การทวนสอบวงจรอสมวารแบบควอไซดีเลย์อินเซนซิทีฟโดยพีชคณิตเชิงกระบวนการ

นาย วิฑูรย์ จันทรเศรษฐเลิศ

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2544
ISBN 974-03-0823-6
ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

A VERIFICATION OF QUASI-DELAY-INSENSITIVE ASYNCHRONOUS CIRCUITS BY THE PROCESS ALGEBRA

Mr. Vitoon Chandarasetalert

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science in Computer Science

Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University

Academic Year 2001
ISBN 974-03-0823-6

หัวข้อวิทยานิพนธ์	การทวนสอบวงจรอสมวารแบบควอไซดีเลย์อินเซนซิทีฟโดยพีชคณิตเชิงกระบวนการ
โดย	นาย วิฑูรย์ จันทรเศรษฐเลิศ
สาขาวิชา	วิทยาศาสตร์คอมพิวเตอร์
อาจารย์ที่ปรึกษา	อาจารย์ ดร.อาทิตย์ ทองทักษ์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์
(ศาสตราจารย์ ดร.สมศักดิ์ ปัญญาแก้ว)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(รองศาสตราจารย์ ดร.ประภาส จงสถิตย์วัฒนา)

..... อาจารย์ที่ปรึกษา
(อาจารย์ ดร.อาทิตย์ ทองทักษ์)

..... กรรมการ
(ผู้ช่วยศาสตราจารย์ บุญชัย โสวรรณวณิชกุล)

..... กรรมการ
(ผู้ช่วยศาสตราจารย์ วิวัฒน์ วัฒนาวุฒิ)

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

วิฑูรย์ จันทรเศรษฐเลิศ : การทวนสอบวงจรอสมวารแบบควอไซดีเลย์อินเซนซิทีฟโดยพีชคณิตเชิงกระบวนการ (A VERIFICATION OF QUASI-DELAY-INSENSITIVE ASYNCHRONOUS CIRCUITS BY THE PROCESS ALGEBRA) อ. ที่ปรึกษา : ดร. อาทิตย์ ทองทักษ์, 95 หน้า. ISBN 974-03-0823-6.

การทวนสอบวงจรเป็นกระบวนการที่ใช้ตรวจสอบความถูกต้องตรงกันระหว่างคุณลักษณะของวงจรที่ได้ออกแบบไว้ กับวงจรที่ได้จากการสังเคราะห์ วิทยานิพนธ์ฉบับนี้นำเสนอการออกแบบ และพัฒนาขั้นตอนวิธีการทวนสอบวงจรอสมวารแบบควอไซดีเลย์อินเซนซิทีฟโดยพีชคณิตเชิงกระบวนการ

ขั้นตอนวิธีการทวนสอบวงจรอสมวารที่นำเสนอในงานวิจัยฉบับนี้เป็นการทวนสอบอย่างมีแบบแผนแบบตรวจสอบแบบจำลองโดยใช้เครื่องจักรสถานะจำกัด ซึ่งทั้งในส่วนของคุณลักษณะของวงจรที่ได้ออกแบบไว้ และวงจรที่ได้จากการสังเคราะห์จะถูกแปลงให้อยู่ในรูปของเครื่องจักรสถานะจำกัดโดยใช้กราฟสถานะ และเทคนิคของพีชคณิตเชิงกระบวนการ โดยที่ในขั้นตอนการสร้างเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ยังได้คำนึงถึงโมเดลสิ่งแวดล้อมที่ได้จากคุณลักษณะของวงจรเพื่อเป็นตัวควบคุมการเกิดการเปลี่ยนแปลงสัญญาณอินพุตของวงจรที่ได้จากสังเคราะห์ จากนั้นจะทำการทวนสอบด้วยการเปรียบเทียบว่าวงจรที่ได้จากการสังเคราะห์มีพฤติกรรมถูกต้องตรงตามคุณลักษณะของวงจรที่ได้ออกแบบไว้หรือไม่โดยใช้วิธีการค้นหาแบบกว้าง

จากผลการทดลองกับวงจรเกณฑ์เปรียบเทียบสมรรถนะแสดงให้เห็นว่าขั้นตอนวิธีการทวนสอบมีประสิทธิภาพ 93.10% สามารถทวนสอบวงจรได้ 27 วงจรจากทั้งหมด 29 วงจร

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาควิชา	วิศวกรรมคอมพิวเตอร์	**ลายมือชื่อนิสิต**	
สาขาวิชา	วิทยาศาสตร์คอมพิวเตอร์	**ลายมือชื่ออาจารย์ที่ปรึกษา**	
ปีการศึกษา	2544	**ลายมือชื่ออาจารย์ที่ปรึกษาร่วม**	

##4170519121 : MAJOR COMPUTER ENGINEERING
KEYWORD: VERIFICATION / ASYNCHRONOUS CIRCUITS / QUASI-DELAY-INSENTITIVE / PROCESS ALGEBRA
VITOON CHANDARASETALERT : A VERIFICATION OF QUASI-DELAY-INSENSITIVE ASYNCHRONOUS CIRCUITS BY THE PROCESS ALGEBRA.
THESIS ADVISOR : ARTHIT THONGTAK, Ph.D., 95 pp. ISBN 974-03-0823-6.

The circuit verification is a process to assure the correctness between the implemented circuit and its specification. This thesis proposes a design and development of verification process for quasi-delay-insensitive (QDI) asynchronous circuits by process algebra.

Our approach is the model checking by finite state machine (FSM) in formal verification method. Both of the specification and implementation will be used to create a finite state machine by using state graph and process algebra technique. In the procress of creating finite state machine of implemented circuit, we also consider the environment model got from specification to control input signal transition of implemented circuit. Then the behavior of the resulting implementation will be compared to its specification by using the breadth-first search method.

Experiments on the benchmark circuits show that verification process had efficiency 93.10% and can verify 27 circuits from 29 circuits

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

Department	Computer Engineering	**Student's signature**	
Field of study	Computer Science	**Advisor's signature**	
Academic year	2001	**Co-Advisor's signature**	

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยความช่วยเหลืออย่างดียิ่งของอาจารย์ ดร.อาทิตย์ ทองทักษ์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ ซึ่งท่านได้ให้คำแนะนำ และข้อคิดเห็นต่างๆ ในการวิจัยมาด้วยดีตลอด

ขอขอบคุณ คุณอริสสรา สุวรรณจีระ คุณโชติมา โอวะกุลวงศ์ คุณมนต์รวี เชียรชัยนิรัติศัย ที่ให้คำปรึกษาที่ดีในด้านต่างๆและเป็นกำลังใจที่ดีเสมอมา

ขอขอบคุณ สมาชิก Digital System Engineering Laboratory ทุกท่านที่ให้ความช่วยเหลือด้านคำปรึกษาและข้อเสนอแนะในการทำวิทยานิพนธ์

ขอขอบคุณ ห้อง Digital System Engineering Laboratory ที่ให้สถานที่พักพิง และเครื่องมือที่ใช้ในการวิจัย

ขอขอบคุณ เพื่อนๆ ทุกๆ คนที่ได้ให้คำปรึกษา และความช่วยเหลือในด้านต่างๆ ซึ่งทำให้ทำงานวิจัยได้อย่างราบรื่น

และท้ายนี้ ผู้วิจัยใคร่ขอกราบขอบพระคุณ บิดา-มารดา ที่ให้การสนับสนุน และให้กำลังใจแก่ผู้วิจัยเสมอมาจนสำเร็จการศึกษา

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ง

บทคัดย่อภาษาอังกฤษ ..... จ

กิตติกรรมประกาศ ..... ฉ

สารบัญ ..... ช

สารบัญภาพ ..... ฌ

สารบัญตาราง ..... ฐ

บทที่

1. บทนำ 1
   - 1.1 การทวนสอบวงจร 1
   - 1.2 ความเป็นมาและความสำคัญของปัญหา 4
   - 1.3 วัตถุประสงค์ของการวิจัย 5
   - 1.4 ขอบเขตของการวิจัย 5
   - 1.5 ขั้นตอนและวิธีดำเนินงานวิจัย 5
   - 1.6 ประโยชน์ที่คาดว่าจะได้รับ 5
   - 1.7 เนื้อหาของวิทยานิพนธ์ 6

2. ทฤษฎีและงานวิจัยที่เกี่ยวข้อง 7
   - 2.1 โมเดลความหน่วงแบบควอไซดีเลย์อินเซนซิทีฟ 7
   - 2.2 ซิกแนลทรานสิชันกราฟ 8
   - 2.3 State Graph 12
   - 2.4 พีชคณิตเชิงกระบวนการ 13
   - 2.5 สรุป 16

3. การทวนสอบวงจรอสมวารอย่างมีแบบแผนแบบ Model Checking 17
   - 3.1 เครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร 17
   - 3.2 เครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ 18
   - 3.3 การทวนสอบ 20
   - 3.4 สรุป 22

สารบัญ (ต่อ)

บทที่ หน้า

4. เครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร 23
   4.1 การสร้างเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร 23
   4.2 สรุป 35

5. เครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ 36
   5.1 กราฟพฤติกรรมของเกต 36
   5.2 โมเดลสิ่งแวดล้อม 37
   5.3 การสร้างเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ 39
   5.4 สรุป 56

6. การทวนสอบวงจร 57
   6.1 การตรวจสอบความสัมพันธ์ระหว่างเครื่องจักรสถานะจำกัด 57
   6.2 ตัวอย่างการทวนสอบวงจร C-element 59
   6.3 ตัวอย่างการทวนสอบวงจรที่ไม่ถูกต้องตรงตามคุณลักษณะที่ได้ออกแบบไว้ 61
   6.4 ผลการทดลอง 63
   6.5 สรุป 65

7. สรุปผลการวิจัยและข้อเสนอแนะ 66
   7.1 สรุปผลการวิจัย 66
   7.2 ข้อจำกัด 66
   7.3 ข้อเสนอแนะ 67

รายการอ้างอิง 68

ภาคผนวก

วงจรเกณฑ์เปรียบเทียบสมรรถนะ 70

ตัวอย่างเครื่องจักรสถานะจำกัดและกราฟโมเดลสิ่งแวดล้อมของวงจรเกณฑ์เปรียบเทียบสมรรถนะ 85

ประวัติผู้เขียนวิทยานิพนธ์ 95

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญภาพ

ภาพประกอบ		หน้า
รูปที่ 1.1	การทวนสอบโดยวิธีจำลอง	1
รูปที่ 1.2	การทวนสอบอย่างมีแบบแผน	2
รูปที่ 1.3	สเปคตรัมของเทคนิคต่างๆ ที่ใช้ทวนสอบวงจร	3
รูปที่ 2.1	(a) เกต C-element ขนาด 2 อินพุต (b) สัญลักษณ์ที่ใช้แทนเกต C-element ขนาด 2 อินพุต	8
รูปที่ 2.2	ตัวอย่างซิกแนลทรานสิชันกราฟ	8
รูปที่ 2.3	ตัวอย่างซิกแนลทรานสิชันกราฟที่มีคุณสมบัติ input choice	10
รูปที่ 2.4	ตัวอย่างซิกแนลทรานสิชันกราฟที่ไม่มีคุณสมบัติ (a) Liveness (b) Safety (c) Persistency (d) Consistency (e) Unique state assignment และ Single cycle transition	12
รูปที่ 2.5	State Graph ของซิกแนลทรานสิชันกราฟที่อ้างอิงจากรูปที่ 2.2	12
รูปที่ 2.6	Termination	13
รูปที่ 2.7	Guarding	13
รูปที่ 2.8	Choice	14
รูปที่ 2.9	Nondeterminism	14
รูปที่ 2.10	Abstraction	14
รูปที่ 2.11	Concurrent Composition	15
รูปที่ 3.1	การทวนสอบวงจรอสมวารอย่างมีแบบแผนแบบ Model Checking	17
รูปที่ 3.2	(a) ซิกแนลทรานสิชันกราฟของวงจร C-element (b) กราฟสถานะของวงจร C-element	18
รูปที่ 3.3	รูปแบบนิพจน์ของวงจร C-element ที่ได้จากการสังเคราะห์	19
รูปที่ 3.4	(a) กราฟแสดงพฤติกรรมของนิพจน์ or2(Ai, P2, P1, s0) (b) กราฟแสดงพฤติกรรมของ Causal Delay (c) กราฟแสดงพฤติกรรมของนิพจน์ or2(Ai, P2, P1, s0) หลังจากเพิ่ม Causal Delay เข้าที่สัญญาณขาออก	19
รูปที่ 3.5	(a) เครื่องจักรสถานะจำกัด A (b) เครื่องจักรสถานะจำกัด B	20
รูปที่ 4.1	Berkeley Logic Interchange Format	23
รูปที่ 4.2	ไฟล์อินพุตซิกแนลทรานสิชันกราฟของวงจร C-element	24
รูปที่ 4.3	โครงสร้างข้อมูลของซิกแนลทรานสิชันกราฟ และแถวลำดับ marking ของวงจร C-element ที่อ้างอิงจากรูปที่ 4.2	25

สารบัญภาพ (ต่อ)

ภาพประกอบ		หน้า
รูปที่ 4.4	ตัวอย่างการวิเคราะห์การเปลี่ยนแปลงสัญญาณเริ่มต้น (a) กรณีที่ 1 (b) กรณีที่ 2.1 (c) กรณีที่ 2.2 (c) กรณีที่ 2.3	27
รูปที่ 4.5	ขั้นตอนวิธีการสร้างเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร C - element	32
รูปที่ 4.6	เครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร C-element ในรูปแบบของกราฟ (a) และโครงสร้างของข้อมูล (b)	34
รูปที่ 5.1	(a) รูปแบบของเท็กไฟล์ของวงจรที่ได้จากการสังเคราะห์ (b) รูปแบบของนิพจน์ในเท็กไฟล์	36
รูปที่ 5.2	กราฟพฤติกรรมของเกต OR ขนาด 2 อินพุต	37
รูปที่ 5.3	(a) กราฟพฤติกรรมของ causal delay (b) กราฟพฤติกรรมของ isochronic fork	37
รูปที่ 5.4	(a) เครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร C-element (b) กราฟโมเดลสิ่งแวดล้อมของวงจร C-element	38
รูปที่ 5.5	(a) วงจรที่ได้จากการสังเคราะห์ (b) ไฟล์อินพุตของวงจรที่ได้จากการสังเคราะห์	39
รูปที่ 5.6	กราฟพฤติกรรมของเกต OR ในวงจร C-element ในรูปแบบของกราฟ (a) และโครงสร้างของข้อมูล (b)	40
รูปที่ 5.7	กราฟพฤติกรรมของ causal delay ที่ใช้ในการเพิ่ม causal delay ให้กับสัญญาณขาออก P1 ของเกต OR ในวงจร C-element ในรูปแบบของกราฟ (a) และโครงสร้างของข้อมูล (b)	42
รูปที่ 5.8	ขั้นตอนวิธีการสร้างกราฟสายงานแสดงพฤติกรรมของเกตที่เพิ่ม causal delay ให้กับสัญญาณขาออก P1 ของเกต OR ในวงจร C-element	45
รูปที่ 5.9	กราฟแสดงพฤติกรรมของเกต OR ที่เพิ่ม causal delay	46
รูปที่ 5.10	กราฟสายงานแสดงพฤติกรรมของเกตที่เพิ่ม causal delay แล้วทั้งหมดในวงจร C-element	47
รูปที่ 5.11	ขั้นตอนวิธีการสร้างเครื่องจักรสถานะจำกัดของวงจร C-element ที่ได้จากการสังเคราะห์	53
รูปที่ 5.12	เครื่องจักรสถานะจำกัดของวงจร C-element ที่ได้จากการสังเคราะห์ ในรูปแบบของกราฟ (a) และโครงสร้างของข้อมูล (b)	55

สารบัญภาพ (ต่อ)

ภาพประกอบ		หน้า
รูปที่ 6.1	FSM Specification	60
รูปที่ 6.2	FSM Implementation	60
รูปที่ 6.3	วงจร C-element ที่เกิดความผิดพลาดในกรณีต่างๆ	61
รูปที่ 6.4	FSM ของวงจร C-element ที่เกิดความผิดพลาดในกรณีต่างๆ อ้างอิงจากรูปที่ 6.3	62

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญตาราง

ตาราง		หน้า
ตารางที่ 2.1	ตารางแสดงสถานะของเกต C-element	8
ตารางที่ 4.1	ตารางเทียบค่าระหว่าง state กับ encode signal สำหรับเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร C-element	33
ตารางที่ 5.1	พฤติกรรมของเกต OR ขนาด 2 อินพุต	36
ตารางที่ 5.2	ตารางเทียบค่าระหว่าง new state กับ old state สำหรับกราฟสายงานของเกต OR ที่เพิ่ม causal delay แล้ว	46
ตารางที่ 5.3	ตารางเทียบค่าระหว่าง new state กับ old state สำหรับเครื่องจักรสถานะจำกัดของวงจร C-element ที่ได้จากการสังเคราะห์	54
ตารางที่ 6.1	ผลการทดลองการทวนสอบกับวงจรเกณฑ์เปรียบเทียบสมรรถนะ	63

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 1
บทนำ

1.1 การทวนสอบวงจร

การทวนสอบวงจร (Circuit Verification) เป็นกระบวนการในการตรวจสอบความถูกต้องตรงกันระหว่างคุณลักษณะของวงจร (Specification) ที่ได้ออกแบบไว้กับวงจรที่ได้จากการสังเคราะห์ (Implementation)

ในปัจจุบันการทวนสอบสามารถแบ่งออกได้เป็น 3 วิธีคือ

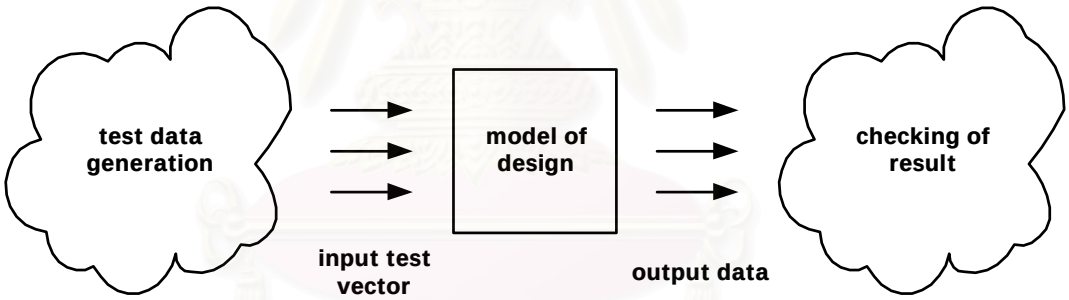
1.การทวนสอบโดยใช้วิธีจำลอง [5] (Simulation Verification)

2. การทวนสอบอย่างมีแบบแผน [3] (Formal Verification)

3. การทวนสอบอย่างกึ่งมีแบบแผน [4] (Semi Formal Verification)

1.) การทวนสอบโดยใช้วิธีจำลอง (Simulation Verification)

การทวนสอบโดยใช้วิธีการจำลองเป็นวิธีการทวนสอบที่นิยมใช้กันอย่างแพร่หลายซึ่งจะทำการทวนสอบโดยใช้วิธีการจำลองพฤติกรรมของวงจร ดังแสดงในรูปที่ 1.1



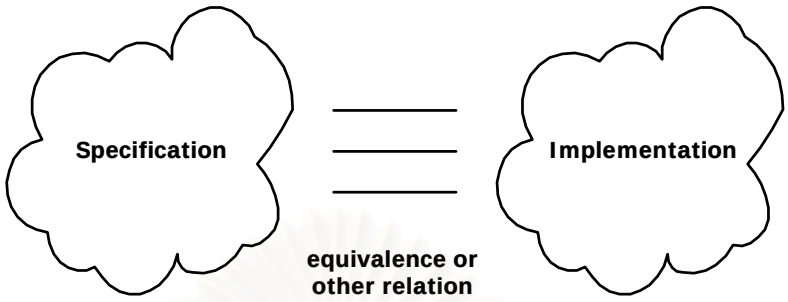
รูปที่ 1.1 การทวนสอบโดยใช้วิธีจำลอง

การทวนสอบโดยใช้วิธีจำลองจำเป็นต้องสร้างเวกเตอร์ทดสอบ ซึ่งได้จากการพิจารณาคุณลักษณะของวงจร และเวกเตอร์ทดสอบนี้จะถูกนำไปเป็นเวกเตอร์ทดสอบอินพุตของแบบจำลองที่ได้จากการออกแบบเพื่อจำลองพฤติกรรมของวงจร และจะนำเวกเตอร์ทดสอบไปเปรียบเทียบกับข้อมูลเอาต์พุตที่ได้จากการจำลองพฤติกรรมของวงจร เพื่อตรวจสอบความถูกต้องตรงกันระหว่างคุณลักษณะของวงจรกับวงจรที่ได้จากการสังเคราะห์

2.) การทวนสอบอย่างมีแบบแผน (Formal Verification)

การทวนสอบอย่างมีแบบแผนจะนำเทคนิคทางคณิตศาสตร์มาใช้ในการพิสูจน์ โดยที่ทั้งคุณลักษณะของวงจรกับวงจรที่ได้จากการสังเคราะห์ จะถูกแสดงในรูปแบบของระบบที่มีแบบแผน

อย่างแน่นอน และทั้งสองจะถูกนำมาพิสูจน์ว่ามีความสัมพันธ์แบบสมมูลกันหรือไม่ (equivalence) หรืออาจจะพิจารณาความสัมพันธ์ในรูปแบบอื่นที่สามารถแสดงได้ว่าวงจรที่ได้จากการสังเคราะห์มีพฤติกรรมถูกต้องตรงตามคุณลักษณะของวงจรที่ได้ออกแบบไว้ ดังแสดงในรูปที่ 1.2



รูปที่ 1.2 การทวนสอบอย่างมีแบบแผน

การทวนสอบอย่างมีแบบแผนยังแบ่งย่อยออกได้เป็น 3 วิธีคือ

2.1.) Equivalence Checking เป็นวิธีที่ใช้กันอย่างแพร่หลายโดยจะทำการเปรียบเทียบระหว่าง RTL (Register Transfer Language) กับวงจรในระดับเกต ตัวอย่างที่ถูกนำมาใช้ได้แก่ Abstract, Avant, Cadence, Chrysallis, Synopsys, Verplex, Verysys เป็นต้น

2.2) Model Checking ใช้เครื่องจักรสถานะจำกัด (Finite State Machine; FSM) เป็นแบบจำลองแสดงพฤติกรรมของวงจร เพื่อใช้ทวนสอบคุณสมบัติที่ต้องการด้วยการค้นหาภายใน state space ของเครื่องจักรสถานะจำกัดนั้น ในปัจจุบันการทวนสอบแบบ model checking แบ่งออกเป็น 2 แนวความคิดคือ

2.2.1.) Temporal Model Checking วิธีนี้คุณลักษณะของวงจรจะถูกแสดงในรูปของ temporal logic โดย logic ดังกล่าวมีลักษณะเป็นนิพจน์ที่แสดงพฤติกรรมของวงจรตามเงื่อนไขของเวลาที่กำหนดด้วยตัวปฏิบัติการต่างๆ ส่วนวงจรที่ได้จากการสังเคราะห์จะแสดงในลักษณะของ state transition graph ดังนั้นเราสามารถที่จะทำการทวนสอบด้วยการเปรียบเทียบคุณลักษณะของวงจรกับวงจรที่ได้จากการสังเคราะห์ได้ ตัวอย่าง model checker ที่เป็นที่รู้จักกันมากได้แก่ SMV เป็นต้น

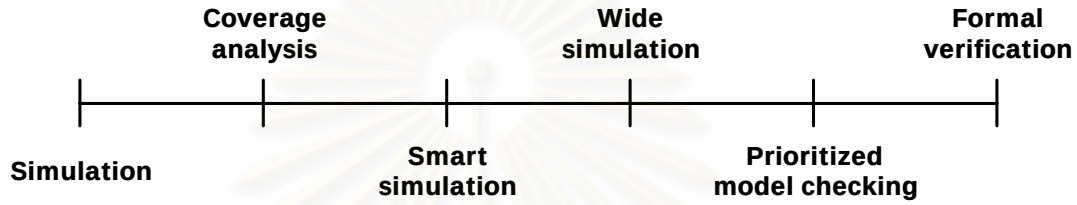
2.2.2.) Finite State Machine (FSM) สำหรับวิธีนี้ทั้งคุณลักษณะของวงจร และพฤติกรรมของวงจรที่ได้จากการสังเคราะห์จะถูกแสดงในรูปของเครื่องจักรสถานะจำกัด และจะถูกทำการทวนสอบด้วยการเปรียบเทียบว่ามีความสัมพันธ์แบบสอดคล้องกันหรือไม่ ตัวอย่างเช่น trace theory ของ Dill เป็นต้น

2.3.) Theorem Proving เป็นวิธีที่ไม่ค่อยนิยมนัก ในการทวนสอบทั้งคุณลักษณะของวงจรและพฤติกรรมของวงจรที่ได้จากการสังเคราะห์จะถูกแสดงในรูปของนิพจน์ของระบบที่มีแบบแผนอย่างแน่นอน ได้แก่ first order predicate logic, high order logic

เป็นต้น และจะต้องทำการพิสูจน์โดยใช้ axiom และ inference rule ของระบบที่มีแบบแผนอย่างแน่นอนเพื่อแสดงให้เห็นว่าพฤติกรรมของวงจรที่ได้จากการสังเคราะห์ถูกต้องตรงตามคุณลักษณะของวงจรที่ได้ออกแบบไว้ ตัวอย่างเช่น NQTHM theorem prove ของ Boyer – Moore, HOL theorem prover เป็นต้น

3.) การทวนสอบอย่างกึ่งมีแบบแผน (Semi Formal Verification)

เป็นวิธีการทวนสอบที่นำเอาแนวความคิดของการทวนสอบโดยใช้วิธีจำลอง และแนวความคิดของการทวนสอบอย่างมีแบบแผนมารวมกัน ดังแสดงในรูปที่ 1.3



รูปที่ 1.3 สเปคตรัมของเทคนิคต่างๆ ที่ใช้ทวนสอบวงจร

จากรูปที่ 1.3 จะเห็นว่าที่จุดปลายทั้งสองข้างของสเปคตรัมเป็นการทวนสอบโดยใช้วิธีการจำลอง และการทวนสอบอย่างมีแบบแผน ดังนั้นในช่วงระหว่างจุดปลายทั้งสองของสเปคตรัมจะเป็นการทวนสอบอย่างกึ่งมีแบบแผนสามารถแบ่งออกได้เป็น 4 วิธีคือ

3.1.) Coverage Analysis เป็นการวิเคราะห์เวกเตอร์ทดสอบที่จะถูกสร้างขึ้นโดยเน้นการใช้เวกเตอร์ทดสอบให้น้อยที่สุด แต่สามารถทวนสอบได้ครอบคลุมพฤติกรรมทั้งหมดของวงจร

3.2.) Smart Simulation เป็นวิธีการทวนสอบที่จะสร้างเวกเตอร์ทดสอบขึ้นอย่างอัตโนมัติบนพื้นฐานของ coverage metric โดยเวกเตอร์ทดสอบที่ถูกสร้างขึ้นอาจเป็นลักษณะ offline หรือ online ก็ได้

3.3.) Wide Simulation เป็นวิธีการทวนสอบที่นำสัญลักษณ์มาใช้แทนเซตของ state เพื่อลดจำนวนของ state space

3.4.) Prioritized Model Checking จะคล้ายกับการทวนสอบอย่างมีแบบแผนแบบ model checking แต่จะแตกต่างกันตรง model checking นั้นวิธีการค้นหาใน state space จะเป็นการค้นหาเส้นทางที่เป็นไปได้ทั้งหมด ส่วน prioritized model checking จะค้นหาโดยการใช้ฮิวริสติกเพื่อนำไปสู่เส้นทางที่ดีที่สุดในการทวนสอบวงจร

1.2 ความเป็นมาและความสำคัญของปัญหา

วงจรดิจิตอลแบ่งออกเป็น 2 ประเภทได้แก่ วงจรแบบสมวาร (Synchronous Circuits) และวงจรแบบอสมวาร (Asynchronous Circuits) โดยข้อแตกต่างระหว่างวงจร 2 ประเภทนี้คือวงจรแบบสมวารจะมีสัญญาณนาฬิกา (Clock Signal) เป็นตัวควบคุมการทำงานของวงจร ดังนั้นการออกแบบวงจรแบบสมวารจึงมีความยืดหยุ่นกว่าวงจรแบบอสมวาร จึงทำให้วงจรแบบสมวารถูกใช้กันอย่างแพร่หลาย แต่ในปัจจุบันเทคโนโลยีด้านการออกแบบวงจรได้ถูกพัฒนาไปอย่างรวดเร็วทำให้วงจรแบบสมวารไม่เหมาะสมกับเทคโนโลยีสมัยใหม่ วงจรแบบอสมวารจึงเป็นอีกทางเลือกหนึ่งที่สำคัญที่นำมาใช้ออกแบบวงจรรวมความจุสูงมาก (Very Large Scale Integration; VLSI) เพื่อหลีกเลี่ยงปัญหาต่างๆ ที่เกิดขึ้นกับวงจรแบบสมวาร ได้แก่ปัญหา การแกว่งของสัญญาณนาฬิกา, ปัญหาการสิ้นเปลืองพลังงาน เป็นต้น [6], [7]

อย่างไรก็ตามพบว่าการออกแบบวงจรแบบอสมวารนั้นเป็นงานที่ยาก เพราะวงจรแบบอสมวารไม่มีสัญญาณนาฬิกาเพื่อควบคุมจังหวะการรับส่งข้อมูลภายในวงจร ดังนั้นจึงเป็นสาเหตุสำคัญที่ทำให้เกิด hazard ขึ้นในวงจร และเป็นผลให้วงจรทำงานผิดพลาด

ในการออกแบบวงจรแบบอสมวารโดยทั่วไปแล้วนิยมใช้ซิกแนลทรานสิชันกราฟ (Signal Transition Graph; STG) [2] ซึ่งเป็นเครื่องมือที่ใช้ในการกำหนดคุณลักษณะของวงจรเพื่อที่จะนำไปสังเคราะห์เป็นวงจรในขั้นตอนต่อไป แต่พบว่ามีบางวงจรที่ได้จากการสังเคราะห์มีพฤติกรรมไม่ถูกต้องตรงกับซิกแนลทรานสิชันกราฟที่ได้ออกแบบไว้ ดังนั้นจึงมีความจำเป็นต้องทำการทวนสอบระหว่างซิกแนลทรานสิชันกราฟที่ได้ออกแบบไว้กับวงจรที่ได้จากการสังเคราะห์

ในปัจจุบันวิธีการทวนสอบวงจรแบ่งออกเป็น 3 วิธีได้แก่ การทวนสอบโดยใช้วิธีจำลอง, การทวนสอบอย่างมีแบบแผน และการทวนสอบอย่างกึ่งมีแบบแผน ซึ่งจะพบว่าการทวนสอบโดยใช้วิธีจำลอง และการทวนสอบอย่างกึ่งมีแบบแผนยังคงมีข้อจำกัดอยู่ นั่นคือสำหรับการทวนสอบโดยใช้วิธีการจำลอง เวกเตอร์ทดสอบมักจะมีแนวโน้มเอียงเอนกับเวกเตอร์ทดสอบที่ตนสร้างขึ้นทำให้วงจรที่ผ่านการทวนสอบโดยใช้วิธีการจำลองจะถูกรับประกันความถูกต้องเฉพาะกับเวกเตอร์ทดสอบเท่านั้น ส่วนการทวนสอบอย่างกึ่งมีแบบแผนถึงแม้ว่าเวกเตอร์ทดสอบจะถูกสร้างขึ้นอย่างมีแบบแผนแต่ก็พบว่ามีปัญหาเรื่องจำนวนเวกเตอร์ทดสอบที่มากเกินไปทำให้เสียเวลาในการทวนสอบอย่างมาก

ด้วยเหตุผลดังกล่าวทำให้งานวิจัยฉบับนี้ จะนำเสนอการทวนสอบวงจรอสมวารแบบควอไซดีเลย์อินเซนซิทีฟ (Quasi Delay Insensitive; QDI) [7] โดยใช้วิธีการทวนสอบอย่างมีแบบแผน

1.3 วัตถุประสงค์ของการวิจัย

เพื่อออกแบบขั้นตอนวิธี และพัฒนาโปรแกรมการทวนสอบวงจรอสมวารแบบควอไซดีเลย์อินเซนซิทีฟโดยใช้วิธีการทวนสอบอย่างมีแบบแผน

1.4 ขอบเขตของการวิจัย

1. ทำการทวนสอบวงจรอสมวารแบบควอไซดีเลย์อินเซนซิทีฟเท่านั้น
2. ทำการทวนสอบวงจรอสมวารที่ซิกแนลทรานสิชันกราฟมีวากยสัมพันธ์ (syntax) และอรรถศาสตร์ (semantics) ถูกต้องเท่านั้น
3. ทำการทวนสอบวงจรอสมวารโดยใช้วิธีการทวนสอบอย่างมีแบบแผน
4. โปรแกรมการทวนสอบที่พัฒนาขึ้นมาจะทดสอบโดยใช้วงจร benchmark ของ Park เป็นกรณีศึกษา [8]

1.5 ขั้นตอนและวิธีการดำเนินงานวิจัย

1. ศึกษาซิกแนลทรานสิชันกราฟซึ่งเป็นเครื่องมือที่ใช้กำหนดคุณลักษณะของวงจรแบบอสมวาร
2. ศึกษาการสร้าง State Graph จากซิกแนลทรานสิชันกราฟ
3. ศึกษาพีชคณิตเชิงกระบวนการซึ่งเป็นเครื่องมือที่อธิบายพฤติกรรมของวงจรแบบอสมวารที่ได้จากการสังเคราะห์
4. ออกแบบขั้นตอนวิธี และพัฒนาโปรแกรมการสร้าง State Graph จากซิกแนลทรานสิชันกราฟ
5. ออกแบบขั้นตอนวิธี และพัฒนาโปรแกรมการสร้างกราฟที่อธิบายพฤติกรรมของวงจรแบบอสมวารที่ได้จากการสังเคราะห์
6. ออกแบบขั้นตอนวิธี และพัฒนาโปรแกรมการทวนสอบวงจรอสมวารโดยใช้วิธีการทวนสอบอย่างมีแบบแผน
7. ทดสอบโปรแกรมทวนสอบที่พัฒนาขึ้นกับวงจรเกณฑ์เปรียบเทียบสมรรถนะ
8. สรุปผลการวิจัย และจัดทำรายงานวิทยานิพนธ์

1.6 ประโยชน์ที่คาดว่าจะได้รับ

งานวิจัยฉบับนี้สามารถนำไปใช้เป็นแนวทางในการพัฒนาขั้นตอนวิธีการทวนสอบวงจรแบบอสมวารสำหรับโมเดลความหน่วงแบบอื่นๆ ต่อไป

1.7 เนื้อหาของวิทยานิพนธ์

เนื้อหาของวิทยานิพนธ์ฉบับนี้จะแบ่งออกเป็นเจ็ดบทย่อยๆ โดยบทที่ 1 จะเป็นบทนำ จากนั้นในบทที่ 2 จะกล่าวถึงทฤษฎีต่างๆ ที่เกี่ยวข้องกับงานวิจัย บทที่ 3 จะเป็นการอธิบายให้เห็นภาพรวมทั้งหมดของขั้นตอนวิธีการทวนสอบวงจรอสมวารอย่างมีแบบแผนแบบ model checking โดยใช้เครื่องจักรสถานะจำกัด ต่อมาในบทที่ 4 และบทที่ 5 จะกล่าวถึงขั้นตอนในการสร้างเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร และวงจรที่ได้จากการสังเคราะห์อย่างละเอียดตามลำดับ จากนั้นในบทที่ 6 จะกล่าวถึงขั้นตอนในการทวนสอบวงจรอสมวารอย่างมีแบบแผนแบบ model checking โดยใช้เครื่องจักรสถานะจำกัดอย่างละเอียด พร้อมตัวอย่างในการทวนสอบวงจรที่ถูกต้องตรง และไม่ถูกต้องตรงตามคุณลักษณะของวงจรที่ได้ออกแบบไว้ นอกจากนี้ยังได้แสดงผลการทดลองที่ได้จากการนำเอาขั้นตอนวิธีการทวนสอบดังกล่าวไปทดสอบกับวงจรเกณฑ์เปรียบเทียบสมรรถนะด้วย ในบทสุดท้ายบทที่ 7 จะเป็นบทสรุปผลการวิจัยรวมทั้งข้อคิดเห็น และข้อเสนอแนะต่างๆ ซึ่งควรนำไปปรับปรุงให้ดียิ่งขึ้น

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงทฤษฎี และงานวิจัยต่างๆ ที่นำมาประยุกต์ใช้เพื่อออกแบบ และพัฒนาขั้นตอนวิธีการทวนสอบวงจรอสมวารแบบควอไซดีเลย์อินเซนซิทีฟ โดยจะประกอบไปด้วย โมเดลความหน่วงแบบควอไซดีเลย์อินเซนซิทีฟ, ซิกแนลทรานสิชันกราฟ, State Graph และพีชคณิตเชิงกระบวนการ

2.1 โมเดลความหน่วงแบบควอไซดีเลย์อินเซนซิทีฟ

ในการออกแบบวงจรแบบอสมวารเราจำเป็นต้องกำหนดโมเดลความหน่วงที่ใช้ เพื่อตั้งสมมติฐานเกี่ยวกับค่าความหน่วงของเกต (gate delay) และค่าความหน่วงในสายสัญญาณ (wire delay)

โมเดลความหน่วงแบบควอไซดีเลย์อินเซนซิทีฟ (QDI) [7] เป็นโมเดลความหน่วงที่ใช้ในการออกแบบวงจรแบบอสมวาร โดยมีนิยามดังนี้

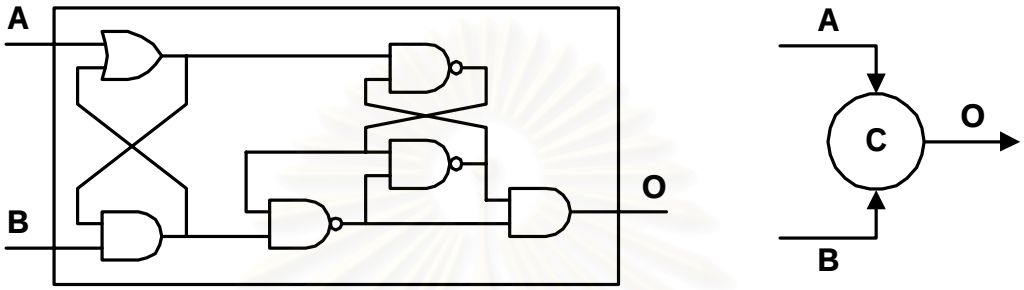
“ค่าความหน่วงของเกต และค่าความหน่วงของสายสัญญาณนั้นไม่ทราบค่าแต่รู้ว่ามีขอบเขตอยู่ที่ค่าหนึ่ง”

แต่ในการสังเคราะห์วงจรที่มีโมเดลความหน่วงแบบดีเลย์อินเซนซิทีฟ สามารถสังเคราะห์วงจรได้ในขอบเขตที่จำกัด ทำให้ไม่สามารถประยุกต์ใช้กับวงจรทั่วไปได้ ดังนั้นจึงได้มีการเพิ่มข้อจำกัดเกี่ยวกับความหน่วงขึ้นคือ ในกรณีที่มีการกระจายของสายสัญญาณ (fork) จะถือว่าค่าความหน่วงของสายสัญญาณแต่ละเส้นมีค่าเท่ากัน (isochronic fork) ซึ่งถือว่าเป็นรูปแบบของโมเดลความหน่วงแบบควอไซดีเลย์อินเซนซิทีฟ

นอกจากนี้ในวงจรแบบอสมวารที่มีโมเดลความหน่วงแบบดีเลย์อินเซนซิทีฟ และควอไซดีเลย์อินเซนซิทีฟ จำเป็นต้องใช้เกตพิเศษเพิ่มอีกตัวหนึ่งคือ เกต C-element โดยจะมีหน้าที่ตรวจสอบสัญญาณอินพุตว่ามีค่าตรงกันหรือไม่ ถ้ามีค่าตรงกันทั้งหมดก็จะให้ค่าของอินพุตนั้นเป็นค่าของเอาต์พุต สำหรับกรณีอื่นนั้นเอาต์พุตจะคงสภาพเดิม ดังแสดงในตารางที่ 2.1 และมีวงจรดังแสดงในรูปที่ 2.1

ตารางที่ 2.1 ตารางแสดงสถานะของเกต C-element

Input	Output
ทั้งหมดเป็น “0”	0
ทั้งหมดเป็น “1”	1
อื่นๆ	สภาวะเดิม

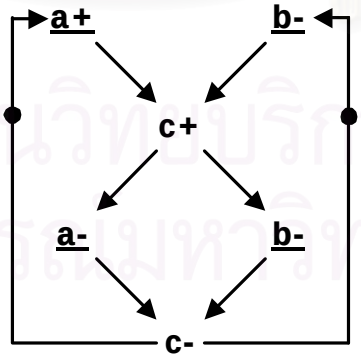


(a) (b)

รูปที่ 2.1 (a) เกต C-element ขนาด 2 อินพุต (b) สัญลักษณ์ที่ใช้แทนเกต C-element ขนาด 2 อินพุต

2.2 ซิกแนลทรานสิชันกราฟ

Signal Transition Graph (STG) เป็นเครื่องมือที่ใช้กำหนดคุณลักษณะของวงจรแบบอสมวารอย่างมีแบบแผน ซึ่งถูกพัฒนาโดย Tam-Anh Chu [2] STG มีลักษณะเป็นกราฟแบบมีทิศทาง (Directed Graph) โดยจะแสดงการเปลี่ยนแปลงของสัญญาณอินพุต, เอาต์พุต และสัญญาณภายในของวงจรบางเส้นสัญญาณซึ่งเทียบได้กับ Petri nets ที่มีคุณสมบัติ Live-Safe Free-Choice (LSFC) nets ตัวอย่างของ STG ดังแสดงในรูปที่ 2.2



รูปที่ 2.2 ตัวอย่างซิกแนลทรานสิชันกราฟ

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ซิกแนลทรานสิชันกราฟประกอบไปด้วย

1. เซตของสัญญาณ เขียนแทนด้วยสัญลักษณ์ J ประกอบไปด้วยเซตของสัญญาณอินพุต (J_I) และเซตของสัญญาณที่ไม่ใช่สัญญาณอินพุต (J_{NI}) ซึ่งประกอบด้วยเซตของสัญญาณเอาต์พุต (J_O) และเซตของสัญญาณภายในวงจร (J_N) โดยที่ $J_{NI} = J_O \cup J_N \neq \phi$ และซิกแนลทรานสิชันกราฟจะแสดงความแตกต่างระหว่างเซตของสัญญาณอินพุต และเซตของสัญญาณที่ไม่ใช่สัญญาณอินพุต ด้วยการขีดเส้นใต้เฉพาะเซตของสัญญาณอินพุต ดังตัวอย่างจากรูปที่ 2.2 J = {a, b, c}, J_I = {a, b}, J_{NI} = {c}

2. เซตของการเปลี่ยนแปลงสัญญาณ เขียนแทนด้วยสัญลักษณ์ T โดยที่ T = J x {+, -} สำหรับทุกๆ สัญญาณ $j \in J$ จะสามารถมีการเปลี่ยนแปลงสัญญาณได้ 2 แบบคือ {j+, j-} j+ (rising) หมายถึง j มีการเปลี่ยนแปลงสัญญาณจาก 0 ไปเป็น 1 และ j- (falling) หมายถึง j มีการเปลี่ยนแปลงสัญญาณจาก 1 ไปเป็น 0

สำหรับการเปลี่ยนแปลงของสัญญาณอินพุตจะเกิดขึ้นจากสิ่งแวดล้อมภายนอก (external environment) ของวงจร แต่การเปลี่ยนแปลงสัญญาณของสัญญาณที่ไม่ใช่สัญญาณอินพุตจะเกิดขึ้นจากการทำงานภายในวงจรเอง ดังตัวอย่างจากรูปที่ 2.2 T = {a+, a-, b+, b-, c+, c-}, T_I = {a+, a-, b+, b-} และ T_{NI} = {c+, c-}

ซิกแนลทรานสิชันกราฟจะใช้สัญลักษณ์ T/n เพื่อแสดงลำดับครั้งของการเปลี่ยนแปลงของสัญญาณ T ในซิกแนลทรานสิชันกราฟ ตัวอย่างเช่น a+/1 หมายถึงสัญญาณ a เกิดการเปลี่ยนแปลงสัญญาณแบบ rising ครั้งที่ 1 และ a+/2 หมายถึงสัญญาณ a เกิดการเปลี่ยนแปลงสัญญาณแบบ rising ครั้งที่ 2

3. สัญลักษณ์ marking หรือที่เรียกว่าโทเค็น (token) เขียนแทนด้วยสัญลักษณ์ ● ไว้ใช้แสดงสถานะ (state) ของการเปลี่ยนแปลงสัญญาณในซิกแนลทรานสิชันกราฟ ณ ขณะใดขณะหนึ่ง สำหรับทุกๆ ซิกแนลทรานสิชันกราฟจะต้องมีสัญลักษณ์ initial marking เพื่อแสดงสถานะเริ่มต้นของการทำงานของวงจร

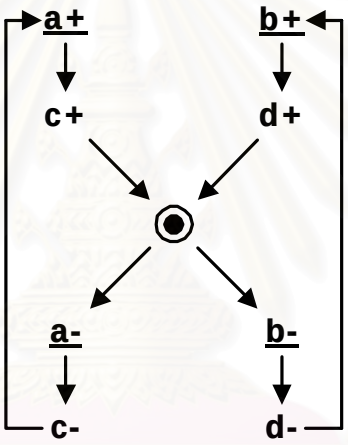
4. คุณสมบัติ causal relation เขียนแทนด้วยสัญลักษณ์ R เช่น t_1Rt_2 หมายถึงการเปลี่ยนแปลงสัญญาณ t_1 จะทำให้เกิดการเปลี่ยนแปลงสัญญาณ t_2 สำหรับในซิกแนลทรานสิชันกราฟจะใช้ลูกศรทางเดียวเป็นสัญลักษณ์เพื่อแสดงความสัมพันธ์ดังกล่าว จากรูปที่ 2.2 a+ $\rightarrow$ c+ หมายถึงการเปลี่ยนแปลงสัญญาณ a แบบ rising จะมีผลทำให้เกิดการเปลี่ยนแปลงสัญญาณ c แบบ rising แต่การเปลี่ยนแปลงสัญญาณ c แบบ rising จะเกิดขึ้นได้ก็ต่อเมื่อต้องเกิดการเปลี่ยนแปลงสัญญาณ b แบบ rising ด้วย

5. คุณสมบัติ temporal relation เขียนแทนด้วยสัญลักษณ์ tr โดยที่ tr = T x T โดยคุณสมบัติ temporal relation จะเป็นความสัมพันธ์แบบทวิภาค (binary relation) ของการเปลี่ยนแปลงสัญญาณใดๆ ในเซตของการเปลี่ยนแปลงสัญญาณ ซิกแนลทรานสิชันกราฟสามารถแสดงความสัมพันธ์

ระหว่างการเปลี่ยนแปลงสัญญาณใดๆ ได้ทั้งในแบบอันดับ (order) และแบบพร้อมกัน (concurrent) ดังตัวอย่างจากรูปที่ 2.2

- การเปลี่ยนแปลงสัญญาณ a กับ c แบบ rising จะมีความสัมพันธ์แบบอับดับ ดังนั้นเซตลำดับของการเปลี่ยนแปลงสัญญาณ a กับ c คือ {a+ c+}
- การเปลี่ยนแปลงสัญญาณ a กับ b แบบ rising จะมีความสัมพันธ์แบบพร้อมกัน ดังนั้นเซตลำดับของการเปลี่ยนแปลงสัญญาณ a กับ b คือ {a+ b+, b+ a+}

6. คุณสมบัติ input choice หรือเรียกอีกอย่งหนึ่งว่า free choice เขียนแทนด้วยสัญลักษณ์ ○ หรือที่เรียกว่า place ซึ่งหมายถึงเมื่อ marking อยู่ที่ place การเกิด firing ของการเปลี่ยนแปลงสัญญาณที่เป็นเอาต์พุตของ place จะมีเพียงสัญญาณเดียวเท่านั้นที่ถูกเลือกแบบเชิงไม่กำหนด (nondeteministic) ในการ firing และการเปลี่ยนแปลงสัญญาณที่เลือกนั้นจะไป disable การเปลี่ยนแปลงสัญญาณอื่นๆ เพื่อไม่ให้เกิดการเปลี่ยนแปลงสัญญาณขึ้น



รูปที่ 2.3 ตัวอย่างซิกแนลทรานสิชันกราฟที่มีคุณสมบัติ input choice

ตัวอย่างจากรูปที่ 2.3 เมื่อ marking อยู่ที่ place การเปลี่ยนแปลงสัญญาณที่เกิดขึ้นได้คือ a- หรือ b- ได้เพียงสัญญาณใดสัญญาณหนึ่งเท่านั้น

แต่อย่างไรก็ตามในการสังเคราะห์วงจรเพื่อให้ได้วงจรที่มีคุณสมบัติ deadlock free และ hazard free นั้นซิกแนลทรานสิชันกราฟยังจะต้องมีคุณสมบัติดังต่อไปนี้

Liveness หมายถึงทุกๆ จุดที่ marking สามารถไปถึงได้จะต้องสามารถทำให้เกิดการเปลี่ยนแปลงสัญญาณได้อีกเมื่อเกิดการเปลี่ยนแปลงของสัญญาณไปแล้ว ตัวอย่างซิกแนลทรานสิชันจากรูปที่ 2.4 (a) จะไม่มีคุณสมบัติ liveness เนื่องจากการเปลี่ยนแปลงสัญญาณ r+ จะเกิดขึ้นเพียงครั้งเดียวเท่านั้น

Safety หมายถึงไม่มี place หรือลูกศรใดๆ ในซิกแนลทรานสิชันกราฟสามารถมี marking ได้มากกว่า 1 marking ตัวอย่างซิกแนลทรานสิชันกราฟจากรูปที่ 2.4 (b) จะไม่มีคุณสมบัติ safety เนื่องจาก r+ $\rightarrow$ y+ สามารถมี marking ได้มากกว่า 1 marking

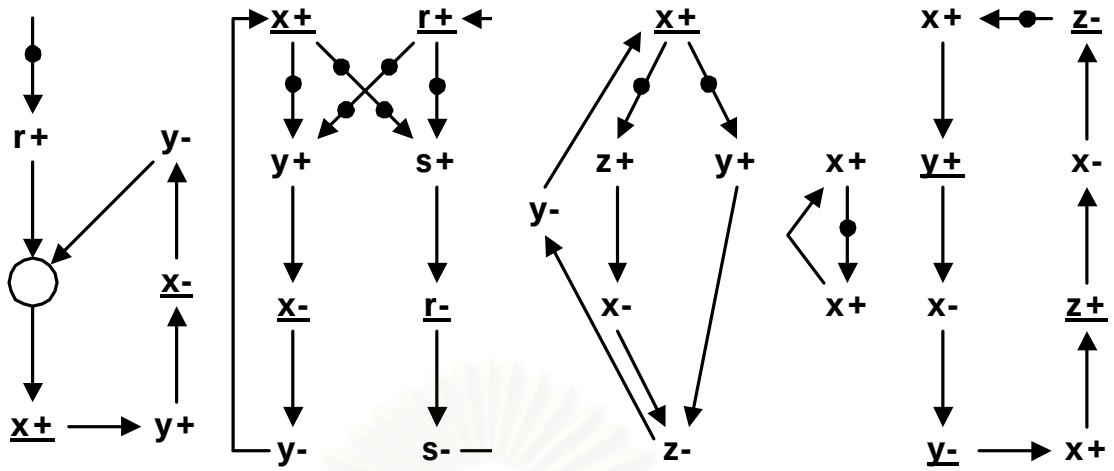
Persistency หมายถึงทุกๆ ความสัมพันธ์ $a^* \rightarrow b^*$ (t^* หมายถึงการเปลี่ยนแปลงของสัญญาณ t+ หรือ t-) ในซิกแนลทรานสิชันกราฟ การเปลี่ยนแปลงสัญญาณ b^* จะต้องเกิดก่อนการเปลี่ยนแปลงสัญญาณ a^* ในทิศตรงกันข้าม ตัวอย่างซิกแนลทรานสิชันกราฟจากรูปที่ 2.4 (c) จะไม่มีคุณสมบัติ presistency ในส่วนของ $\underline{x+} \rightarrow$ y+ เนื่องจาก $\underline{x-}$ สามารถเกิดการเปลี่ยนแปลงสัญญาณได้ก่อนที่ y+ จะเกิดการเปลี่ยนแปลงสัญญาณ แต่คุณสมบัตินี้มีข้อยกเว้นสำหรับสัญญาณอินพุตเช่น $a^* \rightarrow \underline{b^*}$ ถึงแม้ว่าสัญญาณ a^* ในทิศทางตรงกันข้ามจะต้องเกิดก่อนการเปลี่ยนแปลงสัญญาณ b^* ซิกแนลทรานสิชันกราฟนั้นก็ยังคงมีคุณสมบัติ persistency อยู่

Consistency หมายถึงเมื่อมีการเปลี่ยนแปลงของทุกๆ สัญญาณในซิกแนลทรานสิชันกราฟ การเปลี่ยนแปลงสัญญาณในครั้งต่อไปของสัญญาณนั้นๆ จะต้องเปลี่ยนแปลงในทิศทางตรงกันข้ามเท่านั้น ตัวอย่างซิกแนลทรานสิชันกราฟจากรูปที่ 2.4 (d) จะไม่มีคุณสมบัติ consistency เนื่องจากเกิดการเปลี่ยนแปลงของสัญญาณ x+ ต่อกันโดยที่ไม่มีการเปลี่ยนแปลงสัญญาณ x-

Unique state assignment หมายถึงทุกๆ จุดในซิกแนลทรานสิชันกราฟจะต้องมีค่าของสัญญาณทุกสัญญาณเพียงค่าเดียวเท่านั้น และค่านั้น ๆ จะต้องไม่มีค่าซ้ำกับจุดอื่นๆ ในซิกแนลทรานสิชันกราฟ ตัวอย่างซิกแนลทรานสิชันกราฟจากรูปที่ 2.4 (e) จะไม่คุณสมบัติ unique state assignment เนื่องจากค่าของสัญญาณในสถานะเริ่มต้นซ้ำกันกับค่าของสัญญาณที่จุด $\underline{y-} \rightarrow$ x+

Single cycle transitions หมายถึงการเปลี่ยนแปลงของแต่ละสัญญาณใน cycle ใดของซิกแนลทรานสิชันกราฟนั้น จะต้องเกิดการเปลี่ยนแปลงของสัญญาณนั้นๆ 2 ครั้งคือ rising และ falling อย่างละครั้งเท่านั้น ตัวอย่างซิกแนลทรานสิชันกราฟจากรูปที่ 2.4 (e) จะไม่มีคุณสมบัติ single cycle transitions เนื่องจากเกิดการเปลี่ยนแปลงของสัญญาณ x+ และ x- มากกว่า 1 ครั้งใน cycle ของซิกแนลทรานสิชันกราฟ

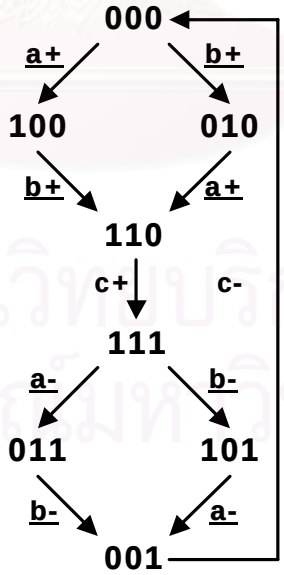
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 2.4 ตัวอย่างซิกแนลทรานสิชันกราฟที่ไม่มีคุณสมบัติ (a) Liveness (b) Safety (c) Persistency (d) Consistency (e) Unique state assignment และ Single cycle transition

2.3 State Graph

State Graph (SG) เป็นกราฟที่แสดงสถานะต่างๆ ของซิกแนลทรานสิชันกราฟ โดยที่สถานะดังกล่าวถูกแทนที่ด้วยเวกเตอร์ของเลขฐานสอง (binary vector) ซึ่งสามารถที่จะแสดงค่าของสัญญาณทั้งหมดในซิกแนลทรานสิชันกราฟ และจะแสดงถึงความสัมพันธ์ระหว่างสถานะต่างๆ ที่ไม่สามารถดูได้จากซิกแนลทรานสิชันกราฟ ดังแสดงในรูปที่ 2.5



รูปที่ 2.5 State Graph ของซิกแนลทรานสิชันกราฟที่อ้างอิงจากรูปที่ 2.2

จากรูปที่ 2.5 โนด (node) บนกราฟจะแสดงสถานะของวงจรด้วยค่าของเส้นสัญญาณแต่ละเส้น ณ เวลาใดเวลาหนึ่งซึ่งจะถูกกำหนดไว้ในโนดต่างๆ บนกราฟ โนดแต่ละโนดจะถูกแทนที่ด้วย

เลข 3 หลักซึ่งแต่ละหลักจะแทนค่าของสัญญาณในซิกแนลทรานสิชันกราฟจากรูปที่ 2.2 โดยแทนสัญญาณ a, b และ c ตามลำดับ เส้นแต่ละเส้นใน state graph จะหมายถึงการเปลี่ยนแปลงของค่าสัญญาณหนึ่งๆ ในซิกแนลทรานสิชันกราฟจากหัวลูกศรไปเป็นสถานะของวงจรที่ปลายลูกศร ตัวอย่างจากรูปที่ 2.5 000 $\rightarrow$ 100 หมายถึงการเปลี่ยนแปลงสัญญาณ a จาก 0 เป็น 1 (rising)

สถานะแต่ละสถานะใน state graph ยังแบ่งออกเป็นสถานะคงที่ และสถานะไม่คงที่ซึ่งสถานะคงที่หมายถึงสถานะที่ไม่สามารถเปลี่ยนแปลงสถานะได้ถ้าไม่มีการเปลี่ยนแปลงสัญญาณอินพุต ตัวอย่างจากรูปที่ 2.5 สถานะที่เป็นสถานะคงที่คือ 000, 100, 010, 011, 101 และ 111 ส่วนสถานะไม่คงที่หมายถึงสถานะที่สามารถเปลี่ยนแปลงสถานะไปยังสถานะต่อไปได้โดยไม่ต้องรอการเปลี่ยนแปลงสัญญาณอินพุต ตัวอย่างจากรูปที่ 2.5 สถานะที่เป็นสถานะไม่คงที่คือ 110 และ 001 โดยในการทำงานของวงจรจะเริ่มต้นการทำงานที่สถานะเริ่มต้นซึ่งถือว่าเป็นสถานะคงที่ ตัวอย่างจากรูปที่ 2.5 สถานะเริ่มต้นคือ 000

2.4 พีชคณิตเชิงกระบวนการ

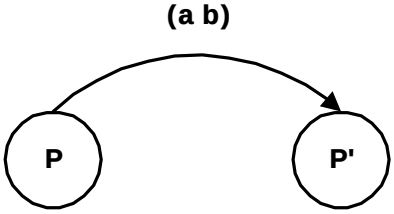
พีชคณิตเชิงกระบวนการ (Process Algebra) [1], [5] เป็นเครื่องมือที่ใช้แนวคิดทางคณิตศาสตร์ในการอธิบาย และวิเคราะห์พฤติกรรมแบบ concurrent system โดยจะยึดหลักที่ว่าในแต่ละกระบวนการ (process) จะมีสถานะ (state) ซึ่งตอบสนองต่อการกระทำ (action)

พีชคณิตเชิงกระบวนการประกอบไปด้วย

1. Termination เขียนแทนด้วนสัญลักษณ์ /\ แสดงถึงกระบวนการที่ไม่ยอมรับการกระทำใดๆ ซึ่งหมายถึงกระบวนการนั้นจบการทำงานหรือเกิด dead lock ดังแสดงในรูปที่ 2.6

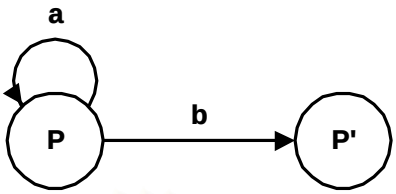
รูปที่ 2.6 Termination

2. Guarding เขียนแทนด้วยสัญลักษณ์ $\rightarrow$ ตัวอย่างเช่น P $\rightarrow$ (a b) P' แสดงถึงกระบวนการ P จะเปลี่ยนสถานะไปยังกระบวนการ P' จากการกระทำ a และ b บนกระบวนการ P ดังแสดงในรูปที่ 2.7



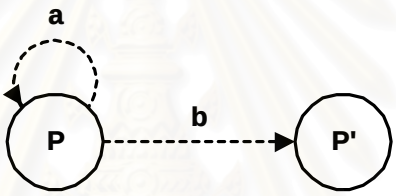
รูปที่ 2.7 Guarding

3. Choice เขียนแทนด้วยสัญลักษณ์ + ตัวอย่างเช่น P $\longrightarrow$ a P + b P' แสดงถึงกระบวนการ P จะเปลี่ยนสถานะไปยังกระบวนการ P โดยการกระทำ a หรือเปลี่ยนสถานะไปยังกระบวนการ P' โดยการกระทำ b บนกระบวนการ P ดังแสดงในรูปที่ 2.8



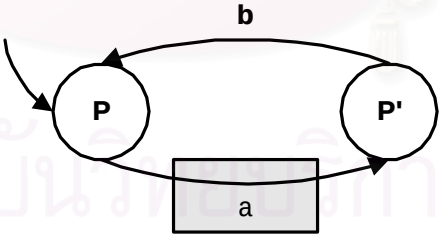
รูปที่ 2.8 Choice

4. Nondeterminism เขียนแทนด้วยสัญลักษณ์ & ตัวอย่างเช่น P $\longrightarrow$ a P & b P' จะเหมือนกับ choice แต่เส้นทางที่เลือกจะถูกเลือกจากสิ่งแวดล้อมภายนอกไม่ใช่จากตัวกระบวนการเอง ดังแสดงในรูปที่ 2.9



รูปที่ 2.9 Nondeterminism

5. Abstraction เขียนแทนด้วยสัญลักษณ์ - ตัวอย่างเช่น P-a โดยที่กำหนดให้ P $\longrightarrow$ a P' และ P' $\longrightarrow$ b P แสดงถึงกระบวนการ P ที่ไม่แสดงการกระทำ a ดังแสดงในรูปที่ 2.10



รูปที่ 2.10 Abstraction

6. Concurrent composition เขียนแทนด้วยสัญลักษณ์ * แสดงถึงกระบวนการที่เกิดการ concurrent กันโดยมีนิยามดังนี้

Given process X of action set M and process Y of action set N, m an action X can perform and n an action Y can perform, and $m \cap N = n \cap M$:

for $X = \sum_{1 \le i \le m} \lambda_i X_i$ of action set M, and $Y = \sum_{1 \le j \le n} \mu_j Y_j$ of action set N

$$X * Y = \sum_{\substack{1 \le i \le m \\ \lambda_i \cap N = \phi}} \lambda_i (X_i * Y) + \sum_{\substack{1 \le j \le n \\ M \cap \mu_j = \phi}} \mu_j (X * Y_j) + \sum_{\substack{1 \le i \le m \\ 1 \le j \le n \\ \lambda_i \cap N = M \cap \mu_j}} (\lambda_i \cup \mu_j)(X_i * Y_j)$$

ตัวอย่างเช่นกำหนดให้

$P \longrightarrow a\,P' + b\,P'$ $\quad P' \longrightarrow a\,P + c\,P$

$Q \longrightarrow a\,Q' + d\,Q'$ $\quad Q' \longrightarrow a\,Q$

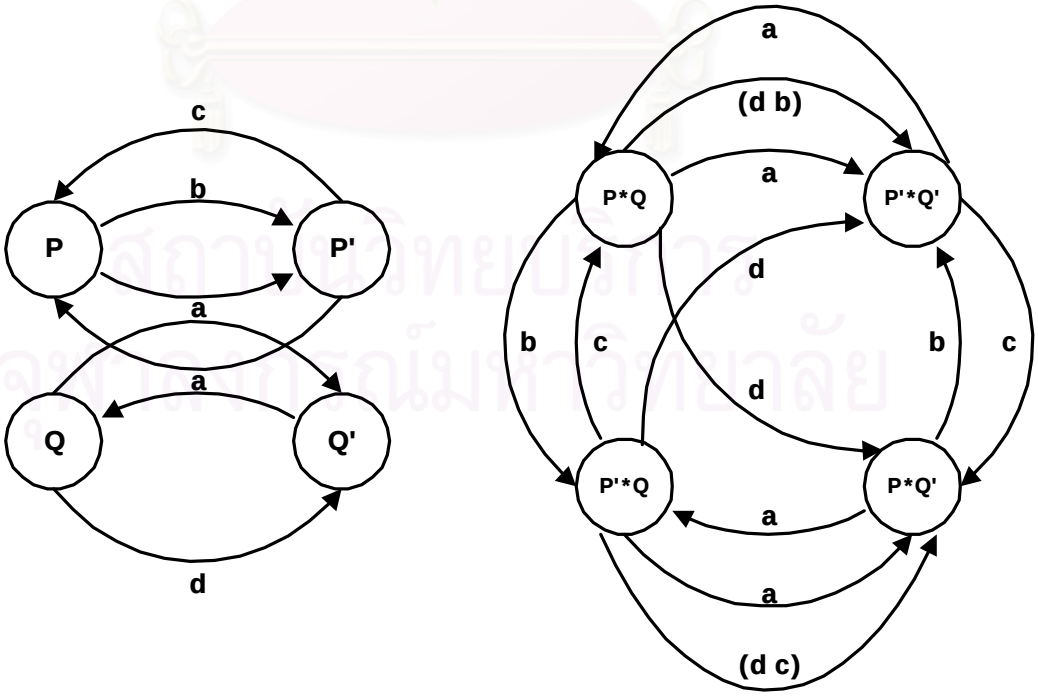
$P * Q = a\,(P' * Q') + d\,(P * Q') + (d\,b)\,(P' * Q') + b\,(P' * Q)$

$P' * Q' = a\,(P * Q) + c\,(P * Q')$

$P * Q' = a\,(P' * Q) + b\,(P' * Q')$

$P' * Q = a\,(P * Q') + d\,(P' * Q') + (d\,c)\,(P * Q') + c\,(P * Q)$

ดังแสดงในรูปที่ 2.11



รูปที่ 2.11 Concurrent Composition

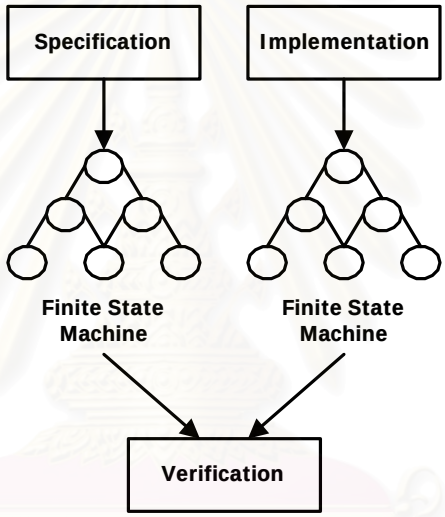
2.5 สรุป

ในบทนี้ได้กล่าวถึงโมเดลความหน่วงแบบควอไซดีเลย์อินเซนซิทีฟซึ่งเป็นโมเดลแบบหนึ่งที่ใช้ในการออกแบบวงจรแบบอสมวาร, ซิกแนลทรานสิชันกราฟโดยได้อธิบายถึงรูปแบบของการกำหนดคุณลักษณะของวงจรด้วยซิกแนลทรานสิชันกราฟ, ความสัมพันธ์ที่เกิดขึ้นภายในเซตของการเปลี่ยนแปลงสัญญาณ, คุณสมบัติต่างๆ ของซิกแนลทรานสิชันกราฟเพื่อให้ได้วงจรที่มีคุณสมบัติ deadlock free และ hazard free, state graph ซึ่งเป็นกราฟที่แสดงสถานะ และความสัมพันธ์ของสถานะต่างๆ ในซิกแนลทรานสิชันกราฟ และพีชคณิตเชิงกระบวนการซึ่งเป็นเครื่องมือที่ใช้แนวคิดทางคณิตศาสตร์ในการอธิบาย และวิเคราะห์พฤติกรรมแบบ concurrent system ซึ่งความรู้ที่ได้กล่าวมาทั้งหมดนี้ เพียงพอสำหรับเป็นความรู้พื้นฐานที่จะให้ผู้อ่านสามารถทำความเข้าใจกับงานวิจัย และจะนำไปประยุกต์ใช้ในงานวิจัยที่จะนำเสนอในบทต่อๆ ไปได้เป็นอย่างดี

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 3
การทวนสอบวงจรอสมวารอย่างมีแบบแผนแบบ Model Checking

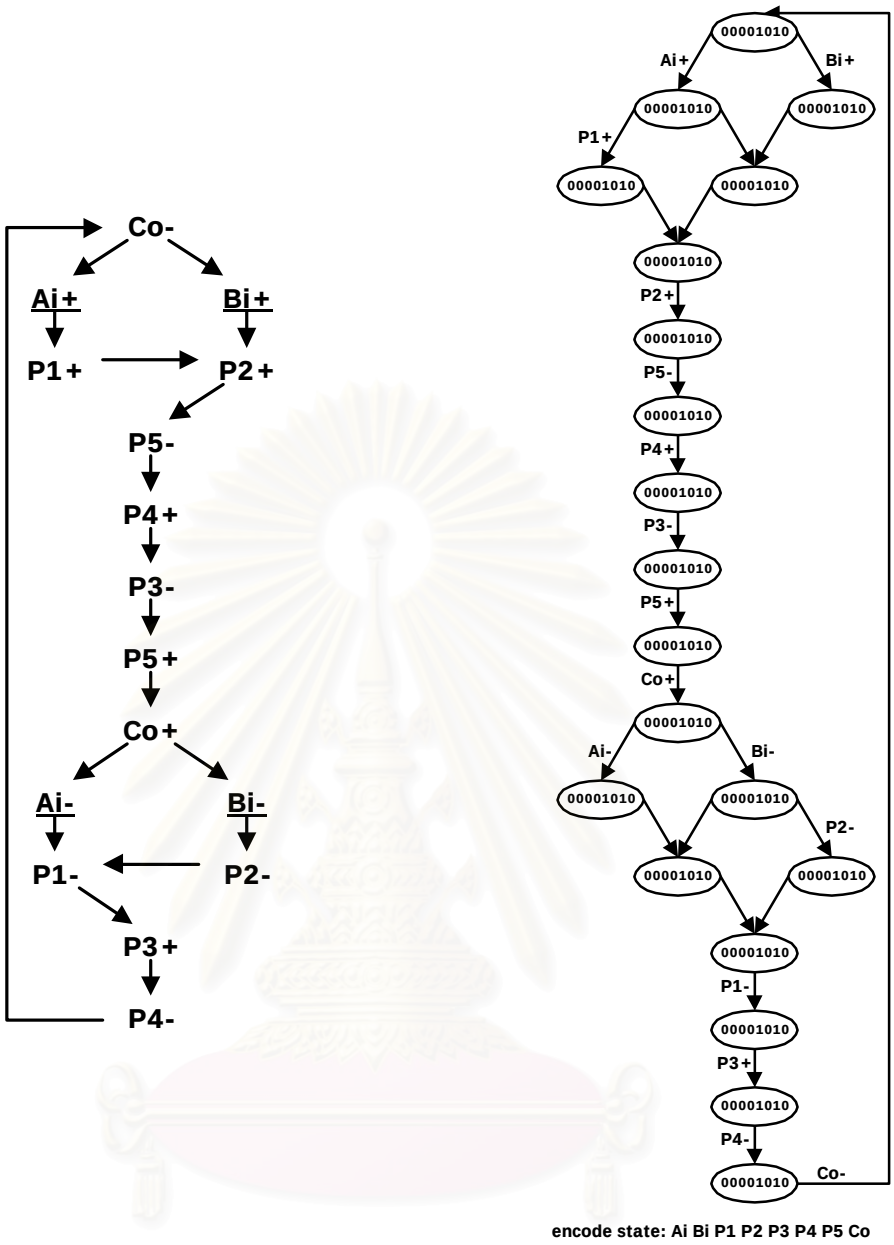
การทวนสอบวงจรอสมวารที่นำเสนอในงานวิจัยฉบับนี้จะเป็นขั้นตอนวิธีการทวนสอบวงจรอสมวารแบบควอไซดีเลย์อินเซนซิทีฟอย่างมีแบบแผนแบบ model checking โดยใช้เครื่องจักรสถานะจำกัด ซึ่งการทวนสอบสำหรับวิธีนี้ทั้งคุณลักษณะของวงจรที่ได้ออกแบบไว้ และวงจรที่ได้จากการสังเคราะห์จะถูกแปลงให้อยู่ในรูปของเครื่องจักรสถานะจำกัด และทั้งคู่จะถูกทำการทวนสอบด้วยการเปรียบเทียบว่า พฤติกรรมของวงจรที่ได้จากการสังเคราะห์ถูกต้องตรงตามพฤติกรรมของคุณลักษณะของวงจรที่ได้ออกแบบไว้หรือไม่ สำหรับขั้นตอนวิธีการทวนสอบวงจรอสมวารที่ได้กล่าวมาในข้างต้นนี้จะแสดงดังรูปที่ 3.1



รูปที่ 3.1 การทวนสอบวงจรอสมวารอย่างมีแบบแผนแบบ Model Checking

3.1 เครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร

คุณลักษณะของวงจรจะถูกแสดงในรูปของซิกแนลทรานสิชันกราฟ (STG) และสามารถที่จะนำมาสร้างเป็นกราฟสถานะ (State graph) ซึ่งเป็นกราฟที่แสดงสถานะต่างๆ ของสัญญาณทั้งหมดที่เกิดขึ้นในซิกแนลทรานสิชันกราฟ ซึ่งจะถูกแทนที่ด้วยเลขฐานสอง และจะแสดงถึงความสัมพันธ์ระหว่างสถานะต่างๆ ด้วยค่าของสัญญาณที่เกิดการเปลี่ยนแปลง โดยที่เราจะใช้กราฟสถานะแทนเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร ดังแสดงในรูปที่ 3.2

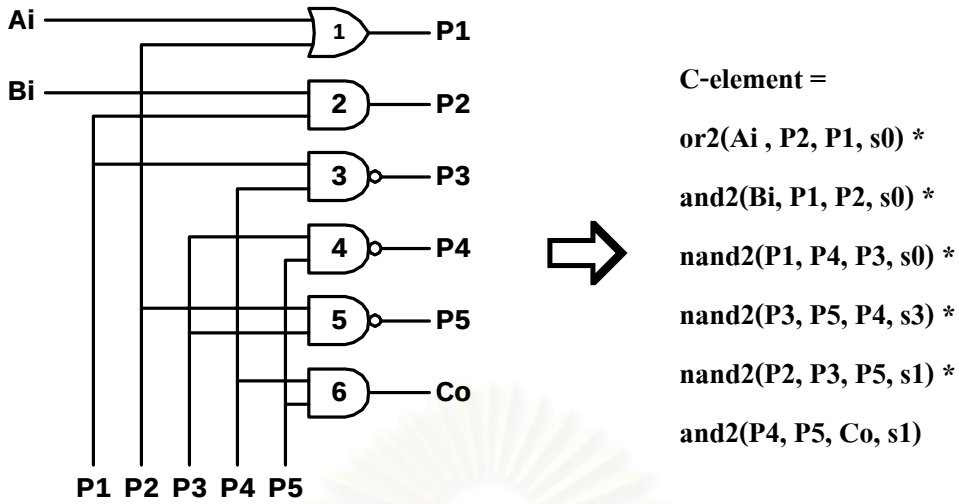


(a) (b)

รูปที่ 3.2 (a) ซิกแนลทรานสิชันกราฟของวงจร C-element (b) กราฟสถานะของวงจร C-elment

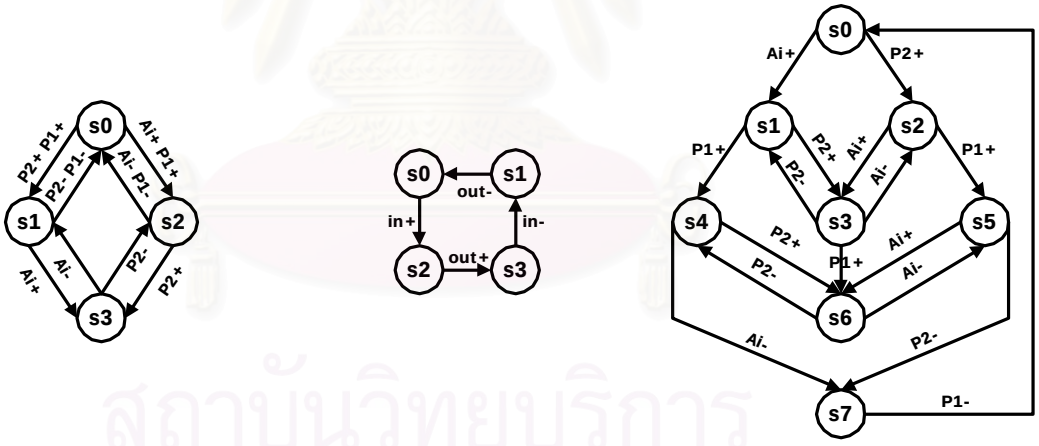
3.2 เครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์

วงจรที่ได้จากการสังเคราะห์จะถูกแสดงในรูปแบบของนิพจน์ที่มาเรียงต่อกัน นิพจน์แต่ละตัวจะแทนเกตหนึ่งเกตในวงจร ซึ่งจะแสดงชื่อของเกต สัญญาณขาเข้า สัญญาณขาออก และค่าสัญญาณเริ่มต้นของเกตนั้นๆ ตามลำดับ ดังแสดงในรูปที่ 3.3



รูปที่ 3.3 รูปแบบนิพจน์ของวงจร C-element ที่ได้จากการสังเคราะห์

นิพจน์แต่ละตัวจะถูกแทนด้วยกราฟที่แสดงพฤติกรรมการเปลี่ยนแปลงสัญญาณของเกตในนิพจน์ หลังจากนั้นจะนำกราฟที่ได้มาทำการเพิ่มดีเลย์ให้กับสัญญาณขาออกเพื่อที่จะให้สัญญาณขาออกเกิดการเปลี่ยนแปลงสัญญาณหลังจากสัญญาณขาเข้าเกิดการเปลี่ยนแปลงสัญญาณแล้ว และการเปลี่ยนแปลงของสัญญาณขาเข้าจะต้องรอให้เกิดการเปลี่ยนแปลงของสัญญาณขาออกเสร็จเรียบร้อยเสียก่อน ดังแสดงในรูปที่ 3.4



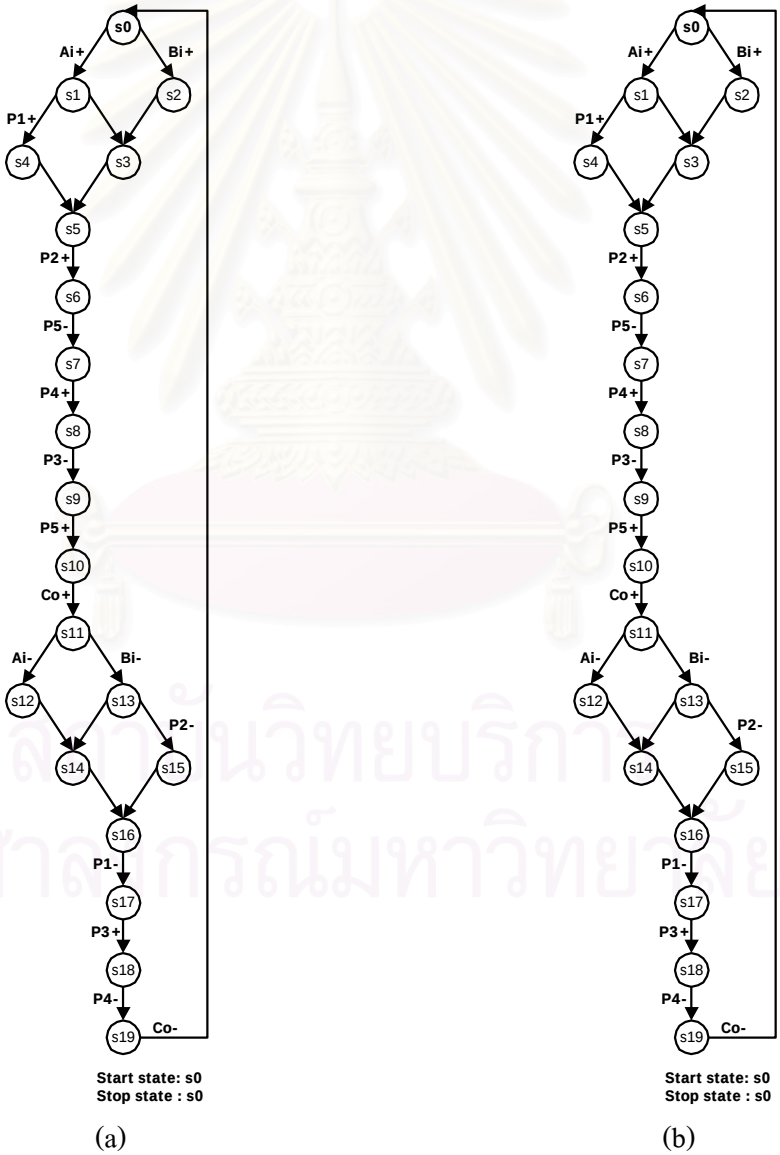
(a) (b) (c)

รูปที่ 3.4 (a) กราฟแสดงพฤติกรรมของนิพจน์ or2(Ai, P2, P1, s0)
(b) กราฟแสดงพฤติกรรมของดีเลย์
(c) กราฟแสดงพฤติกรรมของนิพจน์ or2(Ai, P2, P1, s0) หลังจากเพิ่มดีเลย์เข้าที่สัญญาณขาออก

หลังจากที่ได้ทำการเพิ่มดีเลย์ให้กับเกตทุกเกตในวงจรแล้ว เราจะนำกราฟพฤติกรรมทั้งหมดที่ได้มาพิจารณาแยกตามสัญญาณขาเข้า สัญญาณขาออก และสัญญาณภายใน เพื่อที่จะนำมาสร้างเป็นกราฟเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ต่อไป ซึ่งจะกล่าวโดยละเอียดในบทที่ 5

3.3 การทวนสอบ

การทวนสอบวงจรอสมวารอย่างมีแบบแผนแบบ model checking โดยใช้เครื่องจักรสถานะจำกัด จะใช้วิธีการพิจารณาเครื่องจักรสถานะจำกัดของทั้งคุณลักษณะของวงจร และวงจรที่ได้จากการสังเคราะห์ ว่ามีพฤติกรรมเหมือนกันหรือไม่ โดยพิจารณาจากลำดับการเปลี่ยนแปลงสัญญาณที่เป็นไปได้ทั้งหมดจากสถานะเริ่มต้น ไปยังสถานะสิ้นสุด ดังแสดงในรูปที่ 3.5



รูปที่ 3.5 (a) เครื่องจักรสถานะจำกัด A (b) เครื่องจักรสถานะจำกัด B

ลำดับการเปลี่ยนแปลงสัญญาณของเครื่องจักรสถานะจำกัด A จากสถานะ s0 ไปยังสถานะ s0

Ai+ P1+ Bi+ P2+ P5- P4+ P3- P5+ Co+ Ai- Bi- P2- P1- P3+ P4- Co-

Ai+ P1+ Bi+ P2+ P5- P4+ P3- P5+ Co+ Bi- Ai- P2- P1- P3+ P4- Co-

Ai+ P1+ Bi+ P2+ P5- P4+ P3- P5+ Co+ Bi- P2- Ai- P1- P3+ P4- Co-

Ai+ Bi+ P1+ P2+ P5- P4+ P3- P5+ Co+ Ai- Bi- P2- P1- P3+ P4- Co-

Ai+ Bi+ P1+ P2+ P5- P4+ P3- P5+ Co+ Bi- Ai- P2- P1- P3+ P4- Co-

Ai+ Bi+ P1+ P2+ P5- P4+ P3- P5+ Co+ Bi- P2- Ai- P1- P3+ P4- Co-

Bi+ Ai+ P1+ P2+ P5- P4+ P3- P5+ Co+ Ai- Bi- P2- P1- P3+ P4- Co-

Bi+ Ai+ P1+ P2+ P5- P4+ P3- P5+ Co+ Bi- Ai- P2- P1- P3+ P4- Co-

Bi+ Ai+ P1+ P2+ P5- P4+ P3- P5+ Co+ Bi- P2- Ai- P1- P3+ P4- Co-

ลำดับการเปลี่ยนแปลงสัญญาณของเครื่องจักรสถานะจำกัด B จากสถานะ s0 ไปยังสถานะ s0

Ai+ P1+ Bi+ P2+ P5- P4+ P3- P5+ Co+ Ai- Bi- P2- P1- P3+ P4- Co-

Ai+ P1+ Bi+ P2+ P5- P4+ P3- P5+ Co+ Bi- Ai- P2- P1- P3+ P4- Co-

Ai+ P1+ Bi+ P2+ P5- P4+ P3- P5+ Co+ Bi- P2- Ai- P1- P3+ P4- Co-

Ai+ Bi+ P1+ P2+ P5- P4+ P3- P5+ Co+ Ai- Bi- P2- P1- P3+ P4- Co-

Ai+ Bi+ P1+ P2+ P5- P4+ P3- P5+ Co+ Bi- Ai- P2- P1- P3+ P4- Co-

Ai+ Bi+ P1+ P2+ P5- P4+ P3- P5+ Co+ Bi- P2- Ai- P1- P3+ P4- Co-

Bi+ Ai+ P1+ P2+ P5- P4+ P3- P5+ Co+ Ai- Bi- P2- P1- P3+ P4- Co-

Bi+ Ai+ P1+ P2+ P5- P4+ P3- P5+ Co+ Bi- Ai- P2- P1- P3+ P4- Co-

Bi+ Ai+ P1+ P2+ P5- P4+ P3- P5+ Co+ Bi- P2- Ai- P1- P3+ P4- Co-

จากรูปที่ 3.5 จะเป็นเครื่องจักรสถานะจำกัดของวงจร C-element โดยที่เครื่องจักรสถานะจำกัด A จะเป็นเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร และเครื่องจักรสถานะจำกัด B จะเป็นเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ ซึ่งเครื่องจักรสถานะจำกัดทั้งสองจะมีลำดับการเปลี่ยนแปลงสัญญาณจากสถานะเริ่มต้นไปยังสถานะสิ้นสุดทั้งหมดเหมือนกัน ทำให้สามารถสรุปได้ว่าเครื่องจักรสถานะทั้งสองมีพฤติกรรมเหมือนกัน

3.4 สรุป

ในบทนี้ได้กล่าวถึงขั้นตอนวิธีการทวนสอบวงจรอสมวารอย่างมีแบบแผนแบบ model checking โดยใช้เครื่องจักรสถานะจำกัดซึ่งการทวนสอบสำหรับวิธีนี้ทั้งคุณลักษณะของวงจรที่ได้ออกแบบไว้ และวงจรที่ได้จากการสังเคราะห์จะถูกแปลงให้อยู่ในรูปของเครื่องจักรสถานะจำกัด และทั้งคู่จะถูกทำการทวนสอบด้วยการเปรียบเทียบว่า พฤติกรรมของวงจรที่ได้จากการสังเคราะห์ถูกต้องตรงตามพฤติกรรมของคุณลักษณะของวงจรที่ได้ออกแบบไว้หรือไม่

ขั้นตอนวิธีการทวนสอบที่ได้นำเสนอในบทนี้เป็นเพียงโครงสร้างของขั้นตอนวิธีการทวนสอบ เพื่อให้ผู้อ่านเข้าใจในภาพรวมของขั้นตอนวิธีการทวนสอบทั้งหมด สำหรับในบทที่ 4 และบทที่ 5 จะเป็นการนำเสนอขั้นตอนวิธีการสร้างเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร, เครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ อย่างละเอียดตามลำดับ และในบทที่ 6 จะเป็นการนำเสนอขั้นตอนการตรวจสอบความถูกต้องตรงกันระหว่างเครื่องจักรสถานะจำกัดทั้งสอง

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 4
เครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร

ในบทนี้เราจะกล่าวถึงรายละเอียดของขั้นตอนวิธีการสร้างเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร โดยจะอธิบายเกี่ยวกับขั้นตอนในการวิเคราะห์ทุกๆ การเปลี่ยนแปลงสัญญาณในซิกแนลทรานสิชันกราฟ เพื่อนำมาสร้างกราฟสายงานสำหรับเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร โดยขั้นตอนวิธีในการสร้างกราฟสายงานนั้นจะมีอินพุตเป็นซิกแนลทรานสิชันกราฟในรูปแบบของเท็กไฟล์ ซึ่งจะอ้างอิงกับรูปแบบของ Berkeley Logic Interchange Format [9] (BLIF) ดังแสดงในรูปที่ 4.1

.name	ชื่อวงจร	
.inputs	ชื่อ input signal1	ชื่อ inputsignal 2 …
.outputs	ชื่อ output signal1	ชื่อ output signal 2 …
.internals	ชื่อ internal signal 1	ชื่อ internal signal 2 …
.places	ชื่อ place 1	ชื่อ place 2 …
.graph		
	transition หรือ place 1	transition หรือ place 2
	…	
.markings	{<transition หรือ place 1	transition หรือ place 2>, < … >}
.end		

รูปที่ 4.1 Berkeley Logic Interchange Format

4.1 การสร้างเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร

สำหรับขั้นตอนนี้จะเป็นขั้นตอนการสร้างกราฟสายงานสำหรับเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร มีขั้นตอนดังต่อไปนี้

Algorithm 1 Construct flow graph

Step 1.1 Lexical analysis

Step 1.2 Syntax analysis

Step 1.3 Consider all STG's markings and firing rules to evaluate start node

Step 1.4 Find encode start state

Step 1.5 Construct flow graph for STG

End

จากขั้นตอนวิธีที่ 1 (algorithm 1) เราสามารถอธิบายรายละเอียดได้ดังนี้

Step 1.1 เป็นขั้นตอนการอ่านแฟ้มข้อความซึ่งเป็นไฟล์อินพุตที่ใช้แสดงแทนซิกแนลทรานสิชันกราฟ แล้วทำการวิเคราะห์แฟ้มข้อความทีละอักขระ เพื่อแยกออกเป็นโทเค็น (token) ซึ่งโทเค็นเหล่านี้จะถูกนำไปใช้ในการวิเคราะห์วากยสัมพันธ์ต่อไป ตัวอย่างเท็กซ์ไฟล์ที่ใช้เป็นไฟล์อินพุตแทนซิกแนลทรานสิชันกราฟ ของวงจร C-element ดังแสดงในรูปที่ 4.2

```
.name C-element
.inputs Ai Bi
.outputs Co
.internals P1 P2 P3 P4 P5
.graph
        Co-     Ai+     Bi+
        Ai+     P1+
        P1+     P2+
        Bi+     P2+
        P2+     P5-
        P5-     P4+
        P4+     P3-
        P3-     P5+
        P5+     Co+
        Co+     Ai-     Bi-
        Ai-     P1-
        Bi-     P2-
        P2-     P1-
        P1-     P3+
        P3+     P4-
        P4-     Co-
.markings {<Co- Ai+>, <Co- Bi+>}
.end
```

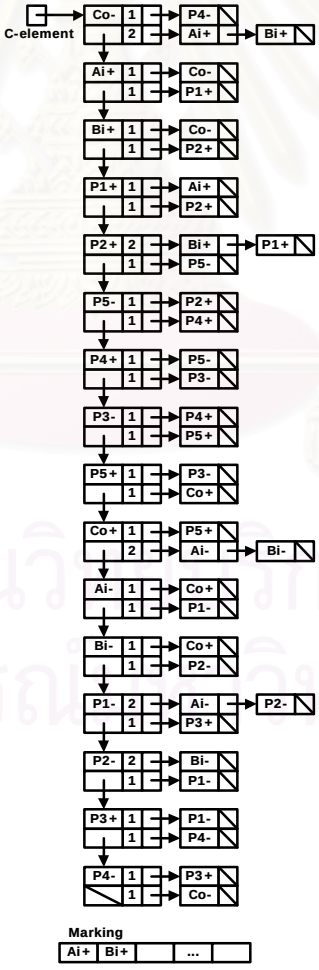
รูปที่ 4.2 ไฟล์อินพุตซิกแนลทรานสิชันกราฟของวงจร C-element

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

Step 1.2 เป็นขั้นตอนของการวิเคราะห์วากยสัมพันธ์ เพื่อวิเคราะห์แต่ละโทเค็นที่ได้มาจาก step 1.1 มาสร้างเป็นโครงสร้างของข้อมูลที่ใช้แทนซิกแนลทรานสิชันกราฟ โดยจะประกอบไปด้วยข้อมูลดังนี้

1. ชื่อของการเปลี่ยนแปลงสัญญาณ
2. จำนวนดีกรีขาเข้าของการเปลี่ยนแปลงสัญญาณ
3. จำนวนดีกรีขาออกของการเปลี่ยนแปลงสัญญาณ
4. รายการเชื่อมโยงของการเปลี่ยนแปลงสัญญาณขาเข้า
5. รายการเชื่อมโยงของการเปลี่ยนแปลงสัญญาณขาออก
6. ตัวชี้ไปยังโครงสร้างข้อมูลถัดไป

นอกจากนี้ยังเก็บข้อมูลของสัญญาณที่เป็น marking ไว้ในแถวลำดับ marking ด้วย ตัวอย่างโครงสร้างของข้อมูลที่ใช้แทนซิกแนลทรานสิชันกราฟ และแถวลำดับ marking ของวงจร C-element ดังแสดงในรูปที่ 4.3



รูปที่ 4.3 โครงสร้างข้อมูลของซิกแนลทรานสิชันกราฟ และแถวลำดับ marking ของวงจร C-element ที่อ้างอิงจากรูปที่ 4.2

Step 1.3 เป็นขั้นตอนการวิเคราะห์หาการเปลี่ยนแปลงสัญญาณเริ่มต้นของซิกแนลทรานสิชันกราฟ โดยทำการวิเคราะห์จากสัญลักษณ์ initial marking ทั้งหมดในซิกแนลทรานสิชันกราฟ เพื่อนำมาเป็นโนดเริ่มต้นในการสร้างกราฟสายงาน โดยจะทำการพิจารณาเฉพาะสัญลักษณ์ initial marking ที่มีสถานะเป็น stable เท่านั้น

สำหรับขั้นตอนวิธีในการวิเคราะห์หาการเปลี่ยนแปลงสัญญาณเริ่มต้นจะเป็นดังนี้

นับจำนวนสัญลักษณ์ initial marking ทั้งหมดของซิกแนลทรานสิชันกราฟ แล้วทำการพิจารณาตามเงื่อนไขดังต่อไปนี้

กรณีที่ 1 ถ้าจำนวนสัญลักษณ์ initial marking ทั้งหมดของซิกแนลทรานสิชันกราฟเท่ากับ 1 อันแสดงว่า การเปลี่ยนแปลงสัญญาณเริ่มต้นก็คือสัญญาณที่จะเกิดการเปลี่ยนแปลงก็ต่อเมื่อสัญลักษณ์ initial marking เกิดการ firing นั่นเอง ตัวอย่างจากรูปที่ 4.4 (a) การเปลี่ยนแปลงสัญญาณเริ่มต้นคือ Ai+

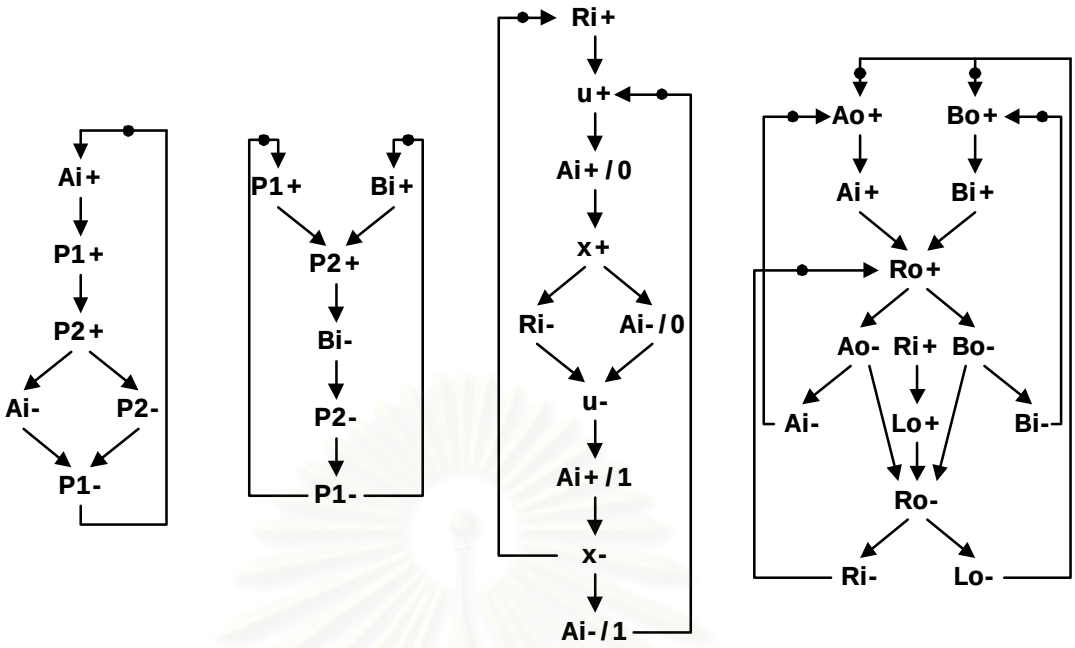
กรณีที่ 2 ถ้าจำนวนสัญลักษณ์ initial marking ทั้งหมดของซิกแนลทรานสิชันกราฟมากกว่า 1 อันให้ทำการพิจารณาจำนวนดีกรีขาเข้าของสัญญาณทุกตัวที่จะเกิดการเปลี่ยนแปลงเมื่อสัญลักษณ์ initial marking เกิดการ firing ตามเงื่อนไขดังต่อไปนี้

กรณีที่ 2.1 ถ้ามีสัญญาณที่มีจำนวนดีกรีขาเข้าเท่ากับ 1 มีมากกว่าหนึ่งอัน และไม่มีสัญญาณที่มีดีกรีขาเข้ามากกว่า 1 แสดงว่าการเปลี่ยนแปลงสัญญาณเริ่มต้นสามารถพิจารณาได้จากสัญญาณทุกตัว ตัวอย่างจากรูปที่ 4.4 (b) การเปลี่ยนแปลงสัญญาณเริ่มต้นคือ P1+ และ Bi+

กรณีที่ 2.2 ถ้ามีสัญญาณที่มีจำนวนดีกรีขาเข้าเท่ากับ 1 มีหนึ่งอัน และมีสัญญาณที่มีจำนวนดีกรีขาเข้ามากกว่า 1 มีมากกว่าหรือเท่ากับหนึ่งอันแสดงว่า การเปลี่ยนแปลงสัญญาณเริ่มต้นก็คือสัญญาณที่มีดีกรีขาเข้าเท่ากับ 1 ตัวอย่างจากรูปที่ 4.4 (c) การเปลี่ยนแปลงสัญญาณเริ่มต้นคือ Ri+

กรณีที่ 2.3 ถ้าไม่มีสัญญาณที่มีจำนวนดีกรีเข้าเท่ากับ 1 และมีสัญญาณที่มีดีกรีข้าเข้ามากกว่า 1 มีมากกว่าหนึ่งอันแสดงว่า การเปลี่ยนแปลงสัญญาณเริ่มต้นให้พิจารณาจากคุณสมบัติ causal relation ของสัญญาณทุกตัว ตัวอย่างจากรูปที่ 4.4 (d) การเปลี่ยนแปลงสัญญาณเริ่มต้นคือ Ao+ และ Bo+

รูปที่ 4.4 แสดงตัวอย่างของการวิเคราะห์หาการเปลี่ยนแปลงสัญญาณเริ่มต้นของแต่ละกรณีที่กล่าวมาข้างต้น



(a) (b) (c) (d)

รูปที่ 4.4 ตัวอย่างการวิเคราะห์การเปลี่ยนแปลงสัญญาณเริ่มต้น (a) กรณีที่ 1 (b) กรณีที่ 2.1 (c) กรณีที่ 2.2 (d) กรณีที่ 2.3

Step 1.4 เป็นขั้นตอนการเข้ารหัสค่าสัญญาณทั้งหมดในซิกแนลทรานสิชันกราฟของสัญญาณเริ่มต้นที่ได้มาจาก step 1.3 ให้อยู่ในรูปของเลขฐานสอง

Algorithm 2 Find encode start state

Step 2.1 Initial encode state to unknow value

Step 2.2 AddQueue(queue, start signal transition)

Step 2.3 While encode state still have unknow value

```
BeginWhile
    tmp_node = DeQueue(queue)
    If encode state value of tmp_node's signal is unknow value
        If tmp_node's value = 1 then
            Set encode state value of tmp_node's signal to 0
        ElseIf tmp_node's value = 0
            Set encode state value of tmp_node's signal to 1
        EndIf
    EndIf
```

AddQueue(queue, next tmp_node's transition)

EndWhile

End

จากขั้นตอนวิธีที่ 2 (algorithm 2) เราสามารถอธิบายรายละเอียดได้ดังนี้

Step 2.1 เป็นขั้นตอนการกำหนดค่าเริ่มต้นของ encode state ให้เป็นตัวไม่ทราบค่าก่อน โดยที่ encode state จะเป็นแถวลำดับของค่าสัญญาณทั้งหมดในซิกแนลทรานสิชันกราฟซึ่งจะบอกค่าของสัญญาณ ณ สถานะนั้นๆ

Step 2.2 เป็นขั้นตอนการเพิ่มสัญญาณเริ่มต้นที่ได้มาจาก step 1.3 ลงใน queue

Step 2.3 เป็นขั้นตอนการหาค่าในการเข้ารหัสของสถานะเริ่มต้น โดยเริ่มจากการเปลี่ยนแปลงสัญญาณเริ่มต้น จากนั้นจะทำการพิจารณาค่าของสัญญาณในลำดับแถวของ encode state ว่าเป็นตัวไม่ทราบค่าหรือไม่ ถ้าเป็นตัวไม่ทราบค่าเราจะทำการเปลี่ยนค่าโดยพิจารณาตามเงื่อนไขดังต่อไปนี้

ถ้าค่าของสัญญาณที่พิจารณามีค่าเท่ากับ 1 ให้เปลี่ยนค่าของสัญญาณในลำดับแถวของ encode state เป็น 0

ถ้าค่าของสัญญาณที่พิจารณามีค่าเท่ากับ 0 ให้เปลี่ยนค่าของสัญญาณในลำดับแถวของ encode state เป็น 1

หลังจากนั้นจะทำการเพิ่มการเปลี่ยนแปลงสัญญาณถัดไปจากสัญญาณที่ทำการพิจารณาอยู่ และทำซ้ำใน step 2.3 จนกว่าค่าของสัญญาณทั้งหมดใน encode state จะเป็นตัวที่ทราบค่าทั้งหมด

Step 1.5 เป็นขั้นตอนการสร้างกราฟสายงานสำหรับเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร ซึ่งจะมีวิธีขั้นตอนดังต่อไปนี้

Algorithm 3 Construct flow graph for STG

Step 3.1 Insert encode start state in flow graph which derived from algorithm 2

Step 3.2 Insert start signal node which derived from algorithm 1 (step 1.3) as encode start state's child Consider current temporal relation

Step 3.3 For each signal transitions in STG traverse by breadth firth search and start at reset signal transition

BeginFor

If encode signal transition not exit in flow graph

Insert encode signal transition (parent node) in flow graph

```
                    EndIf
                    Find children of signal transition in STG
                    For each children in STG
                    BeginFor
                        Consider causal relation of children in STG
                        If has causal relation properties
                            Encode children signal transiton
                            Insert children as encode signal transition's child
                        Else
                            Add children to arry share node
                        EndIf
                    EndFor
                    Consider current temporal relation
                    Consider causal relation with share node
                EndFor
End
```

ก่อนที่จะทำการอธิบายรายละเอียดของขั้นตอนวิธีที่ 3 จะขออธิบายโครงสร้างข้อมูลของกราฟสายงานก่อน โครงสร้างข้อมูลที่ใช้สำหรับกราฟสายงานจะแบ่งเป็นสองโครงสร้างข้อมูลคือ โครงสร้างข้อมูลของ parent node และโครงสร้างข้อมูลของ child node โดยที่โครงสร้างข้อมูลทั้งสองเป็นตัวชี้ ไปยังรายการเชื่อมโยงของระเบียน ซึ่งประกอบด้วยข้อมูลดังนี้

โครงสร้างข้อมูลของ parent node

1. การเปลี่ยนแปลงสัญญาณที่เข้ารหัสแล้ว
2. รายการเชื่อมโยงการเปลี่ยนแปลงสัญญาณ (child node)
3. ตัวชี้ไปยัง parent node ถัดไป

โครงสร้างข้อมูลของ child node

1. การเปลี่ยนแปลงสัญญาณที่เข้ารหัสแล้ว
2. การเปลี่ยนแปลงสัญญาณ
3. ค่าของการเปลี่ยนแปลงสัญญาณ
4. ตัวชี้ไปยัง child node ถัดไป

ดังนั้นเราจึงนำโครงสร้างข้อมูลนี้มาประยุกต์ใช้กับขั้นตอนวิธีที่ 3 ซึ่งมีรายละเอียดดังนี้

Step 3.1 เป็นขั้นตอนเริ่มแรกของการสร้างกราฟสายงาน โดยเราจะทำการแทรกโนดเริ่มต้นเข้าสู่กราฟสายงานซึ่งเป็นโครงสร้างข้อมูลแบบ parent node โดยที่เราจะนำการสถานะเริ่มต้นที่เข้ารหัสแล้วที่ได้จากขั้นตอนวิธีที่ 2 มาเป็นโนดเริ่มต้น ดังแสดงในรูปที่ 4.5 (a)

Step 3.2 เป็นการแทรกโนดการเปลี่ยนแปลงสัญญาณเริ่มต้นที่ได้จากการวิเคราะห์ในขั้นตอนวิธีที่ 1 (step 1.3) ให้เป็นโนดลูกของโนดเริ่มต้น (step 3.1) โดยที่โครงสร้างข้อมูลของโนดที่จะทำการแทรกเป็นโครงสร้างข้อมูลแบบ child node ถ้าหากว่าการเปลี่ยนแปลงสัญญาณเริ่มต้นมีมากกว่า 1 อัน เราจะต้องพิจารณาคุณสมบัติ concurrent temporal relation แล้วทำการสร้างโนดที่จะทำการแทรกเข้าไปในกราฟสายงาน โดยต้องคำนึงถึงลำดับของการเปลี่ยนแปลงสัญญาณที่เป็นไปได้ทั้งหมด ตัวอย่างเช่น โนดการเปลี่ยนแปลงสัญญาณเริ่มต้นของซิกแนลทรานสิชันกราฟของวงจร C-element คือการเปลี่ยนแปลงสัญญาณ Ai+ กับ Bi+ เมื่อทำการแทรกและพิจารณาคุณสมบัติ concurrent temporal ralation จะได้กราฟสายงานดังรูปที่ 4.5 (b) และ 4.5 (c)

Step 3.3 เป็นขั้นตอนการวิเคราะห์การเปลี่ยนแปลงสัญญาณในซิกแนลทรานสิชันกราฟ โดยจะทำการแวะผ่านซิกแนลทรานสิชันกราฟแบบ breadth-first search ซึ่งจะเริ่มต้นจากการเปลี่ยนแปลงสัญญาณที่ได้จากขั้นตอนวิธีที่ 1 (step 1.3)

ในขั้นแรกเราจะหาการเปลี่ยนแปลงสัญญาณขาออกของโนดที่กำลังพิจารณา หลังจากนั้นจะทำการเข้ารหัสสัญญาณขาออก และตรวจสอบคุณสมบัติ causal relation กับโนดที่สร้างไว้แล้วในกราฟสายงาน แล้วทำการแทรกโนดของสัญญาณขาออกนั้นในกราฟสายงาน โดยที่โครงสร้างข้อมูลของสัญญาณขาออกที่จะทำการแทรกเป็นโครงสร้างข้อมูลแบบ child node ถ้าหากว่าไม่มีคุณสมบัติ causal relation ให้ทำการเพิ่มสัญญาณขาออกของโนดที่กำลังพิจารณาในระเบียน share node จากนั้นพิจารณาจำนวนโนดลูกของการเปลี่ยนแปลงสัญญาณที่ทำการพิจารณาถ้ามีค่ามากกว่า 1 โนด เราจะต้องพิจารณาคุณสมบัติ concurrent temporal relation แล้วทำการสร้างโนดที่จะทำการแทรกเข้าไปในกราฟสายงาน โดยต้องคำนึงถึงลำดับของการเปลี่ยนแปลงสัญญาณที่เป็นไปได้ทั้งหมด และนำการเปลี่ยนแปลงสัญญาณที่ได้มาพิจารณาคุณสมบัติ causal relation กับสัญญาณในระเบียน share node ถ้าหากพบว่ามีคุณสมบัติ causal relation ให้แทรกโนดการเปลี่ยนแปลงสัญญาณนั้นๆ เข้าในกราฟสายงาน และแทรกโนดการเปลี่ยนแปลงสัญญาณ share node ให้เป็นโนดลูก

ตัวอย่างต่อไปนี้จะแสดงขั้นตอนวิธีในการสร้างกราฟสายงานของซิกแนลทรานสิชันกราฟของวงจร C-element โดยที่ Ai กับ Bi เป็นการเปลี่ยนแปลงสัญญาณเริ่มต้นและการสถานะเริ่มต้นที่เข้ารหัสแล้วคือ 00001010 ซึ่งเมื่อทำการแวะผ่านซิกแนลทรานสิชันกราฟแบบ breadth-first search

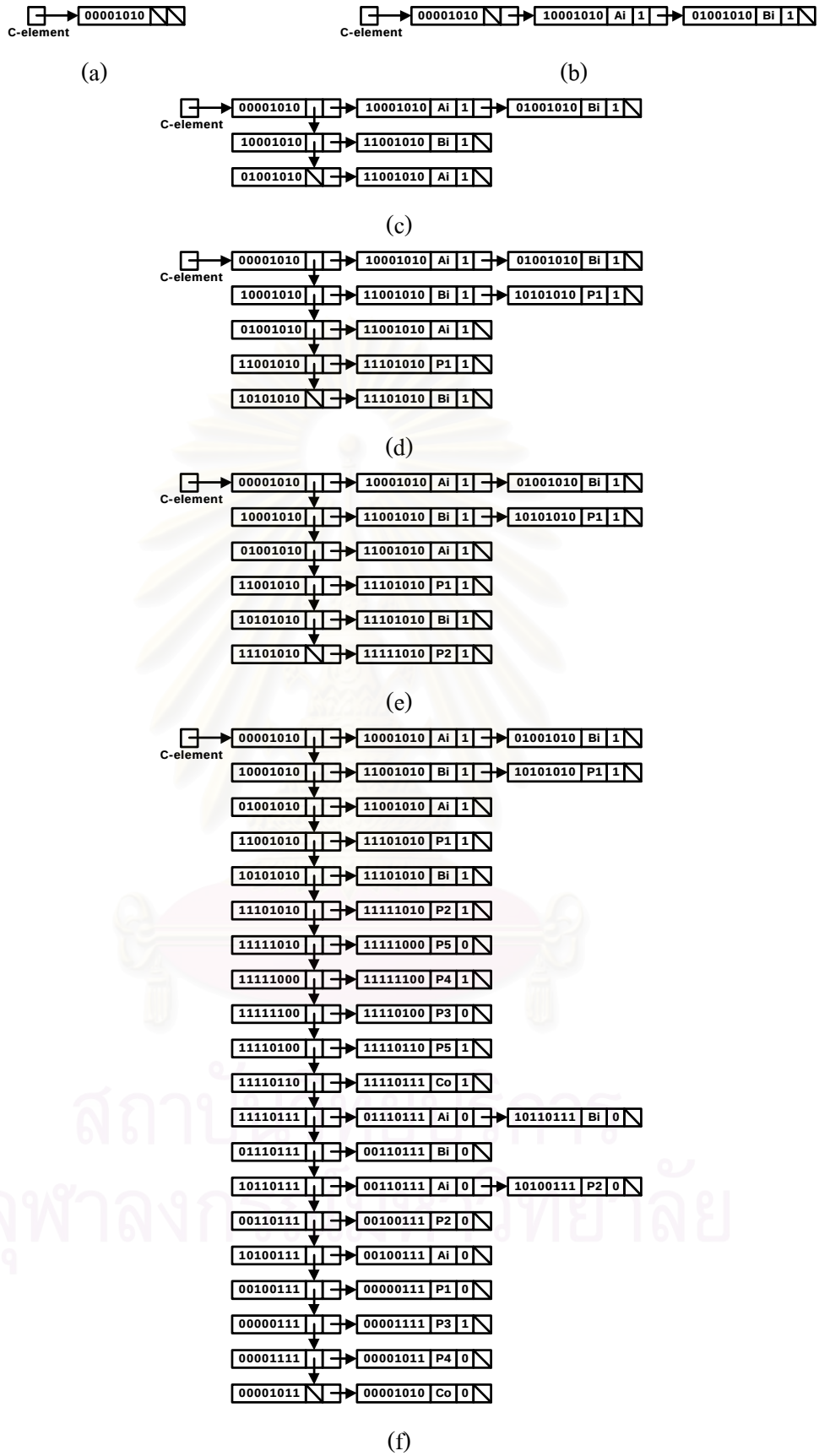
จะได้ลำดับการเปลี่ยนแปลงสัญญาณที่จะทำการพิจารณาดังนี้ Ai+ Bi+ P1+ P2+ P5- P4+ P3- P5+ Co+ Ai- Bi- P1- P2- P3+ P4- Co-

วิเคราะห์การเปลี่ยนแปลงสัญญาณ Ai+ เมื่อพิจารณาในซิกแนลทรานสิชันกราฟ จะพบว่าการเปลี่ยนแปลงสัญญาณขาออกคือ P1+ และมีคุณสมบัติ causal relation ถูกต้องตรงตามซิกแนลทรานสิชันกราฟดังนั้นจึงทำการแทรกโนดการเปลี่ยนแปลงสัญญาณ Ai+ ที่เข้ารหัสแล้วเข้าสู่กราฟสายงานซึ่งมีโครงสร้างข้อมูลแบบ parent node และแทรกโนดการเปลี่ยนแปลงสัญญาณ P1+ ให้เป็นโนดลูกซึ่งมีโครงสร้างข้อมูลแบบ child node ทำให้โนดการเปลี่ยนแปลงสัญญาณ Ai+ ที่เข้ารหัสแล้วมีจำนวนโนดลูกเท่ากับ 2 โนด เราจะต้องพิจารณาคุณสมบัติ concurrent temporal relation แล้วทำการสร้างโนดที่จะทำการแทรกเข้าไปในกราฟสายงาน โดยต้องคำนึงถึงลำดับของการเปลี่ยนแปลงสัญญาณที่เป็นไปได้ทั้งหมด ดังแสดงในรูปที่ 4.5 (d)

วิเคราะห์การเปลี่ยนแปลงสัญญาณ Bi+ เมื่อพิจารณาในซิกแนลทรานสิชันกราฟ จะพบว่าการเปลี่ยนแปลงสัญญาณขาออกคือ P2+ แต่ไม่มีคุณสมบัติ causal relation เนื่องจาก P2+ ต้องรอให้เกิดการเปลี่ยนแปลงสัญญาณของ P1+ และ Bi+ เรียบร้อยเสียก่อน เราจึงทำการเพิ่มการเปลี่ยนแปลงสัญญาณ P2+ ในระเบียน share node

วิเคราะห์การเปลี่ยนแปลงสัญญาณ P1+ เมื่อพิจารณาในซิกแนลทรานสิชันกราฟ จะพบว่าการเปลี่ยนแปลงสัญญารขาออกคือ P2+ แต่ไม่มีคุณสมบัติ causal relation เนื่องจาก P2+ ต้องรอให้เกิดการเปลี่ยนแปลงสัญญาณของ P1+ และ Bi+ เรียบร้อยเสียก่อน เนื่องจากโนด P1+ มีโนดลูกเราจึงต้องทำการพิจารณาคุณสมบัติ causal relation กับสัญญาณในระเบียน share node นั่นคือสัญญาณ P2+ ปรากฏว่ามีคุณสมบัติ causal relation ถูกต้องตรงตามซิกแนลทรานสิชันกราฟ ดังนั้นจึงทำการแทรกโนดการเปลี่ยนแปลงสัญญาณนั้นๆ เข้าสู่กราฟสายงานซึ่งมีโครงสร้างข้อมูลแบบ parent node และแทรกโนดการเปลี่ยนแปลงสัญญาณในระเบียน share node ให้เป็นโนดลูก ดังแสดงในรูปที่ 4.5 (e)

สำหรับการวิเคราะห์การเปลี่ยนแปลงสัญญาณอื่นๆ ก็จะทำเช่นเดียวกันกับที่ยกตัวอย่าง และเมื่อเราทำการวิเคราะห์ครบทุกการเปลี่ยนแปลงสัญญาณในซิกแนลทรานสิชันกราฟแล้วจะได้กราฟสายงานสำหรับเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร C-element ดังแสดงในรูปที่ 4.5 (f)



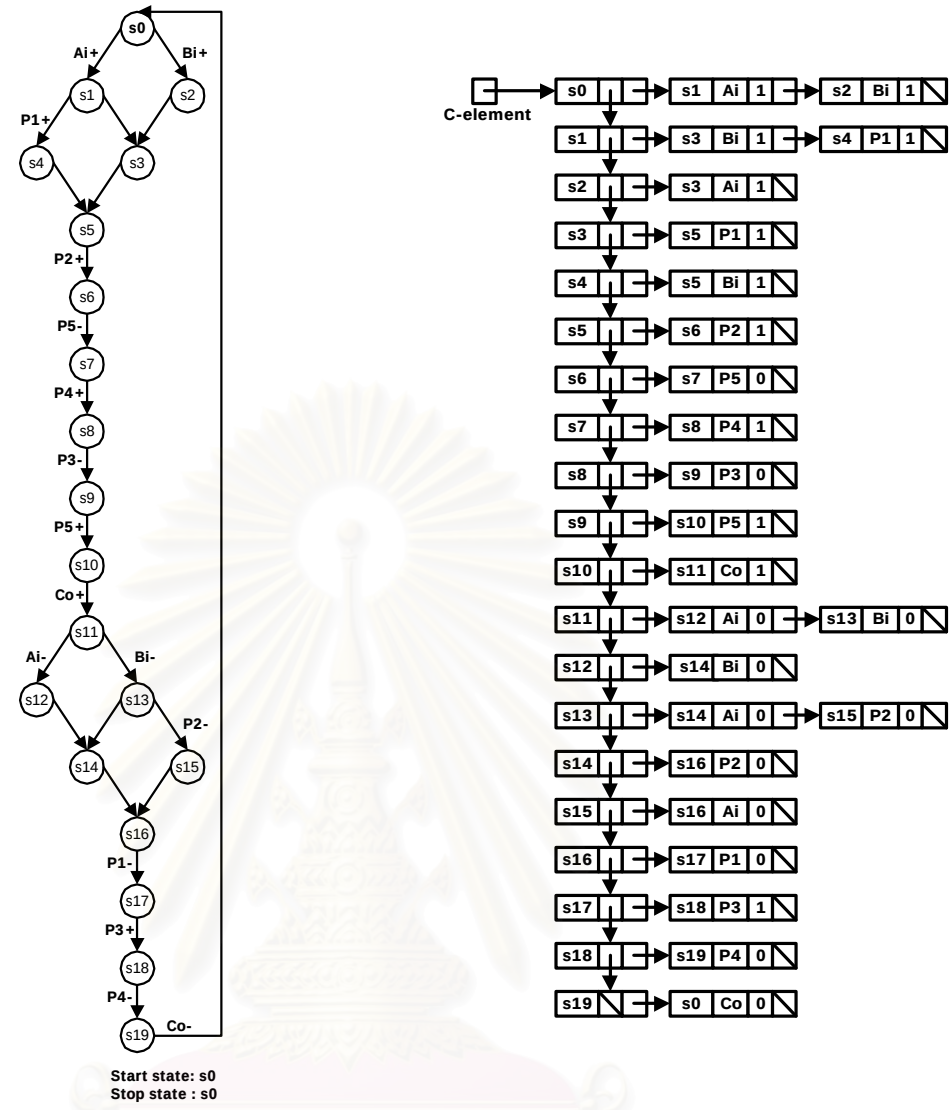
รูปที่ 4.5 ขั้นตอนวิธีการสร้างเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร C-element

เนื่องจากกราฟสายงานที่ได้นี้จะนำไปใช้เป็นเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจรเพื่อจะใช้ในการทวนสอบวงจรต่อไป เราจะทำการเปลี่ยนค่าของการเปลี่ยนแปลงสัญญาณที่เข้ารหัสแล้ว โดยการตั้งชื่อขึ้นใหม่เพื่อที่จะสะดวกในการอ้างอิง โดยเราจะสร้างตารางเทียบค่าของการเปลี่ยนแปลงสัญญาณที่เข้ารหัสแล้วกับสถานะ และทำการแก้ไขค่าของการเปลี่ยนแปลงสัญญาณที่เข้ารหัสแล้วตามตารางดังกล่าว ดังแสดงในตารางที่ 4.1 จะได้กราฟสายงานสำหรับเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร ดังแสดงในรูปที่ 4.6

ตารางที่ 4.1 ตารางเทียบค่าระหว่าง state กับ encode signal สำหรับเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร C-element

State	**Encode Signal**
S0	00001010
S1	10001010
S2	01001010
S3	11001010
S4	10101010
S5	11101010
S6	11111010
S7	11111000
S8	11111100
S9	11110100
S10	11110110
S11	11110111
S12	01110111
S13	10110111
S14	00110111
S15	10100111
S16	00100111
S17	00000111
S18	00001111
S19	00001011

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



(a) (b)

รูปที่ 4.6 เครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร C-element ในรูปแบบของกราฟ (a) และโครงสร้างของข้อมูล (b)

ขั้นตอนวิธีการสร้างเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจรที่กล่าวมาจะมีการรับอินพุตเป็นซิกแนลทรานสิชันกราฟ และจะทำการสร้าง state graph จากซิกแนลทรานสิชันกราฟ ซึ่งสามารถวิเคราะห์ความซับซ้อนของขั้นตอนวิธีได้ดังนี้ โดยให้จำนวนครั้งของการเปลี่ยนแปลงสัญญาณเป็นเวลา

สำหรับซิกแนลทรานสิชันกราฟที่มีเซตของสัญญาณขนาด S ตัว และมีเซตการเปลี่ยนแปลงสัญญาณขนาด T ตัว

กรณีที่ใช้เวลาน้อยที่สุด จะต้องมีสัญญาณอินพุตอย่างน้อย 1 ตัว, สัญญาณที่ไม่ใช่สัญญาณอินพุต อย่างน้อย 1 ตัว และทำการเปลี่ยนแปลงสัญญาณอย่างน้อยที่สุดเท่ากับจำนวนของ T ดังนั้นเวลาที่ดีที่สุดของขั้นตอนนี้คือ T+T=2T นั่นคือ O(T)

กรณีที่ใช้เวลามากที่สุด จะต้องมีเซตของสัญญาณอย่างมากที่สุดขนาด S ตัว และทำการเปลี่ยนแปลงสัญญาณอย่างมากที่สุดเท่ากับจำนวนของ T*n โดยที่ n คือลำดับครั้งของการเปลี่ยนแปลงสัญญาณ ดังนั้นเวลาที่มากที่สุดของขั้นตอนนี้คือ S*T*n นั่นคือ O(STn)

4.2 สรุป

ในบทนี้ได้กล่าวถึงขั้นตอนวิธีการสร้างเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร โดยใช้เทคนิคของ state graph อย่างละเอียด โดยเริ่มจากการอธิบายรูปแบบของไฟล์อินพุตซิกแนลทรานสิชันกราฟ, การวิเคราะห์หาสัญญาณเริ่มต้น, การเข้ารหัสสัญญาณ และขั้นตอนการสร้างกราฟ พร้อมทั้งยกตัวอย่างประกอบ สำหรับเวลาที่ใช้ในการทำงานของขั้นตอนวิธีที่ได้นำเสนอคือ O(T) ในกรณีที่ใช้เวลาน้อยที่สุด และ O(STn) ในกรณีที่ใช้เวลามากที่สุด โดยที่ S คือขนาดของเซตสัญญาณ, T คือขนาดของเซตการเปลี่ยนแปลงสัญญาณ และ n คือลำดับครั้งของการเปลี่ยนแปลงสัญญาณ

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 5
เครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์

ในบทนี้เราจะกล่าวถึงรายละเอียดของขั้นตอนวิธีการสร้างเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ โดยจะอธิบายถึงการนำเอาพฤติกรรมของเกตย่อยในวงจรมาแสดงในรูปแบบของกราฟ และนำกราฟที่ได้ทั้งหมดมาวิเคราะห์หาการเปลี่ยนแปลงสัญญาณที่จะเกิดขึ้นโดยใช้ตัวกระทำการ * ในพีชคณิตเชิงกระบวนการที่ได้กล่าวไว้ในบทที่ 2 และคำนึงถึงโมเดลสิ่งแวดล้อมเป็นหลัก สำหรับขั้นตอนวิธีในการสร้างเครื่องจักรสถานะจำกัดจะมีไฟล์อินพุตเป็นวงจรที่ได้จากการสังเคราะห์ซึ่งอยู่ในรูปแบบของนิพจน์ที่แทนเกตแต่ละเกตในวงจรมาเรียงต่อกัน และยังมี delete signal ซึ่งเป็นสัญญาณในวงจรที่ไม่ต้องการให้แสดงในกราฟเครื่องจักรสถานะจำกัด รูปแบบของเท็กไฟล์ และรูปแบบของนิพจน์จะแสดงในรูปที่ 5.1

ชื่อวงจร = นิพจน์ 1 * นิพจน์ 2 * … * นิพจน์ n [- (delete signal 1, delete signal 2, …)]

(a)

gate n (input signal 1, input signal 2, …, input signal n, output signal, start state)

(b)

รูปที่ 5.1 (a) รูปแบบของเท็กไฟล์ของวงจรที่ได้จากการสังเคราะห์
(b) รูปแบบของนิพจน์ในเท็กไฟล์

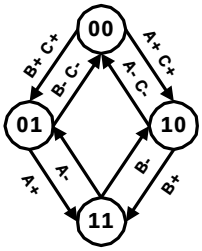
5.1 กราฟพฤติกรรมของเกต

เกตแต่ละเกตเราสามารถที่จะนำพฤติกรรมมาอธิบายในรูปแบบของกราฟได้ โดยที่โนดของกราฟจะแสดงค่าสัญญาณของสายสัญญาณแต่ละเส้น ณ เวลาใดเวลาหนึ่ง และเส้นที่เชื่อมระหว่างโนดจะแสดงถึงการเปลี่ยนแปลงของค่าสัญญาณ ซึ่งก็คือ state graph นั่นเอง

ตัวอย่างเช่นพฤติกรรมของเกต OR ที่มี 2 อินพุตแสดงดังตารางที่ 5.1 และพฤติกรรมอธิบายในรูปแบบของกราฟดังแสดงในรูปที่ 5.2

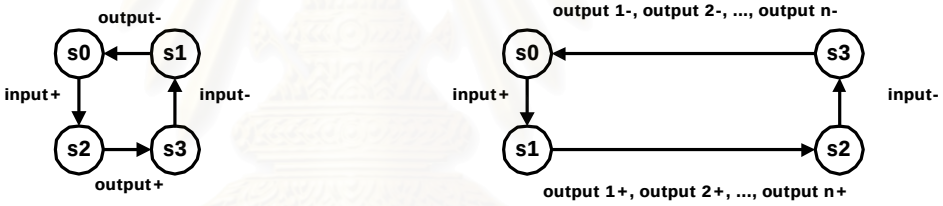
ตารางที่ 5.1 พฤติกรรมของเกต OR ขนาด 2 อินพุต

Input A	Input B	Output C
0	0	0
0	1	1
1	0	1
1	1	1



รูปที่ 5.2 กราฟพฤติกรรมของเกต OR ขนาด 2 อินพุต

เนื่องจากงานวิจัยนี้เป็นการทวนสอบวงจรอสมวารแบบควอไซดีเลย์อินเซนซิทีฟ จะไม่มีสัญญาณนาฬิกาเป็นตัวควบคุมจังหวะในการเกิดอินพุต และเอาต์พุตเราจึงต้องเพิ่มข้อบังคับเพื่อที่จะให้การเปลี่ยนแปลงสัญญาณขาออกเกิดการเปลี่ยนแปลงสัญญาณหลังจากสัญญาณขาเข้าเกิดการเปลี่ยนแปลงสัญญาณแล้ว และการเปลี่ยนแปลงสัญญาณขาเข้าจะต้องรอให้เกิดการเปลี่ยนแปลงของสัญญาณขาออกเสร็จเรียบร้อยเสียก่อน ซึ่งกราฟพฤติกรรมของข้อบังคับที่กล่าวมาเราจะเรียกว่า causal delay และในกรณีที่เกิดการกระจายของสัญญาณในโมเดลความหน่วงแบบควอไซดีเลย์อินเซนซิทีฟจะถือว่าความหน่วงในสายสัญญาณทุกเส้นมีค่าเท่ากันทั้งหมด (isochronic fork) ดังแสดงในรูปที่ 5.3



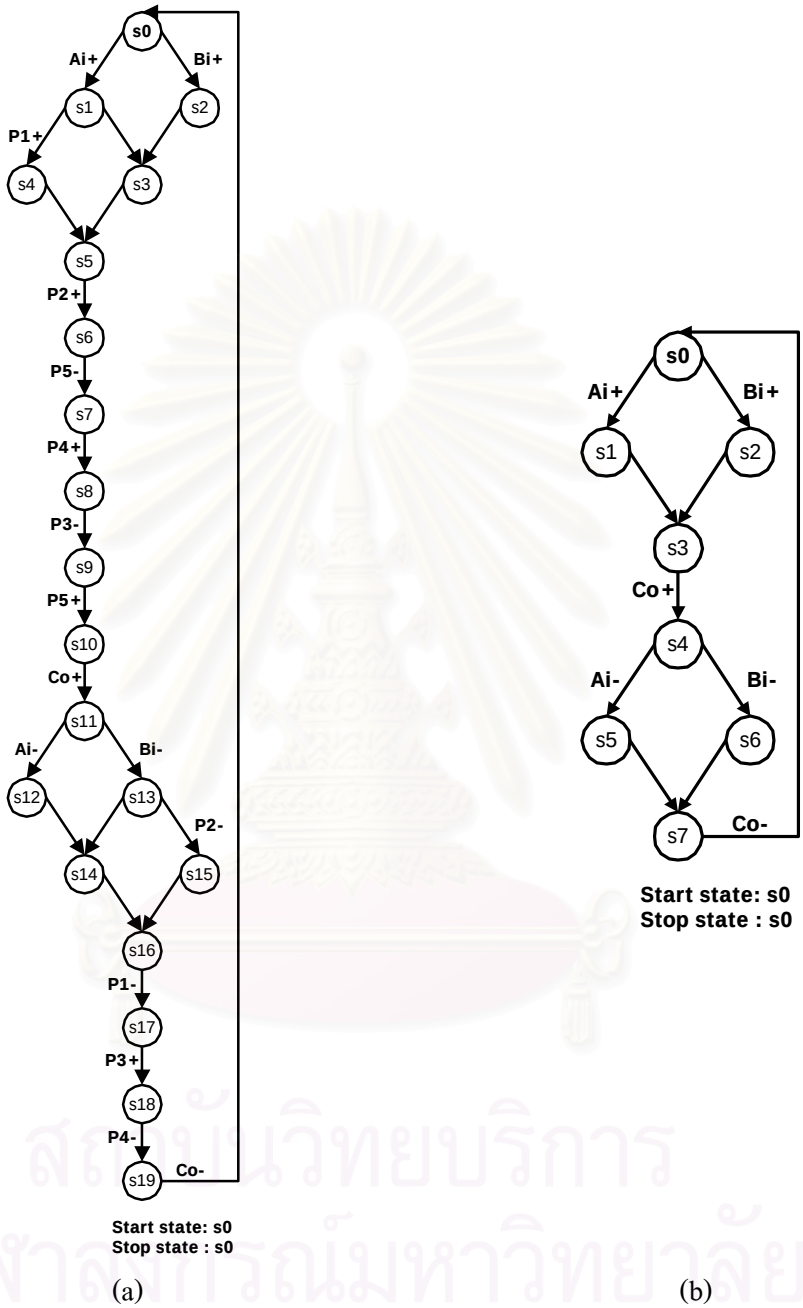
(a) (b)

รูปที่ 5.3 (a) กราฟพฤติกรรมของ causal delay (b) กราฟพฤติกรรมของ isochronic fork

5.2 โมเดลสิ่งแวดล้อม

ปัญหาที่สำคัญของสร้างเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ก็คือจำนวนสถานะที่มีมากเกินไป (state explosion) ซึ่งจะแตกต่างจากเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร เนื่องจากเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์จะยอมรับอินพุตทุกกรณีในขณะที่เครื่องจักรสถานะจำกัดของคุณลักษณะของวงจรจะยอมรับอินพุตที่ถูกออกแบบไว้เท่านั้น ดังนั้นเพื่อที่จะลดขนาดของจำนวนสถานะที่จะเกิดขึ้นในขั้นตอนการสร้างเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์เราจึงคำนึงถึงสิ่งแวดล้อมซึ่งจะเป็นตัวกำหนดจังหวะการเปลี่ยนแปลงสัญญาณอินพุต และเอาต์พุตโดยจะยึดตามโมเดลสิ่งแวดล้อมของคุณลักษณะของวงจรเป็นหลัก

กราฟโมเดลสิ่งแวดล้อมสามารถพิจารณาได้จากเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจรซึ่งก็คือเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจรที่แสดงความสัมพันธ์ของการเปลี่ยนแปลงสัญญาณอินพุต และเอาต์พุต ดังแสดงในรูปที่ 5.4



รูปที่ 5.4 (a) เครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร C-element
(b) กราฟโมเดลสิ่งแวดล้อมของวงจร C-element

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

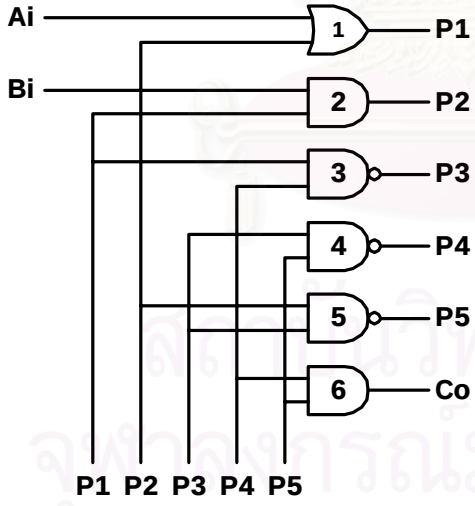
5.3 การสร้างเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์

สำหรับขั้นตอนนี้จะเป็นขั้นตอนการสร้างเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ มีขั้นตอนดังต่อไปนี้

Algorithm 1	Construct FSM of Circuit Implementation
Step 1.1	Lexcical analysis
Step 1.2	Syntax analysis
Step 1.3	Add Causal Delay to Gate Behaviour Graph
Step 1.4	Construct Finite State Machine for Circuit Implementation
End	

จากขั้นตอนวิธีที่ 1 (algorithm 1) เราสามารถอธิบายรายละเอียดได้ดังนี้

Step 1.1 เป็นขั้นตอนการอ่านแฟ้มข้อความซึ่งเป็นอินพุตไฟล์ที่ให้แสดงแทนวงจรที่ได้จากการสังเคราะห์แล้วทำการวิเคราะห์แฟ้มข้อความทีละอักขระ เพื่อแยกออกเป็นนิพจน์ ซึ่งนิพจน์เหล่านี้จะถูกนำไปใช้ในการวิเคราะห์วากยสัมพันธ์ต่อไป ตัวอย่างวงจรที่ได้จากการสังเคราะห์และเท็กไฟล์ที่ใช้เป็นไฟล์อินพุตแทนวงจรที่ได้จากการสังเคราะห์ของวงจร C-element ดังแสดงในรูปที่ 5.5



(a)

```
C-element=
or2(Ai, P2, P1, s0) *
and2(Bi, P1, P2, s0) *
nand2(P1, P4, P3, s0) *
nand2(P3, P5, P4, s3) *
nand2(P2, P3, P5, s1) *
and2(P4, P5, Co, s1)
```

(b)

รูปที่ 5.5 (a) วงจรที่ได้จากการสังเคราะห์ (b) ไฟล์อินพุตของวงจรที่ได้จากการสังเคราะห์

Step 1.2 เป็นขั้นตอนการวิเคราะห์วากยสัมพันธ์ เพื่อวิเคราะห์แต่ละนิพจน์ที่ได้จาก step 1.1 มาสร้างเป็นกราฟสายงานที่แสดงพฤติกรรมของเกตในแต่ละนิพจน์จาก library ที่กำหนดไว้ ซึ่งต้องพิจารณาค่าของการเปลี่ยนแปลงสัญญาณขาเข้า และขาออกของเกตเพื่อที่จะใช้ในการเลือก

สถานะเริ่มต้นของกราฟสายงาน โครงสร้างของข้อมูลที่ใช้สำหรับกราฟสายงานที่แสดงพฤติกรรมของเกตจะแบ่งเป็นสองโครงสร้างข้อมูลคือ โครงสร้างข้อมูล gate_parent node และโครงสร้างข้อมูล gate_child nodeโดยที่โครงสร้างข้อมูลทั้งสองชนิดเป็นตัวชี้ไปยังรายการเชื่อมโยงของระเบียน ซึ่งมีประกอบด้วยข้อมูลดังนี้

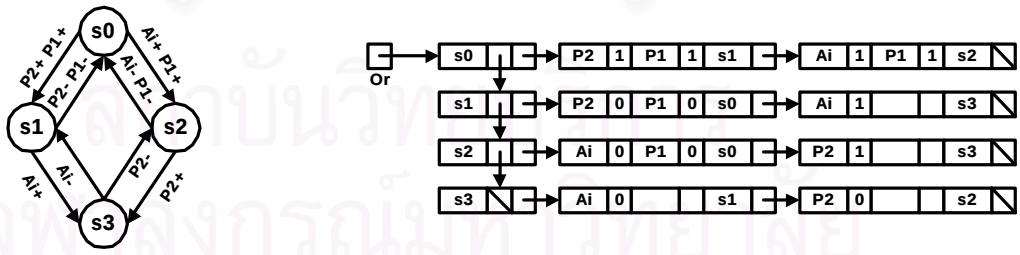
โครงสร้างข้อมูลของ gate_parent node

1. สถานะ
2. รายการเชื่อมโยงของการเปลี่ยนแปลงสัญญาณ (gate_child node)
3. ตัวชี้ไปยัง gate_parent node ถัดไป

โครงสร้างข้อมูลของ gate_child node

1. การเปลี่ยนแปลงสัญญาณขาเข้า
2. การเปลี่ยนแปลงสัญญาณขาออก
3. ค่าของการเปลี่ยนแปลงสัญญาณขาเข้า
4. ค่าของการเปลี่ยนแปลงสัญญาณขาออก
5. สถานะ
6. ตัวชี้ไปยัง gate_child node ถัดไป

ตัวอย่างของกราฟสายงานที่แสดงพฤติกรรมของเกต OR ในวงจร C-element ซึ่งมีการเปลี่ยนแปลงสัญญาณ Ai และการเปลี่ยนแปลงสัญญาณ P2 เป็นการเปลี่ยนแปลงสัญญาณขาเข้า การเปลี่ยนแปลงสัญญาณ P1 เป็นการเปลี่ยนแปลงสัญญาณขาออก และมีสถานะเริ่มต้นที่สถานะ s0 ดังแสดงในรูปที่ 5.6



(a) (b)

รูปที่ 5.6 กราฟพฤติกรรมของเกต OR ในวงจร C-element ในรูปแบบของกราฟ (a) และโครงสร้างของข้อมูล (b)

Step 1.3 เป็นขั้นตอนการเพิ่ม causal delay ให้กับสัญญาณขาออกของกราฟแสดงพฤติกรรมของเกต เพื่อที่จะให้สัญญาณขาออกเกิดการเปลี่ยนแปลงสัญญาณหลังจากสัญญาณขาเข้า

เกิดการเปลี่ยนแปลงสัญญาณแล้ว และการเปลี่ยนแปลงของสัญญาณขาเข้าจะต้องรอให้เกิดการเปลี่ยนแปลงของสัญญาณขาออกเสร็จเรียบร้อยเสียก่อน

Algorithm 2 Add Delay to Gate Behaviour Graph
Step 2.1 Create Causal Delay Behaviour Graph
Step 2.2 Combine Gate Behaviour Graph and Delay Behaviour Graph
End

จากขั้นตอนวิธีที่ 2 (algorithm 2) เราสามารถอธิบายรายละเอียดได้ดังนี้

Step 2.1 เป็นขั้นตอนวิธีการสร้างกราฟสายงานที่แสดงพฤติกรรมของ causal delay ที่ใช้ในการเพิ่ม causal delay ให้กับสัญญาณขาออกจาก library ที่กำหนดไว้ ซึ่งต้องพิจารณาค่าของการเปลี่ยนแปลงสัญญาณขาออกของเกตที่จะทำการเพิ่ม causal delay เพื่อที่จะใช้ในการเลือกสถานะเริ่มต้นของกราฟสายงาน โครงสร้างของข้อมูลที่ใช้สำหรับกราฟสายงานที่แสดงพฤติกรรมของ causal delay จะแบ่งเป็นสองโครงสร้างข้อมูลคือ โครงสร้างข้อมูล parent node และโครงสร้างข้อมูล child node โดยที่โครงสร้างข้อมูลทั้งสองชนิดเป็นตัวชี้ไปยังรายการเชื่อมโยงของระเบียน ซึ่งมีประกอบด้วยข้อมูลดังนี้

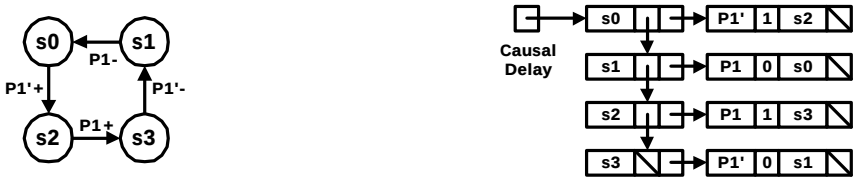
โครงสร้างข้อมูลของ parent node

1. สถานะ
2. รายการเชื่อมโยงของการเปลี่ยนแปลงสัญญาณ (child node)
3. ตัวชี้ไปยัง parent node ถัดไป

โครงสร้างข้อมูลของ child node

1. การเปลี่ยนแปลงสัญญาณ
2. ค่าของการเปลี่ยนแปลงสัญญาณ
3. สถานะ
4. ตัวชี้ไปยัง child ถัดไป

ตัวอย่างกราฟสายงานที่แสดงพฤติกรรมของ causal delay ที่ใช้ในการเพิ่ม causal delay ให้กับสัญญาณขาออก P1 ของเกต OR (step 1.2) ในวงจร C-element และมีสถานะเริ่มต้นที่สถานะ s0 ดังแสดงในรูปที่ 5.7



(a) (b)

รูปที่ 5.7 กราฟพฤติกรรมของ causal delay ที่ใช้ในการเพิ่ม causal delay ให้กับสัญญาณขาออก P1 ของเกต OR ในวงจร C-element ในรูปแบบของกราฟ (a) และโครงสร้างของข้อมูล (b)

Step 2.2 เป็นขั้นตอนการเพิ่ม causal delay ให้กับสัญญาณขาออกของเกต โดยการนำกราฟสายงานที่แสดงพฤติกรรมของเกต (step 1.2) กับกราฟสายงานที่แสดงพฤติกรรมของ causal delay (step 2.1) มารวมกัน โดยการนำเอาสัญญาณขาออกของเกตมาเป็นสัญญาณขาเข้าของ causal delay สถานะของกราฟสายงานใหม่ที่ได้จะเกิดจากการนำเอาสถานะของกราฟสายงานที่แสดงพฤติกรรมของเกต และสถานะของกราฟสายงานที่แสดงพฤติกรรมของ causal delay มารวมกัน

โครงสร้างข้อมูลของกราฟสายงานใหม่ที่ได้จะใช้โครงสร้างข้อมูลแบบเดียวกับโครงสร้างข้อมูลของกราฟสายงานที่แสดงพฤติกรรมของ causal delay (step 2.1)

Algorithm 3 Combine gate behaviour graph and causal delay behaviour graph

Step 3.1 Create and insert start state in flow graph

Step 3.2 AddQueue(queue, start_state)

Step 3.3 While (queue is Not Empty)

BeginWhile

tmp_state = DeQueue(queue)

If (tmp_state first part has only input change)

Insert input signal transition into flow graph

new_state first part = state of input signal transition changed in causal gate behaviour graph

new_state second part not change

ElseIf (tmp_state first part has input & output change and

tmp_state second part has input change and

output signal transition in first part = input signal transition in second part)

Insert input signal transition into flow graph

new_state first part = state of input signal transition changed

in gate behaviour graph

new_state second part = state of input signal transition changed

in delay behaviour graph

ElseIf (tmp_state second part has only output change)

Insert output signal transition into flow graph

new_state first part not change

new_state second part = state of output signal transition changed

in causal delay behaviour graph

EndIf

AddQueue(queue, new_state)

EndWhile

End

จากขั้นตอนวิธีที่ 3 (algorithm 3) เราสามารถอธิบายรายละเอียดได้ดังนี้

Step 3.1 เป็นขั้นตอนวิธีในการสร้างสถานะเริ่มต้น และใช้เป็นโนดเริ่มต้นในการแทรกเข้าสู่กราฟสายงาน สถานะเริ่มต้นสำหรับกราฟสายงานของเกตที่ได้จากการเพิ่ม causal delay ให้กับสัญญาณขาออกเกิดจากการนำเอาสถานะเริ่มต้นของทั้งกราฟสายงานที่แสดงพฤติกรรมของเกตและกราฟสายงานที่แสดงพฤติกรรมของ causal delay มาต่อกันตามลำดับ ดังแสดงในรูปที่ 5.8 (a)

Step 3.2 เป็นขั้นตอนวิธีในการเพิ่มสถานะเริ่มต้นที่ได้มาจาก Step 3.1 ลงใน queue

Step 3.3 เป็นขั้นตอนวิธีในการวิเคราะห์การเปลี่ยนแปลงสัญญาณในกราฟที่แสดงพฤติกรรมของเกต และกราฟที่แสดงพฤติกรรมของ causal delay โดยเริ่มพิจารณาจากสถานะเริ่มต้น (step 3.1) ซึ่งค่าของสถานะในส่วนแรกเป็นสถานะของกราฟที่แสดงพฤติกรรมของเกต และค่าของสถานะในส่วนที่สองเป็นสถานะของกราฟที่แสดงพฤติกรรมของ causal delay สามารถพิจารณาได้ดังนี้

ถ้าสถานะของกราฟที่แสดงพฤติกรรมของเกตที่ทำการพิจารณาเกิดการเปลี่ยนแปลงสัญญาณขาเข้าเพียงอย่างเดียว จะเกิดการเปลี่ยนแปลงของสัญญาณขาเข้าของกราฟที่แสดงพฤติกรรมของเกต จากสถานะที่ทำการพิจารณาไปยังสถานะใหม่ โดยที่ค่าของสถานะในส่วนแรกจะได้จากสถานะของกราฟที่แสดงพฤติกรรมของเกตสำหรับการเปลี่ยนแปลงสัญญาณขาเข้า และค่าของสถานะในส่วนที่สองจะยังคงเดิม

ถ้าสถานะของกราฟที่แสดงพฤติกรรมของเกตที่ทำการพิจารณาเกิดการเปลี่ยนแปลงสัญญาณขาเข้า กับการเปลี่ยนแปลงสัญญาณขาออก และสถานะของกราฟที่แสดงพฤติกรรมของ causal delay ที่ทำการพิจารณาเกิดการเปลี่ยนแปลงสัญญาณขาเข้า โดยที่การเปลี่ยนแปลงสัญญาณ

ขาออกของกราฟที่แสดงพฤติกรรมของเกต กับการเปลี่ยนแปลงสัญญาณขาเข้าของกราฟที่แสดงพฤติกรรมของ causal delay มีการเปลี่ยนแปลงของสัญญาณ และค่าของการเปลี่ยนแปลงสัญญาณเหมือนกัน จะเกิดการเปลี่ยนแปลงของสัญญาณขาเข้าของกราฟที่แสดงพฤติกรรมของเกตจากสถานะที่ทำการพิจารณาไปยังสถานะใหม่ โดยที่ค่าของสถานะในส่วนแรกจะได้จากสถานะของกราฟที่แสดงพฤติกรรมของเกตสำหรับการเปลี่ยนแปลงสัญญาณขาเข้า และค่าของสถานะในส่วนที่สองจะได้จากสถานะของกราฟที่แสดงพฤติกรรมของ causal delay สำหรับการเปลี่ยนแปลงสัญญาณขาออก

ถ้าสถานะของกราฟที่แสดงพฤติกรรมของ causal delay ที่ทำการพิจารณาเกิดการเปลี่ยนแปลงสัญญาณขาออกเพียงอย่างเดียว จะเกิดการเปลี่ยนแปลงของสัญญาณขาออกของกราฟที่แสดงพฤติกรรมของ causal delay จากสถานะที่ทำการพิจารณาไปยังสถานะใหม่ โดยที่ค่าของสถานะในส่วนแรกจะยังคงเดิม และค่าของสถานะในส่วนที่สองจะได้จากสถานะของกราฟที่แสดงพฤติกรรมของ causal delay สำหรับการเปลี่ยนแปลงสัญญาณขาออก

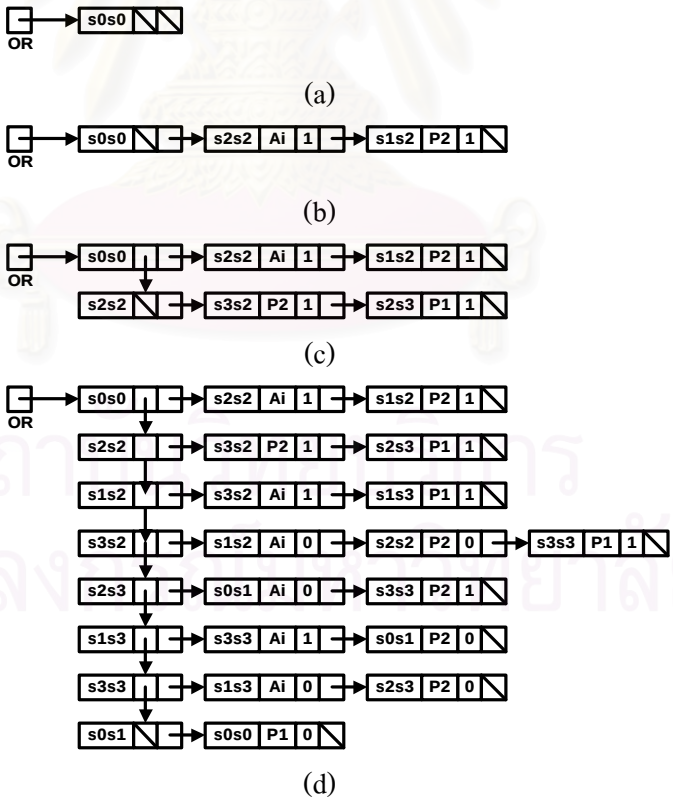
ตัวอย่างต่อไปนี้จะแสดงขั้นตอนวิธีในการสร้างกราฟสายงานที่เกิดจากการเพิ่ม causal delay ให้กับการเปลี่ยนแปลงสัญญาณขาออก P1 ของเกต OR ในวงจร C-element ซึ่งมี Ai, P2 เป็นสัญญาณขาเข้า P1’ เป็นสัญญาณขาออก และ P1’ เป็นสัญญาณขาเข้า P1 เป็นสัญญาณขาออกของกราฟที่แสดงพฤติกรรม causal delay โดยที่ s0 s0 เป็นสถานะเริ่มต้นของกราฟสายงานที่จะทำการสร้าง ซึ่งค่าของสถานะในส่วนแรก s0 คือสถานะเริ่มต้นของกราฟที่แสดงพฤติกรรมของเกต และค่าของสถานะในส่วนที่สอง s0 คือสถานะเริ่มต้นของกราฟที่แสดงพฤติกรรมของ causal delay

วิเคราะห์สถานะ s0 s0 ที่สถานะ s0 ของกราฟที่แสดงพฤติกรรมของเกตเกิดการเปลี่ยนแปลงของสัญญาณคือ Ai, P1’ ไปยังสถานะ s2 กับ P2, P1’ ไปยังสถานะ s1 และที่สถานะ s0 ของกราฟที่แสดงพฤติกรรมของ causal delay เกิดการเปลี่ยนแปลงของสัญญาณคือ P1’ ไปยังสถานะ s2 จากการวิเคราะห์จะได้ว่ากราฟที่แสดงพฤติกรรมของเกตเกิดการเปลี่ยนแปลงสัญญาณขาเข้า กับสัญญาณขาออก และกราฟที่แสดงพฤติกรรมของ causal delay เกิดการเปลี่ยนแปลงสัญญาณขาเข้า โดยที่การเปลี่ยนแปลงสัญญาณขาออกของกราฟที่แสดงพฤติกรรมของเกต กับการเปลี่ยนแปลงสัญญาณขาเข้าของกราฟที่แสดงพฤติกรรมของ causal delay เป็นสัญญาณเดียวกัน และมีค่าการเปลี่ยนแปลงสัญญาณเหมือนกัน ทำให้กราฟสายงานใหม่ที่ได้ที่สถานะ s0 s0 จะเกิดการเปลี่ยนแปลงของสัญญาณ Ai ไปยังสถานะ s2 s2 และ P2 ไปยังสถานะ s1 s2 โดยที่ค่าของการเปลี่ยนแปลงสัญญาณคือ 1 และจะทำการแทรกโนดที่ได้ลงในกราฟสายงานใหม่ และทำการเพิ่มสถานะใหม่ที่ได้ลงใน queue เพื่อที่จะทำการวิเคราะห์สถานะในขั้นตอนต่อไป ดังแสดงในรูปที่5.8 (b)

วิเคราะห์สถานะ s2 s2 ที่สถานะ s2 ของกราฟที่แสดงพฤติกรรมของเกตเกิดการเปลี่ยนแปลงของสัญญาณคือ Ai, P1’ ไปยังสถานะ s0 กับ P2 ไปยังสถานะ s3 และที่สถานะ s2

ของกราฟที่แสดงพฤติกรรมของ causal delay เกิดการเปลี่ยนแปลงของสัญญาณคือ P1 ไปยังสถานะ s3 จากการวิเคราะห์จะได้ว่า สำหรับการเปลี่ยนแปลงสัญญาณ Ai, P1’ จะไม่เกิดขึ้นเพราะไม่มีการเปลี่ยนแปลงสัญญาณ P1’ ในกราฟที่แสดงพฤติกรรมของ causal delay สำหรับการเปลี่ยนแปลงสัญญาณ P2 เป็นการเปลี่ยนแปลงสัญญาณขาเข้าของกราฟที่แสดงพฤติกรรมของเกตทำให้เกิดการเปลี่ยนแปลงสัญญาณ P2 ไปยังสถานะ s3 s2 สำหรับการเปลี่ยนแปลงสัญญาณ P1 เป็นการเปลี่ยนแปลงสัญญาณขาออกของกราฟที่แสดงพฤติกรรมของ causal delay ทำให้เกิดการเปลี่ยนแปลงสัญญาณ P1 ไปยังสถานะ s2 s3 ซึ่งกราฟสายงานใหม่ที่ได้ที่สถานะ s2 s2 จะเกิดการเปลี่ยนแปลงของสัญญาณ P2 ไปยังสถานะ s3 s2 และ P1 ไปยังสถานะ s2 s3 โดยที่ค่าของการเปลี่ยนแปลงสัญญาณคือ 1และจะทำการแทรกโนดที่ได้ลงในกราฟสายงานใหม่ และทำการเพิ่มสถานะใหม่ที่ได้ลงใน queueเพื่อที่จะทำการวิเคราะห์สถานะในขั้นตอนต่อไป ดังแสดงในรูปที่ 5.8 (c)

สำหรับการวิเคราะห์การเปลี่ยนแปลงสัญญาณอื่นๆ ก็จะทำเช่นเดียวกันกับที่ยกตัวอย่าง และเมื่อเราทำการวิเคราะห์ครบทุกสถานะที่เกิดขึ้นแล้ว จะได้กราฟสายงานที่แสดงพฤติกรรมของเกตที่เพิ่ม causal delay ให้กับสัญญาณขาออก ดังแสดงในรูปที่ 5.8 (d)

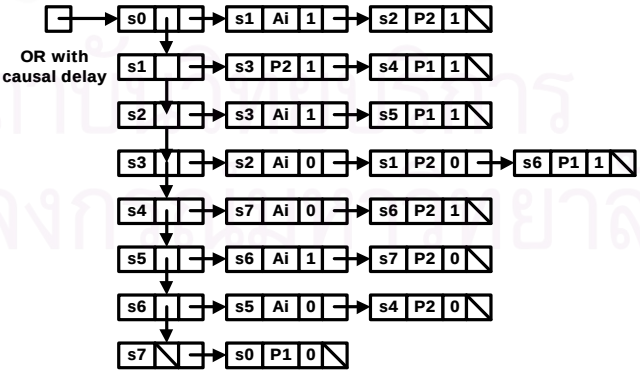


รูปที่ 5.8 ขั้นตอนวิธีการสร้างกราฟสายงานแสดงพฤติกรรมของเกตที่เพิ่ม causal delay ให้กับสัญญาณขาออก P1 ของเกต OR ในวงจร C-element

เนื่องจากกราฟสายงานที่ได้นี้จะนำไปใช้ในการสร้างเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์เพื่อจะใช้ในการทวนสอบต่อไป เราจะทำการเปลี่ยนแปลงค่าของสถานะ ซึ่งเกิดจากสถานะของกราฟที่แสดงพฤติกรรมของเกต กับสถานะของกราฟที่แสดงพฤติกรรมของ causal delay โดยการตั้งชื่อขึ้นใหม่เพื่อที่จะสะดวกในการอ้างอิง โดยจะสร้างตารางเทียบค่าของค่าของสถานะเดิม กับสถานะใหม่ และทำการแก้ไขค่าของสถานะตามตารางดังกล่าว ดังแสดงในตารางที่ 5.2 จะได้กราฟสายงานแสดงพฤติกรรมของเกตที่เพิ่ม causal delay ให้กับสัญญาณขาออก ดังแสดงในรูปที่ 5.9

ตารางที่ 5.2 ตารางเทียบค่าระหว่าง new state กับ old state สำหรับกราฟสายงานของเกต OR ที่เพิ่ม causal delay แล้ว

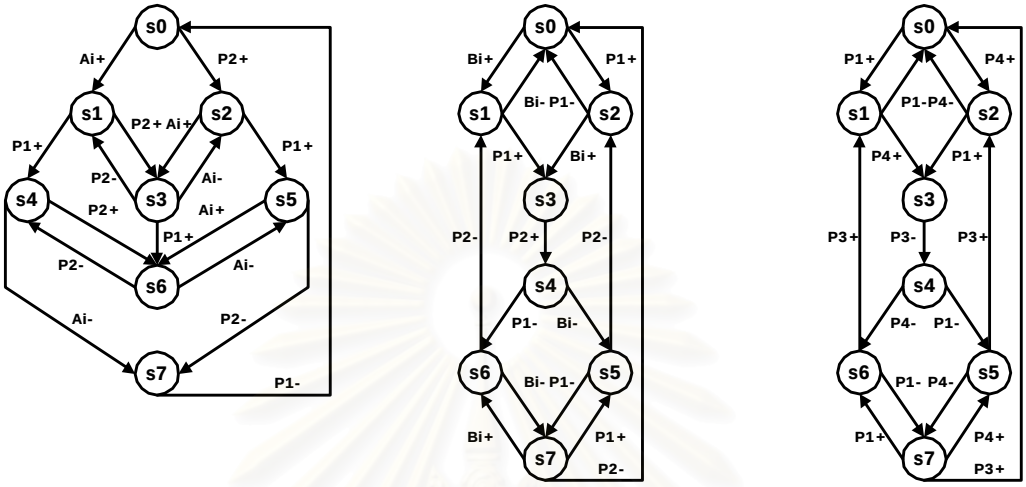
New State	Old State
S0	s0 s0
S1	s2 s2
S2	s1 s2
S3	s3 s2
S4	s2 s3
S5	s1 s3
S6	s3 s3
S7	s0 s1



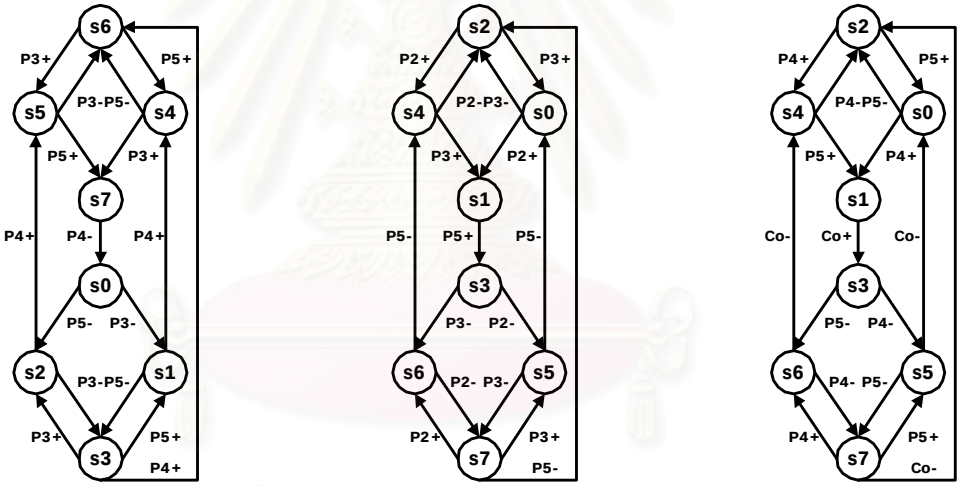
รูปที่ 5.9 กราฟแสดงพฤติกรรมของเกต OR ที่เพิ่ม causal delay

สำหรับกราฟสายงานที่แสดงพฤติกรรมของเกตที่เพิ่ม causal delay แล้วทั้งหมดในวงจร C-element ดังแสดงในรูปที่ 5.10 เรียงตามลำดับของเกตในอินพุตไฟล์โดยอ้างอิงจากรูปที่ 5.5 (b)

เกตที่ 1 or2(Ai, P2, P1, s0) เกตที่ 2 and2(Bi, P1, P2, s0) เกตที่ 3 nand2(P1, P4, P3, s0)



เกตที่ 4 nand2(P3, P5, P4, s3) เกตที่ 5 nand2(P2, P3, P5, s1) เกตที่ 6 and2(P4, P5, Co, s1)



รูปที่ 5.10 กราฟสายงานแสดงพฤติกรรมของเกตที่เพิ่ม causal delay แล้วทั้งหมดในวงจร C-element

Step 1.4 เป็นขั้นตอนวิธีการสร้างเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์หลังจากที่ได้ทำการเพิ่ม causal delay ให้กับสัญญาณขาออกของกราฟที่แสดงพฤติกรรมของเกตทุกตัวในวงจรเรียบร้อยแล้ว เราจะนำกราฟที่ได้มารวมกัน และทำการวิเคราะห์การเปลี่ยนแปลงสัญญาณแยกตามประเภทของสัญญาณ ได้แก่สัญญาณขาเข้า สัญญาณขาออก สัญญาณภายใน และสัญญาณที่ไม่ต้องการให้แสดง โดยคำนึงถึงโมเดลสิ่งแวดล้อมตามคุณลักษณะของวงจร เพื่อที่จะทำการหาพฤติกรรมของการเปลี่ยนแปลงสัญญาณจากสถานะหนึ่งไปยังสถานะหนึ่ง และนำมาสร้างเป็นกราฟสายงานที่แสดงพฤติกรรมของวงจร

โครงสร้างข้อมูลของกราฟสายงานที่แสดงพฤติกรรมของวงจร จะใช้โครงสร้างข้อมูลแบบเดียวกับโครงสร้างข้อมูลของกราฟสายงานที่แสดงพฤติกรรมของ causal delay (step 2.1)

Algorithm 4 Construct Finite State Machine for Circuit Implementation

Step 4.1 Construct environment graph of circuit specification

Step 4.2 Create and insert start state in flow graph

Step 4.3 AddQueue(queue, start_state)

Step 4.4 While (queue is Not Empty)

```
BeginWhile
    tmp_state = DeQueue(queue)
    For each input signal
    BeginFor
        Analyse input signal transition
        new_state = input signal transition from tmp_state
        If (have delete signal)
            new_state = step_look_forward(new_state)
        new_node = Create_Node(input signal, siganl value, new_state)
        Insert new_node into flow graph
        AddQueue(queue, new_state)
    EndFor
    For each output signal
    BeginFor
        Analyse output signal transition
        new_state = output signal transition from tmp_state
        If (have delete signal)
            new_state = step_look_forward(new_state)
        new_node = Create_Node(output signal, siganl value, new_state)
        Insert new_node into flow graph
        AddQueue(queue, new_state)
    EndFor
    For each internal signal
    BeginFor
```

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

```
                    Analyse internal signal transition
                    new_state = internal signal transition from tmp_state
                    If (have delete signal)
                        new_state = step_look_forward(new_state)
                    new_node = Create_Node(internal signal, siganl value, new_state)
                    Insert new_node into flow graph
                    AddQueue(queue, new_state)
                EndFor
            EndWhile
End
```

จากขั้นตอนวิธีที่ 4 (algorithm 4) เราสามารถอธิบายรายละเอียดได้ดังนี้

Step 4.1 เป็นขั้นตอนวิธีในการสร้างกราฟสายงานที่แสดงความสัมพันธ์ของการเปลี่ยนแปลงสัญญาณขาเข้า สัญญาณขาออก และสัญญาณที่เกี่ยวข้อง ตามโมเดลสิ่งแวดล้อมของคุณลักษณะของวงจร ซึ่งสามารถพิจารณาได้จากเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจรที่ได้จากบทที่ 4 เพื่อที่จะนำมาใช้ในการกำหนดจังหวะในการเปลี่ยนแปลงสัญญาณขาเข้าของการสร้างกราฟสายงานที่แสดงพฤติกรรมของวงจรที่ได้จากการสังเคราะห์ ดังแสดงในรูปที่ 5.4

Step 4.2 เป็นขั้นตอนวิธีในการสร้างสถานะเริ่มต้น และใช้เป็นโนดเริ่มต้นในการแทรกเข้าสู่กราฟสายงาน สถานะเริ่มต้นสำหรับกราฟสายงานของเครื่องจักรสถานะจำกัด เกิดจากการนำเอาสถานะเริ่มต้นของกราฟสายงานที่แสดงพฤติกรรมของเกตที่เพิ่ม causal delay แล้วทุกกราฟสายงานกับกราฟโมเดลสิ่งแวดล้อมมาต่อกันตามลำดับ ดังแสดงในรูปที่ 5.11 (a)

Step 4.3 เป็นขั้นตอนวิธีในการเพิ่มสถานะเริ่มต้นที่ได้มาจาก Step 4.2 ลงใน queue

Step 4.4 เป็นขั้นตอนวิธีในการวิเคราะห์การเปลี่ยนแปลงสัญญาณในกราฟที่แสดงพฤติกรรมของเกตที่เพิ่ม causal delay แล้วทุกกราฟสายงาน และกราฟโมเดลสิ่งแวดล้อม โดยเริ่มพิจารณาจากสถานะเริ่มต้น (step 4.2) ซึ่งค่าของสถานะจะประกอบไปด้วยค่าสถานะของแต่ละกราฟสายงานที่แสดงพฤติกรรมของเกต โดยมีลำดับตามเกตที่กำหนดไว้ในอินพุตไฟล์ และค่าสถานะของโมเดลสิ่งแวดล้อมมาเรียงต่อกัน สามารถพิจารณาได้ดังนี้

สำหรับการเปลี่ยนแปลงของสัญญาณขาเข้า ณ สถานะที่ทำการพิจารณา ค่าสถานะในส่วนของกราฟสายงานโมเดลสิ่งแวดล้อมมีการเปลี่ยนแปลงของสัญญาณขาเข้า และค่าสถานะในส่วนของกราฟที่แสดงพฤติกรรมของเกตที่มีการเปลี่ยนแปลงสัญญาณที่ทำการพิจารณา เป็นสัญญาณขาเข้าของเกตนั้น โดยที่การเปลี่ยนแปลงสัญญาณทั้งหมดที่เกิดขึ้นตามที่กล่าวมามีค่าของการเปลี่ยนแปลงสัญญาณเท่ากัน จะเกิดการเปลี่ยนแปลงของสัญญาณขาเข้าในกราฟสายงานสำหรับ

เครื่องจักรสถานะจำกัด จากสถานะที่ทำการพิจารณาไปยังสถานะใหม่ โดยที่ค่าสถานะใหม่เกิดจากค่าสถานะที่ทำการพิจารณา ซึ่งค่าสถานะในแต่ละส่วนที่เกิดการเปลี่ยนแปลงของสัญญาณขาเข้าดังกล่าวจะเกิดการเปลี่ยนแปลงค่าสถานะตามกราฟสายงานของแต่ละส่วน ในส่วนค่าสถานะที่ไม่เกิดการเปลี่ยนแปลงของสัญญาณขาเข้าดังกล่าวจะยังคงค่าสถานะเดิม สำหรับในกรณีที่มีสัญญาณที่ไม่ต้องการให้แสดงในกราฟ เราจะทำการพิจารณาสถานะที่ได้ก่อนที่จะนำไปเพิ่มในกราฟสายงาน โดยเราจะหาการเปลี่ยนแปลงของสัญญาณที่ไม่ต้องการให้แสดงในกราฟจากสถานะที่ได้ไปยังสถานะใหม่จนกว่าสถานะใหม่จะไม่เกิดการเปลี่ยนแปลงของสัญญาณที่ไม่ต้องการให้แสดงในกราฟ และจะนำสถานะนั้นไปเพิ่มในกราฟสายงาน

สำหรับการเปลี่ยนแปลงของสัญญาณขาออก ณ สถานะที่ทำการพิจารณา ค่าสถานะในส่วนของกราฟสายงานโมเดลสิ่งแวดล้อมมีการเปลี่ยนแปลงของสัญญาณขาออก และค่าสถานะในส่วนของกราฟที่แสดงพฤติกรรมของเกตที่มีการเปลี่ยนแปลงสัญญาณที่ทำการพิจารณา เป็นสัญญาณขาออกของเกตนั้น โดยที่การเปลี่ยนแปลงสัญญาณทั้งหมดที่เกิดขึ้นตามที่กล่าวมามีค่าของการเปลี่ยนแปลงสัญญาณเท่ากัน จะเกิดการเปลี่ยนแปลงของสัญญาณขาออกในกราฟสายงานสำหรับเครื่องจักรสถานะจำกัด จากสถานะที่ทำการพิจารณาไปยังสถานะใหม่ โดยที่ค่าสถานะใหม่เกิดจากค่าสถานะที่ทำการพิจารณา ซึ่งค่าสถานะในแต่ละส่วนที่เกิดการเปลี่ยนแปลของสัญญาณขาออกดังกล่าวจะเกิดการเปลี่ยนแปลงค่าสถานะตามกราฟสายงานของแต่ละส่วน ในส่วนค่าสถานะที่ไม่เกิดการเปลี่ยนแปลงของสัญญาณขาออกดังกล่าวจะยังคงสถานะเดิม สำหรับในกรณีที่มีสัญญาณที่ไม่ต้องการให้แสดงในกราฟ เราจะทำการพิจารณาสถานะที่ได้ก่อนที่จะนำไปเพิ่มในกราฟสายงาน โดยเราจะหาการเปลี่ยนแปลงของสัญญาณที่ไม่ต้องการให้แสดงในกราฟจากสถานะที่ได้ไปยังสถานะใหม่จนกว่าสถานะใหม่จะไม่เกิดการเปลี่ยนแปลงของสัญญาณที่ไม่ต้องการให้แสดงในกราฟ และจะนำสถานะนั้นไปเพิ่มในกราฟสายงาน

สำหรับการเปลี่ยนแปลงของสัญญาณภายใน ณ สถานะที่ทำการพิจารณา ค่าสถานะในส่วนของกราฟที่แสดงพฤติกรรมของเกตมีการเปลี่ยนแปลงของสัญญาณภายใน โดยที่การเปลี่ยนแปลงสัญญาณทั้งหมดที่เกิดขึ้นตามที่กล่าวมามีค่าของการเปลี่ยนแปลงสัญญาณเท่ากับค่าของการเปลี่ยนแปลงสัญญาณของกราฟที่แสดงพฤติกรรมของเกตซึ่งมีการเปลี่ยนแปลงสัญญาณนั้นเป็นสัญญาณขาออก จะเกิดการเปลี่ยนแปลงของสัญญาณภายในในกราฟสายงานสำหรับเครื่องจักรสถานะจำกัด จากสถานะที่ทำการพิจารณาไปยังสถานะใหม่ โดยที่ค่าสถานะใหม่เกิดจากค่าสถานะที่ทำการพิจารณา ซึ่งค่าสถานะในแต่ละส่วนที่เกิดการเปลี่ยนแปลงของสัญญาณภายในดังกล่าวจะเกิดการเปลี่ยนแปลงค่าสถานะตามกราฟสายงานของแต่ละส่วน ในส่วนค่าสถานะที่ไม่เกิดการเปลี่ยนแปลงของสัญญาณภายในดังกล่าวจะยังคงสถานะเดิม สำหรับในกรณีที่มีสัญญาณที่ไม่ต้องการให้แสดงในกราฟ เราจะทำการพิจารณาสถานะที่ได้ก่อนที่จะนำไปเพิ่มในกราฟสายงาน โดยเราจะหาการเปลี่ยนแปลงของสัญญาณที่ไม่ต้องการให้แสดงในกราฟจากสถานะที่ได้ไปยัง

สถานะใหม่จนกว่าสถานะใหม่จะไม่เกิดการเปลี่ยนแปลงของสัญญาณที่ไม่ต้องการให้แสดงในกราฟ และจะนำสถานะนั้นไปเพิ่มในกราฟสายงาน

ตัวอย่างต่อไปนี้จะแสดงขั้นตอนวิธีในการสร้างกราฟสายงานสำหรับเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ของวงจร C-element ซึ่งมี Ai, Bi เป็นสัญญาณขาเข้า Co เป็นสัญญาณขาออก และ P1, P2, P3, P4, P5 เป็นสัญญาณภายใน โดยที่ s0 s0 s0 s0 s0 s0 s0 เป็นสถานะเริ่มต้นของกราฟสายงานที่จะทำการสร้าง ซึ่งค่าของสถานะในแต่ละส่วนยกเว้นส่วนสุดท้าย คือสถานะเริ่มต้นของกราฟที่แสดงพฤติกรรมของแต่ละเกต ซึ่งเรียงลำดับตามเกตในอินพุตไฟล์ และค่าของสถานะในส่วนสุดท้าย คือสถานะเริ่มต้นของกราฟสายงานโมเดลสิ่งแวดล้อม (step 4.1) สำหรับกราฟแสดงพฤติกรรมของเกตจะอ้างอิงกับรูปที่ 5.10 และกราฟสายงานโมเดลสิ่งแวดล้อมจะอ้างอิงกับรูปที่ 5.4

วิเคราะห์สถานะ s0 s0 s0 s0 s0 s0 s0 สำหรับสัญญาณขาเข้าพิจารณาค่าสถานะในส่วนสุดท้ายซึ่งเป็นค่าสถานะของกราฟสายงานโมเดลสิ่งแวดล้อมมีสถานะเป็น s0 จะเกิดการเปลี่ยนแปลงของสัญญาณขาเข้านั่นคือสัญญาณ Ai จากสถานะ s0 ไปยังสถานะ s1 และ Bi จากสถานะ s0 ไปยังสถานะ s2 ให้ทำการพิจารณาทีละการเปลี่ยนแปลงของสัญญาณ

พิจารณาการเปลี่ยนแปลงสัญญาณ Ai เกตที่มีสัญญาณ Ai เป็นสัญญาณขาเข้าคือเกตที่ 1 และค่าสถานะในส่วนของเกตที่ 1 เป็นสถานะ s0 ซึ่งกราฟที่แสดงพฤติกรรมของเกตที่ 1 เกิดการเปลี่ยนแปลงของสัญญาณ Ai จากสถานะ s0 ไปยังสถานะ s1 และค่าของการเปลี่ยนแปลงสัญญาณที่เกิดขึ้นเหมือนกับค่าของการเปลี่ยนแปลงสัญญาณในกราฟสายงานโมเดลสิ่งแวดล้อม จะเกิดการเปลี่ยนแปลงของสัญญาณ Ai จากสถานะ s0 s0 s0 s0 s0 s0 s0 ไปยังสถานะ s1 s0 s0 s0 s0 s0 s1 โดยที่ค่าของการเปลี่ยนแปลงสัญญาณคือ 1 และจะทำการแทรกโนดที่ได้ลงในกราฟสายงานสำหรับเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ และเพิ่มสถานะใหม่ที่ได้ลงในqueue เพื่อใช้ในการวิเคราะห์สัญญาณต่อไป ดังแสดงในรูปที่ 5.11 (b)

พิจารณาการเปลี่ยนแปลงสัญญาณ Bi เกตที่มีสัญญาณ Bi เป็นสัญญาณขาเข้าคือเกตที่ 2 และค่าสถานะในส่วนของเกตที่ 2 เป็นสถานะ s0 ซึ่งกราฟที่แสดงพฤติกรรมของเกตที่ 2 เกิดการเปลี่ยนแปลงของสัญญาณ Bi จากสถานะ s0 ไปยังสถานะ s1 และค่าของการเปลี่ยนแปลงสัญญาณที่เกิดขึ้นเหมือนกับค่าของการเปลี่ยนแปลงสัญญาณในกราฟสายงานโมเดลสิ่งแวดล้อม จะเกิดการเปลี่ยนแปลงของสัญญาณ Bi จากสถานะ s0 s0 s0 s0 s0 s0 s0 ไปยังสถานะ s0 s1 s0 s0 s0 s0 s2 โดยที่ค่าของการเปลี่ยนแปลงสัญญาณคือ 1 และจะทำการแทรกโนดที่ได้ลงในกราฟสายงานสำหรับเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ และเพิ่มสถานะใหม่ที่ได้ลงในqueue เพื่อใช้ในการวิเคราะห์สัญญาณต่อไป ดังแสดงในรูปที่ 5.11 (c)

สำหรับสัญญาณขาออก สัญญาณภายใน และสัญญาณที่ไม่ต้องการให้แสดงในกราฟสายงาน จะไม่เกิดการเปลี่ยนแปลงของสัญญาณจากสถานะที่ทำการพิจารณานี้เนื่องจากไม่เป็นไปตามเงื่อนไขที่กล่าวมา

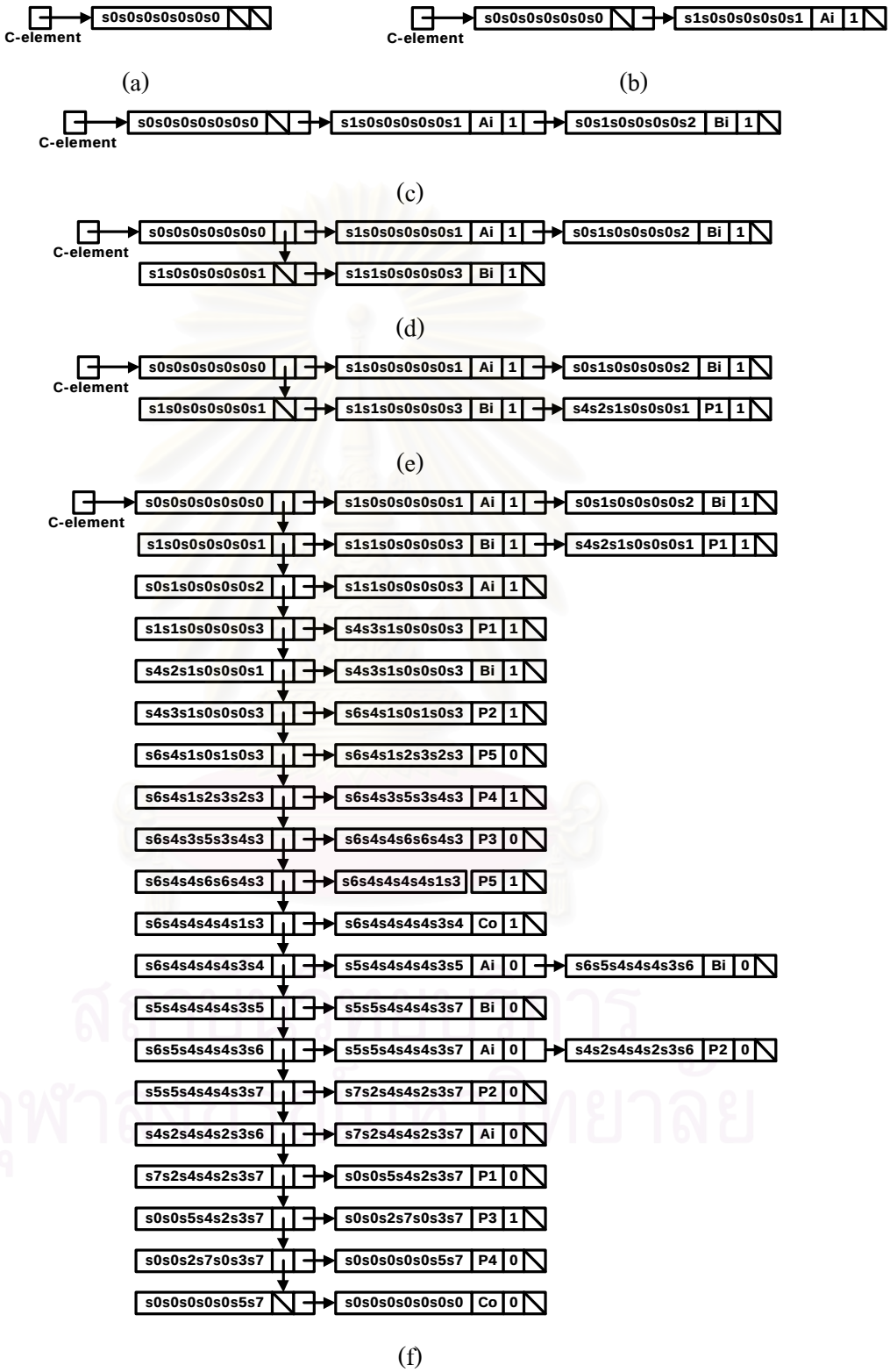
วิเคราะห์สถานะ s1 s0 s0 s0 s0 s0 s1 สำหรับสัญญาณขาเข้า พิจารณาค่าสถานะในส่วนสุดท้ายซึ่งเป็นค่าสถานะของกราฟสายงานโมเดลสิ่งแวดล้อมมีสถานะเป็น s1 จะเกิดการเปลี่ยนแปลงของสัญญาณขาเข้านั่นคือสัญญาณ Bi จากสถานะ s1 ไปยังสถานะ s3

พิจารณาการเปลี่ยนแปลงสัญญาณ Bi เกตที่มีสัญญาณ Bi เป็นสัญญาณขาเข้าคือเกตที่ 2 และค่าสถานะในส่วนของเกตที่ 2 เป็นสถานะ s0 ซึ่งกราฟที่แสดงพฤติกรรมของเกตที่ 2 เกิดการเปลี่ยนแปลงของสัญญาณ Bi จากสถานะ s0 ไปยังสถานะ s1 และค่าของการเปลี่ยนแปลงสัญญาณที่เกิดขึ้นเหมือนกับค่าของการเปลี่ยนแปลงสัญญาณในกราฟสายงานโมเดลสิ่งแวดล้อม จะเกิดการเปลี่ยนแปลงของสัญญาณ Bi จากสถานะ s1 s0 s0 s0 s0 s0 s1 ไปยังสถานะ s1 s1 s0 s0 s0 s0 s3 โดยที่ค่าของการเปลี่ยนแปลงสัญญาณคือ 1 และจะทำการแทรกโนดที่ได้ลงในกราฟสายงานสำหรับเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ และเพิ่มสถานะใหม่ที่ได้ลงใน queue เพื่อใช้ในการวิเคราะห์สัญญาณต่อไป ดังแสดงในรูปที่ 5.11 (d)

สำหรับสัญญาณภายใน พิจารณาสัญญาณ P1 เกตที่มีสัญญาณ P1 เป็นสัญญาณขาออกคือเกตที่ 1 และค่าสถานะในส่วนของเกตที่ 1 เป็นสถานะ s1 ซึ่งกราฟที่แสดงพฤติกรรมของเกตที่ 1 เกิดการเปลี่ยนแปลงของสัญญาณ P1 จากสถานะ s1 ไปยังสถานะ s4 ค่าของการเปลี่ยนแปลงสัญญาณคือ 1 จะเกิดการเปลี่ยนแปลงสัญญาณ P1 นี้ได้ก็ต่อเมื่อกราฟที่แสดงพฤติกรรมของเกตที่มีสัญญาณ P1 เกี่ยวข้องทั้งหมดจากสถานะที่ทำการพิจารณาเกิดการเปลี่ยนแปลงสัญญาณ P1 และค่าของการเปลี่ยนแปลงสัญญาณต้องมีค่าเหมือนกับค่าของการเปลี่ยนแปลงสัญญาณ P1 ของกราฟที่แสดงพฤติกรรมของเกตที่ 1 นั่นคือมีค่าเป็น 1 ซึ่งเกตที่มีสัญญาณ P1 เกี่ยวข้องคือเกตที่ 2 และเกตที่ 3 พิจารณาค่าสถานะในส่วนของเกตที่ 2 ซึ่งเป็นสถานะ s0 จะเกิดการเปลี่ยนแปลงของสัญญาณ P1 จากสถานะ s0 ไปยังสถานะ s2 และค่าของการเปลี่ยนแปลงสัญญาณคือ 1 พิจารณาค่าสถานะในส่วนของเกตที่ 3 ซึ่งเป็นสถานะ s0 จะเกิดการเปลี่ยนแปลงของสัญญาณ P1 จากสถานะ s0 ไปยังสถานะ s1 และค่าของการเปลี่ยนแปลงสัญญาณคือ 1 ทำให้เกิดการเปลี่ยนแปลงของสัญญาณ P1 จากสถานะ s1 s0 s0 s0 s0 s0 s1 ไปยังสถานะ s4 s2 s1 s0 s0 s0 s1 โดยที่ค่าของการเปลี่ยนแปลงสัญญาณคือ 1 และจะทำการแทรกโนดที่ได้ลงในกราฟสายงานสำหรับเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ และเพิ่มสถานะใหม่ที่ได้ลงใน queue เพื่อใช้ในการวิเคราะห์สัญญาณต่อไป ดังแสดงในรูปที่ 5.11 (e)

สำหรับสัญญาณขาออก สัญญาณภายในที่เหลือ และสัญญาณที่ไม่ต้องการให้แสดงในกราฟสายงาน จะไม่เกิดการเปลี่ยนแปลงของสัญญาณจากสถานะที่ทำการพิจารณานี้เนื่องจากไม่เป็นไปตามเงื่อนไขที่กล่าวมา

สำหรับการวิเคราะห์การเปลี่ยนแปลงสัญญาณอื่นๆ ก็จะทำเช่นเดียวกันกับที่ยกตัวอย่าง และเมื่อเราทำการวิเคราะห์ครบทุกสถานะที่เกิดขึ้นแล้ว จะได้กราฟสายงานสำหรับเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ ดังแสดงในรูปที่ 5.11 (f)

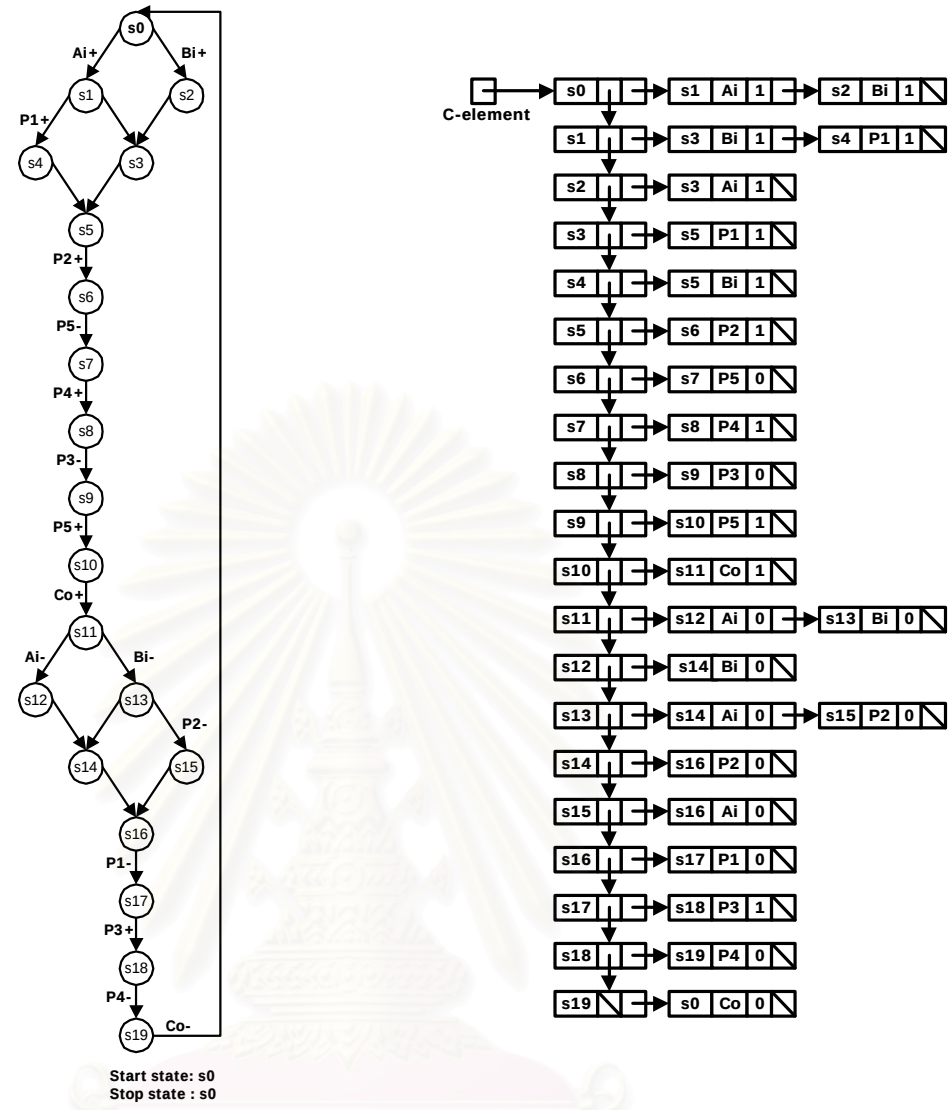


รูปที่ 5.11 ขั้นตอนวิธีการสร้างเครื่องจักรสถานะจำกัดของวงจร C-element ที่ได้จากการสังเคราะห์

เนื่องจากกราฟสายงานที่ได้นี้จะนำไปใช้เป็นเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ เพื่อจะใช้ในการทวนสอบวงจรต่อไป เราจะทำการเปลี่ยนแปลงค่าของสถานะซึ่งเกิดจากสถานะของกราฟที่แสดงพฤติกรรมของเกต โดยการตั้งชื่อขึ้นใหม่เพื่อที่จะสะดวกในการอ้างอิง โดยเราจะสร้างตารางเทียบค่าของสถานะเดิม กับค่าของสถานะใหม่ และทำการเปลี่ยนแปลงค่าของสถานะตามตารางดังกล่าว ดังแสดงในตารางที่ 5.3 จะได้กราฟสายงานสำหรับเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ ดังแสดงในรูปที่ 5.12

ตารางที่ 5.3 ตารางเทียบค่าระหว่าง new state กับ old state สำหรับเครื่องจักรสถานะจำกัดของวงจร C-element ที่ได้จากการสังเคราะห์

New State	Old State
S0	s0 s0 s0 s0 s0 s0 s0
S1	s1 s0 s0 s0 s0 s0 s1
S2	s0 s1 s0 s0 s0 s0 s2
S3	s1 s1 s0 s0 s0 s0 s3
S4	s4 s2 s1 s0 s0 s0 s1
S5	s4 s3 s1 s0 s0 s0 s3
S6	s6 s4 s1 s0 s1 s0 s3
S7	s6 s4 s1 s2 s3 s2 s3
S8	s6 s4 s3 s5 s3 s4 s3
S9	s6 s4 s4 s6 s6 s4 s3
S10	s6 s4 s4 s4 s4 s1 s3
S11	s6 s4 s4 s4 s4 s3 s4
S12	s5 s4 s4 s4 s4 s3 s5
S13	s6 s5 s4 s4 s4 s3 s6
S14	s5 s5 s4 s4 s4 s3 s7
S15	s4 s2 s4 s4 s2 s3 s6
S16	s7 s2 s4 s4 s2 s3 s7
S17	s0 s0 s5 s4 s2 s3 s7
S18	s0 s0 s2 s7 s0 s3 s7
S19	s0 s0 s0 s0 s0 s5 s7



รูปที่ 5.12 เครื่องจักรสถานะจำกัดของวงจร C-element ที่ได้จากการสังเคราะห์ในรูปแบบของกราฟ (a) และโครงสร้างของข้อมูล (b)

ขั้นตอนวิธีการสร้างเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ ที่กล่าวมาจะมีการรับอินพุตเป็นนิพจน์ของเกตที่เรียงต่อกันเป็นวงจร, สร้างกราฟพฤติกรรมของเกตเหล่านั้นและนำมารวมกันโดยใช้ตัวกระทำการ concurrent composition ซึ่งสามารถวิเคราะห์ความซับซ้อนของขั้นตอนวิธีได้ดังนี้ โดยให้จำนวนครั้งของการเปลี่ยนแปลงสัญญาณเป็นเวลา

สำหรับวงจรที่มี เกต g เกต และกราฟพฤติกรรมของเกตมีจำนวนของการเปลี่ยนแปลงสัญญาณเท่ากับ n ครั้ง

กรณีที่ใช้เวลาน้อยที่สุด จะต้องมีเกตอย่างน้อย 1 เกต ดังนั้นเวลาที่ดีที่สุดของขั้นตอนนี้คือ n นั่นคือ O(n)

กรณีที่ใช้เวลามากที่สุด จะต้องมีเกตอย่างมากที่สุด g เกต เวลาที่มากที่สุดคือ g*n และในกรณีที่มี delete signal เราจะต้องทำขั้นตอน step_look_forward อย่างมากที่สุดคือ g*n ดังนั้นเวลาที่มากที่สุดของขั้นตอนนี้คือ $(g*n)^2$ นั่นคือ $O((gn)^2)$

5.3 สรุป

ในบทนี้ได้กล่าวถึงขั้นตอนวิธีการสร้างเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์โดยใช้เทคนิคของพีชคณิตเชิงกระบวนการ โดยเริ่มจากการนำพฤติกรรมของเกตต่างๆ มาสร้างเป็นกราฟที่แสดงพฤติกรรมของเกต แล้วนำมาเพิ่ม causal delay เพื่อให้เหมาะสมกับวงจรแบบอสมวาร จากนั้นจะนำกราฟทั้งหมดที่ได้มารวมกันโดยใช้เทคนิค concurrent composition ของพีชคณิตเชิงกระบวนการ และยังคำนึงถึงโมเดลสิ่งแวดล้อมของคุณลักษณะวงจรเป็นหลัก สำหรับเวลาที่ใช้ในการทำงานของขั้นตอนวิธีที่ได้นำเสนอคือ O(n) ในกรณีที่ใช้เวลาน้อยที่สุด และ $O((gn)^2)$ ในกรณีที่ใช้เวลามากที่สุด โดยที่ g คือจำนวนเกตในวงจรม n คือจำนวนของการเปลี่ยนแปลงสัญญาณในเกต

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 6
การทวนสอบวงจร

ในบทนี้จะกล่าวถึงวิธีการทวนสอบวงจรอย่างมีแบบแผนโดยใช้เครื่องจักรสถานะจำกัด ซึ่งการทวนสอบโดยวิธีนี้ทั้งพฤติกรรมของคุณลักษณะของวงจร กับพฤติกรรมของวงจรที่ได้จากการสังเคราะห์จะถูกแสดงในรูปของเครื่องจักรสถานะจำกัด จากนั้นจะทำการตรวจสอบความสัมพันธ์กันระหว่างเครื่องจักรสถานะจำกัดโดยการค้นหาภายใน state space ของเครื่องจักรสถานะจำกัดว่าพฤติกรรมของวงจรที่ได้จากการสังเคราะห์มีพฤติกรรมตรงตามคุณลักษณะของวงจรที่ได้ออกแบบไว้

นอกจากนี้ยังได้แสดงตัวอย่างการประยุกต์ใช้ขั้นตอนวิธีการทวนสอบวงจรกับวงจร C-element และวงจรที่ไม่ถูกต้องตรงตามคุณลักษณะที่ได้ออกแบบไว้ และแสดงผลการทดลองเมื่อนำขั้นตอนวิธีการทวนสอบที่ได้นำเสนอในงานวิจัยฉบับนี้ไปทดลองกับวงจรเกณฑ์เปรียบเทียบสมรรถนะด้วย

6.1 การตรวจสอบความสัมพันธ์กันระหว่างเครื่องจักรสถานะจำกัด

สำหรับขั้นตอนวิธีการตรวจสอบความสัมพันธ์กันระหว่างเครื่องจักรสถานะจำกัดมีขั้นตอนดังต่อไปนี้

Algorithm 1	Check Relation Between 2 FSM
Input	Specfication FSM, Implementation FSM
Output	Verification result

Step 1.1	InitialQueue (spec_queue)
	InitialQueue (imp_queue)
Step 1.2	AddQueue (spec_queue, spec_startstate)
	AddQueue (imp_queue, imp_startstate)
Step 1.3	While (spec_queue is Not Empty)
	BeginWhile
	spec_state = DeQueue (spec_queue)
	imp_state = DeQueue (imp_queue)
	While (spec_state have Child)

```
                BeginWhile
                    If (imp_state have Child && imp_child = spec_child)
                        AddQueue(spec_queue, spec_child)
                        AddQueue(imp_queue, imp_child)
                    Else
                        Not Verified
                        End
                    EndIf
                    Next Child
                EndWhile
            EndWhile
            Verified
End
```

จากขั้นตอนวิธีที่ 1 (algorithm 1) เราสามารถอธิบายรายละเอียดได้ดังนี้

Step 1.1 เป็นขั้นตอนในการเตรียม queue สำหรับเก็บสถานะของเครื่องจักรสถานะจำกัด โดยจะแบ่งเป็น spec_queue สำหรับคุณลักษณะของวงจร และ imp_queue สำหรับวงจรที่ได้จากการสังเคราะห์เพื่อไว้ใช้ในการตรวจสอบความสัมพันธ์

Step 1.2 ในการทวนสอบเราจะเริ่มต้นพิจารณาจากสถานะเริ่มต้นของเครื่องจักรสถานะจำกัดเป็นอันดับแรก ดังนั้นในขั้นตอนนี้จะเป็นการเพิ่มสถานะเริ่มต้นของเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร (spec_startstate) ลงใน spec_queue และสถานะเริ่มต้นของเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ (imp_startstate) ลงใน imp_queue

Step 1.3 เป็นขั้นตอนในการเปรียบเทียบพฤติกรรมของวงจรที่ได้จากการสังเคราะห์กับคุณลักษณะของวงจร โดยจะเริ่มพิจารณาจากสถานะเริ่มต้นของเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร และวงจรที่ได้จากการสังเคราะห์ ซึ่งจะทำการแวะผ่านกราฟเครื่องจักรสถานะจำกัดแบบ BFS ในการพิจารณาพฤติกรรมจะทำการพิจารณาสถานะของวงจรที่ได้จากการสังเคราะห์ (imp_state) ทีละสถานะว่าการเปลี่ยนแปลงสัญญาณ และจำนวนของการเปลี่ยนแปลงสัญญาณที่เกิดขึ้นทั้งหมดเท่ากันกับการเปลี่ยนแปลงสัญญาณ และจำนวนของการเปลี่ยนแปลงสัญญาณที่เกิดขึ้นทั้งหมดในสถานะของคุณลักษณะของวงจร (spec_state) ถ้าปรากฏว่ามีสถานะใดสถานะหนึ่งของวงจรที่ได้จากการสังเคราะห์มีการเปลี่ยนแปลงสัญญาณ และจำนวนของการเปลี่ยนแปลงสัญญาณไม่ถูกต้องตรงตามสถานะของคุณลักษณะของวงจร จะถือว่าวงจรที่ได้จากการสังเคราะห์มีพฤติกรรมไม่ถูกต้องตรงตามคุณลักษณะของวงจรที่ได้ออกแบบไว้ และวงจรที่ได้

จากการสังเคราะห์นั้นไม่ผ่านการทวนสอบ ถ้าไม่ปรากฏเหตุการณ์ดังกล่าวและได้ทำการแวะผ่านกราฟเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจรจนถึงสถานะปลายทางแล้ว จะถือว่าวงจรที่ได้จากการสังเคราะห์มีพฤติกรรมถูกต้องตรงตามคุณลักษณะของวงจรที่ได้ออกแบบไว้ และวงจรที่ได้จากการสังเคราะห์นั้นผ่านการทวนสอบ

ขั้นตอนวิธีในทวนสอบที่กล่าวมาจะมีการรับอินพุตเป็นเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร และวงจรที่ได้จากการสังเคราะห์ และจะทำการตรวจสอบความสัมพันธ์กันโดยใช้วิธีแวะผ่านกราฟแบบ breadth-first search ซึ่งสามารถวิเคราะห์ความซับซ้อนของขั้นตอนวิธีได้ดังนี้ โดยให้จำนวนของสถานะเป็นเวลา

สำหรับเครื่องจักรสถานะจำกัดที่มีจำนวนของสถานะเป็น n

กรณีที่ใช้เวลาน้อยที่สุด เครื่องจักรสถานะจำกัดทั้งสองมีการเปลี่ยนแปลงของสัญญาณไม่เหมือนกันตั้งแต่สถานะเริ่มต้น ดังนั้นเวลาที่ดีที่สุดของขั้นตอนนี้คือ 1 นั่นคือ O(1)

กรณีที่ใช้เวลามากที่สุด เครื่องจักรสถานะจำกัดทั้งสองมีพฤติกรรมการเปลี่ยนแปลงสัญญาณเหมือนกันทั้งหมด ดังนั้นเวลาที่มากที่สุดของขั้นตอนนี้คือ n นั่นคือ O(n)

6.2 ตัวอย่างการทวนสอบวงจร C-element

ในหัวข้อนี้จะแสดงตัวอย่างการทวนสอบวงจร C-element ตามขั้นตอนวิธีที่ได้นำเสนอไว้ข้างต้น โดยจะมีขั้นตอนวิธีในการทวนสอบดังนี้

1. สร้างเครื่องจักรสถานะจำกัดสำหรับคุณลักษณะของวงจร ดังแสดงในรูปที่ 6.1
2. สร้างเครื่องจักรสถานะจำกัดสำหรับวงจรที่ได้จากการสังเคราะห์ ดังแสดงในรูปที่ 6.2
3. ตรวจสอบความสัมพันธ์กันระหว่างเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจรกับวงจรที่ได้จากการสังเคราะห์

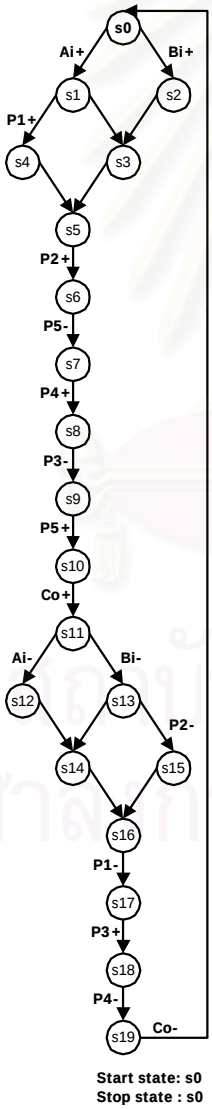
เนื่องจากขั้นตอนวิธี และตัวอย่างในการสร้างเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจร กับวงจรที่ได้จากการสังเคราะห์ ได้อธิบายไว้อย่างละเอียดในบทที่ 4 และ 5 ตามลำดับ ในหัวข้อนี้จึงแสดงตัวอย่างการตรวจสอบความสัมพันธ์กันระหว่างเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจรที่ได้ออกแบบไว้ กับวงจรที่ได้จากการสังเคราะห์เท่านั้น

ในการตรวจสอบความสัมพันธ์เราจะทำการแวะผ่านกราฟเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจรแบบ BFS ตัวอย่างจากรูปที่ 6.1 จะได้ลำดับของสถานะที่จะทำการพิจารณาดังต่อไปนี้ s0 s1 s2 s3 s4 s5 s6 s7 s8 s9 s10 s11 s12 s13 s14 s15 s16 s17 s18 s19 โดยที่สถานะเริ่มต้นและสถานะปลายทางของเครื่องจักรสถานะจำกัดทั้งสองคือ s0

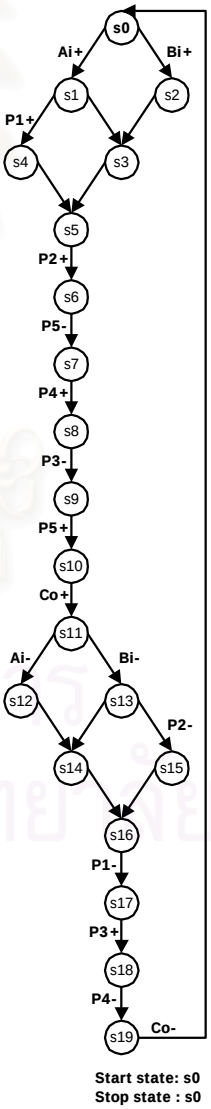
พิจารณาสถานะ s0 เครื่องจักรสถานะจำกัดของวงจรที่ได้จกาการสังเคราะห์ จะเกิดการเปลี่ยนแปลงสัญญาณ Ai จาก 0 ไปเป็น 1 และ Bi จาก 0 ไปเป็น 1 ซึ่งการเปลี่ยนแปลงสัญญาณ และจำนวนการเปลี่ยนแปลงสัญญาณที่เกิดขึ้นเป็นไปตามคุณลักษณะของวงจรที่ได้ออกแบบไว้

พิจารณาสถานะ s 1 เครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ จะเกิดการเปลี่ยนแปลงสัญญาณ Bi จาก 0 ไปเป็น 1 และ P1 จาก 0 ไปเป็น 1 ซึ่งการเปลี่ยนแปลงสัญญาณ และจำนวนการเปลี่ยนแปลงสัญญาณที่เกิดขึ้นเป็นไปตามคุณลักษณะของวงจรที่ได้ออกแบบไว้

สำหรับการพิจารณาสถานะอื่นๆ ก็จะทำเช่นเดียวกัน และเมื่อทำการพิจารณาครบทุกสถานะในเครื่องจักรสถานะจำกัดโดยที่ไม่พบการเปลี่ยนแปลงสัญญาณของเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์มีพฤติกรรมไม่ตรงตามคุณลักษณะของวงจรที่ได้ออกแบบไว้แล้ว จะถือว่าวงจรนั้นผ่านการทวนสอบ



รูปที่ 6.1 FSM Specification



รูปที่ 6.2 FSM Implementation

6.3 ตัวอย่างการทวนสอบวงจรที่ไม่ถูกต้องตรงตามคุณลักษณะที่ได้ออกแบบไว้

ในหัวข้อนี้จะแสดงตัวอย่างการทวนสอบวงจรที่ไม่ถูกต้องตรงตามคุณลักษณะที่ได้ออกแบบไว้ โดยจะใช้วงจร C-element ในการนำเสนอ

ความผิดพลาดที่สามารถเกิดขึ้นในการสังเคราะห์วงจรสามารถแบ่งออกได้เป็น 2 ประเภท

1. เกต

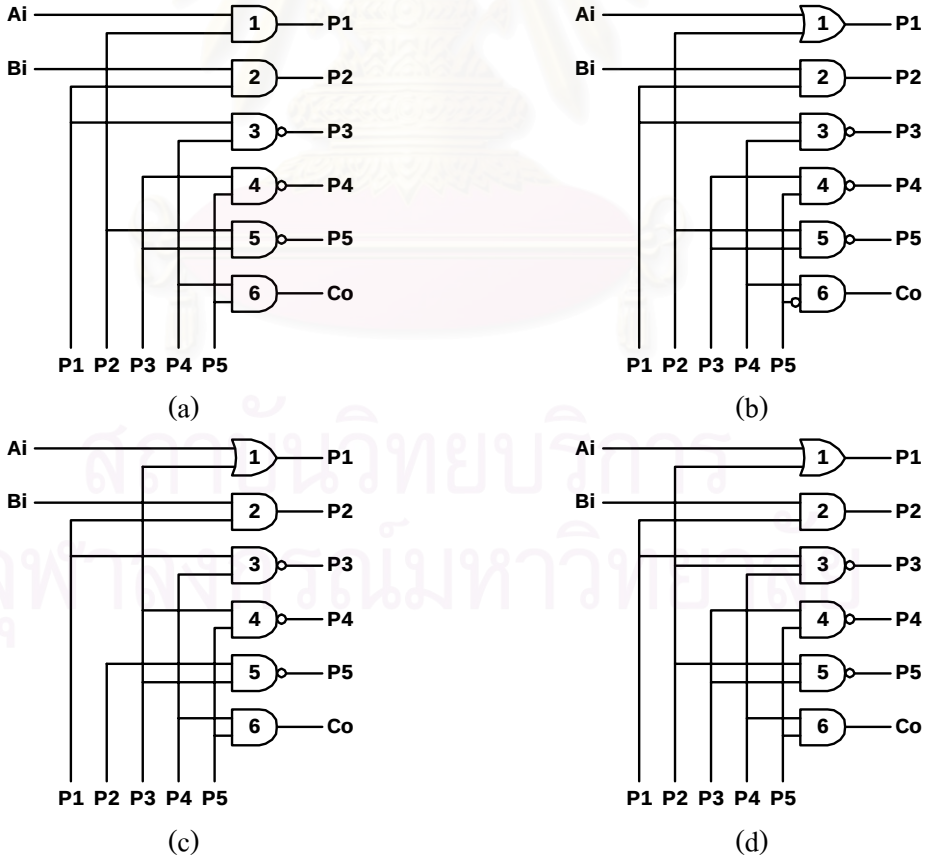
- เกตไม่ถูกต้อง ตัวอย่างจากรูปที่ 3.3 เกต 1 จะต้องเป็นเกต OR แต่เราเปลี่ยนให้เป็นเกต AND ดังแสดงในรูปที่ 6.3 (a)

- จำนวนเกตไม่ถูกต้อง ตัวอย่างจากรูปที่ 3.3 เราจะให้สายสัญญาณ P5 ที่เป็นสัญญาณอินพุตของเกต 6 ผ่านเกต NOT ดังแสดงในรูปที่ 6.3 (b)

2. สายสัญญาณ

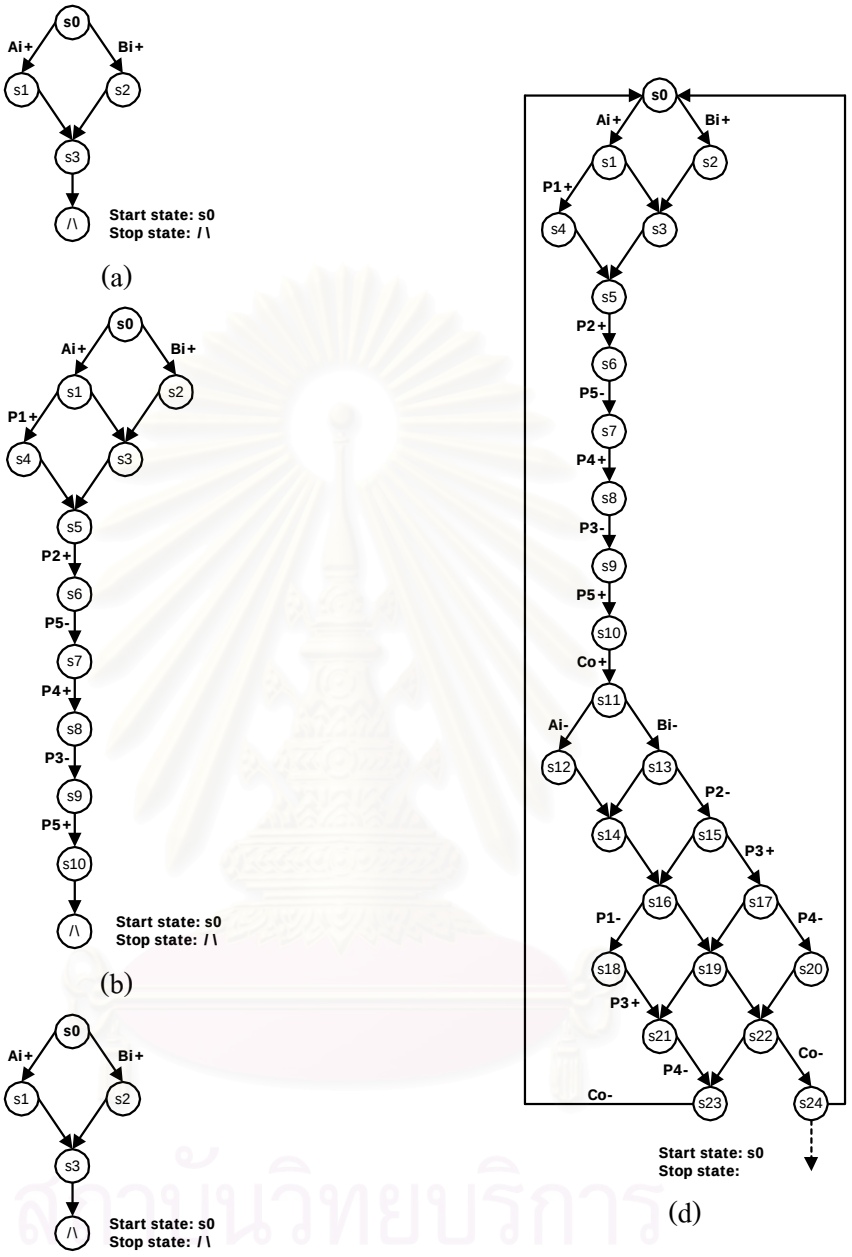
- สายสัญญาณไม่ถูกต้อง ตัวอย่างจากรูปที่ 3.3 เกต 1 จะต้องมี P2 เป็นสายสัญญาณอินพุต แต่เราเปลี่ยนให้เป็นสายสัญญาณ P3 ดังแสดงในรูปที่ 6.3 (c)

- จำนวนสายสัญญาณไม่ถูกต้อง ตัวอย่างจากรูปที่ 3.3 เกต 3 จะมีจำนวนสายสัญญาณอินพุตทั้งหมด 2 เส้น ได้แก่สายสัญญาณ P1 และ P4 แต่เราจะเพิ่มสายสัญญาณ P2 เข้าไป ดังแสดงในรูปที่ 6.3 (d)



รูปที่ 6.3 วงจร C-element ที่เกิดความผิดพลาดในกรณีต่างๆ

จากวงจรรูปที่ 6.3 นำไปสร้างเครื่องจักรสถานะจำกัดของวงจรที่เกิดความผิดพลาดในกรณีต่างๆ ดังแสดงในรูปที่ 6.4



รูปที่ 6.4 FSM ของวงจร C-element ที่เกิดความผิดพลาดในกรณีต่างๆ อ้างอิงจากรูปที่ 6.3

เมื่อเรานำเอาเครื่องจักรสถานะจำกัดที่ได้จากรูปที่ 6.4 นำไปตรวจสอบความสัมพันธ์กันระหว่างเครื่องจักรสถานะจำกัด กับเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจรที่ได้ออกแบบไว้จากรูปที่ 6.1 จะพบว่ามีสถานะบางสถานะของเครื่องจักรสถานะจำกัดที่ได้จากรูปที่ 6.4 มีการเปลี่ยนแปลงสัญญาณ และจำนวนของการเปลี่ยนแปลงสัญญาณไม่ถูกต้องตรงตามเครื่องจักรสถานะจำกัดที่ได้จากรูปที่ 6.1 ทำให้วงจรจากรูปที่ 6.3 ไม่ผ่านการทวนสอบ

6.4 ผลการทดลอง

ในหัวข้อนี้จะแสดงผลการทดลองที่ได้เมื่อนำขั้นตอนวิธีการทวนสอบวงจรที่ได้นำเสนอในงานวิจัยฉบับนี้ไปประยุกต์ใช้ทำการทวนสอบกับวงจรเกณฑ์เปรียบเทียบสมรรถนะจำนวน 29 วงจร ดังแสดงในภาคผนวก ซึ่งขั้นตอนวิธีการทวนสอบที่พัฒนาขึ้นเขียนโดยใช้โปรแกรม Borland C++ Builder V.5 และใช้เครื่องคอมพิวเตอร์ CPU Pentium III 450 MHz Ram 128 Mb ในการประมวลผล

ตารางที่ 6.1 ผลการทดลองการทวนสอบกับวงจรเกณฑ์เปรียบเทียบสมรรถนะ

Circuit	Specification		Implementation		Verification
	State	Time (ms)	State	Time (ms)	
C-element	20	0.41	20	0.44	verified
Converta	20	0.39	20	1.21	verified
Ebergen	18	0.32	18	0.87	verified
Half	14	0.25	14	0.37	verified
Hybridf	80	8.13	80	5.14	verified
Nowick	22	0.46	22	0.89	verified
Trimos-send (R1)	262	123	262	131	verified
Trimos-send (R2)	262	123	262	131	verified
Trimos-send (R3)	262	123	262	131	verified
Vbe5b	24	0.68	24	1.24	verified
Vbe5c	24	0.66	24	1	verified
Vbe6a (R1)	192	89.4	192	51.5	verified
Vbe6a (R2)	192	89.4	192	51.5	verified
Vbe6a (R3)	192	89.4	192	51.5	verified
Vbe6a (R4)	192	89.4	192	51.5	verified
Vbe10b (R1)	256	199.7	256	101.5	verified
Vbe10b (R2)	256	199.7	256	101.5	verified
Vbe10b (R3)	256	199.7	256	101.5	verified
Vbe10b (R4)	256	199.7	256	101.5	verified
Wrdata	24	0.64	24	0.98	verified
Wrdatab	703	1778	*	-	not verified
Chu133	27	0.74	27	0.73	verified

ตารางที่ 6.1 (ต่อ) ผลการทดลองการทวนสอบกับวงจรเกณฑ์เปรียบเทียบสมรรถนะ

Circuit	Specification		Implementation		Verification
	State	Time (ms)	State	Time (ms)	
Sendr-done	9	0.12	9	0.12	verified
Alloc-outbound	22	0.45	22	1.24	verified
Rlm	12	0.17	12	0.21	verified
Full	16	0.31	16	0.24	verified
Input	16	0.28	16	0.38	verified
Master-read	2525	39584	*	-	not verified
Ram-read-sbuf	39	1.64	39	3.84	verified
Sbuf-ram-write	64	6.03	64	12.85	verified
Sbuf-read-ctl	19	0.61	19	1.75	verified
Sbuf-send-ctl	27	1.25	27	2.07	verified
Sbuf-send-pkt2	33	2.18	33	1.83	verified
Rcv-setup	11	0.34	11	0.58	verified
Nak-pa	58	6.79	58	14.06	verified
Mp-forward-pkt	32	1.51	32	1.66	verified
Fifo	26	0.65	26	2.89	verified

หมายเหตุ * หมายถึง State Explosion

จากผลการทดลองในตารางที่ 6.1 จะแสดงจำนวนสถานะของเครื่องจักรสถานะจำกัดทั้งของคุณลักษณะของวงจร และวงจรที่ได้จากการสังเคราะห์ โดยจะมีบางวงจรที่มีสถานะเริ่มต้นได้หลายสถานะ ได้แก่วงจร trimos-send, vbe6a และ vbe10b ทำให้ต้องสร้างเครื่องจักรสถานะจำกัดสำหรับแต่ละสถานะเริ่มต้นที่เป็นไปได้ทั้งหมด และทำการทวนสอบเครื่องจักรสถานะจำกัดเหล่านั้น ส่วนวงจร master-read และ wrdatab เกิด state explosion ในขั้นตอนการสร้างเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์เนื่องจากในส่วน implement ของวงจรจะมีเกต และสัญญาณภายในเป็นจำนวนมากทำให้สถานะที่เป็นไปได้จะมีจำนวนมากโดยเราสามารถที่จะคำนวณหาค่าประมาณของจำนวนสถานะที่เป็นไปได้ทั้งหมดจากความน่าจะเป็นของสถานะในกราฟแสดงพฤติกรรมของเกต ดังนั้นค่าประมาณของจำนวนสถานะที่เป็นไปได้ทั้งหมดของวงจร master-read คือ 8!8!8!16!16!8!12!27!12!27!12!12!12! และค่าประมาณของจำนวนสถานะที่เป็นไปได้ทั้งหมดของ วงจร wrdatab คือ 16!16!8!8!16!16!16!16!8!8!12!27!27!8!12!12!

6.5 สรุป

ในบทนี้ได้กล่าวถึงขั้นตอนวิธีการทวนสอบวงจร, การตรวจสอบความสัมพันธ์กันระหว่างเครื่องจักรสถานะจำกัด และตัวอย่างการประยุกต์ใช้ขั้นตอนวิธีทวนสอบกับวงจร C-element และวงจรที่ไม่ถูกต้องตรงตามคุณลักษณะที่ได้ออกแบบไว้

นอกจากนี้ยังได้แสดงผลการทดลองเมื่อได้นำขั้นตอนวิธีการทวนสอบที่ได้นำเสนอในงานวิจัยฉบับนี้ไปทวนสอบกับวงจรเกณฑ์เปรียบเทียบสมรรถนะจำนวน 29 วงจร ซึ่งผลปรากฏว่ามี 27 วงจรที่สามารถประยุกต์ใช้ขั้นตอนวิธีการทวนสอบได้อย่างมีประสิทธิภาพ โดยที่มี 3 วงจร ได้แก่ trimos-send, vbe6a และ vbe10b ซึ่งสำหรับวงจรเหล่านี้จะมีสถานะเริ่มต้นของวงจรได้หลายสถานะดังนั้นในการทวนสอบจึงจำเป็นต้องทำการทวนสอบวงจรสำหรับแต่ละสถานะเริ่มต้นที่เป็นไปได้ทุกกรณี และอีก 2 วงจรได้แก่ wrdatab และ master-read ไม่สามารถทำการทวนสอบได้เพราะเกิด state explosion ขึ้นในขั้นตอนการสร้างเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ สำหรับเวลาที่ใช้ในการทำงานของขั้นตอนวิธีที่ได้นำเสนอคือ O(1) ในกรณีที่ใช้เวลาน้อยที่สุดและ O(n) ในกรณีที่ใช้เวลามากที่สุด โดยที่ n คือจำนวนของสถานะในเครื่องจักรสถานะจำกัด

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 7
สรุปผลการวิจัยและข้อเสนอแนะ

งานวิจัยฉบับนี้เป็นงานวิจัยเกี่ยวกับการทวนสอบวงจรแบบอสมวาร ซึ่งปัจจุบันในการทวนสอบวงจรอสมวารสามารถแบ่งออกได้เป็น 3 วิธี ได้แก่การทวนสอบโดยใช้วิธีจำลอง, การทวนสอบอย่างมีแบบแผน และการทวนสอบอย่างกึ่งมีแบบแผน โดยที่ในแต่ละวิธีก็ยังคงมีข้อจำกัดอยู่ดังที่ได้กล่าวไว้แล้วในบทนำ จึงทำให้มีงานวิจัยที่นำเสนอวิธีการทวนสอบรูปแบบใหม่ เพื่อที่จะหลีกเลี่ยงข้อจำกัดที่เกิดขึ้น

7.1 สรุปผลการวิจัย

1. งานวิจัยนี้มีวัตถุประสงค์เพื่อออกแบบขั้นตอนวิธี และพัฒนาโปรแกรมการทวนสอบวงจรอสมวารแบบควอไซดีเลย์อินเซนซิทีฟโดยใช้วิธีการทวนสอบอย่างมีแบบแผน

2. ใช้ขั้นตอนวิธีการทวนสอบอย่างมีแบบแผนทำให้สามารถรับประกันได้ว่าทวนสอบวงจรได้ครอบคลุมทุกกรณี

3. ในขั้นตอนการสร้างเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ได้คำนึงถึงโมเดลสิ่งแวดล้อมที่ได้จากคุณลักษณะของวงจรทำให้สามารถลดจำนวนของสถานะที่จะเกิดขึ้นในขั้นตอนการสร้างเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์

4. ในกรณีที่คุณลักษณะของวงจรสามารถมีสถานะเริ่มต้นได้หลายสถานะ เราจะต้องทำการการทวนสอบแต่ละสถานะเริ่มต้นที่เป็นไปได้ทั้งหมด

5. ขั้นตอนในการทวนสอบจะใช้วิธีการแวะผ่านกราฟแบบ breadth-first search ซึ่งใช้เวลา O(1) ในกรณีใช้เวลาน้อยที่สุด และใช้เวลา O(n) ในกรณีที่ใช้เวลามากที่สุด โดยที่ n คือจำนวนของสถานะในเครื่องจักรสถานะจำกัด ทำให้สามารถทวนสอบวงจร และพบพฤติกรรมของวงจรที่ไม่ถูกต้องได้อย่างรวดเร็ว

7.2 ข้อจำกัด

1. ในขั้นตอนการสร้างเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจรที่ได้ออกแบบไว้สำหรับในกรณีที่มีจำนวนสัญลักษณ์ initial marking ในซิกแนลทรานสิชันกราฟมีมากกว่าหรือเท่ากับ 3 จุด และจำนวนคู่ของการเปลี่ยนแปลงสัญญาณที่มีคุณสมบัติ concurrent temporal relation ในซิกแนลทรานสิชันกราฟมีมากกว่า 4 คู่ จะทำให้มีจำนวนของสถานะในเครื่องจักรสถานะจำกัดของคุณลักษณะของวงจรมีจำนวนมาก

2. ในขั้นตอนการสร้างเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ จำเป็นต้องอาศัยโมเดลสิ่งแวดล้อมที่ได้จากคุณลักษณะของวงจรเพื่อใช้ในการกำหนดการเปลี่ยนแปลงสัญญาณอินพุต และสัญญาณเอาต์พุต

3. ในขั้นตอนการสร้างเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์จำเป็นต้องกำหนดจุดเริ่มต้นของวงจรโดยจะยึดจากค่าของสัญญาณ ณ จุดเริ่มต้นของคุณลักษณะของวงจร

4. สำหรับวงจรที่ได้จากการสังเคราะห์ที่มี delete signal จำนวนมากในขั้นตอนการสร้างเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์ ค่าความซับซ้อนของขั้นตอนวิธีจะมีค่าเท่ากับ $O((gn)^2)$ ในกรณีที่ใช้เวลามากที่สุด โดยที่ g คือจำนวนเกตในวงจรม n คือจำนวนของการเปลี่ยนแปลงสัญญาณในเกต ทำให้เสียเวลาเป็นอย่างมาก

5. สำหรับวงจรที่ได้จากการสังเคราะห์ที่มีจำนวนเกตมากกว่า 12 เกต และมีสัญญาณมากกว่า 18 สัญญาณในขั้นตอนการสร้างเครื่องจักรสถานะจำกัดของวงจรที่ได้จากการสังเคราะห์อาจจะเกิด state explosion ขึ้นทำให้ไม่สามารถทำการทวนสอบได้ ตัวอย่างของวงจรได้แก่ wrdatab ซึ่งมี 16 เกต 20 สัญญาณ และ master-read ซึ่งมี 13 เกต 19 สัญญาณ

7.3 ข้อเสนอแนะ

เนื่องจากงานวิจัยที่ได้นำเสนอใช้เครื่องจักรสถานะจำกัดในการแสดงพฤติกรรมของวงจร จึงไม่สามารถการกำหนดค่าความหน่วงของเกต และสายสัญญาณได้ ทำให้ควรพัฒนาให้สามารถกำหนดค่าความหน่วงของเกต และสายสัญญาณเพื่อให้สามารถนำวิธีการทวนสอบที่ได้นำเสนอนี้ไปใช้ในการทวนสอบวงจรอสมวารแบบดีเลย์โมเดลอื่นๆ ได้มีประสิทธิภาพดียิ่งขึ้น

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

รายการอ้างอิง

[1] Alex Cowie, George J. Milne http://www.acrc.unisa.edu.au University of South Australia School of Computer and Information Service, March 1999.

[2] Chu, T. Synthesis of Self-timed VLSI Circuits from Graph-theoretic Specifications. Doctoral dissertation Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, 1987.

[3] Clarke E.M. and Wing J.M. Formal Methods: State of the Art and Future Directions. IEEE Trans. Computers 1996: 1-22

[4] Dill D.L. What's Between Simulation and Formal Verification?. Design Automation Conference 1998: 328-329.

[5] George M. Formal Specification and Verification of Digital Systems. Cambridge: McGraw-Hill 1994.

[6] Hauck S. Asynchronous Design Methodologies: An Overview, Proceeding of the IEEE, Vol.83, No.1, pp. 69-93, January 1995.

[7] Nanya, T., Ueno, Y., Kagotani, H. Kuwako, M. and Takamaru, A. TITAC: Design Of a Quasi-Delay-Insensitive Microprocessor. IEEE Design & Test of Computers, 1994.

[8] Park, S.B. Synthesis of Asynchronous VLSI Circuits from Signal Transition Graph Specifications. Doctoral dissertation, Department of Engineering-Computer Science, Tokyo Institute of Technology, 1996.

[9] Sentovich, E.M., et al. SIS: A System for Sequential Circuit Synthesis. U.C. Berkeley, UCB/ERL memo M92/41, May 1992.

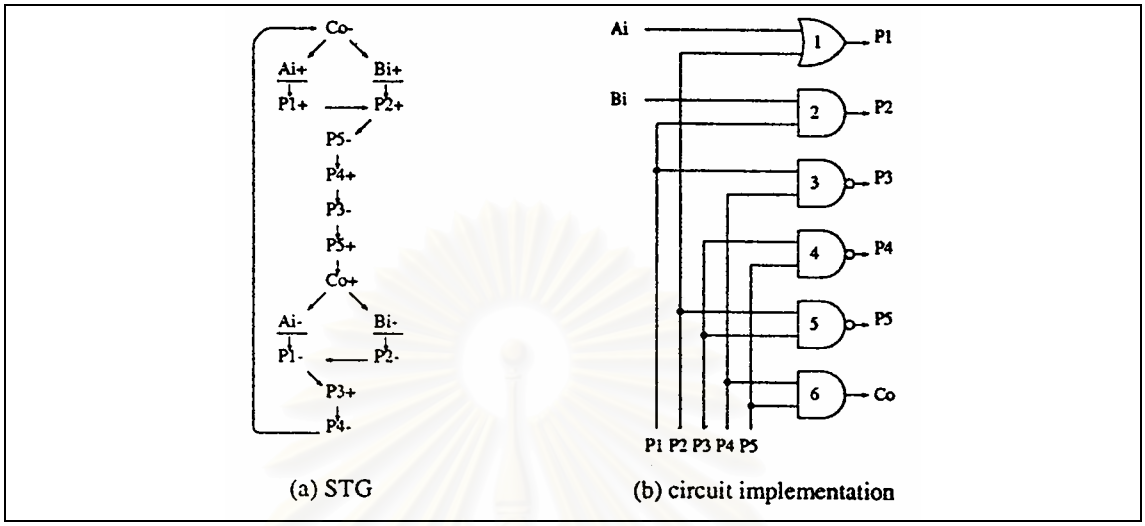
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก

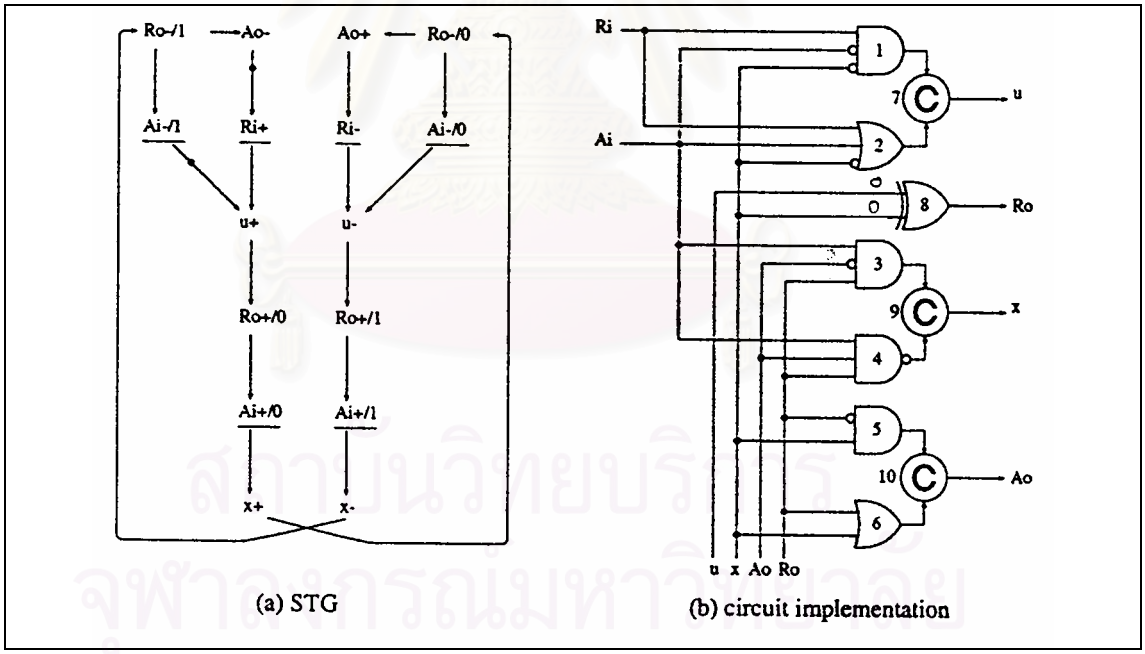
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

วงจรเกณฑ์เปรียบเทียบสมรรถนะ

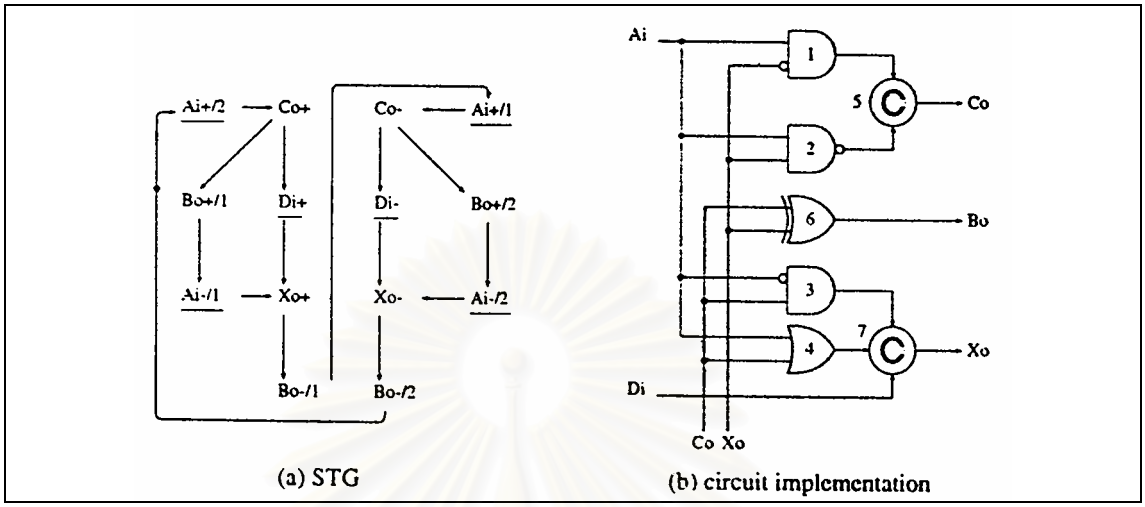
c-element



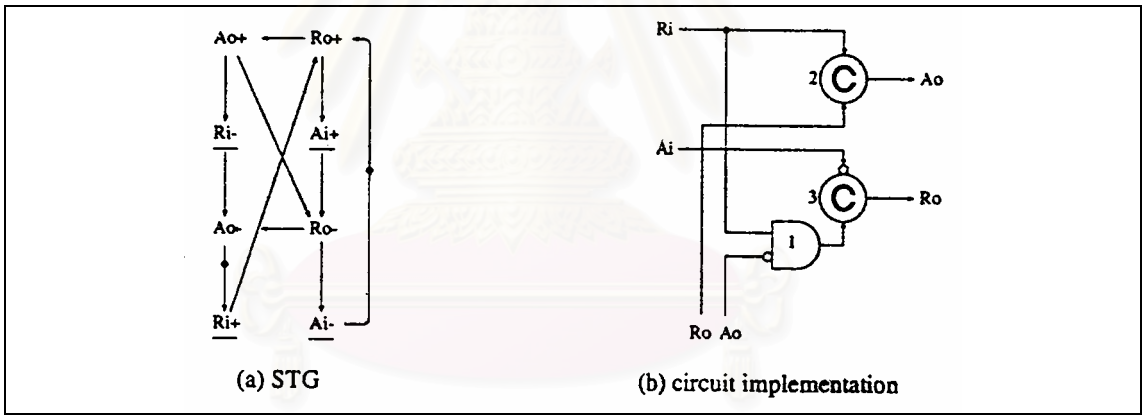
converta



ebergen

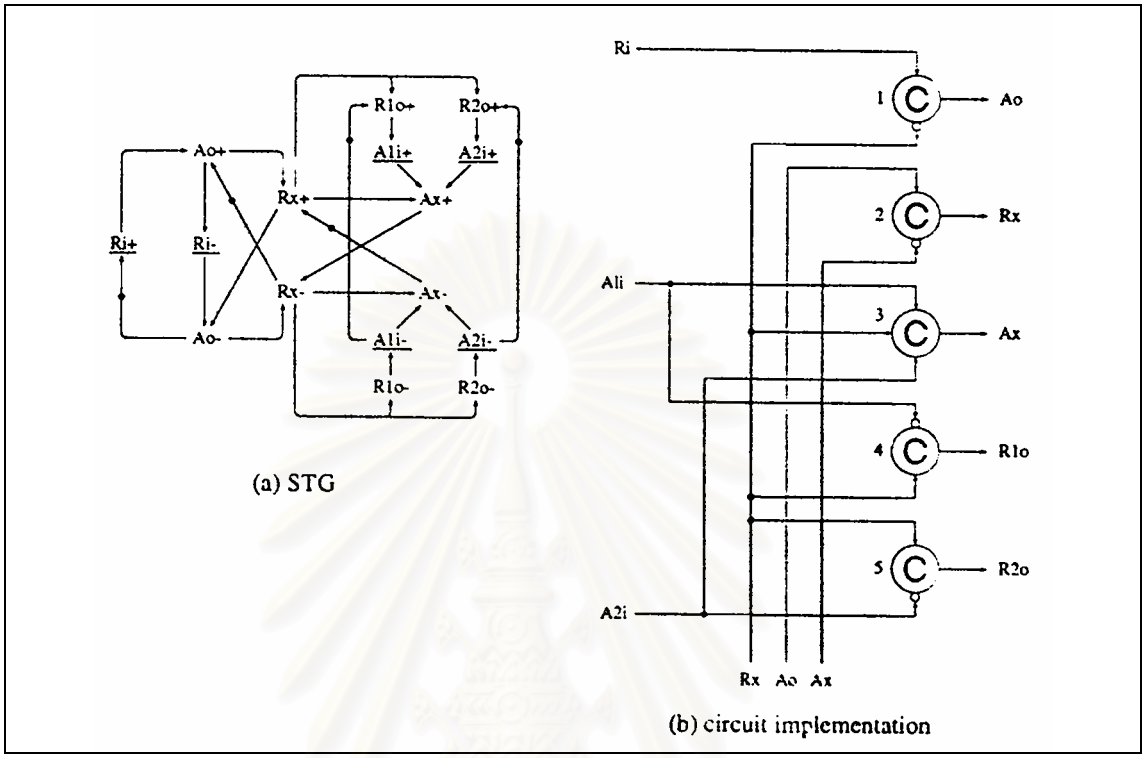


half

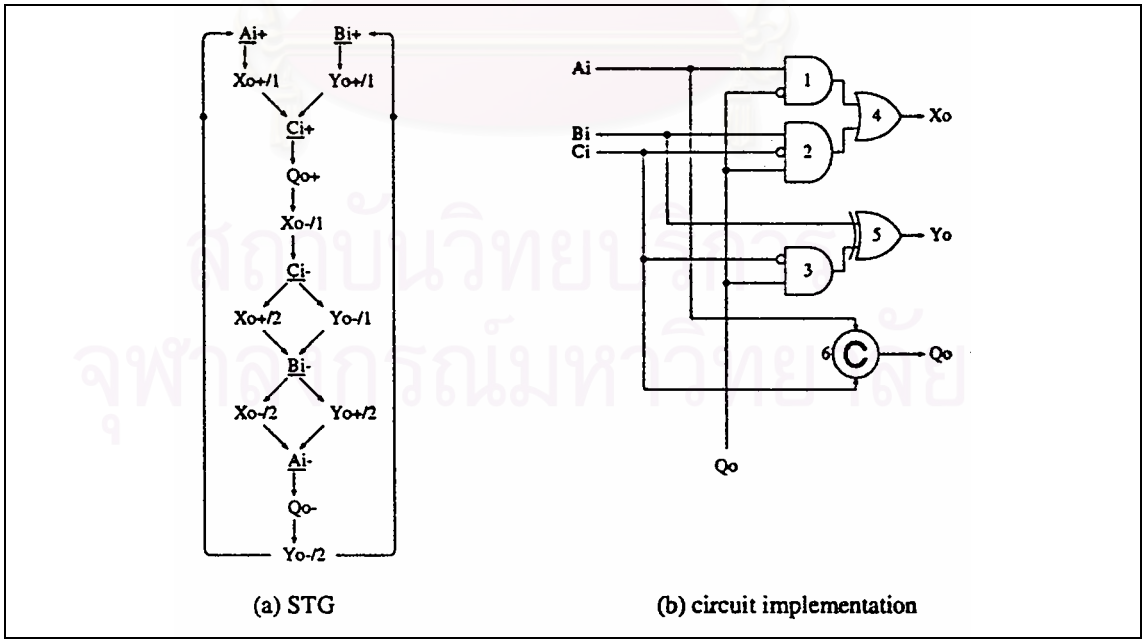


สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

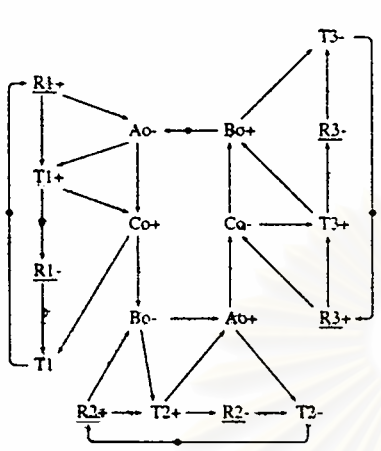
hybridf



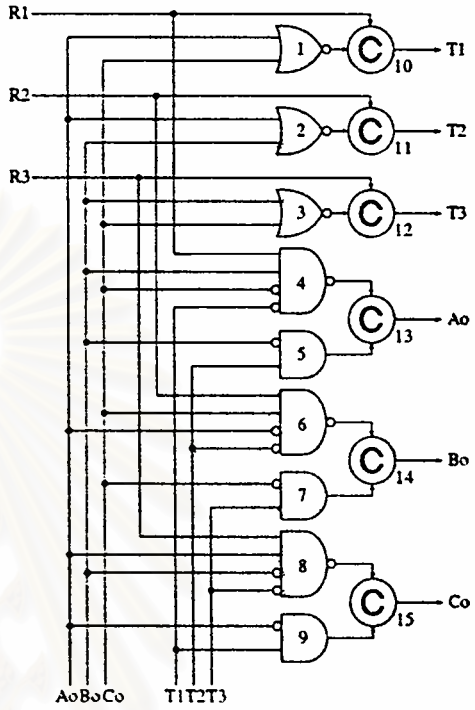
nowick



trimos-send

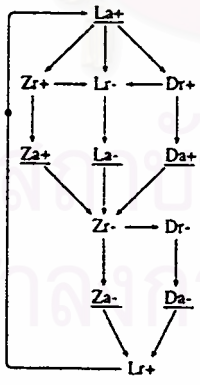


(a) STG

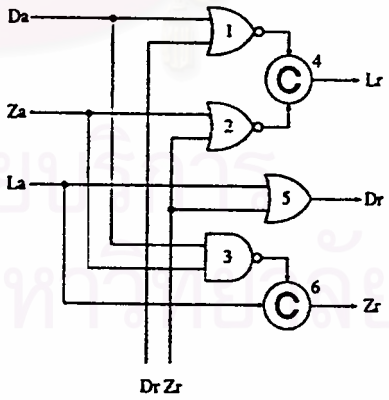


(b) circuit implementation

vbe5b

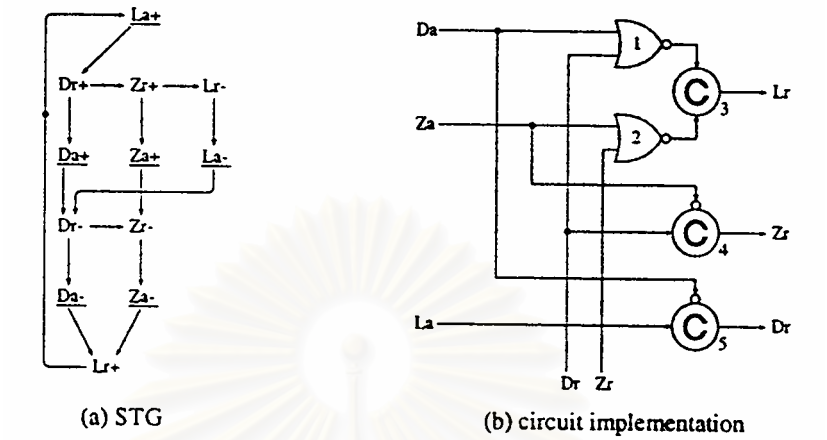


(a) STG

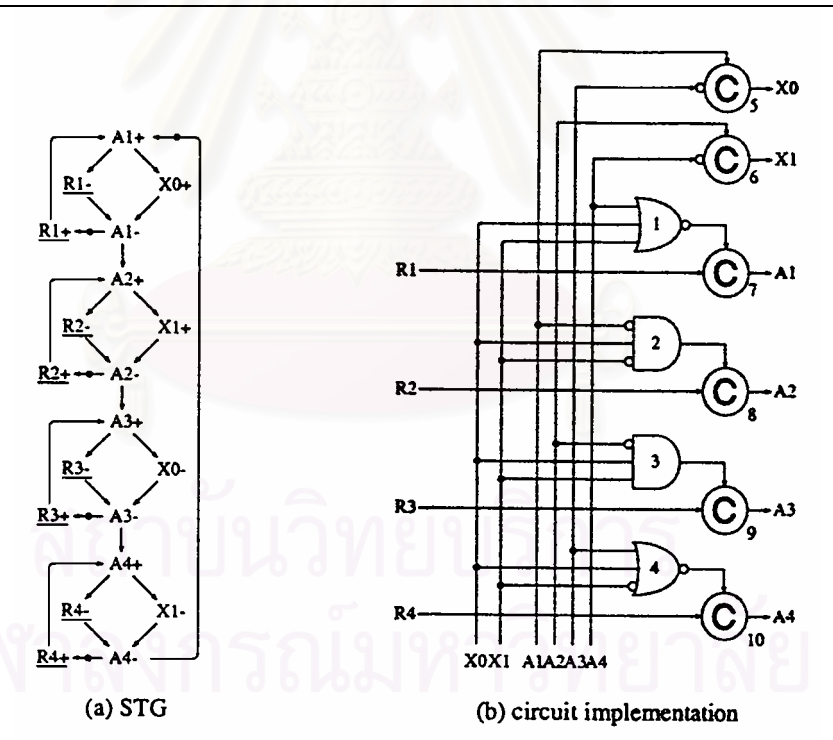


(b) circuit implementation

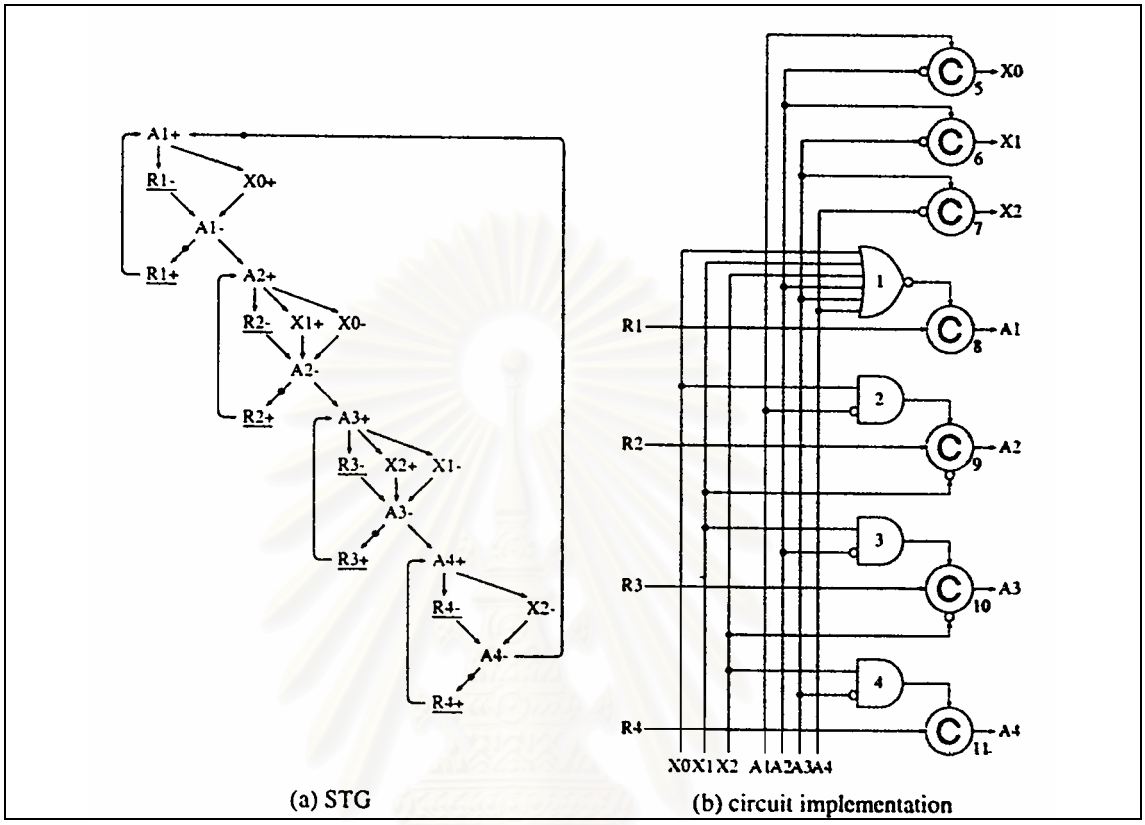
vbe5c


La+
Dr+
Zr+
Lr-
Da+
Za+
La-
Dr-
Zr-
Da-
Za-
Lr+
(a) STG
Da
Za
La
Dr
Zr
Lr
Zr
Dr
1
2
3
4
5
(b) circuit implementation

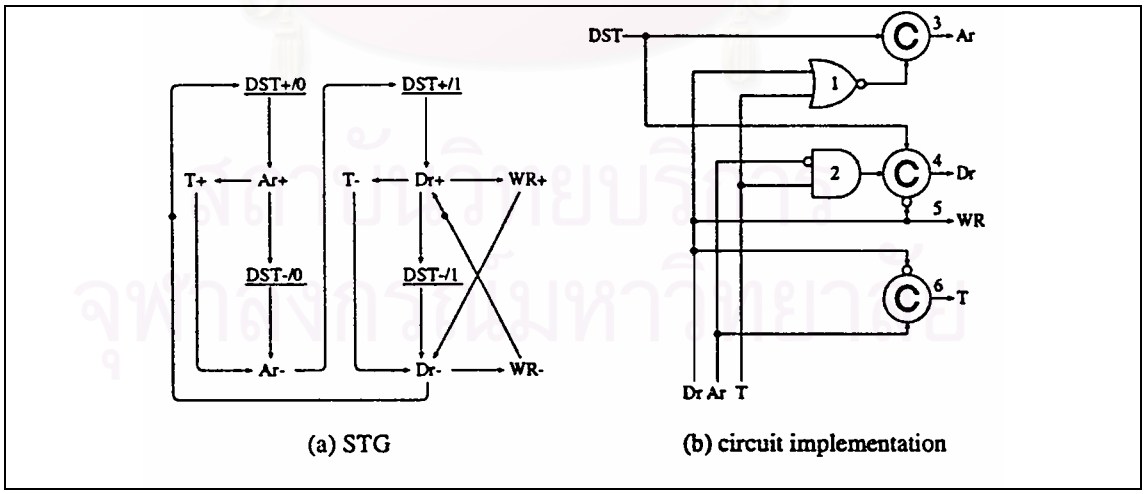
vbe6a


A1+
R1-
X0+
R1+
A1-
A2+
R2-
X1+
R2+
A2-
A3+
R3-
X0-
R3+
A3-
A4+
R4-
X1-
R4+
A4-
(a) STG
R1
R2
R3
R4
X0
X1
A1
A2
A3
A4
X0X1 A1A2A3A4
1
2
3
4
5
6
7
8
9
10
(b) circuit implementation

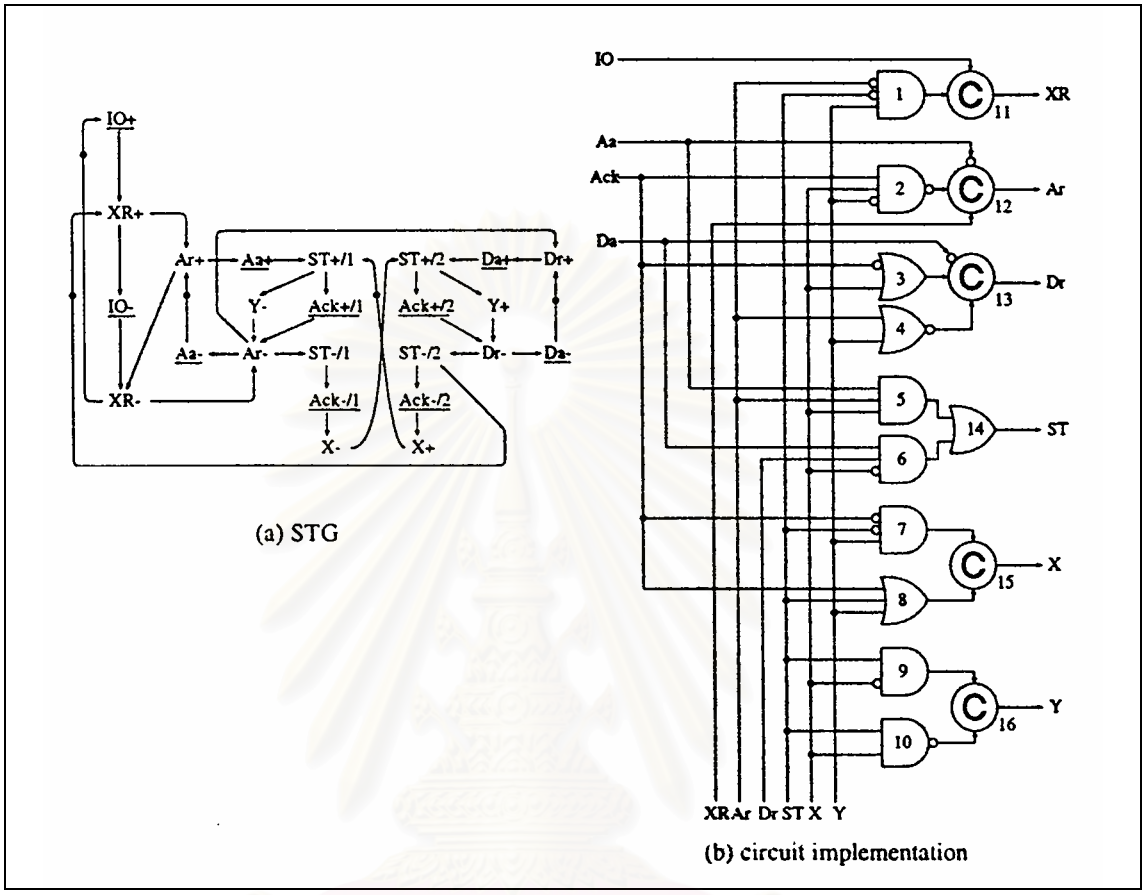
vbe10b



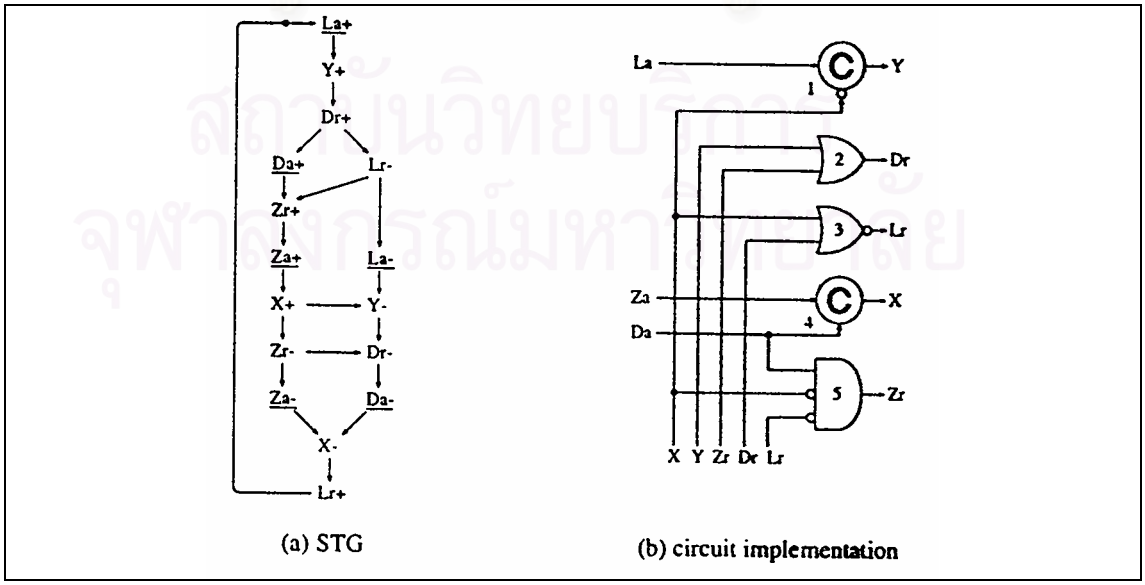
wrdata



wrdatab

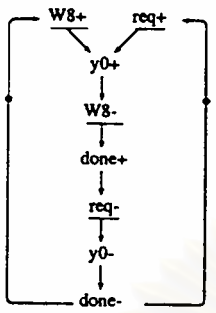


chu133

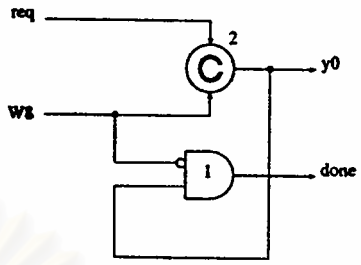


7

sendr-done

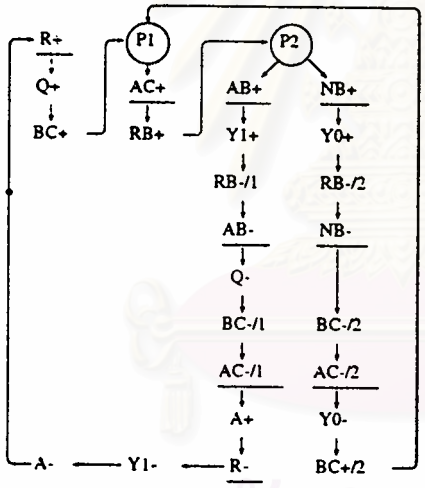


(a) STG

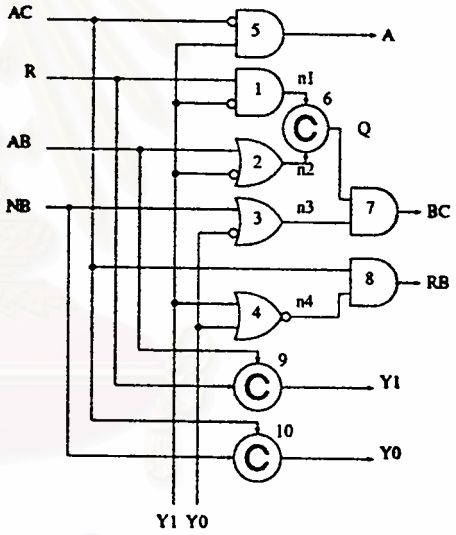


(b) circuit implementation

alloc-outbound



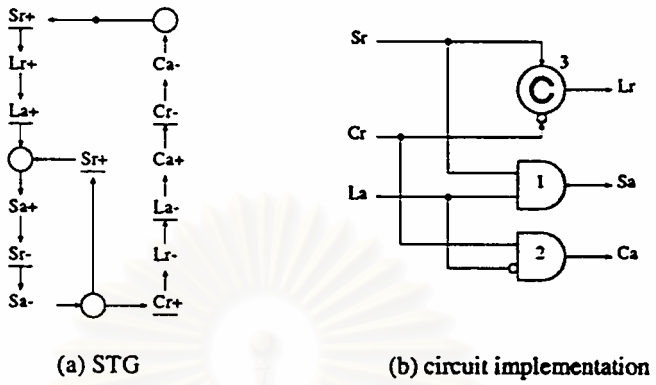
(a) STG



(b) circuit implementation

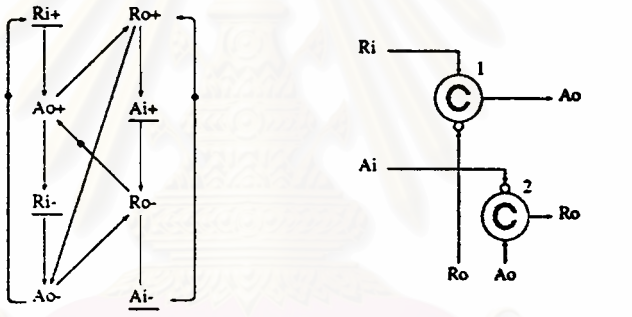
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

rlm



(a) STG (b) circuit implementation

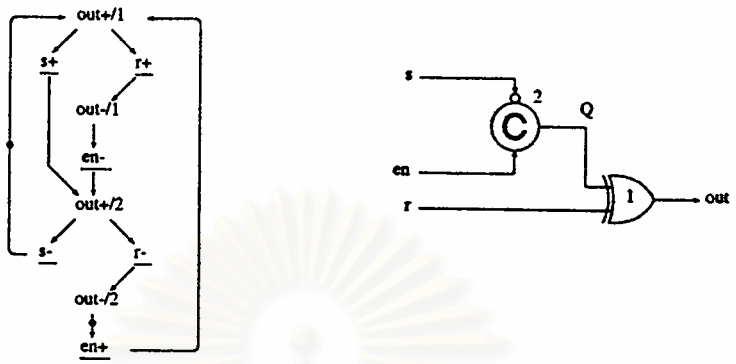
full



(a) STG (b) circuit implementation

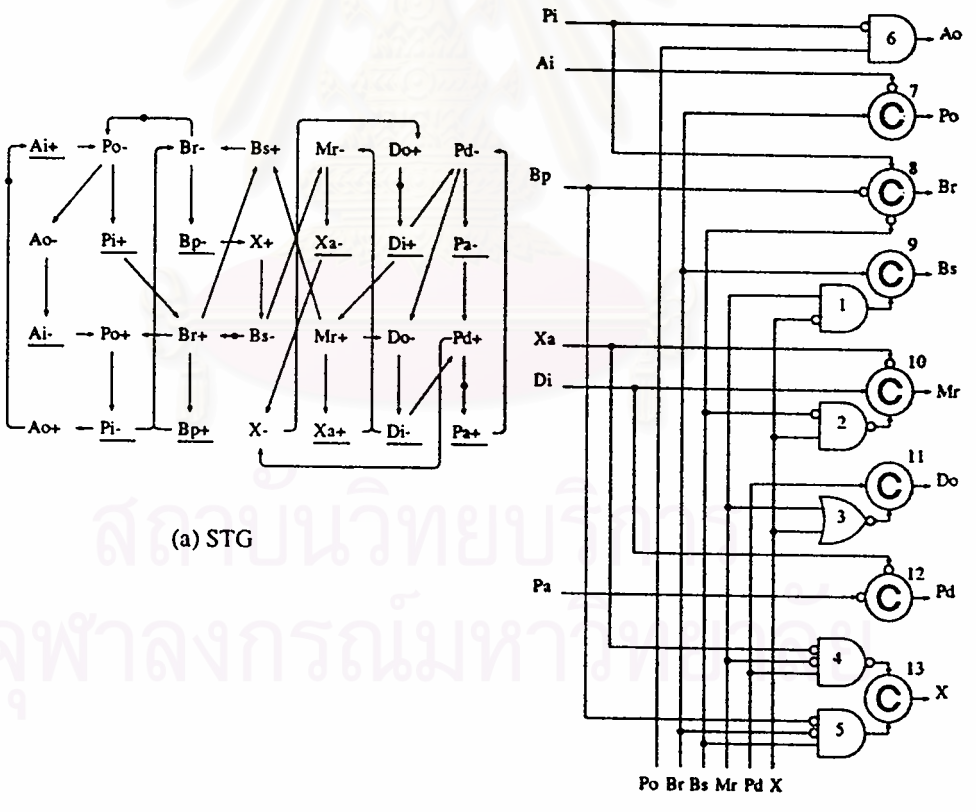
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

input



(a) STG (b) circuit implementation

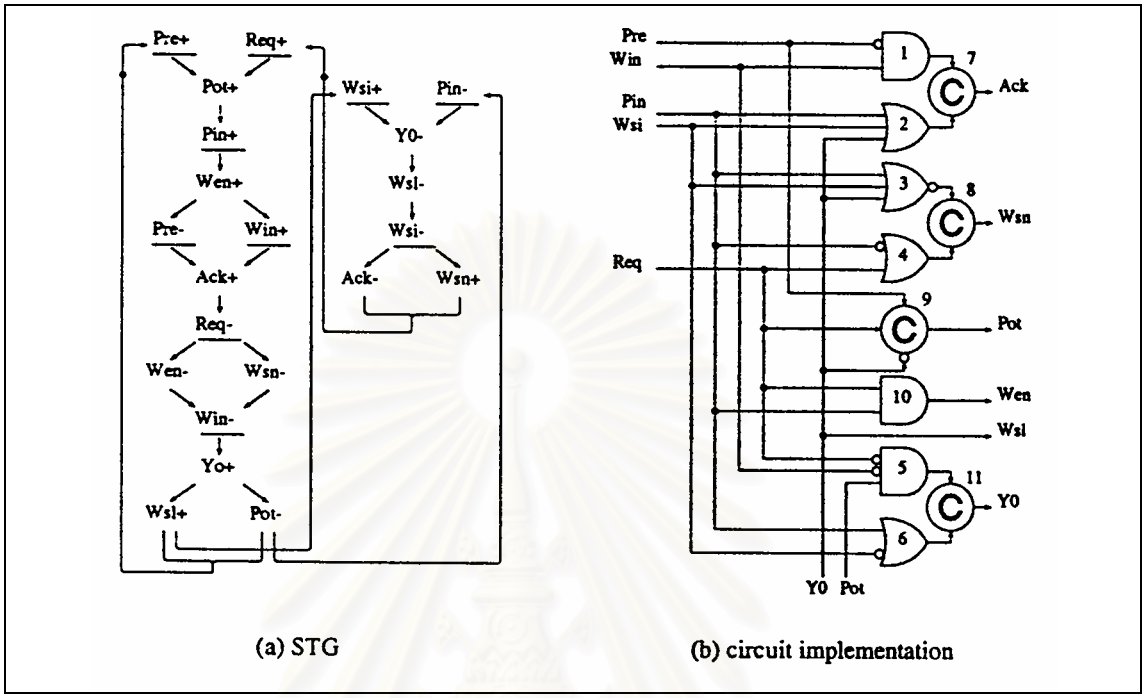
master-read



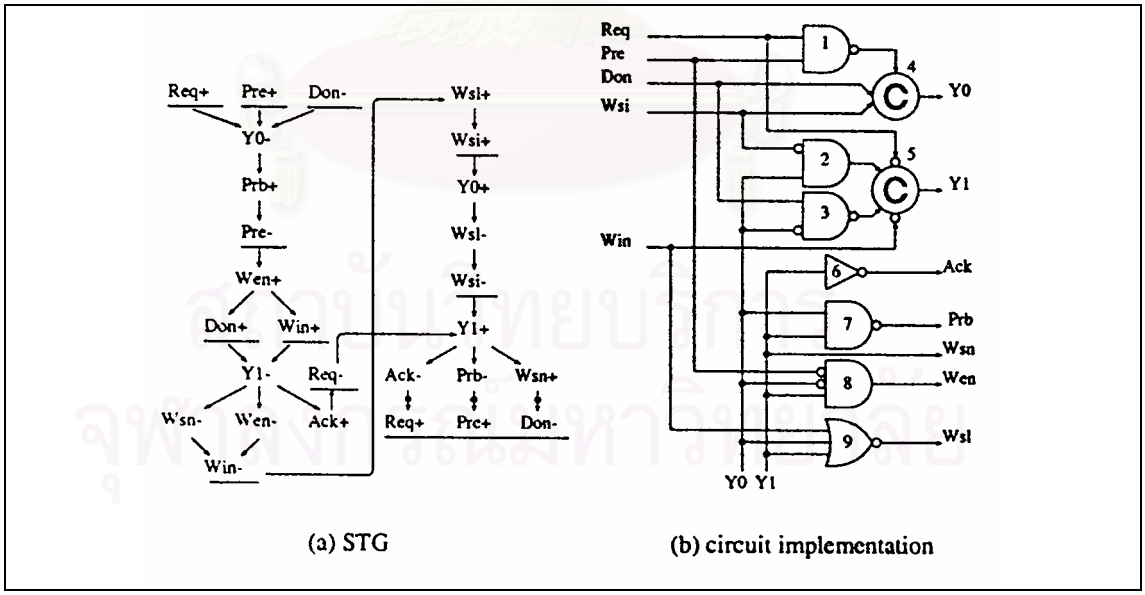
(a) STG (b) circuit implementation

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

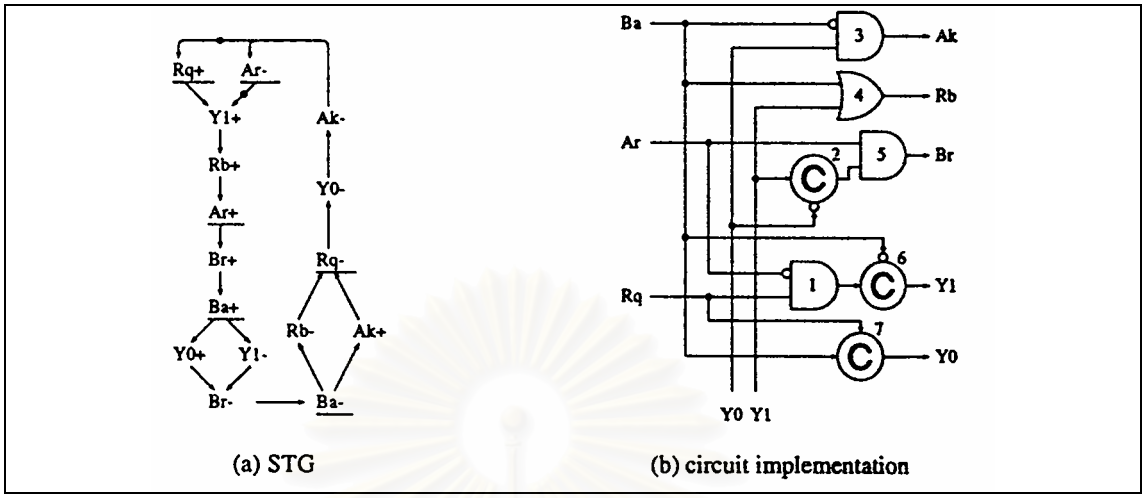
ram-read-sbuf



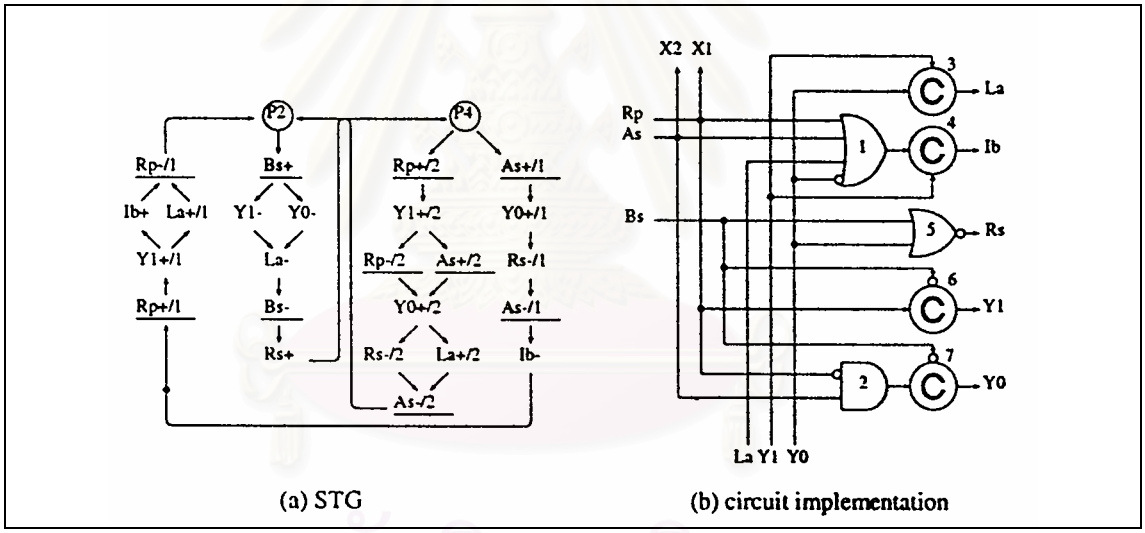
sbuf-ram-write



sbuf-read-ctl

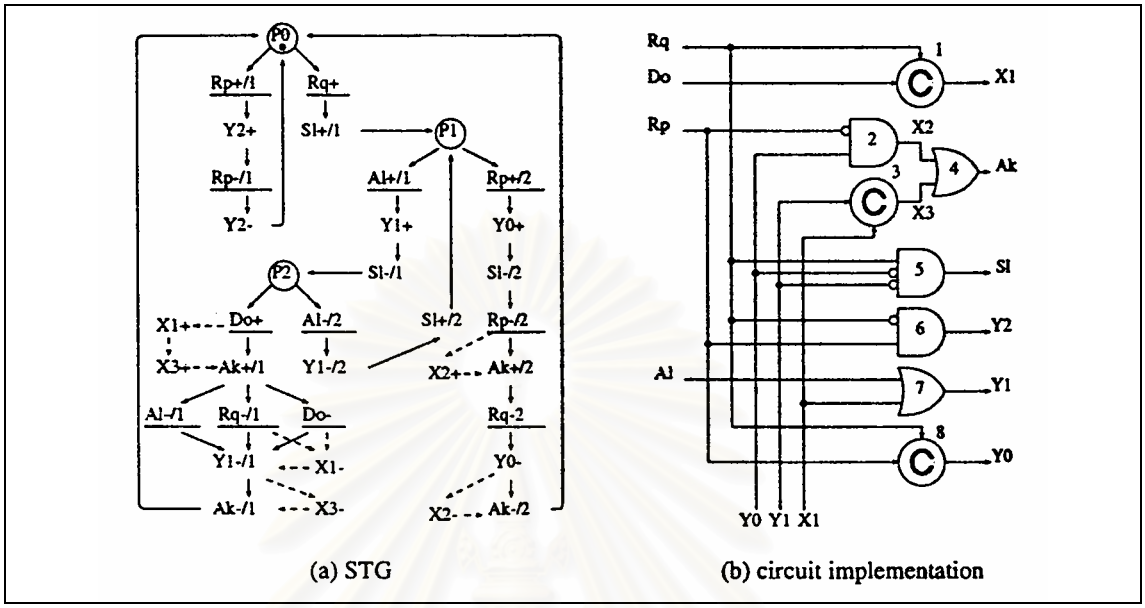


sbuf-send-ctl

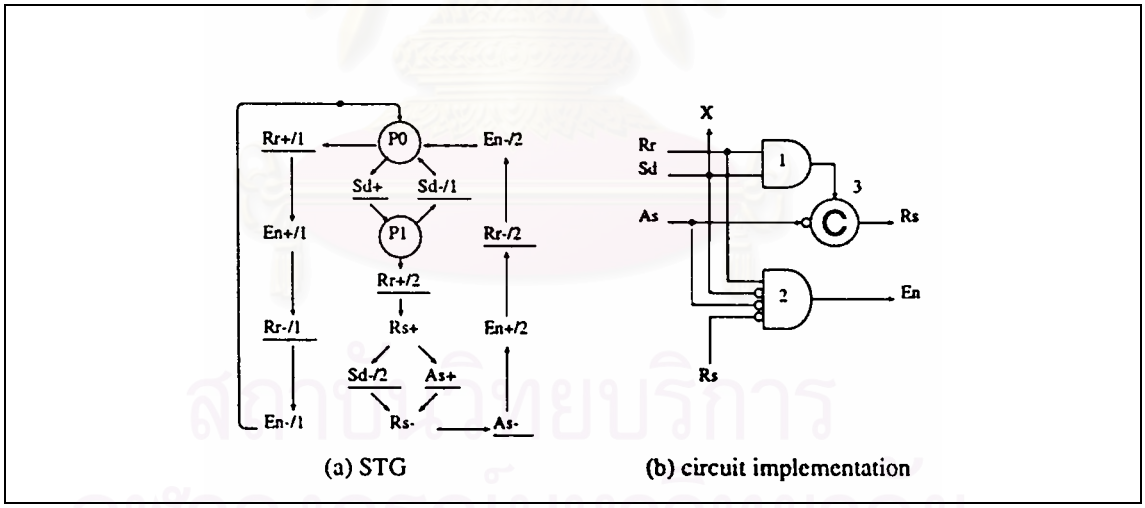


สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

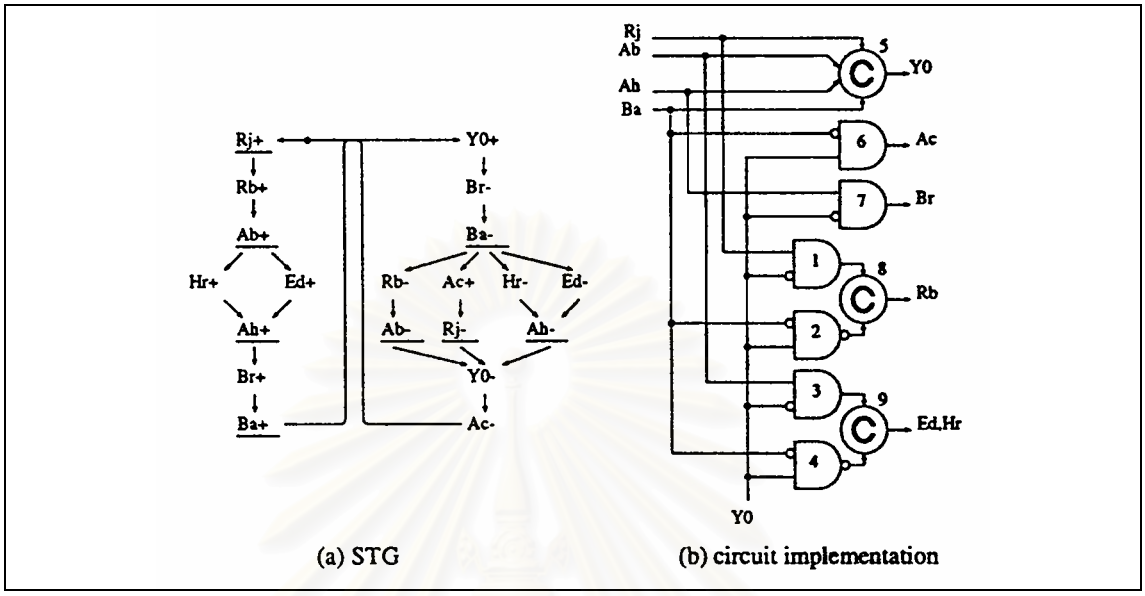
sbuf-send-pkt2



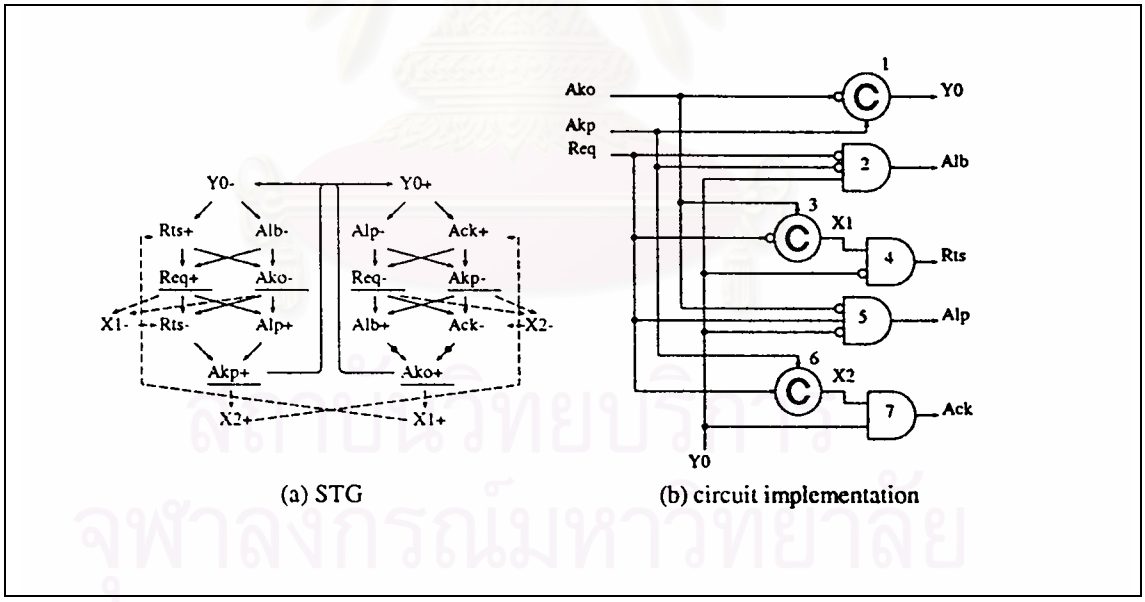
rcv-setup



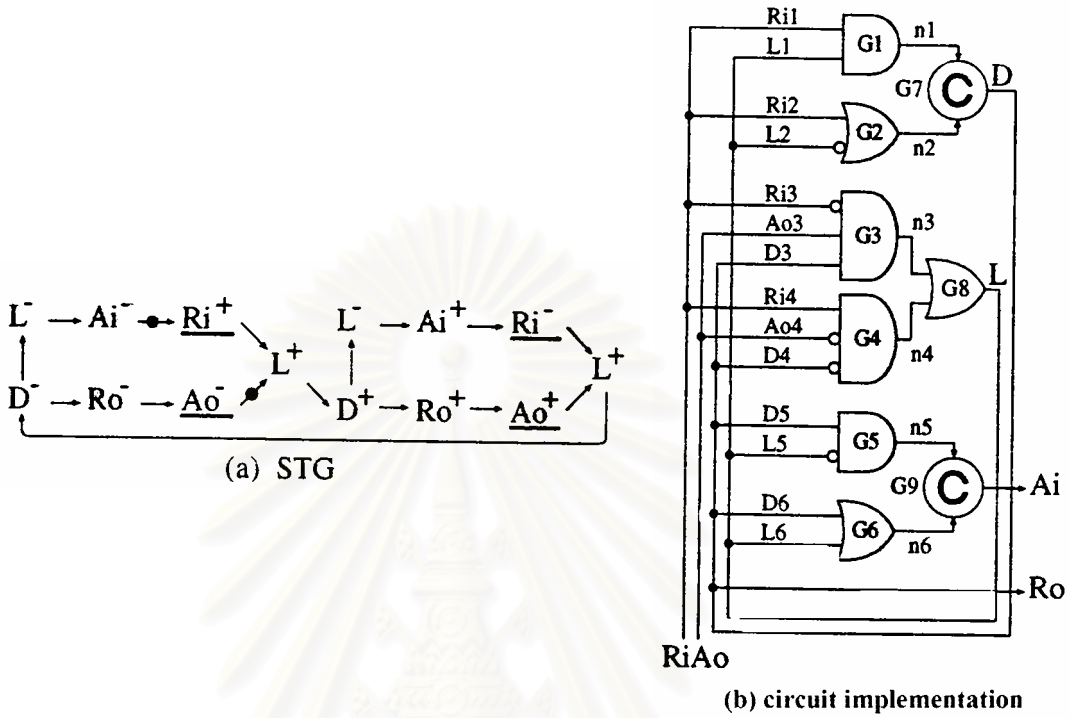
nak-pa



mp-forward-pkt1



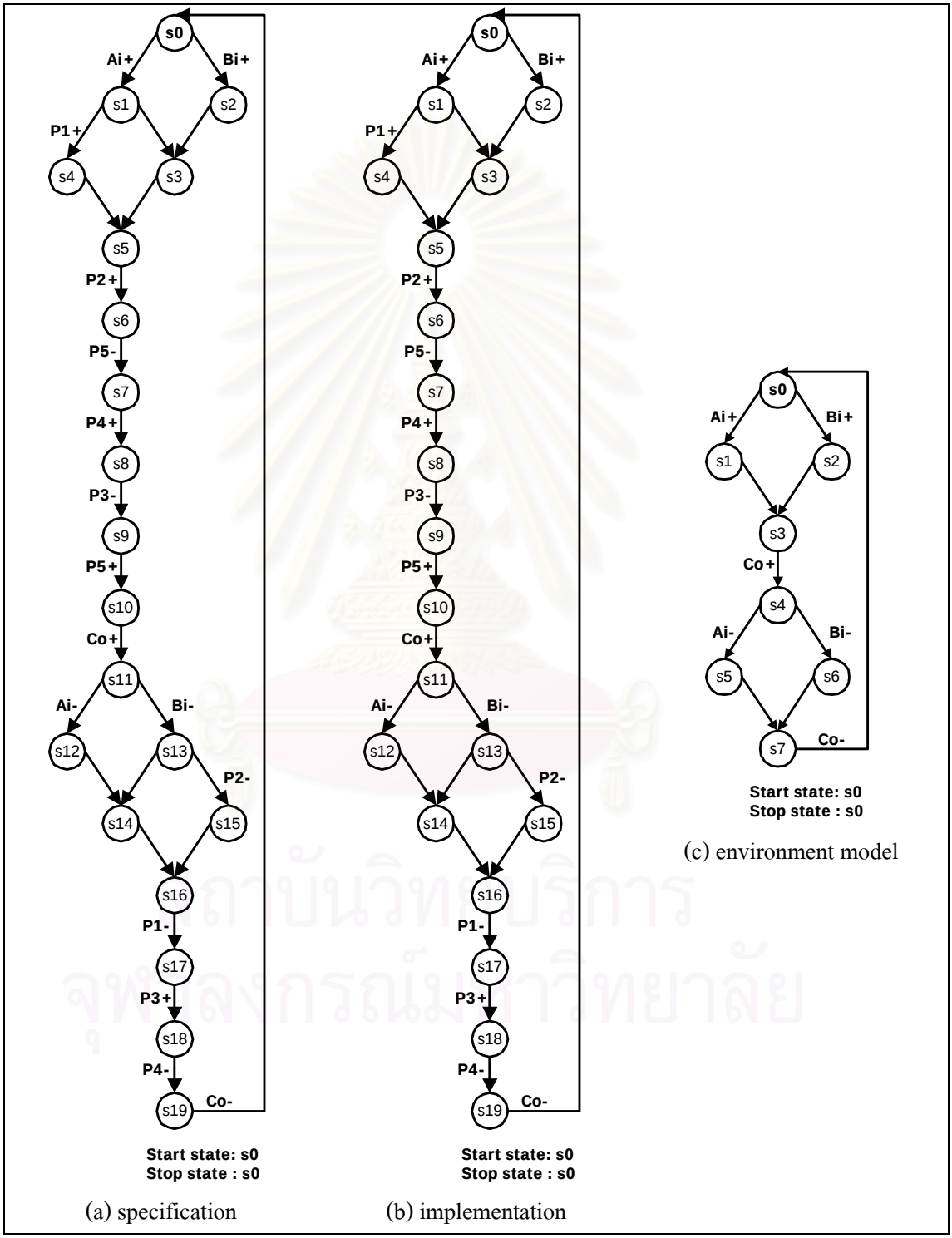
fifo



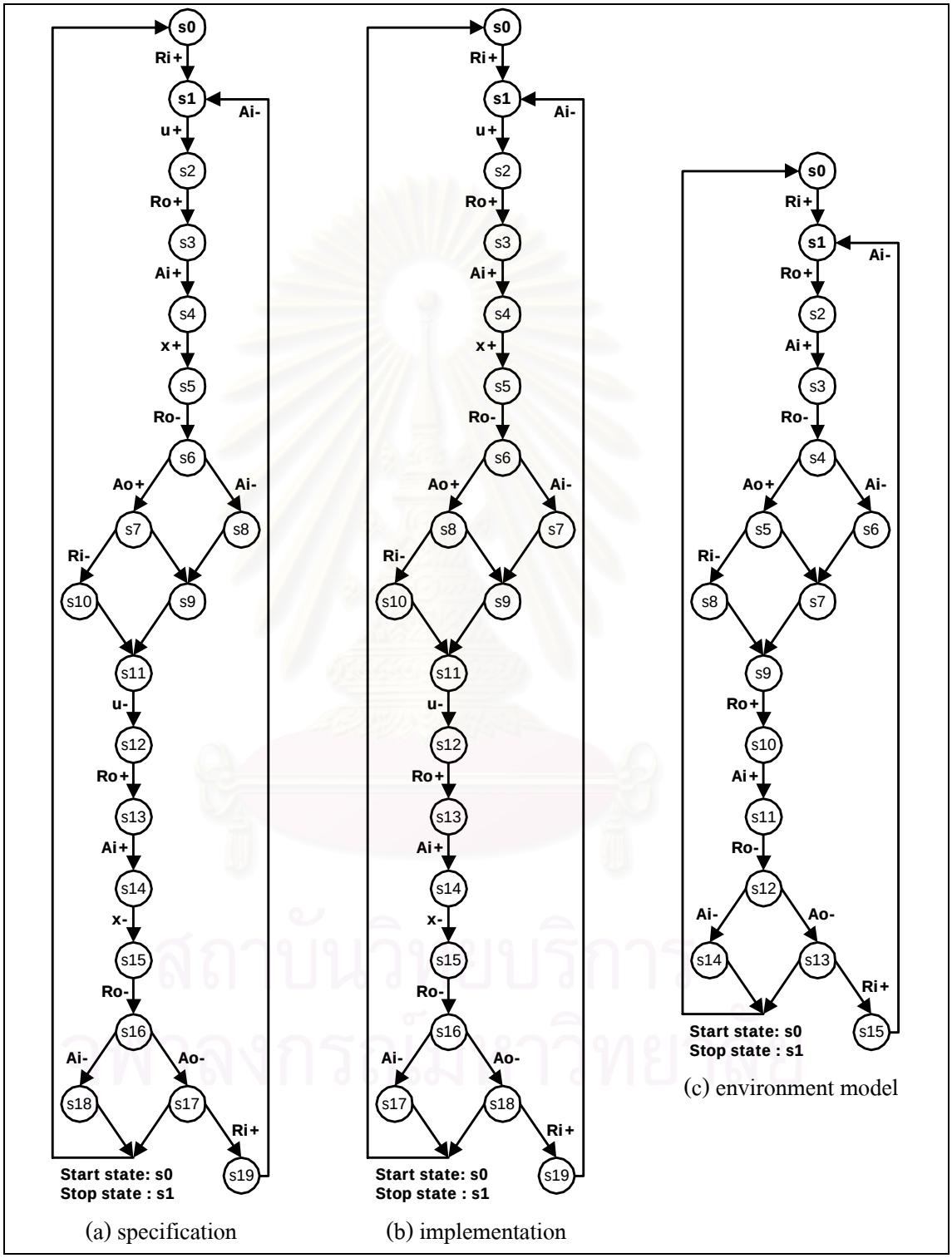
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ตัวอย่างเครื่องจักรสถานะจำกัดและกราฟโมเดลสิ่งแวดล้อม ของวงจรเกณฑ์เปรียบเทียบสมรรถนะ

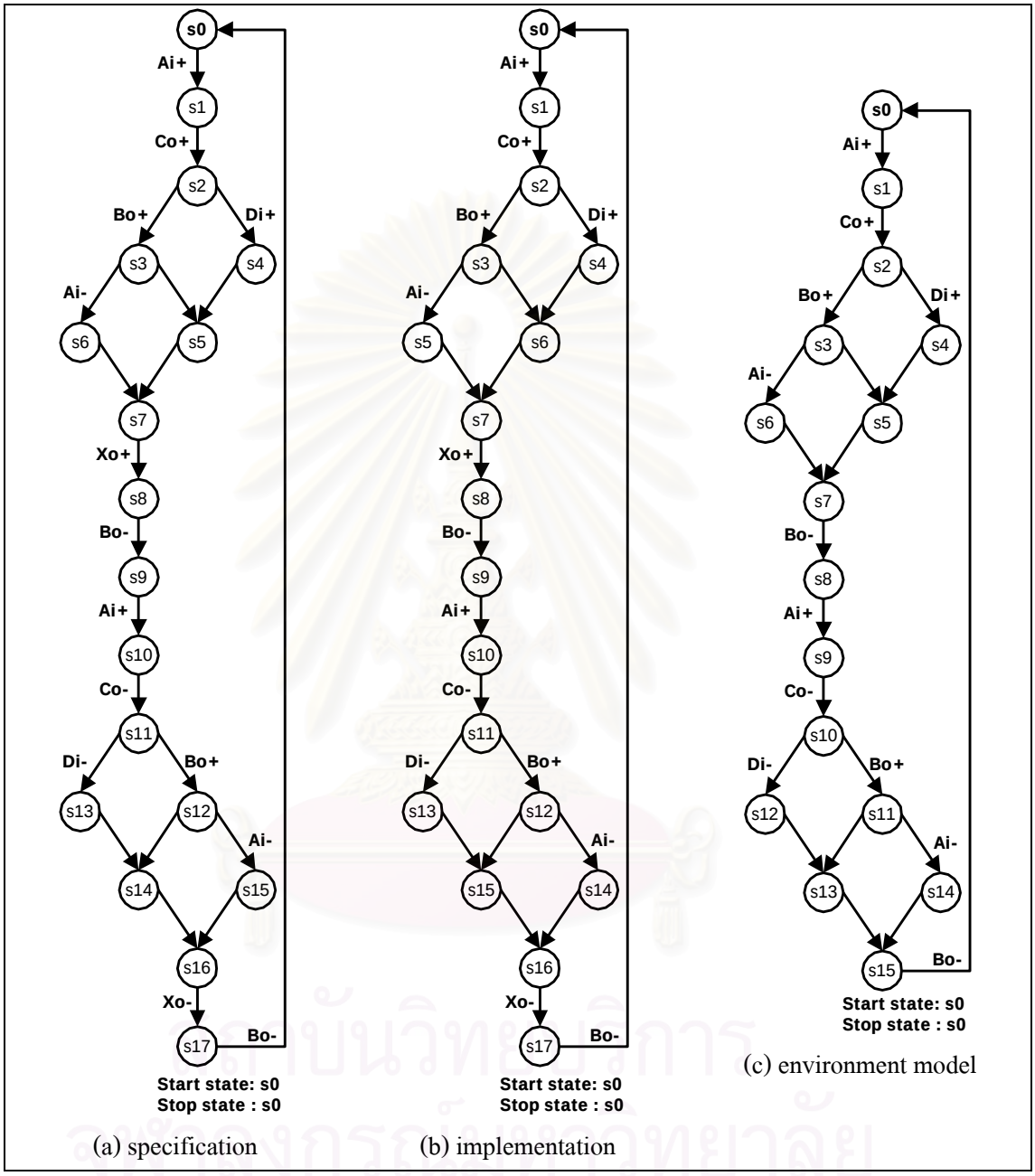
c-element



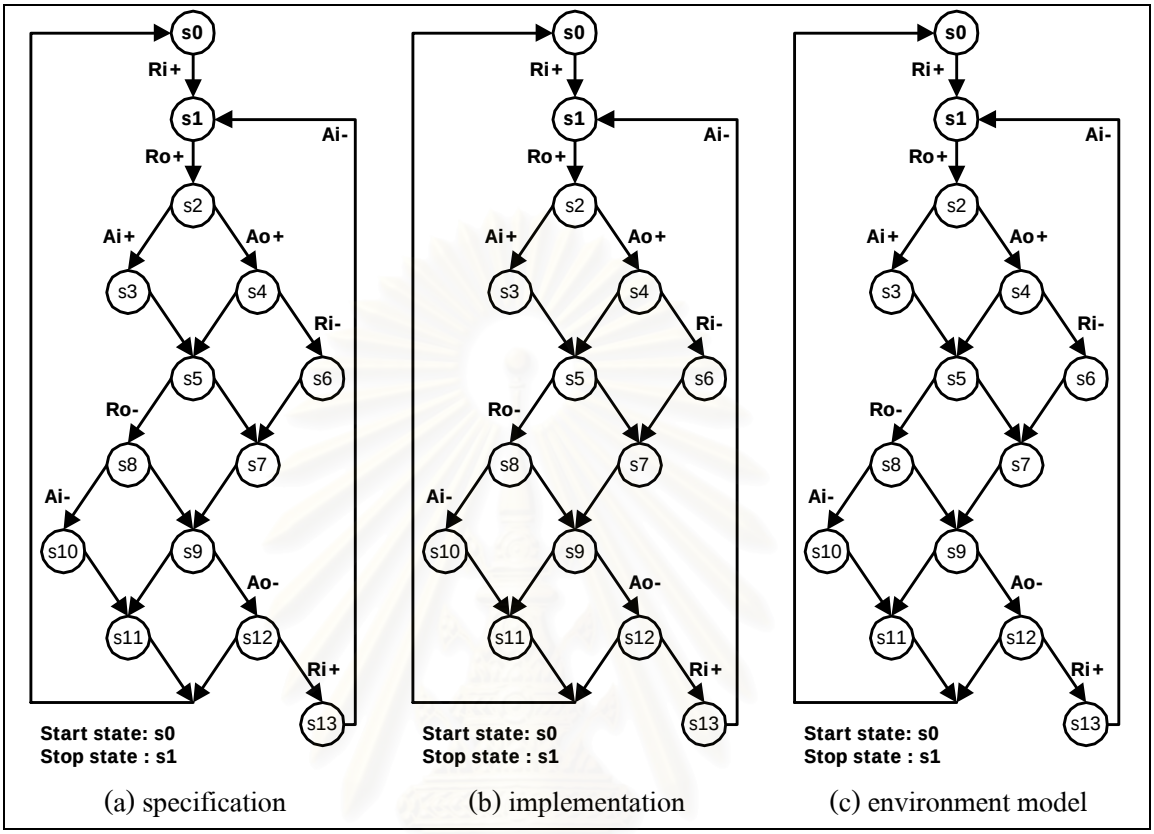
converta


s0
Ri+
s1
Ai-
u+
s2
Ro+
s3
Ai+
s4
x+
s5
Ro-
s6
Ao+
Ai-
s7
s8
Ri-
s10
s9
s11
u-
s12
Ro+
s13
Ai+
s14
x-
s15
Ro-
s16
Ai-
Ao-
s18
s17
Ri+
s19
Start state: s0
Stop state : s1
(a) specification
(b) implementation
(c) environment model

ebergen


s0
Ai+
s1
Co+
s2
Bo+
Di+
s3
s4
Ai-
s6
s5
s7
Xo+
s8
Bo-
s9
Ai+
s10
Co-
s11
Di-
Bo+
s13
s12
Ai-
s14
s15
s16
Xo-
s17
Bo-
Start state: s0
Stop state : s0
(a) specification
(b) implementation
(c) environment model

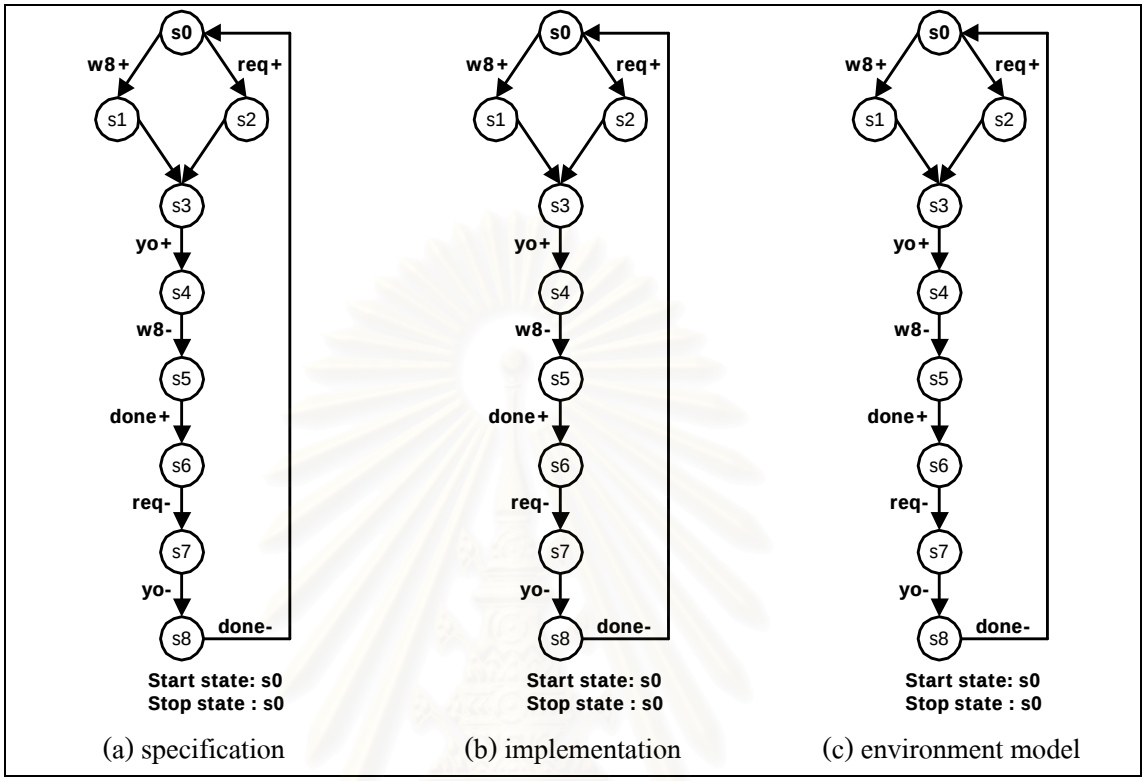
half



(a) specification (b) implementation (c) environment model

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

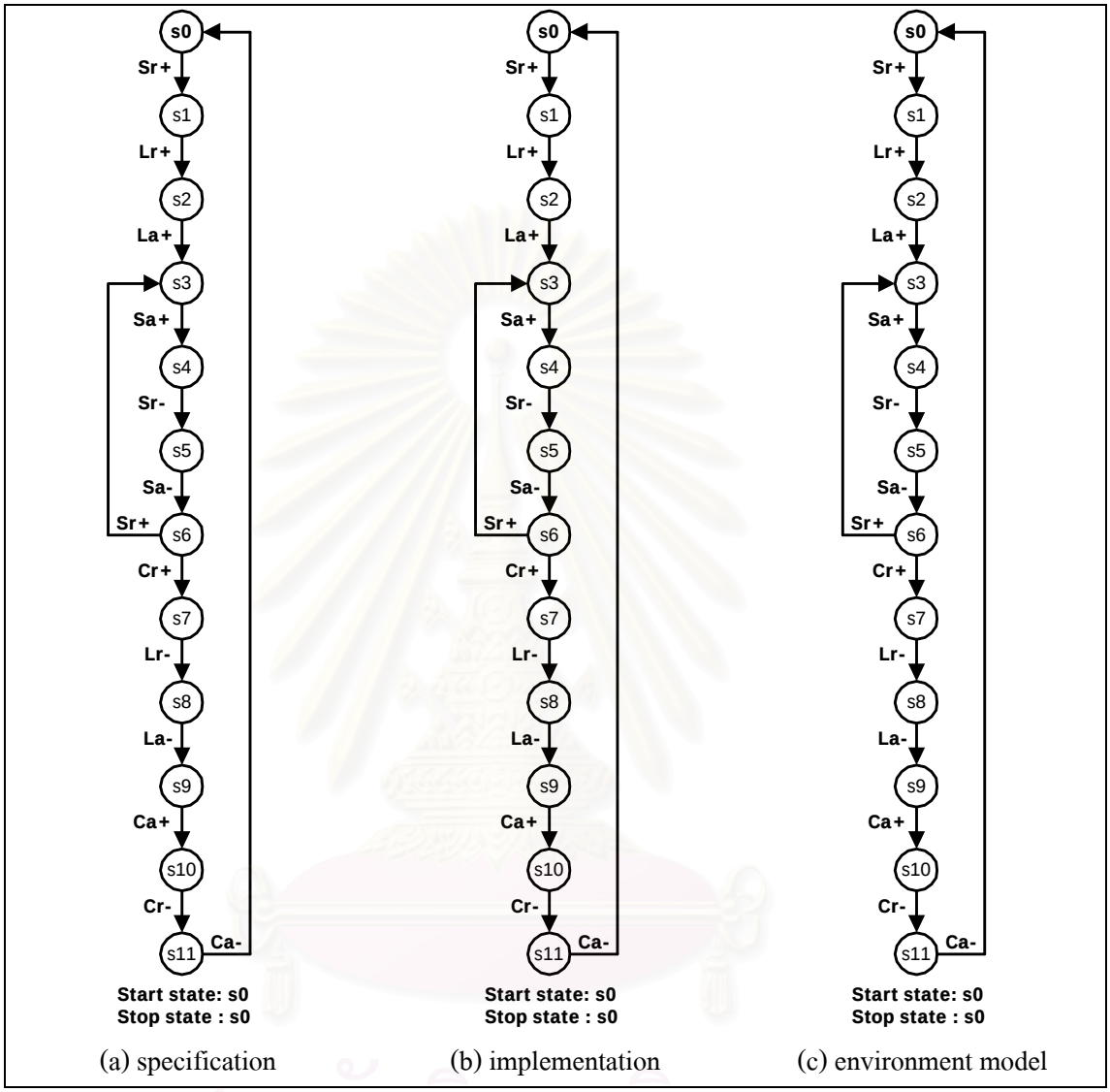
sendr-done



(a) specification (b) implementation (c) environment model

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

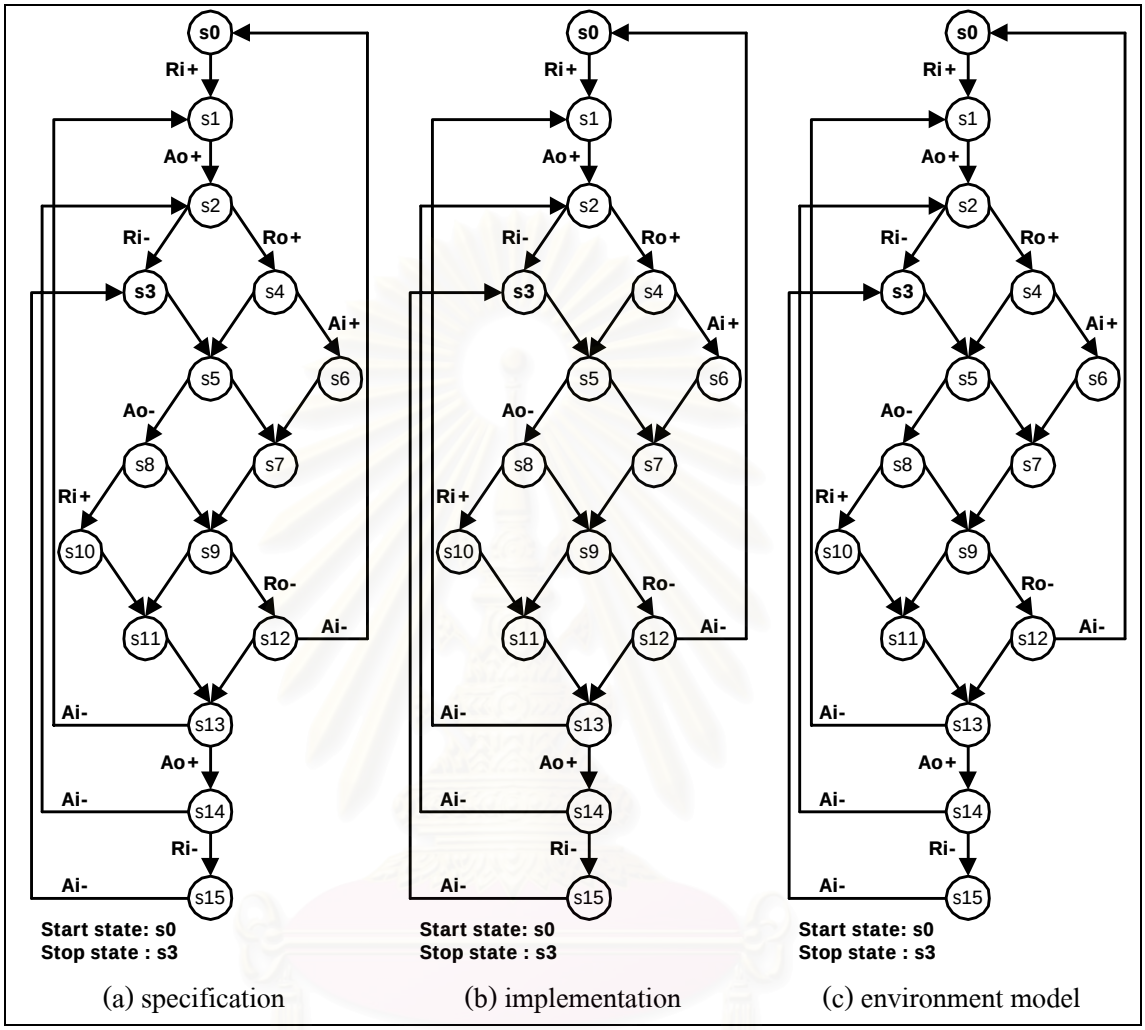
rlm



(a) specification (b) implementation (c) environment model

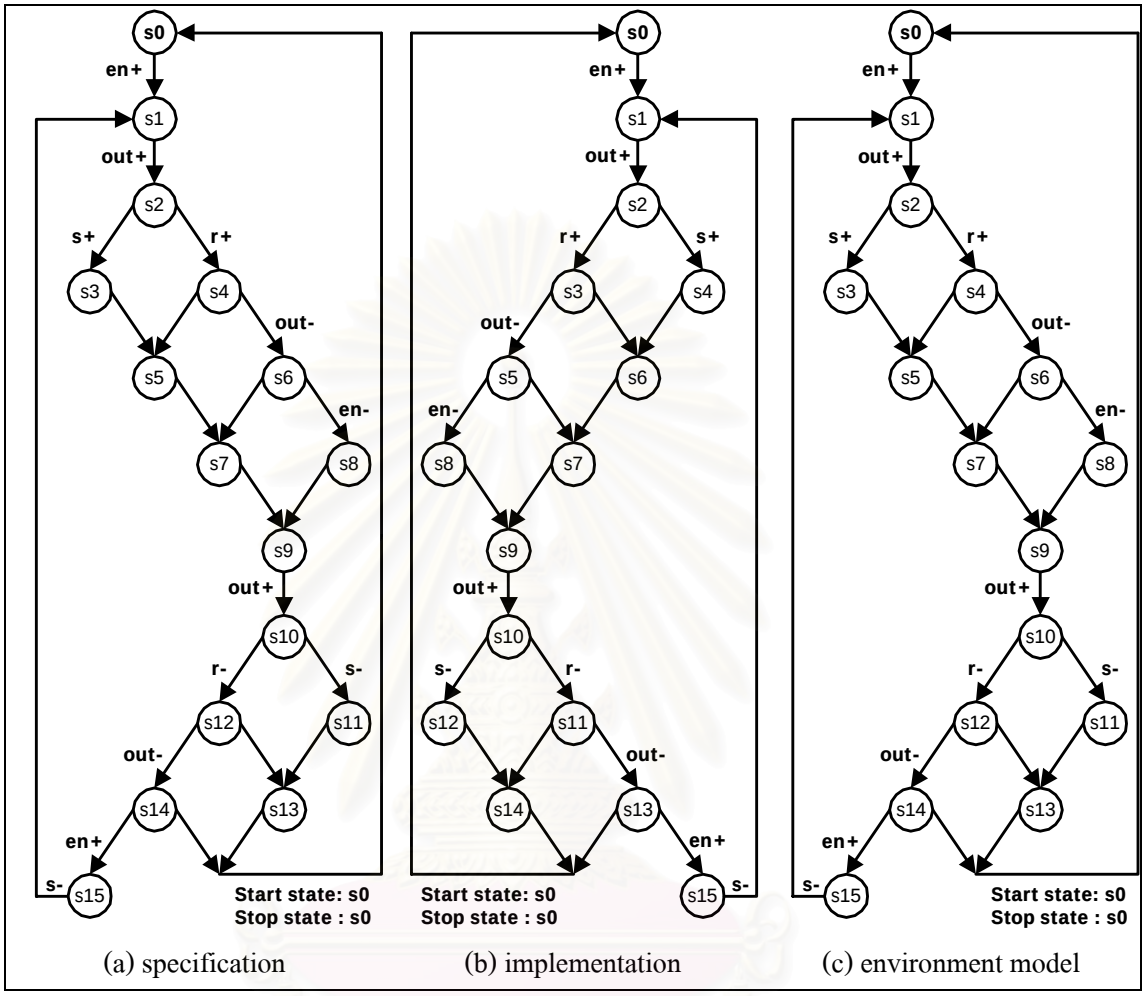
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

full



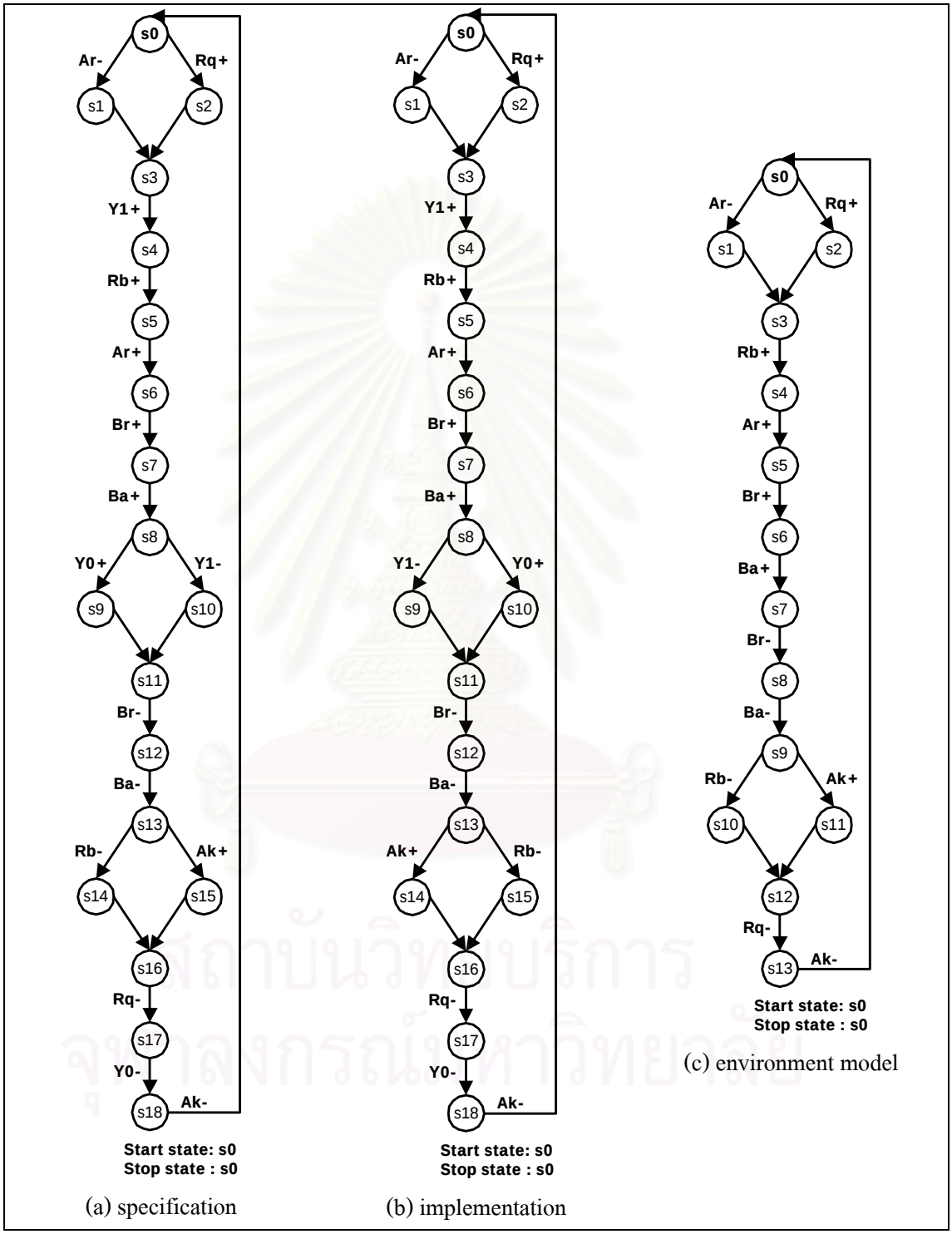
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

input

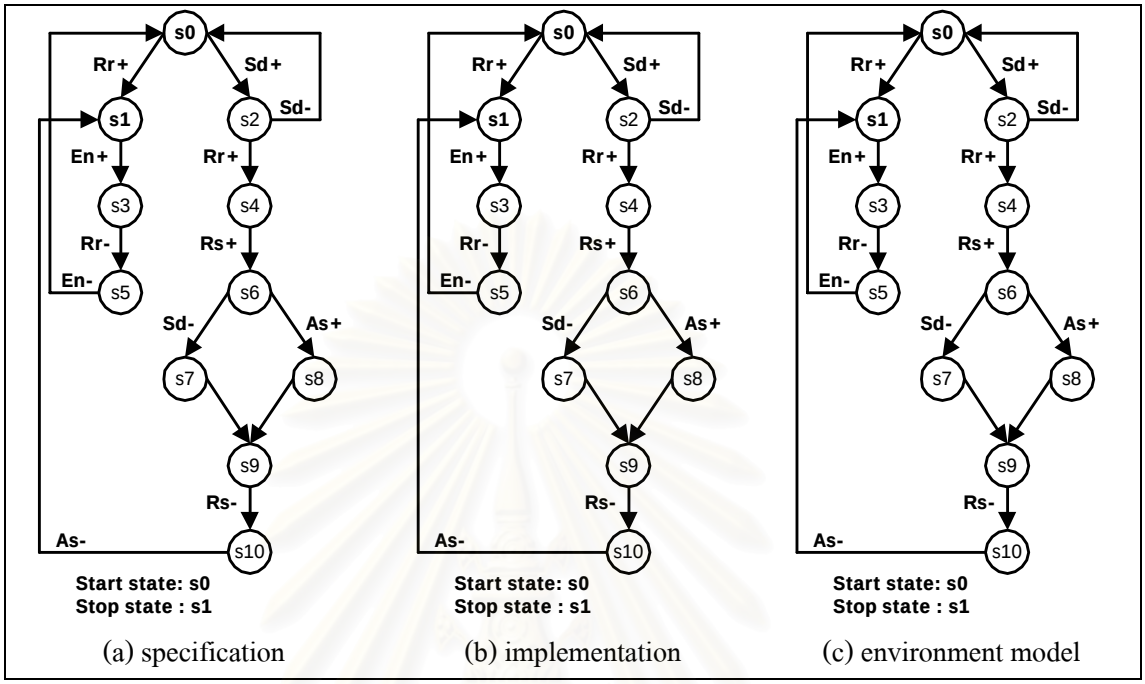


สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

sbuf-read-ctl



rcv-setup



(a) specification (b) implementation (c) environment model

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ประวัติผู้เขียนวิทยานิพนธ์

นายวิฑูรย์ จันทรเศรษฐเลิศ เกิดเมื่อวันที่ 13 เมษายน พ.ศ. 2519 ที่อำเภอสวนหลวง จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาปริญญาตรีวิทยาศาสตรบัณฑิต ภาควิชาคณิตศาสตร์ คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2540 และเข้าศึกษาต่อในหลักสูตรวิทยาศาสตร มหาบัณฑิต สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย เมื่อ พ.ศ. 2541

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย