การทวนสอบวงจรอสมวารด้วยการจำลองลำดับสัญญาณในเอสทีจี

นายวีระศักดิ์ ล่อซุ่นนี้

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

บทคัดย่อและแฟ้มข้อมูลฉบับเต็มของวิทยานิพนธ์ตั้งแต่ปีการศึกษา 2554 ที่ให้บริการในคลังปัญญาจุฬาฯ (CUIR)
เป็นแฟ้มข้อมูลของนิสิตเจ้าของวิทยานิพนธ์ ที่ส่งผ่านทางบัณฑิตวิทยาลัย
The abstract and full text of theses from the academic year 2011 in Chulalongkorn University Intellectual Repository (CUIR)
are the thesis authors' files submitted through the University Graduate School.

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2557
ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

ASYNCHRONOUS CIRCUIT VERIFICATION USING
SIGNAL SEQUENCE SIMLULATION IN STG

Mr. Weerasak Lawsunnee

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science Program in Software Engineering
Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2014
Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	การทวนสอบวงจรอสมวารด้วยการจำลองลำดับสัญญาณในเอสทีจี
โดย	นายวีระศักดิ์ ล่อซุ่นนี้
สาขาวิชา	วิศวกรรมซอฟต์แวร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	รองศาสตราจารย์ ดร.วิวัฒน์ วัฒนาวุฒิ
อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม	ผู้ช่วยศาสตราจารย์ ดร.อาทิตย์ ทองทักษ์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

....คณบดีคณะวิศวกรรมศาสตร์

(ศาสตราจารย์ ดร.บัณฑิต เอื้ออาภรณ์)

คณะกรรมการสอบวิทยานิพนธ์

....ประธานกรรมการ

(รองศาสตราจารย์ ดร.ธาราทิพย์ สุวรรณศาสตร์)

....อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

(รองศาสตราจารย์ ดร.วิวัฒน์ วัฒนาวุฒิ)

....อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม

(ผู้ช่วยศาสตราจารย์ ดร.อาทิตย์ ทองทักษ์)

....กรรมการภายนอกมหาวิทยาลัย

(ดร.เด่นดวง ประดับสุวรรณ)

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

วีระศักดิ์ ล่อซุ่นนี้ : การทวนสอบวงจรอสมวารด้วยการจำลองลำดับสัญญาณในเอสทีจี (ASYNCHRONOUS CIRCUIT VERIFICATION USING SIGNAL SEQUENCE SIMLULATION IN STG) อ.ที่ปรึกษาวิทยานิพนธ์หลัก: รศ. ดร.วิวัฒน์ วัฒนาวุฒิ, อ.ที่ปรึกษาวิทยานิพนธ์ร่วม: ผศ. ดร.อาทิตย์ ทองทักษ์, 101 หน้า.

การทวนสอบวงจรสามารถเลือกทำได้โดยการตรวจสอบพฤติกรรมของวงจรที่เขียนขึ้นด้วยซิกแนลทรานสิชันกราฟหรือที่เรียกว่าเอสทีจีกับวงจรที่ได้จากการสังเคราะห์ วิทยานิพนธ์ฉบับนี้ใช้ประโยชน์จากเทคนิคการจำลองลำดับสัญญาณเพื่อทวนสอบวงจรอสมวารแบบวัฏจักรเชิงเดียวและวัฏจักรที่มีจุดยอดไม่ซ้ำกันเท่านั้น

ประการแรกพฤติกรรมของวงจรที่ต้องการเขียนในรูปแบบของเอสทีจีและวงจรที่ได้จากการสังเคราะห์นั้นจะถูกแปลงให้เป็นภาษาโพรเมล่า จากนั้นสปินจะถูกใช้เป็นเครื่องมือในการจำลองการทำงานรหัสภาษาโพรเมล่าที่แปลงมาได้นั้น และเอาต์พุตที่ได้จากการจำลองจะถูกบันทึกเป็นลำดับการเปลี่ยนแปลงสัญญาณจากการจำลองการทำงานของวงจรเป้าหมายหรือที่เรียกว่าเอสเอส ประการที่สองผู้วิจัยจะค้นหารูปแบบพฤติกรรมที่สำคัญของวงจรที่เรียกว่าลำดับการเปลี่ยนแปลงสัญญาณที่เป็นไปได้ทั้งหมดหรือเรียกสั้นๆ ว่าเอสทีเอส และลำดับความสัมพันธ์เชิงล็อค หรือเรียกสั้นๆ ว่าแอลอาร์เอสจากเอสทีจีที่กำหนด ประการที่สามคุณสมบัติไลฟ์เนสและคุณสมบัติความทนทานจะถูกทวนสอบด้วยวิธีการค้นหารูปแบบพฤติกรรมเอสทีเอสและแอลอาร์เอสออกจากเอสเอส เครื่องมือที่ใช้ในการค้นหาถูกพัฒนาขึ้นเพื่อสนับสนุนการค้นหารูปแบบพฤติกรรมดังกล่าว การออกแบบของวงจรส่วนย่อยซีจะถูกใช้เป็นกรณีศึกษาของงานวิจัยนี้ เราสามารถรายงานผลลัพธ์ความสำเร็จจากการทวนสอบภายในเวลาที่ยอมรับได้ อย่างไรก็ตามข้อจำกัดของเทคนิคการจำลองลำดับสัญญาณนี้ยังไม่เป็นนิยมในการออกแบบวงจรที่มีขนาดใหญ่และมีความซับซ้อน

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ภาควิชา	วิศวกรรมคอมพิวเตอร์	ลายมือชื่อนิสิต	....
สาขาวิชา	วิศวกรรมซอฟต์แวร์	ลายมือชื่อ อ.ที่ปรึกษาหลัก	....
ปีการศึกษา	2557	ลายมือชื่อ อ.ที่ปรึกษาร่วม	....

5471008521 : MAJOR SOFTWARE ENGINEERING
KEYWORDS: VERIFICATION / SPECIFICATION / ASYNCHRONOUS CIRCUITS / SIGNAL TRANSITION GRAPH / SPIN / SIMULATION / SIGNAL TRANSITION SEQUENCE / SIMULATION SEQUENCE / LOCK RELATION / LOCK RELATION SEQUENCE / PERSISTENCE / LIVENESS

WEERASAK LAWSUNNEE: ASYNCHRONOUS CIRCUIT VERIFICATION USING SIGNAL SEQUENCE SIMLULATION IN STG. ADVISOR: ASSOC. PROF. WIWAT VATANAWOOD, Ph.D., CO-ADVISOR: ASSIST. PROF. ARTHIT THONGTAK, Ph.D., 101 pp.

The circuit verification can be alternatively conducted by verifying a circuit behavior, drawn using Signal Transition Graph or called STG, against the circuit implementation. This thesis exploits the Signal Sequence Simulation Technique to the verification of an asynchronous circuit with single and simple cycle only.

Firstly, the expected circuit behavior, in STG, and its implementation are converted in Promela codes. The SPIN is used to simulate these Promela codes and the output of the simulation would be recorded as the Signal Sequence, called SS, of the target circuit. Secondly, we extract two important expected behavioral patterns of the circuit called the Signal Transition Sequence, or STS in short, and the Lock Relation Sequence, or LRS in short, from the given STG. Thirdly, both Liveness and Persistence properties are verified according to the searching STS and LRS behavioral patterns out of the given SS. The searching tools are developed to support the mentioned searching of the behavioral patterns. The C-element circuit designs are used as our case studies. The verification results are successfully reported with the acceptable simulation time. However, the limitation of this signal sequence simulation technique is still not preferable to the huge and complex asynchronous circuit design.

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

Department:	Computer Engineering	Student's Signature	....
Field of Study:	Software Engineering	Advisor's Signature	....
Academic Year:	2014	Co-Advisor's Signature	....

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี เนื่องจากได้รับความกรุณาและความช่วยเหลือเป็นอย่างดียิ่งจาก รองศาสตราจารย์ ดร.วิวัฒน์ วัฒนาวุฒิ อาจารย์ที่ปรึกษาวิทยานิพนธ์หลักและผู้ช่วยศาสตราจารย์ ดร.อาทิตย์ ทองทักษ์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม ซึ่งท่านทั้งสองได้กรุณาให้ความรู้ คำแนะนำ คำปรึกษาและข้อคิดเห็นต่างๆ แก่ผู้วิจัยด้วยความตั้งใจ และเอาใจใส่เป็นอย่างดีมาโดยตลอด จึงขอกราบขอบพระคุณเป็นอย่างสูง ณ ที่นี้ด้วย

ขอขอบพระคุณรองศาสตราจารย์ ดร.ธาราทิพย์ สุวรรณศาสตร์ ประธานกรรมการสอบ และ ดร.เด่นดวง ประดับสุวรรณ กรรมการสอบที่กรุณาสละเวลา ให้คำแนะนำและชี้ให้เห็นถึงข้อบกพร่องต่างๆ ของงานวิจัยนี้

ขอขอบคุณบุคคลที่เกี่ยวข้องทุกท่านที่ได้ให้ความช่วยเหลือ และคอยอำนวยความสะดวกในด้านต่างๆ ทำให้การทำงานวิจัยสำเร็จลุล่วงด้วยดี

ขอขอบคุณบิดา มารดา ที่คอยสนับสนุน เป็นกำลังใจให้แก่ผู้วิจัยเป็นอย่างดีมาโดยตลอดจนงานวิจัยสามารถสำเร็จลุล่วงไปได้ด้วยดี

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

สารบัญ

หน้า

บทคัดย่อภาษาไทย .... ง

บทคัดย่อภาษาอังกฤษ .... จ

กิตติกรรมประกาศ .... ฉ

สารบัญ .... ช

สารบัญรูป .... ฎ

สารบัญตาราง .... ฑ

บทที่ 1 บทนำ 1

1.1 ที่มาและความสำคัญของปัญหา 1

1.2 วัตถุประสงค์งานวิจัย 2

1.3 ขอบเขตงานวิจัย 2

1.4 ขั้นตอนการดำเนินการ 2

1.5 ประโยชน์ที่คาดว่าจะได้รับ 3

1.6 บทความที่ตีพิมพ์จากงานวิจัย 3

1.7 เนื้อหาของวิทยานิพนธ์ 3

บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง 5

2.2 เอสทีจี (Signal Transition Graph) 5

2.1 การทวนสอบวงจร 9

2.3 คุณสมบัติไลฟ์เนส (Liveness) 9

2.4 คุณสมบัติความปลอดภัย (Safety) 10

2.5 คุณสมบัติความทนทาน (Persistency) 10

2.6 คุณสมบัติความต้องกัน (Consistency) 10

2.7 คุณสมบัติการกำหนดสถานะที่ไม่ซ้ำกัน (Unique state assignment) 10

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

หน้า

2.8 ความสัมพันธ์เชิงล็อค (Lock relation) ... 11

2.9 การอธิบายเอสทีจีด้วยแฟ้มข้อความ ... 13

2.10 การทำงานแบบเกิดพร้อมกัน (Concurrent processes) ... 14

2.11 การทำงานแบบมัลติเธรด (Multithread processes) ... 14

2.12 สปิน (SPIN : Simple Promela Interpreter) ... 15

2.13 ภาษาโพรเมล่า ... 17

2.14 งานวิจัยที่เกี่ยวข้อง ... 22

บทที่ 3 การทวนสอบวงจรอสมวารด้วยการจำลองลำดับสัญญาณในเอสทีจี ... 24

3.1 การแปลงพฤติกรรมในเอสทีจีเป็นภาษาโพรเมล่า ... 26

3.2 การแปลงวงจรอสมวารที่ได้จากการสังเคราะห์เป็นภาษาโพรเมล่า ... 31

3.3 การจำลองการทำงานด้วยสปิน ... 43

3.4 การหาเอสทีเอสในเอสทีจี ... 46

3.5 การทวนสอบเอสทีเอส ... 48

3.6 การหาแอลอาร์เอสในเอสทีจี ... 50

3.7 การทวนสอบแอลอาร์เอส ... 50

บทที่ 4 การทวนสอบวงจรอสมวารตัวอย่างด้วยการจำลองลำดับสัญญาณในเอสทีจี ... 56

4.1 การทวนสอบตัวอย่างวงจรอสมวาร full ... 56

4.1.1 การแปลงพฤติกรรมในเอสทีจีและวงจรอสมวารที่สังเคราะห์ได้ของตัวอย่างวงจรอสมวาร full เป็นภาษาโพรเมล่า ... 56

4.1.2 การหาเอสทีเอสในเอสทีจีของตัวอย่างวงจรอสมวาร full ... 57

4.1.3 การทวนสอบเอสทีเอสของตัวอย่างวงจรอสมวาร full ... 57

4.1.4 การหาแอลอาร์เอสในเอสทีจีของตัวอย่างวงจรอสมวาร full ... 58

4.1.5 การทวนสอบแอลอาร์เอสของตัวอย่างวงจรอสมวาร full ... 58

หน้า

4.2 การทวนสอบตัวอย่างวงจรอสมวาร half 63

4.2.1 การแปลงพฤติกรรมในเอสทีจีและวงจรอสมวารที่สังเคราะห์ได้ของตัวอย่างวงจรอสมวาร half เป็นภาษาโพรเมล่า 64

4.2.2 การหาเอสทีเอสในเอสทีจีของตัวอย่างวงจรอสมวาร half 64

4.2.3 การทวนสอบเอสทีเอสของตัวอย่างวงจรอสมวาร half 65

4.2.4 การหาแอลอาร์เอสในเอสทีจีของตัวอย่างวงจรอสมวาร half 66

4.2.5 การทวนสอบแอลอาร์เอสของตัวอย่างวงจรอสมวาร half 66

4.3 การทวนสอบตัวอย่างวงจรอสมวาร C-element แบบพิจารณาค่าที่ไม่ใช่อินพุต 71

4.3.1 การแปลงพฤติกรรมในเอสทีจีและวงจรอสมวารที่สังเคราะห์ได้ของตัวอย่างวงจรอสมวาร C-element เป็นภาษาโพรเมล่า 72

4.3.2 การหาเอสทีเอสในเอสทีจีของตัวอย่างวงจรอสมวาร C-element แบบพิจารณาค่าที่ไม่ใช่อินพุต 72

4.3.3 การทวนสอบเอสทีเอสของตัวอย่างวงจรอสมวาร C-element แบบพิจารณาค่าที่ไม่ใช่อินพุต 73

4.3.4 การหาแอลอาร์เอสในเอสทีจีของตัวอย่างวงจรอสมวาร C-element แบบพิจารณาค่าที่ไม่ใช่อินพุต 75

4.3.5 การทวนสอบแอลอาร์เอสของตัวอย่างวงจรอสมวาร C-element แบบพิจารณาค่าที่ไม่ใช่อินพุต 75

4.4 การทวนสอบตัวอย่างวงจรอสมวาร full ด้วยวงจรที่สังเคราะห์ไม่ถูกต้อง 81

4.5 การทวนสอบตัวอย่างวงจรอสมวาร half ด้วยวงจรที่สังเคราะห์ไม่ถูกต้อง 81

4.6 สรุปผลจากการทวนสอบวงจรอสมวารตัวอย่าง 82

บทที่ 5 สรุปผลงานวิจัยและข้อเสนอแนะ 84

5.1 สรุปผลงานวิจัย 84

5.2 ข้อจำกัด 84

หน้า

5.3 ข้อเสนอแนะ 85
รายการอ้างอิง 86
ภาคผนวก........ 88
ภาคผนวก ก 89
ภาคผนวก ข 93
ประวัติผู้เขียนวิทยานิพนธ์ 101

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

สารบัญรูป

หน้า

รูปที่ 2.1 ตัวอย่างเอสทีจี [1] 5
รูปที่ 2.2 ตัวอย่างเอสทีจีที่มีคุณสมบัติทางเลือกนำเข้า 8
รูปที่ 2.3 ตัวอย่างกราฟเปลี่ยนแปลงระดับสัญญาณที่ไม่มีคุณสมบัติ (a) Liveness (b) Safety (c) Persistency (d) Consistency (e) Unique state assignment และ Single cycle transition 8
รูปที่ 2.4 ตัวอย่างเอสทีจีที่อธิบายโดยคุณสมบัติของความสัมพันธ์เชิงล็อค [6] 12
รูปที่ 2.5 ตัวอย่างเอสทีจี (a) แฟ้มข้อความจากเอสทีจี (b) [5] 13
รูปที่ 2.6 รูปแบบของเธรด [9] 14
รูปที่ 2.7 ส่วนประกอบการทำงานของ SPIN [11] 16
รูปที่ 2.8 การกำหนดพฤติกรรมด้วย proctype และ active proctype 18
รูปที่ 2.9 การประกาศตัวแปรแบบอาร์เรย์ 18
รูปที่ 2.10 โครงสร้างการใช้คำสั่ง atomic 19
รูปที่ 2.11 ตัวอย่างกรณีของแต่ละการเลือกแบบง่าย 19
รูปที่ 2.12 ตัวอย่างกรณีที่เป็นไปตามเงื่อนไขมากกว่า 1 กรณี 20
รูปที่ 2.13 ตัวอย่างกรณีที่เป็นไปได้ทั้ง 2 กรณี 20
รูปที่ 2.14 ตัวอย่างการออกจากการวนซ้ำด้วยคำสั่ง break 20
รูปที่ 2.15 ตัวอย่างการออกจากการวนซ้ำด้วยคำสั่ง goto 21
รูปที่ 2.16 คำสั่งในการตรวจสอบข้อผิดพลาดด้วย SPIN 21
รูปที่ 2.17 การกำหนดโครงสร้างข้อมูลที่ซับซ้อน 21
รูปที่ 2.18 คำหลักของภาษาโพรเมล่า [12] 22
รูปที่ 2.19 วงจรที่ทำให้เป็นผลด้วย C-element [13] 22
รูปที่ 2.20 รหัสต้นฉบับโพรเมล่าที่ได้จากวงจร [13] 23

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

หน้า

รูปที่ 3.1 แผนภาพกิจกรรมขั้นตอนวิธีการทวนสอบที่นำเสนอ (Activity Diagram) 25
รูปที่ 3.2 เอสทีจี (a) แฟ้มข้อความอธิบายเอสทีจี (b) 26
รูปที่ 3.3 proctype MonitorC ที่ได้จากการแปลงแฟ้มข้อความเป็นภาษาโพรเมล่า 29
รูปที่ 3.4 ตัวแปรแบบโกลบอลพร้อมกำหนดค่าเริ่มต้นของสัญญาณ 29
รูปที่ 3.5 ตัวแปรแบบโกลบอลสำหรับนับลำดับการเปลี่ยนแปลงสัญญาณ 29
รูปที่ 3.6 proctype หลังเพิ่มตัวแปรนับในทุกครั้งที่มีการเปลี่ยนแปลงสัญญาณอินพุตของวงจร 30
รูปที่ 3.7 proctype หลังจากการเพิ่มคำสั่ง printf() 30
รูปที่ 3.8 รหัสภาษาโพรเมล่าได้จากการแปลงพฤติกรรมในเอสทีจีตามขั้นตอนที่นำเสนอ 31
รูปที่ 3.9 รูปแบบแฟ้มข้อความนำเข้าของวงจรอสมวารที่ได้จากการสังเคราะห์ 31
รูปที่ 3.10 วงจรอสมวารที่ได้จากการสังเคราะห์ [14] 35
รูปที่ 3.11 ตัวอย่างแฟ้มข้อความนำเข้าของวงจรอสมวารที่ได้จากการสังเคราะห์จากรูปที่ 3.10...... 35
รูปที่ 3.12 proctype ที่แปลงจากเกต Or 37
รูปที่ 3.13 proctype ที่แปลงจากเกต And 37
รูปที่ 3.14 proctype ที่แปลงจากเกต Nand 38
รูปที่ 3.15 proctype ที่แปลงจากเกต Nand 38
รูปที่ 3.16 proctype ที่แปลงจากเกต Nand 39
รูปที่ 3.17 proctype ที่แปลงจากเกต And 39
รูปที่ 3.18 ตัวแปรแบบโกลบอลพร้อมกำหนดค่าเริ่มต้นของสัญญาณ 40
รูปที่ 3.19 proctype AndGate2 หลังเพิ่มตัวแปรนับในทุกครั้งที่มีการเปลี่ยนแปลงสัญญาณ
เอาต์พุตของวงจร 40
รูปที่ 3.20 proctype หลังจากการเพิ่มคำสั่ง printf() 41
รูปที่ 3.21 รหัสโพรเมล่าส่วนพฤติกรรมในเอสทีจีและส่วนของวงจรอสมวารที่ได้จากการ
สังเคราะห์หลังทำการรวมเข้าด้วยกัน 42

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

หน้า

รูปที่ 3.22 โปรแกรมไอสปินสำหรับประมวลผลการจำลองการทำงาน 43

รูปที่ 3.23 ตัวอย่างแฟ้มข้อความจากการจำลองการทำงานด้วยสปิน 43

รูปที่ 3.24 รหัสเทียมในการค้นหาเอสทีเอสในเอสทีจีภายใต้การทำงานแบบวัฏจักรเชิงเดียว
และวัฏจักรที่ไม่มีจุดยอดซ้ำกัน 47

รูปที่ 4.1 เอสทีจี (a) [5] และวงจรที่ได้จากการสังเคราะห์ (b) [14] ของวงจรอสมวาร full 56

รูปที่ 4.2 เอสทีจี (a) [5] และวงจรที่ได้จากการสังเคราะห์ (b) [14] ของวงจรอสมวาร half........ 64

รูปที่ 4.3 เอสทีจี (a) [5] และวงจรที่ได้จากการสังเคราะห์ (b) [14] ของวงจรอสมวาร
C-element แบบพิจารณาค่าที่ไม่ใช่อินพุต 72

รูปที่ 4.4 เอสทีจี (a) [5] และวงจรที่ได้จากการสังเคราะห์ไม่ถูกต้อง (b) ของวงจรอสมวาร full...... 81

รูปที่ 4.5 เอสทีจี (a) [5] และวงจรที่ได้จากการสังเคราะห์ไม่ถูกต้อง (b) ของวงจรอสมวาร half..... 82

รูปที่ ก.1 รหัสภาษาโพรเมล่าวงจร full จากข้อที่ 4.1.1 88

รูปที่ ก.2 รหัสภาษาโพรเมล่าวงจร half จากข้อที่ 4.2.1 89

รูปที่ ก.3 รหัสภาษาโพรเมล่าวงจร C-element แบบสนใจค่าที่ไม่ใช่อินพุต จากข้อที่ 4.3.1 90

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

สารบัญตาราง

หน้า

ตารางที่ 2.1 ชนิดข้อมูลของภาษาโพรเมล่า [12] 18

ตารางที่ 3.1 ตารางแสดงลำดับการเปลี่ยนแปลงสัญญาณจากการอ่านแฟ้มข้อความจากการจำลองการทำงานด้วยสปิน 44

ตารางที่ 3.2 ตารางเอสเอสจากการแบ่งการเปลี่ยนแปลงสัญญาณรอบปัจจุบันและรอบก่อนหน้า.. 46

ตารางที่ 3.3 ผลการทวนสอบเอสทีเอสที่ได้จากข้อที่ 3.4 และจากเอสเอสที่ได้จากข้อที่ 3.3 49

ตารางที่ 3.4 ความสัมพันธ์เชิงล็อคจากเอสทีจี 50

ตารางที่ 3.5 แสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ Ai และ Co ขั้นที่ 1-30 51

ตารางที่ 3.6 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ Bi และ Co ขั้นที่ 1-30 53

ตารางที่ 4.1 ผลการทวนสอบเอสทีเอสกับเอสเอสของตัวอย่างวงจรอสมวาร full 57

ตารางที่ 4.2 ความสัมพันธ์เชิงล็อคจากเอสทีจีของวงจรอสมวาร full 58

ตารางที่ 4.3 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ Ao และ Ro ขั้นที่ 1-30 58

ตารางที่ 4.4 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ Ri และ Ao ขั้นที่ 1-30 60

ตารางที่ 4.5 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ Ro และ Ai ขั้นที่ 1-30 62

ตารางที่ 4.6 ผลการทวนสอบเอสทีเอสกับเอสเอสของตัวอย่างวงจรอสมวาร half 65

ตารางที่ 4.8 ความสัมพันธ์เชิงล็อคจากเอสทีจีของวงจรอสมวาร half 66

ตารางที่ 4.9 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ Ri และ Ao ขั้นที่ 1-30 66

ตารางที่ 4.10 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ Ro และ Ai ขั้นที่ 1-30 68

ตารางที่ 4.11 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ Ro และ Ao ขั้นที่ 1-30 70

ตารางที่ 4.12 ผลการทวนสอบเอสทีเอสกับเอสเอสของตัวอย่างวงจรอสมวาร C-element แบบพิจารณาค่าที่ไม่ใช่อินพุต 73

ตารางที่ 4.13 ความสัมพันธ์เชิงล็อคจากเอสทีจีของวงจรอสมวาร C-element แบบพิจารณาค่าที่ไม่ใช่อินพุต 75

ตารางที่ 4.14 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ P1 และ P4 ขั้นที่ 1-30 76

หน้า

ตารางที่ 4.15 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ P2 และ P4 ขั้นที่ 1-30 78

ตารางที่ 4.16 ตารางสรุปผลการทวนสอบด้วยวิธีการที่นำเสนอกับตัวอย่างวงจรอสมวาร................ 82

ตารางที่ ข-1 ผลการทวนสอบเอสทีเอสกับเอสเอสของตัวอย่างวงจรอสมวาร full จากข้อ 4.1.3..... 92

ตารางที่ ข-2 ผลการทวนสอบเอสทีเอสกับเอสเอสของตัวอย่างวงจรอสมวาร half จากข้อ 4.2.3.... 94

ตารางที่ ข-3 ผลการทวนสอบเอสทีเอสกับเอสเอสของตัวอย่างวงจรอสมวาร C-element แบบสนใจค่าที่ไม่ใช่อินพุต จากข้อที่ 4.3.3 98

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของปัญหา

วงจรอสมวาร (Asynchronous circuits) เป็นวงจรที่ออกแบบได้ยาก เพราะไม่มีสัญญาณนาฬิกาเพื่อควบคุมจังหวะการรับส่งข้อมูลภายในวงจร จึงเป็นสาเหตุให้วงจรทำงานผิดพลาดได้ง่ายในการออกแบบวงจรอสมวารนั้น โดยทั่วไปแล้วนิยมใช้ซิกแนลทรานสิชันกราฟ (STG : Signal Transition Graph) [1] หรือที่เรียกว่าเอสทีจี เป็นเครื่องมือที่ใช้ในการกำหนดคุณลักษณะของวงจร พบว่าในการจำลอง (Simulation) การทำงานของวงจรบางครั้งมีพฤติกรรมการทำงานที่ไม่ถูกต้องตรงกับเอสทีจี

ปัจจุบันมีภาษาที่อธิบายโครงสร้าง (Structure) หรือพฤติกรรม (Behavior) ภาษาของวงจรอสมวารที่นิยมใช้ได้แก่ ภาษาวีเฮชดีแอล (VHDL : VHSIC Hardware Description Language), เวอริล็อก (Verilog Hardware Description Language), ซิสเต็มเวอริล็อก (SystemVerilog) [2] และเครื่องมือในปัจจุบันที่ใช้จำลองการทำงานของวงจรอสมวาร มีทั้งแบบเชิงพาณิชย์ (Commercial) และแบบโอเพนซอร์ส (Open source) ทั้งนี้คุณสมบัติของแต่ละเครื่องมือก็มีแตกต่างกันไป โดยเครื่องมือที่ได้ความนิยมจะเป็นแบบเชิงพาณิชย์ซึ่งมีราคาสูง การใช้งานซับซ้อน และไม่สะดวกเท่าที่ควร

สปิน [3] (SPIN) เป็นเครื่องมือที่ใช้ในการทวนสอบการทำงานของวงจรอสมวารอย่างมีประสิทธิภาพ สามารถใช้ในการจำลองพฤติกรรมการทำงานของระบบ ซึ่งถูกบรรยายด้วยภาษาโพรเมล่า ซึ่งเป็นภาษาที่ใช้ในการบรรยายกระบวนการทำงานแบบเกิดพร้อมกัน (Concurrency) สปินมีความน่าเชื่อถือซึ่งผ่านการใช้งานมาแล้วในหลายโครงการที่สำคัญๆ และเป็นเครื่องมือที่ใช้งานกันอย่างแพร่หลายในการตรวจสอบความถูกต้อง [4] ทั้งนี้สปินยังรองรับการทำงานแบบมัลติเธรด (Multithread) รวมถึงระบบที่มีการทำงานแบบเกิดขึ้นพร้อมกัน (Concurrent system)

ผู้วิจัยศึกษาพบว่าการจำลองการทำงานเชิงโครงสร้างเพื่อทวนสอบความถูกต้องของวงจรไม่ได้พิจารณาถึงความเป็นไปได้ของเส้นทางของการทำงานที่เป็นไปได้ทั้งหมดของวงจรที่นิยมทำกันในปัจจุบัน ดังนั้นในงานวิจัยนี้จึงเลือกเครื่องมือสปินมาใช้ซึ่งเป็นลักษณะวิธีการเชิงรูปนัย (Formal method) แบบโมเดลเช็คกิง (Model checking) นำมาใช้ในการจำลองการทำงานของวงจรและผลที่

ได้จะนำมาใช้ในการทวนสอบด้วยลำดับสัญญาณในเอสทีจีต่อไป ซึ่งเป็นทางเลือกที่สามารถกำหนดขั้นตอนวิธีและพัฒนาเครื่องมือสนับสนุนในการทวนสอบได้ และให้ผลลัพธ์เท่าเทียมกับการจำลองการทำงานด้วยวิธีแบบเดิมเดิม

1.2 วัตถุประสงค์งานวิจัย

เพื่อนำเสนอการทวนสอบวงจรอสมวารด้วยการจำลองลำดับสัญญาณในเอสทีจีด้วยสปิน

1.3 ขอบเขตงานวิจัย

1. ข้อมูลนำเข้าประกอบด้วยเอสทีจีและวงจรอสมวารที่ได้จากการสังเคราะห์ที่ใช้โมเดลความหน่วงแบบไม่ทราบค่า (Unbounded delay model)
2. แปลงพฤติกรรมในเอสทีจีเป็นแฟ้มข้อความตามวิธีการของ Sentovich [5] และแปลงแฟ้มข้อความเป็นภาษาโพรเมล่าด้วยขั้นตอนที่นำเสนอ
3. แปลงวงจรอสมวารที่ได้จากการสังเคราะห์เป็นแฟ้มข้อความรูปแบบรหัสนำเข้าของวงจรอสมวาร และแปลงแฟ้มข้อความของรูปแบบรหัสนำเข้าของของวงจรอสมวารเป็นภาษาโพรเมล่าด้วยขั้นตอนที่นำเสนอ
4. เกตที่ใช้ได้ประกอบด้วย AND, OR, NOT, NAND, NOR และ C-element
5. วงจรที่ศึกษาเป็นการทำงานแบบวัฏจักรเชิงเดียว (Simple cycle) วัฏจักรที่ไม่มีจุดยอดซ้ำกัน (Single cycle) และทำงานด้วยลักษณะไม่มีจุดเลือกอิสระ (Non-free choice)
6. วงจรตัวอย่างที่ใช้ทดสอบเป็นวงจรพื้นฐานของวงจรอสมวาร จำนวน 3 วงจร เป็นอย่างน้อยสำหรับกรณีศึกษา

1.4 ขั้นตอนการดำเนินการ

1. ศึกษาวงจรอสมวารและเอสทีจีรวมถึงทฤษฎีที่เกี่ยวข้อง
2. ศึกษาภาษาโพรเมล่าและเครื่องมือสปิน
3. ทำการแปลงพฤติกรรมในเอสทีจีเป็นภาษาโพรเมล่า
4. ทำการแปลงวงจรอสมวารที่ได้จากการสังเคราะห์เป็นภาษาโพรเมล่า
5. จำลองการทำงานจากรหัสภาษาโพรเมล่าที่แปลงได้ด้วยเครื่องมือสปิน
6. บันทึกลำดับการเปลี่ยนแปลงสัญญาณเป็นแฟ้มข้อความ

7. ค้นหาลำดับการเปลี่ยนแปลงสัญญาณที่เป็นไปได้ทั้งหมดในเอสทีจี
8. ทวนสอบลำดับการเปลี่ยนแปลงสัญญาณที่เป็นไปได้ทั้งหมดในเอสทีจีกับลำดับการเปลี่ยนแปลงสัญญาณจากการจำลองการทำงาน
9. ค้นหาลำดับความสัมพันธ์เชิงล็อคในเอสทีจี
10. พัฒนาเครื่องมือสำหรับแสดงผลลำดับการเปลี่ยนแปลงสัญญาณจากการจำลองการทำงาน
11. ทวนสอบคุณสมบัติความทนทานจากการลำดับความสัมพันธ์เชิงล็อคกับลำดับการเปลี่ยนแปลงสัญญาณจากการจำลองการทำงานที่น้อยที่สุด
12. ทดสอบขั้นตอนวิธีการทวนสอบที่นำเสนอกับตัวอย่างวงจรอสมวารเพื่อสรุปผล
13. สรุปผลงานวิจัย
14. จัดทำบทความวิชาการ และนำเสนอผลงานวิจัย
15. เสนอรายงานการวิจัยในรูปแบบของวิทยานิพนธ์

1.5 ประโยชน์ที่คาดว่าจะได้รับ

สามารถทวนสอบวงจรอสมวารด้วยการจำลองลำดับสัญญาณในเอสทีจีด้วยเครื่องมือสปินได้

1.6 บทความที่ตีพิมพ์จากงานวิจัย

ส่วนหนึ่งของวิทยานิพนธ์ฉบับนี้ ได้รับการตีพิมพ์เป็นบทความทางวิชาการ เรื่อง “Signal Persistence Checking of Asynchronous System Implementation using SPIN” โดย วีระศักดิ์ ล่อซุ่นนี้, วิวัฒน์ วัฒนาวุฒิ และอาทิตย์ ทองทักษ์ ในงานประชุมวิชาการ The International MultiConference of Engineers and Computer Scientists (IMECS 2015) โดยสมาคมวิศวกรระหว่างประเทศ เมื่อวันที่ 18-20 มีนาคม พ.ศ. 2558 ณ เขตบริหารพิเศษฮ่องกงแห่งสาธารณรัฐประชาชนจีน

1.7 เนื้อหาของวิทยานิพนธ์

เนื้อหาของวิทยานิพนธ์ฉบับนี้จะแบ่งออกเป็น 5 บทย่อยๆ โดยบทที่ 1 จะเป็นบทนำจากนั้นในบทที่ 2 จะกล่าวถึงทฤษฎีต่างๆ ที่เกี่ยวข้องกับงานวิจัย บทที่ 3 จะเป็นการอธิบายให้เห็นภาพรวมทั้งหมดของขั้นตอนการทวนสอบวงจรอสมวารด้วยการจำลองลำดับสัญญาณในเอสทีจีที่ได้นำเสนอใน

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

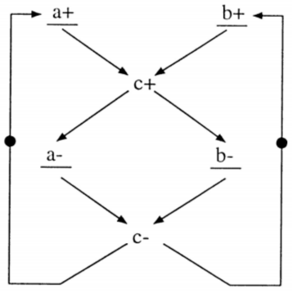
งานวิจัยฉบับนี้ 4 จะเป็นการทวนสอบตัวอย่างวงจรอสมวารด้วยการจำลองลำดับสัญญาณในเอสทีจีจากวิธีการที่ได้นำเสนอ และบทที่ 5 จะเป็นบทสรุปผลงานวิจัยรวมทั้งข้อเสนอแนะซึ่งควรนำปรับปรุงให้ดียิ่งขึ้นต่อไป

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.2 เอสทีจี (Signal Transition Graph)

เอสทีจี [1] เป็นเครื่องมือที่ใช้กำหนดคุณลักษณะของวงจรอสมวาร อย่างมีแบบแผน (Formal specification) ถูกเสนอด้วย Chu เอสทีจีจะมีลักษณะเป็นกราฟแบบมีทิศทาง (Direct graph) เทียบได้กับ Petri net [6] ทั้งนี้วงจรอสมวารที่ใช้งานวิจัยนี้ใช้โมเดลความหน่วงแบบไม่ทราบค่า (Unbounded delay model) หมายถึงการเปลี่ยนแปลงของสัญญาณในวงจรไม่ทราบค่าว่ามีความหน่วงเท่าใดแต่ทราบว่ามีขีดจำกัด ตัวอย่างของเอสทีจีดังรูปที่ 2.1



รูปที่ 2.1 ตัวอย่างเอสทีจี [1]

ส่วนประกอบของเอสทีจีประกอบด้วย

1. เซตของสัญญาณ (Set of signals) เขียนแทนด้วยสัญลักษณ์ J ได้แก่

1.1) เซตของสัญญาณอินพุต (Set of input signals) เขียนแทนด้วยสัญลักษณ์ J_I

1.2) เซตของสัญญาณที่ไม่ใช่สัญญาณอินพุต (Set of non - input signals) เขียนแทนด้วยสัญลักษณ์ J_{NI} จะประกอบด้วย

- เซตของสัญญาณอินพุต (Set of output signals) เขียนแทนด้วยสัญลักษณ์ JO
- เซตของสัญญาณภายใน (Set of internal signals) เขียนแทนด้วยสัญลักษณ์ J_N โดยที่ $J_{NI} = J_O \cup J_N \neq \phi$

เอสทีจีจะแสดงความแตกต่างระหว่างเซตของสัญญาณอินพุตและเซตของสัญญาณที่ไม่ใช่อินพุต ด้วยการขีดเส้นใต้เฉพาะเซตของสัญญาณอินพุต จากรูป J = {a, b, c}, J_I = {a, b} และ J_{NI} = {c}

2. เซตของการเปลี่ยนแปลงสัญญาณ (Set of signal transitions) เขียนด้วยสัญลักษณ์ T โดยที่ T = J x {+,-} สำหรับทุกๆ สัญญาณ j ϵ J และ {+, -} จะเป็นสัญลักษณ์แสดงทิศทางการเปลี่ยนแปลงสัญญาณแบบจาก 0 เป็น 1 (Rising) และการเปลี่ยนแปลงสัญญาณแบบจาก 1 เป็น 0 (Falling) ตามลำดับ

ในแต่ละเอสทีจีจะมีคู่ของการเปลี่ยนแปลงสัญญาณที่สัมพันธ์กันอยู่ได้แก่ j+ และ j– และเราสามารถแบ่งเซตของการเปลี่ยนแปลงสัญญาณโดยพิจารณาให้สอดคล้องกับเซตของสัญญาณได้ดังนี้

2.1 เซตของการเปลี่ยนแปลงสัญญาณอินพุต (Set of output signal transitions) เขียนแทนด้วยสัญลักษณ์ T_I = J_I x {+,-}

2.2 เซตของการเปลี่ยนแปลงสัญญาณเอาต์พุต (Set of output signal transitions) เขียนแทนด้วยสัญลักษณ์ T_O = J_O x {+,-}

2.3 เซตของการเปลี่ยนแปลงสัญญาณภายใน (Set of internal signal transitions) เขียนแทนด้วยสัญลักษณ์ J_N = J_N x {+,-}

สำหรับเซตของการเปลี่ยนแปลงสัญญาณอินพุต จะแสดงด้วยการขีดเส้นใต้ เพื่อแยกให้เห็นความแตกต่างจากเซตของการเปลี่ยนแปลงสัญญาณที่ไม่ใช่สัญญาณอินพุต เช่นเดียวกับเซตของสัญญาณ นอกจากนี้จะมีความแตกต่างระหว่าง การเปลี่ยนแปลงของสัญญาณอินพุตกับการเปลี่ยนแปลงของสัญญาณที่ไม่ใช่สัญญาณอินพุตก็คือ การเปลี่ยนแปลงของสัญญาณอินพุตจะเกิดจากสิ่งแวดล้อมภายนอก (External environment) ของวงจร แต่การเปลี่ยนแปลงของสัญญาณที่ไม่ใช่สัญญาณอินพุต จะเกิดขึ้นจากการทำงานภายในวงจรเอง จากรูปที่ 2.1 T = {a+, a-, b+, b-, c+, c-}, T_I = {a+, a-, b+, b-} และ T_{NI} = {c+, c-}

เอสทีจีจะใช้สัญลักษณ์ T/n เพื่อแสดงลำดับครั้งของการเปลี่ยนแปลง สัญญาณ T ใน เอสทีจีมีสัญลักษณ์ดังนี้ คือ a+/1 และ a+/2 จะหมายถึงสัญญาณ a จะเกิดการเปลี่ยนแปลงจาก 0 เป็น 1 ครั้งที่ 1 และครั้งที่ 2 ตามลำดับ

3. สัญลักษณ์โทเค็น (Token) เขียนแทนด้วยสัญลักษณ์วงกลมทึบสีดำไว้ใช้แสดงสถานะ (State) ของการเปลี่ยนแปลงสัญญาณในเอสทีจี ณ ขณะใดขณะหนึ่ง ทุกๆ เอสทีจี จะต้องมีสัญลักษณ์โทเค็นเริ่มต้น (Initial token) เพื่อแสดงสถานะเริ่มต้นในการทำงานของวงจรและการเปลี่ยนแปลงสถานะของสัญลักษณ์โทเค็นอันเนื่องมาจาก การเปลี่ยนแปลงสัญญาณในเอสทีจี (Firing)

4. คุณสมบัติความสัมพันธ์เกี่ยวกับต้นเหตุ (Casual relation) เขียนแทนด้วยสัญลักษณ์ R ดังนี้ t_1Rt_2 หมายถึงการเปลี่ยนแปลงสัญญาณ t_1 จะทำให้เกิดการเปลี่ยนแปลงของสัญญาณ t_2 สำหรับเอสทีจีจะใช้ลูกศรเป็นสัญลักษณ์เพื่อแสดงความสัมพันธ์ที่เกิดขึ้นดังกล่าว ตัวอย่างเช่นจากรูปที่ 1 a+ ➞ c+ หมายถึง การเปลี่ยนแปลง a จาก 0 เป็น 1 จะมีผลทำให้เกิดการเปลี่ยนแปลงสัญญาณ c จาก 0 เป็น 1 แต่จากรูปที่ 2.1 การเปลี่ยนแปลงสัญญาณ c จาก 0 เป็น 1 จะเกิดขึ้นได้ก็ต่อเมื่อต้องรองให้เกิดการเปลี่ยนแปลงสัญญาณ b จาก 0 เป็น 1 ด้วย

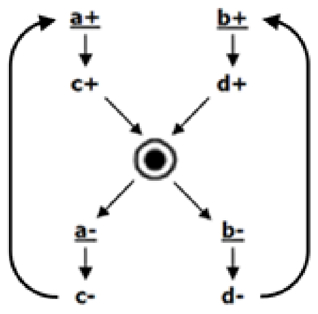
5. คุณสมบัติความสัมพันธ์ชั่วขณะ (Temporal relation) เขียนแทนด้วยสัญลักษณ์ tr สามารถนิยามได้ดังนี้ tr = T x T โดยคุณสมบัติความสัมพันธ์ชั่วขณะ จะเป็นความสัมพันธ์แบบทวิภาค (Binary relation) ของการเปลี่ยนแปลงสัญญาณใดๆ ในเซตของการเปลี่ยนแปลงสัญญาณเอสทีจีสามารถแสดงความสัมพันธ์ระหว่างการเปลี่ยนแปลงสัญญาณใดๆ ได้ทั้งในแบบอันดับ (Order) และแบบพร้อมกัน (Concurrent) ดังตัวอย่างรูปที่ 2.1

- การเปลี่ยนแปลงสัญญาณ a และ b จาก 0 เป็น 1 จะมีความสัมพันธ์พร้อมกัน และใช้เครื่องหมาย “||” แทนความสัมพันธ์แบบพร้อมกัน ดังนั้นจึงสามารถเขียนสัญลักษณ์เพื่อแสดงความสัมพันธ์ที่เกิดขึ้นระหว่างการเปลี่ยนแปลงสัญญาณ a กับ b ได้ดังนี้ a+ || b+ และเซตลำดับของการเปลี่ยนแปลงสัญญาณ (Set of sequence of signal transitions) a กับ b ได้แก่ {a+b+, b+a+}

- การเปลี่ยนแปลงสัญญาณ a กับ c จาก 0 เป็น 1 จะมีความสัมพันธ์แบบอันดับ ดังนั้นเซตลำดับการเปลี่ยนแปลงสัญญาณได้แก่ {a+c+}

ในการกำหนดคุณลักษณะของวงจร นอกจากจะสามารถกำหนดพฤติกรรมแบบพร้อมกันเชิงกำหนด (Deterministic concurrent behavior) เรายังสามารถกำหนดพฤติกรรมแบบพร้อมกันเชิงไม่กำหนดได้อีกด้วย (Non-deterministic concurrent behavior) โดยใช้คุณสมบัติทางเลือกนำเข้าเขียนแทนด้วยสัญลักษณ์สถานที่ (Place) ซึ่งมีลักษณะเป็นวงกลม ดังรูปที่ 2.2 คุณสมบัติทางเลือกนำเข้า (Input choice) หรือเรียกอีกอย่างหนึ่งว่าทางเลือกอิสระ (Free choice) เขียนแทนด้วยสัญลักษณ์สถานที่ ซึ่งหมายถึงเมื่อโทเค็นอยู่ที่สถานที่การเกิดการเปลี่ยนแปลงสัญญาณในเอสทีจีของการเปลี่ยนแปลงสัญญาณที่เป็นเอาต์พุตของสถานที่ จะมีเพียงสัญญาณเดียวเท่านั้นที่ถูกเลือกแบบเชิงไม่กำหนด (Non-deterministic) ในการเปลี่ยนแปลงสัญญาณในเอสทีจี และการ

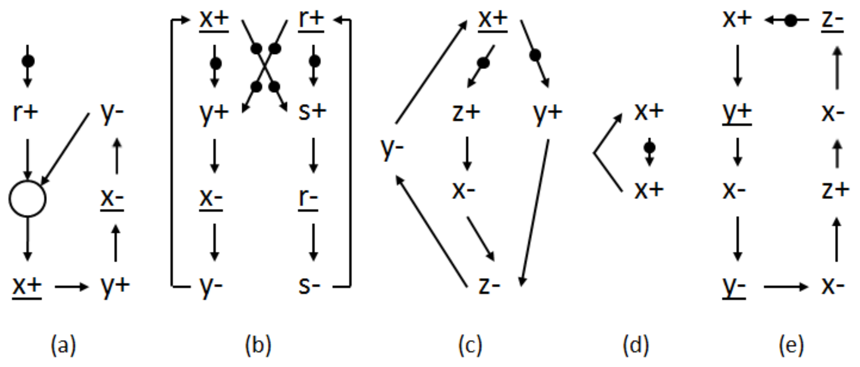
เปลี่ยนแปลงสัญญาณที่เลือกนั้นจะไปปิดทางการเปลี่ยนแปลงสัญญาณอื่นๆ เพื่อไม่ให้เกิดการเปลี่ยนแปลงสัญญาณขึ้น



รูปที่ 2.2 ตัวอย่างเอสทีจีที่มีคุณสมบัติทางเลือกนำเข้า

จากที่กล่าวมาเป็นรูปแบบที่ใช้ในการกำหนดคุณลักษณะของวงจร ดังนั้นจึงสามารถวิเคราะห์พฤติกรรมของเอสทีจี ด้วยเซตลำดับการเปลี่ยนแปลงสัญญาณได้ หลังจากนั้นสามารถนำคุณลักษณะของวงจรดังกล่าวไปสังเคราะห์เป็นวงจรได้ อย่างไรก็ตามในการสังเคราะห์วงจรเพื่อให้ได้วงจรที่มีคุณลักษณะปราศจากการติดตาย (Deadlock free) และปราศจากอันตราย (Hazard free) นั้น กราฟเปลี่ยนแปลงระดับสัญญาณจำเป็นต้องมีคุณสมบัติไลฟ์เนส และคุณสมบัติความทนทาน แต่จะพบว่าการออกแบบวงจรเราจะเกี่ยวข้องกับเอสทีจีเท่านั้น ดังนั้นเพื่อให้เซตลำดับการเปลี่ยนแปลงที่มีคุณสมบัติดังกล่าว จึงต้องพิจารณาคุณสมบัติของเอสทีจีที่สัมพันธ์กับเซตลำดับการเปลี่ยนแปลงด้วย คุณสมบัติดังกล่าวเรียกว่าคุณสมบัติวากยสัมพันธ์ (Syntactic properties) ซึ่งได้แก่คุณสมบัติไลฟ์เนสและคุณสมบัติความทนทาน

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY



รูปที่ 2.3 ตัวอย่างกราฟเปลี่ยนแปลงระดับสัญญาณที่ไม่มีคุณสมบัติ (a) Liveness (b) Safety (c) Persistency (d) Consistency (e) Unique state assignment และ Single cycle transition

2.1 การทวนสอบวงจร

การรวมกันของชิปขนาดใหญ่มาก (VLSI : Very Large Scale Integration) [7] มีราคาแพงมากและใช้เวลานาน ดังนั้นจึงเป็นเรื่องที่สำคัญในการตรวจสอบข้อผิดพลาดของการออกแบบให้มากที่สุดเท่าที่เป็นไปได้ก่อนการผลิต เป็นเรื่องที่เกี่ยวข้องกับความผิดพลาดในการออกแบบพฤติกรรม ตัวอย่างเช่น เป็นสาเหตุของพฤติกรรมที่ไม่ถูกต้องของระบบฮาร์ดแวร์ที่ออกแบบมา

ปัจจุบันการจำลองแบบเดิมเป็นเครื่องมือหลักที่ใช้ในการตรวจสอบข้อผิดพลาด ของการออกแบบพฤติกรรมในช่วงเริ่มต้น อย่างไรก็ตามการจำลองอาจไม่สามารถตรวจพบได้ทุกข้อผิดพลาด ตั้งแต่การจำลองแบบครบถ้วนสมบูรณ์ของวงจรที่ซับซ้อนและระบบที่เป็นไปไม่ได้ เพื่อที่จะเอาชนะปัญหานี้ วิธีการตรวจสอบฮาร์ดแวร์อย่างเป็นทางการได้รับการพัฒนาอยู่ในขณะนี้ และมีแนวโน้มที่จะกลายเป็นเครื่องมือที่มีประโยชน์สำหรับการตรวจสอบข้อผิดพลาดของการออกแบบการทำงาน ตัวอย่างนี้เป็นความตั้งใจที่เป็นภาพรวมบางส่วนของวิธีการที่มีแนวโน้มมากขึ้นที่จะตรวจสอบอย่างเป็นทางการของการออกแบบฮาร์ดแวร์ของความซับซ้อนระดับกลางหรือระดับสูง [7]

แนวความคิดอื่น ๆ ของความถูกต้องที่มีอยู่ บางครั้งมีประโยชน์ในการพิจารณาการออกแบบที่มีอยู่ และการตรวจสอบบางส่วนของคุณสมบัติของวงจร โดยงานวิจัยนี้จัดกลุ่มคุณสมบัติดังกล่าวออกเป็นสองระดับหลักคือ คุณสมบัติความไลฟ์เนส และคุณสมบัติความทนทาน

2.3 คุณสมบัติไลฟ์เนส (Liveness)

คุณสมบัติไลฟ์เนส [7] แสดงเงื่อนไขของรูปแบบ สิ่งที่ดีที่จะเกิดขึ้นในอนาคต ยกตัวอย่างเช่น “สำหรับเส้นทุกทางในอนาคต หากมีการส่งสัญญาณร้องขอไปแล้ว ในที่สุดก็จะมีการตอบรับสัญญาณในการตอบสนองต่อการร้องขออย่างน้อยหนึ่งจุดต่อบนเส้นทาง”

เมื่อพิจารณาจากเอสทีจีทุกๆ จุดที่โทเค็นสามารถไปถึงได้จะต้องสามารถทำให้เกิดการเปลี่ยนแปลงสัญญาณได้อีกเมื่อเกิดการเปลี่ยนแปลงของสัญญาณไปแล้ว ตัวอย่างการเปลี่ยนแปลงระดับสัญญาณจากรูปที่ 2.3(a) จะไม่มีคุณสมบัติไลฟ์เนส เนื่องจากการเปลี่ยนแปลงสัญญาณ r+ จะเกิดขึ้นเพียงครั้งเดียวเท่านั้น

2.4 คุณสมบัติความปลอดภัย (Safety)

คุณสมบัติความปลอดภัย [7] แสดงเงื่อนไขของรูปแบบ สิ่งที่ไม่ดีจะไม่เกิดขึ้น ตัวอย่างของคุณสมบัติความปลอดภัยมีลักษณะดังต่อไปนี้ "สำหรับทุกเส้นทางในอนาคตและทุกจุดต่อ (Node) บนเส้นทาง ถ้าหากสัญญาณร้องขออยู่ในระดับต่ำนั้น มันจะต้องยังคงอยู่ต่ำจนมีการยอมรับสัญญาณไประดับต่ำ"

เมื่อพิจารณาจากเอสทีจีจะไม่มีสถานที่หรือลูกศรใดๆ ในกราฟการเปลี่ยนแปลงระดับสัญญาณสามารถมีโทเค็นได้มากกว่า 1 โทเค็น ตัวอย่างกราฟการเปลี่ยนแปลงระดับสัญญาณจากรูปที่ 2.3 (b) จะไม่มีคุณสมบัติความปลอดภัยเนื่องจาก r+ ➔ y+ สามารถมีโทเค็นได้มากกว่า 1 โทเค็น

2.5 คุณสมบัติความทนทาน (Persistency)

โดยทุกๆ ความสัมพันธ์ a* ➔ b* (t* หมายถึงการเปลี่ยนแปลงของสัญญาณ t+ หรือ t-) ในกราฟเปลี่ยนแปลงระดับสัญญาณ โดยการเปลี่ยนแปลงสัญญาณ b* จะต้องเกิดก่อนการเปลี่ยนแปลงสัญญาณ a* ในทิศตรงกันข้าม ตัวอย่างกราฟเปลี่ยนแปลงระดับสัญญาณจากรูปที่ 2.3 (c) จะไม่มีคุณสมบัติความทนทาน ในส่วนของ x+ ➔ y+ เนื่องจาก x- สามารถเกิดการเปลี่ยนแปลงสัญญาณได้ก่อนที่ y+ จะเกิดการเปลี่ยนแปลงสัญญาณ แต่คุณสมบัตินี้มีข้อยกเว้นสำหรับสัญญาณอินพุตเช่น a* ➔ b* ถึงแม้ว่าสัญญาณ a* ในทิศทางตรงกันข้ามจะต้องเกิดก่อนการเปลี่ยนแปลงสัญญาณ b* กราฟเปลี่ยนแปลงระดับสัญญาณนั้นก็ยังคงมีคุณสมบัติความทนทานอยู่

2.6 คุณสมบัติความต้องกัน (Consistency)

เมื่อมีการเปลี่ยนแปลงของทุกๆ สัญญาณในกราฟเปลี่ยนแปลงระดับสัญญาณ มีการเปลี่ยนแปลงสัญญาณในครั้งต่อไปของสัญญาณนั้นๆ จะต้องเปลี่ยนแปลงในทิศทางตรงกันข้ามเท่านั้น ตัวอย่างกราฟเปลี่ยนแปลงระดับสัญญาณจากรูป 2.3 (d) จะไม่มีคุณสมบัติความต้องกัน เนื่องจากเกิดการเปลี่ยนแปลงของสัญญาณ x+ ต่อกันโดยที่ไม่มีการเปลี่ยนแปลงสัญญาณ x-

2.7 คุณสมบัติการกำหนดสถานะที่ไม่ซ้ำกัน (Unique state assignment)

ทุกๆ จุดในกราฟเปลี่ยนแปลงระดับสัญญาณต้องมีค่าสถานะของสัญญาณสำหรับทุกสัญญาณเพียงค่าเดียวเท่านั้น และค่าสถานะจะต้องไม่ซ้ำกับสัญญาณอื่นๆ ในกราฟเปลี่ยนแปลงระดับสัญญาณ ตัวอย่างกราฟเปลี่ยนแปลงระดับสัญญาณจากรูปที่ 2.3 (e) จะไม่มีคุณสมบัติการกำหนด

CHULALONGKORN UNIVERSITY

สถานะที่ไม่ซ้ำกัน (Unique state assignment) เนื่องจากค่าของสัญญาณในสถานะเริ่มต้นซ้ำกันกับค่าของสัญญาณที่จุด y- → x+ การเปลี่ยนแปลงวัฏจักรเชิงเดียว (Single cycle transitions) หมายถึงการเปลี่ยนแปลงของแต่ละสัญญาณในวัฏจักร (Cycle) ใดของกราฟเปลี่ยนแปลงระดับสัญญาณนั้น จะต้องเกิดการเปลี่ยนแปลงของสัญญาณนั้นๆ 2 ครั้งคือจาก 0 เป็น 1 และจาก 1 เป็น 0 อย่างละครั้งเท่านั้น ตัวอย่างกราฟเปลี่ยนแปลงระดับสัญญาณจากรูปที่ 2.3(e) จะไม่มีคุณสมบัติการเปลี่ยนแปลงวัฏจักร เนื่องจากเกิดการเปลี่ยนแปลงของสัญญาณ x+ และ x- มากกว่า 1 ครั้งในวัฏจักรของกราฟเปลี่ยนแปลงระดับสัญญาณ

2.8 ความสัมพันธ์เชิงล็อค (Lock relation)

ความสัมพันธ์เชิงล็อค [6] ถูกแบ่งออกเป็น 5 ประเภท ดังนี้ ฟูลล็อค, เซมิล็อค, อะโซซีเอทล็อค, แทรนซิทิฟล็อค และซูเปอร์ล็อค โดยที่ความสัมพันธ์เชิงล็อคถูกนำมาตรวจสอบหลังจากของทุกการลดเส้นสัญญาณที่ซ้ำซ้อน ที่เอสทีจีเส้นสัญญาณระหว่างสองสัญญาณใดๆ s* และ t* ถูกลดลงถ้าเอสทีจีนั้นมีคุณสมบัติของวัฏจักรเชิงเดียวกล่าวคือ การทำงานของสัญญาณสามารถสื่อสารกลับมายังจุดเริ่มได้ (s*→ t*→ s*→ t*)

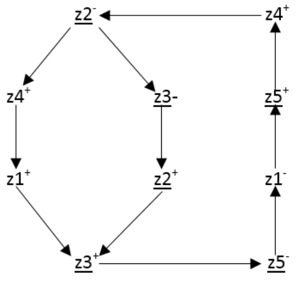
คุณสมบัติของความสัมพันธ์เชิงล็อค

- ฟูลล็อค (Full-lock) ถ้าสองสัญญาณใดๆ a และ b แสดงตามคุณสมบัติดังนี้ $a \rightarrow b \rightarrow \overline{a^*} \rightarrow \overline{b^*}$ บนวัฏจักรเชิงเดียว
- เซมิล็อค (Semi-lock) ถ้าสองสัญญาณใดๆ a และ b แสดงตามคุณสมบัติดังนี้ $a^* \rightarrow b^* \rightarrow \overline{a^*}$ หรือ $b^* \rightarrow a^* \rightarrow \overline{b^*}$ บนวัฏจักรเชิงเดียว
- อะโซซีเอทล็อค (Associate-lock) เมื่อชุดสัญญาณอย่างน้อยที่สุด ชุด A เป็นชุดสัญญาณของฟูลล็อคและสัญญาณ b แสดงตามคุณสมบัติดังนี้ $\exists a1,a2 \in A : a1 \rightarrow b^* \rightarrow a2 \rightarrow \overline{b^*}$ บนวัฏจักรเชิงเดียวดังนั้น A และ b เป็นอะโซซีเอทล็อคและ A U b เป็นแทรนซิทิฟล็อคที่ระดับ 0 (Level-0) เช่นกัน เมื่อ a1 และ a2 เป็นสมาชิกใดๆ ของฟูลล็อค
- ซูเปอร์ล็อค (Super-lock) เมื่อชุดสัญญาณ A เป็นสัญญาณฟูลล็อคและสัญญาณ t เป็นสัญญาณอะโซซีเอทล็อคโดยที่ t* เป็นสัญญาณที่เกิดขึ้นพร้อมกัน (Concurrent) กับชุดของ

การเปลี่ยนแปลงระดับสัญญาณฟูลล็อคของ A ดังนั้น t* เป็นสัญญาณซูเปอร์ล็อคกับชุดสัญญาณ A

- แทรนซิทิฟล็อค (Transitive-lock) มีคุณสมบัติคล้ายกันกับอะโซซีเอทล็อค ซึ่งมีความแตกต่างกันโดยคุณสมบัติของชุดสัญญาณฟูลล็อคจำนวนสมาชิกมากกว่า 2 สัญญาณ ดังนี้ ดังนี้ $\exists a1,a2,a3 \in A : a1 \rightarrow a2 \rightarrow b^* \rightarrow a3 \rightarrow \overline{b^*}$ ดังนั้น A U b เป็น t แทรนซิทิฟล็อคที่ระดับ i+1 (Level (i+1))

ความสัมพันธ์เชิงล็อคสามารถอธิบายคุณสมบัติของวงจรโดยความสัมพันธ์ของแต่ละสัญญาณบนเอสทีจีได้ดังนี้ หากเอสทีจีมีคุณสมบัติของความสัมพันธ์ของฟูลล็อคระหว่างสองสัญญาณใดๆ สามารถบอกได้ว่าพฤติกรรมของวงจรนั้นสามารถใช้สัญญาณการติดต่อสื่อสารรูปแบบ 4 ขั้นโพรโทคอล (Phase protocol) ระหว่างสองสัญญาณ เพื่อทำการหาความสัมพันธ์ของฟูลล็อคสามารถทำการหาสัมพันธ์ของเซมิล็อคก่อนได้ เนื่องจากความสัมพันธ์ของเซมิล็อคเป็นส่วนหนึ่งในความสัมพันธ์ของฟูลล็อค หากพิจารณาแต่ความสัมพันธ์ของฟูลล็อคเพียงอย่างเดียวจะทำให้เอสทีจีถูกจำกัดขอบเขตในการออกแบบ เพราะสัญญาณในชุดความสัมพันธ์ของฟูลล็อคสามารถมีความสัมพันธ์กับสัญญาณอื่นๆ ได้ ซึ่งถูกเรียกว่าความสัมพันธ์ของอะโซซีเอทล็อค และการติดต่อสื่อสารของสัญญาณเป็นรูปแบบ 4 ขั้นเช่นกัน อย่างไรก็ตามหากความสัมพันธ์ของฟูลล็อคระหว่างสัญญาณ ที่มีมากกว่าสองสัญญาณมีความสัมพันธ์ของอะโซซีเอทล็อค ความสัมพันธ์ของสัญญาณดังกล่าวเรียกว่า แทรนซิทิฟล็อคเมื่อชุดความสัมพันธ์ที่มีขนาดใหญ่ มีความสัมพันธ์ของแทรนซิทิฟล็อคร่วมกับทุก สัญญาณที่ไม่ใช่สัญญาณอินพุต (Non-input signal) ในเอสทีจีจะกล่าวได้ว่า เอสทีจีนั้นรองรับคุณสมบัติของการเข้ารหัสสถานะที่สมบูรณ์ (Complete State Coding) และความสัมพันธ์เชิงล็อคสามารถอธิบายในรูปแบบของการเกิดขึ้นพร้อมกันของสัญญาณใดกับชุดสัญญาณฟูลล็อคได้โดยความสัมพันธ์ของซูเปอร์ล็อค

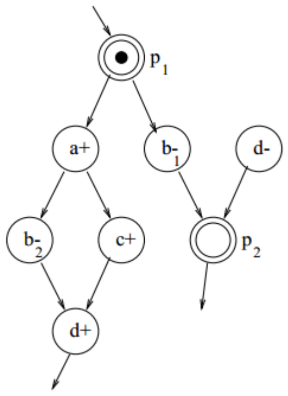


รูปที่ 2.4 ตัวอย่างเอสทีจีที่อธิบายโดยคุณสมบัติของความสัมพันธ์เชิงล็อค [6]

จากรูปที่ 2.4 เอสทีจีมีสัญญาณอินพุตหลัก (Primary input signal) คือ z2, z5 และ z3 และสัญญาณที่ไม่ใช่สัญญาณอินพุตคือ z1และ z4 จากตัวอย่างดังกล่าว ไม่มีคุณสมบัติความทนทาน เนื่องจาก z2- เป็นการเปลี่ยนแปลงทริกเกอร์ (Trigger transition) กับ z4- และ z2+ เป็นแบบเกิดพร้อมกันกับ z4+ สัญญาณ z1 มีความสัมพันธ์เซมิล็อค กับสัญญาณ z2, z3 และ z5 ส่วนสัญญาณ z2 มีความสัมพันธ์เซมิล็อคกับ z1, z3 และ z4 ในความสัมพันธ์ของฟูลล็อคมีสัญญาณสองชุดคือ z1 กับ z5 และ z2 กับ z3 สัญญาณ z4 มีความสัมพันธ์ของอะโซซีเอทล็อคกับ ชุดสัญญาณฟูลล็อค ของ z2 กับ z3 อีกทั้งมีความสัมพันธ์ของซูเปอร์ล็อคเนื่องจาก z4+ เป็นแบบเกิดพร้อมกัน กับสัญญาณ z2+ ของชุดสัญญาณฟูลล็อคของ z2 กับ z3 ในความสัมพันธ์ของแทรนซิทิฟล็อค z1 มีความสัมพันธ์แทรนซิทิฟล็อคกับชุดสัญญาณ z2, z3 และ z4 และ z5 มีความสัมพันธ์แทรนซิทิฟล็อคกับทุกสัญญาณ

2.9 การอธิบายเอสทีจีด้วยแฟ้มข้อความ

งานวิจัย Sentovich [5] ตัวอย่างการเขียนอธิบายเอสทีจีด้วยแฟ้มข้อความ (Text file)




```
.model example
.inputs a b
.outputs c d
.graph
p1 a+ b-/1
a+ b-/2 c+
b-/2 d+
c+ d+
b-/1 p2
d- p2
....
.marking { p1 }
.end
```

(b)

รูปที่ 2.5 ตัวอย่างเอสทีจี (a) แฟ้มข้อความจากเอสทีจี (b) [5]

ตัวอย่างเอสทีจีรูปที่ 2.5(a) สามารถเขียนให้อยู่ในรูปแฟ้มข้อความโดยลักษณะการเขียนมีดังนี้ สามบรรทัดแรก หลังคำอธิบาย .model, .inputs และ .outputs จะอธิบายชื่อและสัญญาณอินพุตและเอาต์พุตตามลำดับ บรรทัดหลังคำอธิบาย .graph จะอธิบายชุดของเส้นเชื่อมต่อ ตัวอย่างจากรูปที่ 2.5(b) คำอธิบาย p1 a+ b-/1 อธิบายได้ว่าเส้นการเชื่อมต่อระหว่าง p1 กับ a+ และ p1 กับ b1- ตามลำดับ ตัวเลือกบรรทัด .marking อธิบายชุดของสถานที่เริ่มต้นจะแยกออกมาและ

อธิบายจุดเริ่มต้นภายในสัญลักษณ์วงเล็บปีกกา ถัดมาปิดคำอธิบายด้วย .end เป็นการสิ้นสุดการเขียนเอสทีจีด้วยแฟ้มข้อความ

2.10 การทำงานแบบเกิดพร้อมกัน (Concurrent processes)

การทำงานแบบเกิดพร้อมกัน [8] เป็นคุณสมบัติของระบบ ซึ่งอาจมีการคำนวณหรือดำเนินการหลายงานพร้อมกัน อาจมีความสัมพันธ์กัน หรืออาจเป็นอิสระไม่ขึ้นต่อกัน ในแต่ละงานที่ดำเนินการ เครื่องมือที่นิยมได้แก่ เอสทีจี

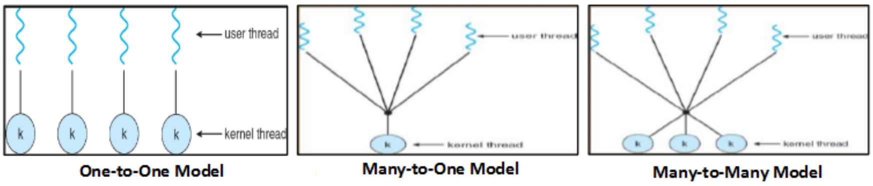
2.11 การทำงานแบบมัลติเธรด (Multithread processes)

เธรด (Thread) คือ หน่วยการทำงานย่อยที่อยู่ในกระบวนการ มีการแบ่งปันทรัพยากรต่างๆ ในกระบวนการนั้นๆ โดยปกติกระบวนการที่มี 1 เธรด จะเรียกว่าซิงเกิลเธรด (Single thread) หรือเรียกว่ากระบวนการที่มีน้ำหนักหนัก (Heavy weight process) กรณี 1 กระบวนการมีมากกว่า 1 เธรด จะเรียกว่ามัลติเธรด (Multithread) [9] หรือเรียกว่ากระบวนการที่มีน้ำหนักเบา (Light weight process)

เธรด คือการเรียกใช้ซีพียู (CPU) ให้เกิดประโยชน์สูงสุด โดยทำให้การทำงานของโปรแกรมง่าย มีประสิทธิภาพมากขึ้น และมีประโยชน์ต่อระบบที่เป็นสถาปัตยกรรมแบบมัลติโพรเซสเซอร์ Multiprocessor Architecture) เพราะสามารถเรียกใช้เธรด หลายๆ ตัวได้พร้อมๆ กัน โดยเธรดแต่ละตัวของกระบวนการเดียวกันจะทำงานแตกต่างกัน แต่มีความเกี่ยวข้องกันบางอย่างและต้องทำงานอยู่ภายใต้สภาพแวดล้อมเดียวกัน

ประโยชน์ของเรียกว่ามัลติเธรด คือ การตอบสนอง (Responsiveness), การใช้ต้นทุนที่มีประสิทธิภาพ (Cost Effective), การกระจายทรัพยากร (Resource Distribution), การกระจายการประมวลผล(Cross-Processor Distribution)

เธรด แบ่งออกเป็น 2 ประเภท คือ เธรดผู้ใช้ (User threads), เธรดเคอร์เนล (Kernel threads) โดยมี 3 รูปแบบ ดังรูปที่ 2.6



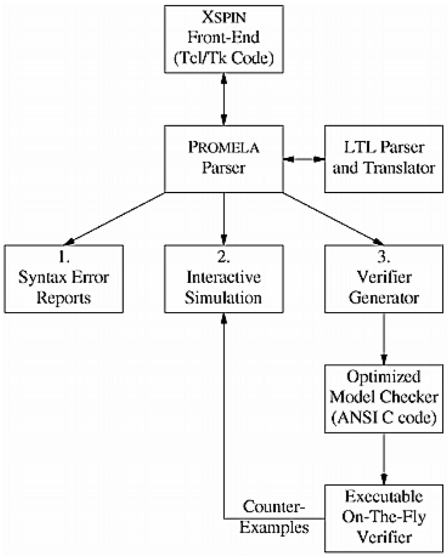
รูปที่ 2.6 รูปแบบของเธรด [9]

2.12 สปิน (SPIN : Simple Promela Interpreter)

สปิน [10] เป็นเครื่องมือสำหรับตรวจสอบโมเดลซอฟต์แวร์ พัฒนาขึ้นโดยทีมซึ่งนำโดย Gerard J. Holzmann ที่เบลล์แล็บ สปินเป็นเครื่องมือตรวจสอบบนพื้นฐานออโตมาตา (การทำงานของเครื่องมือตรวจสอบอาศัยหลักทฤษฎีออโตมาตา) ในการตรวจสอบ, ระบบเป้าหมายสำหรับการตรวจสอบถูกเขียนบรรยายด้วยภาษาโพรเมล่า (PROMELA : Process Meta Language) ซึ่งมีความสามารถในการบรรยายถึง แบบจำลองของอัลกอริทึมการกระจายแบบอสมวาร (Asynchronous distributed algorithms) ในรูปแบบของออโตมาตาที่ไม่สามารถกำหนดได้ (Non-deterministic automata) ส่วนคุณสมบัติที่ต้องการตรวจสอบเขียนระบุด้วยสูตรตรรกศาสตร์เวลาแบบเชิงเส้น (LTL : Linear Temporal Logic) ซึ่งจะถูกนำไปหาผลลบและแปลงไปสู่บุชิออโตมาตา เพื่อนำไปใช้ในกระบวนการตรวจสอบขั้นต่อไป นอกจากจะเป็นเครื่องมือสำหรับตรวจสอบโมเดลแล้ว สปินยังสามารถใช้เพื่อจำลองวิถีการประมวลผลที่เป็นไปได้ของระบบ และแสดงผลเป็นสายของการประมวลผลของระบบได้

สปินแตกต่างไปจากเครื่องมือตรวจสอบโมเดลอื่นๆ คือ ไม่ได้กระทำการตรวจสอบด้วยตัวเอง แต่จะสร้างโค้ดภาษาซี (C programming language) ขึ้นมาเป็นตัวตรวจสอบสำหรับโมเดลเป้าหมายแต่ละโมเดล ซึ่งเทคนิคนี้ช่วยประหยัดหน่วยความจำและทำให้การตรวจสอบมีประสิทธิภาพ นอกจากนั้นยังทำให้การเพิ่มเติม/แก้ไขโมเดล โดยเพิ่มเติม/แก้ไขเฉพาะบางส่วนของโค้ดผลลัพธ์ที่ได้ สปินยังมีทางเลือกที่ช่วยให้ กระบวนการตรวจสอบมีความรวดเร็วมากยิ่งขึ้นและประหยัดหน่วยความจำมากยิ่งขึ้น เช่น การลดลำดับบางส่วน (Partial order reduction), การบีบอัดสถานะ (State compression), การแฮชบิทสถานะ (Bit state hashing), การบังคับความเท่าเทียมแบบหละหลวม (Weak fairness enforcement)

สปินเป็นเครื่องมือที่ใช้ในการตรวจสอบความถูกต้อง และนิยามของแบบจำลองวิธีการเชิงรูปนัย (Formal model) ของระบบซอฟต์แวร์ที่มีการทำงานแบบอสมวาร โดยสามารถแสดงส่วนประกอบดังรูปที่ 2.7



รูปที่ 2.7 ส่วนประกอบการทำงานของ SPIN [11]

- เอกซ์สปิน เครื่องมือพัฒนาโพรเมล่า โมเดล (XSPIN Front-End)

  เป็นส่วนต่อประสานกราฟิกกับผู้ใช้ (GUI) ทำหน้าที่เพื่อช่วยเหลือในการสร้างแบบจำลองฟอร์มอลโมเดลโดยการใช้ภาษาโพรเมล่า

- ส่วนอ่านภาษาโพรเมล่า (PROMELA Parser)

  มีหน้าที่ตรวจสอบอ่านและตรวจสอบโครงสร้างภาษาโพรเมล่า

- ส่วนอ่านและแปลภาษา LTL (LTL Parser and Translator)

  มีหน้าที่อ่านและแปลงตรรกศาสตร์เวลาแบบเชิงเส้น และส่งต่อผลลัทธ์ให้กับ ส่วนอ่านภาษาโพรเมล่า เพื่อนำไปใช้ในการตรวจสอบโพรเมล่าโมเดล จากรูปที่ 2.7 สามารถอธิบายขั้นตอนการทำงานได้ดังต่อไปนี้

1) ทำการตรวจสอบโครงสร้างภาษาโพรเมล่า และรายงานผลการตรวจสอบโครงสร้างโมเดล

2) ระบบจะจำลองการทำงานของโมเดลที่เขียนด้วยภาษาโพรเมล่าตามกระบวนการและเงื่อนไขข้อกำหนดที่ได้ระบุไว้ในโมเดล

3) เมื่อตรวจสอบเรียบร้อยแล้ว สปินจะถูกใช้ในทำการสร้างโปรแกรมตรวจสอบ (Verification program) โดยโปรแกรมตรวจสอบจะมีกระบวนการตรวจสอบตามอาร์กิวเมนต์ (Argument) ที่ได้ระบุไว้ในเวลาคอมไพล์ (Compile time) ถ้าโปรแกรมตรวจพบความผิดปกติ ผู้ใช้สามารถทวนสอบกลับไปจำลองการทำงานตรงส่วนงานที่ ผิดปกติเพื่อแก้ไขต่อไป

2.13 ภาษาโพรเมล่า

ภาษาโพรเมล่า [12] เป็นภาษาแบบจำลองการตรวจสอบ ที่ช่วยสำหรับการสร้างแบบไดนามิกของกระบวนการที่เกิดขึ้นพร้อมกัน (Concurrent processes) สู่โมเดล และภาษาโพรเมล่าสามารถวิเคราะห์ได้ด้วย สปินเพื่อตรวจสอบว่าโมเดลของระบบเป็นไปตามพฤติกรรมที่ต้องการ โดยการสุ่มแบบซ้ำแล้วซ้ำอีก (Iterative simulations) ของการดำเนินการของระบบที่ถูกโมเดล หรือสามารถสร้างโปรแกรมภาษาซีที่ดำเนินการตรวจสอบอย่างละเอียดถี่ถ้วนรวดเร็วบนสภาพพื้นที่ของระบบ (State space) ในระหว่างการจำลองและตรวจสอบ

สปินใช้ตรวจสอบสำหรับกรณีติดตาย (Deadlocks), งานที่ไม่ได้ถูกระบุ (Unspecified receptions), หรือรหัสที่ไม่ได้กระทำการ (Unexecutable code) สามารถใช้ตรวจสอบค่าคงที่ของระบบ สามารถหารอบการดำเนินการที่ไม่เกิดความคืบหน้า (Non-process execution cycles) โดยแต่ละโมเดลตรวจสอบด้วยสปิน ภายใต้ความแตกต่างของสมมติฐานเกี่ยวกับสภาพแวดล้อม ความถูกต้องที่เกิดจากสปิน ทำให้เห็นว่าสามารถนำมาใช้กับการสร้างและตรวจสอบรูปแบบภายหลังได้ทั้งหมด

โปรแกรมภาษาโพลเมล่าประกอบด้วย กระบวนการ (Processes), ช่องข้อความ (Message channels) และตัวแปร (Variables) กระบวนการที่เป็นวัตถุโกลบอล (Global objects) ที่แสดงถึงเอนทิตี (Entities) ที่เกิดขึ้นพร้อมกันของระบบกระจาย (Distributed system) ช่องข้อความและตัวแปรสามารถประกาศเป็นได้ทั้งแบบโกลบอล หรือแบบท้องถิ่นในกระบวนการ กระบวนการที่ระบุพฤติกรรมการทำงาน, ช่องทาง และตัวแปรโกลบอลกำหนดสภาพแวดล้อมที่กระบวนการทำงาน

ตัวอย่างส่วนประกอบของภาษาโพรเมล่า

1. กระบวนการ (Processes)

ใช้ในการกำหนดพฤติกรรมของกระบวนการเท่านั้น คำสั่งนี้ยังไม่สามารถกระทำการได้ ในภาษาโพรเมล่าจะมีเพียง 1 กระบวนการที่เป็นชนิด init ซึ่งต้องถูกประกาศอย่างชัดเจน และกระบวนการใหม่สามารถเกิดขึ้นด้วยคำสั่ง run ตามด้วยอาร์กิวเมนต์และต้องสอดคล้องกับชื่อ proctype ที่สร้างไว้ หรือสามารถกำหนด ให้เริ่มกระทำการตั้งแต่สถานะเริ่มต้นของระบบ กำหนดได้โดยเพิ่มคำสั่ง active ไว้ก่อนชื่อของ proctype ดังรูปที่ 2.8

```
proctype A() {
    byte state;
    state = 3;
}
active proctype B() { .... }
init
{ run A(); }
```

รูปที่ 2.8 การกำหนดพฤติกรรมด้วย proctype และ active proctype

2. ชนิดข้อมูล (Data types)

ตารางที่ 2.1 ชนิดข้อมูลของภาษาโพรเมล่า [12]

Typename	C-equivalent	Macro in limits.h	Typical Range
bit or bool	bit-field	-	0..1
byte	uchar	CHAR_BIT (width in bits)	0..255
short	short	SHRT_MIN..SHRT_MAX	-2^15 - 1 .. 2^15 - 1
int	int	INT_MIN..INT_MAX	-2^31 - 1 .. 2^31 - 1

ตัวแปรสามารถประกาศเป็นตัวแปรแบบอาร์เรย์ได้ แต่ไม่สามารถแจกแจงในคำสั่งสร้างได้ ดังรูปที่ 2.9 กรณีต้องการประกาศตัวแปรแบบอาร์เรย์ที่มากกว่า 1 มิติ สามารถกำหนดทางอ้อมได้ด้วยคำสั่ง typedef

```
int x [10];
x[0] = x[1] + x[2];
int x[3];
x[0] = 1;
x[1] = 2;
x[2] = 3;
```

รูปที่ 2.9 การประกาศตัวแปรแบบอาร์เรย์

3. โครงสร้างออโตมิกซ์ (Atomic construct)

ภายใต้คำสั่ง atomic การทำงานแต่ละชุดคำสั่งที่เป็นลำดับนั้นจะไม่มีการแทรกสอดจากกระบวนการอื่นๆ ดังรูปที่ 2.10

```
atomic
{
    /*  statements; */
}
```

รูปที่ 2.10 โครงสร้างการใช้คำสั่ง atomic

4. โครงสร้างการควบคุมการไหล (Control flow constructs)

แบ่งเป็น 3 ส่วน โครงสร้างการควบคุมการไหล

4.1) การเลือกแต่ละกรณี (Case selection)

ตัวอย่างแรกกรณีการเลือกแต่ละกรณีแบบง่าย ดังรูปที่ 2.11

```
if
:: (a != b) -> option1
:: (a == b) -> option2
fi
```

รูปที่ 2.11 ตัวอย่างกรณีของแต่ละการเลือกแบบง่าย

Chulalongkorn University

ตัวอย่างที่สอง มีกรณีที่เป็นไปตามเงื่อนไขมากกว่า 1 กรณี จะมีเพียง 1 เงื่อนไขที่ถูกเลือก สามารถเป็นไปได้ทั้ง a==true หรือ b=true ได้ทั้งสองกรณี และจะไม่ถูกเลือกพร้อมกันและไม่สามารถกำหนดได้ (Non-deterministically) ว่าจะเป็นกรณีใด แต่ถ้าไม่มีกรณีใดถูกเลือกเลย กระบวนการจะถูกบล็อกจนกว่าจะมีกรณีใดกรณีหนึ่งถูกเลือก (ตรงข้ามกับภาษาโปรแกรมทั่วไปที่หยุดหรือไม่กระทำการใดๆ ถ้าไม่มีกรณีใดถูกเลือกเลย) ดังรูปที่ 2.12

```
if
:: (A == true) -> option1;
:: (B == true) -> option2; /*
May arrive here also if
A==true */
:: else -> fallthrough_option;
fi
```

รูปที่ 2.12 ตัวอย่างกรณีที่เป็นไปตามเงื่อนไขมากกว่า 1 กรณี

ตัวอย่างที่สามกรณีแบบไม่สามารถกำหนดได้คือเป็นได้ทั้ง value=3 หรือ value=4 ดังรูปที่ 2.13

```
if
:: value = 3;
:: value = 4;
fi
```

รูปที่ 2.13 ตัวอย่างกรณีที่เป็นไปได้ทั้ง 2 กรณี

4.2) การวนซ้ำ (Repetition)

มีเพียง 1 ตัวเลือกเท่านั้นที่ถูกเลือกแต่ละเวลา เมื่อตัวเลือกสำเร็จแล้ววิธีการออกจากการวนซ้ำคือการใช้คำสั่ง break ดังรูปที่ 2.14

```
do
:: count = count + 1
:: a = b + 2
:: (count == 0) -> break
od
```

รูปที่ 2.14 ตัวอย่างการออกจากการวนซ้ำด้วยคำสั่ง break

4.3) ข้ามแบบไม่มีเงื่อนไข (Unconditional jumps)

กรณีการออกจากการวนซ้ำด้วยคำสั่ง goto ดังรูปที่ 2.15

```
do
:: count = count + 1
:: a = b + 2
:: (count == 0) -> goto done
od
done:
skip;
```

รูปที่ 2.15 ตัวอย่างการออกจากการวนซ้ำด้วยคำสั่ง gotoการยืนยัน (Assertions)

คำสั่งนี้จะถูกกระทำการเสมอๆ และไม่ส่งผลใดกับกระบวนการ โดยคำสั่งนี้จะใช้ในการตรวจสอบข้อผิดพลาด (error) ขณะการทำการตรวจสอบด้วย SPIN ดังรูปที่ 2.16

```
assert(any_boolean_condition)
```

รูปที่ 2.16 คำสั่งในการตรวจสอบข้อผิดพลาดด้วย SPIN

4.4) โครงสร้างข้อมูลที่ซับซ้อน (Complex data structures)

คำสั่งใช้สำหรับกำหนดชื่อของข้อมูลชนิดใหม่แบบรายการที่ได้กำหนดไว้ โดยชนิดข้อมูลใหม่ที่กำหนดไว้สามารถนำไปประกาศสร้างเป็นตัวแปรได้ ดังรูปที่ 2.17

```
typedef MyStruct
{
    short Field1;
    byte  Field2;
};
MyStruct x;
x.Field1 = 1;
```

รูปที่ 2.17 การกำหนดโครงสร้างข้อมูลที่ซับซ้อน

4.5) ความเห็น (Comments)

ขึ้นต้นด้วยชุดอักษร /* และสิ้นสุดที่ */ โดยอาจซ้อนทับกันได้

4.6) คำหลัก (Keywords)

คำหลักของภาษาโพรเมล่า ดังรูปที่ 2.18

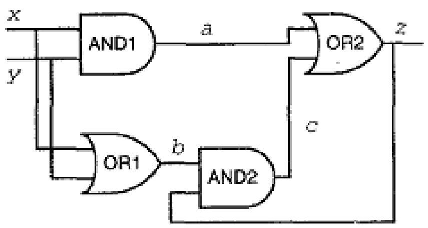
active	assert	atomic	bit
bool	break	byte	chan
d_step	D_proctype	do	aelse
empty	enabled	fi	full
goto	hidden	if	init
int	len	mtype	nempty
never	nfull	od	of
pc_value	printf	priority	proctype
provided	run	short	skip
timeout	typedef	unless	unsigned
xr	xs		

รูปที่ 2.18 คำหลักของภาษาโพรเมล่า [12]

2.14 งานวิจัยที่เกี่ยวข้อง

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

จากงานวิจัยก่อนหน้านี้ B. Rahardjo และ R. D. McLeod [13] พยายามนำเสนอวิธีการตรวจสอบความถูกต้องของโครงสร้างวงจรอสมวารซึ่งถูกทำให้เป็นผลด้วย C-element และตรวจสอบด้วยสปิน



รูปที่ 2.19 วงจรที่ทำให้เป็นผลด้วย C-element [13]

โดยขั้นตอนแรกคือ การอธิบายโครงสร้างของวงจรที่เชื่อมต่อกัน ตัวอย่างวงจรรูปที่ 2.19 ด้วยภาษาโพรเมล่าดังรูปที่ 2.20

```
/* declaration */
bit a, b, c, x, y, z, oldz, wait;
/* macros */
#define AND(x,y,out) (out != (x&&y)) -> out = x&&y
#define   OR(in1,in2,out)   (out!=(in1||in2))->out=
in1||in2
/* main block */
init {
    run stimulus();
    run monitor();
    do
    :: AND(x,y,a) /* AND1 */
    :: OR(x,y,b) /* OR1 */
    :: AND(b,z,c) /* AND2 */
    :: OR(a,c,z) /* OR2 */
    od }
```

```
proctype stimulus() {
    do
    :: wait == 0 ->
        oldz=z;
        if
        :: x = 1-x
        :: y = 1-y
        fi;
        wait = 1
    od }
```

```
proctype monitor() {
    do
    :: wait == 1 ->
        if
        :: (x==0 && y==0 && z==0)
        :: (x==1 && y==1 && z==1)
        :: (x!=y && z==oldz)
        fi;
        wait = 0
    od }
```

รูปที่ 2.20 รหัสต้นฉบับโพรเมล่าที่ได้จากวงจร [13]

จากนั้นสร้างตารางความจริงของ C-element จากนั้นประมวลผลด้วยสปินสังเกตค่าที่เปลี่ยนแปลงไปของแต่ละค่าและบันทึกลงในตารางเพื่อตรวจสอบว่าถูกต้องหรือไม่ ในงานวิจัยพบว่าเกิดข้อผิดพลาดของการเปลี่ยนแปลงค่าเอาต์พุต Z ที่ไม่ถูกต้องสัมพันธ์กับความจริง ดังนั้นผู้เขียนจึงใช้เทคนิคในการบล็อกค่าอินพุตไม่ให้เปลี่ยนแปลง หากวงจรยังไม่ได้ส่งค่าเอาต์พุตออกมา โดยสังเกต proctype ชื่อ Stimulas และ Monitor มีการตรวจสอบเงื่อนไขคือ wait == 0 หรือ wait == 1 ถ้าเงื่อนไขเป็นจริงจึงจะดำเนินการต่อไป

จากงานวิจัยนี้ได้แสดงตัวอย่าง และเทคนิคบางประการในการเขียนคำอธิบายวงจรอสมวารในรูปแบบภาษาโพรเมล่า แต่ในส่วนของรหัสที่อธิบายถึงเกตแต่ละตัวนั้น ผู้วิจัยเห็นว่าชุดรหัสในส่วนการทำงานของเกต AND, OR นั้น เมื่อแยกเขียนเป็น proctype ของแต่ละเกตจะแสดงให้เห็นได้ชัดมากกว่า และงานวิจัยนี้จะเห็นได้ว่าเป็นการอธิบายโครงสร้างของวงจรด้วยภาษาโพรเมล่าเท่านั้น ไม่มีการอธิบายพฤติกรรมของวงจรด้วยเอสทีจี ทำให้การแปลงมาเป็นภาษาโพรเมล่านั้นอาจไม่ตรงตามพฤติกรรมที่ได้ออกแบบไว้

บทที่ 3
การทวนสอบวงจรอสมวารด้วยการจำลองลำดับสัญญาณในเอสทีจี

ในบทนี้จะกล่าวถึงขั้นตอนวิธีการทวนสอบวงจรอสมวารด้วยการจำลองลำดับสัญญาณในเอสทีจีที่นำเสนอ โดยแบ่งเป็น 7 ขั้นตอนดังนี้

1) การแปลงพฤติกรรมในเอสทีจีเป็นภาษาโพรเมล่า

2) การแปลงวงจรอสมวารที่ได้จากการสังเคราะห์เป็นภาษาโพรเมล่า

3) การจำลองการทำงานด้วยสปิน

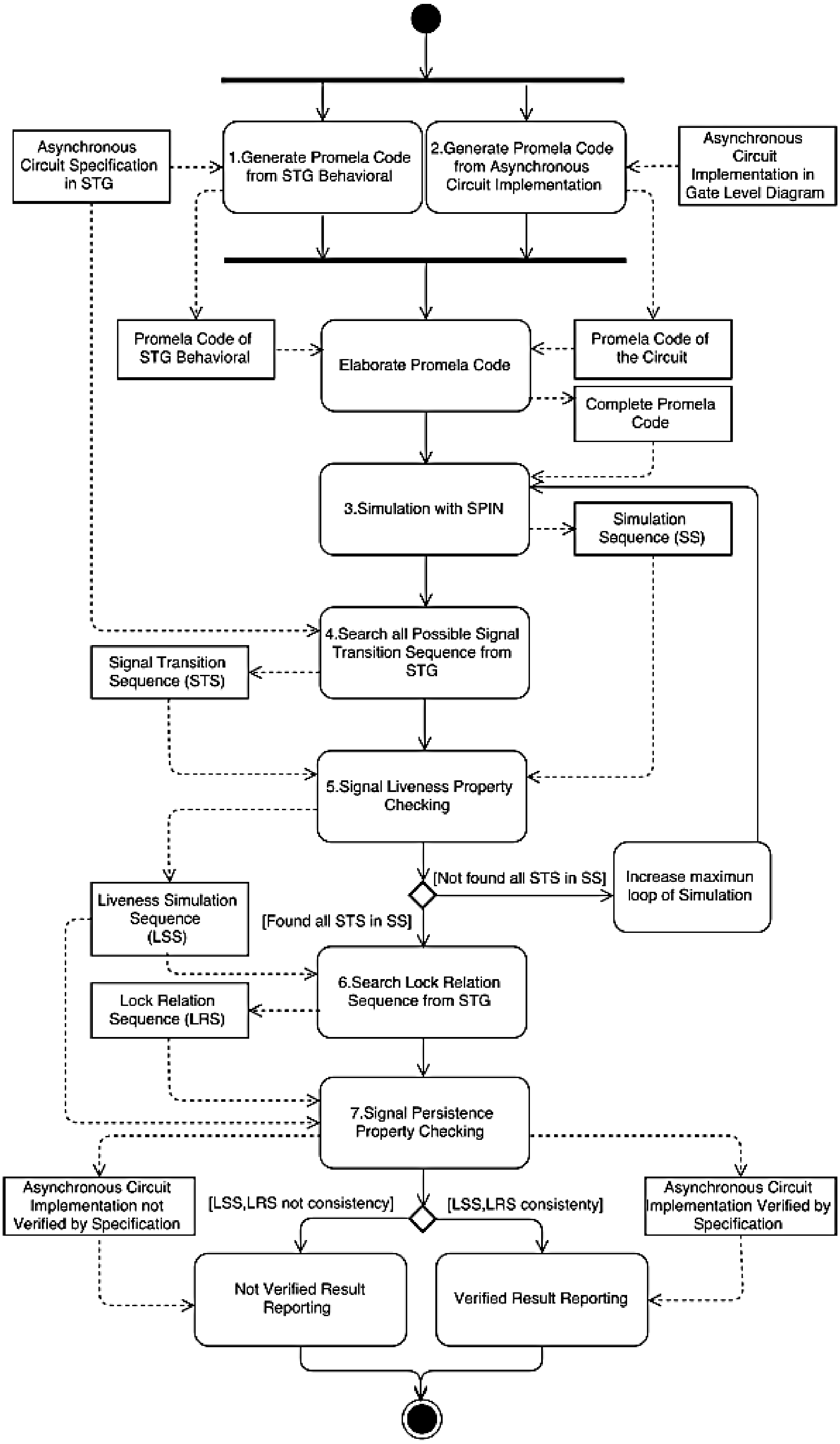
4) การหาลำดับการเปลี่ยนแปลงสัญญาณที่เป็นไปได้ทั้งหมดในเอสทีจี

5) การทวนสอบลำดับการเปลี่ยนแปลงสัญญาณที่เป็นไปได้ทั้งหมด

6) การหาลำดับความสัมพันธ์เชิงล็อคในเอสทีจี

7) การทวนสอบลำดับความสัมพันธ์เชิงล็อค

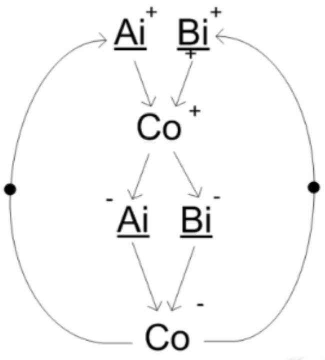
ขั้นตอนที่ 1 การแปลงพฤติกรรมในเอสทีจีเป็นภาษาโพรเมล่า ขั้นตอนที่ 2 การแปลงวงจรอสมวารที่ได้จากการสังเคราะห์เป็นภาษาโพรเมล่า เมื่อได้รหัสภาษาโพรเมล่าจาก 2 ขั้นตอนแรกแล้ว นำไปเป็นข้อมูลอินพุตในขั้นตอนที่ 3 ทำการจำลองการทำงานด้วยสปิน ผลลัพธ์ที่ได้คือ ลำดับการเปลี่ยนแปลงสัญญาณของการจำลองการทำงานหรือที่เรียกว่า เอสเอส (SS : Simulation Sequence) ขั้นตอนที่ 4 การหาลำดับการเปลี่ยนแปลงสัญญาณที่เป็นไปได้ทั้งหมดจากเอสทีจีหรือเรียกว่า เอสทีเอส (STS : Signal Transition Sequence) ถัดมาขั้นตอนที่ 5 การทวนสอบลำดับการเปลี่ยนแปลงสัญญาณที่เป็นไปได้ทั้งหมดของเอสทีเอส เป็นขั้นตอนการตรวจสอบคุณสมบัติไลฟ์เนส โดย ทำการตรวจสอบค้นหาทุกเอสทีเอสในเอสเอส เมื่อตรวจสอบพบทุกเอสทีเอสในเอสเอส จะได้ผลลัพธ์ในขั้นตอนนี้คือ ลำดับการเปลี่ยนแปลงสัญญาณจากการจำลองการทำงานที่น้องที่สุดที่แสดงถึงคุณสมบัติไลฟ์เนส หรือเรียกว่า แอลเอสเอส (LSS : Liveness Simulation Sequence) กรณีตรวจสอบไม่ครบทุกเอสทีเอสให้เพิ่มรอบการจำลองการทำงานและตรวจสอบอีกครั้ง ขั้นตอนที่ 6 การหาลำดับความสัมพันธ์เชิงล็อคในเอสทีจี หรือเรียกว่า แอลอาร์เอส (LRS : Lock Relation Sequence) และขั้นตอนที่ 7 การทวนสอบลำดับความสัมพันธ์เชิงล็อค เป็นขั้นตอนการตรวจสอบคุณสมบัติความทนทาน โดยทำการตรวจสอบ แอลเอสเอสกับแอลอาร์เอสว่ามีลำดับการเปลี่ยนแปลงสอดคล้องกันหรือไม่ โดยภาพรวมขั้นตอนวิธีการที่นำเสนอจะแสดงในรูปที่ 3.1



รูปที่ 3.1 แผนภาพกิจกรรมขั้นตอนวิธีการทวนสอบที่นำเสนอ (Activity Diagram)

3.1 การแปลงพฤติกรรมในเอสทีจีเป็นภาษาโพรเมล่า

จากรูปที่ 3.1 ขั้นตอนที่ 1 การแปลงพฤติกรรมในเอสทีจีให้เป็นภาษาโพรเมล่า ในงานวิจัยนี้ใช้วิธีการไม่อัตโนมัติในการแปลงด้วยข้อมูลนำเข้าของขั้นตอนนี้คือเอสทีจี เริ่มจากเขียนเอสทีจีด้วยแฟ้มข้อความ [5] ตามวิธีในข้อ 2.9 จากเอสทีจีรูปที่ 3.2(a) จะได้แฟ้มข้อความอธิบายเอสทีจีดังรูปที่ 3.2(b) จากนั้นแปลงแฟ้มข้อความอธิบายเอสทีจีเป็นภาษาโพรเมล่าด้วยขั้นตอนดังนี้




```
.module SampleSTG
.inputs Ai,Bi
.outputs Co
.graph
Ai+ Co+
Bi+ Co+
Co+ Ai- Bi-
Ai- Co-
Bi- Co-
Co- Ai+ Bi+
.marking { Co- }
.end
```

(a) (b)

รูปที่ 3.2 เอสทีจี (a) แฟ้มข้อความอธิบายเอสทีจี (b)

การแปลงแฟ้มข้อความอธิบายเอสทีจีเป็นภาษาโพรเมล่า มี 7 ขั้นตอน

1. ค้นหาบรรทัดที่ 2 คำอธิบาย .inputs และบรรทัดที่ 3 .outputs เพื่อทราบถึงแต่ละสัญญาณอินพุตและสัญญาณเอาต์พุต
2. ค้นหาบรรทัดหลังคำอธิบาย .graph เฉพาะบรรทัดที่ขึ้นต้นด้วยสัญญาณเอาต์พุต
3. แต่ละบรรทัดจากข้อ 2 ค้นหาจุดเชื่อมต่อสัญญาณถัดไปในบรรทัดนั้นๆ เฉพาะสัญญาณอินพุต เพื่อทราบถึงผลจากการเปลี่ยนแปลงสัญญาณเอาต์พุตที่ส่งผลให้เกิดการเปลี่ยนค่าที่สัญญาณอินพุต
4. สร้าง proctype แบบ active ด้วยภาษาโพรเมล่า 1 proctype โดย
   4.1 ใน proctype ประกอบด้วยคำสั่งการวนซ้ำ (do…od)
   4.2 ภายในคำสั่งวนซ้ำประกอบด้วยแต่ละเงื่อนไข เมื่อมีการเปลี่ยนแปลงสัญญาณเอาต์พุตในกรณีเป็นค่าบวกแทนค่าด้วย 1 และกรณีเป็นค่าลบแทนค่าด้วย 0
   4.3 แต่ละเงื่อนไขจากข้อ 4.2 ให้อยู่ภายใต้คำสั่ง atomic เพื่อป้องกันการแทรกสอดจาก proctype อื่นขณะจำลองการทำงาน

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

4.4 แต่ละเงื่อนไขจากข้อ 4.2 จะส่งผลให้เกิดการเปลี่ยนแปลงค่าของสัญญาณอินพุตที่เชื่อมต่อกับสัญญาณเอาต์พุตนั้นๆ

4.5 จากข้อ 4.4 กรณีมีการเปลี่ยนแปลงค่าของสัญญาณอินพุตมากกว่า 1 สัญญาณอินพุต ลักษณะการเปลี่ยนแปลงสัญญาณอินพุตต้องเป็นแบบไม่สามารถกำหนดได้ (Non-deterministically) ด้วยคำสั่ง if

4.6 เงื่อนไขใน if จากข้อ 4.5 ให้เพิ่มกรณีที่ไม่ตรงกับเงื่อนไขใดๆ ด้วยคำสั่ง skip และก่อนการเปลี่ยนแปลงที่สัญญาณอินพุตต้องตรวจสอบค่าของสัญญาณอินพุตที่จะเปลี่ยนแปลงไป ต้องไม่ใช่ค่าเดิมก่อนการเปลี่ยนแปลง

5. ค้นหาบรรทัดคำอธิบาย .marking ตรวจสอบในวงเล็บปีกกา จะระบุสถานที่เริ่มต้นของโทเค็น นำมาเพื่อกำหนดค่าเริ่มต้นของแต่ละสัญญาณโดย

5.1 ค้นหาบรรทัดหลังคำอธิบาย .graph ที่ขึ้นต้นด้วยสัญญาณที่ตรงกับสถานที่เริ่มต้นของโทเค็นที่ระบุในวงเล็บปีกกาหลังคำอธิบาย .marking

5.2 ในบรรทัดที่ได้จากข้อ 5.1 แต่ละสัญญาณถัดมาหลังสัญญาณที่ขึ้นต้นของบรรทัดให้กลับค่าของแต่ละสัญญาณ จากค่าบวกเป็นค่าลบหรือจากค่าลบเป็นค่าบวก จะได้ค่าเริ่มต้นของแต่ละสัญญาณ

5.3 ประกาศตัวแปรแบบโกลบอลพร้อมกำหนดค่าเริ่มต้นจากข้อ 5.2

6. ประกาศตัวแปรแบบโกลบอลสำหรับนับลำดับของการเปลี่ยนแปลงสัญญาณและเพิ่มตัวแปรนับในทุกครั้งที่มีการเปลี่ยนแปลงค่าของสัญญาณอินพุตและสัญญาณเอาต์พุต

7. เพิ่มคำสั่ง printf(“\n Output > |%d| N1=|%d|, N2=|%d|,..., Nn=|%d| \n”, Counter, N1, N2,..., Nn) เพื่อติดตามทุกครั้งที่มีการเปลี่ยนแปลงค่าของสัญญาณอินพุตและสัญญาณเอาต์พุต โดยคำสั่ง printf อธิบายได้ดังนี้คือ

- ภายใต้เครื่องหมายอัญประกาศคำสั่ง \n คือ การขึ้นบรรทัดใหม่เวลาแสดงผล
- ภายใต้เครื่องหมายอัญประกาศคำสั่ง |%d| คือ พารามิเตอร์ซึ่งจะแสดงค่าตามตัวแปรที่ต้องการ โดยกำหนดตามลำดับหลังเครื่องหมายอัญประกาศ ตัวอย่าง เช่น printf(“Varible1=|%d|”, V_1); กรณี V_1 มีค่าเท่ากับ 1 เมื่อแสดงผล จะมีผลลัพธ์เป็น Varible1=1 เป็นต้น

- รูปแบบที่กำหนดจะขึ้นต้นบรรทัดใหม่ ตามด้วยตัวแปรนับแสดงลำดับการเปลี่ยนแปลงสัญญาณเป็นลำดับแรก ถัดมาเป็นสัญญาณอินพุตและสัญญาณเอาต์พุตทั้งหมด

ตัวอย่างการแปลงแฟ้มข้อความอธิบายเอสทีจีเป็นภาษาโพรเมล่าจากแฟ้มข้อความอธิบายเอสทีจีจากรูปที่ 3.2(b) ด้วย 7 ขั้นตอนที่ได้นำเสนอ

ขั้นที่ 1 ค้นหาสัญญาณอินพุตและเอาต์พุตจากบรรทัดที่ 2 .inputs Ai,Bi และ บรรทัดที่ 3 .outputs Co สัญญาณอินพุตคือ Ai, Bi และเอาต์พุตคือ Co

ขั้นที่ 2 ค้นหาบรรทัดหลังคำอธิบาย .graph เฉพาะบรรทัดที่ขึ้นต้นด้วยสัญญาณเอาต์พุต ค้นหาพบ 2 บรรทัด Co- Ai+ Bi+ และ Co+ Ai- Bi-

ขั้นที่ 3 แต่ละบรรทัดจากข้อ 2 ค้นหาจุดเชื่อมต่อสัญญาณถัดไปในบรรทัดนั้นๆ เฉพาะสัญญาณอินพุต เพื่อทราบถึงผลจากการเปลี่ยนแปลงสัญญาณเอาต์พุตที่จะส่งผลให้เกิดการเปลี่ยนค่าที่สัญญาณอินพุต

จากบรรทัด Co- Ai+ Bi+ สัญญาณอินพุต Ai และ Bi ที่จะเกิดการเปลี่ยนแปลงสัญญาณเป็นบวก หลังสัญญาณเอาต์พุต Co เกิดการเปลี่ยนแปลงค่าเป็นลบ

จากบรรทัด Co+ Ai- Bi- สัญญาณอินพุต Ai และ Bi ที่จะเกิดการเปลี่ยนแปลงสัญญาณเป็นลบ หลังสัญญาณเอาต์พุต Co เกิดการเปลี่ยนแปลงค่าเป็นบวก

ขั้นที่ 4 สร้าง proctype แบบ active ด้วยภาษาโพรเมล่า 1 proctype โดยใน proctype ประกอบด้วยคำสั่งการวนซ้ำ ภายในคำสั่งวนซ้ำประกอบด้วยแต่ละเงื่อนไข เมื่อมีการเปลี่ยนแปลงสัญญาณเอาต์พุตในกรณีเป็นค่าบวกแทนค่าด้วย 1 และกรณีเป็นค่าลบแทนค่าด้วย 0 แต่ละเงื่อนไขให้อยู่ภายใต้คำสั่ง atomic เพื่อป้องกันการแทรกสอดจาก proctype อื่นขณะจำลองการทำงาน แต่ละเงื่อนไขจะส่งผลให้เกิดการเปลี่ยนแปลงค่าของสัญญาณอินพุตที่เชื่อมต่อกับสัญญาณเอาต์พุตนั้นๆ กรณีมีการเปลี่ยนแปลงค่าของสัญญาณอินพุตมากกว่า 1 สัญญาณอินพุต ลักษณะการเปลี่ยนแปลงสัญญาณอินพุตต้องเป็นแบบไม่สามารถกำหนดได้ด้วยคำสั่ง if โดยเงื่อนไขใน if ให้เพิ่มกรณีที่ไม่ตรงกับเงื่อนไขใดๆ ด้วยคำสั่ง skip และก่อนการเปลี่ยนแปลงที่สัญญาณอินพุตต้องตรวจสอบค่าของสัญญาณอินพุตที่จะเปลี่ยนแปลงไป ต้องไม่ใช่ค่าเดิมก่อนการเปลี่ยนแปลง รหัสภาษาโพรเมล่า proctype ที่ได้ชื่อ MonitorC ดังรูปที่ 3.3

```
active proctype MonitorC(){
do
:: (Co == 1) ->  atomic {
            if
            :: (Ai == 1) -> { Ai = 0 ;}
            :: (Bi == 1) -> { Bi = 0 ;}
            :: skip
            fi;}
:: (Co == 0) ->  atomic {
            if
            :: (Ai == 0) -> { Ai = 1 ;}
            :: (Bi == 0) -> { Bi = 1 ;}
            :: skip
            fi;}
od;
}
```

รูปที่ 3.3 proctype MonitorC ที่ได้จากการแปลงแฟ้มข้อความเป็นภาษาโพรเมล่า

<u>ขั้นที่ 5</u> ค้นหาบรรทัดคำอธิบาย .marking ตรวจสอบในวงเล็บปีกกา จะระบุสถานที่เริ่มต้นของโทเค็น นำมาเพื่อกำหนดค่าเริ่มต้นของแต่ละสัญญาณโดย ค้นหาบรรทัดหลังคำอธิบาย .graph ที่ขึ้นต้นด้วยสัญญาณที่ตรงกับสถานที่เริ่มต้นของโทเค็นที่ระบุในวงเล็บปีกกาหลังคำอธิบาย .marking เมื่อพบบรรทัดที่ต้องการ พิจาณาแต่ละสัญญาณถัดมาหลังสัญญาณที่ขึ้นต้นของบรรทัดให้กลับค่าของแต่ละสัญญาณ จากค่าบวกเป็นค่าลบหรือจากค่าลบเป็นค่าบวก จะได้ค่าเริ่มต้นของแต่ละสัญญาณ

บรรทัดที่กำหนดสถานที่เริ่มต้นคือ .marking { Co- } และบรรทัดหลังคำอธิบาย .graph ที่ตรงกับสถานที่เริ่มต้น คือ Co- Ai+ Bi+ จากนั้นกลับค่าตรงข้ามของสัญญาณอินพุต Ai และ Bi จะได้ค่าเริ่มต้นของสัญญาณคือ Co=0, Ai=0 และ Bi=0 ประกาศตัวแปรแบบโกลบอลพร้อมกำหนดค่าเริ่มต้นของสัญญาณดังรูปที่ 3.4

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

```
bit Ai=0, Bi=0, Co=0;
```

รูปที่ 3.4 ตัวแปรแบบโกลบอลพร้อมกำหนดค่าเริ่มต้นของสัญญาณ

<u>ขั้นที่ 6</u> ประกาศตัวแปรแบบโกลบอล i สำหรับนับลำดับของการเปลี่ยนแปลงสัญญาณ และเพิ่มในทุกขณะที่มีการเปลี่ยนแปลงสัญญาณใดๆ ดังรูปที่ 3.5 และรูปที่ 3.6

```
int i=0;
```

รูปที่ 3.5 ตัวแปรแบบโกลบอลสำหรับนับลำดับการเปลี่ยนแปลงสัญญาณ

```
active proctype MonitorC(){
do
:: (Co == 1) ->  atomic {
        if
        :: (Ai == 1) -> {
                Ai = 0 ; i = i + 1; }
        :: (Bi == 1) -> {
                Bi = 0 ; i = i + 1; }
        :: skip
        fi;}
:: (Co == 0) ->  atomic {
        if
        :: (Ai == 0) -> {
                Ai = 1 ; i = i + 1; }
        :: (Bi == 0) -> {
                Bi = 1 ; i = i + 1; }
        :: skip
        fi;}
od;
}
```

รูปที่ 3.6 proctype หลังเพิ่มตัวแปรนับในทุกครั้งที่มีการเปลี่ยนแปลงสัญญาณอินพุตของวงจร

<u>ขั้นที่ 7</u> เพิ่มคำสั่ง printf(“\n Output > |%d| N_1=|%d|, N_2=|%d|,..., N_n=|%d| \n”, Counter, N_1, N_2,..., N_n) เพื่อติดตามทุกครั้งที่มีการเปลี่ยนแปลงค่าของสัญญาณอินพุตและสัญญาณเอาต์พุต ดังรูปที่ 3.7

```
active proctype MonitorC(){
do
:: (Co == 1) ->  atomic {
        if
        :: (Ai == 1) -> {
                Ai = 0 ; i = i + 1;
                printf("\n Output >  |%d| Ai=|%d|, Bi=|%d|, Co =|%d| \n", i, Ai,Bi,Co); }
        :: (Bi == 1) -> {
                Bi = 0 ; i = i + 1;
                printf("\n Output >  |%d| Ai=|%d|, Bi=|%d|, Co =|%d| \n", i, Ai,Bi,Co); }
        :: skip
        fi;}
:: (Co == 0) ->  atomic {
        if
        :: (Ai == 0) -> {
                Ai = 1 ; i = i + 1;
                printf("\n Output >  |%d| Ai=|%d|, Bi=|%d|, Co =|%d| \n", i, Ai,Bi,Co); }

        :: (Bi == 0) -> {
                Bi = 1 ; i = i + 1;
                printf("\n Output >  |%d| Ai=|%d|, Bi=|%d|, Co =|%d| \n", i, Ai,Bi,Co); }
        :: skip
        fi;}
od;
}
```

รูปที่ 3.7 proctype หลังจากการเพิ่มคำสั่ง printf()

การแปลงพฤติกรรมในเอสทีจีรูปที่ 3.2(a) เป็นภาษาโพรเมล่า ผลลัพธ์จากขั้นตอนนี้เป็นรหัสภาษาโพรเมล่าดังรูปที่ 3.8

```
bit Ai=0, Bi=0, Co=0;
int i=0;

active proctype MonitorC(){
do
:: (Co == 1) ->  atomic {
        if
        :: (Ai == 1) -> {
                Ai = 0 ; i = i + 1;
                printf("\n Output >  |%d| Ai=|%d|, Bi=|%d|, Co =|%d| \n", i, Ai,Bi,Co); }
        :: (Bi == 1) -> {
                Bi = 0 ; i = i + 1;
                printf("\n Output >  |%d| Ai=|%d|, Bi=|%d|, Co =|%d| \n", i, Ai,Bi,Co); }
        :: skip
        fi;}
:: (Co == 0) ->  atomic {
        if
        :: (Ai == 0) -> {
                Ai = 1 ; i = i + 1;
                printf("\n Output >  |%d| Ai=|%d|, Bi=|%d|, Co =|%d| \n", i, Ai,Bi,Co); }

        :: (Bi == 0) -> {
                Bi = 1 ; i = i + 1;
                printf("\n Output >  |%d| Ai=|%d|, Bi=|%d|, Co =|%d| \n", i, Ai,Bi,Co); }
        :: skip
        fi;}
od;
}
```

รูปที่ 3.8 รหัสภาษาโพรเมล่าได้จากการแปลงพฤติกรรมในเอสทีจีตามขั้นตอนที่นำเสนอ

Chulalongkorn University

3.2 การแปลงวงจรอสมวารที่ได้จากการสังเคราะห์เป็นภาษาโพรเมล่า

จากรูปที่ 3.1 ขั้นตอนที่ 2 การแปลงวงจรอสมวารที่ได้จากการสังเคราะห์เป็นภาษาโพรเมล่า ในงานวิจัยนี้ใช้วิธีการไม่อัตโนมัติในการแปลง ด้วยข้อมูลนำเข้าของขั้นตอนนี้คือวงจรอสมวารที่ได้จากการสังเคราะห์ เริ่มจากเขียนอธิบายวงจรอสมวารที่ได้จากการสังเคราะห์โดยรูปแบบแฟ้มข้อความนำเข้า โดยกำหนดรูปแบบดังรูปที่ 3.9

```
module Name(Input1, Input2, ..., InputN, Output)
input Input1, Input, ..., InputN;
output Output;
wire Wire1, Wire2, ..., WireN;
or(Output/WireN, Input/Wire1, Input/Wire2, ..., Input/WireN);
and(Output/WireN, Input/Wire1, Input/Wire2, ..., Input/WireN);
nand(Output/WireN, Input/Wire1, Input/Wire2, ..., Input/WireN);
nor(Output/WireN, Input/Wire1, Input/Wire2, ..., Input/WireN);
C-element(Output/WireN, Input/Wire1, Input/Wire2, ..., Input/WireN);
endmodule
```

รูปที่ 3.9 รูปแบบแฟ้มข้อความนำเข้าของวงจรอสมวารที่ได้จากการสังเคราะห์

จากรูปที่ 3.9 บรรทัดที่ 1 กำหนดชื่อวงจรหลังคำอธิบาย module พร้อมกำหนดชื่อสัญญาณอินพุตตั้งแต่สัญญาณที่ 1 ถึงสัญญาณที่ N ใดๆ และเอาต์พุตของวงจรเป็นชื่อสัญญาณสุดท้ายภายในสัญลักษณ์วงเล็บเปิดและวงเล็บปิด บรรทัดที่ 2 แสดงสัญญาณอินพุตของวงจรหลังคำอธิบาย Input ตามด้วยชื่อสัญญาณอินพุตทั้งหมดของวงจร บรรทัดที่ 3 แสดงสัญญาณเอาต์พุตของวงจรหลังคำอธิบาย Output ตามด้วยชื่อสัญญาณเอาต์พุต บรรทัดที่ 4 แสดงชื่อสัญญาณเชื่อมต่อภายในวงจรทั้งหมดตั้งแต่สัญญาณเชื่อมต่อภายในวงจรที่ 1 ถึงสัญญาณเชื่อมต่อภายในวงจรที่ N เฉพาะสัญญาณที่ไม่ใช่ทั้งสัญญาณอินพุตและเอาต์พุตของวงจร บรรทัดที่ 5 ถึงบรรทัดที่ 9 แสดงแต่ละเกตของวงจรนั้นๆ ประกอบเริ่มด้วยชื่อของประเภทเกต และชื่อสัญญาณเอาต์พุตหรือสัญญาณเชื่อมต่อใดๆ ที่เป็นเอาต์พุตขึ้นต้นเป็นลำดับแรกตามด้วยชื่อสัญญาณอินพุตหรือสัญญาณเชื่อมต่อใดๆ ที่เป็นอินพุตของเกตนั้นๆ เมื่อครบทุกเกตของวงจรแล้วปิดด้วยคำอธิบาย endmodule เป็นการสิ้นสุดของรูปแบบแฟ้มข้อความนำเข้าของวงจรอสมวารที่ได้จากการสังเคราะห์

เมื่อเขียนอธิบายวงจรด้วยรูปแบบแฟ้มข้อความนำเข้าของวงจรอสมวารที่ได้จากการสังเคราะห์แล้วให้ทำการแปลงเป็นภาษาโพรเมล่าด้วยขั้นตอนดังนี้

การแปลงแฟ้มข้อความนำเข้าของวงจรอสมวารที่ได้จากการสังเคราะห์เป็นภาษาโพรเมล่า มี 6 ขั้นตอน

1. ตรวจสอบบรรทัดที่ 2 คำอธิบาย input, บรรทัดที่ 3 output และบรรทัดที่ 4 wire เพื่อทราบถึงแต่ละสัญญาณอินพุต, สัญญาณเอาต์พุต และสัญญาณเชื่อมต่อภายในวงจร
2. ตรวจสอบบรรทัดถัดมาจากข้อ 1 ถึงบรรทัดก่อนคำอธิบาย endmodule เพื่อทราบถึงเกตทั้งหมดของวงจร
3. แต่ละบรรทัดที่แสดงถึงเกตจากข้อย่อยที่ 2 สร้าง proctype แบบ active ด้วยภาษาโพรเมล่า โดย 1 proctype ต่อ 1 เกต
    3.1 ใน proctype ประกอบด้วยคำสั่งการวนซ้ำ (do...od)
    3.2 ภายในคำสั่งวนซ้ำประกอบด้วยแต่ละเงื่อนไข โดยเงื่อนไขขึ้นอยู่กับประเภทของเกตและสัญญาณอินพุตหรือสัญญาณเชื่อมต่อที่เป็นอินพุตของเกตนั้นๆ โดยเงื่อนไขในแต่ละประเภทของเกตมีดังนี้ เมื่อกำหนดสัญญาณอินพุตหรือสัญญาณเชื่อมต่อที่เป็นอินพุตของเกตด้วย N_1, N_2, ...,N_n

- And เกต เงื่อนไขมี 2 กรณีดังนี้

  - กรณีเอาต์พุตเป็นบวกแทนค่าด้วย 1

    เงื่อนไขคือ (N_1==1 && N_2==1 && … && N_n == 1)

  - กรณีเอาต์พุตเป็นลบแทนค่าด้วย 0

    เงื่อนไขคือ (N_1==0 || N_2== 0 || … || N_n == 0)

- Or เกต เงื่อนไขมี 2 กรณีดังนี้

  - กรณีเอาต์พุตเป็นบวกแทนค่าด้วย 1

    เงื่อนไขคือ (N_1==1 || N_2== 1 || … || N_n == 1)

  - กรณีเอาต์พุตเป็นลบแทนค่าด้วย 0

    เงื่อนไขคือ (N_1==0 && N_2== 0 && … && N_n == 0)

- Not เกต เนื่องจาก Not เกตมีเพียง 1 สัญญาณอินพุตหรือสัญญาณเชื่อมต่อที่เป็นอินพุตของเกตเท่านั้นจึงเขียนเฉพาะ N_1 เพียงสัญญาณเดียว เงื่อนไขมี 2 กรณีดังนี้

  - กรณีเอาต์พุตเป็นบวกแทนค่าด้วย 1 เงื่อนไขคือ (N_1==0)

  - กรณีเอาต์พุตเป็นลบแทนค่าด้วย 0 เงื่อนไขคือ (N_1==1)

- Nand เกตเงื่อนไขมี 2 กรณีดังนี้

  - กรณีเอาต์พุตเป็นบวกแทนค่าด้วย 1

    เงื่อนไขคือ (N_1==0 || N_2== 0 || … || N_n == 0)

  - กรณีเอาต์พุตเป็นลบแทนค่าด้วย 0

    เงื่อนไขคือ (N_1==1 && N_2==1 && … && N_n == 1)

- Nor เกตเงื่อนไขมี 2 กรณีดังนี้

  - กรณีเอาต์พุตเป็นบวกแทนค่าด้วย 1

    เงื่อนไขคือ (N_1==0 && N_2== 0 && … && N_n == 0)

  - กรณีเอาต์พุตเป็นลบแทนค่าด้วย 0

    เงื่อนไขคือ (N_1==1 || N_2== 1 || … || N_n == 1)

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

- C-element เกต เนื่องจาก C-element เกตมีเพียง 2 สัญญาณอินพุตหรือสัญญาณเชื่อมต่อที่เป็นอินพุตของเกตเท่านั้นจึงเขียนเฉพาะ N_1, N_2 เพียง 2 สัญญาณ เงื่อนไขมี 3 กรณีดังนี้
  - กรณีเอาต์พุตเป็นบวกแทนค่าด้วย 1 เงื่อนไขคือ (N_1==1 && N_2== 1)
  - กรณีเอาต์พุตเป็นลบแทนค่าด้วย 0 เงื่อนไขคือ (N_1==0 && N_2== 0)
  - กรณีเอาต์พุตไม่เปลี่ยนแปลง เงื่อนไขคือ (N_1 != N_2)

3.3 จากข้อย่อยที่ 3.2 แต่ละเงื่อนไขตามประเภทของเกตให้อยู่ภายใต้คำสั่ง atomic เพื่อป้องกันการแทรกสอดจาก proctype อื่นขณะจำลองการทำงาน

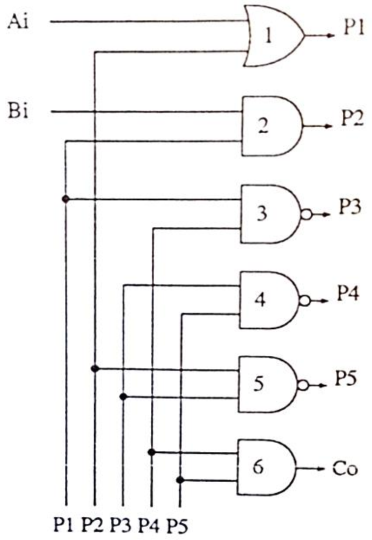
3.4 จากข้อย่อยที่ 3.2 แต่ละเงื่อนไขตามประเภทของเกตที่ได้ กรณีมีตัวผกผัน (Inverter) ที่สัญญาณอินพุตหรือสัญญาณเชื่อมต่อที่เป็นอินพุตของเกตนั้นให้กลับค่าในเงื่อนไขของสัญญาณที่มีตัวผกผันอยู่ จากค่าบวกแทนค่าด้วย 1 เป็นค่าลบแทนค่าด้วย 0 หรือจากค่าลบแทนค่าด้วย 0 เป็นค่าบวกแทนค่าด้วย 1 เสมือนสัญญาณผ่าน Not เกตก่อนสัญญาณเข้าสู่เกตนั้นๆ

3.5 สัญญาณเอาต์พุตที่ได้จากข้อย่อยที่ 3.2, 3.3 เกิดเปลี่ยนแปลงค่าภายใต้คำสั่ง if จำเป็นต้องมีกรณีที่ไม่ตรงกับเงื่อนไขใดๆ ด้วยคำสั่ง skip และก่อนการเปลี่ยนแปลงที่สัญญาณเอาต์พุตต้องตรวจสอบค่าของสัญญาณเอาต์พุตที่จะเปลี่ยนแปลงไป ต้องไม่ใช่ค่าเดิมก่อนการเปลี่ยนแปลง

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

4. ประกาศตัวแปรของเฉพาะสัญญาณเชื่อมต่อภายในวงจรแบบโกลบอล เนื่องจากสัญญาณอินพุตและสัญญาณเอาต์พุตของวงจร มีการประกาศตัวแปรและกำหนดค่าเริ่มต้นของสัญญาณไว้ก่อนหน้าแล้วในขั้นตอนการแปลงพฤติกรรมของเอสทีจีเป็นภาษาโพรเมล่า
5. ตัวแปรแบบโกลบอลสำหรับนับลำดับของการเปลี่ยนแปลงสัญญาณ ถูกประกาศอยู่ในขั้นตอนการแปลงพฤติกรรมของเอสทีจีเป็นภาษาโพรเมล่า ให้เพิ่มตัวแปรนับในทุกครั้งที่มีการเปลี่ยนแปลงค่าของสัญญาณอินพุตและสัญญาณเอาต์พุตของวงจร
6. เพิ่มคำสั่ง printf(“\n Output > |%d| N1=|%d|, N2=|%d|,..., Nn=|%d| \n”, Counter, N1, N2,..., Nn) แสดงผลทุกครั้งที่มีการเปลี่ยนแปลงค่าของสัญญาณอินพุตและสัญญาณเอาต์พุตของวงจร

ตัวอย่างวงจรอสมวาร รูปที่ 3.10 เป็นวงจรที่สังเคราะห์ได้จากเอสทีจีรูปที่ 3.2 (a) ประกอบด้วยสัญญาณอินพุต Ai และ Bi และสัญญาณเอาต์พุต Co และสัญญาณเชื่อมต่อภายในวงจร P1, P2, P3, P4, และ P5 เกตในวงจรมีทั้งหมด 6 เกตประกอบด้วยเกต Or 1 เกต, And 2 เกต และเกต Nand 3 เกต



รูปที่ 3.10 วงจรอสมวารที่ได้จากการสังเคราะห์ [14]

สามารถเขียนอธิบายด้วยรูปแบบแฟ้มข้อความนำเข้าของวงจรอสมวารที่ได้จากการสังเคราะห์ได้ดังรูปที่ 3.11

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

```
module test(Ai,Bi,Co)
input Ai,Bi;
output Co;
wire P1,P2,P3,P4,P5;
or(P1,Ai,P);
and(P2,Bi,P1);
nand(P3,P1,P4);
nand(P4,P3,P5);
nand(P5,P2,P3);
and(Co,P4,P5);
endmodule
```

รูปที่ 3.11 ตัวอย่างแฟ้มข้อความนำเข้าของวงจรอสมวารที่ได้จากการสังเคราะห์จากรูปที่ 3.10

ตัวอย่างการแปลงแฟ้มข้อความนำเข้าของวงจรอสมวารที่ได้จากการสังเคราะห์ เป็นภาษาโพรเมล่าจากแฟ้มข้อความนำเข้าของวงจรอสมวารที่ได้จากการสังเคราะห์จากรูปที่ 3.11 ด้วย 6 ขั้นตอนที่ได้นำเสนอ

<u>ขั้นที่ 1</u> ค้นหาบรรทัดที่ 2 input Ai, Bi, บรรทัดที่ 3 output Co และบรรทัดที่ 4 wire P1, P2, P3, P4, P5 สัญญาณอินพุตคือ Ai และ Bi สัญญาณเอาต์พุตคือ Co และสัญญาณเชื่อมต่อภายในวงจรคือ P2, P3, P4 และ P5

<u>ขั้นที่ 2</u> ค้นหาบรรทัดถัดมาจากขั้นที่ 1 ถึงบรรทัดก่อนคำอธิบาย endmodule เพื่อทราบถึงเกตทั้งหมดของวงจร ค้นหาพบ 6 บรรทัด or(P1,Ai,P), and(P2,Bi,P1), nand(P3,P1,P4), nand(P4,P3,P5), nand(P5,P2,P3) และ and(Co,P4,P5);

<u>ขั้นที่ 3</u> แต่ละบรรทัดที่แสดงถึงเกตจากขั้นที่ 2 สร้าง proctype แบบ active ด้วยภาษาโพรเมล่า โดย 1 proctype ต่อ 1 เกต ใน proctype ประกอบด้วยคำสั่งการวนซ้ำ (do…od) และภายในคำสั่งวนซ้ำประกอบด้วยแต่ละเงื่อนไข ตามประเภทของเกตและสัญญาณอินพุตหรือสัญญาณเชื่อมต่อที่เป็นอินพุตของเกตนั้นๆ แต่ละเงื่อนไขตามประเภทของเกต ให้อยู่ภายใต้คำสั่ง atomic เพื่อป้องกันการแทรกสอดจาก proctype อื่นขณะจำลองการทำงาน กรณีมีตัวผกผัน (Inverter) ที่สัญญาณอินพุตหรือสัญญาณเชื่อมต่อที่เป็นอินพุตของเกตนั้นให้กลับค่าในเงื่อนไขของสัญญาณที่มีตัวผกผันอยู่ จากค่าบวกแทนค่าด้วย 1 เป็นค่าลบแทนค่าด้วย 0 หรือจากค่าลบแทนค่าด้วย 0 เป็นค่าบวกแทนค่าด้วย 1 เสมือนสัญญาณผ่าน Not เกตก่อนสัญญาณเข้าสู่เกตนั้นๆ สัญญาณเอาต์พุตที่ได้จากเกต กำหนดให้เกิดการเปลี่ยนแปลงค่าภายใต้คำสั่ง if และจำเป็นต้องมีกรณีที่ไม่ตรงกับเงื่อนไขใดๆ ด้วยคำสั่ง skip และก่อนการเปลี่ยนแปลงที่สัญญาณเอาต์พุตต้องตรวจสอบค่าของสัญญาณเอาต์พุตที่จะเปลี่ยนแปลงไป ต้องไม่ใช่ค่าเดิมก่อนการเปลี่ยนแปลง

เกต Or(1) บรรทัดที่ 5 or(P1,Ai,P2) สัญญาณอินพุตหรือสัญญาณเชื่อมต่อที่เป็นอินพุตประกอบด้วย สัญญาณอินพุต Ai สัญญาณเชื่อมต่อภายในวงจร P2 และสัญญาณเอาต์พุตคือสัญญาณเชื่อมต่อภายในวงจร P1 แปลงเกตที่กำหนดให้เป็น proctype เมื่อสัญญาณอินพุตของเกต Or เป็นค่าลบทั้ง 2 ค่าแทนค่าด้วย 0 (Ai==0 && P2==0) จะทำให้สัญญาณเอาต์พุตที่ได้เป็นค่าลบแทนค่าด้วย 0 (P1=0) และเมื่อสัญญาณอินพุตของเกต Or ทั้ง 2 ค่ามีค่าใดค่าหนึ่งเป็นค่าบวกแทนค่าด้วย 1 (Ai==1 || P2==1) จะทำให้สัญญาณเอาต์พุตที่ได้เป็นค่าบวกแทนค่าด้วย 1 (P1=1) ดังรูปที่ 3.12

```
active proctype OrGate1(){
do
:: (Ai==0 && P2==0) ->  atomic {
          if
          :: (P1 == 1) -> { P1 = 0 ; }
          :: skip
          fi;}
:: (Ai==1 || P2==1) -> atomic {
          if
          :: (P1 == 0) -> { P1 = 1 ; }
          :: skip
          fi;}
od;
}
```

รูปที่ 3.12 proctype ที่แปลงจากเกต Or

เกต And(1) บรรทัดที่ 6 and(P2,Bi,P1) สัญญาณอินพุตหรือสัญญาณเชื่อมต่อที่เป็นอินพุตประกอบด้วย สัญญาณอินพุต Bi สัญญาณเชื่อมต่อภายในวงจร P1 และสัญญาณเอาต์พุตคือสัญญาณเชื่อมต่อภายในวงจร P2 แปลงเกตที่กำหนดให้เป็น proctype เมื่อสัญญาณอินพุตของเกต And มีค่าใดค่าหนึ่งเป็นค่าลบแทนค่าด้วย 0 (Bi==0 || P1==0) จะทำให้สัญญาณเอาต์พุตที่ได้เป็นค่าลบแทนค่าด้วย 0 (P2=0) และเมื่อสัญญาณอินพุตของเกต And เป็นค่าบวกทั้ง 2 ค่าแทนค่าด้วย 1 (Bi==1 && P1==1) จะทำให้สัญญาณเอาต์พุตที่ได้เป็นค่าบวกแทนค่าด้วย 1 (P2=1) ดังรูปที่ 3.13

```
active proctype AndGate1(){
do
:: (Bi == 0 || P1 == 0) -> atomic {
          if
          :: (P2 == 1) -> { P2 = 0; }
          :: skip
          fi;}
:: (Bi == 1 && P1 == 1) -> atomic {
          if
          :: (P2 == 0) -> { P2 = 1; }
          :: skip
          fi;}
od;
}
```

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

รูปที่ 3.13 proctype ที่แปลงจากเกต And

เกต Nand(1) บรรทัดที่ 7 nand(P3,P1,P4) สัญญาณอินพุตหรือสัญญาณเชื่อมต่อที่เป็นอินพุตประกอบด้วย สัญญาณเชื่อมต่อภายในวงจร P1 และ P4 และสัญญาณเอาต์พุตคือสัญญาณเชื่อมต่อภายในวงจร P3 แปลงเกตที่กำหนดให้เป็น proctype เมื่อสัญญาณอินพุตของเกต Nand มีค่าใดค่าหนึ่งเป็นค่าลบแทนค่าด้วย 0 (P1==0 || P4 ==0) จะทำให้สัญญาณเอาต์พุตที่ได้เป็นค่าบวก

แทนค่าด้วย 1 (P3 = 1) และเมื่อสัญญาณอินพุตของเกต Nand เป็นค่าบวกทั้ง 2 ค่าแทนค่าด้วย 1 (P1==1 && P4==1) จะทำให้สัญญาณเอาต์พุตที่ได้เป็นค่าลบแทนค่าด้วย 0 (P3=0) ดังรูปที่ 3.14

```
active proctype NandGate1(){
do
:: (P1 == 0 || P4 == 0) -> atomic {
          if
          :: (P3 == 0) -> { P3 = 1; }
          :: skip
          fi;}
:: (P1 == 1 && P4 == 1) -> atomic {
          if
          :: (P3 == 1) -> { P3 = 0; }
          :: skip
          fi;}
od;
}
```

รูปที่ 3.14 proctype ที่แปลงจากเกต Nand

เกต Nand(2) บรรทัดที่ 7 nand(P4,P3,P5) สัญญาณอินพุตหรือสัญญาณเชื่อมต่อที่เป็นอินพุตประกอบด้วย สัญญาณเชื่อมต่อภายในวงจร P4 และ P5 และสัญญาณเอาต์พุตคือสัญญาณเชื่อมต่อภายในวงจร P4 แปลงเกตที่กำหนดให้เป็น proctype เมื่อสัญญาณอินพุตของเกต Nand มีค่าใดค่าหนึ่งเป็นค่าลบแทนค่าด้วย 0 (P3==0 || P5 ==0) จะทำให้สัญญาณเอาต์พุตที่ได้เป็นค่าบวกแทนค่าด้วย 1 (P4 = 1) และเมื่อสัญญาณอินพุตของเกต Nand เป็นค่าบวกทั้ง 2 ค่าแทนค่าด้วย 1 (P3==1 && P5==1) จะทำให้สัญญาณเอาต์พุตที่ได้เป็นค่าลบแทนค่าด้วย 0 (P4=0) ดังรูปที่ 3.15

Chulalongkorn University

```
active proctype NandGate2(){
do
:: (P3 == 0 || P5 == 0) -> atomic {
          if
          :: (P4 == 0) -> { P4 = 1; }
          :: skip
          fi;}
:: (P3 == 1 && P5 == 1) -> atomic {
          if
          :: (P4 == 1) -> { P4= 0; }
          :: skip
          fi;}
od;
}
```

รูปที่ 3.15 proctype ที่แปลงจากเกต Nand

เกต Nand(3) บรรทัดที่ 9 nand(P5,P2,P3) สัญญาณอินพุตหรือสัญญาณเชื่อมต่อที่เป็นอินพุตประกอบด้วย สัญญาณเชื่อมต่อภายในวงจร P2 และ P3 และสัญญาณเอาต์พุตคือสัญญาณ

เชื่อมต่อภายในวงจร P5 แปลงเกตที่กำหนดให้เป็น proctype เมื่อสัญญาณอินพุตของเกต Nand มีค่าใดค่าหนึ่งเป็นค่าลบแทนค่าด้วย 0 (P2==0 || P3 ==0) จะทำให้สัญญาณเอาต์พุตที่ได้เป็นค่าบวกแทนค่าด้วย 1 (P5 = 1) และเมื่อสัญญาณอินพุตของเกต Nand เป็นค่าบวกทั้ง 2 ค่าแทนค่าด้วย 1 (P2==1 && P3==1) จะทำให้สัญญาณเอาต์พุตที่ได้เป็นค่าลบแทนค่าด้วย 0 (P5=0) ดังรูปที่ 3.16

```
active proctype NandGate3(){
do
:: (P2 == 0 || P3 == 0) -> atomic {
        if
        :: (P5 == 0) -> { P5 = 1; }
        :: skip
        fi;}
:: (P2 == 1 && P3 == 1) -> atomic {
        if
        :: (P5 == 1) -> { P5= 0; }
        :: skip
        fi;}
od;
}
```

รูปที่ 3.16 proctype ที่แปลงจากเกต Nand

เกต And(2) บรรทัดที่ 10 and(Co,P4,P5) สัญญาณอินพุตหรือสัญญาณเชื่อมต่อที่เป็นอินพุตประกอบด้วย สัญญาณเชื่อมต่อภายในวงจร P4 และ P5 สัญญาณเอาต์พุตคือสัญญาณเอาต์พุตของวงจร Co แปลงเกตที่กำหนดให้เป็น proctype เมื่อสัญญาณอินพุตของเกต And มีค่าใดค่าหนึ่งเป็นค่าลบแทนค่าด้วย 0 (P4==0 || P5==0) จะทำให้สัญญาณเอาต์พุตที่ได้เป็นค่าลบแทนค่าด้วย 0 (Co=0) และเมื่อสัญญาณอินพุตของเกต And เป็นค่าบวกทั้ง 2 ค่าแทนค่าด้วย 1 (P4==1 && P5==1) จะทำให้สัญญาณเอาต์พุตที่ได้เป็นค่าบวกแทนค่าด้วย 1 (Co=1) ดังรูปที่ 3.17

```
active proctype AndGate2(){
do
:: (P4 == 0 || P5 == 0) -> atomic {
        if
        :: (Co == 0) -> { Co = 1; }
        :: skip
        fi;}
:: (P4 == 1 && P5 == 1) -> atomic {
        if
        :: (Co == 1) -> { Co = 0; }
        :: skip
        fi;}
od;
}
```

รูปที่ 3.17 proctype ที่แปลงจากเกต And

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

<u>ขั้นที่ 4</u> ประกาศตัวแปรของเฉพาะสัญญาณเชื่อมต่อภายในวงจรแบบโกลบอล เนื่องจากสัญญาณอินพุตและสัญญาณเอาต์พุตของวงจร มีการประกาศตัวแปรและกำหนดค่าเริ่มต้นของสัญญาณไว้ก่อนหน้าแล้วในขั้นตอนการแปลงพฤติกรรมของเอสทีจีเป็นภาษาโพรเมล่า ดังนั้นจึงประกาศตัวแปรเฉพาะสัญญาณเชื่อมต่อภายในวงจร P1, P2, P3, P4 และ P5 และกำหนดค่าเริ่มต้นของสัญญาณเชื่อมต่อภายในวงจร ดังรูปที่ 3.18

```
bit P1=0, P2=0, P3=0, P4=0, P5=0;
```

รูปที่ 3.18 ตัวแปรแบบโกลบอลพร้อมกำหนดค่าเริ่มต้นของสัญญาณ

<u>ขั้นที่ 5</u> ตัวแปรแบบโกลบอลสำหรับนับลำดับของการเปลี่ยนแปลงสัญญาณ ถูกประกาศไว้แล้วในขั้นตอนการแปลงพฤติกรรมของเอสทีจีเป็นภาษาโพรเมล่า คือตัวแปรนับ i โดยให้เพิ่มตัวแปรนับในทุกครั้งที่มีการเปลี่ยนแปลงค่าของสัญญาณอินพุตและสัญญาณเอาต์พุตของวงจร proctype ที่มีการเปลี่ยนแปลงที่สัญญาณเอาต์พุตคือ AndGate2 ดังรูปที่ 3.19

```
active proctype AndGate2(){
do
:: (P4 == 0 || P5 == 0) -> atomic {
        if
        :: (Co == 0) -> { Co = 1; i = i + 1; }
        :: skip
        fi;}
:: (P4 == 1 && P5 == 1) -> atomic {
        if
        :: (Co == 1) -> { Co = 0; i = i + 1; }
        :: skip
        fi;}
od;
}
```

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

รูปที่ 3.19 proctype AndGate2 หลังเพิ่มตัวแปรนับในทุกครั้งที่มีการเปลี่ยนแปลงสัญญาณเอาต์พุตของวงจร

<u>ขั้นที่ 7</u> เพิ่มคำสั่ง printf(“\n Output > |%d| N1=|%d|, N2=|%d|,..., Nn=|%d| \n”, Counter, N1, N2,..., Nn) เพื่อแสดงผลการเปลี่ยนแปลงค่าของสัญญาณอินพุตและสัญญาณเอาต์พุตของวงจรในแต่ละครั้ง ดังรูปที่ 3.20 โดย

- เกต Or(1) เอาต์พุตเป็นสัญญาณเชื่อมต่อภายในวงจร P1 ดังนั้น จึงไม่จำเป็นต้องมีคำสั่ง printf
- เกต And(1) เอาต์พุตเป็นสัญญาณเชื่อมต่อภายในวงจร P2 ดังนั้น จึงไม่จำเป็นต้องมีคำสั่ง printf

- เกต Nand(1) เอาต์พุตเป็นสัญญาณเชื่อมต่อภายในวงจร P3 ดังนั้น จึงไม่จำเป็นต้องมีคำสั่ง printf
- เกต Nand(2) เอาต์พุตเป็นสัญญาณเชื่อมต่อภายในวงจร P4 ดังนั้น จึงไม่จำเป็นต้องมีคำสั่ง printf
- เกต Nand(3) เอาต์พุตเป็นสัญญาณเชื่อมต่อภายในวงจร P5 ดังนั้น จึงไม่จำเป็นต้องมีคำสั่ง printf
- เกต And(2) เอาต์พุตเป็นสัญญาณเอาต์พุต Co ของวงจร ดังนั้น จึงจำเป็นต้องมีคำสั่ง printf

```
active proctype AndGate2(){
do
:: (P4 == 0 || P5 == 0) -> atomic {
            if
            :: (Co == 0) -> { Co = 1; i = i + 1;
                printf("\n Output >  |%d| Ai=|%d|, Bi=|%d|, Co =|%d| \n", i, Ai,Bi,Co); }
            :: skip
            fi;}
:: (P4 == 1 && P5 == 1) -> atomic {
            if
            :: (Co == 1) -> { Co = 0; i = i + 1; }
                printf("\n Output >  |%d| Ai=|%d|, Bi=|%d|, Co =|%d| \n", i, Ai,Bi,Co); }
            :: skip
            fi;}
od;
}
```

รูปที่ 3.20 proctype หลังจากการเพิ่มคำสั่ง printf()

จุฬาลงกรณ์มหาวิทยาลัย

การแปลงแฟ้มข้อความนำเข้าของวงจรอสมวารที่ได้จากการสังเคราะห์ เป็นภาษาโพรเมล่า จากตัวอย่างแฟ้มข้อความนำเข้าของวงจรอสมวารที่ได้จากการสังเคราะห์ รูปที่ 3.11 รหัสภาษาโพรเมล่าที่ได้ นำมารวมกับรหัสภาษาโพรมาล่าที่ได้จากการแปลงพฤติกรรมในเอสทีจี จะได้รหัสภาษาโพรเมล่าที่ทำการแปลงสำเร็จ ดังรูปที่ 3.21

```
bit Ai=0, Bi=0, Co=0;
bit P1=0, P2=0, P3=0, P4=0, P5=0;
int i = 0;

active proctype OrGate1(){
do
:: (Ai==0 && P2==0) ->  atomic {
            if
            :: (P1 == 1) -> { P1 = 0 ; }
            :: skip
            fi;
            }
:: (Ai==1 || P2==1) -> atomic {
            if
            :: (P1 == 0) -> { P1 = 1 ; }
            :: skip
            fi;
            }
od;
}

active proctype AndGate1(){
do
:: (Bi == 0 || P1 == 0) -> atomic {
            if
            :: (P2 == 1) -> { P2 = 0; }
            :: skip
            fi;
            }
:: (Bi == 1 && P1 == 1) -> atomic {
            if
            :: (P2 == 0) -> { P2 = 1; }
            :: skip
            fi;
            }
od;
}

active proctype NandGate1(){
do
:: (P1 == 0 || P4 == 0) -> atomic {
            if
            :: (P3 == 0) -> { P3 = 1; }
            :: skip
            fi;
            }
:: (P1 == 1 && P4 == 1) -> atomic {
            if
            :: (P3 == 1) -> { P3 = 0; }
            :: skip
            fi;
            }
od;
}

active proctype NandGate2(){
do
:: (P3 == 0 || P5 == 0) -> atomic {
            if
            :: (P4 == 0) -> { P4 = 1; }
            :: skip
            fi;
            }
:: (P3 == 1 && P5 == 1) -> atomic {
            if
            :: (P4 == 1) -> { P4 = 0; }
            :: skip
            fi;
            }
od;
}
```

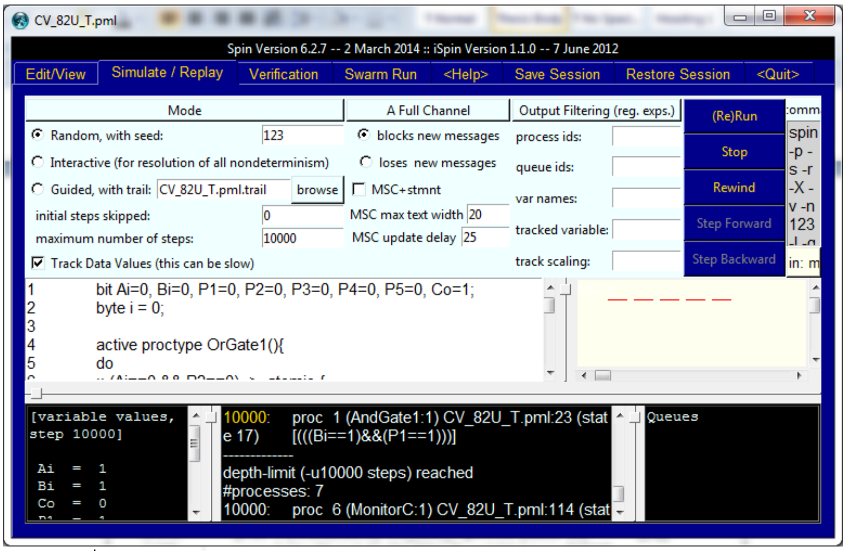
```
active proctype NandGate3(){
do
:: (P2 == 0 || P3 == 0) -> atomic {
            if
            :: (P5 == 0) -> { P5 = 1; }
            :: skip
            fi;
            }
:: (P2 == 1 && P3 == 1) -> atomic {
            if
            :: (P5 == 1) -> { P5 = 0; }
            :: skip
            fi;
            }
od;
}

active proctype AndGate2(){
do
:: (P4 == 0 || P5 == 0) -> atomic {
            if
            :: (Co == 1) -> { Co = 0; i = i + 1;
               printf("\n Output >  |%d| Ai=|%d|,
Bi=|%d|, Co =|%d| \n", i, Ai,Bi,Co); }
            :: skip
            fi;
            }
:: (P4 == 1 && P5 == 1) -> atomic {
            if
            :: (Co == 0) -> { Co = 1; i = i + 1;
               printf("\n Output >  |%d| Ai=|%d|,
Bi=|%d|, Co =|%d| \n", i, Ai,Bi,Co); }
            :: skip
            fi;
            }
od;
}

active proctype MonitorC(){
do
:: (Co == 1) ->  atomic {
            if
            :: (Ai == 1) -> { Ai = 0 ; i = i + 1;
               printf("\n Output >  |%d| Ai=|%d|,
Bi=|%d|, Co =|%d| \n", i, Ai,Bi,Co); }
            :: skip
            fi;
            if
            :: (Bi == 1) -> { Bi = 0 ; i = i + 1;
               printf("\n Output >  |%d| Ai=|%d|,
Bi=|%d|, Co =|%d| \n", i, Ai,Bi,Co); }
            :: skip
            fi;
            }
:: (Co == 0) ->  atomic {
            if
            :: (Ai == 0) -> { Ai = 1 ; i = i + 1;
               printf("\n Output >  |%d| Ai=|%d|,
Bi=|%d|, Co =|%d| \n", i, Ai,Bi,Co); }
            :: skip
            fi;
            if
            :: (Bi == 0) -> { Bi = 1 ; i = i + 1;
               printf("\n Output >  |%d| Ai=|%d|,
Bi=|%d|, Co =|%d| \n", i, Ai,Bi,Co); }
            :: skip
            fi;
            }
od;
}
```

รูปที่ 3.21 รหัสโพรเมล่าส่วนพฤติกรรมในเอสทีจีและส่วนของวงจรอสมวารที่ได้จากการสังเคราะห์หลังทำการรวมเข้าด้วยกัน

3.3 การจำลองการทำงานด้วยสปิน

จากรูปที่ 3.1 ขั้นตอนที่ 3 การจำลองการทำงานด้วยสปิน ข้อมูลนำเข้าของขั้นตอนนี้คือรหัสภาษาโพรเมล่าที่ได้จากพฤติกรรมในเอสทีจีและรหัสภาษาโพรเมล่าที่ได้จากวงจรอสมวารที่ได้จากการสังเคราะห์หลังทำการรวมกันแล้ว ในขั้นตอนนี้เป็นการจำลองการทำงานเพื่อให้ได้แฟ้มข้อความจากการจำลองการทำงาน จากนั้นคัดกรองเฉพาะส่วนที่ต้องการในแฟ้มข้อความเพื่อให้ได้ เป็นเอสเอสซึ่งนำไปใช้ในการทวนสอบคุณสมบัติไลฟ์เนส และคุณสมบัติความทนทานต่อไป

โปรแกรมที่ใช้ในการจำลองคือโปรแกรมไอสปิน (iSPIN) โดยใช้คุณสมบัติการทำงานเฉพาะในส่วนของการจำลองการทำงานของโปรแกรมเท่านั้น ดังรูปที่ 3.22



รูปที่ 3.22 โปรแกรมไอสปินสำหรับประมวลผลการจำลองการทำงาน

CHULALONGKORN UNIVERSITY

เมื่อจำลองการทำงานแล้วให้บันทึกผลจากการจำลองการทำงานลงในแฟ้มข้อความด้วยวิธีไม่อัตโนมัติ ดังรูปที่ 3.23

```
...
2493:   proc  4 (InvertAndGate4:1) CV_82U_T.pml:75 (state 17)
        [((((P2==0)&&(P3==0))||(P2!=P3)))]
2494:   proc  3 (InvertAndGate3:1) CV_82U_T.pml:65 (state 16)     [(1)]
2496:   proc  6 (MonitorC:1) CV_82U_T.pml:114 (state 37)  [((Co==0))]
2497:   proc  5 (AndGate5:1) CV_82U_T.pml:93 (state 10)  [(1)]
2499:   proc  1 (AndGate1:1) CV_82U_T.pml:23 (state 17)
        [((((Bi==0)&&(P1==0))||(Bi!=P1)))]
2500:   proc  6 (MonitorC:1) CV_82U_T.pml:129 (state 36)  [((Ai==0))]
2501:   proc  6 (MonitorC:1) CV_82U_T.pml:133 (state 24)  [Ai = 1]
2502:   proc  6 (MonitorC:1) CV_82U_T.pml:132 (state 22)  [i = (i+1)]
 Output >  |21| Ai=|1|, Bi=|1|, Co =|0|
...
```

รูปที่ 3.23 ตัวอย่างแฟ้มข้อความจากการจำลองการทำงานด้วยสปิน

นำแฟ้มข้อความที่ได้จากการจำลองการทำงาน มาคัดกรองเฉพาะบรรทัดที่แสดงผลการเปลี่ยนแปลงค่าของสัญญาณอินพุต และสัญญาณเอาต์พุตของวงจรในแต่ละครั้งจากการจำลองการทำงาน ประกอบด้วยลำดับขั้นของการเปลี่ยนแปลงสัญญาณ และค่าของแต่ละสัญญาณในขณะที่เกิดการเปลี่ยนแปลงสัญญาณ

การคัดกรองจากแฟ้มข้อความจากการจำลองการทำงาน มี 4 ขั้นตอน

1. ค้นหาแต่ละบรรทัดที่ขึ้นต้นด้วยคำว่า Output

2. ตัดแบ่งข้อความในบรรทัดด้วยเครื่องหมาย “|”

3. ตำแหน่งที่ 2 หมายถึงลำดับขั้นของการเปลี่ยนแปลงสัญญาณ

4. ตำแหน่งถัดไปหลังตำแหน่งที่ 2 จาก ให้แบ่งเป็นชุด ชุดละ 2 ตำแหน่ง โดยในแต่ละชุดตำแหน่งแรกเป็นชื่อของสัญญาณและตำแหน่งที่สองจะเป็นค่าของสัญญาณนั้นๆ

ตัวอย่างจากรูปที่ 3.23 บรรทัดที่ 20 Output > |21| Ai=|1|, Bi=|1|, Co =|0| เมื่อคัดกรอกด้วยขั้นตอนที่กล่าว จะได้ผลลัพธ์เป็น ลำดับขั้นของการเปลี่ยนแปลงสัญญาณที่ 20 สัญญาณอินพุต Ai มีค่าเป็นบวกซึ่งถูกแทนค่าด้วย 1, สัญญาณอินพุต Bi มีค่าเป็นบวกซึ่งถูกแทนค่าด้วย 1และสัญญาณเอาต์พุต Co มีค่าเป็นลบซึ่งถูกแทนค่าด้วย 0

จากรหัสภาษาโพรเมล่าที่ได้จากการแปลงจากพฤติกรรมในเอสทีจีและวงจรอสมวารที่ได้จากการสังเคราะห์จากรูปที่ 3.21 ทำจำลองการทำงานในสปิน ได้ผลลัพธ์เป็นแฟ้มข้อความจากการจำลองการทำงาน ทำการคัดกรองตามขั้นตอนการคัดกรองจากแฟ้มข้อความตามที่กล่าว สามารถแสดงลำดับการเปลี่ยนแปลงสัญญาณดังตารางที่ 3.1

ตารางที่ 3.1 ตารางแสดงลำดับการเปลี่ยนแปลงสัญญาณจากการอ่านแฟ้มข้อความจากการจำลองการทำงานด้วยสปิน

ขั้นของการเปลี่ยนแปลงสัญญาณ	สัญญาณอินพุต Ai	สัญญาณอินพุต Bi	สัญญาณเอาต์พุต Co
1	0	0	0
2	1	0	0
3	1	1	0

ตารางที่ 3.1 ตารางแสดงลำดับการเปลี่ยนแปลงสัญญาณจากการอ่านแฟ้มข้อความจากการจำลองการทำงานด้วยสปิน (ต่อ)

ขั้นของการเปลี่ยนแปลงสัญญาณ	สัญญาณอินพุต Ai	สัญญาณอินพุต Bi	สัญญาณเอาต์พุต Co
4	1	1	1
5	0	1	1
6	0	0	1
7	0	0	0
8	1	0	0
9	1	1	0
10	1	1	1
11	0	1	1
12	0	0	1
13	0	0	0
…	…	…	…

ตางรางที่ 3.1 แสดงลำดับการเปลี่ยนแปลงของแต่ละสัญญาณในแต่ละขั้นโดยรูปแบบตารางนี้จะนำไปใช้ในขั้นตอนการทวนสอบแอลอาร์เอสต่อไป เนื่องจากแสดงให้เห็นลำดับการเปลี่ยนแปลงที่ชัดเจนในแต่ละสัญญาณ

เนื่องจากเอสเอสนั้นมีความยาวมากตามจำนวนรอบวนซ้ำการจำลองการทำงาน และไม่สะดวกในการทวนสอบกับเอสทีเอส ดังนั้นผู้วิจัยจึงใช้การแบ่งเอสเอส โดยในเอสเอสจะประกอบด้วยลำดับขั้นของการเปลี่ยนแปลงทุกสัญญาณในเอสทีจี ทำการแบ่งเป็นแต่ละรอบการเปลี่ยนแปลงสัญญาณของวงจร ด้วยรอบปัจจุบันและรอบก่อนหน้านำมาต่อกัน ยกเว้นรอบที่หนึ่งให้ใช้เพียงการเปลี่ยนแปลงสัญญาณของรอบที่ 1 เท่านั้น เนื่องจากไม่มีรอบก่อนหน้า จากตัวอย่างตารางที่ 3.1 การเปลี่ยนแปลงสัญญาณในวงจรรอบที่สอง คือ Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 ของขั้นการเปลี่ยนแปลงสัญญาณที่ 8-13 และรอบก่อนหน้าคือ Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 ของขั้นการเปลี่ยนแปลงสัญญาณที่ 2-7 เมื่อรวมทั้งสองรอบจะได้ Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 > Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 สำหรับนำไปใช้ในขั้นตอนการทวนสอบเอสทีเอสต่อไป ดังตารางที่ 3.2

ตารางที่ 3.2 ตารางเอสเอสจากการแบ่งการเปลี่ยนแปลงสัญญาณรอบปัจจุบันและรอบก่อนหน้า

ขั้นแรก	**ขั้นสุดท้าย**	**เอสเอสที่ถูกแบ่งด้วยรอบปัจจุบันและก่อนหน้า**
2	7	Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 >
2	13	Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 > Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 >
8	19	Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 > Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 >
14	25	Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 > Bi1 > Ai1 > Co1 > Ai0 > Bi0 > Co0 >
20	31	Bi1 > Ai1 > Co1 > Ai0 > Bi0 > Co0 > Ai1 > Bi1 > Co1 > Bi0 > Ai0 > Co0 >
26	37	Ai1 > Bi1 > Co1 > Bi0 > Ai0 > Co0 > Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 >
32	43	Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 > Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 >
38	49	Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 > Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 >
44	55	Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 > Bi1 > Ai1 > Co1 > Bi0 > Ai0 > Co0 >
50	61	Bi1 > Ai1 > Co1 > Bi0 > Ai0 > Co0 > Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 >
...	...	...

3.4 การหาเอสทีเอสในเอสทีจี

จากรูปที่ 3.1 ขั้นตอนที่ 4 การหาเอสทีเอสในเอสทีจี ข้อมูลนำเข้าในขั้นตอนนี้คือเอสทีจีภายใต้การทำงานแบบวัฏจักรเชิงเดียวและวัฏจักรที่ไม่มีจุดยอดซ้ำกัน โดยแสดงลำดับการเปลี่ยนแปลงสัญญาณจากสถานะเริ่มต้น (Initial state) ไปจนถึงสถานะที่ไม่ซ้ำกัน (Unique state) สุดท้ายก่อนที่โทเค็นจะเคลื่อนไปสถานะที่ซ้ำกันในลำดับการเปลี่ยนแปลงนั้นๆ วิธีการค้นหานั้นใช้วิธีการค้นหาแบบครบถ้วนสมบูรณ์ (Exhaustive search) ภายในเงื่อนไขของเอสทีจี ผลลัพธ์ที่ได้คือเอสทีเอส ซึ่งจะนำไปใช้ในการทวนสอบคุณสมบัติไลฟ์เนสต่อไป

เครื่องมือที่ใช้พัฒนาขึ้นด้วยภาษาซีชาร์ป (C# programming language) และใช้โปรแกรม Visual Studio 2010 ในการพัฒนา โดยใช้รหัสเทียม (Pseudo code) ดังรูปที่ 3.24

```
BEGIN
SET Signals
SET Behavior into EachSignal
SET LevelStep into EachSignal
SET InitialState in EachSignal
SET CountPath = 0
DeepLevel = Level_X
IF (AllPosibleNextStateSignal OF ListSignal_Y) = 0
FOR Step on (ListSignal_Y. Signals ORDER BY LevelStep)
        PRINT LevelStep
END FOR
CountPath = CountPath + 1
ELSE
FOR PosibleNextStateSignal on (AllPosibleNextStateSignal OF ListSignal_Y)
        FOR Signal on ListSignal_Y
                FOR Step on Signal
                        IF Step >= Level_X
                                BackTrack(Signal)
                        END IF
                END IF
        END FOR
ChangeToNextState(PosibleNextStateSignal)
CALL RecursiveThisMethod(ListSignal_Y, DeepLevel + 1)
END FOR
END IF
END
```

รูปที่ 3.24 รหัสเทียมในการค้นหาเอสทีเอสในเอสทีจีภายใต้การทำงานแบบวัฏจักรเชิงเดียวและวัฏจักรที่ไม่มีจุดยอดซ้ำกัน

การค้นหาเอสทีเอสจากเอสทีจี จากรหัสเทียมรูปที่ 3.24 สามารถสรุปขั้นตอนได้ดังนี้

1. กำหนดสัญญาณทั้งหมดของวงจร พร้อมค่าเริ่มต้นของแต่ละสัญญาณ

3. กำหนดพฤติกรรมของแต่ละสัญญาณ

4. สร้างตัวนับระดับความลึกมีค่าเท่ากับ 0

5. สร้างตัวเก็บบันทึกการเปลี่ยนแปลงแต่ละครั้งของสัญญาณ

6. ค้นหาทุกสถานที่ถัดไปของโทเค็น ที่จะสามารถเคลื่อนไปได้พร้อมบันทึกในตัวนับความลึกโดยเท่ากับค่าความลึกเดิม + 1

7. กรณีไม่มีสถานที่ถัดไปที่โทเค็นจะสามารถเคลื่อนไปได้แล้วให้บันทึกเป็นเอสทีเอส และลดระดับความลึกในตัวนับความลึกโดยกำหนดให้เท่ากับ ค่าความลึกเดิม – 1

8. กรณีมีสถานที่ถัดที่โทเค็นจะสามารถเคลื่อนไปได้ ให้วนรอบแต่ละสถานที่ของทุกสถานที่ถัดไปที่โทเค็นจะสามารถเคลื่อนไปได้ และแต่ละการวนรอบแต่ละสถานที่ให้ เรียกซ้ำข้อ 6 - 8 ไปเรื่อยๆ

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ผลลัพธ์จากการค้นหาเอสทีเอสจากเอสทีจี รูปที่ 3.2(a) ได้ผลลัพธ์ทั้งหมด 4 เอสทีเอส

เอสทีเอส 1. Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0

เอสทีเอส 2. Ai1 > Bi1 > Co1 > Bi0 > Ai0 > Co0

เอสทีเอส 3. Bi1 > Ai1 > Co1 > Ai0 > Bi0 > Co0

เอสทีเอส 4. Bi1 > Ai1 > Co1 > Bi0 > Ai0 > Co0

3.5 การทวนสอบเอสทีเอส

จากรูปที่ 3.1 ขั้นตอนที่ 5 การทวนสอบเอสทีเอส ในขั้นตอนนี้ข้อมูลนำเข้าคือเอสทีเอสและเอสเอส เป็นการทวนสอบเอสเอสที่สอดคล้องกับรูปแบบของเอสทีเอส เนื่องจากเอสทีเอสนั้นเป็นลำดับการเปลี่ยนแปลงที่คาดว่าน่าจะเกิดขึ้นกับพฤติกรรมของวงจรที่ได้ออกแบบไว้ แต่เอสเอสเป็นผลจากการจำลองการทำงานจริงของวงจรที่ได้จากการสังเคราะห์ ดังนั้นจึงจำเป็นต้องทวนสอบว่าวงจรที่ได้จากการสังเคราะห์นั้นสามารถทำงานได้ตามรูปแบบของเอสทีเอสได้ทุกกรณีหรือไม่เพื่อพิสูจน์คุณสมบัติไลฟ์เนส

จากขั้นตอนที่ 3 ข้อที่ 3.3 เอสเอสที่มีการแบ่งรอบการเปลี่ยนแปลงสัญญาณของวงจรปัจจุบันและรอบก่อนหน้าที่ได้ทั้งหมดนำมาใช้ในการทวนสอบเอสทีเอส โดยขั้นตอนในการทวนสอบมีดังนี้

การทวนสอบเอสทีเอสกับเอสเอส มี 2 ขั้นตอน

1. นำเอสทีเอสแต่ละเส้นทางมาวนรอบตรวจสอบกับเอสเอส ที่มีการแบ่งรอบการเปลี่ยนแปลงสัญญาณของวงจรรอบปัจจุบันและรอบก่อนหน้าที่ได้ทั้งหมด โดยรูปแบบของแต่ละ เอสทีเอสต้องพบทุกรูปแบบของเอสทีเอสในเอสเอส

2. กรณีไม่สามารถพบครบทุกรูปแบบของเอสทีเอสในเอสเอส ให้เพิ่มรอบการวนซ้ำการจำลองในสปินและนำผลที่ได้มาทวนสอบใหม่ กรณียังไม่สามารถพบได้ครบให้ดำเนินการวนซ้ำตามข้อที่ 1 และ 2 ใหม่จนกว่าจะสามารถค้นพบครบทุกรูปแบบของเอสทีเอสในเอสเอส

ตัวอย่างการทวนสอบเอสทีเอสกับเอสเอส จากเอสทีเอสผลลัพธ์ที่หาได้จากข้อที่ 3.4 และจากเอสเอสผลลัพธ์ที่ได้จากข้อที่ 3.3 ผลจากการทวนสอบแสดงในตารางที่ 3.3

ตารางที่ 3.3 ผลการทวนสอบเอสทีเอสที่ได้จากข้อที่ 3.4 และจากเอสเอสที่ได้จากข้อที่ 3.3

ขั้นแรก	ขั้นสุดท้าย	เอสเอสที่ถูกแบ่งด้วยรอบปัจจุบันและก่อนหน้า	เอสทีเอสที่พบ
2	7	Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 >	STS 1
2	13	Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 > Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 >	
8	19	Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 > Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 >	
14	25	Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 > Bi1 > Ai1 > Co1 > Ai0 > Bi0 > Co0 >	STS 3
20	31	Bi1 > Ai1 > Co1 > Ai0 > Bi0 > Co0 > Ai1 > Bi1 > Co1 > Bi0 > Ai0 > Co0 >	STS 2
26	37	Ai1 > Bi1 > Co1 > Bi0 > Ai0 > Co0 > Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 >	
32	43	Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 > Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 >	
38	49	Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 > Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 >	
44	**55**	**Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 > Bi1 > Ai1 > Co1 > Bi0 > Ai0 > Co0 >**	**STS 4**
50	61	Bi1 > Ai1 > Co1 > Bi0 > Ai0 > Co0 > Ai1 > Bi1 > Co1 > Ai0 > Bi0 > Co0 >	
...	...	...	...

จากตารางที่ 3.3 จำนวนรอบวนซ้ำการทำงานน้อยที่สุดของเอสเอส และสามารถพบทุกรูปแบบของเอสทีเอส คือลำดับที่ 55 แสดงให้เห็นว่าเอสเอสสามารถตรวจพบได้ทุกรูปแบบของเอสทีเอส ซึ่งเป็นการพิสูจน์เชิงคุณสมบัติไลฟ์เนส กรณีการจำลองการทำงานนั้นไม่สามารถตรวจพบได้ทุกรูปแบบของเอสทีเอส ให้ทำการเพิ่มรอบวนซ้ำการจำลองทำงาน จากนั้นจำลองการทำงานและทวนสอบใหม่อีกครั้ง ผลลัพธ์ในขั้นตอนนี้คือ จำนวนรอบวนซ้ำการทำงานน้อยที่สุดของเอสเอสที่แสดงถึงคุณสมบัติไลฟ์เนส หรือเรียกว่าแอลเอสเอส ซึ่งจะนำไปใช้ทวนสอบในขั้นตอนทวนสอบคุณสมบัติความทนทานต่อไป

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

3.6 การหาแอลอาร์เอสในเอสทีจี

จากรูปที่ 3.1 ขั้นตอนที่ 6 การหาแอลอาร์เอส ข้อมูลนำเข้าของขั้นตอนนี้คือเอสทีจี ภายใต้การทำงานแบบวัฏจักรเชิงเดียวและวัฏจักรที่ไม่มีจุดยอดซ้ำกันกล่าวคือ สัญญาณที่มีคุณสมบัติความทนทานนั้น ต้องมีคุณสมบัติเชิงล็อคแบบฟูลล็อคหรือแบบเซมิล็อคอย่างใดอย่างหนึ่งในรูปเอสทีจี โดยอาศัยคุณสมบัติความสัมพันธ์เชิงล็อค [6] ในการค้นหา

การหาแอลอาร์เอสในเอสทีจี มี 2 ขั้นตอนดังนี้

1. ค้นหาแอลอาร์เอสแบบฟูลล็อค ถ้าสองสัญญาณใดๆ a และ b แสดงคุณสมบัติดังนี้ $a \rightarrow b \rightarrow \overline{a^*} \rightarrow \overline{b^*}$ ถือว่าสองสัญญาณนั้นมีความสัมพันธ์เชิงล็อคแบบฟูลล็อคร่วมกัน
2. ค้นหาแอลอาร์เอสแบบเซมิล็อค ถ้าสองสัญญาณใดๆ a และ b แสดงคุณสมบัติดังนี้ $a^* \rightarrow b^* \rightarrow \overline{a^*}$ หรือ $b^* \rightarrow a^* \rightarrow \overline{b^*}$ ถือว่าสองสัญญาณนั้นมีความสัมพันธ์เชิงล็อคแบบเซมิล็อคร่วมกัน

ตัวอย่างเอสทีจีจากรูปที่ 3.1(a) เมื่อนำมาหาความสัมพันธ์เชิงล็อคด้วยขั้นตอนดังกล่าวได้ผลลัพธ์ดังตารางที่ 3.1

ตารางที่ 3.4 ความสัมพันธ์เชิงล็อคจากเอสทีจี

ลำดับ	คู่สัญญาณ	แอลอาร์เอส	ประเภทความสัมพันธ์เชิงล็อค
1	Ai, Co	Ai0 > Co0 > Ai1	เซมิล็อค
2	Bi, Co	Bi0 > Co0 > Bi1	เซมิล็อค
3	Ai, Co	Ai0 > Co0 > Ai1 > Co1	ฟูลล็อค
4	Bi, Co	Bi0 > Co0 > Bi1 > Co1	ฟูลล็อค

3.7 การทวนสอบแอลอาร์เอส

จากรูปที่ 3.1 ขั้นตอนที่ 7 การทวนสอบแอลอาร์เอส โดยข้อมูลนำเข้าในขั้นตอนนี้คือแอลอาร์เอส และแอลเอสเอส เป็นการทวนสอบคุณสมบัติความทนทานอีกระดับหนึ่ง หลังจากทวนสอบคุณสมบัติไลฟ์เนสมาแล้ว โดยตรวจสอบแอลเอสเอสว่ามีลำดับการเปลี่ยนแปลงสัญญาณจากการจำลองการทำงานของคู่สัญญาณนั้นๆ ว่าถูกต้องตรงกับแอลอาร์เอสตามคุณสมบัติความสัมพันธ์เชิงล็อคหรือไม่ โดยมีขั้นตอนดังนี้

การทวนสอบแอลอาร์เอสกับแอลเอสเอส มี 3 ขั้นตอน

1. ตรวจสอบลำดับการเปลี่ยนแปลงสัญญาณในแอลเอสเอส ด้วยความสัมพันธ์เชิงล็อคแบบเซมิล็อคในแอลอาร์เอส ของสัญญาณ a และ b ใดๆ ต้องมีลำดับการเปลี่ยนแปลงสัญญาณ ดังนี้ $a \rightarrow b \rightarrow \overline{a^*}$ หมายถึงดังนี้
   - a ต้องเปลี่ยนแปลงสัญญาณจาก 0 เป็น 1 ก่อน b เปลี่ยนจาก 0 เป็น 1 เสมอ
   - b ต้องเปลี่ยนแปลงสัญญาณจาก 0 เป็น 1 ก่อน a เปลี่ยนจาก 1 เป็น 0 เสมอ
2. ตรวจสอบลำดับการเปลี่ยนแปลงสัญญาณในแอลเอสเอส ด้วยความสัมพันธ์เชิงล็อคแบบฟูลล็อคในแอลอาร์เอส ของสัญญาณ a และ b ใดๆ ต้องมีลำดับการเปลี่ยนแปลงสัญญาณ ดังนี้ $a \rightarrow b \rightarrow \overline{a^*} \rightarrow \overline{b^*}$ หมายถึงดังนี้
   - a ต้องเปลี่ยนแปลงสัญญาณจาก 0 เป็น 1 ก่อน b เปลี่ยนจาก 0 เป็น 1 เสมอ
   - b ต้องเปลี่ยนแปลงสัญญาณจาก 0 เป็น 1 ก่อน a เปลี่ยนจาก 1 เป็น 0 เสมอ
   - a ต้องเปลี่ยนแปลงสัญญาณจาก 1 เป็น 0 ก่อน b เปลี่ยนจาก 1เป็น 0 เสมอ
3. ตรวจสอบให้ครบทุกคู่สัญญาณที่มีความสัมพันธ์เชิงล็อคร่วมกันตามแอลอาร์เอส

ตัวอย่างจากการทวนสอบแอลอาร์เอสกับแอลเอสเอส จากแอลอาร์เอสผลลัพธ์ที่หาได้จากข้อที่ 3.6 และแอลเอสเอสผลลัพธ์ที่ได้จากข้อที่ 3.5 โดยตรวจสอบแยกแต่ละคู่สัญญาณที่มีความสัมพันธ์เชิงล็อคร่วมกันมีรายละเอียดดังนี้

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

1. การตรวจสอบคู่ความสัมพันธ์เชิงล็อคของคู่สัญญาณ Ai และ Co

จากขั้นตอนการทวนสอบแอลเอสเอสกับแอลอาร์เอสของคู่ความสัมพันธ์เชิงล็อค Ai และ Co ในตารางที่ 3.5 ตารางแสดงลำดับการเปลี่ยนแปลงสัญญาณจากการจำลองการทำงาน

ตารางที่ 3.5 แสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ Ai และ Co ขั้นที่ 1-30

ขั้นที่	ผลจากการจำลองการทำงาน	Ai	Bi	Co
1	Output > |1| Ai=|0|, Bi=|0|, Co =|0|	0	0	0
2	Output > |2| Ai=|1|, Bi=|0|, Co =|0|	1	0	0
3	Output > |3| Ai=|1|, Bi=|1|, Co =|0|	1	1	0
4	Output > |4| Ai=|1|, Bi=|1|, Co =|1|	1	1	1
5	Output > |5| Ai=|0|, Bi=|1|, Co =|1|	0	1	1

ตารางที่ 3.5 แสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ Ai และ Co ขั้นที่ 1-30 (ต่อ)

<table>
<tr><th>ขั้นที่</th><th>ผลจากการจำลองการทำงาน</th><th>Ai</th><th>Bi</th><th>Co</th></tr>
<tr><td>6</td><td>Output > |6| Ai=|0|, Bi=|0|, Co =|1|</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>7</td><td>Output > |7| Ai=|0|, Bi=|0|, Co =|0|</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>8</td><td>Output > |8| Ai=|1|, Bi=|0|, Co =|0|</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>9</td><td>Output > |9| Ai=|1|, Bi=|1|, Co =|0|</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>10</td><td>Output > |10| Ai=|1|, Bi=|1|, Co =|1|</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>11</td><td>Output > |11| Ai=|0|, Bi=|1|, Co =|1|</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>12</td><td>Output > |12| Ai=|0|, Bi=|0|, Co =|1|</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>13</td><td>Output > |13| Ai=|0|, Bi=|0|, Co =|0|</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>14</td><td>Output > |14| Ai=|1|, Bi=|0|, Co =|0|</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>15</td><td>Output > |15| Ai=|1|, Bi=|1|, Co =|0|</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>16</td><td>Output > |16| Ai=|1|, Bi=|1|, Co =|1|</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>17</td><td>Output > |17| Ai=|0|, Bi=|1|, Co =|1|</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>18</td><td>Output > |18| Ai=|0|, Bi=|0|, Co =|1|</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>19</td><td>Output > |19| Ai=|0|, Bi=|0|, Co =|0|</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>20</td><td>Output > |20| Ai=|0|, Bi=|1|, Co =|0|</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>21</td><td>Output > |21| Ai=|1|, Bi=|1|, Co =|0|</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>22</td><td>Output > |22| Ai=|1|, Bi=|1|, Co =|1|</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>23</td><td>Output > |23| Ai=|0|, Bi=|1|, Co =|1|</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>24</td><td>Output > |24| Ai=|0|, Bi=|0|, Co =|1|</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>25</td><td>Output > |25| Ai=|0|, Bi=|0|, Co =|0|</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>26</td><td>Output > |26| Ai=|1|, Bi=|0|, Co =|0|</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>27</td><td>Output > |27| Ai=|1|, Bi=|1|, Co =|0|</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>28</td><td>Output > |28| Ai=|1|, Bi=|1|, Co =|1|</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>29</td><td>Output > |29| Ai=|1|, Bi=|0|, Co =|1|</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>30</td><td>Output > |30| Ai=|0|, Bi=|0|, Co =|1|</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>...</td><td>...</td><td>...</td><td>...</td><td>...</td></tr>
</table>

จุฬาลงกรณ์มหาวิทยาลัย Chulalongkorn University

1.1) ความสัมพันธ์เชิงล็อคประเภทเซมิล็อค ระหว่างสัญญาณ Ai และ Co

จากคุณสมบัติความสัมพันธ์เชิงล็อค จากตารางคู่ความสัมพันธ์ของสัญญาณ Ai และ Co มีลักษณะแบบเซมิล็อค Ai0 > Co0 > Ai1 โดนแทนค่าลบด้วย 0 และแทนค่าบวกด้วย 1 เมื่ออ่านจากทริกเกอร์แทนค่าด้วยคู่สี (0, 1) การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ai จะต้องเกิดก่อนการ

เปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Co เสมอ ในทำนองเดียวกันการเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ Ai จะต้องเกิดหลังการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Co เสมอเช่นกัน

1.2) ความสัมพันธ์เชิงล็อคประเภทฟูลล็อค ระหว่างสัญญาณ Ai และ Co

จากคุณสมบัติความสัมพันธ์เชิงล็อค จากตารางคู่ความสัมพันธ์ของสัญญาณ Ai และ Co มีลักษณะแบบฟูลล็อค Ai0 > Co0 > Ai1 > Co1 โดนแทนค่าลบด้วย 0 และแทนค่าบวกด้วย 1 เมื่ออ่านจากทริกเกอร์แทนค่าด้วยคู่สี (0, 1) การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Bi จะต้องเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีขาว (-) ของ Co เสมอ ในทำนองเดียวกันการเปลี่ยนแปลง ทริกเกอร์สีเทา(+) ของ Bi จะต้องเกิดหลังการเปลี่ยนแปลงทริกเกอร์สีขาว (-) ของ Co เสมอ และเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ Co เสมอเช่นกัน

จากการทวนสอบลำดับของคู่ความสัมพันธ์เชิงล็อคของสัญญาณ Ai และ Co จนถึงลำดับขั้นการทำงานน้อยที่สุดที่แสดงถึงลำดับการเปลี่ยนแปลงสัญญาณที่เป็นไปได้ทั้งหมดของเอสทีจี ปรากฏว่ามีลำดับการเปลี่ยนแปลงของสัญญาณระหว่างคู่สัญญาณได้ถูกต้องทั้งหมด

2. การตรวจสอบคู่ความสัมพันธ์เชิงล็อคของคู่สัญญาณ Bi และ Co

จากขั้นตอนการทวนสอบแอลเอสเอสกับแอลอาร์เอสของคู่ความสัมพันธ์เชิงล็อค Bi และ Co ในตารางที่ 3.6 ตารางแสดงลำดับการเปลี่ยนแปลงสัญญาณจากการจำลองการทำงาน

ตารางที่ 3.6 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ Bi และ Co ขั้นที่ 1-30

<table>
<tr><th>ขั้นที่</th><th>ผลจากการจำลองการทำงาน</th><th>Ai</th><th>Bi</th><th>Co</th></tr>
<tr><td>1</td><td>Output > |1| Ai=|0|, Bi=|0|, Co =|0|</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>2</td><td>Output > |2| Ai=|1|, Bi=|0|, Co =|0|</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>3</td><td>Output > |3| Ai=|1|, Bi=|1|, Co =|0|</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>4</td><td>Output > |4| Ai=|1|, Bi=|1|, Co =|1|</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>5</td><td>Output > |5| Ai=|0|, Bi=|1|, Co =|1|</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>6</td><td>Output > |6| Ai=|0|, Bi=|0|, Co =|1|</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>7</td><td>Output > |7| Ai=|0|, Bi=|0|, Co =|0|</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>8</td><td>Output > |8| Ai=|1|, Bi=|0|, Co =|0|</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>9</td><td>Output > |9| Ai=|1|, Bi=|1|, Co =|0|</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>10</td><td>Output > |10| Ai=|1|, Bi=|1|, Co =|1|</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>11</td><td>Output > |11| Ai=|0|, Bi=|1|, Co =|1|</td><td>0</td><td>1</td><td>1</td></tr>
</table>

ตารางที่ 3.6 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ Bi และ Co ขั้นที่ 1-30 (ต่อ)

<table>
<tr><th>ขั้นที่</th><th>ผลจากการจำลองการทำงาน</th><th>Ai</th><th>Bi</th><th>Co</th></tr>
<tr><td>12</td><td>Output > |12| Ai=|0|, Bi=|0|, Co =|1|</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>13</td><td>Output > |13| Ai=|0|, Bi=|0|, Co =|0|</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>14</td><td>Output > |14| Ai=|1|, Bi=|0|, Co =|0|</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>15</td><td>Output > |15| Ai=|1|, Bi=|1|, Co =|0|</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>16</td><td>Output > |16| Ai=|1|, Bi=|1|, Co =|1|</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>17</td><td>Output > |17| Ai=|0|, Bi=|1|, Co =|1|</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>18</td><td>Output > |18| Ai=|0|, Bi=|0|, Co =|1|</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>19</td><td>Output > |19| Ai=|0|, Bi=|0|, Co =|0|</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>20</td><td>Output > |20| Ai=|0|, Bi=|1|, Co =|0|</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>21</td><td>Output > |21| Ai=|1|, Bi=|1|, Co =|0|</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>22</td><td>Output > |22| Ai=|1|, Bi=|1|, Co =|1|</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>23</td><td>Output > |23| Ai=|0|, Bi=|1|, Co =|1|</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>24</td><td>Output > |24| Ai=|0|, Bi=|0|, Co =|1|</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>25</td><td>Output > |25| Ai=|0|, Bi=|0|, Co =|0|</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>26</td><td>Output > |26| Ai=|1|, Bi=|0|, Co =|0|</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>27</td><td>Output > |27| Ai=|1|, Bi=|1|, Co =|0|</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>28</td><td>Output > |28| Ai=|1|, Bi=|1|, Co =|1|</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>29</td><td>Output > |29| Ai=|1|, Bi=|0|, Co =|1|</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>30</td><td>Output > |30| Ai=|0|, Bi=|0|, Co =|1|</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>...</td><td>...</td><td>...</td><td>...</td><td>...</td></tr>
</table>

2.1) ความสัมพันธ์เชิงล็อคประเภทเซมิล็อค ระหว่างสัญญาณ Bi และ Co

จากคุณสมบัติความสัมพันธ์เชิงล็อค จากตารางคู่ความสัมพันธ์ของสัญญาณ Bi และ Co มีลักษณะแบบเซมิล็อค Bi0 > Co0 > Bi1 โดนแทนค่าลบด้วย 0 และแทนค่าบวกด้วย 1 เมื่ออ่านจากทริกเกอร์แทนค่าด้วยคู่สี (0, 1) การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Bi จะต้องเกิดก่อน การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Co เสมอ ในทำนองเดียวกัน การเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ Bi จะต้องหลังก่อน การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Co เสมอเช่นกัน

2.2) ความสัมพันธ์เชิงล็อคประเภทฟูลล็อค ระหว่างสัญญาณ Bi และ Co

จากคุณสมบัติความสัมพันธ์เชิงล็อค จากตารางคู่ความสัมพันธ์ของสัญญาณ Bi และ Co มีลักษณะแบบฟูลล็อค Bi0 > Co0 > Bi1 > Co1 โดนแทนค่าลบด้วย 0 และแทนค่าบวกด้วย 1 เมื่อ

CHULALONGKORN UNIVERSITY

อ่านจากทริกเกอร์แทนค่าด้วยคู่สี (0, 1) การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Bi จะต้องเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีขาว (-) ของ Co เสมอ ในทำนองเดียวกัน การเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ Bi จะต้องหลัง การเปลี่ยนแปลงทริกเกอร์สีขาว (-) ของ Co เสมอและเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ Co เสมอเช่นกัน

จากการทวนสอบแอลอาร์เอสกับแอลเอสเอสพบว่าไม่มีการเปลี่ยนแปลงเอาต์พุต (Output transition) ที่ขัดแย้งกับเอสทีจี และเอสทีจีสามารถวนรอบซ้ำการทำงานได้ครบทุกโทเค็นสถานะ (State) จึงสามารถพิสูจน์ได้ว่าจะไม่เกิดข้อขัดแย้งกับแอลอาร์เอสของแต่ละคู่ความ สัมพันธ์หากยังวนซ้ำการทำงานต่อไป กล่าวคือ เอสทีจีใดๆ จะมีคุณสมบัติ ความทนทานก็ต่อเมื่อการเปลี่ยนแปลงสัญญาณใดๆ ที่เกิดขึ้น จะไม่ไปปิดกั้นการเกิดการเปลี่ยนแปลงสัญญาณสัญญาณอื่นๆ นั่นคือ สำหรับทุกสัญญาณ c ที่ไม่ใช่สัญญาณอินพุต แล้วทุกๆ การเปลี่ยนแปลงสัญญาณ c ซึ่งเกิดขึ้นโดยสัญญาณ a แล้ว การเปลี่ยนแปลงสัญญาณ c จะต้องมีความสัมพันธ์แบบอันดับกับสัญญาณ a’ ด้วย คุณสมบัติดังกล่าวนี้จะรับประกันว่าจะไม่เกิดปัญหาเงื่อนไขการแข่งขัน (Race condition) ซึ่งนำไปสู่การทำงานที่ผิดพลาดของวงจร

เมื่อทวนสอบผ่านคุณสมบัติไลฟ์เนสและคุณสมบัติความทนทานได้นั้น สามารถสรุปได้ว่าวงจรอสมวารที่ได้จากการสังเคราะห์นั้นๆ สามารถทำงานได้ถูกต้องตามพฤติกรรมที่ได้ออกแบบไว้ในเอสทีจี

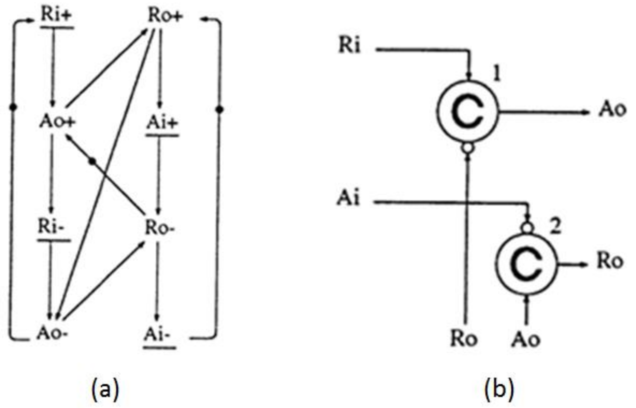
จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทที่ 4
การทวนสอบวงจรอสมวารตัวอย่างด้วยการจำลองลำดับสัญญาณในเอสทีจี

ในบทนี้จะกล่าวถึง การทวนสอบวงจรอสมวารตัวอย่างด้วยการจำลองลำดับสัญญาณในเอสทีจี ด้วยตัวอย่างวงจรอสมวารที่สังเคราะห์ได้ ทั้งหมด 3 วงจร ประกอบด้วยวงจรอสมวาร full, half, c-element แบบพิจารณาค่าที่ไม่ใช่อินพุต และวงจรที่สังเคราะห์ไม่ถูกต้อง 2 วงจร

4.1 การทวนสอบตัวอย่างวงจรอสมวาร full

ตัวอย่างวงจรอสมวาร full ประกอบด้วยเอสทีจีรูปที่ 4.1(a) และวงจรที่ได้จากการสังเคราะห์รูปที่ 4.1(b)



รูปที่ 4.1 เอสทีจี (a) [5] และวงจรที่ได้จากการสังเคราะห์ (b) [14] ของวงจรอสมวาร full

4.1.1 การแปลงพฤติกรรมในเอสทีจีและวงจรอสมวารที่สังเคราะห์ได้ของตัวอย่างวงจรอสมวาร full เป็นภาษาโพรเมล่า

จากขั้นตอนในข้อที่ 3.1 สามารถแปลงพฤติกรรมในเอสทีจีของวงจร full เป็นภาษาโพรเมล่า และขั้นตอนในข้อที่ 3.2 สามารถแปลงวงจรอสมวาร full เป็นภาษาโพรเมล่า ซึ่งรหัสภาษาโพรเมล่าที่แปลงได้จะแสดงในภาคผนวกรูปที่ ก-1 จากนั้นนำรหัสภาษาโพรเมล่าที่ได้จำลองการทำงานด้วยเครื่องมือสปินและบันทึกผลการจำลองการทำงานตามขั้นตอนในข้อ 3.3

4.1.2 การหาเอสทีเอสในเอสทีจีของตัวอย่างวงจรอสมวาร full

จากขั้นตอนในข้อที่ 3.4 สามารถหาเอสทีเอสในเอสทีจีของวงจรอสมวาร full ได้ทั้งหมด 5 เอสทีเอสดังนี้

เอสทีเอส 1. Ri1 > Ao1 > Ri0 > Ro1 > Ao0 > Ai1 > Ro0 > Ai0

เอสทีเอส 2. Ri1 > Ao1 > Ri0 > Ro1 > Ai1 > Ao0 > Ro0 > Ai0

เอสทีเอส 3. Ri1 > Ao1 > Ro1 > Ai1 > Ri0 > Ao0 > Ro0 > Ai0

เอสทีเอส 4. Ri1 > Ao1 > Ro1 > Ri0 > Ao0 > Ai1 > Ro0 > Ai0

เอสทีเอส 5. Ri1 > Ao1 > Ro1 > Ri0 > Ai1 > Ao0 > Ro0 > Ai0

4.1.3 การทวนสอบเอสทีเอสของตัวอย่างวงจรอสมวาร full

จากขั้นตอนในข้อที่ 3.5 การทวนสอบเอสทีเอสกับเอสเอสของวงจรอสมวาร full โดยในตารางที่ 4.2 จะคัดเฉพาะเอสเอสที่ถูกแบ่งด้วยรอบปัจจุบันและก่อนหน้าในขั้นนั้นๆ ที่สามารถตรวจพบครบทุกเอสทีเอสจากเอสทีจี

ตารางที่ 4.1 ผลการทวนสอบเอสทีเอสกับเอสเอสของตัวอย่างวงจรอสมวาร full

ขั้นแรก	ขั้นสุดท้าย	เอสเอสที่ถูกแบ่งด้วยรอบปัจจุบันและก่อนหน้า	เอสทีเอสที่พบ
1	8	Ri1 > Ao1 > Ro1 > Ai1 > Ri0 > Ao0 > Ro0 > Ai0 >	STS 3
593	608	Ai0 > Ao1 > Ri0 > Ro1 > Ao0 > Ai1 > Ro0 > Ai0 > Ri1 > Ao1 > Ro1 > Ri0 > Ai1 > Ao0 > Ro0 > Ai0 >	STS 5
777	792	Ri1 > Ao1 > Ro1 > Ri0 > Ai1 > Ao0 > Ro0 > Ai0 > Ri1 > Ao1 > Ro1 > Ri0 > Ao0 > Ai1 > Ro0 > Ai0 >	STS 4
817	832	Ri1 > Ao1 > Ro1 > Ai1 > Ri0 > Ao0 > Ro0 > Ai0 > Ri1 > Ao1 > Ri0 > Ro1 > Ao0 > Ai1 > Ro0 > Ai0 >	STS 1
1,745	**1,760**	**Ai0 > Ao1 > Ro1 > Ai1 > Ri0 > Ao0 > Ro0 > Ai0 > Ri1 > Ao1 > Ri0 > Ro1 > Ai1 > Ao0 > Ro0 > Ai0 >**	**STS 2**
…	…	…	**…**

ผลการทวนสอบเอสทีเอสกับเอสเอสของวงจรอสมวาร full จากตารางที่ 4.2 ผลลัพธ์แอลเอสเอสที่ได้คือ จำนวนรอบวนซ้ำการทำงานน้อยที่สุดของเอสเอสที่พบทุกรูปแบบของเอสทีเอสคือลำดับที่ 1,760 จากตารางที่ 4.1 สามารถอ่านเพิ่มเติมได้ในภาคผนวกตารางที่ ข.1

4.1.4 การหาแอลอาร์เอสในเอสทีจีของตัวอย่างวงจรอสมวาร full

จากขั้นตอนในข้อที่ 3.6 สามารถหาแอลอาร์เอสในเอสทีจีของวงจรอสมวาร full ได้ผลลัพธ์ 6 แอลอาร์เอส ดังในตารางที่ 4.1

ตารางที่ 4.2 ความสัมพันธ์เชิงล็อคจากเอสทีจีของวงจรอสมวาร full

ลำดับ	คู่สัญญาณ	แอลอาร์เอส	ประเภทความสัมพันธ์เชิงล็อค
1	Ao, Ro	Ao0 > Ro0 > Ao1	เซมิล็อค
2	Ri, Ao	Ri0 > Ao0 > Ri1	เซมิล็อค
3	Ro, Ai	Ro0 > Ai0 > Ro1	เซมิล็อค
4	Ao, Ro	Ao0 > Ro0 > Ao1 > Ro1	ฟูลล็อค
5	Ri, Ao	Ri0 > Ao0 > Ri1 > Ao1	ฟูลล็อค
6	Ro, Ai	Ro0 > Ai0 > Ro1 > Ai1	ฟูลล็อค

4.1.5 การทวนสอบแอลอาร์เอสของตัวอย่างวงจรอสมวาร full

จากขั้นตอนในข้อที่ 3.7 สามารถทวนสอบแอลอาร์เอสของวงจรอสมวาร full ของแต่ละคู่สัญญาณที่มีความสัมพันธ์เชิงล็อคร่วมกันดังนี้

1. การตรวจสอบคู่ความสัมพันธ์เชิงล็อคของคู่สัญญาณ Ao และ Ro

จากขั้นตอนการทวนสอบแอลเอสเอสกับแอลอาร์เอสของคู่ความสัมพันธ์เชิงล็อค Ao และ Ro ในตารางที่ 4.3 ตารางแสดงลำดับการเปลี่ยนแปลงสัญญาณจากการจำลองการทำงาน

ตารางที่ 4.3 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ Ao และ Ro ขั้นที่ 1-30

<table>
<tr><th>ขั้นที่</th><th>ผลจากการจำลองการทำงาน</th><th>Ri</th><th>Ro</th><th>Ai</th><th>Ao</th></tr>
<tr><td>1</td><td>Output > |1| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|0|</td><td>1</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>2</td><td>Output > |2| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|1|</td><td>1</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>3</td><td>Output > |3| Ri=|1|, Ro=|1|, Ai=|0|, Ao=|1|</td><td>1</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>4</td><td>Output > |4| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|1|</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>5</td><td>Output > |5| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|1|</td><td>0</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>6</td><td>Output > |6| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|0|</td><td>0</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>7</td><td>Output > |7| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|0|</td><td>0</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>8</td><td>Output > |8| Ri=|0|, Ro=|0|, Ai=|0|, Ao=|0|</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>9</td><td>Output > |9| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|0|</td><td>1</td><td>0</td><td>0</td><td>0</td></tr>
</table>

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ตารางที่ 4.3 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ Ao และ Ro ขั้นที่ 1-30 (ต่อ)

ขั้นที่	ผลจากการจำลองการทำงาน	Ri	Ro	Ai	Ao
10	Output > |10| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|1|	1	0	0	1
11	Output > |11| Ri=|1|, Ro=|1|, Ai=|0|, Ao=|1|	1	1	0	1
12	Output > |12| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|1|	1	1	1	1
13	Output > |13| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|1|	0	1	1	1
14	Output > |14| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|0|	0	1	1	0
15	Output > |15| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|0|	0	0	1	0
16	Output > |16| Ri=|1|, Ro=|0|, Ai=|1|, Ao=|0|	1	0	1	0
17	Output > |17| Ri=|1|, Ro=|0|, Ai=|1|, Ao=|1|	1	0	1	1
18	Output > |18| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|1|	0	0	1	1
19	Output > |19| Ri=|0|, Ro=|0|, Ai=|0|, Ao=|1|	0	0	0	1
20	Output > |20| Ri=|0|, Ro=|1|, Ai=|0|, Ao=|1|	0	1	0	1
21	Output > |21| Ri=|0|, Ro=|1|, Ai=|0|, Ao=|0|	0	1	0	0
22	Output > |22| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|0|	0	1	1	0
23	Output > |23| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|0|	1	1	1	0
24	Output > |24| Ri=|1|, Ro=|0|, Ai=|1|, Ao=|0|	1	0	1	0
25	Output > |25| Ri=|1|, Ro=|0|, Ai=|1|, Ao=|1|	1	0	1	1
26	Output > |26| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|1|	1	0	0	1
27	Output > |27| Ri=|0|, Ro=|0|, Ai=|0|, Ao=|1|	0	0	0	1
28	Output > |28| Ri=|0|, Ro=|1|, Ai=|0|, Ao=|1|	0	1	0	1
29	Output > |29| Ri=|0|, Ro=|1|, Ai=|0|, Ao=|0|	0	1	0	0
30	Output > |30| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|0|	0	1	1	0
...	...	...	...	...	...

1.1) ความสัมพันธ์เชิงล็อคประเภทเซมิล็อค ระหว่างสัญญาณ Ao และ Ro

จากคุณสมบัติความสัมพันธ์เชิงล็อค จากตารางคู่ความสัมพันธ์ของสัญญาณ Ao และ Ro มีลักษณะแบบเซมิล็อค Ao0 > Ro0 > Ao1 โดนแทนค่าลบด้วย 0 และแทนค่าบวกด้วย 1 เมื่ออ่านจากทริกเกอร์แทนค่าด้วยคู่สี (0, 1) การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ao จะต้องเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ro เสมอ ในทำนองเดียวกัน การเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ Ao จะต้องเกิดหลังการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ro เสมอเช่นกัน

1.2) ความสัมพันธ์เชิงล็อคประเภทฟูลล็อค ระหว่างสัญญาณ Ao และ Ro

จากคุณสมบัติความสัมพันธ์เชิงล็อค จากตารางคู่ความสัมพันธ์ของสัญญาณ Ao และ Ro มีลักษณะแบบฟูลล็อค Ao0 > Ro0 > Ao1 > Ro1 โดนแทนค่าลบด้วย 0 และแทนค่าบวกด้วย 1 เมื่ออ่านจากทริกเกอร์แทนค่าด้วยคู่สี (0, 1) การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ao จะต้องเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ro เสมอ ในทำนองเดียวกัน การเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ Ao จะต้องเกิดหลัง การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ro เสมอ และเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ Ro เสมอเช่นกัน

2. การตรวจสอบคู่ความสัมพันธ์เชิงล็อคของคู่สัญญาณ Ri และ Ao

จากขั้นตอนการทวนสอบแอลเอสเอสกับแอลอาร์เอสของคู่ความสัมพันธ์เชิงล็อค Ri และ Ao ในตารางที่ 4.4 ตารางแสดงลำดับการเปลี่ยนแปลงสัญญาณจากการจำลองการทำงาน

ตารางที่ 4.4 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ Ri และ Ao ขั้นที่ 1-30

ขั้นที่	ผลจากการจำลองการทำงาน	Ri	Ro	Ai	Ao
1	Output > |1| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|0|	1	0	0	0
2	Output > |2| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|1|	1	0	0	1
3	Output > |3| Ri=|1|, Ro=|1|, Ai=|0|, Ao=|1|	1	1	0	1
4	Output > |4| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|1|	1	1	1	1
5	Output > |5| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|1|	0	1	1	1
6	Output > |6| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|0|	0	1	1	0
7	Output > |7| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|0|	0	0	1	0
8	Output > |8| Ri=|0|, Ro=|0|, Ai=|0|, Ao=|0|	0	0	0	0
9	Output > |9| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|0|	1	0	0	0
10	Output > |10| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|1|	1	0	0	1
11	Output > |11| Ri=|1|, Ro=|1|, Ai=|0|, Ao=|1|	1	1	0	1
12	Output > |12| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|1|	1	1	1	1
13	Output > |13| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|1|	0	1	1	1
14	Output > |14| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|0|	0	1	1	0
15	Output > |15| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|0|	0	0	1	0
16	Output > |16| Ri=|1|, Ro=|0|, Ai=|1|, Ao=|0|	1	0	1	0
17	Output > |17| Ri=|1|, Ro=|0|, Ai=|1|, Ao=|1|	1	0	1	1
18	Output > |18| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|1|	0	0	1	1

ตารางที่ 4.4 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ Ri และ Ao ขั้นที่ 1-30 (ต่อ)

ขั้นที่	ผลจากการจำลองการทำงาน	Ri	Ro	Ai	Ao
19	Output > |19| Ri=|0|, Ro=|0|, Ai=|0|, Ao=|1|	0	0	0	1
20	Output > |20| Ri=|0|, Ro=|1|, Ai=|0|, Ao=|1|	0	1	0	1
21	Output > |21| Ri=|0|, Ro=|1|, Ai=|0|, Ao=|0|	0	1	0	0
22	Output > |22| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|0|	0	1	1	0
23	Output > |23| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|0|	1	1	1	0
24	Output > |24| Ri=|1|, Ro=|0|, Ai=|1|, Ao=|0|	1	0	1	0
25	Output > |25| Ri=|1|, Ro=|0|, Ai=|1|, Ao=|1|	1	0	1	1
26	Output > |26| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|1|	1	0	0	1
27	Output > |27| Ri=|0|, Ro=|0|, Ai=|0|, Ao=|1|	0	0	0	1
28	Output > |28| Ri=|0|, Ro=|1|, Ai=|0|, Ao=|1|	0	1	0	1
29	Output > |29| Ri=|0|, Ro=|1|, Ai=|0|, Ao=|0|	0	1	0	0
30	Output > |30| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|0|	0	1	1	0
...	...	...	...	...	...

2.1) ความสัมพันธ์เชิงล็อคประเภทเซมิล็อค ระหว่างสัญญาณ Ri และ Ao

จากคุณสมบัติความสัมพันธ์เชิงล็อค จากตารางคู่ความสัมพันธ์ของสัญญาณ Ri และ Ao มีลักษณะแบบเซมิล็อค Ri0 > Ao0 > Ri1 โดนแทนค่าลบด้วย 0 และแทนค่าบวกด้วย 1 เมื่ออ่านจากทริกเกอร์แทนค่าด้วยคู่สี (0, 1) การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ri จะต้องเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ao เสมอ ในทำนองเดียวกันการเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ Ri จะต้องเกิดหลังการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ao เสมอเช่นกัน

2.2) ความสัมพันธ์เชิงล็อคประเภทฟูลล็อค ระหว่างสัญญาณ Ri และ Ao

จากคุณสมบัติความสัมพันธ์เชิงล็อค จากตารางคู่ความสัมพันธ์ของสัญญาณ Ri และ Ao มีลักษณะแบบ ฟูลล็อค Ri0 > Ao0 > Ri1 > Ao1 โดนแทนค่าลบด้วย 0 และแทนค่าบวกด้วยค่า 1 เมื่ออ่านจากทริกเกอร์แทนค่าด้วยคู่สี (0, 1) การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ri จะต้องเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ao เสมอ ในทำนองเดียวกัน การเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ Ri จะต้องเกิดหลัง การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ao เสมอ และเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ Ao เสมอเช่นกัน

3. การตรวจสอบคู่ความสัมพันธ์เชิงล็อคของคู่สัญญาณ Ro และ Ai

จากขั้นตอนการทวนสอบแอลเอสเอสกับแอลอาร์เอสของคู่ความสัมพันธ์เชิงล็อค Ro และ Ai ในตารางที่ 4.5 ตารางแสดงลำดับการเปลี่ยนแปลงสัญญาณจากการจำลองการทำงาน

ตารางที่ 4.5 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ Ro และ Ai ขั้นที่ 1-30

ขั้นที่	ผลจากการจำลองการทำงาน	Ri	Ro	Ai	Ao
1	Output > |1| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|0|	1	0	0	0
2	Output > |2| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|1|	1	0	0	1
3	Output > |3| Ri=|1|, Ro=|1|, Ai=|0|, Ao=|1|	1	1	0	1
4	Output > |4| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|1|	1	1	1	1
5	Output > |5| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|1|	0	1	1	1
6	Output > |6| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|0|	0	1	1	0
7	Output > |7| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|0|	0	0	1	0
8	Output > |8| Ri=|0|, Ro=|0|, Ai=|0|, Ao=|0|	0	0	0	0
9	Output > |9| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|0|	1	0	0	0
10	Output > |10| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|1|	1	0	0	1
11	Output > |11| Ri=|1|, Ro=|1|, Ai=|0|, Ao=|1|	1	1	0	1
12	Output > |12| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|1|	1	1	1	1
13	Output > |13| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|1|	0	1	1	1
14	Output > |14| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|0|	0	1	1	0
15	Output > |15| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|0|	0	0	1	0
16	Output > |16| Ri=|1|, Ro=|0|, Ai=|1|, Ao=|0|	1	0	1	0
17	Output > |17| Ri=|1|, Ro=|0|, Ai=|1|, Ao=|1|	1	0	1	1
18	Output > |18| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|1|	0	0	1	1
19	Output > |19| Ri=|0|, Ro=|0|, Ai=|0|, Ao=|1|	0	0	0	1
20	Output > |20| Ri=|0|, Ro=|1|, Ai=|0|, Ao=|1|	0	1	0	1
21	Output > |21| Ri=|0|, Ro=|1|, Ai=|0|, Ao=|0|	0	1	0	0
22	Output > |22| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|0|	0	1	1	0
23	Output > |23| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|0|	1	1	1	0
24	Output > |24| Ri=|1|, Ro=|0|, Ai=|1|, Ao=|0|	1	0	1	0
25	Output > |25| Ri=|1|, Ro=|0|, Ai=|1|, Ao=|1|	1	0	1	1
26	Output > |26| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|1|	1	0	0	1
27	Output > |27| Ri=|0|, Ro=|0|, Ai=|0|, Ao=|1|	0	0	0	1

ตารางที่ 4.5 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ Ro และ Ai ขั้นที่ 1-30 (ต่อ)

ขั้นที่	ผลจากการจำลองการทำงาน	Ri	Ro	Ai	Ao
28	Output > |28| Ri=|0|, Ro=|1|, Ai=|0|, Ao=|1|	0	1	0	1
29	Output > |29| Ri=|0|, Ro=|1|, Ai=|0|, Ao=|0|	0	1	0	0
30	Output > |30| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|0|	0	1	1	0
...	...	...	...	...	...

3.1) ความสัมพันธ์เชิงล็อคประเภทเซมิล็อค ระหว่างสัญญาณ Ro และ Ai

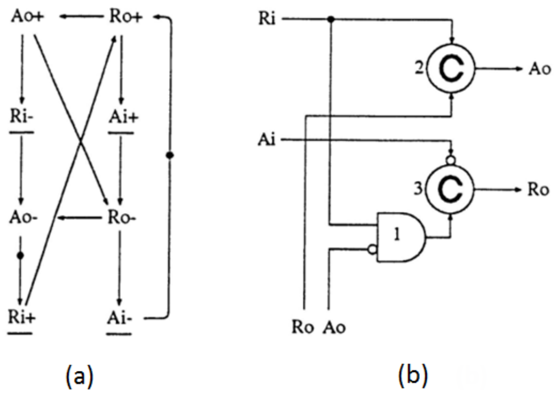
จากคุณสมบัติความสัมพันธ์เชิงล็อค จากตารางคู่ความสัมพันธ์ของสัญญาณ Ro และ Ai มีลักษณะแบบเซมิล็อค Ro0 > Ai0 > Ro1 โดนแทนค่าลบด้วย 0 และแทนค่าบวกด้วย 1 เมื่ออ่านจากทริกเกอร์แทนค่าด้วยคู่สี (0, 1) การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ro จะต้องเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ai เสมอ ในทำนองเดียวกัน การเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ Ro จะต้องเกิดหลังการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ai เสมอเช่นกัน

3.2) ความสัมพันธ์เชิงล็อคประเภทฟูลล็อค ระหว่างสัญญาณ Ro และ Ai

จากคุณสมบัติความสัมพันธ์เชิงล็อค จากตารางคู่ความสัมพันธ์ของสัญญาณ Ro และ Ai มีลักษณะแบบฟูลล็อค Ro0 > Ai0 > Ro1 > Ai1 โดนแทนค่าลบด้วย 0 และแทนค่าบวกด้วย 1 เมื่ออ่านจากทริกเกอร์แทนค่าด้วยคู่สี (0, 1) การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ro จะต้องเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ai เสมอในทำนองเดียวกัน การเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ Ro จะต้องเกิดหลังการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ai เสมอ และเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ Ai เสมอเช่นกัน

4.2 การทวนสอบตัวอย่างวงจรอสมวาร half

ตัวอย่างวงจรอสมวาร half ประกอบด้วยเอสทีจีรูปที่ 4.2(a) และวงจรที่ได้จากการสังเคราะห์รูปที่ 4.2(b)



รูปที่ 4.2 เอสทีจี (a) [5] และวงจรที่ได้จากการสังเคราะห์ (b) [14] ของวงจรอสมวาร half

4.2.1 การแปลงพฤติกรรมในเอสทีจีและวงจรอสมวารที่สังเคราะห์ได้ของตัวอย่างวงจรอสมวาร half เป็นภาษาโพรเมล่า

จากขั้นตอนในข้อที่ 3.1 สามารถแปลงพฤติกรรมในเอสทีจีของวงจร half เป็นภาษาโพรเมล่า และขั้นตอนในข้อที่ 3.2 สามารถแปลงวงจรอสมวาร half เป็นภาษาโพรเมล่า ซึ่งรหัสภาษาโพรเมล่าที่แปลงได้จะแสดงในภาคผนวก รูปที่ ก.2 จากนั้นนำรหัสภาษาโพรเมล่าที่ได้จำลองการทำงานด้วยเครื่องมือสปินและบันทึกผลการจำลองการทำงานตามขั้นตอนในข้อ 3.3

4.2.2 การหาเอสทีเอสในเอสทีจีของตัวอย่างวงจรอสมวาร half

จากขั้นตอนในข้อที่ 3.4 สามารถหาเอสทีเอสในเอสทีจีของวงจรอสมวาร half ได้ทั้งหมด 12 เอสทีเอสดังนี้

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

เอสทีเอส 1. Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ai0 > Ao0

เอสทีเอส 2. Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ai0

เอสทีเอส 3. Ri1 > Ro1 > Ai1 > Ao1 > Ro0 > Ai0 > Ri0 > Ao0

เอสทีเอส 4. Ri1 > Ro1 > Ai1 > Ao1 > Ro0 > Ri0 > Ai0 > Ao0

เอสทีเอส 5. Ri1 > Ro1 > Ai1 > Ao1 > Ro0 > Ri0 > Ao0 > Ai0

เอสทีเอส 6. Ri1 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ai0 > Ao0

เอสทีเอส 7. Ri1 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ao0 > Ai0

เอสทีเอส 8. Ri1 > Ro1 > Ao1 > Ai1 > Ro0 > Ai0 > Ri0 > Ao0

เอสทีเอส 9. Ri1 > Ro1 > Ao1 > Ai1 > Ro0 > Ri0 > Ai0 > Ao0

เอสทีเอส 10. Ri1 > Ro1 > Ao1 > Ai1 > Ro0 > Ri0 > Ao0 > Ai0

เอสทีเอส 11. Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ai0 > Ao0

เอสทีเอส 12. Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ai0

4.2.3 การทวนสอบเอสทีเอสของตัวอย่างวงจรอสมวาร half

จากขั้นตอนในข้อที่ 3.5 การทวนสอบเอสทีเอสกับเอสเอสของวงจรอสมวาร half โดยในตารางที่ 4.7 จะคัดเฉพาะเอสเอสที่ถูกแบ่งด้วยรอบปัจจุบันและก่อนหน้าในขั้นนั้นๆ ที่สามารถตรวจพบครบทุกเอสทีเอสจากเอสทีจี

ตารางที่ 4.6 ผลการทวนสอบเอสทีเอสกับเอสเอสของตัวอย่างวงจรอสมวาร half

ขั้นแรก	**ขั้นสุดท้าย**	**ลำดับการเปลี่ยนแปลงสัญญาณรอบปัจจุบันและก่อนหน้า**	**เอสทีเอสที่พบ**
1	10	Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ai0 >	STS 2
54	70	Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ai0 > Ri1 > Ro1 > Ai1 > Ao1 > Ro0 > Ai0 > Ri0 > Ao0 >	STS 3
84	100	Ai0 > Ro1 > Ai1 > Ao1 > Ro0 > Ri0 > Ao0 > Ai0 > Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ai0 > Ao0 >	STS 11
104	120	Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ai0 > Ao0 > Ri1 > Ro1 > Ai1 > Ao1 > Ro0 > Ri0 > Ao0 > Ai0 >	STS 5
134	150	Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ai0 > Ri1 > Ro1 > Ao1 > Ai1 > Ro0 > Ri0 > Ai0 > Ao0 >	STS 9
144	160	Ri1 > Ro1 > Ao1 > Ai1 > Ro0 > Ri0 > Ai0 > Ao0 > Ri1 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ao0 > Ai0 >	STS 7
204	220	Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ai0 > Ri1 > Ro1 > Ao1 > Ai1 > Ro0 > Ri0 > Ao0 > Ai0 >	STS 10
224	240	Ri1 > Ro1 > Ao1 > Ai1 > Ro0 > Ri0 > Ai0 > Ao0 > Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ai0 >	STS 12
234	250	Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ai0 > Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ai0 > Ao0 >	STS 1
304	320	Ai0 > Ro1 > Ao1 > Ai1 > Ro0 > Ri0 > Ai0 > Ao0 > Ri1 > Ro1 > Ai1 > Ao1 > Ro0 > Ri0 > Ai0 > Ao0 >	STS 4
414	430	Ai0 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ai0 > Ao0 > Ri1 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ai0 > Ao0 >	STS 6
564	**580**	**Ri1 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ai0 > Ao0 > Ri1 > Ro1 > Ao1 > Ai1 > Ro0 > Ai0 > Ri0 > Ao0 >**	**STS 8**
…	…	…	

ผลการทวนสอบเอสทีเอสกับเอสเอสของวงจรอสมวาร half จากตารางที่ 4.7 ผลลัพธ์แอลเอสเอสที่ได้คือ จำนวนรอบวนซ้ำการทำงานน้อยที่สุดของเอสเอสที่พบทุกรูปแบบของเอสทีเอส คือลำดับที่ 580 จากตารางที่ 4.7 สามารถอ่านเพิ่มเติมได้ในภาคผนวกตารางที่ ข.2

4.2.4 การหาแอลอาร์เอสในเอสทีจีของตัวอย่างวงจรอสมวาร half

จากขั้นตอนในข้อที่ 3.6 สามารถหาแอลอาร์เอสในเอสทีจีของวงจรอสมวาร half ได้ผลลัพธ์ 6 แอลอาร์เอส ดังในตารางที่ 4.6

ตารางที่ 4.7 ความสัมพันธ์เชิงล็อคจากเอสทีจีของวงจรอสมวาร half

ลำดับ	คู่สัญญาณ	แอลอาร์เอส	ประเภทความสัมพันธ์เชิงล็อค
1	Ri, Ao	Ri0 > Ao0 > Ri1	เซมิล็อค
2	Ro, Ai	Ro0 > Ai0 > Ro1	เซมิล็อค
3	Ro, Ao	Ro0 > Ao0 > Ro1	เซมิล็อค
4	Ri, Ao	Ri0 > Ao0 > Ri1 > Ao1	ฟูลล็อค
5	Ro, Ai	Ro0 > Ai0 > Ro1 > Ai1	ฟูลล็อค
6	Ro, Ao	Ro0 > Ao0 > Ro1 > Ao1	ฟูลล็อค

4.2.5 การทวนสอบแอลอาร์เอสของตัวอย่างวงจรอสมวาร half

จากขั้นตอนในข้อที่ 3.7 สามารถทวนสอบแอลอาร์เอสของวงจรอสมวาร half ของแต่ละคู่สัญญาณที่มีความสัมพันธ์เชิงล็อคร่วมกันดังนี้

1. การตรวจสอบคู่ความสัมพันธ์เชิงล็อคของคู่สัญญาณ Ri และ Ao

จากขั้นตอนการทวนสอบแอลเอสเอสกับแอลอาร์เอสของคู่ความสัมพันธ์เชิงล็อค Ri และ Ao ในตารางที่ 4.8 ตารางแสดงลำดับการเปลี่ยนแปลงสัญญาณจากการจำลองการทำงาน

ตารางที่ 4.8 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ Ri และ Ao ขั้นที่ 1-30

ขั้นที่	ผลจากการจำลองการทำงาน	Ri	Ro	Ai	Ao
1	Output > |1| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|0|	1	0	0	0
2	Output > |2| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|0|	1	0	0	0
3	Output > |3| Ri=|1|, Ro=|1|, Ai=|0|, Ao=|0|	1	1	0	0
4	Output > |4| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|0|	1	1	1	0
5	Output > |5| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|1|	1	1	1	1

ตารางที่ 4.8 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ Ri และ Ao ขั้นที่ 1-30 (ต่อ)

ขั้นที่	ผลจากการจำลองการทำงาน	Ri	Ro	Ai	Ao
6	Output > |6| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|1|	0	1	1	1
7	Output > |7| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|1|	0	1	1	1
8	Output > |8| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|1|	0	0	1	1
9	Output > |9| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|0|	0	0	1	0
10	Output > |10| Ri=|0|, Ro=|0|, Ai=|0|, Ao=|0|	0	0	0	0
11	Output > |11| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|0|	1	0	0	0
12	Output > |12| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|0|	1	0	0	0
13	Output > |13| Ri=|1|, Ro=|1|, Ai=|0|, Ao=|0|	1	1	0	0
14	Output > |14| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|0|	1	1	1	0
15	Output > |15| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|1|	1	1	1	1
16	Output > |16| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|1|	0	1	1	1
17	Output > |17| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|1|	0	1	1	1
18	Output > |18| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|1|	0	0	1	1
19	Output > |19| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|0|	0	0	1	0
20	Output > |20| Ri=|1|, Ro=|0|, Ai=|1|, Ao=|0|	1	0	1	0
21	Output > |21| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|0|	1	0	0	0
22	Output > |22| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|0|	1	0	0	0
23	Output > |23| Ri=|1|, Ro=|1|, Ai=|0|, Ao=|0|	1	1	0	0
24	Output > |24| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|0|	1	1	1	0
25	Output > |25| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|1|	1	1	1	1
26	Output > |26| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|1|	0	1	1	1
27	Output > |27| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|1|	0	1	1	1
28	Output > |28| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|1|	0	0	1	1
29	Output > |29| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|0|	0	0	1	0
30	Output > |30| Ri=|1|, Ro=|0|, Ai=|1|, Ao=|0|	1	0	1	0
…	…	…	…	…	…

1.1) ความสัมพันธ์เชิงล็อคประเภทเซมิล็อค ระหว่างสัญญาณ Ri และ Ao

จากคุณสมบัติความสัมพันธ์เชิงล็อค จากตารางคู่ความสัมพันธ์ของสัญญาณ Ri และ Ao มีลักษณะแบบเซมิล็อค Ri0 > Ao0 > Ri1 โดนแทนค่าลบด้วย 0 และแทนค่าบวกด้วย 1 เมื่ออ่านจากทริกเกอร์แทนค่าด้วยคู่สี (0, 1) การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ri จะต้องเกิดก่อนการ

เปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ao เสมอ ในทำนองเดียวกัน การเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ Ri จะต้องเกิดหลัง การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ao เสมอเช่นกัน

1.2) ความสัมพันธ์เชิงล็อคประเภทฟูลล็อค ระหว่างสัญญาณ Ri และ Ao

จากคุณสมบัติความสัมพันธ์เชิงล็อค จากตารางคู่ความสัมพันธ์ของสัญญาณ Ri และ Ao มีลักษณะแบบ ฟูลล็อค Ri0 > Ao0 > Ri1 > Ao1 โดนแทนค่าลบด้วย 0 และแทนค่าบวกด้วย 1 เมื่ออ่านจากทริกเกอร์แทนค่าด้วยคู่สี (0, 1) การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ri จะต้องเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ao เสมอ ในทำนองเดียวกัน การเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ Ri จะต้องเกิดหลัง การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ao เสมอ และเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ Ao เสมอเช่นกัน

2. การตรวจสอบคู่ความสัมพันธ์เชิงล็อคของคู่สัญญาณ Ro และ Ai

จากขั้นตอนการทวนสอบแอลเอสเอสกับแอลอาร์เอสของคู่ความสัมพันธ์เชิงล็อค Ro และ Ai ในตารางที่ 4.9 ตารางแสดงลำดับการเปลี่ยนแปลงสัญญาณจากการจำลองการทำงาน

ตารางที่ 4.9 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ Ro และ Ai ขั้นที่ 1-30

ขั้นที่	ผลจากการจำลองการทำงาน	Ri	Ro	Ai	Ao
1	Output > |1| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|0|	1	0	0	0
2	Output > |2| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|0|	1	0	0	0
3	Output > |3| Ri=|1|, Ro=|1|, Ai=|0|, Ao=|0|	1	1	0	0
4	Output > |4| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|0|	1	1	1	0
5	Output > |5| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|1|	1	1	1	1
6	Output > |6| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|1|	0	1	1	1
7	Output > |7| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|1|	0	1	1	1
8	Output > |8| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|1|	0	0	1	1
9	Output > |9| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|0|	0	0	1	0
10	Output > |10| Ri=|0|, Ro=|0|, Ai=|0|, Ao=|0|	0	0	0	0
11	Output > |11| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|0|	1	0	0	0
12	Output > |12| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|0|	1	0	0	0
13	Output > |13| Ri=|1|, Ro=|1|, Ai=|0|, Ao=|0|	1	1	0	0
14	Output > |14| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|0|	1	1	1	0
15	Output > |15| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|1|	1	1	1	1

ตารางที่ 4.9 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ Ro และ Ai ขั้นที่ 1-30 (ต่อ)

<table>
<tr><th>ขั้นที่</th><th>ผลจากการจำลองการทำงาน</th><th>Ri</th><th>Ro</th><th>Ai</th><th>Ao</th></tr>
<tr><td>16</td><td>Output > |16| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|1|</td><td>0</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>17</td><td>Output > |17| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|1|</td><td>0</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>18</td><td>Output > |18| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|1|</td><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>19</td><td>Output > |19| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|0|</td><td>0</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>20</td><td>Output > |20| Ri=|1|, Ro=|0|, Ai=|1|, Ao=|0|</td><td>1</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>21</td><td>Output > |21| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|0|</td><td>1</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>22</td><td>Output > |22| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|0|</td><td>1</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>23</td><td>Output > |23| Ri=|1|, Ro=|1|, Ai=|0|, Ao=|0|</td><td>1</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>24</td><td>Output > |24| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|0|</td><td>1</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>25</td><td>Output > |25| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|1|</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>26</td><td>Output > |26| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|1|</td><td>0</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>27</td><td>Output > |27| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|1|</td><td>0</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>28</td><td>Output > |28| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|1|</td><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>29</td><td>Output > |29| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|0|</td><td>0</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>30</td><td>Output > |30| Ri=|1|, Ro=|0|, Ai=|1|, Ao=|0|</td><td>1</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>...</td><td>...</td><td>...</td><td>...</td><td>...</td><td>...</td></tr>
</table>

2.1) ความสัมพันธ์เชิงล็อคประเภทเซมิล็อค ระหว่างสัญญาณ Ro และ Ai

จากคุณสมบัติความสัมพันธ์เชิงล็อค จากตารางคู่ความสัมพันธ์ของสัญญาณ Ro และ Ai มีลักษณะแบบเซมิล็อค Ro0 > Ai0 > Ro1 โดนแทนค่าลบด้วย 0 และแทนค่าบวกด้วย 1 เมื่ออ่านจากทริกเกอร์แทนค่าด้วยคู่สี (0, 1) การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ro จะต้องเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ai เสมอ ในทำนองเดียวกัน   การปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ Ro จะต้องเกิดหลัง การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ai เสมอเช่นกัน

2.2) ความสัมพันธ์เชิงล็อคประเภทฟูลล็อค ระหว่างสัญญาณ Ro และ Ai

จากคุณสมบัติความสัมพันธ์เชิงล็อค จากตารางคู่ความสัมพันธ์ของสัญญาณ Ro และ Ai มีลักษณะแบบฟูลล็อค Ro0 > Ai0 > Ro1 > Ai1 โดนแทนค่าลบด้วย 0 และแทนค่าบวกด้วย 1 เมื่ออ่านจากทริกเกอร์แทนค่าด้วยคู่สี (0, 1) การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ro จะต้องเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ai เสมอ ในทำนองเดียวกันการปลี่ยนแปลงทริกเกอร์สีเทา(+)

ของ Ro จะต้องเกิดหลัง การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ai เสมอ และเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ Ai เสมอเช่นกัน

3. การตรวจสอบคู่ความสัมพันธ์เชิงล็อคของคู่สัญญาณ Ro และ Ao

จากขั้นตอนการทวนสอบแอลเอสเอสกับแอลอาร์เอสของคู่ความสัมพันธ์เชิงล็อค Ro และ Ao ในตารางที่ 4.10 ตารางแสดงลำดับการเปลี่ยนแปลงสัญญาณจากการจำลองการทำงาน

ตารางที่ 4.10 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ Ro และ Ao ขั้นที่ 1-30

ขั้นที่	ผลจากการจำลองการทำงาน	Ri	Ro	Ai	Ao
1	Output > |1| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|0|	1	0	0	0
2	Output > |2| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|0|	1	0	0	0
3	Output > |3| Ri=|1|, Ro=|1|, Ai=|0|, Ao=|0|	1	1	0	0
4	Output > |4| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|0|	1	1	1	0
5	Output > |5| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|1|	1	1	1	1
6	Output > |6| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|1|	0	1	1	1
7	Output > |7| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|1|	0	1	1	1
8	Output > |8| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|1|	0	0	1	1
9	Output > |9| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|0|	0	0	1	0
10	Output > |10| Ri=|0|, Ro=|0|, Ai=|0|, Ao=|0|	0	0	0	0
11	Output > |11| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|0|	1	0	0	0
12	Output > |12| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|0|	1	0	0	0
13	Output > |13| Ri=|1|, Ro=|1|, Ai=|0|, Ao=|0|	1	1	0	0
14	Output > |14| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|0|	1	1	1	0
15	Output > |15| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|1|	1	1	1	1
16	Output > |16| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|1|	0	1	1	1
17	Output > |17| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|1|	0	1	1	1
18	Output > |18| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|1|	0	0	1	1
19	Output > |19| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|0|	0	0	1	0
20	Output > |20| Ri=|1|, Ro=|0|, Ai=|1|, Ao=|0|	1	0	1	0
21	Output > |21| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|0|	1	0	0	0
22	Output > |22| Ri=|1|, Ro=|0|, Ai=|0|, Ao=|0|	1	0	0	0
23	Output > |23| Ri=|1|, Ro=|1|, Ai=|0|, Ao=|0|	1	1	0	0
24	Output > |24| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|0|	1	1	1	0

ตารางที่ 4.10 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ Ro และ Ao ขั้นที่ 1-30 (ต่อ)

ขั้นที่	ผลจากการจำลองการทำงาน	Ri	Ro	Ai	Ao
25	Output > |25| Ri=|1|, Ro=|1|, Ai=|1|, Ao=|1|	1	1	1	1
26	Output > |26| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|1|	0	1	1	1
27	Output > |27| Ri=|0|, Ro=|1|, Ai=|1|, Ao=|1|	0	1	1	1
28	Output > |28| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|1|	0	0	1	1
29	Output > |29| Ri=|0|, Ro=|0|, Ai=|1|, Ao=|0|	0	0	1	0
30	Output > |30| Ri=|1|, Ro=|0|, Ai=|1|, Ao=|0|	1	0	1	0
...	...	...	...	...	...

3.1) ความสัมพันธ์เชิงล็อคประเภทเซมิล็อค ระหว่างสัญญาณ Ro และ Ao

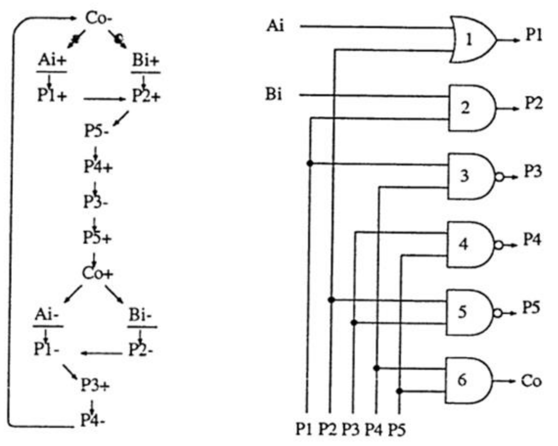
จากคุณสมบัติความสัมพันธ์เชิงล็อค จากตารางคู่ความสัมพันธ์ของสัญญาณ Ro และ Ao มีลักษณะแบบเซมิล็อค Ro0 > Ao0 > Ro1 โดนแทนค่าลบด้วย 0 และแทนค่าบวกด้วย 1 เมื่ออ่านจากทริกเกอร์แทนค่าด้วยคู่สี (0, 1) การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ro จะต้องเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ao เสมอ ในทำนองเดียวกัน การเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ Ro จะต้องเกิดหลังการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ao เสมอเช่นกัน

3.2) ความสัมพันธ์เชิงล็อคประเภทฟูลล็อค ระหว่างสัญญาณ Ro และ Ao

จากคุณสมบัติความสัมพันธ์เชิงล็อค จากตารางคู่ความสัมพันธ์ของสัญญาณ Ro และ Ao มีลักษณะแบบฟูลล็อค Ro0 > Ao0 > Ro1 > Ao1 โดนแทนค่าลบด้วย 0 และแทนค่าบวกด้วย 1 เมื่ออ่านจากทริกเกอร์แทนค่าด้วยคู่สี (0, 1) การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ro จะต้องเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ao เสมอ ในทำนองเดียวกัน การเปลี่ยนแปลงทริกเกอร์ สีเทา(+) ของ Ro จะต้องเกิดหลังการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ Ao เสมอ และเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ Ao เสมอเช่นกัน

4.3 การทวนสอบตัวอย่างวงจรอสมวาร C-element แบบพิจารณาค่าที่ไม่ใช่อินพุต

ตัวอย่างวงจรอสมวาร C-element แบบพิจารณาค่าที่ไม่ใช่อินพุต ประกอบด้วยเอสทีจีรูปที่ 4.3(a) และวงจรที่ได้จากการสังเคราะห์รูปที่ 4.3(b)



(a) (b)

รูปที่ 4.3 เอสทีจี (a) [5] และวงจรที่ได้จากการสังเคราะห์ (b) [14] ของวงจรอสมวาร C-element แบบพิจารณาค่าที่ไม่ใช่อินพุต

4.3.1 การแปลงพฤติกรรมในเอสทีจีและวงจรอสมวารที่สังเคราะห์ได้ของตัวอย่างวงจรอสมวาร C-element เป็นภาษาโพรเมล่า

จากขั้นตอนในข้อที่ 3.1 สามารถแปลงพฤติกรรมในเอสทีจีของวงจร C-element แบบพิจารณาค่าที่ไม่ใช่อินพุตเป็นภาษาโพรเมล่า และขั้นตอนในข้อที่ 3.2 สามารถแปลงวงจรอสมวาร C-element แบบพิจารณาค่าที่ไม่ใช่อินพุตเป็นภาษาโพรเมล่า ซึ่งรหัสภาษาโพรเมล่าที่แปลงได้จะแสดงในภาคผนวกรูปที่ ก.3 จากนั้นนำรหัสภาษาโพรเมล่าที่ได้จำลองการทำงานด้วยเครื่องมือสปินและบันทึกผลการจำลองการทำงานตามขั้นตอนในข้อ 3.3

4.3.2 การหาเอสทีเอสในเอสทีจีของตัวอย่างวงจรอสมวาร C-element แบบพิจารณาค่าที่ไม่ใช่อินพุต

จากขั้นตอนในข้อที่ 3.4 สามารถหาเอสทีเอสในเอสทีจีของวงจรอสมวาร C-element แบบพิจารณาค่าที่ไม่ใช่อินพุต ได้ทั้งหมด 9 เอสทีเอสดังนี้

เอสทีเอส 1. Ai1 > Bi1 > P11 > P21 > P50 > P41 > P30 > P51 >
Co1 > Ai0 > Bi0 > P20 > P10 > P31 > P40 > Co0

เอสทีเอส 2. Ai1 > Bi1 > P11 > P21 > P50 > P41 > P30 > P51 >
Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0

เอสทีเอส 3. Ai1 > Bi1 > P11 > P21 > P50 > P41 > P30 > P51 >
Co1 > Bi0 > P20 > Ai0 > P10 > P31 > P40 > Co0

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

เอสทีเอส 4. Ai1 > P11 > Bi1 > P21 > P50 > P41 > P30 > P51 >
Co1 > Ai0 > Bi0 > P20 > P10 > P31 > P40 > Co0

เอสทีเอส 5. Ai1 > P11 > Bi1 > P21 > P50 > P41 > P30 > P51 >
Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0

เอสทีเอส 6. Ai1 > P11 > Bi1 > P21 > P50 > P41 > P30 > P51 >
Co1 > Bi0 > P20 > Ai0 > P10 > P31 > P40 > Co0

เอสทีเอส 7. Bi1 > Ai1 > P11 > P21 > P50 > P41 > P30 > P51 >
Co1 > Ai0 > Bi0 > P20 > P10 > P31 > P40 > Co0

เอสทีเอส 8. Bi1 > Ai1 > P11 > P21 > P50 > P41 > P30 > P51 >
Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0

เอสทีเอส 9. Bi1 > Ai1 > P11 > P21 > P50 > P41 > P30 > P51 >
Co1 > Bi0 > P20 > Ai0 > P10 > P31 > P40 > Co0

4.3.3 การทวนสอบเอสทีเอสของตัวอย่างวงจรอสมวาร C-element แบบพิจารณาค่าที่ไม่ใช่อินพุต

จากขั้นตอนในข้อที่ 3.5 การทวนสอบเอสทีเอสกับเอสเอสของวงจรอสมวาร C-element แบบพิจารณาค่าที่ไม่ใช่อินพุต โดยในตารางที่ 4.12 จะคัดเฉพาะเอสเอสที่ถูกแบ่งด้วยรอบปัจจุบันและก่อนหน้าในขั้นนั้นๆ ที่สามารถตรวจพบครบทุกเอสทีเอสจากเอสทีจี

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ตารางที่ 4.11 ผลการทวนสอบเอสทีเอสกับเอสเอสของตัวอย่างวงจรอสมวาร C-element แบบพิจารณาค่าที่ไม่ใช่อินพุต

ขั้นแรก	ขั้นสุดท้าย	ลำดับการเปลี่ยนแปลงสัญญาณรอบปัจจุบันและก่อนหน้า	เอสทีเอสที่พบ
1	33	P11 > Bi1 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0 > Ai1 > Bi1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0 > Bi1 >	STS 2
18	49	Bi1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0 > Bi1 > Ai1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Ai0 > Bi0 > P20 > P10 > P31 > P40 > Co0 > Bi1 >	STS 7

ตารางที่ 4.12 ผลการทวนสอบเอสทีเอสกับเอสเอสของตัวอย่างวงจรอสมวาร C-element แบบพิจารณาค่าที่ไม่ใช่อินพุต (ต่อ)

ขั้นแรก	ขั้นสุดท้าย	ลำดับการเปลี่ยนแปลงสัญญาณรอบปัจจุบันและก่อนหน้า	เอสทีเอสที่พบ
82	113	Bi1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0 > Bi1 > Ai1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0 > Ai1 >	STS 8
130	161	Ai1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0 > Ai1 > Bi1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Ai0 > Bi0 > P20 > P10 > P31 > P40 > Co0 > Ai1 >	STS 1
146	177	Bi1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Ai0 > Bi0 > P20 > P10 > P31 > P40 > Co0 > Ai1 > P11 > Bi1 > P21 > P50 > P41 > P30 > P51 > Co1 > Ai0 > Bi0 > P20 > P10 > P31 > P40 > Co0 > Ai1 >	STS 4
226	257	Bi1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Ai0 > Bi0 > P20 > P10 > P31 > P40 > Co0 > Ai1 > P11 > Bi1 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0 > Ai1 >	STS 5
258	289	Bi1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Ai0 > Bi0 > P20 > P10 > P31 > P40 > Co0 > Ai1 > Bi1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > P20 > Ai0 > P10 > P31 > P40 > Co0 > Ai1 >	STS3
290	321	P11 > Bi1 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0 > Bi1 > Ai1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > P20 > Ai0 > P10 > P31 > P40 > Co0 > Ai1 >	STS9
1,474	**1,505**	**Ai1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Ai0 > Bi0 > P20 > P10 > P31 > P40 > Co0 > Ai1 > P11 > Bi1 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > P20 > Ai0 > P10 > P31 > P40 > Co0 > Bi1 >**	**STS6**
...	...	...	

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ผลการทวนสอบเอสทีเอสกับเอสเอสของวงจรอสมวาร C-element แบบพิจารณาค่าที่ไม่ใช่อินพุตจากตารางที่ 4.12 ผลลัพธ์แอลเอสเอสที่ได้คือ จำนวนรอบวนซ้ำการทำงานน้อยที่สุดของเอสเอสที่พบทุกรูปแบบของเอสทีเอสคือลำดับที่ 1,505 จากตารางที่ 4.12 สามารถอ่านเพิ่มเติมได้ในภาคผนวกตารางที่ ข.3

4.3.4 การหาแอลอาร์เอสในเอสทีจีของตัวอย่างวงจรอสมวาร C-element แบบพิจารณาค่าที่ไม่ใช่อินพุต

จากขั้นตอนในข้อที่ 3.6 สามารถหาแอลอาร์เอสในเอสทีจีของวงจรอสมวาร C-element แบบพิจารณาค่าที่ไม่ใช่อินพุตได้ แต่ในตัวอย่างวงจรนี้เป็นวงจรเดียวกันกับบทที่ 3 รูปภาพที่ 3.10 แต่มีความแตกต่างกันที่เอสทีจีเป็นแบบแสดงค่าที่ไม่ใช่ค่าอินพุต ดังนั้นจึงขอยกตัวอย่างของแอลอาร์เอสของคู่สัญญาณใดๆ ที่ไม่ใช่ค่าอินพุตเพียง 2 คู่สัญญาณเท่านั้น ดังในตารางที่ 4.11

ตารางที่ 4.12 ความสัมพันธ์เชิงล็อคจากเอสทีจีของวงจรอสมวาร C-element แบบพิจารณาค่าที่ไม่ใช่อินพุต

ลำดับ	คู่สัญญาณ	แอลอาร์เอส	ประเภทความสัมพันธ์เชิงล็อค
1	P1, P4	P10 > P40 > P11	เซมิล็อค
2	P2, P4	P10 > P40 > P11	เซมิล็อค
3	P1, P4	P10 > P40 > P11 > P41	ฟูลล็อค
4	P2, P4	P10 > P40 > P11 > P41	ฟูลล็อค
...	...	...	...

4.3.5 การทวนสอบแอลอาร์เอสของตัวอย่างวงจรอสมวาร C-element แบบพิจารณาค่าที่ไม่ใช่อินพุต

จากขั้นตอนในข้อที่ 3.7 สามารถทวนสอบแอลอาร์เอสของวงจรอสมวาร C-element แบบพิจารณาค่าที่ไม่ใช่อินพุตของแต่ละคู่สัญญาณที่มีความสัมพันธ์เชิงล็อคร่วมกันดังนี้

1. การตรวจสอบคู่ความสัมพันธ์เชิงล็อคของคู่สัญญาณ P1 และ P4

จากขั้นตอนการทวนสอบแอลเอสเอสกับแอลอาร์เอสของคู่ความสัมพันธ์เชิงล็อค P1 และ P4 กับ ในตารางที่ 4.13 ตารางแสดงลำดับการเปลี่ยนแปลงสัญญาณจากการจำลองการทำงาน

ตารางที่ 4.13 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ P1 และ P4 ขั้นที่ 1-30

ขั้นที่	ผลจากการจำลองการทำงาน	P1	P2	P3	P4
1	Output > |1| Ai=|1|, Bi=|0|, P1=|0|, P2=|0|, P3=|1|, P4=|0|, P5=|1|, Co=|0|,	0	0	1	0
2	Output > |2| Ai=|1|, Bi=|0|, P1=|1|, P2=|0|, P3=|1|, P4=|0|, P5=|1|, Co=|0|,	1	0	1	0
3	Output > |3| Ai=|1|, Bi=|1|, P1=|1|, P2=|0|, P3=|1|, P4=|0|, P5=|1|, Co=|0|,	1	0	1	0
4	Output > |4| Ai=|1|, Bi=|1|, P1=|1|, P2=|1|, P3=|1|, P4=|0|, P5=|1|, Co=|0|,	1	1	1	0
5	Output > |5| Ai=|1|, Bi=|1|, P1=|1|, P2=|1|, P3=|1|, P4=|0|, P5=|0|, Co=|0|,	1	1	1	0
6	Output > |6| Ai=|1|, Bi=|1|, P1=|1|, P2=|1|, P3=|1|, P4=|1|, P5=|0|, Co=|0|,	1	1	1	1
7	Output > |7| Ai=|1|, Bi=|1|, P1=|1|, P2=|1|, P3=|0|, P4=|1|, P5=|0|, Co=|0|,	1	1	0	1
8	Output > |8| Ai=|1|, Bi=|1|, P1=|1|, P2=|1|, P3=|0|, P4=|1|, P5=|1|, Co=|0|,	1	1	0	1
9	Output > |9| Ai=|1|, Bi=|1|, P1=|1|, P2=|1|, P3=|0|, P4=|1|, P5=|1|, Co=|1|,	1	1	0	1
10	Output > |10| Ai=|1|, Bi=|0|, P1=|1|, P2=|1|, P3=|0|, P4=|1|, P5=|1|, Co=|1|,	1	1	0	1
11	Output > |11| Ai=|0|, Bi=|0|, P1=|1|, P2=|1|, P3=|0|, P4=|1|, P5=|1|, Co=|1|,	1	1	0	1
12	Output > |12| Ai=|0|, Bi=|0|, P1=|1|, P2=|0|, P3=|0|, P4=|1|, P5=|1|, Co=|1|,	1	0	0	1
13	Output > |13| Ai=|0|, Bi=|0|, P1=|0|, P2=|0|, P3=|0|, P4=|1|, P5=|1|, Co=|1|,	0	0	0	1
14	Output > |14| Ai=|0|, Bi=|0|, P1=|0|, P2=|0|, P3=|1|, P4=|1|, P5=|1|, Co=|1|,	0	0	1	1
15	Output > |15| Ai=|0|, Bi=|0|, P1=|0|, P2=|0|, P3=|1|, P4=|0|, P5=|1|, Co=|1|,	0	0	1	0
16	Output > |16| Ai=|0|, Bi=|0|, P1=|0|, P2=|0|, P3=|1|, P4=|0|, P5=|1|, Co=|0|,	0	0	1	0

ตารางที่ 4.13 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ P1 และ P4 ขั้นที่ 1-30 (ต่อ)

<table>
<tr><th>ขั้นที่</th><th>ผลจากการจำลองการทำงาน</th><th>P1</th><th>P2</th><th>P3</th><th>P4</th></tr>
<tr><td>17</td><td>Output > |17| Ai=|1|, Bi=|0|, P1=|0|, P2=|0|, P3=|1|, P4=|0|, P5=|1|, Co=|0|,</td><td>0</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>18</td><td>Output > |18| Ai=|1|, Bi=|1|, P1=|0|, P2=|0|, P3=|1|, P4=|0|, P5=|1|, Co=|0|,</td><td>0</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>19</td><td>Output > |19| Ai=|1|, Bi=|1|, P1=|1|, P2=|0|, P3=|1|, P4=|0|, P5=|1|, Co=|0|,</td><td>1</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>20</td><td>Output > |20| Ai=|1|, Bi=|1|, P1=|1|, P2=|1|, P3=|1|, P4=|0|, P5=|1|, Co=|0|,</td><td>1</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>21</td><td>Output > |21| Ai=|1|, Bi=|1|, P1=|1|, P2=|1|, P3=|1|, P4=|0|, P5=|0|, Co=|0|,</td><td>1</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>22</td><td>Output > |22| Ai=|1|, Bi=|1|, P1=|1|, P2=|1|, P3=|1|, P4=|1|, P5=|0|, Co=|0|,</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>23</td><td>Output > |23| Ai=|1|, Bi=|1|, P1=|1|, P2=|1|, P3=|0|, P4=|1|, P5=|0|, Co=|0|,</td><td>1</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>24</td><td>Output > |24| Ai=|1|, Bi=|1|, P1=|1|, P2=|1|, P3=|0|, P4=|1|, P5=|1|, Co=|0|,</td><td>1</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>25</td><td>Output > |25| Ai=|1|, Bi=|1|, P1=|1|, P2=|1|, P3=|0|, P4=|1|, P5=|1|, Co=|1|,</td><td>1</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>26</td><td>Output > |26| Ai=|1|, Bi=|0|, P1=|1|, P2=|1|, P3=|0|, P4=|1|, P5=|1|, Co=|1|,</td><td>1</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>27</td><td>Output > |27| Ai=|0|, Bi=|0|, P1=|1|, P2=|1|, P3=|0|, P4=|1|, P5=|1|, Co=|1|,</td><td>1</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>28</td><td>Output > |28| Ai=|0|, Bi=|0|, P1=|1|, P2=|0|, P3=|0|, P4=|1|, P5=|1|, Co=|1|,</td><td>1</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>29</td><td>Output > |29| Ai=|0|, Bi=|0|, P1=|0|, P2=|0|, P3=|0|, P4=|1|, P5=|1|, Co=|1|,</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>30</td><td>Output > |30| Ai=|0|, Bi=|0|, P1=|0|, P2=|0|, P3=|1|, P4=|1|, P5=|1|, Co=|1|,</td><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>...</td><td>...</td><td>..</td><td>...</td><td>...</td><td>...</td></tr>
</table>

1.1) ความสัมพันธ์เชิงล็อคประเภทเซมิล็อค ระหว่างสัญญาณ P1 และ P4

จากคุณสมบัติความสัมพันธ์เชิงล็อค จากตารางคู่ความสัมพันธ์ของสัญญาณ P1 และ P4 มีลักษณะแบบเซมิล็อค P10 > P40 > P11 โดนแทนค่าลบด้วย 0 และแทนค่าบวกด้วย 1 เมื่ออ่านจากทริกเกอร์แทนค่าด้วยคู่สี (0, 1) การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ P1 จะต้องเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ P4 เสมอ ในทำนองเดียวกัน การเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ P1 จะต้องเกิดหลังการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ P4 เสมอเช่นกัน

1.2) ความสัมพันธ์เชิงล็อคประเภทฟูลล็อค ระหว่างสัญญาณ P1 และ P4

จากคุณสมบัติความสัมพันธ์เชิงล็อค จากตารางคู่ความสัมพันธ์ของสัญญาณ P1 และ P4 มีลักษณะแบบฟูลล็อค P10 > P40 > P11 > P41 โดนแทนค่าลบด้วย 0 และแทนค่าบวกด้วย 1 เมื่ออ่านจากทริกเกอร์แทนค่าด้วยคู่สี (0, 1) การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ P1 จะต้องเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ P4 เสมอ ในทำนองเดียวกัน การเปลี่ยนแปลง ทริกเกอร์สีเทา(+) ของ P1 จะต้องเกิดหลังการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ P4 เสมอ และเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ P4 เสมอเช่นกัน

2. การตรวจสอบคู่ความสัมพันธ์เชิงล็อคของคู่สัญญาณ P2 และ P4

จากขั้นตอนการทวนสอบแอลเอสเอสกับแอลอาร์เอสของคู่ความสัมพันธ์เชิงล็อค P2 และ P4 ในตารางที่ 4.14 ตารางแสดงลำดับการเปลี่ยนแปลงสัญญาณจากการจำลองการทำงาน

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ตารางที่ 4.14 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ P2 และ P4 ขั้นที่ 1-30

ขั้นที่	ผลจากการจำลองการทำงาน	P1	P2	P3	P4
1	Output > |1| Ai=|1|, Bi=|0|, P1=|0|, P2=|0|, P3=|1|, P4=|0|, P5=|1|, Co=|0|,	0	0	1	0
2	Output > |2| Ai=|1|, Bi=|0|, P1=|1|, P2=|0|, P3=|1|, P4=|0|, P5=|1|, Co=|0|,	1	0	1	0
3	Output > |3| Ai=|1|, Bi=|1|, P1=|1|, P2=|0|, P3=|1|, P4=|0|, P5=|1|, Co=|0|,	1	0	1	0
4	Output > |4| Ai=|1|, Bi=|1|, P1=|1|, P2=|1|, P3=|1|, P4=|0|, P5=|1|, Co=|0|,	1	1	1	0
5	Output > |5| Ai=|1|, Bi=|1|, P1=|1|, P2=|1|, P3=|1|, P4=|0|, P5=|0|, Co=|0|,	1	1	1	0

ตารางที่ 4.14 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ P2 และ P4 ขั้นที่ 1-30 (ต่อ)

ขั้นที่	ผลจากการจำลองการทำงาน	P1	P2	P3	P4
6	Output > |6| Ai=|1|, Bi=|1|, P1=|1|, P2=|1|, P3=|1|, P4=|1|, P5=|0|, Co=|0|,	1	1	1	1
7	Output > |7| Ai=|1|, Bi=|1|, P1=|1|, P2=|1|, P3=|0|, P4=|1|, P5=|0|, Co=|0|,	1	1	0	1
8	Output > |8| Ai=|1|, Bi=|1|, P1=|1|, P2=|1|, P3=|0|, P4=|1|, P5=|1|, Co=|0|,	1	1	0	1
9	Output > |9| Ai=|1|, Bi=|1|, P1=|1|, P2=|1|, P3=|0|, P4=|1|, P5=|1|, Co=|1|,	1	1	0	1
10	Output > |10| Ai=|1|, Bi=|0|, P1=|1|, P2=|1|, P3=|0|, P4=|1|, P5=|1|, Co=|1|,	1	1	0	1
11	Output > |11| Ai=|0|, Bi=|0|, P1=|1|, P2=|1|, P3=|0|, P4=|1|, P5=|1|, Co=|1|,	1	1	0	1
12	Output > |12| Ai=|0|, Bi=|0|, P1=|1|, P2=|0|, P3=|0|, P4=|1|, P5=|1|, Co=|1|,	1	0	0	1
13	Output > |13| Ai=|0|, Bi=|0|, P1=|0|, P2=|0|, P3=|0|, P4=|1|, P5=|1|, Co=|1|,	0	0	0	1
14	Output > |14| Ai=|0|, Bi=|0|, P1=|0|, P2=|0|, P3=|1|, P4=|1|, P5=|1|, Co=|1|,	0	0	1	1
15	Output > |15| Ai=|0|, Bi=|0|, P1=|0|, P2=|0|, P3=|1|, P4=|0|, P5=|1|, Co=|1|,	0	0	1	0
16	Output > |16| Ai=|0|, Bi=|0|, P1=|0|, P2=|0|, P3=|1|, P4=|0|, P5=|1|, Co=|0|,	0	0	1	0
17	Output > |17| Ai=|1|, Bi=|0|, P1=|0|, P2=|0|, P3=|1|, P4=|0|, P5=|1|, Co=|0|,	0	0	1	0
18	Output > |18| Ai=|1|, Bi=|1|, P1=|0|, P2=|0|, P3=|1|, P4=|0|, P5=|1|, Co=|0|,	0	0	1	0
19	Output > |19| Ai=|1|, Bi=|1|, P1=|1|, P2=|0|, P3=|1|, P4=|0|, P5=|1|, Co=|0|,	1	0	1	0
20	Output > |20| Ai=|1|, Bi=|1|, P1=|1|, P2=|1|, P3=|1|, P4=|0|, P5=|1|, Co=|0|,	1	1	1	0
21	Output > |21| Ai=|1|, Bi=|1|, P1=|1|, P2=|1|, P3=|1|, P4=|0|, P5=|0|, Co=|0|,	1	1	1	0

ตารางที่ 4.14 ตารางแสดงลำดับการเปลี่ยนแปลงของคู่สัญญาณ P2 และ P4 ขั้นที่ 1-30 (ต่อ)

<table>
<tr><th>ขั้นที่</th><th>ผลจากการจำลองการทำงาน</th><th>P1</th><th>P2</th><th>P3</th><th>P4</th></tr>
<tr><td>22</td><td>Output > |22| Ai=|1|, Bi=|1|, P1=|1|, P2=|1|, P3=|1|, P4=|1|, P5=|0|, Co=|0|,</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>23</td><td>Output > |23| Ai=|1|, Bi=|1|, P1=|1|, P2=|1|, P3=|0|, P4=|1|, P5=|0|, Co=|0|,</td><td>1</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>24</td><td>Output > |24| Ai=|1|, Bi=|1|, P1=|1|, P2=|1|, P3=|0|, P4=|1|, P5=|1|, Co=|0|,</td><td>1</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>25</td><td>Output > |25| Ai=|1|, Bi=|1|, P1=|1|, P2=|1|, P3=|0|, P4=|1|, P5=|1|, Co=|1|,</td><td>1</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>26</td><td>Output > |26| Ai=|1|, Bi=|0|, P1=|1|, P2=|1|, P3=|0|, P4=|1|, P5=|1|, Co=|1|,</td><td>1</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>27</td><td>Output > |27| Ai=|0|, Bi=|0|, P1=|1|, P2=|1|, P3=|0|, P4=|1|, P5=|1|, Co=|1|,</td><td>1</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>28</td><td>Output > |28| Ai=|0|, Bi=|0|, P1=|1|, P2=|0|, P3=|0|, P4=|1|, P5=|1|, Co=|1|,</td><td>1</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>29</td><td>Output > |29| Ai=|0|, Bi=|0|, P1=|0|, P2=|0|, P3=|0|, P4=|1|, P5=|1|, Co=|1|,</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>30</td><td>Output > |30| Ai=|0|, Bi=|0|, P1=|0|, P2=|0|, P3=|1|, P4=|1|, P5=|1|, Co=|1|,</td><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>...</td><td>...</td><td>..</td><td>...</td><td>...</td><td>...</td></tr>
</table>

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

2.1) ความสัมพันธ์เชิงล็อคประเภทเซมิล็อค ระหว่างสัญญาณ P2 และ P4

จากคุณสมบัติความสัมพันธ์เชิงล็อค จากตารางคู่ความสัมพันธ์ของสัญญาณ P2 และ P4 มีลักษณะแบบเซมิล็อค P20 > P40 > P21 โดนแทนค่าลบด้วย 0 และแทนค่าบวกด้วย 1 เมื่ออ่านจากทริกเกอร์แทนค่าด้วยคู่สี (0, 1) การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ P2 จะต้องเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ P4 เสมอ ในทำนองเดียวกัน การเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ P2 จะต้องเกิดหลังการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ P4 เสมอเช่นกัน

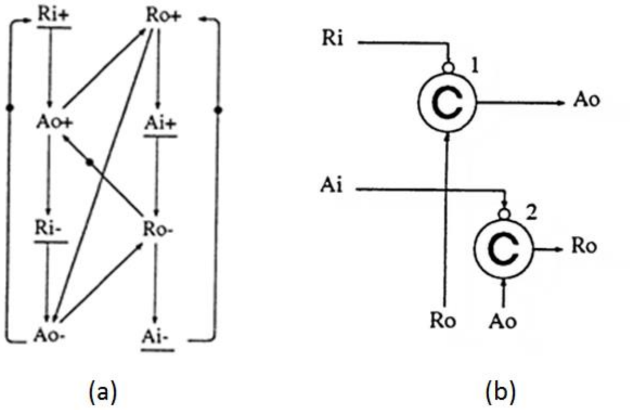
2.2) ความสัมพันธ์เชิงล็อคประเภทฟูลล็อค ระหว่างสัญญาณ P2 และ P4

จากคุณสมบัติความสัมพันธ์เชิงล็อค จากตารางคู่ความสัมพันธ์ของสัญญาณ P2 และ P4 มีลักษณะแบบฟูลล็อค P20 > P40 > P21 > P41 โดนแทนค่าลบด้วย 0 และแทนค่าบวกด้วย 1 เมื่อ

อ่านจากทริกเกอร์แทนค่าด้วยคู่สี (0, 1) การเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ P2 จะต้องเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ P4 เสมอ ในทำนองเดียวกัน การเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ P2 จะต้องเกิดหลังการเปลี่ยนแปลงทริกเกอร์สีขาว(-) ของ P4 เสมอ และเกิดก่อนการเปลี่ยนแปลงทริกเกอร์สีเทา(+) ของ P4 เสมอเช่นกัน

4.4 การทวนสอบตัวอย่างวงจรอสมวาร full ด้วยวงจรที่สังเคราะห์ไม่ถูกต้อง

ตัวอย่างวงจรอสมวาร full ประกอบด้วยเอสทีจีรูปที่ 4.4(a) และวงจรที่ได้จากการสังเคราะห์ที่ไม่ถูกต้องดังรูปที่ 4.4(b)

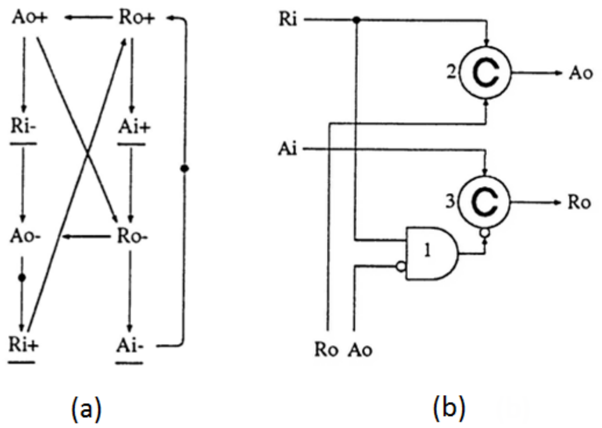


รูปที่ 4.4 เอสทีจี (a) [5] และวงจรที่ได้จากการสังเคราะห์ไม่ถูกต้อง (b) ของวงจรอสมวาร full

เมื่อจำลองการทำงานจากรหัสภาษาโพรเมล่าที่ได้ทำการแปลงไว้ด้วยสปินปรากฏว่าวงจรหยุดการทำงานในขั้นที่ 7 ทำให้ตรวจพบว่าวงจรที่สังเคราะห์มีข้อผิดพลาดเกิดขึ้น

4.5 การทวนสอบตัวอย่างวงจรอสมวาร half ด้วยวงจรที่สังเคราะห์ไม่ถูกต้อง

ตัวอย่างวงจรอสมวาร half ประกอบด้วยเอสทีจีรูปที่ 4.5(a) และวงจรที่ได้จากการสังเคราะห์ที่ไม่ถูกต้องดังรูปที่ 4.5(b)



รูปที่ 4.5 เอสทีจี (a) [5] และวงจรที่ได้จากการสังเคราะห์ไม่ถูกต้อง (b) ของวงจรอสมวาร half

เมื่อจำลองการทำงานจากรหัสภาษาโพรเมล่าที่ได้ทำการแปลงไว้ด้วยสปินปรากฏว่าวงจรหยุดการทำงานในขั้นที่ 1 ทำให้ตรวจพบว่าวงจรที่สังเคราะห์มีข้อผิดพลาดเกิดขึ้น

4.6 สรุปผลจากการทวนสอบวงจรอสมวารตัวอย่าง

การทวนสอบจากตัวอย่างวงจรอสมวารที่นำเสนอประกอบด้วยวงจรสังเคราะห์ตัวอย่างทั้งหมด 6 วงจร จากตารางที่ 4.16

ตารางที่ 4.15 ตารางสรุปผลการทวนสอบด้วยวิธีการที่นำเสนอกับตัวอย่างวงจรอสมวาร

ชื่อวงจร	เอสทีจี		วงจรที่ได้จากการสังเคราะห์		จำนวนเอสทีเอส	แอลเอสเอส (รอบการทำงานที่น้อยที่สุด)	แอลอาร์เอส	
	จำนวนตัวแปรอินพุต	จำนวนตัวแปรที่ไม่ใช่อินพุต	จำนวนตัวแปรอินพุต	จำนวนตัวแปรที่ไม่ใช่อินพุต			เซมิล็อค	ฟูลล็อค
full	2	2	2	2	5	1,760	3	3
half	2	2	2	2	12	580	3	3
C-element	2	1	2	6	4	54	2	2
C-element (non-input)	2	6	2	6	9	1,505	-	-
full (วงจรที่สังเคราะห์ไม่ถูกต้อง)	2	2	2	2	5	หยุดการทำงานที่ขั้นตอน 7	3	3
half (วงจรที่สังเคราะห์ไม่ถูกต้อง)	2	2	2	2	12	หยุดการทำงานที่ขั้นตอน 1	3	3

สรุปได้ว่าวงจร full, half, C-element และ C-element แบบสนใจค่าที่ไม่ใช่อินพุต เมื่อทวนสอบด้วยขั้นตอนที่นำเสนอสามารถทวนสอบพบทุกกรณีของ เอสทีเอสภายใต้การวนรอบการทำงานที่น้อยที่สุด และเมื่อทวนสอบกับแอลอาร์เอสของแต่ละคู่สัญญาณ ปรากฏว่ามีลำดับการเปลี่ยนแปลงสัญญาณตามความสัมพันธ์เชิงล็อคประเภทฟูลล็อค และเซมิล็อคได้ถูกต้องสองวงจรตัวอย่างถัดมาเป็นวงจรที่ได้จากการสังเคราะห์ที่ไม่ถูกต้อง คือวงจรอสมวาร full และวงจรอสมวาร half เมื่อจำลองการทำงานด้วยสปินพบว่าสามารถจำลองการทำงานไปได้ไม่นาน สัญญาณจะไม่เกิดการเปลี่ยนแปลงอีก ซึ่งแตกต่างจากวงจรที่สังเคราะห์ได้ถูกต้องจะสามารถทำงานต่อไปได้ปราศจากการติดตาย

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทที่ 5
สรุปผลงานวิจัยและข้อเสนอแนะ

5.1 สรุปผลงานวิจัย

การทวนสอบวงจรอสมวารด้วยการจำลองลำดับสัญญาณในเอสทีจีเป็นงานวิจัยที่นำเครื่องมือสปินมาประยุกต์ใช้ในการจำลองการทำงานของวงจรอสมวารจากรหัสภาษาโพรเมล่า ที่ได้จากการแปลงพฤติกรรมในเอสทีจีและวงจรอสมวารที่ได้จากการสังเคราะห์ เอสเอสที่ได้จากการจำลองนั้นจะนำไปทวนสอบเอสทีเอส เพื่อตรวจสอบในเอสเอสว่าสามารถพบทุกรูปแบบของ เอสทีเอสหรือไม่ เมื่อสามารถตรวจสอบพบทุกเอสทีเอสแล้ว จะแสดงถึง คุณสมบัติไลฟ์เนสและผลลัพธ์ที่ได้คือแอลเอสเอส ซึ่งก็คือรอบวนซ้ำการจำลองการทำงานที่น้อยที่สุดที่แสดงถึงคุณสมบัติไลฟ์เนส จากนั้นทำการทวนสอบแอลเอสเอสกับแอลอาร์เอสด้วยลำดับการเปลี่ยนแปลงสัญญาณ ของทุกคู่สัญญาณที่มีความสัมพันธ์เชิงล็อคร่วมกันในวงจร และลำดับการเปลี่ยนแปลงสัญญาณในแอลเอสเอสจะต้องถูกต้องตามแอลอาร์เอส เป็นการยืนยันความถูกต้องอีกระดับหนึ่งด้วยคุณสมบัติความทนทานเพื่อการันตีว่าจากสองคุณสมบัติที่กล่าว วงจรที่ได้จากการสังเคราะห์นั้นๆ จะมีพฤติกรรมที่ถูกต้องตามที่ได้ออกแบบไว้ในเอสทีจี

จากขั้นตอนวิธีการทวนสอบวงจรอสมวารด้วยการจำลองลำดับสัญญาณในเอสทีจี ที่ได้นำเสนอในงานวิจัยนี้ เมื่อทดสอบกับชุดตัวอย่างวงจรอสมวารที่ได้จากการสังเคราะห์ ผลลัพธ์ที่ได้มีความถูกต้องและสอดคล้องกัน ระหว่างพฤติกรรมที่คาดหวังในเอสทีจีและผลจากการจำลองการทำงาน ตามคุณสมบัติไลฟ์เนสและคุณสมบัติความทนทาน

5.2 ข้อจำกัด

1. วิธีการที่นำเสนอนี้ไม่สามารถใช้กับวงจรที่ไม่แสดงสัญญาณเชื่อมต่อภายในวงจร
2. วิธีการที่นำเสนอนี้ไม่สามารถใช้กับวงจรแบบวัฏจักรเชิงซ้อน (Multi cycle) ได้
3. ขณะวนซ้ำการจำลองการทำงานของวงจรกรณีไม่สามารถตรวจพบครบทุกเอสทีเอส ทำให้ต้องเพิ่มรอบการทำงานต่อไปเรื่อยๆ จนกว่าจะตรวจพบ

5.3 ข้อเสนอแนะ

ขั้นตอนวิธีการทวนสอบที่ได้นำเสนอในงานวิจัยนี้ เป็นการประยุกต์นำสปินมาในช่วยในการจำลองการทำงานของวงจรอสมวาร และนำผลลัพธ์ที่ได้จากการจอลองนำมาทำการทวนสอบต่อไป โดยในงานวิจัยนี้อาศัยเพียงคุณสมบัติการจำลองการทำงานของเครื่องมือสปินเท่านั้น ควรมีการพัฒนาปรับปรุงขั้นตอนวิธีการ เพื่อนำความสามารถในส่วนการทวนสอบของสปินมาใช้เพื่อช่วยลดขั้นตอนในหลายๆ ขั้นตอนที่ยังใช้วิธีการไม่อัตโนมัติอยู่

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

รายการอ้างอิง

1. Chu, T., *SYNTHESIS OF SELF-TIMED VLSI CIRCUITS FROM GRAPH-THEORETIC SPECIFICATIONS*. 1987, Massachusetts Institute of Technology.
2. Hauck, S., *Asynchronous Design Methodologies: An Overview.* Proceedings of the IEEE, 1995. **83**(1): p. 69-93.
3. Baier, C. and J.-P. Katoen, *Principles of model checking*. Vol. 26202649. 2008: MIT press Cambridge.
4. Jiang, K., *Model Checking C Programs by Translating C to Promela* in *Department of Information Technology*. 2009, Institutionen för informationsteknologi.
5. Sentovich, E.M., et al., *SIS: A System for Sequential Circuit Synthesis*. 1992, EECS Department, University of California, Berkeley.
6. Park, S.-B., *Synthesis of Asynchronous VLSI Circuits from Signal Transition Graph Specifications*, in *Department of Enigineering-Computer Science*. 1996, Tokyo Institute of Technology.
7. Yoeli, M., *Formal Verification of Hardware Design*. 1990: IEEE Computer Society Press. 340.
8. Holzmann, G.J., *Tutorial: Proving Properties of Concurrent System with SPIN*, in *Proceedings of the 6th International Conference on Concurrency Theory*. 1995, Springer-Verlag. p. 453-455.
9. Haghighat, K. *Multithreading - An Operating System Analysis*. 2008; Available from: http://kevinhaghighat.com/Papers/Multithreading.pdf.
10. Ben-Ari, M., *A primer on model checking.* ACM Inroads, 2010. **1**(1): p. 40-47.
11. Holzmann, G.J., *The Model Checker SPIN.* IEEE Transactions on Software Engineering, 1997. **23**(5): p. 279-295.
12. *Modeling Language*. 2007; Available from: http://spinroot.com/spin/Man/Manual.html.
13. Rahardjo, B. and R.D. McLeod. *Verification of speed-independent asynchronous circuits with protocol validation tools*. in *Communications,*

Computers, and Signal Processing, 1995. Proceedings., IEEE Pacific Rim Conference on. 1995.

14. Thongtak, A., *A Study on testing methodologies of asynchronous logic circuits*, in *Department of Electrical and Electronics Engineering*. Jan.1996, Tokyo Institute of Technology.

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ภาคผนวก

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ภาคผนวก ก

1.รหัสโพรเมล่าจากการแปลงพฤติกรรมในเอสทีจีและวงจรอสมวารที่ได้จากการสังเคราะห์

```
bit Ri=0, Ro=0, Ai=1, Ao=1;
int i=0;

active proctype Celement1(){
do
:: (Ri == 0 && Ro == 1) -> atomic {
        if
        :: (Ao == 1) -> { Ao = 0;  i = i+1;
        printf("\n Output >  |%d| Ri=|%d|,
Ro=|%d|, Ai=|%d|, Ao=|%d| \n", i, Ri,Ro,Ai,Ao); }
        :: skip
        fi;
        }
:: (Ri == 1 && Ro == 0) -> atomic {
        if
        :: (Ao == 0) -> { Ao = 1;  i = i+1;
        printf("\n Output >  |%d| Ri=|%d|,
Ro=|%d|, Ai=|%d|, Ao=|%d| \n", i, Ri,Ro,Ai,Ao); }
        :: skip
        fi;
        }
od;
}

active proctype Celement2(){
do
:: (Ai == 1 && Ao == 0) -> atomic {
        if
        :: (Ro == 1) -> { Ro = 0;  i = i+1;
        printf("\n Output >  |%d| Ri=|%d|,
Ro=|%d|, Ai=|%d|, Ao=|%d| \n", i, Ri,Ro,Ai,Ao); }
        :: skip
        fi;
        }
:: (Ai == 0 && Ao == 1) -> atomic {
        if
        :: (Ro == 0) -> { Ro = 1;  i = i+1;
        printf("\n Output >  |%d| Ri=|%d|,
Ro=|%d|, Ai=|%d|, Ao=|%d| \n", i, Ri,Ro,Ai,Ao); }
        :: skip
        fi;
        }
od;
}
```

```
active proctype MonitorA(){
do
:: (Ao == 1) ->  atomic {
        if
        :: (Ri == 1) -> { Ri = 0 ; i = i + 1;
        printf("\n Output >  |%d| Ri=|%d|,
Ro=|%d|, Ai=|%d|, Ao=|%d| \n", i, Ri,Ro,Ai,Ao); }
        :: skip
        fi;
        }
:: (Ao == 0) ->  atomic {
        if
        :: (Ri == 0) -> { Ri = 1 ; i = i + 1;
        printf("\n Output >  |%d| Ri=|%d|,
Ro=|%d|, Ai=|%d|, Ao=|%d| \n", i, Ri,Ro,Ai,Ao); }
        :: skip
        fi;
        }
od;
}

active proctype MonitorR(){
do
:: (Ro == 1) ->  atomic {
        if
        :: (Ai == 0) -> { Ai = 1 ; i = i + 1;
        printf("\n Output >  |%d| Ri=|%d|,
Ro=|%d|, Ai=|%d|, Ao=|%d| \n", i, Ri,Ro,Ai,Ao); }
        :: skip
        fi;
        }
:: (Ro == 0) ->  atomic {
        if
        :: (Ai == 1) -> { Ai = 0 ; i = i + 1;
        printf("\n Output >  |%d| Ri=|%d|,
Ro=|%d|, Ai=|%d|, Ao=|%d| \n", i, Ri,Ro,Ai,Ao); }
        :: skip
        fi;
        }
od;
}
```

รูปที่ ก-1 รหัสภาษาโพรเมล่าวงจร full จากข้อที่ 4.1.1

```
bit Ao=1, Ri=1, Ro=0, Ai=0, OA=0;
int i=0;

active proctype AndGate1(){
do
:: (Ri == 0 && Ao == 1) || (Ri == Ao) -> atomic {
	if
	:: (OA == 1) -> { OA = 0; i = i + 1;
	printf("\n Output >  |%d| Ri=|%d|,
Ro=|%d|, Ai=|%d|, Ao=|%d| \n", i, Ri,Ro,Ai,Ao); }
	:: skip
	fi;
	}
:: (Ri == 1 && Ao == 0) -> atomic {
	if
	:: (OA == 0) -> { OA = 1; i = i + 1;
	printf("\n Output >  |%d| Ri=|%d|,
Ro=|%d|, Ai=|%d|, Ao=|%d| \n", i, Ri,Ro,Ai,Ao); }
	:: skip
	fi;
	}
od;
}

active proctype Celement2(){
do
:: (Ri == 0 && Ro == 0) -> atomic {
	if
	:: (Ao == 1) -> { Ao = 0;  i = i+1;
	printf("\n Output >  |%d| Ri=|%d|,
Ro=|%d|, Ai=|%d|, Ao=|%d| \n", i, Ri,Ro,Ai,Ao); }
	:: skip
	fi;
	}
:: (Ri == 1 && Ro == 1) -> atomic {
	if
	:: (Ao == 0) -> { Ao = 1;  i = i+1;
	printf("\n Output >  |%d| Ri=|%d|,
Ro=|%d|, Ai=|%d|, Ao=|%d| \n", i, Ri,Ro,Ai,Ao); }
	:: skip
	fi;
	}
od;
}

active proctype Celement3(){
do
:: (Ai == 0 && OA == 1) -> atomic {
	if
	:: (Ro == 0) -> { Ro = 1;  i = i+1;
	printf("\n Output >  |%d| Ri=|%d|,
Ro=|%d|, Ai=|%d|, Ao=|%d| \n", i, Ri,Ro,Ai,Ao); }
	:: skip
	fi;
	}
:: (Ai == 1 && OA == 0) -> atomic {
	if
	:: (Ro == 1) -> { Ro = 0;  i = i+1;
	printf("\n Output >  |%d| Ri=|%d|,
Ro=|%d|, Ai=|%d|, Ao=|%d| \n", i, Ri,Ro,Ai,Ao); }
	:: skip
	fi;
	}
od;
}

active proctype MonitorA(){
do
:: (Ao == 1) ->  atomic {
	if
	:: (Ri == 1) -> { Ri = 0 ; i = i + 1;
	printf("\n Output >  |%d| Ri=|%d|,
Ro=|%d|, Ai=|%d|, Ao=|%d| \n", i, Ri,Ro,Ai,Ao); }
	:: skip
	fi;
	}
:: (Ao == 0) ->  atomic {
	if
	:: (Ri == 0) -> { Ri = 1 ; i = i + 1;
	printf("\n Output >  |%d| Ri=|%d|,
Ro=|%d|, Ai=|%d|, Ao=|%d| \n", i, Ri,Ro,Ai,Ao); }
	:: skip
	fi;
	}
od;
}


active proctype MonitorR(){
do
:: (Ro == 1) ->  atomic {
	if
	:: (Ai == 0) -> { Ai = 1 ; i = i + 1;
	printf("\n Output >  |%d| Ri=|%d|,
Ro=|%d|, Ai=|%d|, Ao=|%d| \n", i, Ri,Ro,Ai,Ao); }
	:: skip
	fi;
	}
:: (Ro == 0) ->  atomic {
	if
	:: (Ai == 1) -> { Ai = 0 ; i = i + 1;
	printf("\n Output >  |%d| Ri=|%d|,
Ro=|%d|, Ai=|%d|, Ao=|%d| \n", i, Ri,Ro,Ai,Ao); }
	:: skip
	fi;
	}
od;
}
```

รูปที่ ก-2 รหัสภาษาโพรเมล่าวงจร half จากข้อที่ 4.2.1

```
bit Ai=0, Bi=0, P1=0, P2=0, P3=1, P4=0, P5=1,
Co=0;
int i = 0;

active proctype OrGate1(){
do
:: (Ai==0 && P2==0) ->  atomic {
        if
        :: (P1 == 1) -> { P1 = 0 ;  i = i + 1;
        printf("\n Output >  |%d| Ai=|%d|,
Bi=|%d|, P1=|%d|, P2=|%d|, P3=|%d|, P4=|%d|,
P5=|%d|, Co=|%d|, \n",
i,Ai,Bi,P1,P2,P3,P4,P5,Co);}
        :: skip
        fi;
        }
:: (Ai==1 || P2==1) -> atomic {
        if
        :: (P1 == 0) -> { P1 = 1 ;  i = i + 1;
        printf("\n Output >  |%d| Ai=|%d|,
Bi=|%d|, P1=|%d|, P2=|%d|, P3=|%d|, P4=|%d|,
P5=|%d|, Co=|%d|, \n",
i,Ai,Bi,P1,P2,P3,P4,P5,Co);}
        :: skip
        fi;
        }
od;}

active proctype AndGate1(){
do
:: (Bi == 0 || P1 == 0) -> atomic {
        if
        :: (P2 == 1) -> { P2 = 0; i = i + 1;
        printf("\n Output >  |%d| Ai=|%d|,
Bi=|%d|, P1=|%d|, P2=|%d|, P3=|%d|, P4=|%d|,
P5=|%d|, Co=|%d|, \n",
i,Ai,Bi,P1,P2,P3,P4,P5,Co);}
        :: skip
        fi;
        }
:: (Bi == 1 && P1 == 1) -> atomic {
        if
        :: (P2 == 0) -> { P2 = 1; i = i + 1;
        printf("\n Output >  |%d| Ai=|%d|,
Bi=|%d|, P1=|%d|, P2=|%d|, P3=|%d|, P4=|%d|,
P5=|%d|, Co=|%d|, \n",
i,Ai,Bi,P1,P2,P3,P4,P5,Co);}
        :: skip
        fi;
        }
od;
}

active proctype AndGate2(){
do
:: (P1 == 0 || P4 == 0) -> atomic {
        if
        :: (P3 == 0) -> { P3 = 1; i = i + 1;
        printf("\n Output >  |%d| Ai=|%d|,
Bi=|%d|, P1=|%d|, P2=|%d|, P3=|%d|, P4=|%d|,
P5=|%d|, Co=|%d|, \n",
i,Ai,Bi,P1,P2,P3,P4,P5,Co);}
```

```
active proctype AndGate4(){
do
:: (P2 == 0 || P3 == 0) -> atomic {
        if
        :: (P5 == 0) -> { P5 = 1; i = i + 1;
        printf("\n Output >  |%d| Ai=|%d|,
Bi=|%d|, P1=|%d|, P2=|%d|, P3=|%d|, P4=|%d|,
P5=|%d|, Co=|%d|, \n",
i,Ai,Bi,P1,P2,P3,P4,P5,Co);}
        :: skip
        fi;
        }
:: (P2 == 1 && P3 == 1) -> atomic {
        if
        :: (P5 == 1) -> { P5 = 0; i = i + 1;
        printf("\n Output >  |%d| Ai=|%d|,
Bi=|%d|, P1=|%d|, P2=|%d|, P3=|%d|, P4=|%d|,
P5=|%d|, Co=|%d|, \n",
i,Ai,Bi,P1,P2,P3,P4,P5,Co);}
        :: skip
        fi;
        }
od;
}

active proctype AndGate5(){
do
:: (P4 == 0 || P5 == 0) -> atomic {
        if
        :: (Co == 1) -> { Co = 0; i = i + 1;
        printf("\n Output >  |%d| Ai=|%d|,
Bi=|%d|, P1=|%d|, P2=|%d|, P3=|%d|, P4=|%d|,
P5=|%d|, Co=|%d|, \n",
i,Ai,Bi,P1,P2,P3,P4,P5,Co);}
        :: skip
        fi;
        }
:: (P4 == 1 && P5 == 1) -> atomic {
        if
        :: (Co == 0) -> { Co = 1; i = i + 1;
        printf("\n Output >  |%d| Ai=|%d|,
Bi=|%d|, P1=|%d|, P2=|%d|, P3=|%d|, P4=|%d|,
P5=|%d|, Co=|%d|, \n",
i,Ai,Bi,P1,P2,P3,P4,P5,Co);}
        :: skip
        fi;
        }
od;
}

active proctype MonitorC(){
do
:: (Co == 1) ->  atomic {
        if
        :: (Bi == 1) -> {
                Bi = 0 ; i = i + 1;
        printf("\n Output >  |%d| Ai=|%d|,
Bi=|%d|, P1=|%d|, P2=|%d|, P3=|%d|, P4=|%d|,
P5=|%d|, Co=|%d|, \n",
i,Ai,Bi,P1,P2,P3,P4,P5,Co);}
        :: skip
        fi;
```

รูปที่ ก-3 รหัสภาษาโพรเมล่าวงจร C-element แบบสนใจค่าที่ไม่ใช่อินพุต จากข้อที่ 4.3.1

```
            :: skip
            fi;
            }
:: (P1 == 1 && P4 == 1) -> atomic {
            if
            :: (P3 == 1) -> { P3 = 0; i = i + 1;
            printf("\n Output >  |%d| Ai=|%d|,
Bi=|%d|, P1=|%d|, P2=|%d|, P3=|%d|, P4=|%d|,
P5=|%d|, Co=|%d|, \n",
i,Ai,Bi,P1,P2,P3,P4,P5,Co);}
            :: skip
            fi;
            }
od;
}

active proctype AndGate3(){
do
:: (P3 == 0 || P5 == 0) -> atomic {
            if
            :: (P4 == 0) -> { P4 = 1; i = i + 1;
            printf("\n Output >  |%d| Ai=|%d|,
Bi=|%d|, P1=|%d|, P2=|%d|, P3=|%d|, P4=|%d|,
P5=|%d|, Co=|%d|, \n",
i,Ai,Bi,P1,P2,P3,P4,P5,Co);}
            :: skip
            fi;
            }
:: (P3 == 1 && P5 == 1) -> atomic {
            if
            :: (P4 == 1) -> { P4 = 0; i = i + 1;
            printf("\n Output >  |%d| Ai=|%d|,
Bi=|%d|, P1=|%d|, P2=|%d|, P3=|%d|, P4=|%d|,
P5=|%d|, Co=|%d|, \n",
i,Ai,Bi,P1,P2,P3,P4,P5,Co);}
            :: skip
            fi;
            }
od;
}
```

```
            if
            :: (Ai == 1) -> {
                        Ai = 0 ; i = i + 1;
            printf("\n Output >  |%d| Ai=|%d|,
Bi=|%d|, P1=|%d|, P2=|%d|, P3=|%d|, P4=|%d|,
P5=|%d|, Co=|%d|, \n",
i,Ai,Bi,P1,P2,P3,P4,P5,Co);}
            :: skip
            fi;
            }
:: (Co == 0) ->  atomic {
            if
            :: (Ai == 0) -> {
                        Ai = 1 ; i = i + 1;
            printf("\n Output >  |%d| Ai=|%d|,
Bi=|%d|, P1=|%d|, P2=|%d|, P3=|%d|, P4=|%d|,
P5=|%d|, Co=|%d|, \n",
i,Ai,Bi,P1,P2,P3,P4,P5,Co);}
            :: skip
fi;
            if
            :: (Bi == 0) -> {
                        Bi = 1 ; i = i + 1;
            printf("\n Output >  |%d| Ai=|%d|,
Bi=|%d|, P1=|%d|, P2=|%d|, P3=|%d|, P4=|%d|,
P5=|%d|, Co=|%d|, \n",
i,Ai,Bi,P1,P2,P3,P4,P5,Co);}
            :: skip
            fi;
            }
od;
}
```

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

รูปที่ ก-3 รหัสภาษาโพรเมล่าวงจร C-element แบบสนใจค่าที่ไม่ใช่อินพุต จากข้อที่ 4.3.1 (ต่อ)

ภาคผนวก ข

2. ตารางการทวนสอบเอสทีเอสกับเอสเอส

ตารางที่ ข-1 ผลการทวนสอบเอสทีเอสกับเอสเอสของตัวอย่างวงจรอสมวาร full จากข้อ 4.1.3

ขั้นแรก	ขั้นสุดท้าย	ลำดับการเปลี่ยนแปลงสัญญาณรอบปัจจุบันและก่อนหน้า	เอสทีเอสที่พบ
1	8	Ri1 > Ao1 > Ro1 > Ai1 > Ri0 > Ao0 > Ro0 > Ai0 >	STS 3
1	16	Ri1 > Ao1 > Ro1 > Ai1 > Ri0 > Ao0 > Ro0 > Ai0 > Ri1 > Ao1 > Ro1 > Ai1 > Ri0 > Ao0 > Ro0 > Ri1 >	
9	24	Ri1 > Ao1 > Ro1 > Ai1 > Ri0 > Ao0 > Ro0 > Ri1 > Ao1 > Ri0 > Ai0 > Ro1 > Ao0 > Ai1 > Ri1 > Ro0 >	
17	32	Ao1 > Ri0 > Ai0 > Ro1 > Ao0 > Ai1 > Ri1 > Ro0 > Ao1 > Ai0 > Ri0 > Ro1 > Ao0 > Ai1 > Ri1 > Ro0 >	
25	40	Ao1 > Ai0 > Ri0 > Ro1 > Ao0 > Ai1 > Ri1 > Ro0 > Ao1 > Ai0 > Ro1 > Ai1 > Ri0 > Ao0 > Ro0 > Ai0 >	
33	48	Ao1 > Ai0 > Ro1 > Ai1 > Ri0 > Ao0 > Ro0 > Ai0 > Ri1 > Ao1 > Ro1 > Ai1 > Ri0 > Ao0 > Ro0 > Ri1 >	
41	56	Ri1 > Ao1 > Ro1 > Ai1 > Ri0 > Ao0 > Ro0 > Ri1 > Ai0 > Ao1 > Ro1 > Ai1 > Ri0 > Ao0 > Ri1 > Ro0 >	
49	64	Ai0 > Ao1 > Ro1 > Ai1 > Ri0 > Ao0 > Ri1 > Ro0 > Ai0 > Ao1 > Ri0 > Ro1 > Ao0 > Ai1 > Ri1 > Ro0 >	
57	72	Ai0 > Ao1 > Ri0 > Ro1 > Ao0 > Ai1 > Ri1 > Ro0 > Ai0 > Ao1 > Ri0 > Ro1 > Ai1 > Ao0 > Ro0 > Ri1 >	
65	80	Ai0 > Ao1 > Ri0 > Ro1 > Ai1 > Ao0 > Ro0 > Ri1 > Ao1 > Ai0 > Ri0 > Ro1 > Ai1 > Ao0 > Ro0 > Ri1 >	
73	88	Ao1 > Ai0 > Ri0 > Ro1 > Ai1 > Ao0 > Ro0 > Ri1 > Ao1 > Ai0 > Ri0 > Ro1 > Ao0 > Ri1 > Ai1 > Ro0 >	
81	96	Ao1 > Ai0 > Ri0 > Ro1 > Ao0 > Ri1 > Ai1 > Ro0 > Ao1 > Ri0 > Ai0 > Ro1 > Ao0 > Ai1 > Ro0 > Ri1 >	
89	104	Ao1 > Ri0 > Ai0 > Ro1 > Ao0 > Ai1 > Ro0 > Ri1 > Ao1 > Ai0 > Ri0 > Ro1 > Ao0 > Ai1 > Ro0 > Ri1 >	
97	112	Ao1 > Ai0 > Ri0 > Ro1 > Ao0 > Ai1 > Ro0 > Ri1 > Ai0 > Ao1 > Ro1 > Ai1 > Ri0 > Ao0 > Ro0 > Ri1 >	

ตารางที่ ข-1 ผลการทวนสอบเอสทีเอสกับเอสเอสของตัวอย่างวงจรอสมวาร full จากข้อ 4.1.3 (ต่อ)

ขั้นแรก	ขั้นสุดท้าย	ลำดับการเปลี่ยนแปลงสัญญาณรอบปัจจุบันและก่อนหน้า	เอสทีเอสที่พบ
105	120	Ai0 > Ao1 > Ro1 > Ai1 > Ri0 > Ao0 > Ro0 > Ri1 > Ai0 > Ao1 > Ro1 > Ri0 > Ao0 > Ai1 > Ri1 > Ro0 >	
113	128	Ai0 > Ao1 > Ro1 > Ri0 > Ao0 > Ai1 > Ri1 > Ro0 > Ao1 > Ai0 > Ri0 > Ro1 > Ai1 > Ao0 > Ri1 > Ro0 >	
121	136	Ao1 > Ai0 > Ri0 > Ro1 > Ai1 > Ao0 > Ri1 > Ro0 > Ai0 > Ao1 > Ro1 > Ri0 > Ao0 > Ri1 > Ai1 > Ro0 >	
129	144	Ai0 > Ao1 > Ro1 > Ri0 > Ao0 > Ri1 > Ai1 > Ro0 > Ao1 > Ai0 > Ri0 > Ro1 > Ao0 > Ri1 > Ai1 > Ro0 >	
137	152	Ao1 > Ai0 > Ri0 > Ro1 > Ao0 > Ri1 > Ai1 > Ro0 > Ao1 > Ai0 > Ri0 > Ro1 > Ai1 > Ao0 > Ri1 > Ro0 >	
145	160	Ao1 > Ai0 > Ri0 > Ro1 > Ai1 > Ao0 > Ri1 > Ro0 > Ao1 > Ai0 > Ro1 > Ai1 > Ri0 > Ao0 > Ri1 > Ro0 >	
593	608	Ai0 > Ao1 > Ri0 > Ro1 > Ao0 > Ai1 > Ro0 > Ai0 > Ri1 > Ao1 > Ro1 > Ri0 > Ai1 > Ao0 > Ro0 > Ai0 >	STS 5
777	792	Ri1 > Ao1 > Ro1 > Ri0 > Ai1 > Ao0 > Ro0 > Ai0 > Ri1 > Ao1 > Ro1 > Ri0 > Ao0 > Ai1 > Ro0 > Ai0 >	STS 4
817	832	Ri1 > Ao1 > Ro1 > Ai1 > Ri0 > Ao0 > Ro0 > Ai0 > Ri1 > Ao1 > Ri0 > Ro1 > Ao0 > Ai1 > Ro0 > Ai0 >	STS 1
1,745	**1,760**	**Ai0 > Ao1 > Ro1 > Ai1 > Ri0 > Ao0 > Ro0 > Ai0 > Ri1 > Ao1 > Ri0 > Ro1 > Ai1 > Ao0 > Ro0 > Ai0 >**	**STS 2**
1,753	1,768	Ri1 > Ao1 > Ri0 > Ro1 > Ai1 > Ao0 > Ro0 > Ai0 > Ri1 > Ao1 > Ro1 > Ri0 > Ai1 > Ao0 > Ro0 > Ri1 >	
…	…	…	…

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ตารางที่ ข-2 ผลการทวนสอบเอสทีเอสกับเอสเอสของตัวอย่างวงจรอสมวาร half จากข้อ 4.2.3

ขั้นแรก	ขั้นสุดท้าย	ลำดับการเปลี่ยนแปลงสัญญาณรอบปัจจุบันและก่อนหน้า	เอสทีเอสที่พบ
1	10	Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ai0 >	STS 2
4	20	Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ai0 > Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ri1 >	
14	30	Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ri1 > Ai0 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ri1 >	
24	40	Ai0 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ri1 > Ai0 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ri1 >	
34	50	Ai0 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ri1 > Ai0 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ai0 >	
44	60	Ai0 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ai0 > Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ai0 >	
54	70	Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ai0 > Ri1 > Ro1 > Ai1 > Ao1 > Ro0 > Ai0 > Ri0 > Ao0 >	STS 3
64	80	Ri1 > Ro1 > Ai1 > Ao1 > Ro0 > Ai0 > Ri0 > Ao0 > Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ri1 >	
74	90	Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ri1 > Ai0 > Ro1 > Ai1 > Ao1 > Ro0 > Ri0 > Ao0 > Ai0 >	
84	100	Ai0 > Ro1 > Ai1 > Ao1 > Ro0 > Ri0 > Ao0 > Ai0 > Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ai0 > Ao0 >	STS 11
94	110	Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ai0 > Ao0 > Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ai0 > Ao0 >	
104	120	Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ai0 > Ao0 > Ri1 > Ro1 > Ai1 > Ao1 > Ro0 > Ri0 > Ao0 > Ai0 >	STS 5
114	130	Ri1 > Ro1 > Ai1 > Ao1 > Ro0 > Ri0 > Ao0 > Ai0 > Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ai0 >	
124	140	Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ai0 > Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ai0 >	
134	150	Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ai0 > Ri1 > Ro1 > Ao1 > Ai1 > Ro0 > Ri0 > Ai0 > Ao0 >	STS 9

ตารางที่ ข-2 ผลการทวนสอบเอสทีเอสกับเอสเอสของตัวอย่างวงจรอสมวาร half จากข้อ 4.2.3 (ต่อ)

ขั้นแรก	ขั้นสุดท้าย	ลำดับการเปลี่ยนแปลงสัญญาณรอบปัจจุบันและก่อนหน้า	เอสทีเอสที่พบ
144	160	Ri1 > Ro1 > Ao1 > Ai1 > Ro0 > Ri0 > Ai0 > Ao0 > Ri1 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ao0 > Ai0 >	STS 7
154	170	Ri1 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ao0 > Ai0 > Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ai0 > Ao0 >	
164	180	Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ai0 > Ao0 > Ri1 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ao0 > Ai0 >	
174	190	Ri1 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ao0 > Ai0 > Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ri1 >	
184	200	Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ri1 > Ai0 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ai0 >	
194	210	Ai0 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ai0 > Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ai0 >	
204	220	Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ai0 > Ri1 > Ro1 > Ao1 > Ai1 > Ro0 > Ri0 > Ao0 > Ai0 >	STS 10
214	230	Ri1 > Ro1 > Ao1 > Ai1 > Ro0 > Ri0 > Ao0 > Ai0 > Ri1 > Ro1 > Ao1 > Ai1 > Ro0 > Ri0 > Ai0 > Ao0 >	
224	240	Ri1 > Ro1 > Ao1 > Ai1 > Ro0 > Ri0 > Ai0 > Ao0 > Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ai0 >	STS 12
234	250	Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ai0 > Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ai0 > Ao0 >	STS 1
244	260	Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ai0 > Ao0 > Ri1 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ao0 > Ri1 >	
254	270	Ri1 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ao0 > Ri1 > Ai0 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ai0 > Ao0 >	
264	280	Ai0 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ai0 > Ao0 > Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ai0 > Ao0 >	
274	290	Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ai0 > Ao0 > Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ri1 >	

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ตารางที่ ข-2 ผลการทวนสอบเอสทีเอสกับเอสเอสของตัวอย่างวงจรอสมวาร half จากข้อ 4.2.3 (ต่อ)

ขั้นแรก	ขั้นสุดท้าย	ลำดับการเปลี่ยนแปลงสัญญาณรอบปัจจุบันและก่อนหน้า	เอสทีเอสที่พบ
284	300	Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ri1 > Ai0 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ri1 >	
294	310	Ai0 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ri1 > Ai0 > Ro1 > Ao1 > Ai1 > Ro0 > Ri0 > Ai0 > Ao0 >	
304	320	Ai0 > Ro1 > Ao1 > Ai1 > Ro0 > Ri0 > Ai0 > Ao0 > Ri1 > Ro1 > Ai1 > Ao1 > Ro0 > Ri0 > Ai0 > Ao0 >	STS 4
314	330	Ri1 > Ro1 > Ai1 > Ao1 > Ro0 > Ri0 > Ai0 > Ao0 > Ri1 > Ro1 > Ao1 > Ai1 > Ro0 > Ri0 > Ai0 > Ao0 >	
324	340	Ri1 > Ro1 > Ao1 > Ai1 > Ro0 > Ri0 > Ai0 > Ao0 > Ri1 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ao0 > Ai0 >	
334	350	Ri1 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ao0 > Ai0 > Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ai0 >	
344	360	Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ai0 > Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ai0 > Ao0 >	
354	370	Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ai0 > Ao0 > Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ai0 >	
364	380	Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ai0 > Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ai0 > Ao0 >	
374	390	Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ai0 > Ao0 > Ri1 > Ro1 > Ai1 > Ao1 > Ro0 > Ri0 > Ao0 > Ri1 >	
384	400	Ri1 > Ro1 > Ai1 > Ao1 > Ro0 > Ri0 > Ao0 > Ri1 > Ai0 > Ro1 > Ai1 > Ao1 > Ro0 > Ri0 > Ai0 > Ao0 >	
394	410	Ai0 > Ro1 > Ai1 > Ao1 > Ro0 > Ri0 > Ai0 > Ao0 > Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ri1 >	
404	420	Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ri1 > Ai0 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ai0 > Ao0 >	
414	430	Ai0 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ai0 > Ao0 > Ri1 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ai0 > Ao0 >	STS 6

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ ข-2 ผลการทวนสอบเอสทีเอสกับเอสเอสของตัวอย่างวงจรอสมวาร half จากข้อ 4.2.3 (ต่อ)

ขั้นแรก	ขั้นสุดท้าย	ลำดับการเปลี่ยนแปลงสัญญาณรอบปัจจุบันและก่อนหน้า	เอสทีเอสที่พบ
424	440	Ri1 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ai0 > Ao0 > Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ai0 > Ao0 >	
434	450	Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ai0 > Ao0 > Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ai0 >	
444	460	Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ai0 > Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ai0 > Ao0 >	
454	470	Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ai0 > Ao0 > Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ai0 >	
464	480	Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ai0 > Ri1 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ai0 > Ao0 >	
474	490	Ri1 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ai0 > Ao0 > Ri1 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ai0 > Ao0 >	
484	500	Ri1 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ai0 > Ao0 > Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ai0 > Ao0 >	
494	510	Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ai0 > Ao0 > Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ai0 >	
504	520	Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ao0 > Ai0 > Ri1 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ao0 > Ai0 >	
514	530	Ri1 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ao0 > Ai0 > Ri1 > Ro1 > Ai1 > Ao1 > Ro0 > Ri0 > Ao0 > Ri1 >	
524	540	Ri1 > Ro1 > Ai1 > Ao1 > Ro0 > Ri0 > Ao0 > Ri1 > Ai0 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ai0 > Ao0 >	
534	550	Ai0 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ai0 > Ao0 > Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ai0 > Ao0 >	
544	560	Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ai0 > Ao0 > Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ai0 > Ao0 >	
554	570	Ri1 > Ro1 > Ai1 > Ao1 > Ri0 > Ro0 > Ai0 > Ao0 > Ri1 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ai0 > Ao0 >	

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ตารางที่ ข-2 ผลการทวนสอบเอสทีเอสกับเอสเอสของตัวอย่างวงจรอสมวาร half จากข้อ 4.2.3 (ต่อ)

ขั้นแรก	ขั้นสุดท้าย	ลำดับการเปลี่ยนแปลงสัญญาณรอบปัจจุบันและก่อนหน้า	เอสทีเอสที่พบ
564	**580**	**Ri1 > Ro1 > Ao1 > Ai1 > Ri0 > Ro0 > Ai0 > Ao0 > Ri1 > Ro1 > Ao1 > Ai1 > Ro0 > Ai0 > Ri0 > Ao0 >**	**STS 8**
574	590	Ri1 > Ro1 > Ao1 > Ai1 > Ro0 > Ai0 > Ri0 > Ao0 > Ri1 > Ro1 > Ao1 > Ri0 > Ai1 > Ro0 > Ao0 > Ai0 >	
…	…	…	

ตารางที่ ข-3 ผลการทวนสอบเอสทีเอสกับเอสเอสของตัวอย่างวงจรอสมวาร C-element แบบสนใจค่าที่ไม่ใช่อินพุต จากข้อที่ 4.3.3

ขั้นแรก	ขั้นสุดท้าย	ลำดับการเปลี่ยนแปลงสัญญาณรอบปัจจุบันและก่อนหน้า	เอสทีเอสที่พบ
1	17	P11 > Bi1 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0 > Ai1 >	
1	33	P11 > Bi1 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0 > Ai1 > Bi1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0 > Bi1 >	STS 2
18	49	Bi1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0 > Bi1 > Ai1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Ai0 > Bi0 > P20 > P10 > P31 > P40 > Co0 > Bi1 >	STS 7
34	65	Ai1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Ai0 > Bi0 > P20 > P10 > P31 > P40 > Co0 > Bi1 > Ai1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Ai0 > Bi0 > P20 > P10 > P31 > P40 > Co0 > Ai1 >	
50	81	Ai1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Ai0 > Bi0 > P20 > P10 > P31 > P40 > Co0 > Ai1 > Bi1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0 > Ai1 >	

ตารางที่ ข-3 ผลการทวนสอบเอสทีเอสกับเอสเอสของตัวอย่างวงจรอสมวาร C-element แบบสนใจค่าที่ไม่ใช่อินพุต จากข้อที่ 4.3.3 (ต่อ)

ขั้นแรก	ขั้นสุดท้าย	ลำดับการเปลี่ยนแปลงสัญญาณรอบปัจจุบันและก่อนหน้า	เอสทีเอสที่พบ
66	97	Bi1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0 > Ai1 > Bi1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0 > Bi1 >	
82	113	Bi1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0 > Bi1 > Ai1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0 > Ai1 >	STS 8
98	129	Ai1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0 > Ai1 > Bi1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0 > Bi1 >	
114	145	Bi1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0 > Bi1 > Ai1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0 > Ai1 >	
130	161	Ai1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0 > Ai1 > Bi1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Ai0 > Bi0 > P20 > P10 > P31 > P40 > Co0 > Ai1 >	STS 1
146	177	Bi1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Ai0 > Bi0 > P20 > P10 > P31 > P40 > Co0 > Ai1 > P11 > Bi1 > P21 > P50 > P41 > P30 > P51 > Co1 > Ai0 > Bi0 > P20 > P10 > P31 > P40 > Co0 > Ai1 >	STS 4
162	193	P11 > Bi1 > P21 > P50 > P41 > P30 > P51 > Co1 > Ai0 > Bi0 > P20 > P10 > P31 > P40 > Co0 > Ai1 > Bi1 > P11 > P21 > P50 > P41 > P30 > P51 > Co1 > Bi0 > Ai0 > P20 > P10 > P31 > P40 > Co0 > Bi1 >	

ประวัติผู้เขียนวิทยานิพนธ์

นายวีระศักดิ์ ล่อซุ่นนี้ เกิดเมื่อวันที่ 18 มิถุนายน พ.ศ.2528 ที่จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาปริญญาตรีวิทยาศาสตรบัณฑิต ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยราชภัฏจันทรเกษม ในปีการศึกษา 2551 และเข้าศึกษาต่อในหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย เมื่อ พ.ศ. 2554 ปัจจุบันทำงานที่ บริษัท ซาบีน่า ฟาร์อีสท์ จำกัด ตำแหน่งโปรแกรมเมอร์

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University