

การรู้จำวงจรลอจิกเกตที่เขียนด้วยมือ โดยใช้หลักการแบ่งแยกสายเส้น

เกษตร์ ศิริสันติสัมฤทธิ์

รองศาสตราจารย์

ภาควิชาวิศวกรรมการวัดคุม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

สมคิด อุนการณวงษ์

นักศึกษาปริญญาโท

ภาควิชาวิศวกรรมการวัดคุม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

บทคัดย่อ

บทความนี้เป็นการนำเสนอ การรู้จำวงจรลอจิกเกตที่เขียนด้วยมือ โดยใช้หลักการแบ่งแยกสายเส้น ซึ่งประกอบด้วยสองขั้นตอนหลักดังนี้คือ ขั้นตอนแรก ภาพวงจรลอจิกเกตจะถูกปรับปรุงคุณภาพของสายเส้นที่อาจขาดหรือมีช่องว่างในสายเส้น จากนั้นหาเส้นขอบและเส้นโครงร่างของสายเส้น ซึ่งเส้นขอบภายในวงปิดจะถูกนำมาวิเคราะห์ และจำแนกประเภทตามโครงสร้างหลักของลอจิกเกต 4 ชนิดคือ AND OR BUFFER และ BLOCK ขั้นตอนที่สอง สายเส้นที่จำแนกประเภทตามโครงสร้างหลักของลอจิกเกตแล้วจะถูกนำมาวิเคราะห์หาจำนวนอินพุท-เอาต์พุท และทิศทางของลอจิกเกตแต่ละตัว วิเคราะห์การต่อสายวงจรระหว่างลอจิกเกตกับลอจิกเกต และลอจิกเกตกับจุดต่อร่วม จากนั้นเขียนวงจรลอจิกเกตขึ้นมาใหม่โดยปรับทิศทาง และสเกลลิ่งลอจิกเกตทุกตัวให้เหมาะสม ต่อวงจรระหว่างลอจิกเกตกับลอจิกเกต และลอจิกเกตกับจุดต่อร่วมตามรูปแบบที่เก็บไว้ในฐานข้อมูล ความสามารถของอัลกอริทึม ของการรู้จำวงจรลอจิกเกตที่เขียนด้วยมือที่นำเสนอได้แสดงไว้ในผลการทดลอง

Hand Written Logic Circuit Recognition Using Line Segmentation

Kaset Sirisantisamrid

Associate Professor

Department of Instrumentation Engineering

Faculty of Engineering King Mongkut's Institute of Technology Ladkrabang

Somkid Unkaroonwong

Master Student

Department of Instrumentation Engineering

Faculty of Engineering King Mongkut's Institute of Technology Ladkrabang

Abstract

The following is a presentation of Hand Written Logic Circuit Recognition using Line Segmentation, which consist of two basic steps. In first step, the quality of the logic circuit image is enhanced e.g. line with holes or gap. Then the edge and skeleton lines of the logic circuit image are detected, the edge lines in closed loops are analyzed and classified according to basic structure of logic gate as 4 types AND, OR, BUFFER or BLOCK. In second step, the lines classified according to the basic structure of logic gate are analyzed for numbers of input and output, direction of each logic gates, and connection with other logic gates and line junctions. Then the logic circuit is redrawn, adjusting the direction and scaling appropriately, and connecting the circuit between different logic gates and line junctions according to the format in the database. The effective of the algorithm for Hand Written Logic Circuit Recognition has been demonstrated by experiment.

Keywords: line segmentation, logic gate, line junction

บทนำ

การรู้จำวงจรโลจิกเกตที่เขียนด้วยมือ เป็นการประมวลผลภาพทางดิจิทัล (digital image processing) เพื่อให้คอมพิวเตอร์สามารถรู้จำ (recognition) ได้ว่าลายเส้นในข้อมูลภาพมีโลจิกเกตชนิดใดและเป็นจำนวนเท่าใด แต่ละตัวมีทิศทางเป็นอย่างไร

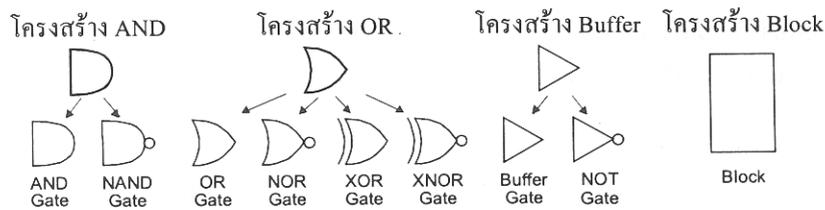
ลอจิกเกตกับลอจิกเกต และลอจิกเกตกับจุดต่อร่วม ต่อถึงกันหรือไม่ เมื่อสายเส้นต่างๆ ถูกวิเคราะห์ และรู้จำได้แล้วจะทำการวาดวงจรลอจิกเกตขึ้นมาใหม่ตามมาตรฐานที่กำหนดไว้ ทั้งนี้ก็เพื่ออำนวยความสะดวก เพิ่มความรวดเร็วในการทำงานของวิศวกรไฟฟ้า และลดต้นทุนการผลิต

การรู้จำวงจรลอจิกเกตที่เขียนด้วยมือ โดยใช้หลักการแบ่งแยกสายเส้นได้ถูกพัฒนาขึ้นโดย รศ.เกษตร์ ศิริสันติสัมฤทธิ์ และคณะฯ[1] และอโณทัย ปฐมมานิต และคณะฯ[2] เป็นการรู้จำโดยพิจารณาจากองค์ประกอบของสายเส้นตรง เส้นโค้งที่ประกอบกันเป็นลอจิกเกต ตัวอย่างเช่น OR gate ประกอบด้วยเส้นโค้งสามเส้น เป็นต้น การพัฒนาอัลกอริทึม (Algorithm) เพื่อการรู้จำวงจรลอจิกเกตที่เขียนด้วยมือโดย รศ.เกษตร์ ศิริสันติสัมฤทธิ์ และคณะฯ[1] และอโณทัย ปฐมมานิต และคณะฯ[2] ยังมีข้อจำกัดหลายประการกล่าวคือ การหาจุดแบ่งแยกสายเส้นไม่ถูกต้องเนื่องจากสัญญาณรบกวนบนสายเส้น (noise) ทำให้การวิเคราะห์องค์ประกอบของสายเส้นผิดพลาด การวิเคราะห์จำนวนอินพุต-เอาต์พุตมีความซับซ้อน และใช้อัลกอริทึมเดียวกันไม่ได้ อีกทั้งยังไม่สามารถตรวจหาวงกลม NOT ที่มีอยู่ที่อินพุตได้

บทความนี้เป็นคำแนะนำเสนอการรู้จำวงจรลอจิกเกตที่เขียนด้วยมือด้วย 2 ขั้นตอน ซึ่งเป็นการพัฒนาปรับปรุงอัลกอริทึมในขั้นตอนต่างๆ ให้ดีขึ้น ลดความซับซ้อนสามารถใช้อัลกอริทึมเดียวกันในการตรวจหาอินพุตและเอาต์พุตที่มีวงกลม NOT ได้ รายละเอียดของการรู้จำวงจรลอจิกเกตที่เขียนด้วยมือจะเป็นดังนี้

ขั้นตอนแรก (First Step)

เมื่อภาพวงจรโลจิกเกตที่เขียนด้วยมือ ถูกสแกนเข้าเครื่องคอมพิวเตอร์เป็นข้อมูลภาพสองระดับ จะถูกประมวลผลในขั้นตอนแรกดังนี้คือ ปรับปรุงคุณภาพของลายเส้นให้สมบูรณ์โดยการทำให้ลายเส้นบาง (thinning) และการทำให้ลายเส้นเบลอ (blur) จากนั้นแยกตัวโลจิกเกตออกจากวงจรโดยการหาขอบภาพ (edge detection) ซึ่งจะได้ลายเส้นขอบที่มีลักษณะเป็นวงปิด หรือเรียกว่า เส้นขอบวงปิดหลายๆ เส้น ต่อจากนั้นเส้นขอบวงปิดแต่ละเส้นจะถูกวิเคราะห์หาจุดแบ่งแยกลายเส้น หรือจุดมุมโดยใช้ตัวกรองเกาส์เซียน (Gaussian filter) และวิเคราะห์ลักษณะลายเส้นโดยการทำให้โค้งกระชับ (curve fitting) ด้วยวิธีกำลังสองน้อยสุด (method of least squares) ทำให้รู้โครงสร้างของเส้นขอบวงปิดนั้น และสามารถจำแนกได้เป็น 4 ชนิดคือ โครงสร้าง AND OR BUFFER และ BLOCK ดังรูปที่ 1 วิธีการที่กล่าวมาทั้งหมดมีรายละเอียดดังต่อไปนี้



รูปที่ 1 โครงสร้างหลักของโลจิกเกต

การปรับปรุงลายเส้น

เนื่องจากภาพข้อมูลที่ใช้เป็นภาพสองระดับ (binary image) จะมีขนาดและความสม่ำเสมอของลายเส้นในภาพแตกต่างกัน รวมทั้งอาจมีช่องว่าง (hole) บนลายเส้นซึ่งทำให้เกิดปัญหาในการวิเคราะห์เช่น ลายเส้นที่มีขนาดน้อยกว่า 3 จุดภาพ (Pixel) การหาขอบภาพจะไม่สามารถแยกตัวโลจิกเกตออกจากวงจรได้ ดังนั้น จึงจำเป็นต้องมีการปรับปรุงลายเส้นให้สมบูรณ์เพื่อป้องกันปัญหาดังกล่าว โดยประยุกต์ใช้วิธีการทำให้ลายเส้นบาง (thinning algorithm) และวิธีการทำให้ลายเส้นเบลอ (blur algorithm) ทั้งนี้เพื่อรักษาข้อมูลภาพให้ผิดเพี้ยนจากเดิมน้อยที่สุด (รูปที่ 5) ซึ่งภาพที่ได้จะมีขนาดของลายเส้นมากกว่าหรือเท่ากับ 3 จุดภาพเสมอ รายละเอียดของวิธีการมีดังนี้

ก) การทำลายเส้นบาง (Thinning Algorithm)

การทำลายเส้นบางเป็นวิธีการลดจุดบริเวณขอบของลายเส้น ซึ่งเป็นวิธีการหนึ่งในการหาสายโครงร่าง (skeleton) ของภาพ[5] โดยใช้ Mask operator ในรูปที่ 2 และกำหนดให้ "1" แทนจุดลายเส้น "0" แทนจุดพื้น โดยมีกระบวนการตามอัลกอริทึมที่ 1 และจะประมวลผลเพียง 1 รอบ ซึ่งผลของการทำลายเส้นบางแสดงในรูปที่ 3

```

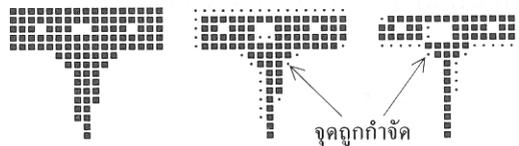
For y = 0 to Image Height
  For x = 0 to Image Width
    If (Pixel(x,y) = 1)
      If ((p1&p3&p7) = 0)&((p1&p5&p7) = 0)
        NTp = p1+p2+...+p8
        STp = (p1^p2)+(p2^p3)+...+(p8^p1)
        If (2 ≤ NTp ≤ 6)&(STp = 2)
          Flag the Pixel(x,y).
  For y = 0 to Image Height
    For x = 0 to Image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

p8	p1	p2
p7	TP	p3
p6	p5	p4



รูปที่ 2 Mask Operator ขนาด 3x3

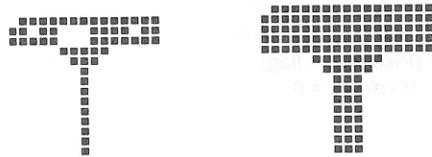
รูปที่ 3 ก่อนและหลังการทำลายเส้นบาง

ข) การทำลายเส้นเบลอ (Blur Algorithm)

การทำลายเส้นเบลอเป็นวิธีการเพิ่มจุดบริเวณขอบของลายเส้น และมีผลทำให้รูปร่างปลีกล้อยที่เกิดขึ้นจากสัญญาณรบกวนที่ขอบของลายเส้นและช่องว่าง (น้อยกว่า 9 จุด) บนลายเส้นหายไป เนื่องจากเป็น software filter[8] วิธีหนึ่ง โดยมีวิธีการตามอัล กอริทึมที่ 2 และผลที่ได้แสดงดังรูปที่ 4

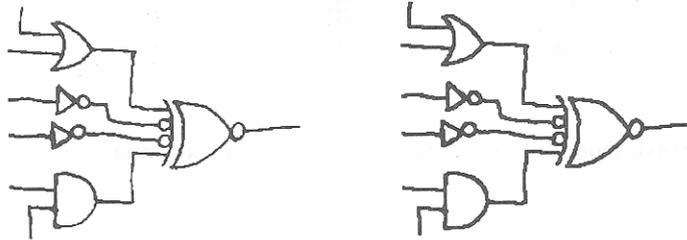
```

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 2 การทำลายเส้นเบลอ

รูปที่ 4 ก่อนและหลังการทำลายเส้นเบลอ



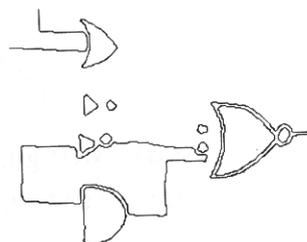
รูปที่ 5 ภาพอินพุตและผลของการปรับปรุงลายเส้น

การแยกตัวโลจิกเกตออกจากวงจร

การแยกตัวลอจิกเกตออกจากวงจรถลอจิกเพื่อวิเคราะห์โครงสร้าง สามารถทำได้โดยการหาขอบภาพ เนื่องจากสายเส้นขอบจะมีขนาดเพียง 1 จุดภาพ และไม่มีการต่อเชื่อมกับสายเส้นอื่นๆ ทำให้ลดความซับซ้อนในการวิเคราะห์กระบวนการหาขอบของภาพ แสดงไว้ในอัลกอริทึมที่ 3 และเมื่อภาพผ่านการหาขอบจะได้สายเส้นขอบวงปิดหลายๆเส้น โดยมีทั้งที่เป็นตัวลอจิกเกต และลายวงจรถลอจิกที่ 6

```

For y = 0 to Image Height
  For x = 0 to Image Width
    If (Pixel(x,y) = 1)
      If ((p1&p3&p5&p7) = 0)&((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 การหาขอบของภาพ

รูปที่ 6 ผลการหาขอบภาพ

การหาจุดแบ่งแยกสายเส้น

การหาจุดแบ่งแยกสายเส้น หรือจุดมุมของเส้นขอบวงปิดสามารถหาได้โดยใช้ตัวกรองเกาซ์เซียนที่นำเสนอในบทความ[3] เนื่องจากวิธีนี้เป็นการทำลายเส้นให้ราบเรียบ (smoothing) ที่มีประสิทธิภาพสูง ทำให้การหาจุดมุมมีความเที่ยงตรงโดยสมการของตัวกรองเกาซ์เซียน และการประสาน (convolution) กับสายเส้นนิยามไว้ดังสมการที่ (1) และ (2)

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

$$\begin{aligned} X(t, \sigma) &= x(t) * g(t, \sigma) \\ Y(t, \sigma) &= y(t) * g(t, \sigma) \end{aligned} \quad (2)$$

เมื่อ

σ คือ ค่าสเกล-สเปซ หรือค่าเบี่ยงเบนมาตรฐาน

t คือ ตัวแปรอิสระ

$*$ คือ การประสาน

$x(t), y(t)$ คือ ตำแหน่งพิกัดทางแกน x และ y ที่ t ใดๆ

การหาค่าความโค้งของจุดใดๆ บนสายเส้นที่ผ่านการประสานแล้วสามารถคำนวณได้ดังนี้

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

$$\begin{aligned} \dot{X} &= X(t+1) - X(t-1), & \ddot{X} &= X(t+1) + X(t-1) - 2X(t) \\ \dot{Y} &= Y(t+1) - Y(t-1), & \ddot{Y} &= Y(t+1) + Y(t-1) - 2Y(t) \end{aligned} \quad (4)$$

เมื่อหาค่าความโค้งของทุกจุดบนสายเส้นแล้วจะพิจารณาจุดที่มีค่าความโค้งสูงกว่าค่าตัดระดับ (threshold) และสูงสุดเป็นจุดมุม โดยเส้นขอบวงปิดที่เป็นตัวโลจิกเกตจะต้องมีจุดมุม 2-4 จุด ดังรูปที่ 7

การวิเคราะห์ลักษณะลายเส้น

การวิเคราะห์ลักษณะลายเส้นที่อยู่ระหว่างจุดมุม 2 จุดของเส้นขอบวงปิดว่าเป็นเส้นตรงหรือเส้นโค้ง สามารถทำได้โดยการทำโค้งกระชับด้วยวิธีกำลังสองน้อยสุดที่นำเสนอในบทความ[3] ซึ่งเป็นวิธีการหาสมการเส้นตรงที่ดีที่สุดสำหรับกลุ่มของจุด โดยมีความสัมพันธ์ของสมการดังนี้

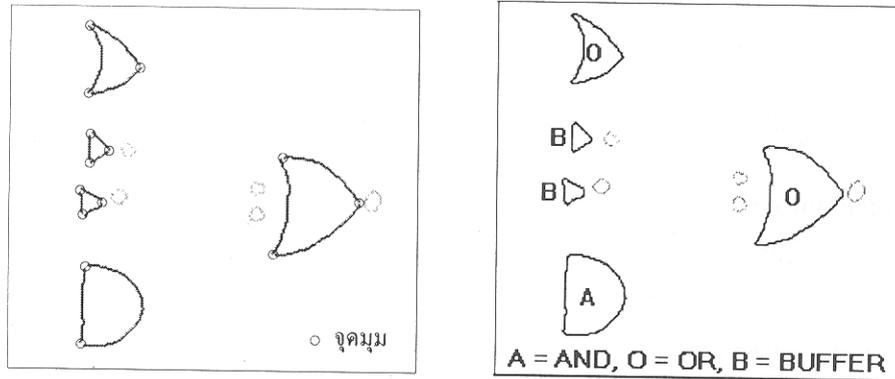
$$R_{av} = \frac{\sum y^2 - 2m\sum xy - 2b\sum y + m^2\sum x^2 + 2mb\sum x + nb^2}{n} \quad (5)$$

เมื่อ

$$m = \frac{n\sum xy - \sum x\sum y}{n\sum x^2 - (\sum x)^2}, \quad b = \frac{\sum x^2\sum y - \sum x\sum xy}{n\sum x^2 - (\sum x)^2} \quad (6)$$

$$\begin{aligned} \sum x &= x_1 + x_2 + \dots + x_n, & \sum y &= y_1 + y_2 + \dots + y_n \\ \sum x^2 &= x_1^2 + x_2^2 + \dots + x_n^2, & \sum y^2 &= y_1^2 + y_2^2 + \dots + y_n^2 \\ \sum xy &= x_1y_1 + x_2y_2 + \dots + x_ny_n \end{aligned} \quad (7)$$

ผลของค่า R_{av} สามารถใช้บ่งชี้ความเป็นเส้นตรงและเส้นโค้งได้คือ ลายเส้นตรงจะมีค่า $R_{av} \leq 1$ และลายเส้นโค้งจะมีค่า $R_{av} > 1$ เมื่อวิเคราะห์ลายเส้นที่เป็นองค์ประกอบของเส้นขอบวงปิดจนครบจะสามารถจำแนกโครงสร้างของเส้นขอบวงปิดได้เป็น 4 ชนิดคือ โครงสร้าง AND OR BUFFER และ BLOCK (รูปที่ 7)



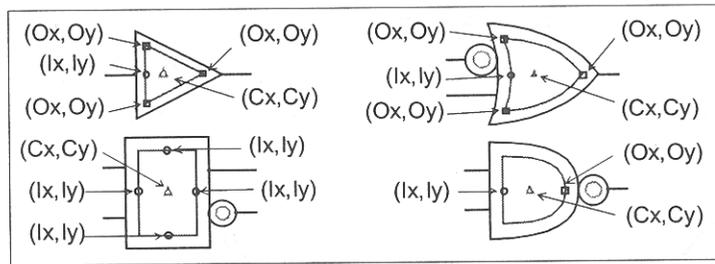
รูปที่ 7 ผลการหาจุดมุม และการจำแนกโครงสร้างของเส้นขอบวงปิด

ขั้นตอนที่สอง (Second Step)

ขั้นตอนที่สองเป็นการวิเคราะห์หาอินพุต-เอาต์พุตของโลจิกเกต และการต่อล่ายวงจรระหว่างโลจิกเกตกับโลจิกเกต และโลจิกเกตกับจุดต่อร่วม โดยจะวิเคราะห์ที่ลายเส้นโครงร่างของภาพที่หาได้โดยใช้วิธีการทำลายเส้นบางดังที่กล่าวมาแล้ว พร้อมทั้งวาดภาพวงจรโลจิกเกตขึ้นมาใหม่ ปรับทิศทาง และสเกลลิ่งโลจิกเกตให้เหมาะสม โดยมีรายละเอียดดังนี้

การหาอินพุต-เอาต์พุตของโลจิกเกต

โดยทั่วไปโลจิกเกตจะมีเพียงหนึ่งเอาต์พุต และมีตำแหน่งที่แน่นอน (ยกเว้น BLOCK ที่ไม่สามารถจำแนกอินพุต-เอาต์พุตได้) ต่างจากอินพุตที่มีอย่างน้อย 1 อินพุต ดังนั้นการตรวจหาเอาต์พุตจึงทำได้ง่ายกว่า และถ้าการตรวจหาเอาต์พุตไม่พบ แสดงว่าไม่ใช่โลจิกเกตจะไม่มีกรตรวจหาอินพุตอีก การหาอินพุต-เอาต์พุตของโลจิกเกตจะใช้วิธีการเดียวกัน ซึ่งนำเสนอในบทความ[10] โดยจะต้องกำหนดจุดเริ่มต้น (Ox, Oy) สำหรับการหาเอาต์พุต และ (Ix, Iy) สำหรับการหาอินพุต ซึ่งเป็นจุดที่อยู่บนเส้นขอบวงปิดที่ใกล้ตำแหน่งอินพุต-เอาต์พุตมากที่สุดดังรูปที่ 8

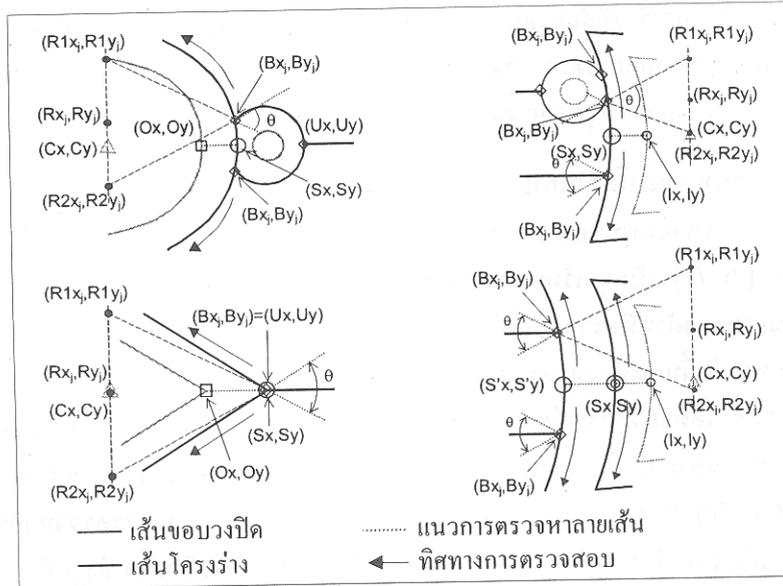


รูปที่ 8 ตำแหน่งจุดต่าง ๆ สำหรับการหาอินพุต-เอาต์พุตของโลจิกเกต

ก) การหาเอาต์พุตของโลจิกเกต

ในการหาเอาต์พุตจะใช้จุดเริ่มต้น (Ox, Oy) เพื่อหาเส้นโครงร่างของตัวโลจิกเกต โดยตรวจหาเป็นเส้นตรงจากจุดศูนย์กลาง (Cx, Cy) ผ่านจุดเริ่มต้น (Ox, Oy) ไปจนกระทั่งพบจุดบนลายเส้นโครงร่าง (Sx, Sy) ดังรูปที่ 9 จากนั้นจะตรวจสอบเส้นโครงร่างทั้งสองด้านของจุด (Sx, Sy) เพื่อหาจุดแยกเอาต์พุต โดยมีระยะการตรวจสอบดังนี้ โลจิกเกตโครงสร้าง AND มีระยะการตรวจสอบรวม $1/2$ ของความยาวลายเส้นโค้ง ส่วน

โลจิกเกตโครงสร้าง OR และ BUFFER มีระยะการตรวจสอบรวม $1/9$ ของขนาดโลจิกเกต ซึ่งการกำหนดระยะดังกล่าวเพื่อป้องกันความผิดพลาด เนื่องจากการเขียนเอาท์พุทไม่ตรงตำแหน่งหรือการหาจุดเริ่มต้น (Ox, Oy) คลาดเคลื่อน เมื่อการตรวจสอบสายพบจุดแยก (Bx_j, By_j) ตำแหน่งของจุดแยกจะถูกเก็บไว้ การหาจุดต่อไปจะต้องเลือกจุดที่มีระยะห่างจากจุดเปรียบเทียบ (Rx_j, Ry_j) น้อยที่สุด ทั้งนี้เพื่อป้องกันการตรวจหาสายผิดพลาด โดยจุดเปรียบเทียบสามารถคำนวณได้จากสมการดังนี้



รูปที่ 9 การหาเอาท์พุท-อินพุทของโลจิกเกต

$$\begin{aligned} Rx_j &= (Bx_j - Sx) + Cx \\ Ry_j &= (By_j - Sy) + Cy \end{aligned} \quad (8)$$

เมื่อ $j =$ จุดแยกจุดที่ $1, 2, 3, \dots, n$

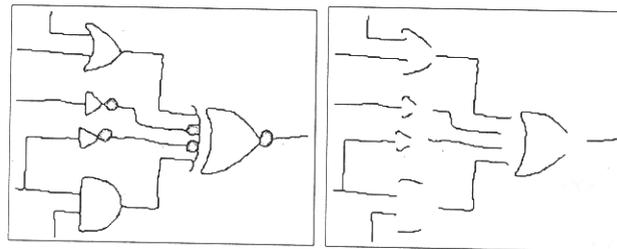
เมื่อได้ตำแหน่งและจำนวนจุดแยก จากนั้นตรวจหาวงกลม NOT บริเวณจุดแยกแต่ละจุด โดยตรวจหาเป็นเส้นตรงสองทิศทางออกจากจุดแยกมีมุมต่างกัน θ องศา ดังรูปที่ 9 ซึ่งจุดอ้างอิง $(R1x_j, R1y_j)$ และ $(R2x_j, R2y_j)$ สามารถคำนวณได้ดังนี้

$$\begin{aligned} R1x_j &= Rx_j + \frac{1}{2}(Sy - Cy), & R1y_j &= Ry_j + \frac{1}{2}(Sx - Cx) \\ R2x_j &= Rx_j - \frac{1}{2}(Sy - Cy), & R2y_j &= Ry_j - \frac{1}{2}(Sx - Cx) \end{aligned} \quad (9)$$

ในการตรวจหาวงกลม NOT ถ้าพบจะต้องตรวจสอบขนาดวงกลม NOT ว่าเหมาะสมกับตัวโลจิกเกตหรือไม่ โดยโลจิกเกตโครงสร้าง AND - OR และ BLOCK ควรมียวงกลม NOT เล็กกว่า 1/3 ของขนาดโลจิกเกต ส่วนโครงสร้าง BUFFER ควรมียวงกลม NOT เล็กกว่าขนาดของตัวโลจิกเกต ถ้าตรงตามเงื่อนไขจะตรวจสอบจุดแยกเอาท์พุทบนลายเส้นโครงร่างที่ล้อมรอบวงกลม NOT จนสุดลายเส้น และบรรจบกับจุดแยกใดจุดแยกนั้นจะถูกกลบออก แต่ถ้าการตรวจหาวงกลม NOT ไม่พบหรือพบวงกลม NOT ที่มีขนาดไม่เหมาะสมจะเก็บจุดแยกนั้นเป็นเอาท์พุท

ข) การหาอินพุทของโลจิกเกต

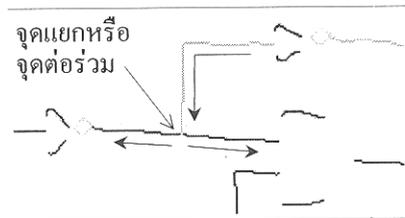
การหาอินพุทของโลจิกเกต สามารถใช้วิธีการเดียวกันกับการหาเอาท์พุทเพียงแต่ใช้จุด (Ix, Iy) เป็นจุดเริ่มต้น และมีระยะของการตรวจหาจุดแยกบนเส้นโครงร่างเท่ากับความยาวของด้านที่ตรวจสอบดังรูปที่ 9 ในกรณีโลจิกเกตเป็น Exclusive การหาอินพุทจะไม่พบ ดังนั้น จะต้องหาลายเส้น Exclusive และตรวจหาอินพุทบนลายเส้นดังกล่าวซึ่งสามารถทำได้ด้วยวิธีการเดียวกันกับการหาเอาท์พุท เพียงแต่ใช้จุด (Sx, Sy) เป็นจุดเริ่มต้น และกำหนดระยะการตรวจหาลายเส้น Exclusive เท่ากับระยะทางระหว่างจุดศูนย์กลาง (Cx, Cy) ของโลจิกเกตกับจุด (Sx, Sy) ซึ่งเมื่อผ่านกระบวนการตรวจหาอินพุท-เอาท์พุทแล้ว ลายเส้นของอินพุท-เอาท์พุทจะถูกตัดออกจากตัวโลจิกเกต ดังรูปที่ 10 เพื่อลดความซับซ้อนในการวิเคราะห์การต่อลายวงจรของโลจิกเกตในขั้นตอนต่อไป



รูปที่ 10 ภาพก่อนและหลังการหาอินพุท-เอาท์พุทของโลจิกเกต

การวิเคราะห์การต่อลายวงจร

การวิเคราะห์การต่อลายวงจร จะเริ่มที่ตำแหน่งอินพุท-เอาต์พุทของโลจิกเกทแต่ละตัว และตรวจสอบตามลายเส้นจนกว่าจะพบจุดแยกหรือสุดปลายลายเส้น ซึ่งการตรวจสอบถ้าไม่พบจุดแยก จุดปลายสุดของลายเส้นจะถูกนำไปเปรียบเทียบกับอินพุท-เอาต์พุทของโลจิกเกททุกตัวเพื่อตรวจสอบการเชื่อมต่อกันระหว่างโลจิกเกท โดยอินพุท-เอาต์พุทของโลจิกเกทที่ถูกเชื่อมต่อจะไม่นำมาตรวจสอบอีก แต่ถ้าการเปรียบเทียบไม่ตรง แสดงว่าอินพุท-เอาต์พุทของโลจิกเกทที่ตรวจสอบไม่มีการต่อ (ปล่อยลอย) กรณีที่การตรวจสอบพบจุดแยก (จุดต่อร่วม) จะเก็บตำแหน่งจุดแยกนั้น และเริ่มตรวจสอบลายเส้นทุกลายเส้นที่เป็นสาขาของจุดแยกด้วยวิธีการเดียวกัน ดังรูปที่ 11 เมื่อวิเคราะห์โลจิกเกทครบทุกตัวจะทำให้เรารู้การต่อลายวงจรทั้งหมด



รูปที่ 11 การตรวจสอบลายเส้นเมื่อพบจุดแยก

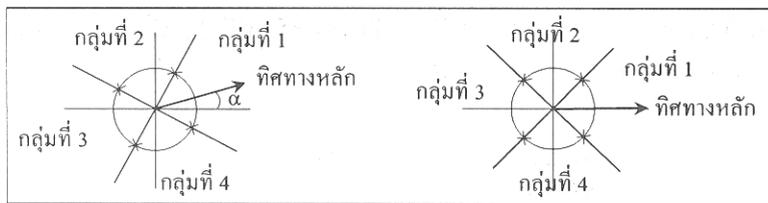
การปรับทิศทาง และการสเกลลิ่ง

โดยทั่วไปภาพโลจิกเกทที่เขียนด้วยมือจะไม่สมมาตร และอาจมีทิศทางแตกต่างกัน ดังนั้นจำเป็นต้องหาทิศทางของโลจิกเกทแต่ละตัว เพื่อจัดกลุ่มและหาทิศทางหลักของโลจิกเกท (α) ซึ่งจะนำไปใช้ในการปรับทิศทางให้เหมาะสมเพื่อการวาดภาพใหม่ การคำนวณทิศทางของโลจิกเกทจะใช้ตำแหน่งของจุดสองจุดคือ จุดศูนย์กลาง (C_x, C_y) กับจุดเอาต์พุท (U_x, U_y) ของโลจิกเกท ยกเว้นกรณี BLOCK ที่ไม่สามารถจำแนกอินพุท-เอาต์พุทได้ จะใช้จุดกึ่งกลางลายเส้นด้านที่ยาวด้านใดด้านหนึ่งแทน และคำนวณโดยใช้สมการหาทิศทางดังนี้

$$\text{ทิศทางของโลจิกเกท} = \tan^{-1} \left(\frac{U_y - C_y}{U_x - C_x} \right) \quad (10)$$

เมื่อคำนวณทิศทางของโลจิกเกทครบทุกตัวจะจัดโลจิกเกท ออกเป็นกลุ่มๆ โดยโลจิกเกทที่มีทิศทางต่างกันน้อยกว่า 45 องศาจะอยู่กลุ่มเดียวกัน จากนั้น จะหาทิศ

ทางหลัก (α) จากการคำนวณทิศทางเฉลี่ยของกลุ่มที่มีจำนวนโพลิกเกตมากที่สุด ซึ่งทิศทางหลักที่ได้นี้จะนำมาใช้เป็นทิศทางอ้างอิงเพื่อจัดกลุ่มของโพลิกเกตใหม่ โดยจัดเป็น 4 กลุ่มซึ่งแต่ละกลุ่มมีทิศทางต่างกัน 90 องศา ซึ่งกลุ่มแรกมีทิศทางต่างจากทิศทางหลักอยู่ในช่วง ± 45 ส่วนกลุ่มที่สอง สาม และสี่ มีทิศทางดังรูปที่ 12 จากนั้นจะปรับหรือหมุนตำแหน่งของจุดต่างๆ เช่น ตำแหน่งโพลิกเกต อินพุต-เอาต์พุต จุดตัดหรือจุดเชื่อม และอื่นๆ โดยมีองศาในการหมุนเท่ากับทิศทางหลัก และสามารถคำนวณตำแหน่งใหม่ได้จากสมการที่ (11)



รูปที่ 12 การจัดกลุ่มก่อน และหลังการปรับทิศทาง

$$\begin{aligned}
 x' &= (x - ICx) \cos(\alpha) - (y - ICy) \sin(\alpha) + ICx \\
 y' &= (y - ICy) \cos(\alpha) - (x - ICx) \sin(\alpha) + ICy
 \end{aligned}
 \tag{11}$$

เมื่อ (ICx, ICy) คือ จุดศูนย์กลางของภาพ

เนื่องจากตัวโพลิกเกต และภาพวงจรที่รับเข้ามามีขนาดไม่แน่นอน แต่การวาดภาพวงจรใหม่จะมีขนาดของตัวโพลิกเกตคงที่ ดังนั้น จะต้องหาอัตราการสเกลลิง (scaling) ภาพ โดยจะหาขนาดเฉลี่ยของโพลิกเกตเฉพาะโครงสร้าง AND และ OR เทียบกับขนาดของ AND หรือ OR gate มาตรฐาน (AND และ OR gate มาตรฐานมีขนาดเท่ากัน) ดังสมการที่ (12)

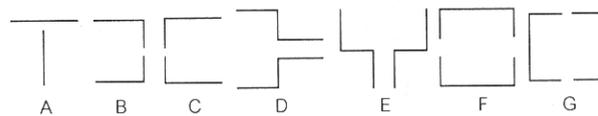
$$\text{Scale} = \frac{\text{ขนาดเฉลี่ยของ AND และ OR gate ในภาพ}}{\text{ขนาดของโพลิกเกตมาตรฐาน}}
 \tag{12}$$

ตำแหน่งใหม่ของจุดต่างๆ จำนวนดังนี้

$$x'' = \frac{x'}{scale}, \quad y'' = \frac{y'}{scale} \quad (13)$$

การวาดภาพ

เมื่อปรับตำแหน่งต่างๆ แล้ว ทิศทางหลักของลอจิกเกตจะเป็น 0 องศา ดังนั้น ลอจิกเกตที่อยู่ในกลุ่มต่างๆ จะมีทิศทางดังนี้คือ ลอจิกเกตที่อยู่ในกลุ่มที่ 1 จะมีทิศทาง 0 องศา ลอจิกเกตที่อยู่ในกลุ่มที่ 2 3 และ 4 จะมีทิศทาง 90 180 และ 270 องศา ตามลำดับ ส่วนการวาดภาพลอจิกเกตใหม่จะวาดในตำแหน่ง และทิศทางตามข้อมูลที่ได้โดยลำดับ และตำแหน่งของอินพุทใหม่จะต้องสัมพันธ์กับลำดับ และตำแหน่งอินพุทที่ได้จากการวิเคราะห์ เพื่อป้องกันการต่อสายวงจรผิดพลาด



รูปที่ 13 รูปแบบสาย

การต่อสายวงจรโดยทั่วไปจะลากเส้นในแนวแกน X และ แกน Y เท่านั้น โดยมีรูปแบบของสายหลายรูปแบบเพื่อความเหมาะสมดังรูปที่ 13 ในการลากเส้นทุกรูปแบบจะมีการตรวจสอบทุกๆ จุดในแนวเส้นที่ลากผ่าน เพื่อป้องกันการลากทับสายหรือตัวลอจิกเกต ด้วยเหตุนี้การต่อสายวงจรโดยใช้รูปแบบสายเพียงรูปแบบเดียวอาจต่อสายวงจรไม่สำเร็จ จึงจำเป็นต้องมีรูปแบบสายสำรองหลายรูปแบบ และทดลองต่อสายวงจรที่ละแบบจนกว่าจะต่อสายวงจรสำเร็จ ส่วนการเลือกใช้รูปแบบสายที่เหมาะสมเพื่อต่อสายวงจรจะต้องพิจารณาทิศทางของลอจิกเกตที่เชื่อมต่อกันดังนี้

ตารางที่ 1

การเลือกรูปแบบต่อลายเส้น

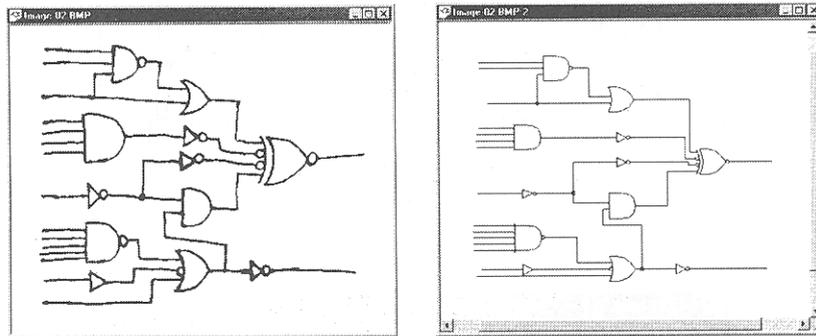
โลจิกเกททิศทาง	โลจิกเกททิศทาง	ใช้รูปแบบลายเส้น
0	0	A,D,B,C,E,F,G
0	90	A,B,C,E,D,F,G
0	180	A,B,C,E,D,F,G
0	270	A,B,C,E,D,F,G
90	90	A,E,C,B,D,G,F
90	180	A,B,C,E,D,F,G
90	270	A,C,B,D,E,G,F
180	180	A,D,B,C,E,F,G
180	270	A,B,C,E,D,F,G
270	270	A,E,C,B,D,G,F
โลจิกเกททิศทางใดๆ	จุดแยก	A,B,C,D,E,F,G
จุดแยก	จุดแยก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

ผลการทดลอง

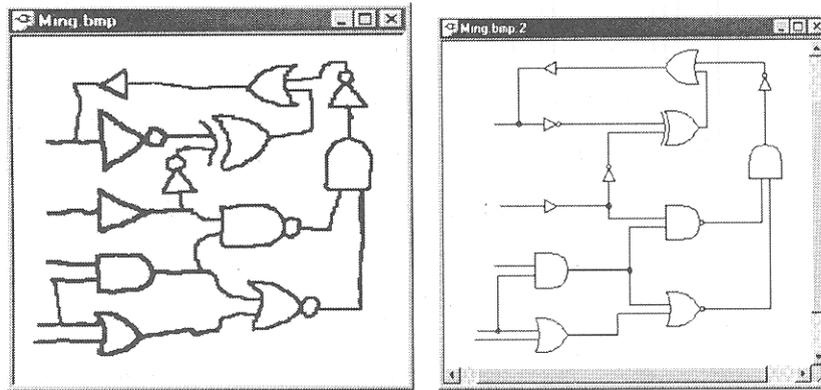
ในการทดลองโปรแกรมเขียนขึ้นด้วย Visual C++ และใช้ภาพวงจรถูกโลจิกเกทที่เขียนด้วยมือ ซึ่งมีขนาดของภาพ ทิศทางของโลจิกเกท และลายวงจรถ่างๆ กัน โดยใช้ค่าสเกล-สเปซกับค่าตัดระดับในการหาจุดแบ่งแยกลายเส้นเท่ากับ 3 และ 0.035 ตามลำดับ และกำหนดขนาดโลจิกเกทมาตรฐานเท่ากับ 120 จุด ซึ่งมีผลการทดลองดังรูปที่ 14 ถึง 16

การทดลองได้ทดลองกับภาพวงจรถูกโลจิกเกทจำนวน 124 ภาพ มีจำนวนโลจิกเกท 642 ตัว การประมวลผลในขั้นตอนแรกสามารถวิเคราะห์ และจำแนกโครงสร้าง AND, OR, BUFFER และ BLOCK ได้ถูกต้องเฉลี่ย 92.4% ผิดพลาด 7.6% เนื่องจากการเขียนโลจิกเกทมีขนาดเล็ก และมีลักษณะเส้นตรงเส้นโค้งไม่เด่นชัด ส่วนการประมวลผลในขั้นตอนที่สองสามารถวิเคราะห์หาอินพุท-เอาต์พุท การต่อลายวงจรถูกและ

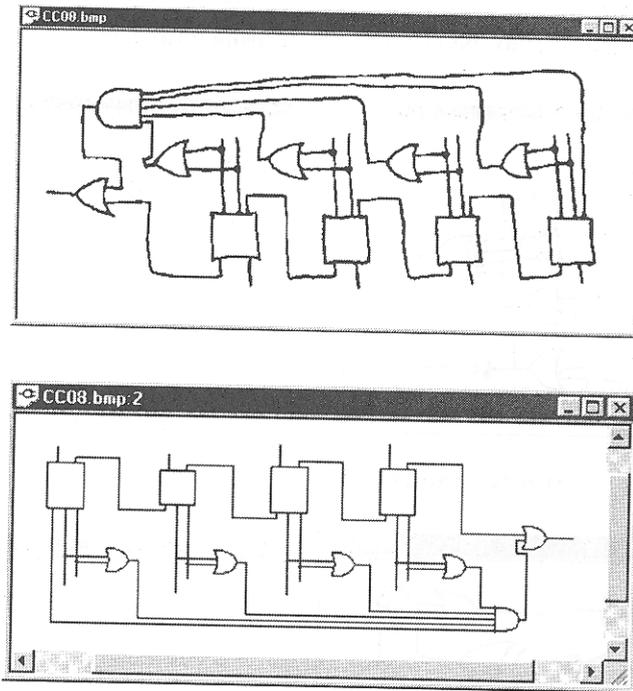
สร้างภาพวงจรขึ้นใหม่ได้ถูกต้อง 89.5% ผิดพลาด 10.5% เนื่องจากการเขียนอินพุตของ
ลอจิกเกตใกล้จุดมุม มีวงกลม NOT ติดกัน และการต่อสายเส้นไม่สำเร็จที่จุดแยกสี่สาขา



รูปที่ 14 ภาพอินพุตและภาพวงจรที่เขียนขึ้นใหม่



รูปที่ 15 ภาพอินพุตและภาพวงจรที่เขียนขึ้นใหม่



รูปที่ 16 ภาพอินพุตและภาพวงจรที่เขียนขึ้นใหม่

บทสรุป

เราได้นำเสนอการรู้จำวงจรลอจิกเกตที่เขียนด้วยมือ โดยใช้หลักการแบ่งแยกสายเส้น โดยพัฒนาปรับปรุงวิธีการ และอัลกอริทึมในขั้นตอนต่างๆ เพื่อเพิ่มขีดความสามารถ ความถูกต้อง และลดความซับซ้อนในการประมวลผล ซึ่งสามารถรู้จำชนิดของลอจิกเกตได้ดังนี้คือ AND gate, NAND gate, OR gate, NOR gate, XOR gate, XNOR gate, NOT gate, BUFFER และ BLOCK รวมทั้งสามารถตรวจสอบวงกลม NOT ที่อินพุตได้ด้วย และสามารถวาดวงจรลอจิกเกตขึ้นมาใหม่ได้ถูกต้อง 89.5%

เอกสารอ้างอิง

- [1] เกษตร์ ศิริสันติสัมฤทธิ์ และ อรุณ กำเหนิดนนท์, การจดจำวงจรลอจิกเกตที่เขียนด้วยมือโดยใช้หลักการแบ่งแยกสายเส้น, การประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 16, พฤศจิกายน 2536

- [2] อโณทัย ปฐมมานิศ สมคิด อุนการุณวงษ์ และเกษตร์ ศิริสันติสัมฤทธิ์, การจดจำวงจรถลจิกเกทที่เขียนด้วยมือ โดยใช้หลักการแบ่งแยกสายเส้น, การประชุมวิชาการวิศวกรรมไฟฟ้า ครั้งที่ 17, ธันวาคม 2537
- [3] เตชา พัฒนประสิทธิ์ชัย สมคิด อุนการุณวงษ์ ชีรชาติ แสนชัย และเกษตร์ ศิริสันติสัมฤทธิ์, การวิเคราะห์โครงสร้างของวงปิดสำหรับการจดจำวงจรถลจิกเกทที่เขียนด้วยมือ, การประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 23, พฤศจิกายน 2543
- [4] Rafael C. Gonzalez and Richard E. Woods, Digital Image Processing, Addison-Wesley Publishing Company, 1992.
- [5] Zhang T.Y. and Suen C.Y., A Fast Parallel Algorithm for Thinning Digital Pattern, Comm. ACM, Vol.27, No.3; pp.236-239; 1984.
- [6] Nirwan Ansari and Edward J. Depl, On Detecting Dominant Points, Pattern Recognition, Vol.24, No.5, pp.441-451, 1991.
- [7] S.C. Pei and C.N. Lin, The Detection of Dominant Points on Digital Curves by Scale-Space Filtering, Pattern Recognition, Vol.25, pp.130, 1992.
- [8] รศ.ดร.ชม กิมปาน, การรู้จำรูปแบบอักษรพิมพ์ภาษาไทยของคอมพิวเตอร์
- [9] ปราโมทย์ เตชะอำไพ “ระเบียบวิธีเชิงตัวเลขในงานวิศวกรรม” สำนักพิมพ์แห่งจุฬาลงกรณ์มหาวิทยาลัย
- [10] เกษตร์ ศิริสันติสัมฤทธิ์ และสมคิด อุนการุณวงษ์, กระบวนการหลังของการรู้จำวงจรถลจิกเกทที่เขียนด้วยมือ, การประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 24, พฤศจิกายน 2544