อัลกอริทึมการบวกและการคูณสำหรับระบบจำนวนฐานคู่ทั่วไป

นายเกรียงยุทธ หวังจิตมั่น

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2549

ISBN 974-14-3012-4

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

ADDITION AND MULTIPLICATION ALGORITHM FOR GENERIC DOUBLE-BASE NUMBER SYSTEM

Mr. Kriangyut Wangjitman

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering Program in Computer Engineering
Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2006
ISBN 974-14-3012-4
Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	อัลกอริทึมการบวกและการคูณสำหรับระบบจำนวนฐานคู่ทั่วไป
โดย	นายเกรียงยุทธ หวังจิตมั่น
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษา	อาจารย์ ดร.อรรถสิทธิ์ สุรฤกษ์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

.... คณบดีคณะวิศวกรรมศาสตร์

(ศาสตราจารย์ ดร.ดิเรก ลาวัณย์ศิริ)

คณะกรรมการสอบวิทยานิพนธ์

ประภาส จงสถิตย์วัฒนา
.... ประธานกรรมการ

(รองศาสตราจารย์ ดร.ประภาส จงสถิตย์วัฒนา)

.... อาจารย์ที่ปรึกษา

(อาจารย์ ดร.อรรถสิทธิ์ สุรฤกษ์)

.........เศรษฐา ปานงาม......... กรรมการ

(อาจารย์ ดร.เศรษฐา ปานงาม)

.... กรรมการ

(ผู้ช่วยศาสตราจารย์ ดร. อานนท์ รุ่งสว่าง)

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

เกรียงยุทธ หวังจิตมั่น : อัลกอริทึมการบวกและการคูณสำหรับระบบจำนวนฐานคู่ทั่วไป. (ADDITION AND MULTIPLICATION ALGORITHM FOR GENERIC DOUBLE-BASE NUMBER SYSTEM) อ. ที่ปรึกษา : อ.ดร.อรรถสิทธิ์ สุรฤกษ์, 55 หน้า. ISBN 974-14-3012-4.

ระบบจำนวนนั้นมีผลกระทบต่อการคำนวนทางเลขคณิตในระบบคอมพิวเตอร์ ดังนั้นนักวิจัยจึงเสนอระบบจำนวนชนิดใหม่ออกมาเป็นจำนวนมาก และระบบจำนวนหนึ่งที่มีความน่าสนใจ คือ ระบบจำนวนฐานคู่ซึ่งเป็นระบบจำนวนที่ใช้ในการแสดงจำนวนเต็มบวก โดยใช้ฐานเป็นจำนวนเต็มบวกสองจำนวน ปกติแล้วเป็น 2 และ 3 คุณสมบัติเด่นของระบบจำนวนฐานคู่ คือ ระบบจำนวนนี้มีความซ้ำซ้อน และการกระจายตัวที่สูงมาก มีงานวิจัยหลายงานที่เสนอเกี่ยวกับปฏิบัติการพื้นฐานทางเลขคณิตของระบบจำนวนนี้ อันได้แก่ การบวก และการคูณ แต่กระบวนการที่ใช้ในงานวิจัยเหล่านั้นไม่มีลำดับการทำงานที่แน่นอน วิทยานิพนธ์ฉบับนี้จึงนำเสนออัลกอริทึมในการบวก และการคูณซึ่งมีการทำงานเชิงกำหนด โดยที่อัลกอริทึมนี้สามารถใช้งานได้กับระบบจำนวนฐานคู่ทั่วไปด้วย การกระบวนการบวกนี้ใช้เวลาเป็นเชิงเส้นขึ้นอยู่กับขนาดของรูปแบบแสดงค่าที่นำมาบวก และยังสามารถทำงานแบบสายท่อได้ ในส่วนของกระบวนการคูณนั้นจะสร้างขึ้นจากกระบวนการบวกที่นำเสนอ โดยมีความซับซ้อนเป็นเชิงเส้นขึ้นกับจำนวนตำแหน่งที่มีค่าของตัวคูณ นอกจากนี้ยังเสนอวิธีการในการพิสูจน์ความถูกต้องของอัลกอริทึมทั้งในทางทฤษฎี และจำลองการทำงานให้อยู่ในรูปของวงจรดิจิตอล ที่ทำงานแบบสมวาร เพื่อแสดงให้เห็นว่าสามารถนำไปใช้งานในระบบคอมพิวเตอร์ได้จริง

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาควิชา ....วิศวกรรมคอมพิวเตอร์ .... ลายมือชื่อนิสิต ...เกรียงยุทธ หวังจิตมั่น...........
สาขาวิชา ....วิศวกรรมคอมพิวเตอร์ .... ลายมือชื่ออาจารย์ที่ปรึกษา ....
ปีการศึกษา.....2549 ....

497 02327 21 : MAJOR COMPUTER ENGINEERING

KEY WORD: DOUBLE-BASE NUMBER SYSTEM / ADDITION / MULTIPLICATION / ALGORITHM

KRIANGYUT WANGJITMAN : ADDITION AND MULTIPLICATION ALGORITHM FOR GENERIC DOUBLE-BASE NUMBER SYTEM. THESIS ADVISOR : ATHASIT SURARERKS, Ph.D., 55 pp. ISBN: 974-14-3012-4.

Number system has a great effect to computer arithmetic calculation thus a lot of researchers have proposed many new number systems. One of them is Double-Base Number System. It represents positive integer by using two bases; usually be *two* and *three*. The advantages of this system are its high redundancy degree and sparseness. Many papers are concentrated on basic arithmetic operations that are addition and multiplication. However, earlier algorithms are non-deterministic. This thesis proposes deterministic addition and multiplication algorithms which can also manipulate the generic double-base number system. The addition time is linearly dependent on the amount of rows. In addition, this addition algorithm is pipeline-able. The multiplication is based on our proposed addition algorithm. The complexity of this operation depends on the number of active cells in the multiplicand. Furthermore, all algorithms are proved theoretically and practically by simulation on synchronized digital circuits.

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

Department ....Computer Engineering...	Student's signature ...Kriangyut Wangjitman...
Field of study ....Computer Engineering...	Advisor's signature ....
Academic year 2006	

กิตติกรรมประกาศ

วิทยานิพนธ์นี้สำเร็จเรียบร้อยได้ก็เป็นเพราะแรงบันดาลใจจาก อ.ดร. อรรถสิทธิ์ สุรฤกษ์ ที่เป็นผู้ชี้แนะแนวทาง อีกทั้งยังจุดประกายให้เห็นถึงความน่าสนใจของระบบจำนวนที่มีความพิเศษนี้ นอกจากนี้ อยากขอขอบพระคุณเป็นพิเศษ สำหรับคณะกรรมการสอบวิทยานิพนธ์ ได้แก่ รศ. ดร. ประภาส จงสถิตย์วัฒนา ผู้ซึ่งเป็นประธาน อ. ดร.เศรษฐา ปานงาม และ ผศ. ดร. อานนท์ รุ่งสว่าง ที่สละเวลาอันมีค่ามาชี้ให้เห็นถึงข้อบกพร่อง รวมทั้งข้อแนะนำที่น่าสนใจ

ขอขอบคุณสมาชิกของห้องปฏิบัติการ ELITE (Engineering Laboratory in Theoretical Enumerable System) ทุกคน รวมทั้งพี่ ๆ และเพื่อน ๆ ระดับบัณฑิตศึกษาที่สร้างบรรยากาศที่ดีในการทำงาน

ขอขอบคุณจุฬาลงกรณ์มหาวิทยาลัย และโรงเรียนเซนต์คาเบรียล ที่ได้ประสิทธิประสาทวิชาความรู้ อีกทั้งความรักในคณิตศาสตร์ โดยเฉพาะอย่างยิ่งมาสเตอร์สุรวุฒิ จันทอง และครูผู้สอนวิชาคณิตศาสตร์ทุกท่าน อีกทั้งมูลนิธิส่งเสริมโอลิมปิกวิชาการ และพัฒนามาตรฐานวิทยาศาสตร์ศึกษาในพระอุปถัมภ์สมเด็จพระเจ้าพี่นางเธอ เจ้าฟ้ากัลยาณิวัฒนา กรมหลวงนราธิวาสราชนครินทร์ (สอวน.) ที่ได้เปิดอีกด้านหนึ่งของคณิตศาสตร์สู่สายตาของผู้เขียน

สุดท้ายนี้ขอกราบขอบพระคุณ คุณพ่อ คุณแม่ และญาติ ๆ ที่ได้ผลักดันจนผู้เขียนประสบความสำเร็จ และมีความเป็นคนอย่างเช่นทุกวันนี้

ทั้งนี้งานวิจัยนี้ได้รับเงินทุกสนับสนุนจากโครงการจัดการศึกษาสาขาวิชาวิศวกรรมศาสตร์เพื่อเพิ่มศักยภาพทางด้านวิทยาศาสตร์ เทคโนโลยีและอุตสาหกรรม หมวดเงินอุดหนุนการศึกษา ประจำปีการศึกษา 2549

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญ

หน้า

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญภาพ

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 1

บทนำ

ความเป็นมาและความสำคัญของปัญหา

คอมพิวเตอร์นั้นถูกใช้งานอย่างแพร่หลายในการคำนวณที่ยาก และต้องการความรวดเร็ว จึงมีนักวิจัยจำนวนมากพยายามจะพัฒนาการคำนวณทางเลขคณิตให้มีประสิทธิภาพที่สูงมากยิ่งขึ้น ทั้งในเชิงความเร็ว และการประหยัดพลังงาน ระบบจำนวน (number system) เป็นปัจจัยที่มีความสำคัญมากปัจจัยหนึ่งต่อประสิทธิภาพของการคำนวณ เนื่องจากเป็นตัวกำหนดวิธีการในการทำปฏิบัติการทั้งหมด โดยระบบจำนวนพื้นฐานที่ใช้กันในคอมพิวเตอร์นั้น สำหรับจำนวนเต็ม จะใช้เป็นระบบจำนวนฐานสอง ซึ่งจะมีขนาดเล็ก แต่ความเร็วในการคำนวณต่ำ เนื่องจากเมื่อมีการบวก หรือลบ อาจจะเกิดการทด หรือ ยืม ซึ่งเวลาที่ใช้ในการทดนั้นไม่สามารถจำกัดให้อยู่ในขอบเขตใด ๆ ได้ แต่ขึ้นอยู่กับลักษณะของค่าที่นำมาทำปฏิบัติการ ดังนั้นจึงมีการเสนอระบบจำนวนชนิดใหม่ที่สามารถจำกัดเวลาที่ใช้ในการทดได้ ซึ่งระบบจำนวนนั้นคือ ระบบจำนวนแบบมีเครื่องหมาย (sign digit number system) ซึ่งเสนอโดยอเวเซียนิส (Avizienis) [1] ระบบจำนวนชนิดนี้จะให้แต่ละหลักมีค่าติดลบได้ ทำให้เกิดความซ้ำซ้อนขึ้น คือ ค่าทางเลขคณิตค่าหนึ่งอาจจะมีรูปแบบในระบบจำนวนชนิดนี้ได้มากกว่าหนึ่งรูปแบบ คุณสมบัติของระบบจำนวนนี้ คือ สามารถจำกัดสายการทดได้ นอกจากนี้ยังสามารถคำนวณแบบขนานได้ด้วย ภายหลังมีงานวิจัยอีกหลายงานที่นำแนวความคิดนี้ เช่น ระบบจำนวนเชิงซ้อนของคนุท (Knuth) ที่ใช้เลขลบมาทำให้เกิดความซ้ำซ้อนของตัวเลข นอกจากนี้ การพัฒนาระบบจำนวนยังมีแนวความคิดอื่นที่ไม่ยึดติดกับลักษณะของระบบจำนวนแบบเดิมที่จะมีฐานคงที่ เช่น ระบบจำนวนแบบส่วนตกค้าง (residue number system) เป็นการใช้เลขขนาดเล็กหลายจำนวน เพื่อที่แสดงค่าจำนวนหนึ่งจำนวน โดยที่จำนวนที่ถูกแสดงนั้นอาจจะมีค่ามากได้ ระบบจำนวนนี้จะไม่มีความซ้ำซ้อน แต่สามารถคำนวณได้เร็วเนื่องจากการคำนวณนั้นสามารถกระทำไปพร้อม ๆ กันสำหรับจำนวนขนาดเล็กทุกตัว และในปี 1997 มีการนำเสนอระบบจำนวนชนิดใหม่ คือ ระบบจำนวนฐานคู่ (double-base number system) ซึ่งเป็นระบบจำนวนที่มีฐานเป็นจำนวนสองจำนวนซึ่งมีค่าที่แตกต่างกัน โดยปกติจะใช้เป็นฐานสอง และสามประกอบกัน ระบบจำนวนชนิดนี้จะมีข้อดีคือ มีความซ้ำซ้อนที่สูง สืบเนื่องจากการที่เพิ่มฐานเป็นสองจำนวน ทำให้สามารถแสดงค่าหนึ่ง ๆ ได้หลายรูปแบบ นอกจากนี้ยังมีการกระจายตัวสูง ทำให้การทดเกิดขึ้นได้ยาก ส่งผลให้การบวกเป็นไปได้อย่างรวดเร็ว

แต่เนื่องจากระบบจำนวนนี้ยังเป็นระบบจำนวนที่ใหม่ ส่งผลให้การพัฒนาวิธีการในการทำปฏิบัติการพื้นฐานทางเลขคณิตนั้นยังไม่สมบูรณ์ มีเพียงงานวิจัย [2-9] ที่กล่าวถึงวิธีการ และการสร้างตัวปฏิบัติการทางเลขคณิต แต่ก็ได้กล่าวถึงเฉพาะการบวก และการคูณ และมีการให้ความ

สนใจกับระบบจำนวนฐานคู่ที่มีฐานเป็นสอง และ สามเท่านั้น ดังนั้นงานวิจัยชิ้นนี้จะทำการเสนอกระบวกการสร้างตัวปฏิบัติการบวกที่มีลำดับการทำงานที่แน่นอน หรือใช้อัลกอริทึมเชิงกำหนดสำหรับระบบจำนวนฐานคู่ ซึ่งใช้ฐานเป็นสอง และจำนวนใด ๆ ที่มีค่ามากกว่าสอง ซึ่งการบวกนี้จะใช้เวลาแปรผันตรงกับขนาดของตัวที่จะนำมาบวก

วัตถุประสงค์ของการวิจัย

งานวิจัยนี้จะออกแบบกระบวนการในการคำนวณเลขคณิตพื้นฐาน ได้แก่ การบวก และการคูณ ของระบบจำนวนฐานคู่ ซึ่งมีฐานเป็นสอง และจำนวนเต็มใด ๆ อีกหนึ่งจำนวน ที่เป็นกระบวนการเชิงกำหนด โดยให้ความซับซ้อนของกระบวนการบวกนี้ขึ้นกับจำนวนแถวของรูปแบบแสดงค่าของข้อมูลที่จะบวก ในส่วนของกระบวนการคูณจะทำการหาความเร็วจากความซับซ้อนของกระบวนการ และกระบวนการทั้งหมดจะสามารถนำไปพัฒนาเป็นวงจรดิจิทัลได้ง่าย

ขอบเขตของการวิจัย

1. ตัวปฏิบัติการบวก และคูณที่สร้างจะรับข้อมูลเป็นจำนวนเต็มบวกในรูปแบบพร้อมบวกสองจำนวน และผลลัพท์ที่ได้จะอยู่ในรูปแบบพร้อมบวก
2. ตัวปฏิบัติการทั้งหมดจะทำงานกับข้อมูลขาเข้า และผลลัพท์เป็นฐานเดียวกัน และจะต้องมีฐานหนึ่งที่มีค่าเป็น 2 เท่านั้น
3. ความเร็วของการบวกนั้นจะวัดจากความซับซ้อนของปัญหาเท่านั้น และความเร็วของการบวกนี้จะต้องไม่มากกว่า O(m) เมื่อ m เป็นขนาดของรูปแบบแสดงค่าข้อมูลขาเข้า

ขั้นตอนการวิจัย

1. ศึกษาสมบัติของระบบจำนวนฐานคู่ และงานวิจัยที่เกี่ยวข้อง
2. ออกแบบกระบวนการบวกของระบบจำนวนฐานคู่ที่มีฐานเป็น สอง และ สาม และพิสูจน์การทำงาน
3. หารูปแบบพร้อมบวกใหม่ สำหรับจำนวนฐานคู่ที่มีฐานเป็นสอง และจำนวนเต็มใด ๆ พร้อมทั้งหาวิธีการในการบวก
4. หาวิธีการในการเปลี่ยนรูปของจำนวนฐานคู่ให้อยู่ในรูปพร้อมบวก พร้อมทั้งสร้างกระบวนการบวกในรูปแบบเครื่องจักรของมัวร์
5. พิสูจน์การทำงานของกระบวนการบวกสำหรับฐานใด ๆ
6. ออกแบบวิธีการในการคูณ โดยให้มีพื้นฐานการทำงานจากการบวก พร้อมทั้งพิสูจน์การทำงาน และ พิจารณาความเร็ว
7. สรุปผลและเรียบเรียงวิทยานิพนธ์

ประโยชน์ที่คาดว่าจะได้รับ

1. เพิ่มระบบจำนวนที่สามารถใช้งานมากยิ่งขึ้น และมีความยืดหยุ่นเพิ่มขึ้น
2. ทำให้ระบบจำนวนฐานคู่มีการใช้งานที่แพร่หลายยิ่งขึ้น

ผลงานที่ตีพิมพ์จากวิทยานิพนธ์

ส่วนหนึ่งของวิทยานิพนธ์นี้ได้รับการตอบรับให้ตีพิมพ์เป็นบทความทางวิชาการในหัวข้อเรื่อง “Addition transducer for double base number system” โดย เกรียงยุทธ หวังจิตมั่น และ อรรถสิทธิ์ สุรฤกษ์ ในงานประชุมวิชาการ “IEEE International Symposium on Communications and Information Technologies 2006 (ISCIT 2006)” ณ. กรุงเทพมหานคร ในระหว่างวันที่ 18-20 ตุลาคม 2549

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

บทนี้จะกล่าวถึงทฤษฎีพื้นฐานที่จำเป็นในการทำความเข้าใจต่อเนื้อหาที่จะนำเสนอในบทถัด ๆ ไป เช่น ทฤษฎีเกี่ยวกับระบบจำนวน, ความซ้ำซ้อนของระบบจำนวน และ ระบบจำนวนที่ศึกษา คือระบบจำนวนแบบฐานคู่ นอกจากนี้ ยังเสนองานวิจัยต่าง ๆ ที่มีส่วนเกี่ยวข้อง ได้แก่งานวิจัยที่กล่าวถึงวิธีการคูณ และ การบวกนั่นเอง

ทฤษฎีที่เกี่ยวข้อง

1. ระบบจำนวน

ระบบจำนวน เป็นข้อตกลงวิธีในการแสดงจำนวน ระบบจำนวนนี้สามารถจำแนกเป็นประเภทดังนี้

1. ประเภทที่ใช้ฐาน (base) เช่น ระบบจำนวนฐานสอง ระบบจำนวนฐานคู่
2. ประเภทที่ไม่ใช้ฐาน เช่น ระบบจำนวนแบบเศษเหลือ (ไม่กล่าวถึงในที่นี้)

ระบบจำนวนแบบใช้ฐานนั้น ประกอบด้วย (β, D) โดยที่ β เป็นเซตของฐาน ซึ่งจะประกอบด้วยจำนวนชนิดใดก็ได้ และ D เป็นเซตของจำนวนใด ๆ ที่เป็นเซตจำกัด

กำหนดให้ X เป็นจำนวนใดๆ X สามารถแสดงได้ในระบบฐาน β ในรูปแบบ

$$X = (x_m x_{m-1} \cdots x_0 . x_{-1} x_{-2} \cdots)_\beta$$

ซึ่ง $x_j \in D$ โดยที่ $j \le m,\ m \in \mathrm{Z}$

โดยค่าเชิงตัวเลขของ X ฐาน β สามารถเขียนให้อยู่ในรูป

$$\|X\| = \sum_{j=m}^{-\infty} x_j \beta_j$$

และ $\beta_j \in \beta$ โดยที่ $j \le m,\ m \in \mathrm{Z}$

ซึ่งค่าเชิงตัวเลขทั้งหมดที่แสดงได้สามารถเขียนให้อยู่รูปของเซต $\mathrm{P}[\beta, D]$ ได้ดังนี้

$$\mathrm{P}_m^n[\beta, D] = \left\{ X = (x_m x_{m-1} \cdots x_{n+1} x_n)_\beta \,\middle|\, x_j \in D, \beta_j \in \beta, n \le j \le m \right\}$$

$$\mathrm{P}_m[\beta, D] = \left\{ X = (x_m x_{m-1} \cdots)_\beta \,\middle|\, x_j \in D, \beta_j \in \beta, j \le m \right\}$$

โดย $\mathrm{P}_m^n[\beta, D]$ และ $\mathrm{P}_m[\beta, D]$ เท่ากับ เซตจำกัดและเซตไม่จำกัด ตามลำดับ ในระบบเลขฐานจำนวนเต็มทั่วไปนิยมให้ $\beta_j = b^j,\ b, j \in Z$ และ $D = \{\ 0, 1, \cdots, |b| - 1\ \}$ ซึ่ง D จะถูกเรียกว่าเป็นชุดตัวเลขคาโนนิคอล (canonical digit Set) และเรียก b ว่าเลขฐาน

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

2. ระบบจำนวนแบบซ้ำซ้อน (Redundant Number System)

ระบบจำนวนแบบซ้ำซ้อน คือ ระบบจำนวนที่ค่าเชิงตัวเลข (numerical value) ของ X อาจจะมีรูปแบบในการแสดงค่าได้หลายรูปแบบ เช่น กำหนดให้ $\beta = \{b^j\}$, $b, j \in Z$ และ b เป็นเลขฐาน โดยที่ b เป็นจำนวนเต็มที่ $b \geq 2$ และกำหนดให้ D เป็นชุดตัวเลขซึ่งอธิบายได้ด้วย $\{d \in Z \mid \alpha_1 \leq d \leq \alpha_2\}$ โดย $-\alpha_1 + \alpha_2 \geq b+1$ ยกตัวอย่างเช่น ชุดตัวเลข $D = \{\bar{2}, \bar{1}, 0, 1, \ldots 4\}$ บนเลขฐาน $b = 5$ และค่าเชิงตัวเลข $X = 39$ จะมีรูปแบบดังนี้

$$
\begin{aligned}
(1 \quad 2 \quad 4)_5 &= 1\times 5^2 + 2\times 5^1 + 4\times 5^0 \\
\left(1 \quad 3 \quad \bar{1}\right)_5 &= 1\times 5^2 + 3\times 5^1 + (-1)\times 5^0 \\
\left(2 \quad \bar{2} \quad \bar{1}\right)_5 &= 2\times 5^2 + (-2)\times 5^1 + (-1)\times 5^0 \\
&= 39
\end{aligned}
$$

ในกรณีที่ α_1 หรือ α_2 ไม่เป็นไปตามเงื่อนไขดังกล่าว ระบบจำนวนก็ไม่มีคุณสมบัติของความซ้ำซ้อน เนื่องจากไม่สามารถแสดงค่าของตัวเลข X ได้มากกว่าหนึ่งรูปแบบ โดยสามารถเขียนสมการอธิบายคุณสมบัติของชุดตัวเลขได้โดยสมการด้านล่างนี้

$$V_{(\beta,D)}(i) = \left\{P \in P[\beta, D] \,\middle|\, \|P\| = i\right\}$$

โดยที่ $P[\beta, D] = \{P = (d_j d_{j-1} \cdots)_\beta \,|\, d_j \in D\}$, $\|P\| = \sum_{j=l}^{n} d_j \beta^j$ และ $j \in Z$

สมการดังกล่าว เป็นการแสดงถึงเซตของรูปแบบที่สามารถแสดงค่าทั้งหมดของค่า i สามารถกล่าวว่าระบบจำนวนเต็มบวกใด ๆ มีคุณสมบัติ

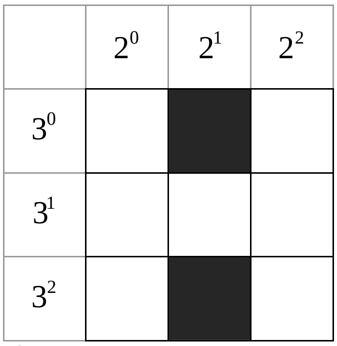
สมบูรณ์ (complete for radix β)	ก็ต่อเมื่อ $\forall j \in Z : \left|V_{(\beta,D)}(j)\right| \geq 1$
กึ่งสมบูรณ์ (semi-complete for radix β)	ก็ต่อเมื่อ $\forall j \in N : \left|V_{(\beta,D)}(j)\right| \geq 1$
ซ้ำซ้อน (redundant for radix β)	ก็ต่อเมื่อ $\exists j \in Z : \left|V_{(\beta,D)}(j)\right| > 1$
ไม่ซ้ำซ้อน (non-redundant for radix β)	ก็ต่อเมื่อ $\forall j \in Z : \left|V_{(\beta,D)}(j)\right| \leq 1$

3. ระบบจำนวนฐานคู่ (Double-Base Number System)

ระบบจำนวนฐานคู่คิดค้นโดยดิมิทรอฟ (Dimitrov) [2] ในปี 1997 เป็นมีระบบจำนวนที่ใช้ฐานสองจำนวนประกอบกัน โดยจำนวนที่นิยมใช้เป็นฐาน คือ สอง และ สาม ระบบจำนวนนี้เป็นระบบจำนวนที่สมบูรณ์และซ้ำซ้อน สำหรับจำนวนเต็ม X ใด ๆ สามารถเขียน X ในรูป

$$\|X\| = \sum_{i,j} d_{i,j} 2^i 3^j, \; d_{i,j} \in \{0,1\}$$

ระบบจำนวนฐานคู่นี้จะสามารถมองได้เป็นตารางสองมิติโดยแกน x และ y จะเป็นแกนของฐานสอง และ สามตามลำดับ ดังรูปที่ 2.1 โดยที่ช่องสีดำจะเป็นตำแหน่งที่ $d_{i,j}$ มีค่าเป็น 1



รูปที่ 2.1 ตารางสองมิติแสดงค่า 20

จากรูปเป็นการแสดงค่าของ $20 = 2^1 3^0 + 2^1 3^2$ และเนื่องจากเป็นระบบจำนวนที่ซ้ำซ้อน ดังนั้นจะมีรูปแบบการแสดงค่าในระบบจำนวนฐานคู่ (double-base number representation: DBNR) อื่นสำหรับ 20 เช่น

$$
\begin{aligned}
20 &= 1 \times 2^1 3^0 + 1 \times 2^1 3^2 \\
&= 1 \times 2^2 3^0 + 1 \times 2^4 3^0 \\
&= 1 \times 2^3 3^0 + 1 \times 2^2 3^1 \\
&= 1 \times 2^1 3^0 + 1 \times 2^1 3^1 + 1 \times 2^2 3^1
\end{aligned}
$$

และเนื่องจากความซ้ำซ้อนนี้ ทำให้จำนวนหนึ่งอาจจะมีรูปแบบการแสดงค่าที่มีจำนวนของเลขหนึ่งที่ไม่เท่ากัน และจะเรียกรูปแบบที่มีจำนวนเลขหนึ่งน้อยที่สุดว่า รูปแบบคาโนนิค (carnonic representation) รูปแบบคาโนนิคนี้มีข้อดีคือ จะใช้พลังงานในการเก็บข้อมูลน้อยที่สุด และสามารถทำการบวกได้เร็ว เนื่องจากโอกาสที่จะมีการทดน้อย แต่ปัญหา คือ กระบวนการในการหารูปแบบคาโนนิคเป็นปัญหาที่มีความยากใช้เวลาในการแก้ปัญหานาน จึงมีงานวิจัย [10,11] ที่เสนอวิธีในการหารูปแบบในการแสดงจำนวนที่มีจำนวนเลขหนึ่งใกล้เคียงกับรูปแบบคาโนนิค โดยจะเรียกรูปแบบในการแสดงจำนวนเหล่านี้ว่ารูปแบบใกล้คาโนนิค (near-carnonic representation) ซึ่งใช้อัลกอริทึมเชิงละโมบ (greedy algorithm) ในการแก้ปัญหา โดยจะใช้เวลาเป็น $O\left(\frac{\log x}{\log \log x}\right)$

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

งานวิจัยที่เกี่ยวข้อง

1. การบวกของระบบจำนวนฐานคู่

งานวิจัยของดิมิทรอฟ [2] เสนอว่าก่อนจะทำการบวกนั้นจะต้องจัดรูปแบบของการแสดงค่าให้อยู่ในรูปแบบเฉพาะรูปแบบหนึ่งก่อน โดยรูปแบบนั้นเรียกว่ารูปแบบพร้อมบวก (addition ready DBNR) ซึ่งมีนิยาม ดังนี้

นิยาม รูปแบบพร้อมบวก *คือ รูปแบบแสดงค่าในระบบจำนวนฐานคู่ที่ไม่มีตำแหน่งที่มีค่าอยู่ติดกันเลย*

การจัดให้อยู่ในรูปแบบพร้อมบวกนั้นมีข้อดี คือ จะทำให้สามารถจำกัดสายการทดได้ และการเปลี่ยนรูปแบบจากรูปแบบใด ๆ ให้กลายเป็นรูปแบบพร้อมบวกนั้น ทำได้โดยการใช้กฎ ดังนี้

กฎที่ 1 กฎการลดแถว

$$2^i3^j + 2^{i+1}3^j = 2^i3^{j+1}$$

กฎที่ 2 กฎการลดหลัก

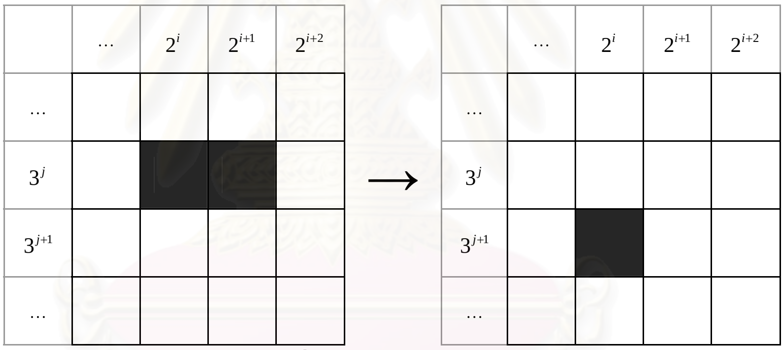
$$2^i3^j + 2^i3^{j+1} = 2^{i+2}3^j$$

นอกจากนี้ยังสามารถสร้างกฎอื่น ๆ ขึ้นมาเพิ่มเติมได้โดยอาศัยสมการดังนี้

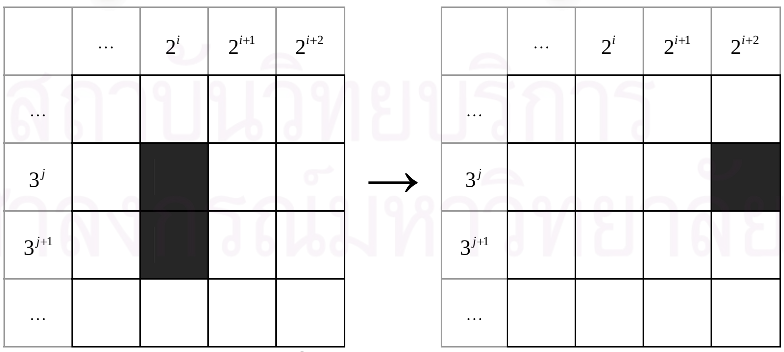
$$2^{i_1}3^{j_1} + 2^{i_2}3^{j_2} + \ldots + 2^{i_k}3^{j_k} = 2^{m_1}3^{n_1} + 2^{m_2}3^{n_2} + \ldots + 2^{m_l}3^{n_l}$$

เมื่อ $l < k$

กฎการลดของแถว และ กฎการลดของหลักสามารถแสดงในรูปของตารางได้ดังรูปที่ 2.2 และ 2.3 ตามลำดับ



รูปที่ 2.2 กฎการลดแถว



รูปที่ 2.3 กฎการลดหลัก

กระบวนการบวกทำได้โดยการนำตารางของตัวตั้ง และตัวบวกมาซ้อนทับกัน ซึ่งการซ้อนทับนี้สามารถกระทำได้โดยอาศัยกฎที่อธิบายได้ด้วยตัวปฏิบัติการทางตรรกศาสตร์ เมื่อทำการ

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บวกค่าของ x และ y ซึ่งให้ผลเป็น z กำหนดให้ $I_x(i,j)$ แทนค่าของตำแหน่งที่ (i,j) ในตารางแสดงค่าของ x กฎจะเป็นดังนี้

กฎ 1 $$I_z(i+1,j) = I_x(i,j) \text{ AND } I_y(i,j)$$

กฎ 2 $$I_z(i,j) = I_x(i,j) \text{ XOR } I_y(i,j)$$

และเมื่อทำการบวกเสร็จแล้ว จะทำการลดรูปให้เหลือตำแหน่งที่มีค่าเป็นหนึ่งน้อยลง และเพื่อให้พร้อมสำหรับการทำปฏิบัติการต่อไป โดยใช้กฎในการลดแถว และหลัก ซึ่งสามารถแสดงในรูปตรรกศาสตร์ได้ดังนี้

กฎ3 $$I_z(i,j+1) = I_z(i+1,j) \text{ AND } I_z(i,j)$$

กฎ 4 $$I_z(i+2,j) = I_z(i,j) \text{ AND } I_z(i,j+1)$$

แนวคิดของดิมิทรอฟนั้นเป็นแนวคิดพื้นฐานสำหรับการทำงานด้านระบบจำนวนฐานคู่สำหรับงานวิจัยอื่น และ การอธิบายการทำงานนั้นสามารถเข้าใจได้ง่าย เพราะใช้ตัวปฏิบัติการพื้นฐานทางตรรกศาสตร์เท่านั้น แต่แนวคิดนี้ยังไม่ได้คำนึงถึงเวลาที่จะใช้ในขั้นตอนของการลดรูป รวมทั้งกรณีพิเศษต่าง ๆ ที่อาจจะเกิดขึ้น และแนวคิดนี้นี้ยังไม่ได้มีการนำเสนอวิธีการนำกฎต่าง ๆ ไปสร้างเป็นวงจรที่สามารถทำงานได้จริงด้วย

ต่อมาในปี 2005 มีงานวิจัยของพานคาลา (Pankaala) [5] ที่เสนอแนวความคิดเกี่ยวกับพัฒนาการบวกของระบบจำนวนฐานคู่เพิ่มเติม โดยมีรายละเอียดบางส่วนแตกต่าง และเพิ่มเติมจากเดิม โดยเสนอว่าการบวกเลขในระบบจำนวนฐานคู่นั้น คือ การซ้อนทับกันของตารางของตัวตั้งและตัวบวก และเมื่อทำการซ้อนทับแล้วจะต้องทำให้ผลของการกระทำนั้นอยู่ในรูปพร้อมบวก เพื่อให้สามารถนำไปใช้ต่อได้สะดวก แต่มีการให้ความหมายของรูปแบบพร้อมบวกว่า เป็นรูปแบบที่ไม่มีตำแหน่งที่มีค่าในตารางซ้อนทับกัน และ ไม่มีตำแหน่งที่มีค่าติดกันในแถวเดียวกัน ซึ่งแตกต่างจากงานวิจัยเดิมที่สนใจตำแหน่งในหลักใด ๆ ด้วย การที่ลดความเข้มงวดของรูปแบบพร้อมบวกนั้น เพื่อให้การออกแบบวงจรสามารถเป็นไปได้ง่ายขึ้น โดยการลดรูปให้เป็นรูปพร้อมบวกนั้นจะใช้กฎ 2 ข้อ คือ

กฎ 1 กฎการลดช่องที่ซ้อนทับ

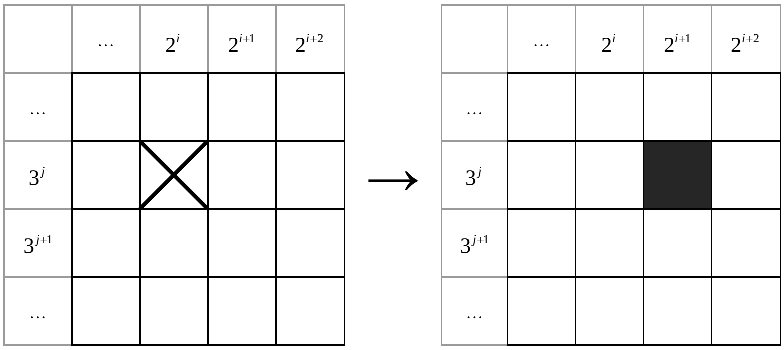
$$2^i3^j + 2^i3^j = 2^{i+1}3^j$$

กฎ 2 กฎการลดแถว

$$2^i3^j + 2^{i+1}3^j = 2^i3^{j+1}$$

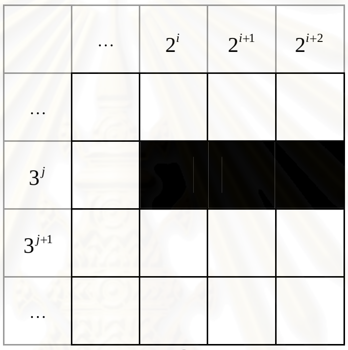
กฎข้อที่ 1 นั้นสามารถแสดงในรูปแบบของตารางได้ดังรูปที่ 2.4 โดยช่องที่มีการซ้อนทับนั้นจะแสดงด้วยเครื่องหมายกากบาท

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 2.4 กฎการลดช่องที่ซ้อนทับ

ซึ่งกฎสองข้อนี้เพียงพอที่จะเปลี่ยนรูปแบบจากรูปแบบใด ๆ ไปยังรูปแบบพร้อมบวกได้เสมอ แต่การนำกฎมาใช้จริงนั้น จะต้องมีการพิจารณาบางกรณีเป็นพิเศษ เช่น กรณีที่มีช่องที่มีค่าสามช่องอยู่ติดกันในแถวเดียวกัน ดังรูปที่ 2.5



รูปที่ 2.5 ตารางแสดงกรณีที่จะต้องพิจารณาเป็นพิเศษ

ถ้านำกฎมาใช้ในทันที จะพบว่าช่องที่มีค่า $2^{i+1}3^j$ จะถูกใช้ถึงสองครั้ง ทำให้ผลที่ได้ออกมามีค่าไม่ตรงกับค่าจริง ดังนั้น จะต้องมีการคำนึงถึงกรณีต่าง ๆ ให้ครบ และ จำนวนของกรณีที่จะต้องพิจารณานั้น จะขึ้นกับจำนวนกฎที่ใช้ เช่นในที่นี้ใช้กฎ 2 กฎ เมื่อจะพิจารณาการลดรูปของช่องที่ 2^i3^j จะต้องทราบถึงค่าของช่องที่ 2^i3^{j+1} และ 2^i3^{j+2} ก่อน จึงจะตัดสินได้ว่าควรให้ช่องที่ 2^i3^j มีค่าเป็นเท่าไร

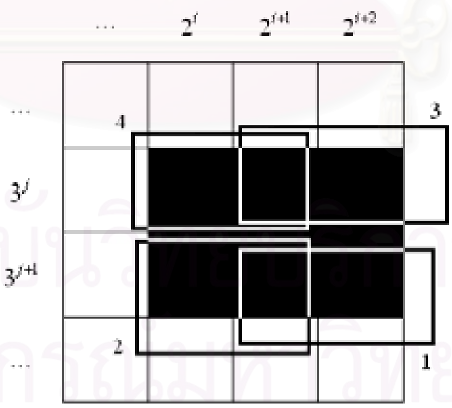
งานวิจัยนี้ยังแนะนำแนวทางในการพัฒนาวงจรสำหรับการบวก โดยจะใช้ข่ายงานประสาทเทียมระดับเซลล์ (cellular neural network) วงจรแอนะล๊อกสำหรับการพิจารณาการลดรูปของแต่ละตำแหน่ง โดยที่จะทำงานแบบอสมวารไปพร้อมกันในทุกตำแหน่ง และมีการทดลองสร้างคำสั่งของโปรแกรม Matlab ในการหาจำนวนสถานะที่จำเป็น เมื่อจะทำการใช้กฎข้อ 1 และ 2 ผลปรากฏว่าจะต้องใช้จำนวนสถานะทั้งสิ้น 6 สถานะ และจะมีวงจรป้อนกลับ (feedback loop) จำนวนมาก ซึ่งยากต่อการออกแบบวงจร จะเห็นว่าในงานวิจัยนี้ได้มีการพิจารณาผลกระทบต่อการออกแบบวงจรซึ่งเกิดจากใช้กฎที่แตกต่างกัน และมีการแนะนำการออกแบบวงจรเบื้องต้น แต่ยังไม่ได้มีการออกแบบวงจรให้เห็นจริง หรือลงไปในรายละเอียดของวงจรแต่อย่างใด หรือแม้กระทั่งวิเคราะห์ความเร็วของการทำงานเมื่อใช้กฎที่แตกต่างกัน

ในปีเดียวกันนั้นมีงานวิจัยของอิบราฮิม (Ibrahim) [6] ทำการสร้างวงจรแอนะล๊อกสำหรับการบวกระบบจำนวนฐานคู่โดยใช้ข่ายงานประสามเทียมระดับเซลล์ โดยมีการนิยามตัวปฏิบัติการบวกว่าเป็นการบวกจำนวนในระบบจำนวนฐานคู่ซึ่งอยู่ในรูปแบบพร้อมบวกสองจำนวน แล้วให้ได้ผลลัพท์เป็นรูปพร้อมบวกด้วยเช่นกัน ดังนั้น การบวกจึงสามารถแยกย่อยได้เป็นสองขั้นตอนใหญ่ คือ ขั้นสำหรับการรวมค่า และขั้นตอนสำหรับการจัดรูปแบบให้อยู่ในรูปแบบพร้อมบวกอีกครั้ง

ขั้นตอนการรวมค่านั้น เกิดจากการนำตารางมาซ้อนทับกัน และเมื่อมีช่องที่เกิดการซ้อนทับกันก็จะใช้กฎเช่นเดียวกับงานวิจัยที่แล้ว ซึ่งขั้นตอนนี้จะใช้เวลาคงที่ค่าหนึ่ง เพราะการบวกของตัวบวกที่เป็นรูปแบบพร้อมบวกสองตัวนั้น จะเกิดสายการทดไม่เกินหนึ่งขั้นเสมอ และสำหรับขั้นตอนการเปลี่ยนรูปเป็นรูปแบบพร้อมบวกนั้น จะใช้กฎการลดแถว และกฎที่จัดการกับช่องที่ซ้อนทับกันเนื่องจากผลของกฎการลดแถวอาจจะทำให้เกิดการซ้อนทับกับช่องที่อยู่ในแถวถัดไปได้ แต่การจะใช้กฎสำหรับแต่ละตำแหน่งนั้นจะมีข้อควรระวังอยู่กรณีหนึ่งคือ เมื่อตำแหน่งใด ๆ สามารถที่จะถูกจับคู่กับตำแหน่งอื่นได้มากกว่าหนึ่งวิธี จะต้องมีการตัดสินใจลำดับขั้นในการทำงาน โดยข้อที่ 1 มีความสำคัญมากที่สุดจะทำเป็นอันดับแรก ลำดับความสำคัญนั้นมีดังนี้

1. $2^{i+1}3^{j+1} + 2^{i+2}3^{j+1} = 2^{i+1}3^{j+2}$
2. $2^{i}3^{j+1} + 2^{i+1}3^{j+1} = 2^{i}3^{j+2}$
3. $2^{i+1}3^{j} + 2^{i+2}3^{j} = 2^{i+1}3^{j+1}$
4. $2^{i}3^{j} + 2^{i+1}3^{j} = 2^{i}3^{j+1}$

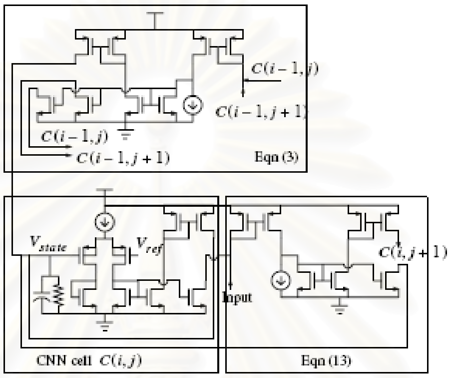
หรือสามารถแสดงลำดับความสำคัญเป็นรูปได้ดังรูปที่ 2.6



รูปที่ 2.6 ลำดับความสำคัญของการลดรูป

ในตอนท้ายมีการออกแบบวงจรสำหรับทำการบวก โดยออกแบบเป็นวงจรอะแนล๊อกซึ่งใช้หลักการของข่ายประสาทเทียมระดับเซลล์ คือ ให้แต่ละตำแหน่งมีวงจรซึ่งทำงานเป็นของตัวเองอิสระจากกัน แต่จะมีการส่งข้อมูลซึ่งกันและกัน ขั้นตอนแรก จะตั้งค่าเริ่มต้นของแต่ละวงจรให้มีค่าตรงกับตัวตั้ง และให้ข้อมูลขาเข้ามีค่าเท่ากับตัวบวก แล้วทำการบวกสองจำนวนเข้าด้วยกัน เมื่อผ่านไปเป็นเวลาคงที่หนึ่ง การบวกจะเสร็จสิ้น แต่ผลลัพท์ที่ได้จะยังไม่อยู่ในรูปแบบพร้อมบวก

ขั้นตอนต่อไปเป็นการลดจำนวนช่องที่มีค่าเป็นหนึ่ง โดยจะใช้สัญญาณไฟฟ้าที่รับจากตำแหน่งที่ (i, j) และ $(i+1, j)$ เพื่อที่กระตุ้นให้เกิดค่าในตำแหน่ง $(i, j+1)$ และใช้สัญญาณตอบกลับเพื่อที่ยกเลิกค่าในสองตำแหน่งแรก ซึ่งการทำงานในขั้นตอนนี้ก็ใช้เวลาคงที่ค่าหนึ่ง ถ้าเกิดช่องที่ซ้อนทับกัน ก็จะใช้วงจรเดียวกับที่ใช้บวก แต่เปลี่ยนข้อมูลให้เหมาะสม ส่วนการจัดการกับลำดับความสำคัญของคู่ลำดับต่าง ๆ นั้นก็จะใช้สัญญาณตอบกลับจากคู่ลำดับที่มีความสำคัญสูง ไปยังคู่ลำดับที่มีความสำคัญต่ำสุด เพื่อไม่ให้คู่ลำดับที่มีความสำคัญต่ำทำงานก่อนคู่ลำดับที่มีความสำคัญสูงกว่าได้ ผลที่ได้จากงานวิจัย [6] จึงสรุปเป็นวงจรสำหรับตำแหน่งใด ๆ ดังรูปที่ 2.7



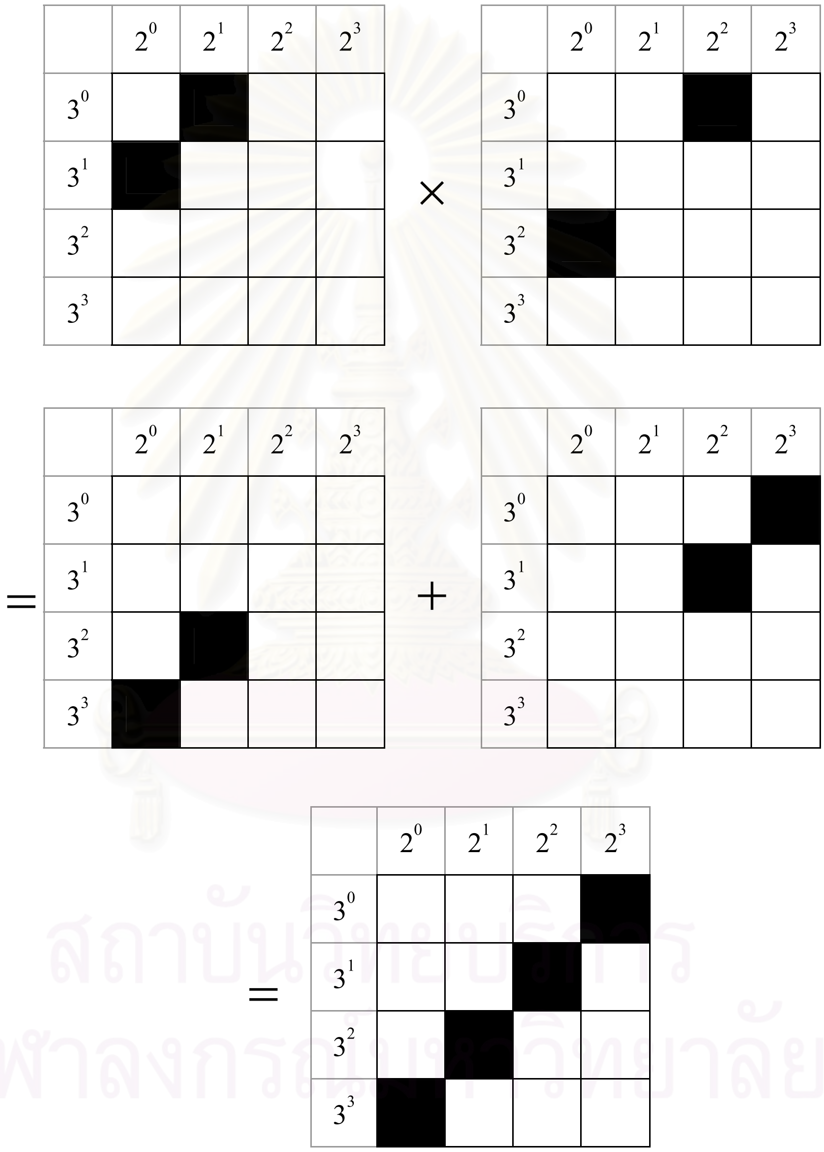
รูปที่ 2.7 วงจรสำหรับการบวกที่ตำแหน่งใด ๆ

ถ้าให้เวลาที่ใช้ในแต่ละขั้นเป็น T และมีรูปแบบแสดงค่ามีทั้งหมด n แถวแล้ว ผลของการจำลองการทำงานของวงจรนี้ พบว่าใช้เวลาในการบวกไม่เกิน $n(T+1)$ โดยเวลาที่ใช้นั้นจะไม่แน่นอนขึ้นอยู่กับลักษณะของข้อมูลทั้งสอง

งานวิจัยชิ้นนี้ได้ทำการพัฒนาวงจรบวกให้สามารถทำงานด้วยความเร็วที่สูง แต่จะไม่สามารถคาดเดาเวลาที่ใช้ที่แน่นอนในการบวกของข้อมูลใด ๆ ได้ ทำให้การนำผลลัพท์บางส่วนที่ได้ไปใช้ต่อก่อนที่การบวกจะเสร็จสิ้น หรือการนำเอาวิธีการอื่น ๆ ที่สามารถพัฒนาความเร็วของวงจรกระทำได้ยาก อีกทั้งวงจรที่ออกแบบมาเป็นวงจรแอนะล๊อกที่อาจจะเกิดสัญญาณรบกวนทำให้ข้อมูลผิดพลาดได้ง่าย และการบวกที่ได้เสนอมากระทำได้ต่อเมื่อฐานเป็น สอง และสามเท่านั้น ทำให้ไม่มีความยืดหยุ่นในการพัฒนาระบบจำนวนฐานคู่เมื่อจะใช้ฐานเป็นจำนวนอื่น ถ้าสามารถสร้างตัวปฏิบัติการบวกสำหรับระบบจำนวนฐานคู่ ซึ่งมีฐานเป็นสอง และจำนวนเต็มบวกใด ๆ อีกหนึ่งจำนวน ก็จะทำให้การพัฒนาระบบจำนวนฐานคู่เป็นไปได้อย่างแพร่หลาย และยืดหยุ่นมากขึ้น

2. การคูณของระบบจำนวนฐานคู่

งานวิจัยของดิมิทรอฟ [2] นั้น ยังได้กล่าวถึงการคูณของระบบจำนวนฐานคู่ไว้ โดยการคูณนั้นทำได้โดยการกระจายค่าตัวตั้งไปคูณกับแต่ละตำแหน่งที่มีค่าของตัวคูณ หลังจากนั้นจึงนำค่าที่ได้ทั้งหมดมารวมเข้าด้วยกัน เมื่อพิจารณาในรูปของตารางแสดงค่าแล้ว การคูณเกิดจากผลรวมของตัวตั้งที่ถูกเลื่อนตำแหน่งไป โดยการเลื่อนตำแหน่งจะเลื่อนตามตำแหน่งที่มีค่าของตัวคูณ ซึ่งเขียนแสดงเป็นภาพได้ดังรูปที่ 2.8



รูปที่ 2.8 การคูณของ 5 และ 13 ได้ผลลัพท์เป็น 65

จากรูปสามารถเขียนในรูปของสมการได้ดังนี้

$$
\begin{aligned}
\left(2^1 3^0 + 2^0 3^1\right) \times \left(2^2 3^0 + 2^0 3^2\right) &= 2^2 3^0 \left(2^1 3^0 + 2^0 3^1\right) + 2^0 3^2 \left(2^1 3^0 + 2^0 3^1\right) \\
&= 2^3 3^0 + 2^1 3^2 + 2^2 3^1 + 2^0 3^3 \\
&= 65
\end{aligned}
$$

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ซึ่งให้ผลที่ถูกต้อง แต่ก็เป็นเพียงการชี้แนะแนวทางเบื้องต้นเท่านั้น ยังไม่มีการกล่าวถึงวิธีการในการสร้าง หรือ ไม่ได้คำนวณเวลาที่ใช้เลย

งานวิจัยของพานคาลา [5] ที่เคยกล่าวมาแล้วนั้น ก็มีการแนะนำแนวทางในการพัฒนากระบวนการคูณ โดยแนะนำกระบวนการในการเลื่อนตัวตั้งว่าสามารถนำบางส่วนของวงจรบวกมาประยุกต์ใช้ได้ โดยการเลื่อนตำแหน่งทางแกน x นั้นสามารถใช้วงจรที่ทำการจัดการตำแหน่งที่เกิดการซ้อนทับกัน ซึ่งผลของการซ้อนทับนั้นก็คือ จำนวนที่ถูกเลื่อนไปตามแนวแกน x ส่วนการเลื่อนไปตามแนวแกน y นั้น แนะนำให้ใช้วงจรในส่วนที่จัดการกับกฎการลดของแถว ซึ่งผลที่ได้จะถูกเลื่อนไปตามแนวแกน y หนึ่งตำแหน่ง แต่งานวิจัยนี้ไม่ได้กล่าวถึงวิธีการคูณหลังจากการเลื่อนตำแหน่งเลย

3. ระบบจำนวนลูกผสมแบบมีเครื่องหมาย (Hybrid Signed-Digit Number Systems)

ระบบจำนวนลูกผสมแบบมีเครื่องหมาย [11] นี้ถูกเสนอเมื่อปี 1994 เป็นการพัฒนาความเร็วของการปฏิบัติการทางเลขคณิตอีกวิธีหนึ่ง ซึ่งนำเอาระบบจำนวนชนิดมีเครื่องหมายมาพัฒนา โดยระบบจำนวนแบบมีเครื่องหมายนี้ เป็นระบบจำนวนที่มีชุดตัวเลขเป็น 0, 1 และ -1 มีข้อดีคือสามารถจำกัดสายการทดให้อยู่เพียงขั้นเดียว แต่จากชุดตัวเลขจะพบว่าต้องใช้พื้นที่ในการเก็บเลขหนึ่งตัวถึงสองบิท เมื่อเปรียบเทียบกับระบบจำนวนส่วนเติมเต็มของสอง (two's complement number system) ซึ่งใช้พื้นที่ในการเก็บเพียงตำแหน่งละหนึ่งบิท แต่มีการส่งต่อตัวทดที่ไม่จำกัด ก็พบว่าการเพิ่มความเร็วในการทำปฏิบัติการทางเลขคณิตนั้นได้มาด้วยการแลกกับพื้นที่ที่ใช้เก็บข้อมูล จึงเสนอระบบจำนวนที่สามารถปรับเปลี่ยนสมบัติให้ตรงกับความต้องการได้ ทั้งในด้านความเร็วและพื้นที่เก็บข้อมูล

ระบบจำนวนลูกผสมแบบมีเครื่องหมายนี้จะให้บางตำแหน่งเป็นตำแหน่งที่มีเครื่องหมาย ส่วนตำแหน่งอื่น ๆ จะไม่มีเครื่องหมาย จะสามารถพิสูจน์ได้ว่าสายการทดนั้นจะไม่เกินระยะทางระหว่างตำแหน่งที่มีเครื่องหมาย ซึ่งการบวกนั้นจะทำเป็นลำดับ เริ่มในตำแหน่งที่มีเครื่องหมายก่อน แล้วตัวทดจะถูกส่งต่อไปเรื่อย ๆ จนไปหยุดที่ตำแหน่งที่มีเครื่องหมายตัวถัดไปการบวกสำหรับตำแหน่งที่มีเครื่องหมายจะใช้กฎมาเลือกคำตอบที่ได้ โดยจะมีการคำนึงถึงค่าของตำแหน่งก่อนหน้าประกอบด้วย เพื่อให้สามารถเก็บค่าของตัวทดที่จะถูกส่งเข้ามาในอนาคตได้ทั้งหมดสำหรับตำแหน่งที่ไม่มีเครื่องหมายนั้นก็จะใช้การบวกทั่วไป

แนวความคิดของระบบจำนวนลูกผสมนี้สามารถนำมาดัดแปลงใช้กับการบวกของจำนวนฐานคู่ได้ เมื่อต้องการจำกัดการทดให้อยู่ในขอบเขตที่ต้องการ เพียงแค่ปรับเปลี่ยนลักษณะ และชุดตัวเลขให้เหมาะสมเท่านั้น

บทที่ 3

อัลกอลิทึมการบวกเชิงกำหนดสำหรับจำนวนฐานคู่

จากที่ได้กล่าวในบทที่ผ่านมาว่ากระบวนการในการบวกที่มีการเสนอมาแล้วนั้น เป็นกระบวนการที่ไม่มีลำดับการทำงานที่แน่นอน กล่าวคือ ลำดับในการทำงานนั้นจะขึ้นอยู่กับลักษณะของข้อมูลที่จะนำมาบวก ดังนั้น จึงไม่สามารถสรุปได้ว่าบริเวณใดของรูปแบบแสดงค่าที่ทำการบวกเสร็จสิ้นแล้วจนกว่ากระบวนการบวกทั้งหมดจะเสร็จสิ้น ส่งผลให้ไม่สามารถนำผลลัพท์บางส่วนมาใช้ก่อนที่กระบวนการบวกจะเสร็จสิ้นได้ โดยสิ่งที่เป็นอุปสรรคต่อการบวกนั้นก็คือ การส่งต่อตัวทด ซึ่งมีการส่งต่อกันทั้งในแถว และ หลัก แต่เมื่อพิจารณาการแพร่ของตัวทดโดยละเอียดแล้วพบว่าการแพร่ของตัวทำนั้นมีลักษณะที่แน่นอน ส่งผลให้สามารถสร้างกระบวกการในการบวกที่ทำงานเป็นลำดับที่แน่นอนได้ โดยมีรายละเอียดดังนี้

การแพร่ตัวทดของระบบจำนวนฐานคู่

การบวกของระบบจำนวนฐานคู่นั้นสามารถกระทำได้โดยการนำตารางแสดงค่าของตัวตั้งและตัวบวกมาซ้อนทับกัน เช่นที่ดิมิทรอฟได้กล่าวไว้ โดยที่รูปแบบแสดงค่าทั้งสองนั้นจะอยู่ในรูปแบบพร้อมบวก ซึ่งมีนิยามดังนี้

นิยาม รูปแบบพร้อมบวก *คือ รูปแบบแสดงค่าในระบบจำนวนฐานคู่ที่ไม่มีตำแหน่งที่มีค่าอยู่ติดกันในแถวเดียวกันเลย*

การซ้อนทับของตารางนั้นจะทำให้เกิดการทด ในกรณีที่ตารางทั้งสองมีค่าในตำแหน่งเดียวกัน หรือ เรียกได้ว่าเกิดการชนกันเพียงกรณีเดียวเท่านั้น โดยตัวทดที่เกิดขึ้นนั้นจะถูกส่งต่อไปยังหลักที่มีค่าประจำหลักมากกว่า หรือ อีกนัยหนึ่งคือมีการส่งตัวทดจากซ้ายไปขวา และการส่งตัวทดนี้จะเกิดขึ้นในแถวเดียวกับตำแหน่งที่เกิดการชน การทดลักษณะนี้ก็คือการทดเช่นเดียวที่เกิดจากการบวกเลขฐานสองนั่นเอง จะเห็นว่า การนำตารางมาซ้อนทับกัน ไม่ทำให้เกิดการทดข้ามแถว ดังนั้น การบวกจึงสามารถมองได้ในอีกมุมหนึ่ง เป็นการนำเอาแถวที่มีค่าประจำแถวเท่ากันมาบวกกัน โดยการบวกของแถวทั้งหมดนั้นสามารถกระทำได้พร้อมกัน เนื่องจากเป็นการกระทำที่อิสระต่อกัน

ในส่วนของตัวทดระหว่างแถวนั้น เกิดจากกฎการลดแถว ที่ใช้เพื่อเปลี่ยนรูปแบบใด ๆ ให้ไปอยู่ในรูปแบบพร้อมบวก ตัวทดนี้มีทิศทางการส่งจากแถวที่มีค่าประจำแถวน้อย ไปยังแถวที่มีค่าประจำแถวมาก และเมื่อรูปแบบแสดงจำนวนอยู่ในรูปแบบพร้อมบวกแล้ว จะไม่สามารถเกิดตัวทดระหว่างแถวนี้ได้อีก

จากลักษณะการแพร่ของตัวทดทั้งสองชนิดขั้นต้น จะเห็นว่า มีการแพร่จากตำแหน่งที่มีค่าน้อย ไปยังตำแหน่งที่มีค่ามากกว่าเสมอ ดังนั้น ถ้ามีกระบวนการบวกที่จัดการกับตำแหน่งที่มีค่าน้อยก่อน แล้วจึงทำไปยังตำแหน่งที่มีค่ามาก ก็จะสามารถนำคำตอบบางส่วนไปใช้ได้ก่อนที่กระบวนการบวกทั้งกระบวนการจะสิ้นสุดลง

บทตั้งที่ 3.1 *การบวกจำนวนที่อยู่ในรูปแบบพร้อมบวก จะก่อให้เกิดสายการทดระหว่างหลักเพียงขั้นเดียวเท่านั้น*

พิสูจน์ ให้ A และ B เป็นจำนวนในระบบจำนวนฐานคู่ที่อยู่ในรูปพร้อมบวก ดั้งนั้น A และ B สามารถเขียนอยู่ในรูปของ

$$A = \sum_{i,j} a_{i,j} 2^i 3^j$$

$$= \sum_j \left(\sum_i a_{i,j} 2^i \right) 3^j$$

$$B = \sum_{i,j} b_{i,j} 2^i 3^j$$

ดังนั้นไม่มี $a_{i,j}$ และ $a_{i+1,j}$ หรือ $b_{i,j}$ และ $b_{i+1,j}$ ใด ๆ ที่มีค่าเป็น 1 ทั้งคู่

โดย $\sum_i a_{i,j} 2^i$ ก็คือจำนวนฐานสองหนึ่งจำนวน ดั้งนั้น การบวกของ A และ B สามารถเขียนให้อยู่ในรูป

$$A + B = \sum_{i,j} a_{i,j} 2^i 3^j + \sum_{i,j} b_{i,j} 2^i 3^j$$

$$= \sum_j \left(\sum_i (a+b)_{i,j} 2^i \right) 3^j$$

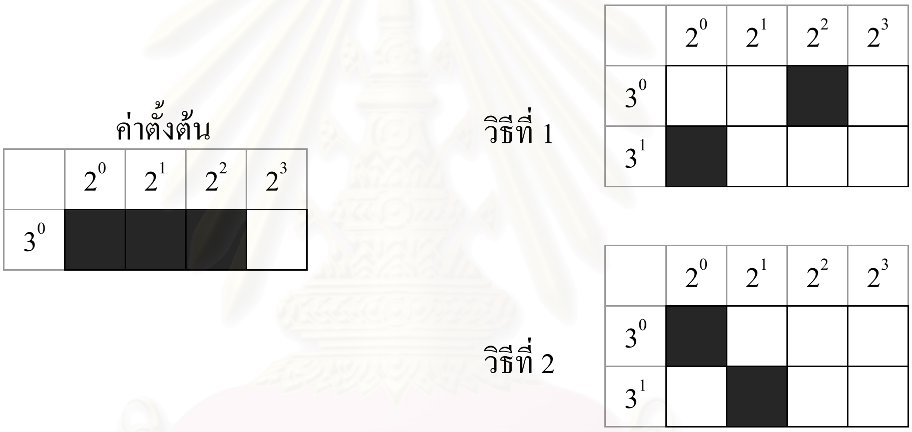
ซึ่งก็คือ การบวกของแต่ละแถวแยกจากกัน โดยที่ทุกแถวสามารถทำการบวกพร้อมกันทั้งหมด และจำนวนทั้งหมดนั้นอยู่ในรูปพร้อมบวก ซึ่งจะไม่มีเลข 1 ติดกันเลย และการบวกเลขฐานสองที่ไม่มี 1 ติดกันเลยนั้น จะเกิดการทดไม่เกิด 1 ขั้นเสมอ สรุป การบวกนี้มีสายการทดเพียงขั้นเดียวเสมอ ■

จากบทตั้งที่ 3.1 จะเห็นว่าการบวกจำนวนที่อยู่ในรูปพร้อมบวกนั้นสามารถทำได้ในเวลาคงที่ค่าหนึ่ง เพราะสายการทดจำกัดอยู่เพียงแค่หนึ่งขั้นเสมอ แต่ผลลัพท์จากการบวกนี้อาจจะยังไม่อยู่ในรูปแบบพร้อมบวก ทำให้การนำไปใช้ต่อนั้นไม่สะดวก ดังนั้นจึงต้องมีการกำจัดตำแหน่งที่มีค่าที่อยู่ติดกันในแถวเดียวกันให้หมด

กระบวนการกำจัดตำแหน่งที่มีเลขหนึ่งติดกันในแถว

เนื่องจากผลลัพท์ของการบวกที่ต้องการนั้นจะอยู่ในรูปพร้อมบวก ดังนั้นจึงต้องมีวิธีการในการกำจัดตำแหน่งที่มีเลขหนึ่งติดกันออกไป ซึ่งส่วนประกอบสำคัญในขั้นตอนนี้ก็คือ กฎการลดแถวนั้นเอง ซึ่งกฎการลดแถวนี้ จะเปลี่ยนตำแหน่งที่มีค่าเป็นหนึ่งที่ติดกัน ไปเป็นตำแหน่งที่มีค่าเป็นหนึ่งในแถวถัดไป ซึ่งเป็นการสร้างตัวทดระหว่างแถว ดังนั้น ถ้านำกฎข้อนี้ไปใช้กับทุกตำแหน่งของตารางพร้อม ๆ กันหมด ก็สามารถจะแยกตัวทดออกมาเป็นอีกตารางได้

แต่กฎข้อนี้มีปัญหา คือ กฎนี้มีความกำกวม กล่าวคือ ในบางสถานการณ์กฎนี้สามารถทำให้เกิดผลลัพท์ที่ถูกต้องได้มากกว่าหนึ่งผลลัพท์ โดยสถานการณ์นี้จะเกิดเมื่อมีตำแหน่งที่มีค่าเป็นหนึ่งมากกว่าสองตัวขึ้นไปอยู่ติดกัน ดังรูปที่ 3.1 ซึ่งเป็นแถวที่มีค่าเท่ากับ 7 นั้นสามารถจะนำกฎลดแถวนี้ ไปใช้ได้สองวิธีดังรูป โดยวิธีที่หนึ่งจะยุบค่าของหลักที่ 0 และ 1 เข้าด้วยกัน ส่วนวิธีที่สองนั้นจะยุบค่าของหลักที่ 1 และ 2 ซึ่งทั้งสองวิธีนั้นให้ค่าที่ถูกต้องเท่ากันทั้งคู่



รูปที่ 3.1 ความกำกวมของกฎการลดแถว

จากที่ได้แสดงให้เห็นแล้วว่ากฎการลดแถวนั้นมีความกำกวม แต่การบวกที่ต้องการนั้นจะต้องเป็นการบวกที่ทำงานอย่างเป็นลำดับที่แน่นอน ดังนี้จึงต้องกำจัดความกำกวมนี้ให้หมดไปโดยการให้ความสำคัญกับหลักคู่และหลักคี่ไม่เหมือนกัน ในทฤษฎีบทที่ 3.1 เสนอกระบวนการในการนำกฎการลดแถวมาใช้กับตารางแสดงค่าในระบบจำนวนฐานคู่ทั้งตาราง โดยเป็นการทำงานที่มีลำดับที่แน่นอน ไม่มีความกำกวม

ทฤษฎีบทที่ 3.1 *ให้ A เป็นจำนวนฐานคู่ที่อยู่ในรูปแบบแสดงค่าใด ๆ A จะสามารถถูกแบ่งโดยกฎการลดแถวเป็นสองจำนวน เรียกว่า จำนวนเหลือ A' และ จำนวนทด R โดยที่จำนวนทั้งสองไม่มีตำแหน่งที่มีค่าเป็น 1 ติดกันในแถวเดียวกัน ภายในเวลาคงที่ ด้วยอัลกอริทึมแยกตาราง*

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

อัลกอริทึม 1 อัลกอริทึมแยกตาราง

Input: $A = \sum_{i,j} a_{i,j} 2^i 3^j$

Output: $A' = \sum_{i,j} a'_{i,j} 2^i 3^j$, $R = \sum_{i,j} r_{i,j} 2^i 3^j$ โดยที่ $A = A' + R$

ขั้นตอน

1: if $(a_{2i,j} = 1$ and $a_{2i+1,j} = 1)$

do $a'_{2i,j} = 0, r_{2i,j+1} = 1, a_{2i+1,j} = 0$

else $a'_{2i,j} = a_{2i,j}$, $r_{2i,j+1} = 0$

2: if $(a_{2i+1,j} = 1$ and $a'_{2i+2,j} = 1)$

do $a'_{2i+1,j} = 0, r_{2i+1,j+1} = 1, a'_{2i+2,j} = 0$

else $a'_{2i+1,j} = a_{2i+1,j}$, $r_{2i+1,j+1} = 0$

พิสูจน์ จะต้องพิสูจน์ว่า ผลจากอัลกอริทึมนี้ไม่มีเลข 1 ติดกันในแถวเดียวกัน, ผลรวมของคำตอบมีค่าเท่ากับตัวตั้ง และ อัลกอริทึมนี้สามารถทำงานได้ในเวลาคงที่

1) A' จะไม่มีเลข 1 ติดกัน เพราะ ถ้ามี $a_{i,j}$ และ $a_{i+1,j}$ คู่ใดที่เป็นหนึ่งทั้งคู่ $a'_{i,j}$ จะถูกตั้งค่าให้เป็น 0 เสมอ

ส่วน R นั้น จะไม่มีเลข 1 ติดกัน เนื่องจาก ถ้าสมมติให้ มีคู่ใดเป็น 1 ทั้งคู่ จะสามารถแบ่งเป็น 2 กรณี

กรณีที่ 1 $r_{2i,j} = 1,\ r_{2i+1,j} = 1$

เมื่อ $r_{2i,j} = 1$ จะได้ว่า $a_{2i,j-1} = 1$ และ $a_{2i+1,j-1} = 1$ เมื่อผ่านขั้นตอนที่ 1 ค่าของ $a_{2i+1,j-1}$ จะถูกเปลี่ยนให้เป็น 0 ทำให้ในขั้นตอนที่สอง $r_{2i+1,j} = 0$ ขัดแย้งกับสมมติฐาน

กรณีที่ 2 $r_{2i+1,j} = 1,\ r_{2i+1,j} = 1$

ใช้วิธีพิสูจน์เช่นเดียวกันกับกรณีแรก ซึ่งจะเกิดข้อขัดแย้งกับสมมติฐาน

จากทั้งสองกรณี สรุปได้ว่า ผลจากอัลกอริทึมนี้จะไม่มีเลข 1 ติดกันเลย

2) ค่าประจำหลักของ ตำแหน่ง (i, j) มีค่าเป็น $2^i 3^j$ ในขั้นตอนที่ 1 กรณีที่ $a_{2i,j} = 1$ และ $a_{2i+1,j} = 1$ มีค่าเชิงตัวเลขเป็น $2^{2i}3^j + 2^{2i+1}3^j$ ผลลัพท์ที่ได้เป็น $a'_{2i,j} = 0$, $r_{2i,j+1} = 1$ และ $a_{2i+1,j} = 0$ มีค่าเป็น $2^{2i}3^{2j+1}$ ซึ่งเท่ากัน ส่วนกรณีอื่น สามารถพิสูจน์ได้ในลักษณะเดียวกัน

3) ในทั้งสองขั้นตอนนั้น จะเห็นว่า ทุก ๆ คู่ สามารถทำงานได้พร้อมกัน เพราะไม่ต้องมีการส่งข้อมูลกันระหว่างคู่ใด ๆ เลย ดังนั้น เวลาที่ใช้ในการตรวจสอบ และตั้งค่าสำหรับในแต่ละขั้นตอนจึงคงที่ เวลาที่ใช้ในอัลกอริทึมนี้จึงคงที่ ■

บทแทรกที่ 3.1 *อัลกอริทึมแยกตารางนั้น จะทำให้เกิดจำนวนทดที่มีแถวที่มีค่าประจำแถวน้อยที่สุดเป็น* 0 *เสมอ*

พิสูจน์ จากอัลกอริทึมแยกตารางนั้น ค่าของแถวที่ j ของจำนวนทด R จะเกิดจากการใช้กฎการลดแถวกับแถวที่ $j-1$ ของตัวตั้ง และแถวที่น้อยที่สุดเป็นแถวที่ 0 ทำให้ผลเก็บอยู่ในแถวที่ 1 ของจำนวนทด ซึ่งทำให้แถวที่ 0 ของจำนวนทดไม่มีค่าเสมอ ■

บทแทรกที่ 3.2 *ถ้าข้อมูลนำเข้าของอัลกอริทึมแยกตาราง มีแถวที่มีค่าประจำแถวน้อยที่สุด* n *แถวอยู่ในรูปพร้อมบวกแล้ว แถวที่มีค่าประจำแถวน้อยที่สุด* $n+1$ *แถวของจำนวนทดที่เกิดขึ้นจะมีมีค่าเป็น* 0

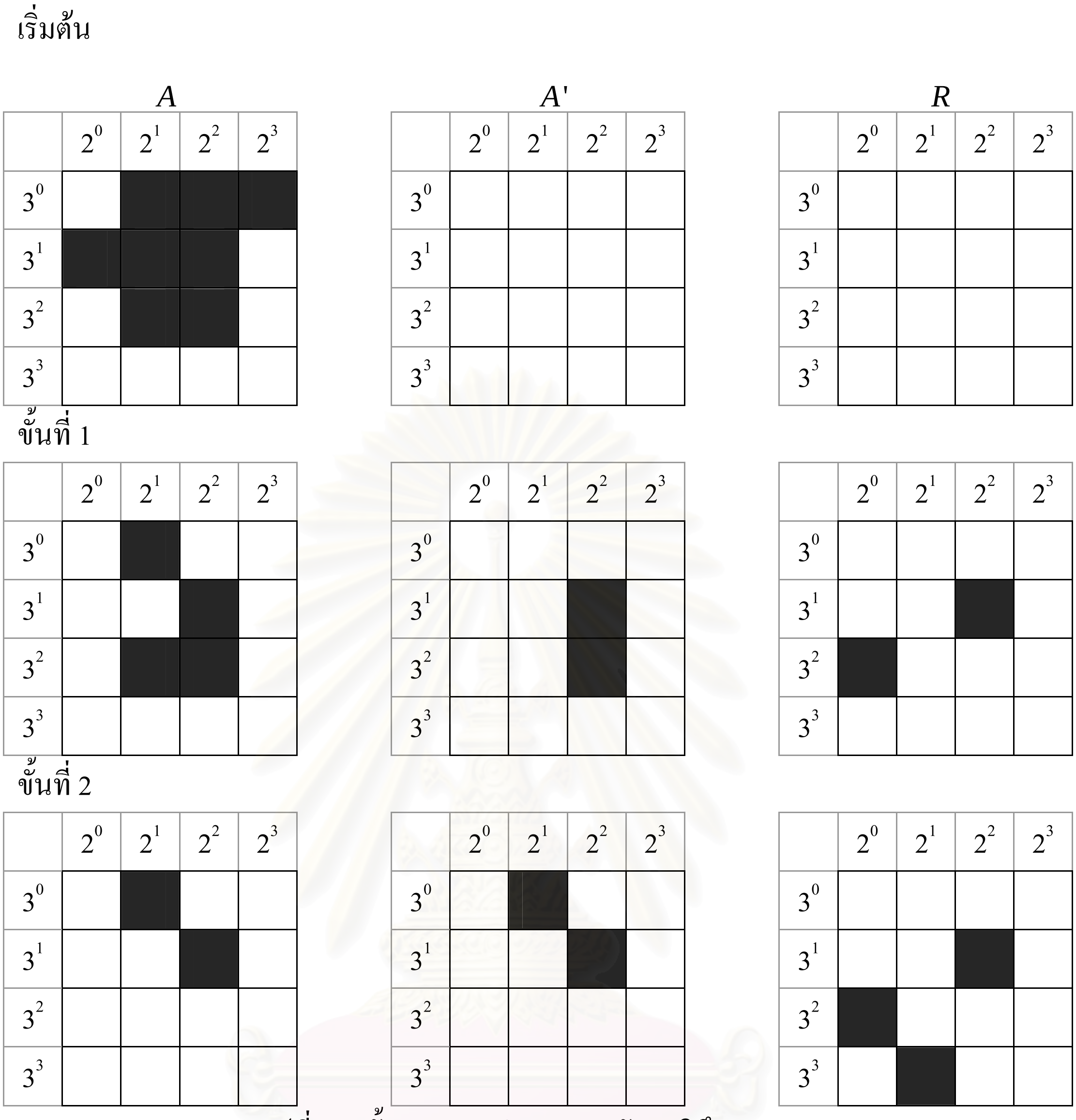
พิสูจน์ จากอัลกอริทึมเห็นได้ชัดว่า ถ้าแถวใดของข้อมูลนำเข้าไม่มีตำแหน่งที่มีค่าติดกัน จำนวนทดที่เกิดขึ้นจากแถวนั้น จะมีค่าเป็น 0 เสมอ ดังนั้น ถ้าข้อมูลนำเข้ามีแถวที่มีค่าประจำแถวน้อยสุด n แถวอยู่ในรูปพร้อมบวก จำนวนทดจะมีแถวที่ 1 ถึง $n+1$ เป็น 0 และจากบทแทรกที่ 3.1 กล่าวว่าแถวที่ 0 ของจำนวนทดมีค่าเป็น 0 เสมอ จึงสรุปได้ว่า เมื่อข้อมูลนำเข้าของอัลกอริทึมแยกตารางนั้นมีแถวที่มีค่าประจำแถวน้อยที่สุด n แถวอยู่ในรูปพร้อมบวกแล้ว จำนวนทดที่ได้จะมีแถวที่มีค่าประจำแถวน้อยที่สุด $n+1$ แถวมีค่าเป็น 0 ■

บทแทรกที่ 3.3 *จำนวนทด และจำนวนเหลือจากอัลกอริทึมการลดแถวนั้นอยู่ในรูปพร้อมบวกทั้งสองจำนวน*

พิสูจน์ จากทฤษฎีบทที่ 3.1 ผลจากอัลกอริทึมการลดแถวทั้งสองจำนวนจะไม่มีตำแหน่งที่มีค่าติดกันในแถวเดียวกัน ซึ่งตรงกับนิยามของจำนวนพร้อมบวกที่จะไม่มีตำแหน่งที่มีค่าติดกันในแถวเดียวกันเลย ■

รูปที่ 3.2 แสดงตัวอย่างการทำงานของอัลกอริทึมแยกตาราง โดยเป็นการนำเลข 89 มาแยกออกเป็นสองจำนวน โดยผลทั้งสองจำนวนมีค่าเป็น 14 และ 75 ตามลำดับ

ถึงแม้ผลจากอัลกอริทึมนี้จะอยู่ในรูปพร้อมบวก แต่จะแยกเป็นสองจำนวน ซึ่งไม่ตรงกับความต้องการ เพราะผลจากการบวกจะต้องรวมเป็นจำนวนเดียวเท่านั้น ทำให้จะต้องมีการรวมจำนวนเหลือ และ จำนวนทดเข้าด้วยกันอีกครั้ง



รูปที่ 3.2 ขั้นตอนการทำงานของอัลกอริทึมแยกตาราง

อัลกอริทึมการบวกจำนวนฐานคู่

จากบทแทรกที่ 3.1 นั้นทำให้เห็นว่าจำนวนทดจากการแยกตางรางนั้นจะมีแถวที่มีค่าประจำแถวน้อยที่สุดเป็น 0 เสมอ และจากบทแทรกที่ 3.3 จะเห็นว่าแถวที่มีค่าประจำแถวน้อยที่สุดของจำนวนเหลือนั้นอยู่ในรูปจำนวนพร้อมบวก ดังนั้น ถ้านำจำนวนเหลือ และจำนวนทดมาบวกเข้าด้วยกัน แถวที่มีค่าประจำแถวน้อยที่สุดก็จะอยู่ในรูปจำนวนพร้อมบวกเสมอ ถ้านำผลรวมนี้ส่งเข้าไปยังอัลกอริทึมแยกตารางอีก จำนวนทดที่ได้จะมีแถวที่มีค่าประจำแถวน้อยที่สุด 2 แถวเป็น 0 ทั้งคู่ ซึ่งมีการพิสูจน์แล้วในบทแทรกที่ 3.2 ดังนั้น ถ้าทำการบวกจำนวนเหลือ และจำนวนทด แล้วนำผลที่ได้ส่งเข้าไปยังอัลกอริทึมแยกตารางเป็นรอบ ๆ จำนวนทดจะมีแถวที่มีค่าเป็น 0 มากขึ้นเรื่อย ๆ และถ้าขนาดของรูปแบบแสดงจำนวนนั้นคงที่ คือ ไม่สามารถเพิ่มจำนวนแถว และหลักได้ เมื่อทำ

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ทั้งสองขั้นตอนนี้ไปเท่ากับจำนวนแถวของรูปแบบแสดงจำนวน ค่าของจำนวนทดนั้นจะมีค่าเป็น 0 ซึ่งหมายความว่า จำนวนเหลือจะมีค่าเท่ากับผลบวกที่ต้องการ

ทฤษฎีบทที่ 3.2 *การบวกจำนวนพร้อมบวกในระบบจำนวนฐานคู่ แล้วได้ผลลัพท์ซึ่งอยู่ในรูปแบบพร้อมบวกนั้น ทำได้ในเวลาเชิงเส้น สัมพันธ์กับจำนวนแถวของรูปแบบแสดงค่า โดยอัลกอริทึมการบวก*

อัลกอริทึม 2 อัลกอริทึมการบวก

Input: $A = \sum_{i,j} a_{i,j} 2^i 3^j$

$B = \sum_{i,j} b_{i,j} 2^i 3^j$

Output: $C = \sum_{i,j} c_{i,j} 2^i 3^j$ โดยที่ $C = A + B$

Define $n =$ the amount of rows in the representation

ขั้นตอน

```
j = 0
while (j < n)
do    A = A + B
      A, B ← split(A)
      j = j + 1
C = A
```

พิสูจน์ จะต้องพิสูจน์ว่า การบวกให้ค่าที่ถูกต้อง, ผลลัพท์อยู่ในรูปแบบพร้อมบวก, อัลกอริทึมใช้เวลาแปรผันตรงกับจำนวนแถวของรูปแบบแสดงค่า

1) อัลกอริทึมนี้ประกอบด้วยการบวกซึ่งถูกต้องอยู่แล้ว และการแยกตารางนั้นได้มีการพิสูจน์ความถูกต้องของค่าไว้ในทฤษฎีบทที่ 3.1 แล้ว ดังนั้นการทำงานในแต่ละรอบจึงให้ผลลัพท์ที่ถูกต้อง นอกจากนี้ บทแทรกที่ 3.1 ถึง 3.3 สามารถสรุปว่า หลังจากผ่านไป n รอบ จำนวนทดจากอัลกอริทึมแยกตารางจะมีค่าเป็น 0 ทำให้จำนวนเหลือมีค่าเท่ากับผลบวกที่ต้องการ และอัลกอริทึมนี้คืนจำนวนเหลือออกมา ทำให้ได้ผลบวกที่ถูกต้อง

2) ผลลัพท์จากอัลกอริทึมนี้ จะเป็นผลลัพท์เดียวกับผลลัพท์ของอัลกอริทึมแยกตาราง ซึ่งบทแทรกที่ 3.3 กล่าวว่า ผลลัพท์จากอัลกอริทึมแยกตารางจะอยู่ในรูปพร้อมบวกทั้งสองจำนวน ดังนั้นผลลัพท์จากอัลกอริทึมนี้จึงอยู่ในรูปพร้อมบวกเช่นกัน

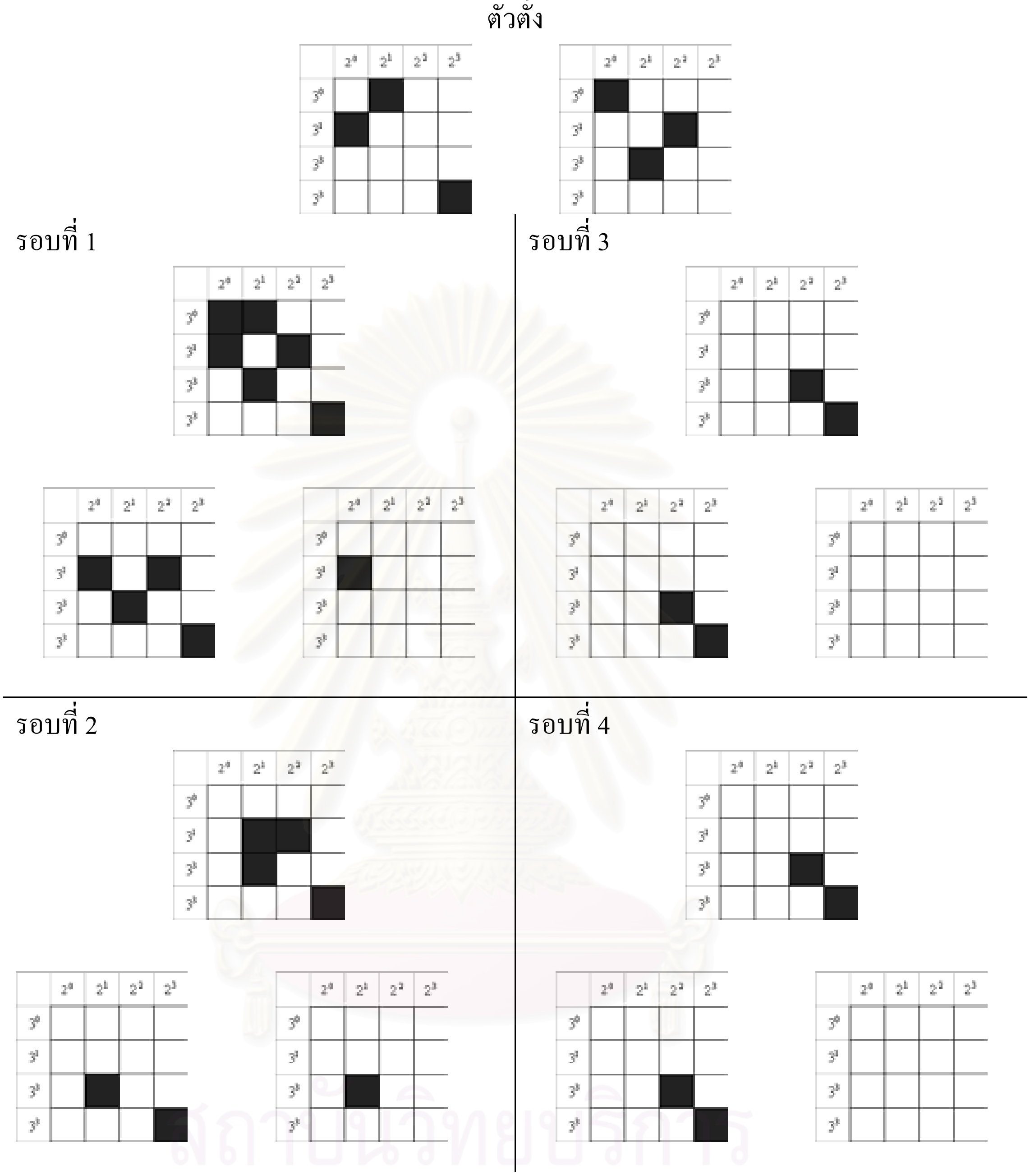
3) อัลกอริทึมนี้ทำงานทั้งหมด n รอบ และในแต่ละรอบนั้น ประกอบด้วยขั้นตอนย่อยสองขั้นตอน คือ การบวก และการแยกตาราง จากบทตั้งที่ 3.1 และทฤษฎีบทที่ 3.1 สรุปไว้ว่าแต่ละ

ขั้นตอนใช้เวลาคงที่ ทำให้เวลาที่ใช้ทั้งหมดของอัลกอริทึมนี้ ขึ้นอยู่กับค่าของ n หรือจำนวนแถวของรูปแบบแสดงค่าเพียงอย่างเดียวเท่านั้น ■

ตัวอย่างของการบวกเป็นไปดังรูปที่ 3.3 ซึ่งแสดงผลของ 221 และ 31 โดยแจกแจงผลของแต่ละขั้นตอน และผลที่ได้มีค่าเป็น 252

จากทฤษฎีบทที่ 3.2 จะพบว่ามีอัลกอริทึมที่สามารถบวกจำนวนในรูปแบบพร้อมบวกสองจำนวน ให้ผลลัพท์อยู่ในรูปพร้อมบวก และมีการทำงานเป็นเชิงกำหนด การบวกแบบนี้หลังจากผ่านไปช่วงเวลาหนึ่ง ซึ่งเท่ากับระยะเวลาในการรวมค่า และ การแยกตาราง จะได้คำตอบที่ถูกต้องของแถวที่มีค่าประจำแถวน้อยที่สุดออกมา ซึ่งจะนำไปใช้เป็นข้อมูลขาเข้าของการดำเนินการต่อไปได้ทันที

ข้อเสียของการบวกแบบนี้ คือ ยังไม่มีการจัดการกับส่วนล้นใด ๆ ทั้งสิ้น ถ้าเกิดส่วนล้นขึ้นการคำนวณจะผิดพลาด ซึ่งส่วนล้นนั้น อาจจะเป็นส่วนล้นจริง หรือส่วนล้นปลอม คือ ส่วนที่ล้นออกจากขนาดรูปแบบแสดงค่านั้น มีรูปแบบแสดงค่าอื่นอีกที่สามารถแสดงค่าที่ต้องการได้ นอกจากนี้ข้อเสียอีกข้อหนึ่งคือ การบวกนี้จะทำงานได้ถูกต้องกับระบบจำนวนฐานคู่ที่มีฐานเป็น 2 และ 3 เท่านั้น

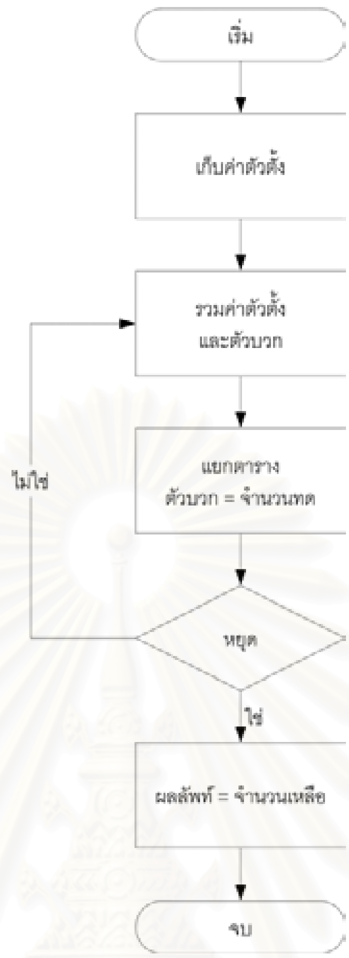


รูปที่ 3.3 ตัวอย่างการทำงานของอัลกอริทึมการบวก

การออกแบบวงจร

การบวกด้วยกระบวนการนี้จะมีการทำงานพร้อม ๆ กันในทุกตำแหน่งของรูปแบบแสดงค่า ดังนั้น จึงสามารถออกให้แบบแต่ละตำแหน่ง มีวงจรซึ่งคำนวณค่าของตำแหน่งตัวเองแยกจากกัน โดยจะมีการส่งข้อมูลระหว่างกัน เพื่อให้ทำงานได้ถูกต้อง และ วงจรเหล่านั้นจะเป็นวงจรที่ทำงานแบบประสานเวลา ซึ่งจะมีสัญญาณนาฬิกาภายนอกจากกำหนดจังหวะในการทำงานให้เป็นไปโดยพร้อมเพรียงกัน โดยขั้นตอนการทำงานคร่าว ๆ จะมีลักษณะดังรูปที่ 3.4

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 3.4 ลำดับการทำงานของการบวกโดยย่อ

รายละเอียดของแต่ละส่วนของการทำงานนั้นจะเป็นดังนี้

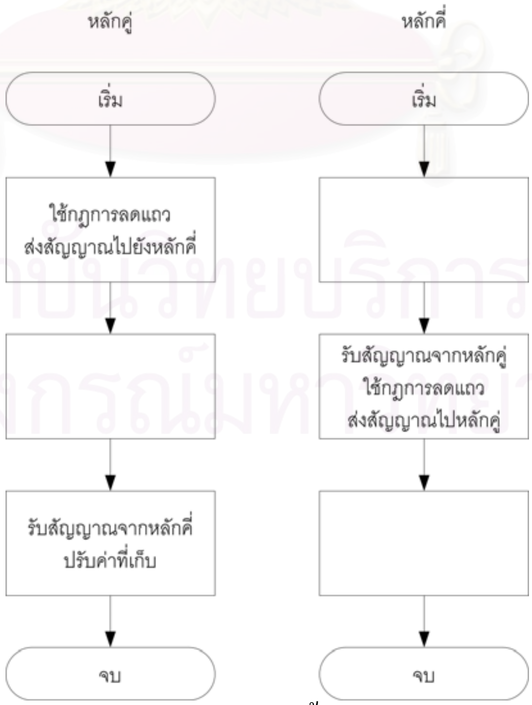
1. การออกแบบวงจรในการรวมค่า

ในขั้นตอนนี้จะนำค่าของตัวตั้งและตัวบวกของทุกตำแหน่งมารวมกัน โดยสามารถทำพร้อม ๆ กันในทุกตำแหน่งได้ทันที และหลังจากที่นำค่ามารวมกันแล้ว จะเกิดการตัวทดออกจากทุกตำแหน่ง ตัวทดนี้จะส่งไปยังหลักที่มีค่าประจำหลักมากกว่า ค่าของตัวทดนี้ จะถูกนำเข้าไปรวมกันค่าประจำตำแหน่งในขณะนั้น หลังจากการบวกครั้งที่สองนี้ จะไม่เกิดตัวทดขึ้นอีกเลย จากคุณสมบัติของจำนวนพร้อมบวก ที่มีสายการทดเพียงหนึ่งขั้นเท่านั้น สามารถเขียนรายละเอียดได้ตามรูปที่ 3.5

รูปที่ 3.5 การทำงานของขั้นตอนรวมค่า

2. การออกแบบวงจรในการแยกตาราง

สำหรับขั้นตอนนี้ จะต้องมีการแยกชนิดของวงจรเป็น 2 ชนิด ตามตำแหน่งของวงจร การที่ต้องมีสองชนิด เพราะจากอัลกอริทึมแยกตาราง มีการจัดการกับหลักคู่ และหลักคี่ต่างกัน โดยกฎการลดแถวจะถูกนำมาใช้กับหลักคู่ก่อน ซึ่งในขณะเดียวกัน หลักคี่จะยังไม่ดำเนินการใด ๆ การลดแถวนั้น จะต้องมีการส่งสัญญาณไปยังหลักคี่ด้วยในกรณีที่มีการส่งตัวทด ในขั้นตอนที่สอง หลักคู่จะไม่มีการทำงาน ส่วนหลักคี่จะถูกใช้ด้วยกฎการลดแถว ซึ่งผลจากขั้นนี้ จะต้องมีการส่งสัญญาณไปยังหลักคู่อีกครั้ง เพื่อให้ปรับค่าให้มีความถูกต้อง ลักษณะการทำงานเป็นไปดังรูปที่ 3.6



รูปที่ 3.6 การทำงานของขั้นตอนแยกตาราง

เมื่อวงจรทำงานทั้งสองขั้นย่อยเสร็จแล้ว จะมีการวนกลับไปทำซ้ำไปเรื่อย ๆ จนกว่าจะครบตามจำนวนแถว การตรวจสอบว่าครบจะหยุดเมื่อไรนั้น สามารถทำได้โดยใช้สัญญาณหยุด ที่จะมีการส่งต่อไปยังแถวถัดไปเมื่อทำงานครบหนึ่งรอบ โดยเมื่อเริ่มบวก สัญญาณนี้จะถูกส่งเข้าไปยังแถวที่มีค่าประจำแถวน้อยสุด ในรอบต่อมา สัญญาณนี้จะถูกส่งต่อไปยังแถวที่สอง และเมื่อแถวใดได้รับสัญญาณแล้ว จะทำงานรอบนั้นเป็นรอบสุดท้าย

การบวกนี้จะต้องใช้วงจรอีกวงจรหนึ่ง ซึ่งจะทำหน้าที่เก็บจำนวนทดออกมา แล้วส่งไปยังวงจรบวก เพื่อที่ทำหน้าที่เป็นตัวบวกในรอบถัดไป การทำงานมีเพียงแค่เก็บจำนวนทดที่วงจรบวกส่งออกมา แต่การเก็บค่านี้จะต้องสอดคล้องไปกับจังหวะของวงจรบวก

3. ข้อมูลที่จำเป็นในวงจรระดับตำแหน่ง

วงจรที่คำนวณค่าของแต่ละตำแหน่งนั้น จะต้องมีการเก็บค่า และติดต่อสื่อสารระหว่างกัน โดยข้อมูลที่วงจรจะต้องเก็บนั้น มีเพียงค่าของคำแหน่งนั้น ๆ เพียงอย่างเดียว ในส่วนข้อมูลที่จะต้องรับมานั้น ได้แก่

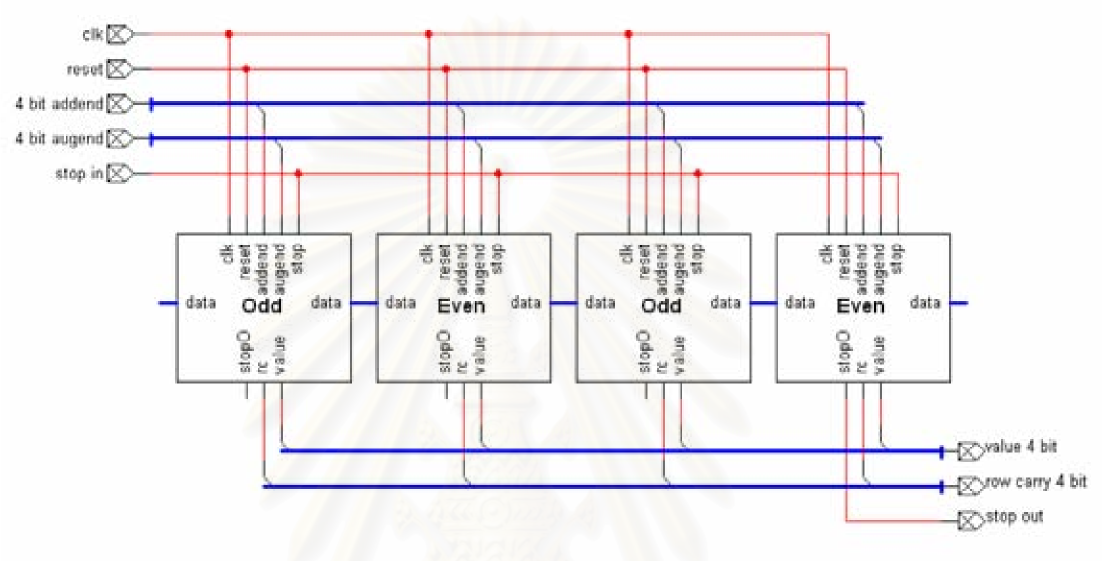
1. สัญญาณนาฬิกา รับมาจากสัญญาณนาฬิกากลาง
2. สัญญาณ reset เป็นการเริ่มต้นการบวก จะนำค่าของตัวตั้ง มาเก็บไว้ในวงจร
3. ค่าตัวตั้ง เป็นค่าตั้งต้นของวงจร
4. ค่าตัวบวก ขณะที่เริ่มบวก จะเป็นค่าของตัวบวกที่ใส่เข้ามาในวงจรบวก แต่ในรอบหลังจะเป็นค่าของจำนวนทด
5. ค่าตัวทด เป็นค่าของตัวทดจากการรวมค่าตัวตั้ง และ ตัวบวก จากหลักที่มีค่าประจำหลักน้อยกว่า
6. ค่าของหลักถัดไป ใช้ในการพิจารณาในขั้นการแยกตาราง
7. สัญญาณแจ้งการเกิดการลดแถวของหลักก่อนหน้า จะเกิดเมื่อหลักก่อนหน้าถูกยุบรวมจากกฎการลดแถว ซึ่งมีผลกระทบให้ค่าในหลักที่สนใจลดลงด้วย
8. สัญญาณหยุด เป็นสัญญาณส่งต่อเผื่อให้หยุดการทำงาน จะรับมาจากแถวก่อนหน้า

ส่วนข้อมูลที่จะต้องส่งออกไป ได้แก่

1. ค่าที่เก็บจะส่งไปให้หลักก่อนหน้า เพื่อพิจารณาในขั้นแยกตาราง และ ใช้แสดงผลลัพท์
2. ค่าตัวทด ส่งไปยังหลักถัดไป หลังจากรวมค่าของตัวตั้ง และตัวบวก
3. สัญญาณแจ้งการเกิดการลดแถว ส่งไปยังหลักถัดไป เพื่อให้ทำการปรับค่าให้ถูกต้องหลังจากที่มีการใช้กฎการลดแถวแล้ว นอกจากนี้จะส่งไปยังวงจรที่เก็บจำนวนทดด้วย โดยส่งไปยังแถวถัดไปของวงจรเก็บจำนวนทด
4. สัญญาณหยุด ส่งไปยังแถวถัดไป เพื่อให้หยุดทำงาน ซึ่งส่งหลังจากทำงานครบรอบแล้ว

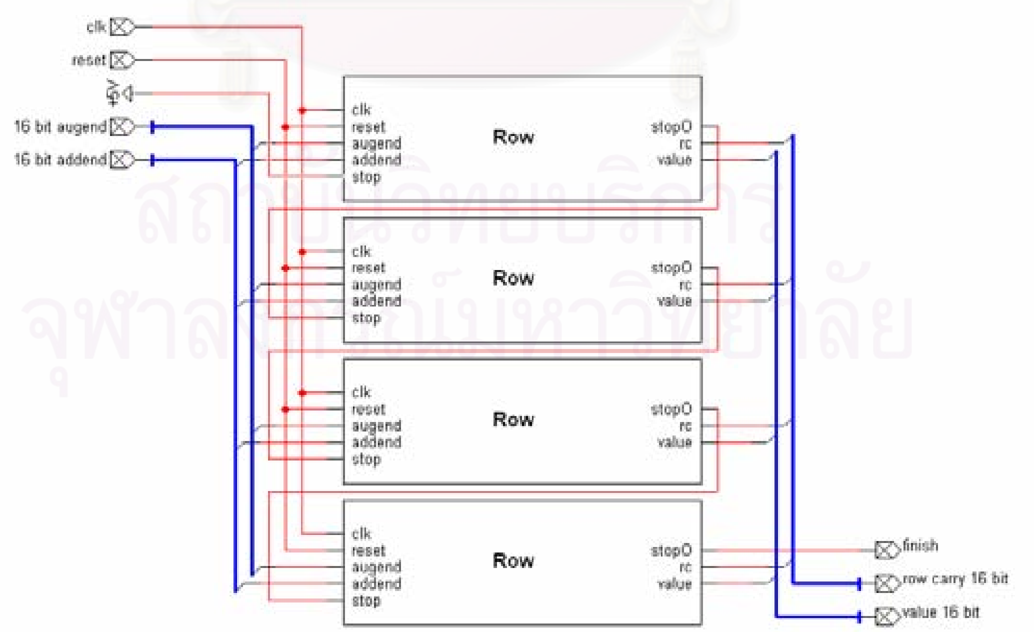
การทดลองจำลองวงจร

จากข้อแนะนำในการออกแบบวงจร จะทำการจำลองวงจรการบวกนี้โดยใช้ภาษา verilog วงจรจำลองนั้นจะบวกระบบจำนวนฐานคู่ที่รูปแบบแสดงค่ามีขนาดกว้าง 4 บิท 4 แถว โครงสร้างของวงจร ส่วนประกอบของวงจรนั้นเริ่มจากส่วนที่เล็กที่สุดเป็นวงจรบวกหนึ่งบิทของหลักคู่ และหลักคี่ชนิดละ 2 ตัวมาประกอบกัน เพื่อทำหน้าที่บวกแต่ละแถวของรูปแบบแสดงค่าการประกอบเป็นดังรูปที่ 3.7



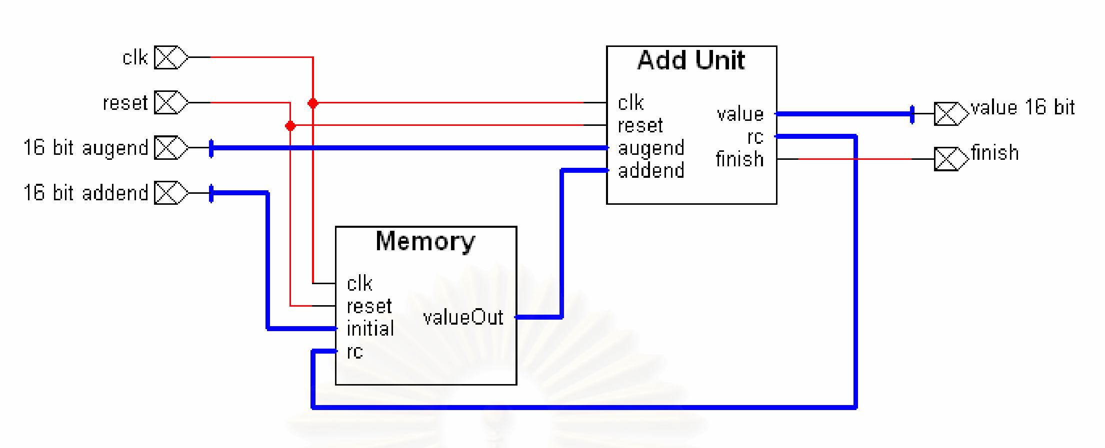
รูปที่ 3.7 การประกอบวงจรบวกสำหรับแต่ละแถว

และเมื่อได้วงจรสำหรับแต่ละแถวแล้วก็นำมาประกอบกันเป็น ส่วนที่ทำหน้าที่บวก ดังรูปที่ 3.8



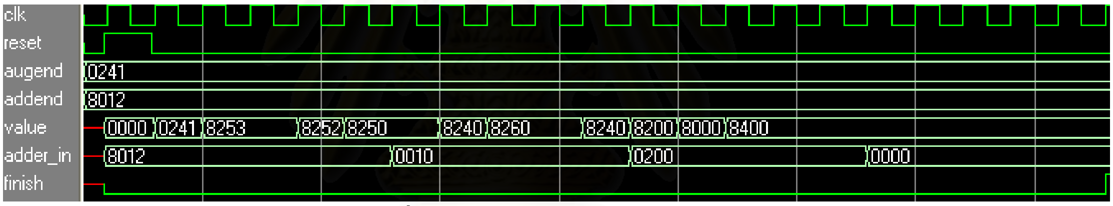
รูปที่ 3.8 การประกอบวงจรสำหรับส่วนทำหน้าที่บวก

วงจรบวกทั้งหมดนั้นจะประกอบด้วยส่วนที่ทำหน้าที่บวก และความจำที่ใช้ในการเก็บตัวบวก รวมทั้งจำนวนทด ซึ่งมีการประกอบกันดังรูปที่ 3.9



รูปที่ 3.9 การเชื่อมต่อระหว่างส่วนทำหน้าที่บวกและหน่วยความจำ

เมื่อสร้างวงจรโดยมีโครงสร้างดังนี้แล้วก็ทำการทดสอบวงจรด้วยการจำลองแบบจำลองความประพฤติ ซึ่งจะให้วงจรทำการบวกเลขสองจำนวนเข้าด้วยกัน การทดลองจำใช้ตัวตั้ง และตัวบวกเช่นเดียวกับรูปที่ 3.3 เพื่อใช้ในการพิจารณาความถูกต้อง โดยผลที่ได้เป็นดังรูปที่ 3.10



รูปที่ 3.10 ผลการบวก 221 และ 31

จากรูปมีข้อมูลขาเข้าสี่ชนิด คือ clk, reset, augend และ addend โดยที่ augend และ addend เขียนอยู่ในรูปของเลขฐานสิบหก โดยที่เลขฐานสิบหกหนึ่งตัวมีค่าเท่ากับแถวหนึ่งแถวของรูปแบบแสดงผล หลักที่มีค่าประจำหลักสูง จะแทนแถวที่มีค่าประจำแถวสูง ในส่วนของ value เป็นค่าที่ออกมาจากส่วนทำหน้าที่บวก และ value เป็นค่าของความจำ ซึ่งก็คือ ตัวบวก และ จำนวนทดนั่นเอง

จะเห็นว่าผลจากการจำลองวงจรนั้นเป็นไปตามขั้นตอนการทำงานของอัลกอริทึมการบวกทุกประการ ซึ่งสรุปได้ว่า อัลกอริทึมนี้สามารถนำมาสร้างเป็นวงจรได้จริง สำหรับเวลาที่ใช้ตั้งแต่เริ่มต้นการบวก จนได้ผลลัพท์ที่สมบูรณ์นั้น จะใช้เวลาทั้งสิ้น 21 รอบสัญญาณนาฬิกา หรือเขียนให้อยู่ในรูปของจำนวนแถว n ได้เป็น $5n+1$ รอบ

บทที่ 4

อัลกอริทึมการบวกเชิงกำหนดสำหรับจำนวนฐานคู่ทั่วไป

ในบทที่แล้ว ได้กล่าวถึงอัลกอริทึมเชิงกำหนดสำหรับจำนวนฐานคู่อย่างละเอียดแล้ว แต่อัลกอริทึมนั้นสามารถทำการบวกได้ก็ต่อเมื่อเป็นจำนวนในระบบจำนวนฐานคู่ที่มีฐานเป็น สอง และ สาม เท่านั้น ไม่สามารถใช้กับฐานอื่นได้ รวมทั้งข้อมูลขาเข้า จะต้องอยู่ในรูปพร้อมบวก แต่บางกรณี อาจจะมีความต้องการใช้ระบบจำนวนฐานคู่ ซึ่งมีฐานเป็นจำนวนเต็มอื่น ที่ไม่ใช่สองและสาม ซึ่งจะมีสมบัติที่แตกต่างออกไป ดังนั้น ในบทนี้จะนำเสนออัลกอริทึมสำหรับจำนวนฐานคู่ซึ่งมีฐานอื่น

ความสำคัญของฐาน

ระบบจำนวนฐานคู่นั้น เป็นระบบจำนวนที่สมบูรณ์ และมีความซ้ำซ้อน โดยปัจจัยที่ส่งผลต่อสมบัติต่าง ๆ ได้แก่ ชุดตัวเลข และฐานทั้งสองจำนวน โดยจะเรียกว่า ฐาน α และฐาน β การเปลี่ยนแปลงปัจจัยเหล่านี้จะส่งผลต่อระบบจำนวนอย่างมาก เมื่อทำการพิจารณาชุดตัวเลขนั้นพบว่าประกอบด้วยจำนวนสองจำนวน คือ 0 และ 1 ซึ่งใช้ที่เก็บเพียงแค่ 1 บิท ซึ่งเป็นคุณสมบัติที่ต้องการคงไว้ และการที่จะคงสมบัติข้อนี้ พร้อมทั้งยังให้ระบบจำนวนคงความสมบูรณ์ไว้ด้วย ก็จำเป็นต้องบังคับให้ฐาน α มีค่าเป็น 2 เสมอ ซึ่งส่งผลให้เลขฐานสองเป็นส่วนหนึ่งของระบบจำนวนนี้ ดังนั้นจะเหลือค่าที่จะปรับได้อยู่ค่าเดียว คือ ฐาน β

นิยาม ระบบจำนวนฐานคู่ $(2,\beta)$ *คือระบบจำนวนฐานคู่ที่มีฐานเป็น* 2 *และ* β *ให้* x *เป็นจำนวนในระบบจำนวนฐานคู่* $(2,\beta)$ *ค่าเชิงเลขคณิตของ* x *จะเป็น*

$$\|x\| = \sum_{i,j} d_{i,j} 2^i \beta^j$$

สำหรับฐาน β *นั้นสามารถเขียนอยู่ในรูปของเลขฐานสองได้ดังนี้*

$$\beta = 2^{i_0} + 2^{i_1} + \ldots + 2^{i_l};\ i_k \in N \cup \{0\} \wedge i_0 < i_1 < \ldots < i_l$$

จะเรียก L *ว่าเป็นความยาวของฐาน* β *ซึ่งมีค่าเป็น* $i_l - i_0 + 1$

ถ้าคิดระดับความซ้ำซ้อนของระบบจำนวนจากอัตราส่วนของค่าที่แสดงได้ต่อจำนวนรูปแบบทั้งหมด การเปลี่ยนฐาน β นี้จะมีผลโดยตรงต่อระดับความซ้ำซ้อนของระบบจำนวน เนื่องจากการเปลี่ยนฐานนี้จะทำให้ขอบเขตของค่าที่สามารถแสดงได้นั้นเปลี่ยนแปลงไป ในขณะที่จำนวนรูปแบบแสดงค่านั้นเท่าเดิม เมื่อรูปแบบแสดงค่าที่มีขนาด $m \times n$ จะมีรูปแบบแสดงค่าที่สามารถแสดงได้ทั้งหมด 2^{mn} แบบ ส่วนค่าสูงสุดที่จะแสดงได้นั้น สามารถหาได้จากการรวมค่า

ประจำตำแหน่งทุกตำแหน่งเข้าด้วยกัน ถ้าระบบจำนวนมีฐานเป็น 2 และ β สามารถเขียนได้ดังสมการ

$$\max = \sum_{i=0,\, j=0}^{m-1,n-1} 2^i \beta^j$$
$$= \sum_{i=0}^{m-1} 2^i \sum_{j=0}^{n-1} \beta^j$$
$$= \frac{\left(2^n - 1\right)\left(\beta^m - 1\right)}{\beta - 1}$$

อัตราระดับความซ้ำซ้อนของระบบจำนวนมีค่าเป็น $\left(2^n - 1\right)\left(\beta^m - 1\right) / 2^{mn}\left(\beta - 1\right)$ ซึ่งสรุปได้ว่าระดับความซ้ำซ้อนนั้น แปรผันกับค่าของฐาน ดังนั้น ในสถานการณ์ที่พื้นที่ในการเก็บข้อมูลมีน้อยหรือมีค่าใช้จ่ายสูง ก็ควรจะใช้ฐานที่มีค่ามากขึ้น เพื่อที่จะแสดงข้อมูลได้มากขึ้น

นอกจากนี้การที่ฐาน β เป็นจำนวนคู่ หรือ คี่ก็มีผลกระทบต่อระบบจำนวนเช่นกัน โดยถ้าฐานทั้งสองตัวเป็นจำนวนคู่จะทำให้ความซ้ำซ้อนของการแสดงเลขคี่ลดลง เนื่องจากทุกรูปแบบแสดงค่าของจำนวนคี่จะถูกบังคับให้มีตำแหน่ง $2^0\beta^0$ เป็น 1 เสมอ และการที่ฐานเป็นเลขคู่นั้นยังจะก่อให้เกิดปัญหากับกฎการลดแถว ซึ่งจะอธิบายในรายละเอียดต่อไป

ความแตกต่างในการบวกจำนวนฐานคู่ (2,β) และ (2,3)

การเปลี่ยนฐานนอกจากทำให้สมบัติของระบบจำนวนเปลี่ยนแปลงแล้ว ยังทำให้ลักษณะของปัญหาในการบวกเปลี่ยนแปลงไปด้วย ซึ่งความแตกต่างที่มีผลต่อการบวกมีดังนี้

1. กฎการลดแถว

กฎการลดแถวนั้นสามารถมองว่าเป็นการสร้างตัวทดอย่างหนึ่ง โดยที่จะนำค่าจากหลาย ๆ ตำแหน่งในแถวเดียวกัน เปลี่ยนเป็นค่าในอีกตำแหน่งหนึ่ง ซึ่งอยู่ในแถวที่มีค่าประจำแถวมากกว่า โดยจะกำหนดให้กฎการลดแถวเป็นดังนี้

กฎ กฎการลดแถวสำหรับระบบจำนวนฐานคู่ $(2,\beta)$ เป็น

$$2^i\beta^{j+1} = 2^{i+i_0}\beta^j + 2^{i+i_1}\beta^j + \ldots + 2^{i+i_l}\beta^j$$

ตัวอย่างของกฎการลดแถว เช่น

ฐาน $(2,3)$ กฎการลดแถวเป็น $2^i3^{j+1} = 2^i3^j + 2^{i+1}3^j$

ฐาน $(2,7)$ กฎการลดแถวเป็น $2^i7^{j+1} = 2^i7^j + 2^{i+1}7^j + 2^{i+2}7^j$

ฐาน $(2,14)$ กฎการลดแถวเป็น $2^i14^{j+1} = 2^{i+1}14^j + 2^{i+2}14^j + 2^{i+3}14^j$

ฐาน $(2,19)$ กฎการลดแถวเป็น $2^i19^{j+1} = 2^i19^j + 2^{i+1}19^j + 2^{i+4}19^j$

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

กฎการลดแถวนั้นมีข้อสังเกต คือเมื่อระบบจำนวนที่มีฐาน β เป็นเลขคู่นั้น จะทำให้ผลที่เกิดจากการยุบรวมอยู่ในตำแหน่งที่ติดลบ เช่น ระบบจำนวนฐาน $(2,14)$ จะเกิดกรณี $2^{-1}14^{1} = 2^{0}14^{0} + 2^{1}14^{0} + 2^{2}14^{0}$ กรณีเช่นนี้สามารถแก้ไขได้ แต่จะต้องมีการออกแบบกระบวกการพิเศษที่จะจัดการกับตำแหน่งเหล่านี้ แยกจากตำแหน่งอื่น ๆ ทำให้มีความซับซ้อนมากขึ้น ดังนั้นฐาน β จึงควรมีค่าเป็นจำนวนคี่เท่านั้น

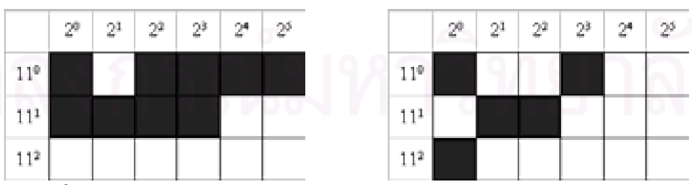
นอกจากนี้ กฎการลดแถวยังกระทบต่อส่วนประกอบที่สำคัญอีกส่วนหนึ่งของกระบวกการบวกนั้นคือ รูปแบบพร้อมบวก ซึ่งจะแตกต่างกันออกไปตามฐาน β

2. รูปแบบพร้อมบวก

ในระบบจำนวนฐานคู่ที่มีฐานเป็น $(2,3)$ นั้น จะมีรูปแบบพร้อมบวก เป็นรูปแบบที่ไม่มีตำแหน่งที่ติดกันในแถวเดียวกันมีค่าเป็น 1 พร้อมกัน โดยจำนวนนี้เกิดจากการเรียกใช้กฎการลดแถวกับทุก ๆ ตำแหน่งของรูปแบบแสดงผล จนไม่มีตำแหน่งใดตรงกับข้อกำหนดของกฎการลดแถวเหลืออยู่เลย ดังนั้นจึงนิยามของรูปพร้อมบวกเป็นดังนี้

นิยาม *กำหนดให้ระบบจำนวนฐานคู่ $(2,\beta)$ โดยที่ $\beta = 2^{i_0} + 2^{i_1} + \ldots + 2^{i_l}$ เมื่อ L คือความยาวในรูปฐานสองของ β* รูปพร้อมบวกสำหรับจำนวนฐานคู่ $(2,\beta)$ *คือ รูปแบบแสดงค่าที่มีตำแหน่งที่มีค่าไม่ตรงกับตำแหน่ง $2^{i+i_0}\beta^{j}, 2^{i+i_1}\beta^{j}, \ldots, 2^{i+i_l}\beta^{j}$ พร้อมกันทั้งหมด*

ตัวอย่างของจำนวนที่อยู่และไม่อยู่ในรูปพร้อมบวกนั้นแสดงในรูปที่ 4.1 ซึ่งเป็นรูปแบบแสดงค่าของ 196 ในระบบจำนวนฐานคู่ $(2,11)$ รูปพร้อมบวกจะต้องไม่มีข้อมูลที่มีค่าตรงกับตำแหน่ง $2^{i+0}11^{j}, 2^{i+2}11^{j}, 2^{i+3}11^{j}$ พร้อมกันทั้งหมด โดยรูปทางซ้ายนั้น ในแถวที่ 0 มีอยู่สองชุดที่ตรงกับตำแหน่งที่กำหนด ซึ่งได้แก่ $2^{0}11^{0}, 2^{2}11^{0}, 2^{3}11^{0}$ และ $2^{2}11^{0}, 2^{4}11^{0}, 2^{5}11^{0}$ นอกจากนี้แถวที่ 1 ยังมีข้อมูลตามตำแหน่ง $2^{0}11^{1}, 2^{2}11^{1}, 2^{3}11^{1}$ ตรงกับข้อกำหนดด้วย ในขณะที่รูปทางขวาไม่มีส่วนใดของตารางที่ตรงกับข้อกำหนดนั้น



รูปที่ 4.1 ตัวอย่างของจำนวนในรูปปกติ และรูปพร้อมบวก

จากนิยามนี้พบว่ารูปแบบพร้อมบวกนั้นจะมีลักษณะที่แตกต่างกันไป ตามฐาน β แต่รูปแบบพร้อมบวกทั้งหมดนั้น จะมีลักษณะร่วมกันอยู่หนึ่งลักษณะคือ จะเป็นรูปแบบที่ไม่มีตำแหน่งที่มีค่าติดกันตั้งแต่ $L-1$ ตัวขึ้นไป เช่น เมื่อ $\beta = 2^{0} + 2^{1}$ ผลจะไม่มีตำแหน่งที่มีค่าติดกันตั้งแต่ 1 ตัว หรือเมื่อ $\beta = 2^{0} + 2^{1} + 2^{2}$ ผลจะไม่มีตำแหน่งที่มีค่าติดกันตั้งแต่ 2 ตัว

ฐาน β บางลักษณะจะทำให้เกิดรูปแบบพร้อมบวกที่มีลักษณะเฉพาะ เช่น β ที่เขียนอยู่ในรูป $2^{i_0}+2^{i_l}$ นั้น จะทำให้เกิดรูปแบบพร้อมบวกที่สามารถจำกัดตัวทดได้ โดยสายการทดจะไม่เกิน $L-1$ ขั้น ซึ่งสามารถพิสูจน์ได้โดยการพิสูจน์ทางตรง แต่ฐานบางฐานไม่สามารถทำให้เกิดรูปแบบพร้อมบวกที่จะจำกัดสายการทดได้ เช่น β ที่อยู่ในรูป $2^0+2^1+2^2+\ldots+2^{i_l}$ รูปแบบพร้อมบวกจะเป็นรูปแบบที่ไม่มีตำแหน่งที่มีค่าติดกัน i_l เท่านั้น ซึ่งเห็นได้ชัดว่าสามารถเกิดสายการทดที่มีขนาดเป็นอนันต์ได้

จากปัญหาของรูปแบบพร้อมบวกที่ไม่สามารถจำกัดสายการทดได้นี้ จึงต้องทำการออกแบบอัลกอริทึมสำหรับการบวกใหม่ โดยที่จะต้องไม่พึ่งคุณสมบัติของรูปแบบพร้อมบวก แต่จะต้องทำการบวกจำนวนที่อยู่ในรูปแบบแสดงค่าใด ๆ ก็ได้ เพื่อให้สามารถใช้งานได้กับระบบจำนวนฐานคู่ทั่วไป

ส่วนประกอบของอัลกอริทึมการบวกจำนวนฐานคู่ทั่วไป

จากปัญหาที่พบทั้งเชิงความซ้ำซ้อนของระบบจำนวน หรือ ปัญหาจากกฎการลดแถวของเมื่อระบบจำนวนฐานคู่ที่มีฐาน β เป็นเลขคู่ ดังนั้น จึงจะทำการออกแบบการบวกเฉพาะระบบจำนวนฐานคู่ที่มีฐาน β เป็นเลขคี่เท่านั้น และจากลักษณะของที่แตกต่างไปจากจำนวนฐานคู่ปกติ จึงต้องหากระบวนการในการบวกใหม่ ที่จะสามารถทำงานได้ในเวลาที่เป็นสัดส่วนโดยตรงกับจำนวนแถวของรูปแบบแสดงค่า อัลกอริทึมใหม่นี้จะยังมีลักษณะโดยรวมเหมือนกับอัลกอริทึมเดิม คือจะมีการทำงานเป็นรอบ ๆ แต่ละรอบจะแบ่งการทำงานออกเป็นสองส่วนใหญ่ ส่วนแรกนำค่าของตัวตั้ง และ ตัวบวกมารวมกัน ผลจากการทำงานส่วนนี้จะถูกเรียกว่า ผลบวกชั่วคราว ในส่วนที่สอง จะทำการแยกค่าของผลบวกชั่วคราวออกเป็นสองส่วน โดยมีการใช้กฎการลดแถวเป็นเครื่องมือสำคัญ

1. กระบวนการรวมค่า

การทำงานในส่วนนี้จะต้องทำให้ได้ในเวลาคงที่ แต่จำนวนที่จะนำมาบวกนั้น เกิดสายการทดได้ไม่จำกัด จึงนำเทคนิคของระบบจำนวนลูกผสมมาประยุกต์ใช้ ในที่นี้ข้อมูลที่จะนำมาบวกกันมีชุดตัวเลขเป็น $\{0,1\}$ และจะใช้ผลบวกชั่วคราวนี้มีชุดตัวเลขในบางหลักเป็น $\{0,1,2\}$ เพื่อจำกัดสายการทด โดยที่ผลบวกชั่วคราวนั้นจะสามารถแยกเป็นส่วนย่อยหลาย ๆ ส่วน โดยแต่ละส่วนมีนิยามดังนี้

นิยาม *กำหนดให้ระบบจำนวนฐานคู่* $(2,\beta)$ *โดยที่* $\beta = 2^0 + 2^{i_1} + \ldots + 2^{i_l}$ *เมื่อ* L *คือความยาวในรูปฐานสองของ* β ส่วนย่อยที่ x, y *หรือ* $part(x,y)$ *คือส่วนหนึ่งของรูปแบบแสดงค่าขนาด* L *ตำแหน่งซึ่งเขียนได้ดังนี้*

$$part(x,y) = d_{Lx,y} d_{Lx+1,y} d_{Lx+2,y} \ldots d_{Lx+L-1,y}$$

$$\text{เมื่อ } d_{Lx+k,y} \in \begin{cases} \{0,1\}; k \neq 0 \\ \{0,1,2\}; k = 0 \end{cases}$$

ค่าเชิงตัวเลขของ $part(x,y)$ *มีค่าเป็น*

$$\left\| part(x,y) \right\| = \sum_{k=0}^{L-1} d_{Lx+k,y} 2^{Lx+k} \beta^y$$

นิยาม *กำหนดให้ระบบจำนวนฐานคู่* $(2,\beta)$ *โดยที่* $\beta = 2^0 + 2^{i_1} + \ldots + 2^{i_l}$ *เมื่อ* L *คือความยาวในรูปฐานสองของ* β ตำแหน่งที่มีค่าประจำตำแหน่งน้อยเป็นลำดับที่ z ของส่วนย่อยที่ x, y หรือ $pos(x,y,z)$ *คือค่าที่อยู่ในตำแหน่งที่มีค่าประจำตำแหน่งเป็น* $2^{Lx+z} \beta^y$ *ค่าเชิงตัวเลขของ* $pos(x,y,z)$ *เท่ากับ* $d_{Lx+z} 2^{Lx+z} \beta^y$

นิยาม *กำหนดให้ระบบจำนวนฐานคู่* $(2,\beta)$ *โดยที่* $\beta = 2^0 + 2^{i_1} + \ldots + 2^{i_l}$ *เมื่อ* L *คือความยาวในรูปฐานสองของ* β ส่วนย่อยที่ x, y ที่ถูกเลื่อนไป z ตำแหน่ง หรือ $part(x,y,z)$ *คือส่วนหนึ่งของรูปแบบแสดงค่าขนาด* L *ตำแหน่งซึ่งเขียนได้ดังนี้*

$$part(x,y,z) = d_{Lx+z,y} d_{Lx+z+1,y} d_{Lx+z+2,y} \ldots d_{Lx+z+L-1,y}$$

$$\text{เมื่อ } d_{Lx+z+k,y} \in \begin{cases} \{0,1\}; k \neq 0 \\ \{0,1,2\}; k = 0 \end{cases}$$

ค่าเชิงตัวเลขของ $part(x,y,z)$ *มีค่าเป็น*

$$\left\| part(x,y,z) \right\| = \sum_{k=0}^{L-1} d_{Lx+z+k,y} 2^{Lx+z+k} \beta^y$$

ตัวอย่างของการเรียกส่วนย่อยแต่ละส่วนเป็นไปดังรูปที่ 4.2

<table>
<tr><td></td><td>2^0</td><td>2^1</td><td>2^2</td><td>2^3</td><td>2^4</td><td>2^5</td></tr>
<tr><td>7^0</td><td colspan="3">$part(0,0)$</td><td colspan="3">$part(0,1)$</td></tr>
<tr><td>7^1</td><td>$pos(0,1,0)$</td><td></td><td></td><td></td><td>$pos(1,1,1)$</td><td></td></tr>
<tr><td>7^2</td><td></td><td colspan="3">$part(0,2,1)$</td><td></td><td></td></tr>
</table>

รูปที่ 4.2 ตัวอย่างการเรียกส่วนย่อยแบบต่าง ๆ

สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

นิยาม ผลชั่วคราว T *หมายถึงจำนวนในระบบจำนวนฐานคู่ลูกผสม โดยประกอบขึ้นจากส่วนย่อยรวมกัน ผลชั่วคราว* T *สามารถเขียนได้ดังนี้*

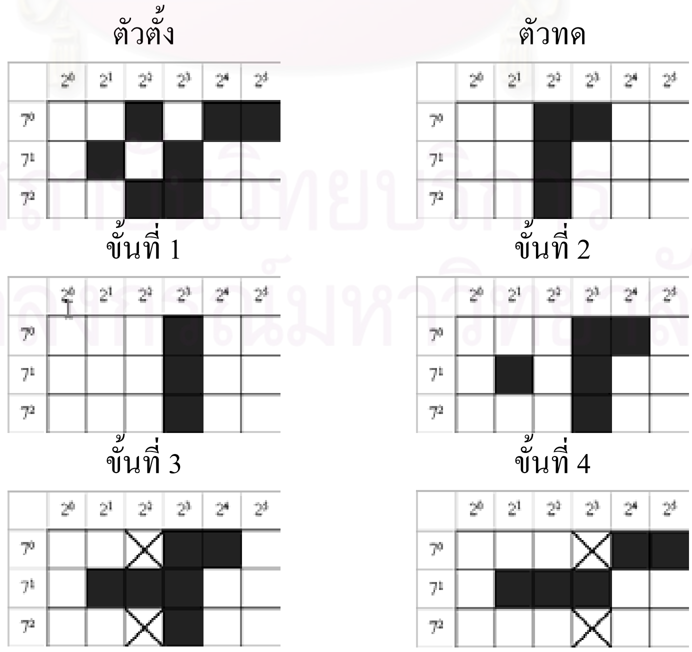
$$T = \begin{array}{cccc} part(0,0) & part(1,0) & \ldots & part(x,0) \\ part(0,1) & part(1,1) & \ldots & part(x,1) \\ \ldots & \ldots & & \ldots \\ part(0,y) & part(1,y) & \ldots & part(x,y) \end{array}$$

ค่าเชิงตัวเลขของ T *มีค่าเป็น*

$$\|T\| = \sum_{x,y} \|part(x,y)\|$$

สำหรับกระบวนการในการรวมค่านี้ทำเหมือนกับการบวกโดยทั่วไป แต่จะบวกส่วนย่อยทุกส่วนย่อยพร้อม ๆ กัน จากตำแหน่งที่มีค่าประจำตำแหน่งน้อยที่สุด ไล่ไปยังค่าประจำตำแหน่งมากที่สุด ตัวทดที่เกิดจากการบวกนั้นจะส่งต่อไปเรื่อย ๆ และเมื่อบวกครบทุกตำแหน่งในส่วนย่อยแล้ว จะมีตัวทดจาก $pos(x,y,L-1)$ ไปยัง $pos(x,y+1,0)$ และตำแหน่ง $pos(x,y+1,0)$ ก็จะเก็บค่าไว้โดยที่ไม่ส่งต่อ ทำให้การทำงานในส่วนของการรวมค่านั้นสามารถทำได้ในเวลาคงที่ในฐาน β หนึ่ง ๆ ไม่ขึ้นกับขนาดของจำนวน

ตัวอย่างของการรวมค่านั้นเป็นไปดังรูปที่ 4.3 โดยในรูปนั้นเป็นจำนวนในระบบจำนวนฐานคู่ $(2,7)$ ตัวตั้งมีค่า 710 และตัวบวกที่มีค่า 236 ในขั้นแรกจะทำการบวก $pos(x,y,0)$ แล้วบวก $pos(x,y,1)$ และ $pos(x,y,2)$ ตามลำดับ จากรูปตำแหน่งที่มีเครื่องหมายกากบาทคือตำแหน่งที่จะเกิดการทดไปยังตำแหน่งถัดไป และขั้นสุดท้าย จะนำตัวทดจาก $pos(x,y,2)$ ไปเก็บไว้ใน $pos(x+1,y,0)$ ผลที่ได้เป็นจำนวนลูกผสม ที่มีค่า 946



รูปที่ 4.3 ขั้นตอนการรวมค่า

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

2. กระบวนการแยกตาราง

ผลบวกชั่วคราวจากกระบวกการรวมค่านั้นจะอยู่ในรูปของจำนวนฐานคู่ลูกผสม ซึ่งไม่อยู่ในรูปแบบที่ต้องการ ดังนั้นจะต้องมีกระบวนการในการเปลี่ยนรูป ให้ผลอยู่ในรูปจำนวนฐานคู่ทั่วไป ซึ่งมีชุดตัวเลขเป็น {0,1} เท่านั้น การเปลี่ยนรูปนี้สรุปง่าย ๆ ก็คือการส่งผ่านตัวทดที่ได้เก็บไว้ให้หมด แต่การกระทำนั้นจะไม่สามารถจำกัดเวลาได้ ดังนั้นจึงต้องนำกฎการลดแถวเข้ามาช่วย โดยที่ก่อนที่จะมีการส่งต่อตัวทดนั้น ถ้าส่วนย่อยใดมีลักษณะตรงกับข้อกำหนดของกฎการลดแถว ก็จะถูกเปลี่ยนรูปตามกฎการลดแถวนั้น และส่งค่านี้ไปเป็นตัวทดระหว่างแถวต่อไป การกระทำนี้จะทำให้การเปลี่ยนรูปทำได้ในเวลาคงที่ ซึ่งจะแสดงวิธีการพิสูจน์ในรายละเอียดในส่วนต่อไป นอกจากนี้ยังทำให้จำนวนของตำแหน่งที่มีค่าลดน้อยลงด้วย

ทฤษฎีบทที่ 4.1 *ให้ A เป็นผลบวกชั่วคราวที่มีฐาน $(2,\beta)$ A จะสามารถถูกแบ่งด้วยอัลกอริทึมแยกตารางทั่วไปเป็นสองจำนวน เรียกว่า จำนวนเหลือ A' และ จำนวนทด R โดยที่จำนวนทั้งสองเป็นรูปแบบแสดงค่าของระบบจำนวนฐานคู่ $(2,\beta)$ ภายในเวลาคงที่*

อัลกอริทึม 3 อัลกอริทึมแยกตารางทั่วไป

Input: $A=\sum_{x,k,j} a_{Lx+k,j} 2^{Lx+k} \beta^j$

Output: $A'=\sum_{x,k,j} a'_{Lx+k,j} 2^{Lx+k} \beta^j$, $R=\sum_{x,k,j} r_{Lx+k,j} 2^{Lx+k} \beta^j$ โดยที่ $A=A'+R$

Define: $\beta = 2^{i_0}+2^{i_1}+\ldots+2^{i_l}$, $L=i_l-i_0+1$

ขั้นตอน

$z=0$

while $(z<L)$

do if $part(x,y,z)$ match row reduction rule

do $pos(x,y,i)=pos(x,y,i)-1$

$r_{Lx+z,y+1}=1$

if $pos(x,y,z)=2$

do $pos(x,y,z)=0$

$pos(x,y,z+1)=pos(x,y,z+1)+1$

$z=z+1$

$A'=A$

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

พิสูจน์ จะต้องพิสูจน์ว่า ผลจากอัลกอริทึมนี้จะมีชุดตัวเลขเป็น $\{0,1\}$, ผลรวมของคำตอบมีค่าเท่ากับตัวตั้ง และ อัลกอริทึมนี้สามารถทำงานได้ในเวลาคงที่

1) ผลจากอัลกอริทึมนี้แบ่งเป็นสองตัว คือ A' และ R สำหรับ A' จะแบ่งเป็น 3 กรณีตามส่วนย่อย

กรณี 1 ไม่มีจำนวนใดในส่วนย่อยเป็น 2 เลย ผลที่ได้ก็จะไม่มี 2 แน่นอน

กรณี 2 ในส่วนย่อย มี 2 และ มีบางตัวในส่วนย่อยมีค่าเป็น 0 จะเห็นว่าอัลกอริทึมนี้ทำงานทั้งหมด L รอบ เท่ากับความยาวส่วนย่อย ทำให้ค่า 2 ที่เก็บไว้สามารถจะทดไปจนถึงตำแหน่งที่มีค่าเป็น 0 ในส่วนย่อยนั้น และทำให้ตำแหน่งนั้นมีค่าเป็น 1 แทน ผลจากกรณีนี้จึงไม่มีค่าเป็น 2 เหลืออยู่

กรณี 3 ในส่วนย่อยมี 2 และ ทุกตัวในส่วนย่อยมีค่าเป็น 1 ทั้งหมด กรณีนี้จะเห็นว่า ส่วนย่อยนั้นจะไปตรงกับกฎการลดแถวแน่นอน และเมื่อใช้กฎการลดแถวแล้ว จะทำให้เกิดตำแหน่งที่มีค่าเป็น 0 ขึ้น อย่างน้อยหนึ่งตำแหน่ง ทำให้ค่าของ 2 นั้นสามารถทดไปจนถึงตำแหน่งนั้นได้

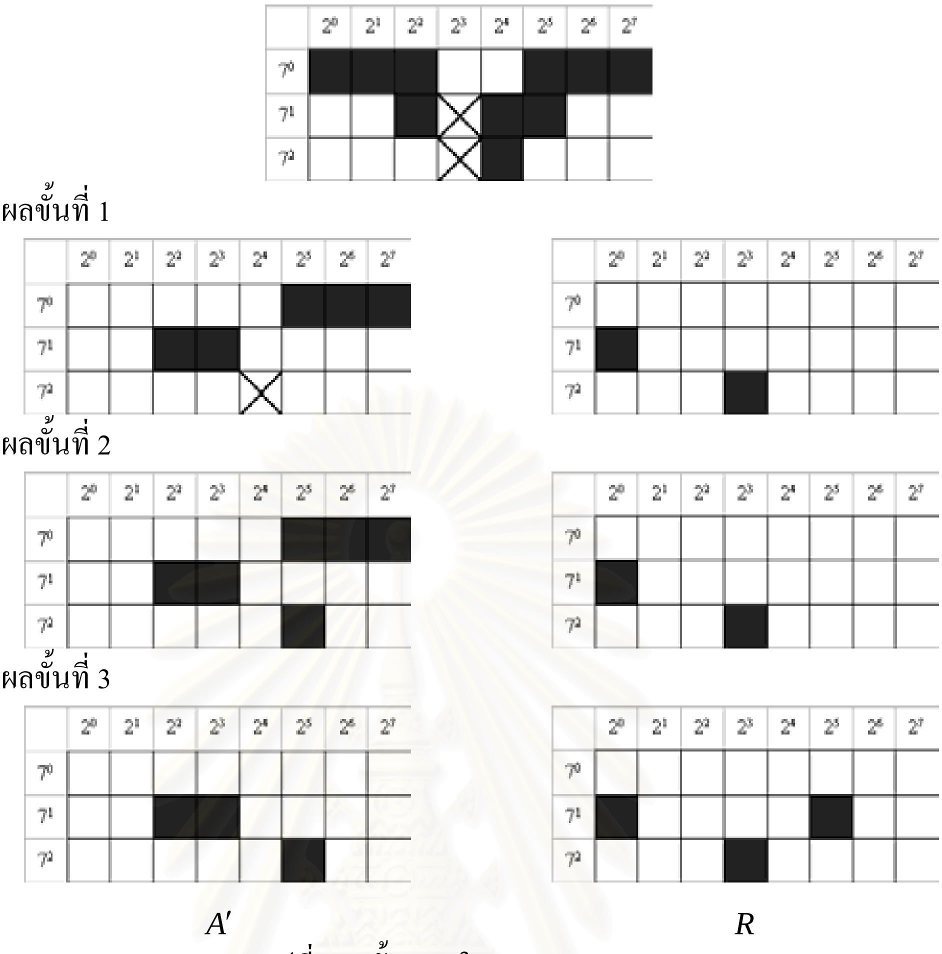
จากทั้ง 3 กรณีจะสรุปได้ว่า A' จะไม่มีตำแหน่งใดที่มีค่าเป็น 2 เหลืออยู่เลย

ในส่วนของ R นั้น ค่าเริ่มต้นมีค่าเป็น 0 ทั้งหมด และการกระทำที่เป็นการกำหนดค่าให้ R ทุกทีจะปรับค่าในตำแหน่งนั้น ๆ ของ R จาก 0 เป็น 1 ทั้งสิ้น ทำให้สรุปได้ว่าผลจากอัลกอริทึมนี้จะมีชุดตัวเลขเป็น $\{0,1\}$ เท่านั้น

2) อัลกอริทึมนี้สร้างขึ้นจากกฎการลดแถว และการสร้างตัวทด ซึ่งมีค่าของข้อมูลขาเข้า และขาออกเท่ากัน ดังนั้น อัลกอริทึมนี้จึงไม่มีการเปลี่ยนแปลงค่าของข้อมูลขาเข้า เพราะฉะนั้น $A = A' + R$

3) สำหรับฐาน β หนึ่ง ๆ อัลกอริทึมนี้จะทำงานเป็นจำนวน L รอบเสมอ และในแต่ละรอบนั้นประกอบด้วยการใช้กฎการลดแถว ซึ่งใช้เวลาคงที่ และการทด 1 ขั้น ซึ่งใช้เวลาคงที่เช่นกัน ดังนั้นเวลาในแต่ละรอบจึงคงที่ ทำให้ระยะเวลาทั้งหมดที่ใช้เป็นค่าคงที่ค่าหนึ่งด้วย ■

ตัวอย่างของการแบ่งตารางแสดงในรูปที่ 4.4 เริ่มจากผลบวกชั่วคราวโดยตำแหน่งที่มีกากบาทจะแสดงตำแหน่งที่มีค่าเป็น 2 ในขั้นแรกจะพิจารณา $part(x, y, 0)$ และพบว่ามี $part(0,0,0)$ และ $part(1,1,0)$ ที่ตรงตามเงื่อนไขของกฎการลดแถว ซึ่งจะเปลี่ยนค่าเป็นตัวทดในจำนวนทด นอกจากนี้ยังมี $pos(1,2,0)$ ที่จะทำการส่งต่อตัวทดไปยัง $pos(1,2,1)$ ในขั้นตอนที่สองก็มีการทดจาก $pos(1,2,1)$ ในขั้นตอนสุดท้าย $part(2,0,3)$ ตรงกับกฎการลดแถว และเปลี่ยนไปเป็นตัวทด



รูปที่ 4.4 ขั้นตอนในการแยกตาราง

อัลกอริทึมการบวกจำนวนฐานคู่ทั่วไป

จากส่วนประกอบของอัลกอริทึมทั้งสองส่วน จะมีข้อสังเกต เช่นเดียวกับการบวกของจำนวนฐานคู่ปกติ คือ จำนวนทดจะมีแถวที่มีค่าเป็น 0 เพิ่มขึ้นเรื่อย ๆ เมื่อแถวที่มีค่าเป็น 0 ของจำนวนทด ไปรวมกับค่าของจำนวนเหลือ ผลที่ได้จะมีค่าอยู่ในชุดตัวเลข $\{0,1\}$ เท่านั้น ดังนั้น ในการทำงานรอบที่ j ของอัลกอริทึมการบวกทั่วไป จึงไม่จำเป็นต้องทำการแยกตารางของแถวที่ 0 ถึงแถวที่ $j-1$

สรุป อัลกอริทึมการบวกสำหรับจำนวนฐานคู่ทั่วไปนั้น จะทำงานเป็นรอบ ๆ โดยมีจำนวนรอบเท่ากับจำนวนแถวของรูปแบบแสดงค่าที่นำมาบวกกัน การทำงานในแต่ละรอบจะเริ่มจากการบวกแต่ละส่วนย่อยเข้าด้วยกัน โดยที่ตัวทดนั้นจะส่งต่อไปเรื่อย ๆ จนถึงตำแหน่งแรกของส่วนย่อยถัดไป ซึ่งมีทั้งหมด $L+1$ ขั้น หลังจากนั้นจะนำไปแยกออกเป็นสองตารางโดยใช้อัลกอริทึมแยกตาราง ตั้งแต่แถวที่ j ขึ้นไป พร้อมกันทุกแถว ซึ่งใช้เวลาทั้งหมด L ขั้น อัลกอริทึมการบวกทั้งหมดสรุปเป็นทฤษฎีบทได้ดังนี้

ทฤษฎีบทที่ 4.2 *การบวกจำนวนฐานคู่* $(2,\beta)$ *ซึ่ง* $\beta = 2^0 + 2^{i_1} + \ldots + 2^{i_l}$ *สามารถทำได้ด้วยอัลกอริทึมการบวกทั่วไป ภายในเวลาที่เป็นสัดส่วนโดยตรงกับจำนวนแถวของรูปแบบแสดงค่า*

อัลกอริทึม 4 อัลกอริทึมการบวกทั่วไป

Input: $A = \sum_{i,j} a_{i,j} 2^i \beta^j$

$B = \sum_{i,j} b_{i,j} 2^i \beta^j$

Output: $C = \sum_{i,j} c_{i,j} 2^i \beta^j$ โดยที่ $C = A + B$

Define $n =$ the amount of rows in the representation

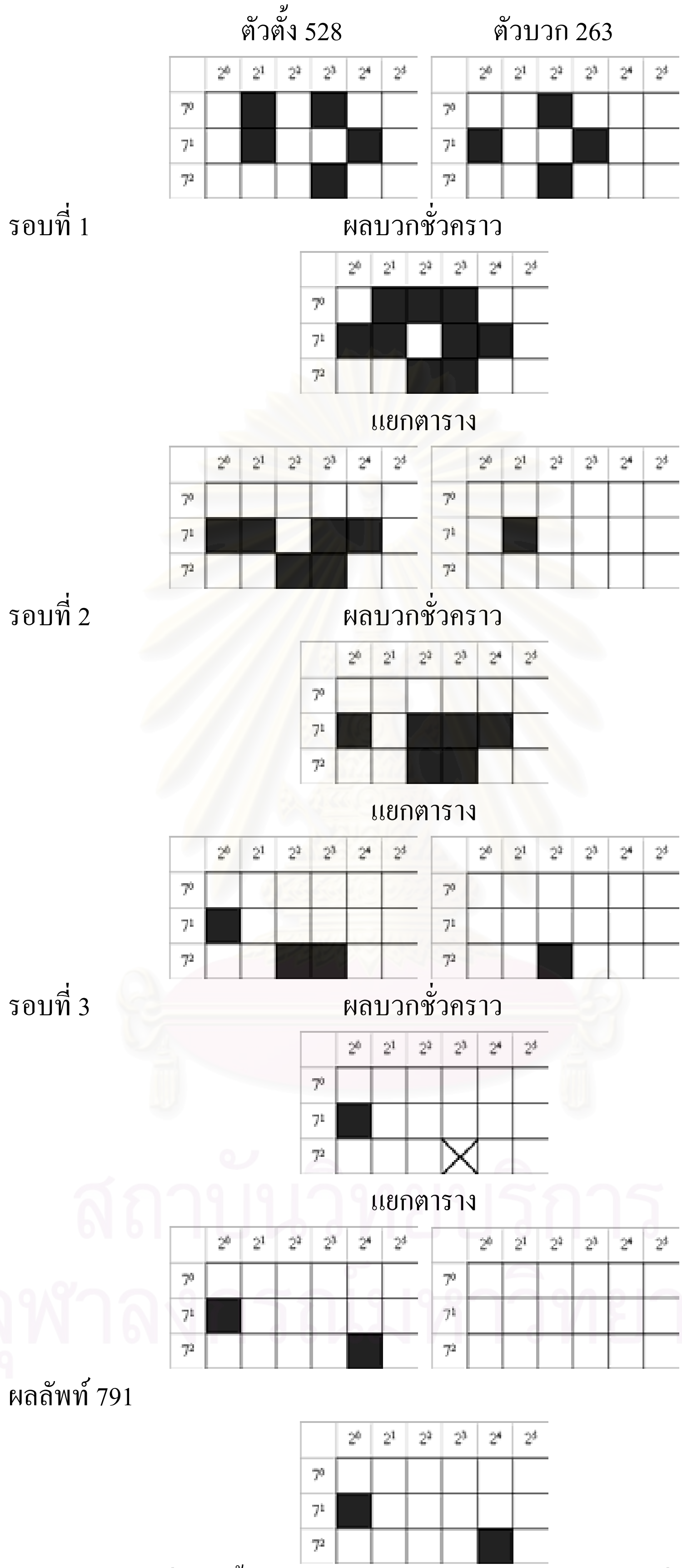
ขั้นตอน

```
j = 0
while (j < n)
do    A = A + B
      A, B ← split(A)
      j = j + 1
C = A
```

พิสูจน์ อัลกอริทึมนี้เกิดจากอัลกอริทึมย่อย ที่มีการพิสูจน์แล้วว่าไม่ทำให้ค่าผลลัพท์เปลี่ยนไป ทำให้ค่าของผลลัพท์เท่ากับผลบวกที่ต้องการ นอกจากนี้ หลังจากทำไป n รอบแล้วค่าของ B จะเป็น 0 ทำให้ผลลัพท์ มีค่าเท่ากับจำนวนเหลือ จึงสรุปได้ว่าการอัลกอริทึมนี้ให้ผลการบวกที่ถูกต้อง เมื่อพิจารณาด้านความเร็วแล้ว จะพบว่าอัลกอริทึมนี้มีการทำงานเป็นรอบ โดยจำนวนรวมเท่ากับจำนวนแถวของรูปแบบแสดงผล ซึ่งการทำงานในแต่ละรอบนั้นใช้เวลาคงที่ ทำให้เวลาที่ใช้เป็นสัดส่วนโดยตรงกับจำนวนแถว ■

รูปที่ 4.5 เป็นตัวอย่างของการใช้อัลกอริทึมการบวกทั่วไป โดยทำการบวก 528 และ 263 เข้าด้วยกัน โดยจำนวนทั้งสองมีฐานเป็น $(2,7)$

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 4.5 ขั้นตอนการทำงานของอัลกอริทึมการบวกทั่วไป

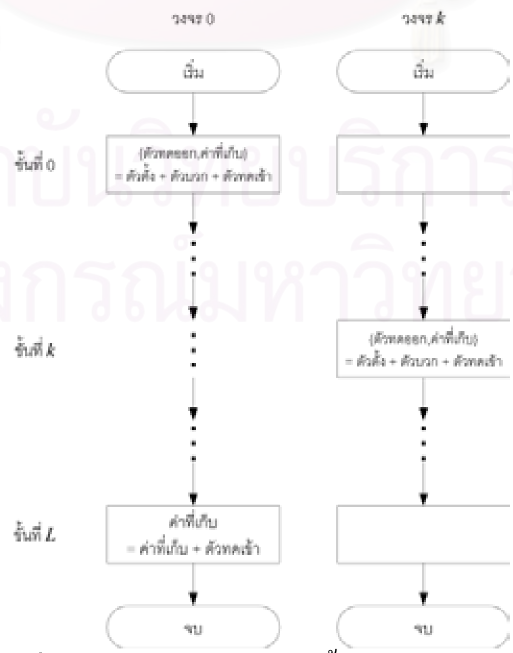
การออกแบบวงจร

วงจรสำหรับอัลกอริทึมนี้สามารถนำโครงสร้างจากวงจรการบวกเดิมมาใช้ได้ คือ จะแบ่งวงจรออกเป็นส่วนย่อย ๆ วงจรหนึ่งคำนวนค่าหนึ่งตำแหน่งของตารางแสดงค่า วงจรเหล่านี้จะทำงานประสานกันตามสัญญาณนาฬิกาส่วนกลาง โครงสร้างการทำงานของวงจรนี้จะเหมือนกับการบวกเลขฐานคู่ปกติ ดังรูปที่ 3.4 แต่วงจรสำหรับแต่ละตำแหน่งนั้นจะแตกต่างกัน โดยจำนวนชนิดของวงจรที่แตกต่างกันขึ้นกับความยาวของ β ในเลขฐานสอง เมื่อ $\beta = 2^0 + 2^{i_1} + \ldots + 2^{i_l}$ จำนวนวงจรจะเป็น $L = i_l + 1$ รูปแบบ โดยจะเรียกวงจรคำนวณค่าของ $pos(x, y, k)$ ว่าวงจร k ขั้นตอนการทำงานของวงจรขึ้นอยู่กับตำแหน่งที่วงจรนั้นอยู่ รายละเอียดของวงจรมีดังนี้

1. การออกแบบวงจรในการรวมค่า

การรวมค่านี้จะทำการบวกทุกส่วนย่อยพร้อมกัน โดยจะเริ่มรวมค่าจาก $pos(x, y, 0)$ ไล่ไปยัง $pos(x, y, i_l)$ ค่าที่จะต้องนำมารวมกันนั้น มี 3 ชนิดคือ ค่าของตัวตั้ง ค่าของตัวบวก และค่าตัวทดจากหลักก่อนหน้า ซึ่งผลรวมจากค่าทั้งสามนี้จะไม่เกิน 3 หรือ 11 ในฐานสอง ทำให้ทุกตำแหน่งทำงานได้เหมือนกัน คือ ค่าตัวทดไม่เพิ่มค่าขึ้นตามตำแหน่ง และหลักจากบวก $pos(x, y, i_l)$ แล้ว จะนำตัวทดไปบวกเข้ากับ $pos(x+1, y, 0)$ ไม่ทำการส่งต่อ ทำให้ค่าที่เก็บใน $pos(x+1, y, 0)$ มีโอกาสเป็นสองได้

จากการทำงานที่เป็นลำดับ ตามตำแหน่ง ดังนั้นวงจร k จะทำงานในรอบที่ k ของสัญญาณนาฬิกา และวงจร 0 จะทำการรวมค่าตัวทดในรอบที่ L ดังนั้น ณ เวลา k จะมีวงจร k ทำงานเพียงวงจรเดียวเท่านั้น วงจรอื่นจะไม่ทำงานใด ๆ ทั้งสิ้น ลำดับการทำงานในขั้นตอนรวมค่าของวงจร k จะอธิบายได้ดังรูปที่ 4.6



รูปที่ 4.6 ลำดับการทำงานในขั้นตอนการรวมค่า

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

2. การออกแบบวงจรในการแยกตาราง

ข้อมูลเริ่มต้นของขั้นตอนนี้จะเป็นระบบจำนวนลูกผสมซึ่ง $pos(x,y,0)$ จะมีค่าเป็น 2 ได้ การทำงานในขั้นนี้จะคล้ายกับขั้นที่แล้ว คือ การประมวลผลจะไล่ไปทีละวงจร ตามลำดับของสัญญาณนาฬิกา การทำงานของขั้นนี้จะเริ่มจากการที่วงจร 0 พิจารณ $part(x,y,0)$ ว่าตรงกับกฎการลดแถวหรือไม่ ถ้าตรงจะลดค่าของตำแหน่งตัวเอง และส่งสัญญาณลดค่าไปยังตำแหน่งตามข้อกำหนดของกฎการลดแถวนั้น นอกจากนี้ จะส่งสัญญาณไปยังวงจรอีกวงจรหนึ่ง ซึ่งทำหน้าที่เก็บจำนวนทด เพื่อทำการปรับค่าของจำนวนทดในตำแหน่งนั้น ๆ ให้มีค่าที่ถูกต้อง แต่ถ้าไม่ จะพิจารณาว่าค่าที่เก็บอยู่นั้นเป็น 2 หรือไม่ ถ้าเป็นจะปรับค่าตัวเองให้เป็น 0 และส่งค่านั้นเป็นตัวทดไปยังหลักต่อไป ในรอบถัดไปของสัญญาณนาฬิกา ตำแหน่งนอกเหนือจาก $pos(x,y,0)$ จะปรับค่าของตำแหน่งตัวเองตามสัญญาณ หรือตัวทดตามที่ได้รับจาก $pos(x,y,0)$ และในรอบที่ $2k$ วงจร k จะพิจารณา $part(x,y,k)$ กับกฎการลดแถว และส่งสัญญาณ รวมทั้งพิจารณาการส่งต่อตัวทด ในรอบถัดไปวงจรอื่น ๆ จะปรับค่าตามสัญญาณจากวงจร k ซึ่งลำดับการทำงานนี้สามารถเขียนเป็นแผนภูมิได้ดังรูปที่ 4.7

รูปที่ 4.7 ลำดับการทำงานในขั้นตอนแยกตาราง

ในส่วนของวงจรที่เก็บจำนวนทดนั้น จะเพียงเก็บข้อมูลให้ตรงตามจังหวะการทำงานของการแยกตารางเท่านั้น

3. ข้อมูลที่จำเป็นในวงจรระดับตำแหน่ง

วงจรที่คำนวณค่าจากการบวกทั่วไปนั้นจะใช้ข้อมูลในการพิจารณามากกว่าแบบปกติ โดยมีสัญญาณหลัก ๆ เหมือนเดิม คือชนิดของข้อมูลที่จะต้องรับเพื่อใช้ในการทำงาน เหมือนกับวงจรการบวกปกติ แต่ เพื่อให้ครอบคลุมกับการทำงานที่มีฐานเป็น $(2,\beta)$ ใด ๆ จึงต้องเพิ่มจำนวนของข้อมูลต่อไปนี้

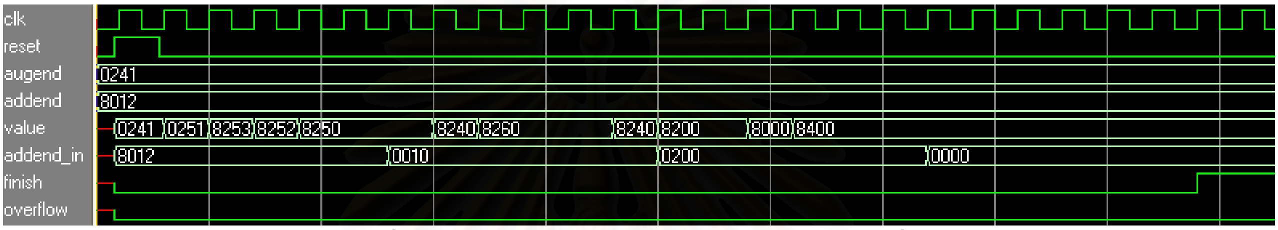
1. ค่าที่เก็บในหลักถัดไป จากเดิมรับเพียงหนึ่งหลักเท่านั้น แต่ในวงจรนี้จะรับมาทั้งหมด $L-1$ หลัก เพื่อให้พอเพียงต่อการพิจารณากฎการลดแถว
2. สัญญาณแจ้งการเกิดการลดแถวของหลักก่อนหน้า จากเดิมจะรับมาจากหนึ่งหลักก่อนหน้าเท่านั้น แต่ในกรณีนี้จะต้องเปลี่ยนไปรับจากทั้งหมด $L-1$ หลัก เนื่องจากทั้ง $L-1$ หลักนั้นสามารถจะส่งผลกระทบมายังตำแหน่งที่คำนวณอยู่ได้

ในส่วนข้อมูลที่จะต้องส่งออกไปนั้นจะเหมือนกับวงจรบวกเดิม แต่ค่าที่ส่งไปนั้น มีวงจรอื่นจำนวนมากขึ้นที่จะรับค่าเหล่านี้ไปพิจารณา

การทดลองจำลองวงจร

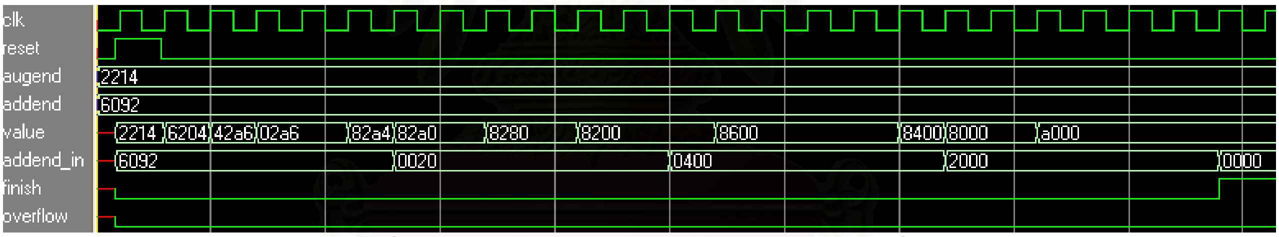
เพื่อแสดงให้เห็นว่าอัลกอริทึมการบวกทั่วไปสามารถนำมาสร้างเป็นวงจรได้จริง จึงจำลองวงจรการบวกจำนวนฐานคู่ (2,3) และ (2,7) ขึ้น โดยมีโครงสร้างของวงจรเดิม แต่การทำงานของส่วนวงจรบวกหนึ่งบิทเปลี่ยนไป

สำหรับวงจรบวกของจำนวนฐานคู่ (2,3) นั้นใช้ตัวตั้ง และตัวบวกตามรูปที่ 3.3 คือ 221 และ 31 ได้ผลการทำงานในรูปของคลื่นดังรูปที่ 4.8



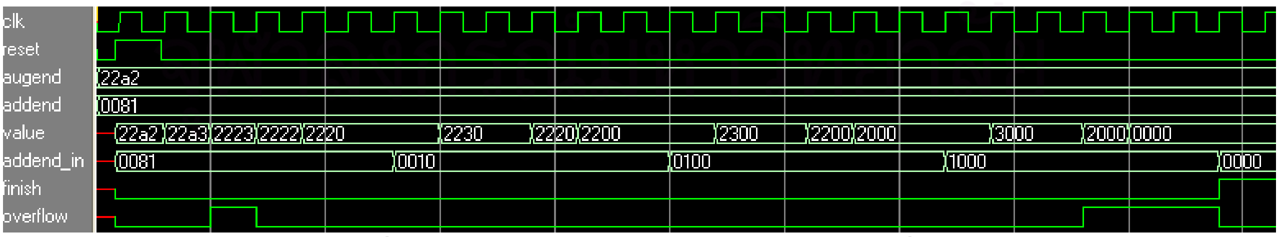
รูปที่ 4.8 ผลการบวก 221 และ 31 โดยใช้ตัวบวกทั่วไป

และทดสอบบวก 79 และ 270 ซึ่งอยู่ในรูปแบบแสดงค่าเป็น 2214 และ 6092 โดยเป็นรูปแบบแสดงค่าที่มีตำแหน่งที่มีค่าอยู่ติดกัน ซึ่งผลการบวกในรูปคลื่นเป็นดังรูปที่ 4.9



รูปที่ 4.9 ผลการบวก 49 และ 270 โดยใช้ตัวบวกทั่วไป

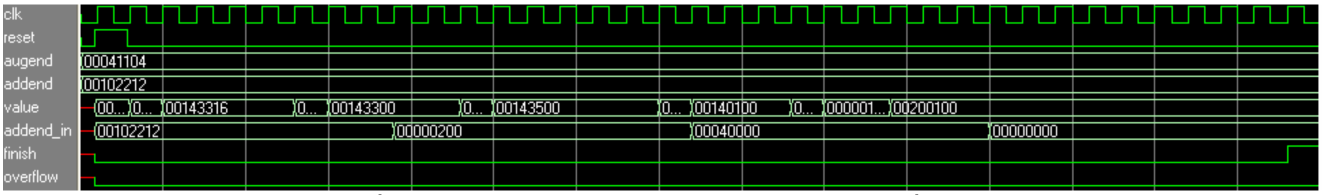
นอกจากนี้ยังปรับปรุงให้แจ้งเมื่อเกิดส่วนล้นขึ้นทั้งในแถว และหลักใด ๆ โดยบวก 104 และ 25 ในรูป 22a2 และ 0081 ส่วนล้นที่เกิดขึ้นนั้นจะเป็นส่วนล้นในแนวแถวในรอบแรกของการบวก และ เกิดส่วนล้นในแนวหลักในรอบที่สี่ ดังรูปที่ 4.10



รูปที่ 4.10 ผลการบวก 104 และ 25 โดยใช้ตัวบวกทั่วไป

ในส่วนของตัวบวกของระบบจำนวนฐานคู่ (2,7) นั้นจำลองวงจรที่บวกจำนวน 6 บิท จำนวน 4 แถว ดังนั้นจึงใช้เลขฐานแปดสองจำนวนเพื่อแสดงค่าของแต่ละแถว และให้ให้เลขหลัก

สูง แทนแถวที่มีค่าประจำหลักสูง ทำการบวกค่าของ 528 และ 268 ในรูปแบบแสดงค่า 00041104 และ 00102212 ตามรูปที่ 4.5 ซึ่งผลจากการจำลองการบวกเป็นดังรูปที่ 4.11



รูปที่ 4.11 ผลการบวก 528 และ 268 โดยใช้ตัวบวกทั่วไป

จากการทดลองจะพบว่าอัลกอริทึมการบวกทั่วไปนั้นสามารถนำมาสร้างเป็นวงจรได้จริง โดยค่าที่นำมาบวกนั้นจะอยู่ในรูปแบบแสดงค่าใด ๆ ก็ได้ พร้อมทั้งสามารถแจ้งเมื่อเกิดส่วนล้นขึ้นได้อีกด้วย

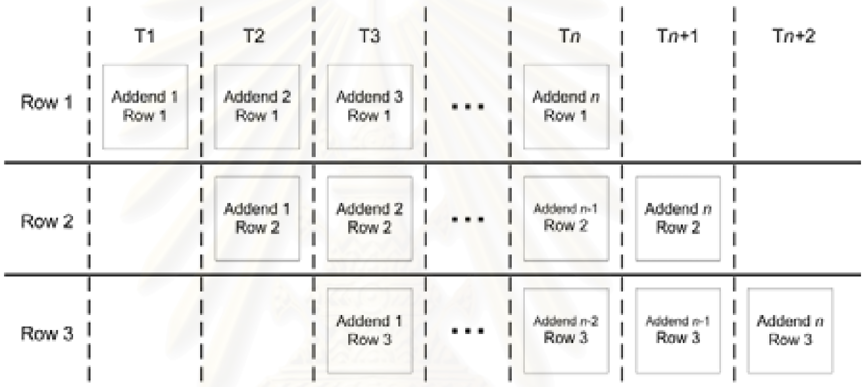
ในแง่ของระยะเวลาในการทำงานนั้น สำหรับระบบจำนวนฐานคู่ $(2,3)$ จะทำงานช้ากว่าการบวกด้วยอัลกอริทึมการบวกแรก เมื่อรูปแบบแสดงค่ามีทั้งหมด n แถว จะใช้เวลาตั้งแต่เริ่มป้อนข้อมูล จนได้ผลลัพท์ที่ถูกต้องเป็น $6n+1$ รอบของสัญญาณนาฬิกา ส่วนระบบจำนวนฐานคู่ $(2,7)$ นั้นจะใช้เวลาเป็น $9n+1$ หรือถ้ามองในรูปแบบทั่วไปเวลาที่ใช้สำหรับฐาน β หนึ่ง ๆ จะเป็น $cn+1$ เมื่อ c เป็นค่าคงที่ หรือกล่าวได้ว่าการบวกใช้เวลาแปรผันตรงตามจำนวนแถวของรูปแบบแสดงค่า และเมื่อมองในรายละเอียดแล้วพบว่า c นั้นจะแปรผันตรงกับความยาวของฐาน β ในรูปฐานสอง หรือแทนด้วย L นั้น $c=3L$ ดังนั้นระยะเวลาในการบวกจึงแปรผันตาม Ln

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 5

การบวกแบบสายท่อ และการคูณ

กระบวนการบวกเชิงกำหนดนั้นมีข้อดีที่สามารถบอกได้ว่าบริเวณใดจะบวกเสร็จเมื่อใด โดยไม่ต้องรอให้กระบวนการบวกทั้งหมดเสร็จสิ้นก่อน ซึ่งวิธีการที่จะนำข้อดีตรงนี้ไปใช้ก็คือ การนำวิธีการทำงานแบบสายท่อมาใช้ การทำงานแบบสายท่อนี้ จะเป็นวิธีการทำงานที่แบ่งการทำงานออกเป็นขั้น ๆ โดยที่แต่ละขั้นเป็นอิสระต่อกัน หรือสามารถทำงานไปพร้อม ๆ กันได้ เมื่อพิจารณากระบวนการบวกเชิงกำหนดที่ได้เสนอมานั้นจะพบว่า การบวกของแต่ละแถวเป็นอิสระต่อกัน สามารถที่จะทำการบวกพร้อม ๆ กันได้ และเมื่อบวกในแถวนั้น ๆ เสร็จแล้ว จึงจะส่งตัวทดระหว่างแถวไปยังแถวถัดไป การบวกแบบสายท่อนี้สามารถเขียนได้ดังรูปที่ 5.1



รูปที่ 5.1 การบวกแบบสายท่อ

จากรูปนั้นจะเห็นว่าการบวกจะเริ่มจากการส่งข้อมูลเข้ามายังตัวดำเนินการบวกทีละแถว ในรอบการทำงานแรก จะมีเพียงแถวแรกของตัวบวกตัวแรก และในรอบการทำงานถัด ๆ ไปข้อมูลของตัวบวกตัวอื่น ๆ จะถูกป้อนเข้าไปยังตัวดำเนินการทีละแถว กระบวนการเช่นนี้เป็นการเพิ่มความเร็วให้กับการบวกเป็นอย่างมาก ถ้ามีการบวกติดต่อกันมากกว่าหนึ่งตัว เช่น รูปแบบแสดงค่ามีที่ทั้งหมด n แถว และใช้กระบวนการบวกทั่วไป จะใช้ระยะเวลาในการบวกเป็น nT ถ้าจะบวกทั้งหมด k ตัวต่อเนื่องกัน จะใช้เวลาเป็น nkT แต่เมื่อพัฒนาให้เป็นการบวกแบบสายท่อ การบวกหนึ่งตัวจะใช้เวลา nT เท่ากัน แต่ถ้ามีการบวกสองตัวต่อเนื่องกัน จะใช้เวลาเป็น $(n+1)T$ และถ้าจะบวกทั้งหมด k ตัวต่อเนื่องกัน ก็จะใช้เวลาเพียง $(n+k-1)T$ เท่านั้น

จากแนวความคิดการบวกแบบสายท่อนี้ จะสามารถนำมาดัดแปลงขึ้น จากรูป 5.1 จะเห็นว่าการส่งข้อมูลเข้าไปทำการบวกนั้นในเวลาหนึ่งจะส่งไปทั้งหมด n แถว จากตัวบวก n ตัว ดังนั้นถ้าเปลี่ยนการส่งข้อมูลทั้ง n แถวนั้น ให้มาจากตัวบวกเพียงแค่ตัวเดียวก็สามารถทำงานได้เช่นกัน และใช้ระยะเวลาเท่ากัน เพียงแต่จำนวนทดในแต่ละขั้นนั้นจะเปลี่ยนไปเป็นจำนวนทดที่เกิดจากหลายค่ามาประกอบกัน การเปลี่ยนแปลงนี้จะมีผลต่อการออกแบบวงจรเท่านั้น

กระบวนการบวกนี้จะเห็นผลมากเมื่อไปเป็นส่วนหนึ่งของตัวปฏิบัติการคูณ ซึ่งการคูณจะทำโดยการกระจายตัวตั้งไปยังทุกตำแหน่งที่มีค่าของตัวคูณ แล้วจึงนำผลจากการคูณนั้นหมดนั้นรวมเข้าไว้ด้วยกัน หรือเขียนให้อยู่ในรูปของสมการได้ดังนี้

$$A \times B = A \times \sum_{i,j} b_{i,j} 2^i \beta^j$$
$$= \sum_{i,j} A b_{i,j} 2^i \beta^j$$

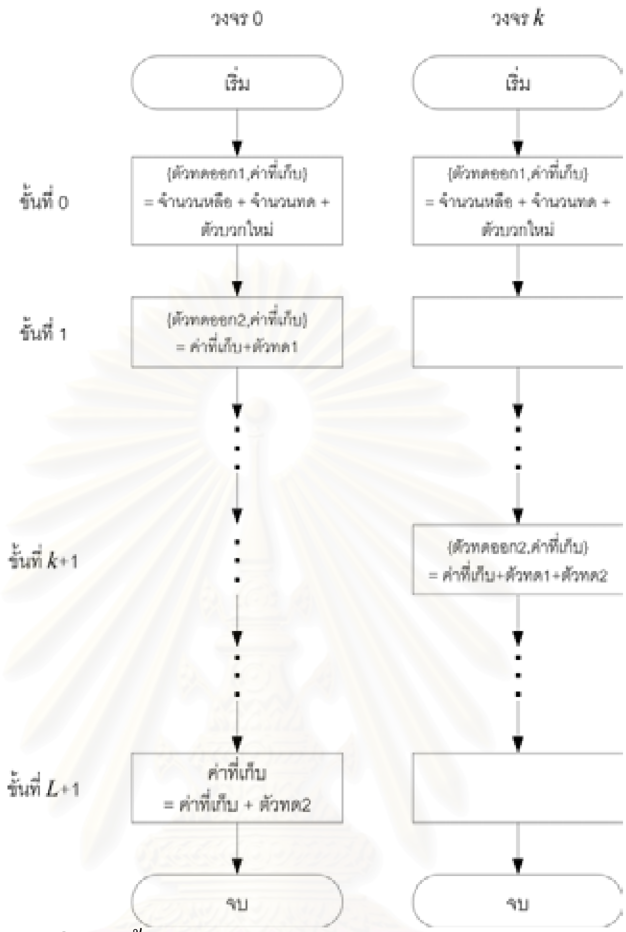
การคูณของตัวตั้ง กับ $b_{i,j}$ ใด ๆ นั้นก็คือ การคูณของของตัวตั้ง กับค่าประจำตำแหน่ง $2^i \beta^j$ ถ้ามองในรูปของตารางจะเห็นว่าการคูณนี้ เป็นการเลื่อนตำแหน่งของตัวตั้ง เพิ่มขึ้น i ตำแหน่งในแนวนอน และ j ตำแหน่งในแนวตั้ง แล้วจึงนำผลการเลื่อนตำแหน่งทั้งหมดมาบวกกัน ถ้ากระบวนการเลื่อนตำแหน่งนี้ทำได้ในเวลาคงที่แล้ว ระยะเวลาของการคูณเมื่อใช้กระบวกการบวกทั่วไปจะถึงกับผลคูณของทั้งจำนวนแถว และปริมาณตำแหน่งที่มีค่าของตัวคูณ แต่เมื่อใช้กระบวนการบวกแบบสายท่อมาช่วย ระยะเวลาที่ใช้ในการคูณก็จะขึ้นกับจำนวนแถว หรือจำนวนตำแหน่งที่มีค่าของตัวคูณเพียงอย่างเดียว จะเห็นได้ว่าความเร็วเพิ่มขึ้นอย่างมาก แต่จะต้องมีการปรับปรุงโครงสร้างจากการบวกทั่วไปเล็กน้อย

การบวกแบบสายท่อ

การบวกแบบสายท่อนั้นจะต้องมีการเปลี่ยนแปลงโครงสร้างจากแบบเดิมเล็กน้อย โดยการบวกแบบเดิมนั้นในแต่ละรอบจะมีจำนวนที่ต้องนำมารวมกันอยู่สองจำนวน คือ ตัวตั้ง และตัวบวก โดยในรอบแรก จะเป็นตัวตั้ง และตัวบวกจริง ๆ ของการบวก แต่ในรอบการทำงานอื่น จะเป็นจำนวนเหลือ และจำนวนทด และผลการรวมค่านั้นจะอยู่ในรูปของผลบวกชั่วคราว ซึ่งเป็นระบบจำนวนฐานคู่ลูกผสม ที่ยอมให้ตำแหน่งที่มีค่าประจำตำแหน่งน้อยที่สุดของแต่ละส่วนย่อยเก็บค่าของ 2 ได้

แต่เมื่อเปลี่ยนเป็นการทำงานแบบสายท่อ จะต้องมีการเพิ่มจำนวนที่จะนำมาบวกขึ้นอีกหนึ่งจำนวนในแต่ละรอบ คือ มีค่าจากตัวบวกตัวใหม่เพิ่มเข้ามายังตำแหน่งที่กำลังพิจารณาอยู่ด้วย ทำให้ไม่สามารถที่จะบวกจำนวนทั้งหมดพร้อมกันได้ เพราะว่าในหลักที่ k จะต้องมีการรวมค่าทั้งหมด 4 ค่า ได้แก่ สามค่าที่ได้กล่าวมาแล้ว และ ค่าตัวทดจากหลักที่ $k-1$ ด้วย ส่งผลให้ตัวทดที่จะส่งไปยังตำแหน่ง $k+1$ มีโอกาสที่มีค่าเป็น 2 ซึ่งจะกระทบกับผลบวกชั่วคราว จึงต้องเพิ่มกระบวนการรวมค่าขึ้นอีกหนึ่งขั้น ขั้นตอนแรกนี้จะทำงานพร้อมกัน ทุกตำแหน่งจะนำค่าของจำนวนเหลือ จำนวนทด และตัวบวกใหม่มารวมกัน และจะเก็บผลไว้ ร่วมทั้งส่งตัวทดออกไปเป็นตัวทด 1 ในขั้นที่ 1 วงจร 0 จะนำค่าของตัวทด 1 ที่ได้รับมารวมกับค่าที่เก็บ แล้วเปลี่ยนค่าที่เก็บนั้นพร้อมกับสร้างตัวทด 2 ออกไปยังตำแหน่งถัดไป ในขั้นที่ $k+1$ วงจร k จะเอาค่าที่เก็บ ตัวทด 1 และตัวทด 2 มารวมกัน จากนั้นเปลี่ยนแปลงค่าที่เก็บ และส่งเป็นตัวทด 2 ออกไปเรื่อย ๆ จนถึงวงจรที่ $L-1$ วงจรนี้ก็จะทำงานเช่นเดียวกัน และส่งตัวทด 2 ไปยังวงจร 0 และในขั้นสุดท้าย วงจร 0

จะรวมค่าของตัวทดนี้กับค่าที่เก็บอยู่ กลายเป็นผลบวกชั่วคราวในที่สุด รูปที่ 5.2 เป็นรูปที่แสดงลำดับการทำงานนี้



รูปที่ 5.2 ขั้นตอนการรวมค่าของการบวกแบบสายท่อ

เมื่อการทำงานเปลี่ยนไปข้อมูลที่จะต้องรับ และข้อมูลที่ส่งออกไปยังวงจรบวกบิทอื่น ๆ ก็จะต้องมีการเปลี่ยนแปลงเป็นดังนี้

ข้อมูลขาเข้า

1. สัญญาณนาฬิกา
2. สัญญาณ reset
3. ค่าตัวบวก การบวกแบบนี้จะไม่มีการรับค่าตัวตั้ง โดยค่าตั้งต้นของตำแหน่งเป็น 0 เสมอ
4. ค่าจำนวนทด
5. ค่าตัวทด จำนวน 2 บิท รับมาจากหลักก่อนหน้า
6. สัญญาณการเกิดการลดแถว เป็นสัญญาณที่รับจาก $L-1$ หลักก่อนหน้า เพื่อทำการปรับลดค่าของตำแหน่งปัจจุบันให้ถูกต้อง
7. ค่าของหลักถัดไป จาก $L-1$ หลักถัดไป

8. สัญญาณหยุด

ส่วนสัญญาณที่ส่งออกมีดังนี้

1. ค่าของตำแหน่งปัจจุบัน
2. ค่าตัวทด ซึ่งเกิดการขั้นตอนการรวมค่าจำนวน 2 บิท
3. สัญญาณเกิดการลดแถว เพื่อแจ้งไปยังหลักถัดไปให้ปรับค่าให้ถูกต้อง
4. สัญญาณหยุด
5. สัญญาณแจ้งการคำนวณครบหนึ่งรอบ เพื่อแจ้งส่วนควบคุมให้เปลี่ยนตัวบวก

นอกจากการทำงานส่วนการรวมค่าแล้ว ยังมีส่วนที่จะต้องปรับเปลี่ยน คือ ต้องมีการสร้างส่วนควบคุมเพื่อให้การส่งข้อมูลเข้าไปยังสายท่อทำได้ถูกเวลา รวมทั้งเป็นตัวสั่งให้หยุดการทำงานด้วย

การคูณระบบจำนวนฐานคู่ทั่วไป

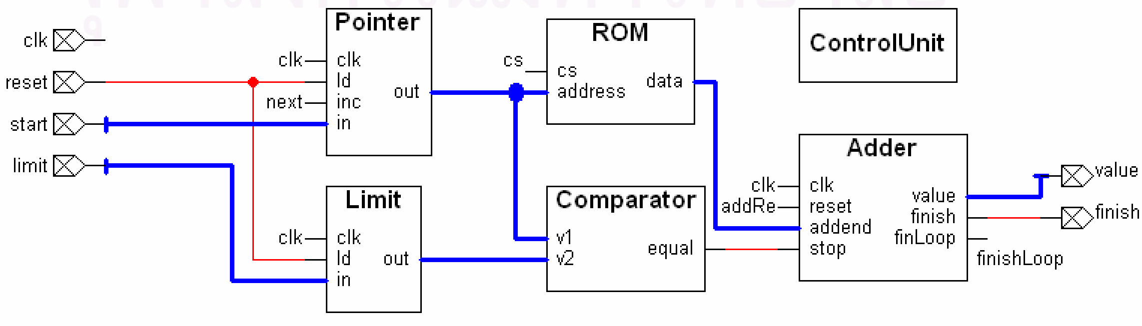
ดังที่ได้กล่าวมาแล้วว่าการคูณจะมีพื้นฐานจากการบวก และไม่มีอัลกอริทึมที่ซับซ้อนและสามารถเขียนลำดับการทำงานเพื่อการสร้างเป็นวงจร ดังนี้

1. หาตำแหน่งที่มีค่าของตัวคูณ และ ปรับค่าตำแหน่งนั้นให้เป็น 0
2. เลื่อนตัวตั้งตามผลจากข้อ 1
3. ส่งข้อมูลที่เลื่อนแล้วไปทำการบวก

ทั้งสามขั้นตอนนั้นจะทำงานต่อเนื่องกัน ซึ่งแต่ละขั้นสามารถทำงานในเวลาคงที่ได้ทั้งหมด

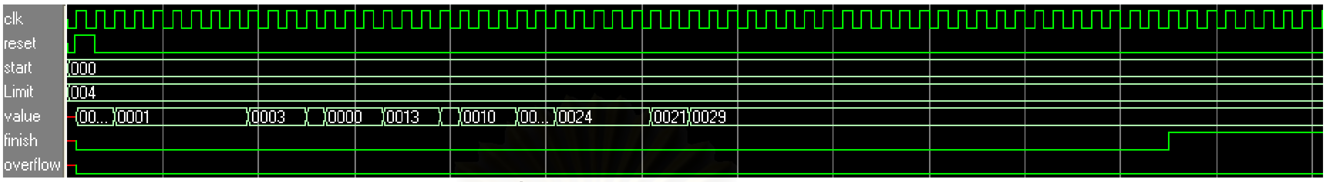
การทดลองจำลองวงจร

ในขั้นแรกทำการสร้างตัวบวกแบบสายท่อที่บวกจำนวนฐานคู่ (2,3) ขนาด 4 บิท 4 แถวรับตัวบวกรอบละหนึ่งตัว โดยจะจำลองวงจรในส่วนที่ส่งข้อมูลให้กับตัวบวกนี้ด้วย โครงสร้างของวงจรบวกนั้นจะยังคล้ายตัวบวกแบบทั่วไป เพียงเปลี่ยนสัญญาณเข้า และออกจากวงจรให้ตรงตามสัญญาณที่เปลี่ยนไป ในขณะที่ส่วนที่ส่งข้อมูลไปยังตัวบวกนั้น จะประกอบด้วย ตัวควบคุม หน่วยความจำ รวมทั้งวงจรที่ทำหน้าที่ชี้ตำแหน่งตัวบวกปัจจุบัน และ วงจรเก็บตำแหน่งตัวบวกตัวสุดท้าย โดยสร้างส่วนที่ส่งข้อมูลไปบวกเป็นดังรูปที่ 5.3



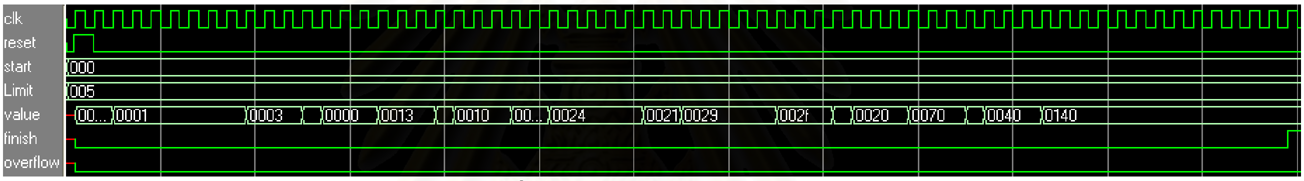
รูปที่ 5.3 เส้นทางการส่งข้อมูลของวงจรบวก

การทดลองจะเก็บเลข 1 ถึง 10 ไว้ในหน่วยความจำตำแหน่งที่ 0 ถึง 9 ในครั้งแรกทำการบวกจำนวนที่เก็บในหน่วยความจำตั้งแต่ตำแหน่งที่ 0 ถึงตำแหน่งที่ 4 ซึ่งมีค่าเป็น 15 ผลการบวกในรูปของคลื่นเป็นดังรูปที่ 5.4 จะเห็นว่าใช้เวลาทั้งหมด 58 รอบสัญญาณนาฬิกา ซึ่งเป็นเวลาที่มีการป้อนตัวบวกใหม่ 5 ตัว 35 รอบสัญญาณนาฬิกา และ ใช้ส่งตัวทดอีก 3 ครั้ง 21 รอบสัญญาณนาฬิกา รวมแล้ว 56 รอบสัญญาณ และรวมการระยะเวลาในการตั้งค่าเริ่มต้นอีก 2 รอบสัญญาณ



รูปที่ 5.4 ผลการบวกเลข 1 ถึง 5

และเมื่อเพิ่มตัวบวกเข้าไปอีกหนึ่งตัวเป็นการบวกตั้งแต่ตำแหน่งที่ 0 ถึงตำแหน่งที่ 5 ผลที่ได้มีค่าเป็น 21 และใช้เวลาทั้งสิ้น 65 รอบสัญญาณนาฬิกา โดยผลการบวกในรูปคลื่นเป็นดังรูปที่ 5.5



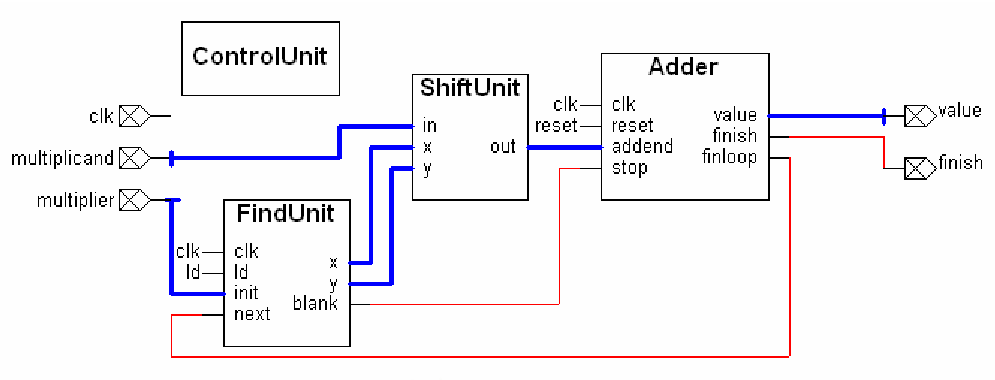
รูปที่ 5.5 ผลการบวกเลข 1 ถึง 6

ผลจากการใช้การบวกแบบสายท่อทำให้ระยะเวลาที่ใช้ในการบวกนั้นขึ้นกับจำนวนตัวบวก หรือจำนวนแถวเพียงจำนวนเดียวเท่านั้น

ในการจำลองวงจรคูณนั้นจะจำลองวงจรคูณในระบบจำนวนฐานคู่ $(2,3)$ ขนาด 4 บิท 4 แถว โดยใช้วงจรบวกแบบสายท่อที่ได้จำลองไปเป็นส่วนประกอบหลัก และต้องมีการจำลองวงจรเพิ่มเติมดังนี้

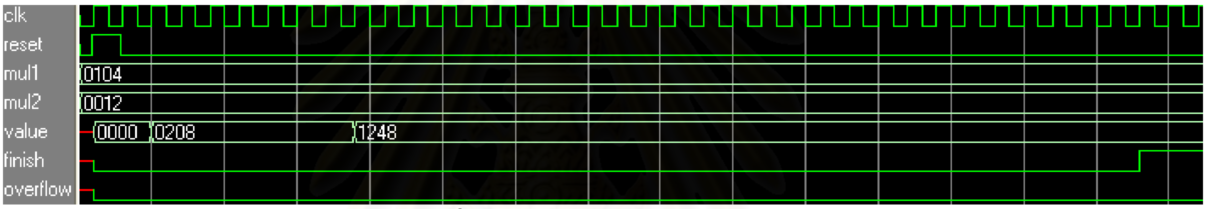
1. วงจรหาตำแหน่งที่มีค่า วงจรนี้จะรับตัวคูณเข้ามา แล้วทำการหาตำแหน่งที่มีค่าเพื่อส่งไปยังวงจรเลื่อนตำแหน่ง และจำต้องสามารถปรับตำแหน่งนั้นให้มีค่าเป็น 0 เพื่อที่จะส่งตำแหน่งที่มีค่าตำแหน่งอื่นต่อไป
2. วงจรเลื่อนตำแหน่ง จะรับตัวตั้ง และ ระยะทางที่จะต้องเลื่อนตามแนวแกน x และ แกน y โดยวงจรนี้จะประกอบด้วยมักซ์เป็นส่วนประกอบหลัก
3. ส่วนควบคุม ที่จะส่งสัญญาณไปยังวงจรต่าง ๆ ให้ทำงานประสานกันได้อย่างถูกต้อง

การคูณนั้นจะมีเส้นทางการส่งข้อมูลของวงจรรูปที่ 5.6



รูปที่ 5.6 เส้นทางการส่งข้อมูลของวงจรคูณ

การทดสอบความถูกต้องจะนำค่าของ 5 และ 13 มาคูณกัน เช่นเดียวกับรูปที่ 2.8 ซึ่งผลที่ได้จะมีค่าเป็น 5 และผลการจำลองการทำงานในรูปคลื่นเป็นดังรูปที่ 5.7 เวลาที่ใช้ในการคูณทั้งหมดเป็น 37 รอบสัญญาณนาฬิกา ซึ่งเกิดจากการที่ตัวคูณมีตำแหน่งที่มีค่าทั้งหมด 2 ตัว ทำให้ใช้เวลาเท่ากับการบวกจำนวนสองจำนวน



รูปที่ 5.7 ผลการคูณของ 5 และ 13

สรุปเวลาที่ใช้ในการคูณนั้นจะขึ้นกับจำนวนตำแหน่งที่มีค่าของตัวคูณ หรือจำนวนแถวของรูปแบบแสดงผลเท่านั้น

บทที่ 6
สรุปผลการวิจัย

ผลการวิจัย

จากการนำเสนอระบบจำนวนฐานคู่โดยดิมิทรอฟและคณะ ซึ่งเป็นระบบจำนวนที่มีความน่าสนใจ เนื่องจากมีความซ้ำซ้อนสูง และมีการกระจายตัวสูง แต่การนำเสนอผลงานเกี่ยวกับระบบจำนวนนี้ยังมีน้อย ซึ่งอาจจะเกิดจากการที่ยังไม่มีกระบวนการ และลำดับวิธีการทำงานที่แน่นอนสำหรับตัวปฏิบัติการพื้นฐาน อันได้แก่ การบวก และการคูณ ซึ่งการทำปฏิบัติการทั้งสองที่มีการเสนอก่อนหน้านี้ ใช้กระบวนการที่ไม่ใช่กระบวนการเชิงกำหนด และมีการออกแบบวงจรโดยใช้วงจรแบบอะแนล็อก ทั้งยังเป็นระบบอสมวาร งานวิจัยนี้จึงมุ่งเน้นที่จะนำเสนอลำดับของตัวปฏิบัติการพื้นฐานโดยให้เป็นกระบวนการเชิงกำหนด และแนะนำแนวทางในการออกแบบวงจรโดยใช้ระบบดิจิทัล และมีการทำงานแบบสมวาร ถึงแม้การทำงานของลำดับขั้นตอนที่นำเสนอนี้จะช้ากว่าลำดับการทำงานที่มีการนำเสนอมาก่อน แต่สามารถที่จะนำสถาปัตยกรรมแบบสายท่อมาใช้ในการเพิ่มความเร็วขึ้นได้มาก นอกเหนือจากนี้ยังได้นำเสนอความสำคัญ และผลกระทบของฐานซึ่งเป็นส่วนประกอบสำคัญของระบบจำนวนชนิดนี้ เพื่อให้ระบบจำนวนนี้มีความหลากหลาย และเหมาะสมต่อการใช้งานในแต่ละประเภทมากยิ่งขึ้น

สิ่งที่นำเสนอนั้นเริ่มต้นจากกระบวนการบวกระบบจำนวนฐานคู่ปกติซึ่งเป็นพื้นฐานของตัวปฏิบัติการอื่น ๆ ข้อจำกัดของตัวอัลกอริทึมนี้ คือ จำนวนที่นำมาบวกจะต้องอยู่ในรูปพร้อมบวกเสมอ แต่ผลจากการบวกนั้นก็อยู่ในรูปจำนวนพร้อมบวกเช่นเดียวกัน โดยการทำงานจะทำงานเป็นรอบ ๆ ในแต่ละรอบของการทำงานนั้นจะมีแถวในรูปแบบแสดงผลที่คำนวณเสร็จเรียบร้อย เพิ่มขึ้นรอบละหนึ่งแถวเสมอ ดังนั้นระยะเวลาในการบวกจึงขึ้นอยู่กับจำนวนแถวของรูปแบบแสดงผล

ความเร็วในการบวกนี้เมื่อเทียบกับความเร็วในการบวกของจำนวนฐานสองทั่วไป พบว่าเวลาในการบวกเลขฐานสองนั้นเพิ่มขึ้นเป็นเชิงเส้นกับจำนวนหลักของเลขที่บวก ความเร็วทั้งสองนี้ถึงแม้จะเติบโตเป็นเชิงเส้นเหมือนกัน แต่เวลาในการบวกของระบบจำนวนฐานคู่จะต่ำกว่าเนื่องจากจำนวนแถวของระบบจำนวนฐานคู่นั้น จะน้อยกว่าจำนวนหลักของเลขฐานสองเสมอ เมื่อทำการบวกเลขจำนวนเท่ากัน โดยสามารถพิสูจน์ได้ดังนี้

พิสูจน์ กำหนดให้เลขฐานสองที่บวกมีความยาว n ดังนั้น ค่าที่มากที่สุดที่จำนวนนี้จะเก็บได้ไม่เกิน 2^n แต่เมื่อทำการเก็บจำนวนนี้ในรูปของระบบจำนวนฐานคู่ โดยบังคับให้มีตำแหน่งที่มีค่าเพียงตำแหน่งเดียว และใช้จำนวนแถวมากที่สุด ตำแหน่งนั้นจะต้องอยู่ในหลัก 0 เสมอ และตำแหน่งนั้นจะมีค่าเป็น $2^0 3^m$ ซึ่งเห็นได้ชัดว่า $m < n$ ■

ส่วนการสร้างวงจรที่ทำงานนี้นั้นจะแบ่งรูปแบบแสดงผลออกเป็นส่วน ๆ แต่ละส่วนมีขนาดเท่ากับหนึ่งตำแหน่งในตาราง และแต่ละส่วนนี้มีการคำนวณค่าของตำแหน่งตัวเอง ทั้งนี้ก็เพื่อความรวดเร็วเพราะทุกส่วนสามารถทำงานไปพร้อม ๆ กันได้ วงจรย่อยทุกวงจรจะมีส่วนเชื่อมต่อกับวงจรอื่นเหมือน ๆ กัน แต่มีการทำงานที่แตกต่างกันเล็กน้อยเท่านั้น วงจรทั้งหมดจะมีการทำงานเป็นสถานะ การเปลี่ยนสถานะนั้นเป็นไปตามจังหวะของสัญญาณนาฬิกา และจากการทดลองจำลองวงจรนี้ก็พบว่าการทำงานเป็นไปได้อย่างถูกต้อง

เมื่อได้อัลกอริทึมสำหรับบวกจำนวนฐานคู่ปกติที่สามารถสร้างเป็นวงจรได้แล้ว จึงขยายของเขตของอัลกอริทึมให้ครอบคลุมสำหรับจำนวนฐานคู่ทั่วไป ความแตกต่างของระบบจำนวนฐานคู่ทั่วไปกับแบบปกตินั้น จะต่างกันที่ฐาน ๆ หนึ่งที่จากเดิมมีค่าเป็น 3 ถูกเปลี่ยนให้เป็นจำนวนคี่ใด ๆ ที่มีค่ามากกว่าหนึ่ง ซึ่งฐานนี้จะเป็นตัวกำหนดทั้งระดับความซ้ำซ้อน โดยเมื่อยิ่งมีค่ามากระดับความซ้ำซ้อนก็จะยิ่งลดลง นอกจากนี้การที่ฐานจะต้องเป็นเลขคี่เนื่องมาจากฐานจะเป็นตัวกำหนดกฎการลดแถว เมื่อฐานเป็นจำนวนคู่จะทำให้มีปัญหาต่อกฎการลดแถวได้ อัลกอริทึมสำหรับการบวกจำนวนฐานคู่ทั่วไปนี้ จะทำงานด้วยความซับซ้อนเท่ากับจำนวนแถวของรูปแบบแสดงค่า แต่จำนวนที่นำมาบวกกันนั้นไม่จำเป็นต้องอยู่ในรูปแบบพร้อมบวก เพราะมีการนำแนวคิดของระบบจำนวนลูกผสมมาช่วย เมื่อนำมาสร้างเป็นวงจรก็จะใช้แนวคิดในการแบ่งวงจรเช่นเดียวกับวงจรบวกปกติ แต่ระยะเวลาที่ใช้ในการทำงานแต่ละรอบจะเพิ่มขึ้นตามความความยาวของฐาน ดังนั้นระยะเวลาในการบวกทั้งหมดจึงขึ้นกับผลคูณของความยาวของฐาน และจำนวนแถวของรูปแบบแสดงผล เมื่อทำการจำลองวงจรก็พบว่าทำงานได้ถูกต้อง

ในขั้นตอนสุดท้าย เป็นการนำแนวคิดของการทำงานแบบสายท่อมาประยุกต์เข้ากับกระบวนการบวกทั่วไป โดยสิ่งที่จะต้องปรับปรุงนั้นมีเพียงขั้นตอนการรวมค่า รวมทั้งข้อมูลที่นำเข้า และส่งออกเล็กน้อย เนื่องจากมีจำนวนที่จะต้องนำมาบวกกันเพิ่มขึ้น จากการที่รับข้อมูลตัวบวกใหม่เข้ามาเรื่อย ๆ ในส่วนการออกแบบวงจรนั้นจะคล้ายคลึงกับวงจรบวกทั่วไป แต่เพิ่มขั้นตอนขึ้นหนึ่งขั้น เพื่อจำกัดค่าของตัวทดให้เป็นไปดังที่ต้องการ ความเร็วที่เพิ่มขึ้นจะปรากฏเมื่อมีการบวกหลาย ๆ จำนวนต่อเนื่องกัน โดยจะใช้ระยะเวลาขึ้นกับจำนวนตัวที่นำมาบวก หรือจำนวนแถวของรูปแบบแสดงค่าเท่านั้น จากเดิมที่ระยะเวลาที่ใช้นั้นขึ้นอยู่กับผลคูณของจำนวนทั้งสอง

สำหรับตัวปฏิบัติการคูณนั้นจะนำตัวบวกที่สามารถทำงานแบบสายท่อได้ มาเป็นส่วนประกอบหลัก โดยจะมีส่วนประกอบอื่นเพิ่มเติม คือ วงจรที่ใช้ในการหาตำแหน่งที่มีค่าของตัวคูณ วงจรเลื่อนตำแหน่งของตัวตั้ง และส่วนควบคุม เพื่อให้การทำงานทุกส่วนเป็นไปได้อย่างสัมพันธ์กัน ระยะเวลาที่ใช้ในการคูณนั้นจึงขึ้นอยู่กับจำนวนตำแหน่งที่มีค่าของตัวคูณ หรือ จำนวนแถวของรูปแบบแสดงค่า

ข้อเสนอแนะ

งานวิจัยนี้ยังสามารถทำการพัฒนาได้อีกในหลายทาง ได้แก่ การศึกษาความสัมพันธ์ระหว่างฐาน และสมบัติของระบบจำนวนฐานคู่เพิ่มเติม รวมทั้งหาค่าของฐานที่เหมาะสมสำหรับงานแต่ละชนิด หาวิธีคำนวนระดับความซ้ำซ้อนของระบบจำนวนที่มีประสิทธิภาพมากยิ่งขึ้น เนื่องจากระดับความซ้ำซ้อนนั้นไม่ได้ขึ้นกับค่าของฐานเพียงอย่างเดียว แต่ยังขึ้นกับรูปร่าง และขนาดของรูปแบบแสดงค่าของระบบจำนวนด้วย

การบวกนั้นสามารถทำการพัฒนาให้มีความเร็วมากยิ่งขึ้นได้ โดยการรวม หรือลดขั้นตอนที่ไม่จำเป็นบางขั้นตอนออก หรือนำวิธีการอื่นมาประยุกต์ใช้ เช่น ประยุกต์สถาปัตยกรรมออนเดอะฟลาย [12] (On-the-fly) ซึ่งจะมีการคำนวนผลลัพท์บางส่วนไว้ล่วงหน้าก่อนที่จะต้องบวกจริง นอกจากนี้การบวกที่เสนอไปยังมีสิ่งที่ถูกละเลยอีก คือ เรื่องของส่วนล้น ซึ่งส่วนล้นบางตัวเป็นส่วนล้นปลอม คือ ค่าจากรูปแบบแสดงค่าที่มีส่วนล้นนั้น จะมีรูปแบบแสดงค่าอื่นที่สามารถแสดงค่านั้น ๆ ได้ภายในขอบเขตที่กำหนด หรือ พัฒนาส่วนที่ไปรอบรับส่วนล้นเหล่านั้น โดยประยุกต์จากสิ่งที่มีเสนอมาแล้วในเรื่องตัวเลขป้องกัน [13] เพื่อพิจารณาอีกครั้งว่าส่วนล้นนั้นเป็นส่วนล้นจริงหรือไม่ ถ้าเป็นส่วนล้นปลอม จะต้องมีกระบวนการในการเปลี่ยนรูปแบบให้ไปอยู่ในรูปแบบแสดงค่าที่ถูกต้องได้

สำหรับการคูณนั้นสามารถจะทำการเพิ่มความเร็วได้มาก เพราะการคูณนั้นทำให้เกิดการเลื่อนของตัวตั้งไปตามแนวแกนตั้ง ทำให้แถวที่มีค่าประจำแถวน้อยมีค่าเป็น 0 เสมอ ซึ่งจะเสียเวลาโดยเปล่าประโยชน์เมื่อส่งเข้าไปยังการบวกแบบสายท่อ ทั้งที่การบวกของแถวเหล่านั้นไม่ก่อให้เกิดการทดระหว่างแถวอย่างแน่นอน รวมทั้งอาจจะยังสามารถเปลี่ยนรูปแบบของตัวคูณให้มีจำนวนตำแหน่งที่มีค่าลดลงได้อีกด้วย

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

รายการอ้างอิง

[1] Avizienis, A. Signed-Digit Number Representations for Fast Parallel Arithmetic. IRE Transaction on Electronic Computers 10 (1961): 389-400.

[2] Dimitrov, V.S., G.A. Jullien, and W.C. Miller. Theory and applications for a Double-Base Number System. The 13th IEEE Symposium on Computer Arithmetic (1997): 44-51.

[3] Dimitrov, V.S., G.A. Jullien, and W.C. Miller. Theory and applications of the double-base number system. IEEE Transactions on Computers 10 (1999): 1098-1106.

[4] Dimitrov, V.S., and G.A. Jullien, Loading the bases: a new number representation with applications. IEEE Circuits and Systems Magazine 3 (2003): 6-23.

[5] Pankaala, M., A. Paasio, and M. Laiho. Implementation alternatives of a DBNS adder. International Workshop on Cellular Neural Network and Their Applications (2005): 138-141.

[6] Ibrahim, Y., et al. DBNS addition using cellular neural networks. IEEE International Symposium on Circuits and Systems 4 (2005): 3914-3917.

[7] Berthé, V., L. Imbert, and G.A. Jullien, More on Converting Numbers to the Double-Base Number System, Research Report LIRMM-04031 (2004).

[8] Sadeghi-Emamchaie, S., G.A. Jullien, V. Dimitrov, and W.C. Miller. Digital arithmetic using analog arrays. Proceeding of the 8th Great Lakes Symposium on VLSI (1998): 202-205.

[9] Dimitrov, V.S., S. Sadeghi-Emamchaie, G.A. Jllien, and W.C. Miller. A near canonic double-based number system (DBNS) with applications in digital signal processing. Proc. SPIE, Advance Signal Processing Algorithms, Architectures, and Implementations VI (1996): 14-25.

[10] Gilbert, G., and J.M.P. Langlois. Multipath greedy algorithm for canonical representation of numbers in the double base number system. The 3rd International IEEE-NEWCAS Conference (2005): 39-42.

[11] Phantak, D.S., and I. Koren, Hybrid signed-digit number systems: a unified framework for redundant number representations with bounded carry propagation chains. IEEE Transactions on Computers 43 (1994): 880-891.

[12] Kornerup, P., Digit-set conversions: generalizations and applications. IEEE Transactions on Computers 43 (1994): 622-629.

[13] Kornerup, P., and J.M. Muller, Leading guard digits in finte precision redundant representations. IEEE Transactions on Computers 55 (2006): 541-548.

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ประวัติผู้เขียนวิทยานิพนธ์

นายเกรียงยุทธ หวังจิตมั่น เกิดเมื่อวันที่ 28 ตุลาคม พ.ศ. 2527 ที่จังหวัดกรุงเทพฯ สำเร็จการศึกษาระดับประถมศึกษา ถึงมัธยมศึกษาตอนปลายจากโรงเรียนเซนต์คาเบรียล สำเร็จการศึกษาระดับปริญญาบัณฑิต ในสาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมศาสตร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัยในปีการศึกษา 2548

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย