษาระดับปริญญาตรี คณะวิศวกรรมศาสตร์ สาขาคอมพิวเตอร์ชุดสาหกรรม จากสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ปัจจุบันกำลังศึกษาระดับปริญญาโท สาขาวิศวกรรมการวัดคุม สนใจงานวิจัยทางด้าน Pattern Recognition, Digital Image Processing.

ประวัติผู้เขียน

นายสมคิด อุนการุณวงษ์ เกิดเมื่อวันที่ 3 กุมภาพันธ์ พ.ศ. 2512 ที่จังหวัดอุทัยธานี สำเร็จการศึกษาอุตสาหกรรมศาสตร์บัณฑิต สาขาคอมพิวเตอร์อุตสาหกรรม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง เมื่อปีการศึกษา 2536

ปี พ.ศ. 2532 เข้าทำงานที่บริษัทกระเบื้องทิพย์ ตำแหน่งหัวหน้าหน่วยผลิต ปัจจุบันกำลังศึกษาระดับปริญญาโท สาขาเทคโนโลยีการวัดคุม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง