บทที่ 4
ผลการวิจัย

เนื้อหาในบทนี้อธิบายถึงผลการทดลองที่พิสูจน์ให้เห็นถึงประสิทธิภาพของระเบียบวิธีที่นำเสนอในโครงการวิจัยดังต่อไปนี้

4.1 ผลการวิเคราะห์จุดพาริโต

เพื่อที่จะพิสูจน์ว่าระเบียบวิธีที่นำเสนอสามารถสังเคราะห์จุดออกแบบที่เป็นจุดพาริโตที่เป็นไปได้ออกมาทั้งหมด การทดลองจึงถูกดำเนินการเปรียบเทียบกับระเบียบวิธีของ [Blythe00] ได้ผลการวิเคราะห์ดังตารางที่ 1 แต่ละแถวแสดงผลการสร้างวงจร Diff eq โดยเปรียบที่ resource-constraint ต่าง ๆ จากขนาดวงจร (area) เล็กจนถึงขนาดใหญ่ตามลำดับ ซึ่งแต่ละแถวก็คือแต่ละจุดออกแบบ D(area, speed) นั่นเอง ในที่นี้วงจรคูณ (M) มีขนาดพื้นที่เท่ากับ 150 หน่วย และมีดีเลย์เท่ากับ 163ns วงจร ALU (ภายในมี +,-,<) มีขนาดพื้นที่เท่ากับ 100 หน่วย และมีดีเลย์เท่ากับ 48ns

คอลัมน์ที่ 2 แสดงขนาดของพื้นที่ของแต่ละ resource-constraint คอลัมน์ที่ 3 แสดงผลจากการสังเคราะห์โดยวิธีของ [Blythe00] โดย "1045:19x55, P" หมายถึง เวลา DFG ทั้งหมดมีค่าเท่ากับ 1045ns ซึ่งประกอบด้วย คาบสัญญาณนาฬิกาขนาด 55ns จำนวน19 clock cycles และจุดออกแบบนี้เป็นจุดพาริโต สำหรับจุดออกแบบที่ไม่เป็นจุดพาริโตถูกแสดงด้วย "nP"

คอลัมน์ที่ 4 แสดงผลจากการสังเคราะห์โดยวิธีที่นำเสนอ โดย "1026: 6x163+48, P" หมายถึง เวลา DFG ทั้งหมดมีค่าเท่ากับ 1026ns ซึ่งประกอบด้วย คาบสัญญาณนาฬิกาขนาด 163ns จำนวน 6 clock cycles และคาบสัญญาณนาฬิกาขนาด 48ns จำนวน 1 clock cycle และจุดออกแบบนี้เป็นจุดพาริโต

ตารางที่ 1: ผลการวิเคราะห์จุดพาริโต

Resource-constraint	Area	[Blythe00] method	Proposed method
1M, 1ALU	250	1045:19x55, P	1026: 6x163+48, P
2M, 1ALU	400	605: 11x55, P	585: 3x163+2x48, P
2M, 2ALU	500	652: 4x163, nP	537: 3x163+48, P
3M, 1ALU	550	480: 20x24, P	470: 2x163+3x48, P
3M, 2ALU	650	432: 18x24, P	422: 2x163+2x48, P

จากตารางที่ 1 สังเกตเห็นว่าระเบียบวิธีที่นำเสนอสามารถสังเคราะห์จุดออกแบบที่เป็นจุดพาริโตออกมาได้หมด ในขณะที่ระเบียบวิธีของ [Blythe00] ระบุว่า จุดออกแบบ D(400, 605) ไม่เป็นจุดพาริโตเนื่องจากยังมีจุดออกแบบอื่นที่เร็วกว่านี้ แต่ไม่สามารถสร้างได้จริงเนื่องจากถูกจำกัดด้วยคาบสัญญาณนาฬิกา ระเบียบวิธีของ [Blythe00] จึงตัดจุดนี้ออกจากการเป็นจุดพาริโต นอกจากนี้สังเกตเห็นได้ว่าระเบียบวิธีที่นำเสนอยังให้วงจรที่ทำงานได้เร็วกว่า [Blythe00] ทุกจุดออกแบบ ทั้งนี้เป็นผลเนื่องจากการนำหลักการทางการออกแบบวงจรอะซิงโครนัสมาประยุกต์ใช้นั่นเอง

4.2 ผลการวิเคราะห์วงจรเบนช์มาร์ค

เพื่อที่จะพิสูจน์ถึงประสิทธิภาพของระเบียบวิธีการคำนวณคาบสัญญาณนาฬิกาที่นำเสนอ การทดลองระเบียบวิธีประยุกต์ใช้กับวงจร DSP benchmarks ได้ถูกดำเนินการและเปรียบเทียบกับระเบียบวิธีที่เกี่ยวข้องมากที่สุดคือ [Chang96] ดังแสดงในตารางที่ 2 คอลัมน์ที่ 1 แสดงชื่อของวงจร คอลัมน์ที่ 2 แสดงระเบียบวิธีที่เปรียบเทียบกัน คอลัมน์ที่ 3 แสดงคาบสัญญาณนาฬิกาที่คำนวณได้จากแต่ละวิธี คอลัมน์ที่ 4 แสดงจำนวน clock cycles ที่ใช้ในการทำงานจนเสร็จจากขั้นเวลาแรกจนถึงขั้นเวลาสุดท้าย และคอลัมน์ที่ 5 แสดงเวลาในการทำงานดังกล่าว

ตารางที่ 2: Benchmark results

Example	Method	Clock period (ns)	Clock cycles	DFG time(ns)
Diff Eq	Max. operation	90.9	4	363.6
(2M, 2A,	Slack minimization	3.134	101	316.6
1S, 1C)	Proposed method	90.9, 34.2	4	306.9
AR-lattice	Max. operation	90.9	10	909
(2M, 2A)	Slack minimization	2.597	356	916.8
	Proposed method	90.9, 33.7	10	794.6
EWF	Max. operation	90.9	16	1454.4
(2M, 2A)	Slack minimization	3.37	218	734.7
	Proposed method	23.5, 33.7	22	700.6
EWF	Max. operation	90.9	14	1272.6
(2M, 4A)	Slack minimization	3.37	208	701.0
	Proposed method	23.5, 33.7	22	700.6

ในตาราง Diff Eq (2M, 2A, 1S, 1C) หมายถึง วงจรแก้สมการอนุพันธ์ถูกสร้างด้วย 2 วงจรคูณ 2 วงจรบวก 1 วงจรลบ และ 1 วงจรเปรียบเทียบ ทำนองเดียวกันสำหรับ วงจรกรอง AR-lattice และ EWF ในที่นี้ดีเลย์ของโอเปอเรชันต่าง ๆ ถูกนำมาจาก [Chang96] ดังนี้ M=90.9ns A=33.7ns และ S=34.2ns

จากตารางสามารถสังเกตได้ว่า ระเบียบวิธีที่นำเสนอให้วงจรที่มีประสิทธิภาพมาก วงจรสามารถทำงานเสร็จด้วยจำนวน clock cycles ที่ใกล้เคียงกับระเบียบวิธี Max. operation ซึ่งเป็นวิธีที่เลือกคาบเวลาจากโอเปอเรชันที่ทำงานช้าที่สุด และวงจรที่นำเสนอยังมีความเร็วสูงที่สุด ถึงแม้ว่าระเบียบวิธี Slack minimization [Chang96] สามารถสังเคราะห์วงจรที่มีความเร็วใกล้เคียงจุดที่เหมาะสมที่สุด แต่คาบของสัญญาณนาฬิกาเล็กเกินไปทำให้วงจรควบคุม FSM มีขนาดใหญ่มากและต้องการหลาย clock cycles มากในการทำงานเสร็จ นั่นหมายถึงการใช้กำลังไฟฟ้าที่มากนั่นเอง

นอกจากนี้ระเบียบวิธีที่นำเสนอสามารถทำให้การใช้ทรัพยากรลดลงด้วย ดังแสดงในผลของวงจร EWF(2M, 2A) และ EWF(2M, 4A) ด้วยระเบียบวิธีอื่นผลของทั้งสองกรณีต่างกัน วงจร EWF(2M, 4A) ใช้ทรัพยากรมากกว่าจึงทำงานได้เร็วกว่า ในขณะที่วงจรที่ได้จากระเบียบวิธีที่นำเสนอให้ผลความเร็วและจำนวน clock cycles ที่เหมือนกันของวงจร EWF ทั้งสองกรณี นั่นคือการสร้างวงจร EWF ต้องการวงจรบวกเพียง 2 วงจรก็พอ เหตุผลที่ได้ผลดังกล่าวเนื่องจากระบียบวิธี ASAP scheduling แบบอะซิงโครนัส ไม่ถูกจำกัดด้วยคาบสัญญาณนาฬิกา ทุกโอเปอเรชันสามารถเริ่มทำงานได้ทันทีที่ทรัพยากรและข้อมูลพร้อม