การเพิ่มจุดทดสอบสำหรับการทดสอบวงจรอสมวารแบบควอไซดีเลย์อินเซนซิทีฟ

นาย วุฒิชัย เลิศศิริสัมพันธ์

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2543
ISBN 974-347-073-5
ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

TEST POINT INSERTION FOR TESTING QUASI-DELAY-INSENSITIVE ASYNCHRONOUS CIRCUITS

Mr. Wuthichai Lertsirisumpan

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

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

Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University

Academic Year 2000

ISBN 974-347-073-5

หัวข้อวิทยานิพนธ์	การเพิ่มจุดทดสอบสำหรับการทดสอบวงจรอสมวารแบบควอไซดีเลย์อินเซนซิทีฟ
โดย	นาย วุฒิชัย เลิศศิริสัมพันธ์
ภาควิชา	วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษา	ดร. อาทิตย์ ทองทักษ์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์

(ศาสตราจารย์ ดร. สมศักดิ์ ปัญญาแก้ว)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ

(ผู้ช่วยศาสตราจารย์ ดร. สาธิต วงศ์ประทีป)

..... อาจารย์ที่ปรึกษา

(อาจารย์ ดร. อาทิตย์ ทองทักษ์)

..... กรรมการ

(ผู้ช่วยศาสตราจารย์ ดร. สมชาย ประสิทธิ์จูตระกูล)

..... กรรมการ

(ผู้ช่วยศาสตราจารย์ บุญชัย โสวรรณวณิชกุล)

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

วุฒิชัย เลิศศิริสัมพันธ์ : การเพิ่มจุดทดสอบสำหรับทดสอบวงจรอสมวารแบบควอไซดีเลย์อินเซนซิทีฟโดยซิกแนลทรานสิชันกราฟ (TEST POINT INSERTION FOR TESTING QUASI-DELAY-INSENSITIVE ASYNCHRONOUS CIRCUITS) อ.ที่ปรึกษา : ดร.อาทิตย์ ทองทักษ์, 97 หน้า. ISBN 974-347-073-5.

การทดสอบวงจรเป็นกระบวนการที่ใช้สำหรับตรวจสอบการทำงานของวงจรเมื่อมีข้อผิดพร่องเกิดขึ้นในวงจร วิทยานิพนธ์ฉบับนี้นำเสนอการออกแบบขั้นตอนวิธีการเติมจุดทดสอบในวงจรอสมวารแบบควอไซดีเลย์อินเซนซิทีฟที่มีโมเดลข้อผิดพร่องแบบ Stuck-at Fault ซึ่งขั้นตอนการเติมจุดทดสอบเป็นส่วนหนึ่งของการทดสอบวงจรอสมวาร โดยจะใช้วิธีการเติมจุดทดสอบลงในวงจรซึ่งแบ่งเป็นจุดควบคุม (Control Point) และจุดสังเกต (Observe Point) เพื่อให้วงจรสามารถทดสอบได้

ในขั้นตอนวิธีการทดสอบวงจรอสมวารจะใช้การจำลองพฤติกรรมโดยใช้แผนภาพคล้ายซิกแนลทรานสิชันกราฟและประยุกต์ SSG ช่วยในการลดความซับซ้อนในการแสดงพฤติกรรมของวงจร และปรับปรุงค่า Fault Coverage ของวงจรให้มากขึ้นโดยใช้ SSG และ ESSG เพื่อให้วงจรเติมจุดทดสอบน้อยลง

จากผลการทดลองแสดงให้เห็นว่าการเติมจุดทดสอบในวงจรที่เกิดข้อผิดพร่องสามารถทำให้วงจรสามารถทดสอบได้ทุกกรณี

ภาควิชา……วิศวกรรมคอมพิวเตอร์…… ลายมือชื่อนิสิต ……

สาขาวิชา…..วิทยาศาสตร์คอมพิวเตอร์.. ลายมือชื่ออาจารย์ที่ปรึกษา ……

ปีการศึกษา….2543…….. ลายมือชื่ออาจารย์ที่ปรึกษาร่วม ……

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

##4070430821 **MAJOR** COMPUTER ENGINEERING

KEYWORD: TESTING / FAULT / ASYNCHRONOUS CIRCUITS / QUASI-DELAY INSENSITIVE / SIGNAL TRANSITION GRAPH

WUTHICHAI LERTSIRISUMPAN : TEST POINT INSERTION FOR TESTING QUASI-DELAY INSENSITIVE ASYNCHRONOUS CIRCUITS. THESIS ADVISOR : ARTHIT THONGTAK, Ph.D. 97 pp. ISBN. 974-347-073-5.

Testing is process to detect the fault. This Thesis proposes a design of the test point insertion for testing quasi-delay insensitive (QDI) asynchronous circuits that use stuck-at fault model. Test point insertion is one process in testing process by using control point and observe point for increasing fault coverage of the circuit.

In asynchronous circuits testing process, we use stg-like diagram for showing circuit behavior and apply stable state graph (SSG) for decreasing complex of circuit behavior. We use Stable state graph and Extend Stable state graph (ESSG) for increasing fault coverage of the circuits.

Experiments on the benchmark circuits show that test point insertion can detect all permanent stuck-at faults.

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

Department.... Computer Engineering.... **Student's signature**.....

Field of study.. Computer Science..... **Advisor's signature**.....

Academic year.....2000..... **Co-Advisor's signature**.....

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยความช่วยเหลืออย่างยิ่งของอาจารย์ ดร.อาทิตย์ ทองทักษ์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ ซึ่งท่านได้ให้คำแนะนำและข้อคิดเห็นต่าง ๆ ในการวิจัยมาด้วยดี

ขอขอบคุณ คุณวิมล เลิศศิริสัมพันธ์ ที่ให้การสนับสนุนทางด้านการเงิน

ขอขอบคุณ สมาชิก Digital System Engineering Laboratory ทุกท่านที่ให้ความช่วยเหลือด้านคำปรึกษาและข้อเสนอแนะในการทำวิทยานิพนธ์

ขอขอบคุณ ห้อง Digital System Engineering Laboratory ที่ให้ความช่วยเหลือในด้านสถานที่และเครื่องมือที่ใช้ในการวิจัย

ขอขอบคุณ เพื่อน ๆ ทุก ๆ คนที่ได้ให้คำปรึกษาและความช่วยเหลือในด้านต่าง ๆ ซึ่งทำให้ทำงานวิจัยได้อย่างราบรื่น

ท้ายนี้ ผู้วิจัยใคร่ขอกราบขอบพระคุณ บิดา-มารดา ที่ให้การสนับสนุนและให้กำลังใจแก่ผู้วิจัยเสมอมาจนสำเร็จการศึกษา

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญ

หน้า

บทคัดย่อภาษาไทย..... ง
บทคัดย่อภาษาอังกฤษ..... จ
กิตติกรรมประกาศ..... ฉ
สารบัญ..... ช
สารบัญภาพ..... ฌ
สารบัญตาราง..... ฎ
บทที่
1 บทนำ..... 1
1.1 ความเป็นมาและความสำคัญของปัญหา..... 1
1.2 วัตถุประสงค์การวิจัย..... 2
1.3 ขอบเขตการวิจัย..... 2
1.4 ขั้นตอนและวิธีดำเนินการวิจัย..... 2
1.5 ประโยชน์ที่คาดว่าจะได้รับ..... 3
1.6 เนื้อหาของวิทยานิพนธ์..... 3
2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง..... 4
2.1 ข้อผิดพร่อง..... 4
2.2 การทดสอบวงจรสมวาร..... 4
2.3 โมเดลความหน่วง..... 6
2.4 ข้อกำหนดของวงจรอสมวาร..... 8
2.5 คุณสมบัติการตอบรับ..... 9
2.6 สรุป..... 9
3 ซิกแนลทรานสิชันกราฟ..... 11
3.1 ซิกแนลทรานสิชันกราฟ..... 11
3.2 คุณสมบัติของซิกแนลทรานสิชันกราฟ..... 12
3.3 ขั้นตอนการสังเคราะห์วงจรจากซิกแนลทรานสิชันกราฟ..... 15
3.4 แผนภาพคล้ายซิกแนลทรานสิชันกราฟ..... 17
3.5 สรุป..... 18

สารบัญ (ต่อ)

หน้า

4 การทดสอบวงจรอสมวาร..... 19
4.1 เงื่อนไขการทดสอบวงจรอสมวาร..... 19
4.2 การหาสถานะเริ่มต้นของวงจร..... 19
4.3 การหาสถานะคงที่ของวงจร..... 21
4.4 Stable State Graph..... 22
4.5 การทดสอบวงจรอสมวาร..... 23
4.6 Extend Stable State Graph..... 25
4.7 การหาสถานะวิกฤต..... 27
4.8 การจำลองข้อผิดพร่องโดยใช้แผนภาพคล้ายซิกแนลทรานสิชันกราฟ..... 28
4.9 พฤติกรรมของวงจรเมื่อเกิดข้อผิดพร่อง..... 32
4.10 สรุป..... 34
5 การเติมจุดทดสอบวงจรอสมวาร..... 36
5.1 พฤติกรรมของวงจรที่เกิดข้อผิดพร่องและไม่สามารถทดสอบได้..... 36
5.2 จุดทดสอบ..... 38
5.3 เงื่อนไขในการเติมจุดทดสอบ..... 39
5.4 ขั้นตอนวิธีการเติมจุดทดสอบ..... 39
5.5 ผลการทดลองการเติมจุดทดสอบ..... 42
5.6 แนวทางการลดจุดทดสอบ..... 46
5.7 ผลการทดลองและสรุปผลการทดลอง..... 50
5.8 สรุป..... 50
6 สรุปผลการวิจัยและข้อเสนอแนะ..... 54
6.1 สรุปผลการวิจัย..... 54
6.2 ข้อเสนอแนะ..... 55
รายการอ้างอิง..... 56
ภาคผนวก
ภาคผนวก ก..... 58
ประวัติผู้วิจัย..... 97

สารบัญภาพ

		หน้า
รูปที่ 2.1	รูปแบบของวงจรเชิงลำดับแบบสมวาร……..	5
รูปที่ 2.2	Pseudocombinational iterative array ของวงจรเชิงลำดับแบบสมวาร……..	5
รูปที่ 2.3	(a) รูปแบบของวงจรเชิงลำดับแบบสมวาร (b) Pseudocombinational iterative array ของวงจรเชิงลำดับแบบสมวาร (c) รูปแบบ Time Frame ขณะเวลา T(i) ……	6
รูปที่ 2.4	C-Element……..	8
รูปที่ 2.5	ซิกแนลทรานสิชันกราฟของวงจร fifo……	9
รูปที่ 3.1	ตัวอย่างซิกแนลทรานสิชันกราฟ……	11
รูปที่ 3.2	ตัวอย่างซิกแนลทรานสิชันกราฟที่มีคุณสมบัติ input choice……..	13
รูปที่ 3.3	ตัวอย่างซิกแนลทรานสิชันกราฟที่ไม่มีคุณสมบัติ (a) liveness (b) Safety (c) Persistency (d) Consistency (e) Unique state assignment และ Single-cycle transition……	14
รูปที่ 3.4	ขั้นตอนการสังเคราะห์วงจรจากซิกแนลทรานสิชันกราฟ……..	15
รูปที่ 3.5	State Graph ของซิกแนลทรานสิชันกราฟรูป 3.1……..	15
รูปที่ 3.6	ขั้นตอนการสร้าง Karnaugh Map จาก State Graph……..	16
รูปที่ 3.7	วงจรที่สังเคราะห์ได้จาก state graph……..	17
รูปที่ 3.8	แผนภาพคล้ายซิกแนลทรานสิชันกราฟของวงจรรูปที่ 3.7……	18
รูปที่ 4.1	ตัวอย่าง Stable State Graph ของ C-Element……..	23
รูปที่ 4.2	วงจรและซิกแนลทรานสิชันกราฟของวงจร C-Element……..	24
รูปที่ 4.3	ตัวอย่างซิกแนลทรานสิชันกราฟและ Stable State Graph……	25
รูปที่ 4.4	ตัวอย่าง Extend Stable State Graph……	25
รูปที่ 4.5	วงจรและซิกแนลทรานสิชันกราฟของวงจร Chu133……..	26
รูปที่ 4.6	แผนภาพคล้ายซิกแนลทรานสิชันกราฟของวงจรรูป 4.2 (a) วงจรปกติ (b) เมื่อวงจรเกิดข้อผิดพร่อง stuck-at 1 ที่สัญญาณ P52……	30
รูปที่ 4.7	วงจรและซิกแนลทรานสิชันกราฟของวงจร half……..	31
รูปที่ 4.8	แผนภาพคล้ายซิกแนลทรานสิชันกราฟแสดงพฤติกรรมของวงจร half (a) ปกติ (b) เมื่อเกิดข้อผิดพร่องแบบ stuck –at 1 ที่สัญญาณ Ro……	31
รูปที่ 5.1	รูปแบบการเกิดการเปลี่ยนแปลงของสัญญาณ (a) ปกติ (b) เกิดข้อผิดพร่อง	38

สารบัญภาพ (ต่อ)

หน้า

รูปที่ 5.2 ตัวอย่างจุดทดสอบชนิด (a) จุดควบคุม เติมสัญญาณ (b) จุดควบคุม ชนิดเกต (c) จุดสังเกต..... 39
รูปที่ 5.3 วงจรและซิกแนลทรานสิชันกราฟของวงจร ebergen..... 43
รูปที่ 5.4 แผนภาพคล้ายซิกแนลทรานสิชันกราฟแสดงพฤติกรรมของวงจร ebergen เมื่อเกิดข้อผิดพร่องแบบ stuck –at 0 ที่สัญญาณ Ai4..... 43
รูปที่ 5.5 วงจรและซิกแนลทรานสิชันกราฟของวงจร C-Element..... 44
รูปที่ 5.6 แผนภาพคล้ายซิกแนลทรานสิชันกราฟแสดงพฤติกรรมของวงจร C-element (a) ปกติ (b) เกิดข้อผิดพร่องแบบ stuck –at 1 ที่สัญญาณ P52..... 44
รูปที่ 5.7 วงจรและซิกแนลทรานสิชันกราฟของวงจร Converta..... 45
รูปที่ 5.8 แผนภาพคล้ายซิกแนลทรานสิชันกราฟแสดงพฤติกรรมของวงจร C-element (a) ปกติ (b) เกิดข้อผิดพร่องแบบ stuck –at 1 ที่สัญญาณ P31..... 45
รูปที่ 5.9 แผนภาพคล้ายซิกแนลทรานสิชันกราฟของวงจร C-element เมื่อเติมจุดควบคุม 0 ในสัญญาณ P3..... 46
รูปที่ 6.1 แผนภาพคล้ายซิกแนลทรานสิชันกราฟ..... 55

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญตาราง

หน้า

ตารางที่ 2.1 สถานะของ C-Element..... 7

ตารางที่ 4.1 แสดงผลการทดสอลวงจรเกณฑ์เปรียบเทียบสมรรถนะ..... 35

ตารางที่ 5.1 แสดงการเติมจุดทดสอบโดยใช้ขั้นตอนวิธีการเติมจุดทดสอบ..... 51

ตารางที่ 5.2 ผลการทดลองแสดงจุดทดสอบที่เติมในวงจรเมื่อเกิดข้อผิดพร่องในวงจร..... 52

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

วงจรดิจิตอลแบ่งออกเป็น 2 ประเภทได้แก่ วงจรแบบสมวาร (Synchronous) และวงจรแบบอสมวาร (Asynchronous) โดยข้อแตกต่างระหว่างวงจร 2 ประเภทนี้คือ วงจรแบบสมวารจะมีสัญญาณนาฬิกา (Clock signal) เป็นตัวควบคุมการทำงานของวงจร ดังนั้นการออกแบบวงจรแบบสมวารจึงมีความยืดหยุ่นกว่าวงจรแบบอสมวาร จึงทำให้วงจรแบบสมวารถูกใช้อย่างแพร่หลาย แต่ปัจจุบันนี้เทคโนโลยีด้านการออกแบบวงจรได้ถูกพัฒนาไปอย่างรวดเร็วทำให้วงจรแบบสมวารไม่เหมาะสมกับเทคโนโลยีสมัยใหม่ เนื่องจากปัญหาการแกว่งของสัญญาณนาฬิกา (Clock skew), สิ้นเปลืองพลังงานมาก ฯลฯ สำหรับปัญหาต่าง ๆ เหล่านี้จะไม่พบในวงจรแบบอสมวาร เนื่องจากวงจรประเภทนี้ไม่มีสัญญาณนาฬิกา ดังนั้นจึงทำให้วงจรแบบอสมวารเริ่มเป็นที่สนใจอีกครั้ง

ดังนั้นเมื่อวงจรแบบอสมวารเริ่มเป็นที่สนใจอีกครั้งจึงได้มีการศึกษาวิธีการออกแบบวงจรชนิดนี้กันอย่างกว้างขวาง แต่งานวิจัยส่วนใหญ่มักจะมุ่งเน้นไปทางด้านการสังเคราะห์วงจรเป็นส่วนใหญ่ ปัญหาการทดสอบวงจรยังไม่ได้รับความสนใจกันมากนัก ทั้งที่ขั้นตอนการทดสอบวงจรเป็นขั้นตอนที่สำคัญในการออกแบบวงจร

ในระยะแรกก็ได้นำวิธีการทดสอบของวงจรแบบสมวารมาประยุกต์ใช้ แต่พบว่าวิธีการเหล่านั้นไม่สามารถนำมาประยุกต์ใช้ได้เนื่องจากวงจรแบบอสมวารไม่มีสัญญาณนาฬิกาซึ่งทำให้เกิดปัญหาตามมาคือวงจรจะมีค่าควบคุม (Controllability) และค่าสังเกต (Observability) ต่ำ และที่สำคัญวิธีการเหล่านั้นยังไม่มีการคำนึงถึงโมเดลความหน่วงซึ่งเป็นสิ่งสำคัญในการออกแบบและทดสอบวงจรอสมวาร

ในการศึกษาวิธีทดสอบวงจรแบบอสมวาร สิ่งสำคัญที่จะต้องรู้คือข้อกำหนดของวงจร ซึ่งสามารถแสดงได้หลายรูปแบบ เช่น เพตตริเนต (Petri net), ซิกแนลทรานสิชันกราฟ (Signal Transition Graph) เป็นต้น เมื่อเกิดข้อผิดพร่องแบบ single stuck-at ในวงจรอสมวารจะเป็นผลทำให้เกิดเหตุการณ์ได้ดังนี้ คือไม่เกิดการเปลี่ยนแปลงสัญญาณตามข้อกำหนดของวงจร หรือเกิดการเปลี่ยนแปลงสัญญาณก่อนข้อกำหนด หรือเกิดทั้งสองเหตุการณ์[5] ในกรณีที่ไม่เกิดการเปลี่ยนแปลงของสัญญาณนั้นสามารถทำการตรวจสอบได้ง่ายเพราะผลลัพธ์ของวงจรไม่เกิดขึ้นทำให้ทราบว่าวงจรมีข้อผิดพร่องเกิดขึ้น แต่สำหรับการเกิดการเปลี่ยนแปลงสัญญาณก่อนข้อกำหนดนั้นสามารถทำการทดสอบได้ยากเพราะผลลัพธ์ที่ได้อาจจะตรงตามผลลัพธ์จากข้อกำหนดของวงจร[5] แนวทางหนึ่งในการแก้ไขเพื่อให้วงจรที่เกิดเหตุการณ์นี้สามารถทดสอบหาข้อผิดพร่องได้นั้น

คือการปรับปรุงค่าความสามารถในการทดสอบ (Testability) โดยการเติมจุดทดสอบลงในวงจรเพื่อให้ค่านี้เพิ่มขึ้น

ดังนั้นในงานวิจัยนี้จึงได้นำเสนอวิธีการเติมจุดทดสอบลงในวงจรอสมวารแบบควอไซดีเลย์อินเซนซิทีฟ (Quasi-Delay-Insensitive; QDI) โดยอาศัยแผนภาพคล้ายซิกแนลทรานสิชันกราฟ (STG-like Diagram) แสดงพฤติกรรมของวงจรที่มีข้อผิดพร่องและวิเคราะห์จุดที่ต้องเติมจุดทดสอบเพื่อให้มีค่า Fault Coverage มากขึ้น

1.2 วัตถุประสงค์การวิจัย

เพื่อออกแบบขั้นตอนวิธีการในการเติมจุดทดสอบสำหรับวงจรอสมวารที่ใช้โมเดลความหน่วงแบบควอไซดีเลย์อินเซนซิทีฟเพื่อให้วงจรสามารถทดสอบได้

1.3 ขอบเขตของการวิจัย

1. วงจรที่นำมาพิจารณาจะใช้โมเดลความหน่วงแบบควอไซดีเลย์อินเซนซิทีฟ
2. ข้อผิดพร่องของวงจรเป็นแบบ Single stuck-at
3. เป็นการทดสอบหาข้อผิดพร่องของวงจรในระดับเกต
4. นำเอาวงจรเกณฑ์เปรียบเทียบสมรรถนะ (Benchmark) ของ Park [8] เป็นกรณีศึกษา

1.4 ขั้นตอนและวิธีดำเนินงานวิจัย

1. ศึกษาลักษณะการทำงานของวงจรแบบอสมวาร
2. ศึกษาลักษณะโมเดลความหน่วงชนิดต่าง ๆ สำหรับวงจรแบบอสมวาร
3. ศึกษารูปแบบการนำเสนอข้อกำหนดของวงจรโดยซิกแนลทรานสิชันกราฟ
4. ศึกษาโมเดลความหน่วงของวงจรดิจิตอล
5. ศึกษาวิธีการหาค่าเวกเตอร์ทดสอบ
6. ศึกษาวิธีการสร้างและวิเคราะห์ข้อกำหนดของวงจรแบบอสมวารที่มีข้อผิดพร่อง
7. ออกแบบขั้นตอนวิธีการเพิ่มจุดทดสอบของวงจรแบบอสมวาร
8. ทดสอบขั้นตอนวิธีการโดยใช้วงจรเปรียบเทียบสมรรถนะของ Park [8] วัดผลระหว่างจำนวนจุดทดสอบที่เพิ่มกับ Fault Coverage ที่ต้องการ
9. สรุปผลการวิจัยและประเมินผล
10. เสนอรายงานการวิจัยในรูปแบบของวิทยานิพนธ์

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. เป็นแนวทางในการพัฒนาวิธีการทดสอบวงจรแบบอสมวาร
2. ทำให้สามารถเพิ่มอัตรา Fault Coverage ของวงจรแบบอสมวารได้
3. เป็นแนวทางในการพัฒนาวิธีการทดสอบวงจรแบบอสมวารที่ใช้โมเดลข้อผิดพร่องชนิดอื่น

1.6 เนื้อหาของวิทยานิพนธ์

เนื้อหาของวิทยานิพนธ์ฉบับนี้จะแบ่งออกเป็นหกบทย่อย ๆ โดยบทที่ 1 จะเป็นบทนำ ส่วนในบทที่ 2 จะกล่าวถึงทฤษฎีต่าง ๆ ที่เกี่ยวข้องกับงานวิจัย ในบทที่ 3 จะอธิบายเกี่ยวกับซิกแนลทรานสิชันกราฟและแผนภาพคล้ายซิกแนลทรานสิชันกราฟซึ่งเป็นเครื่องมือที่ใช้ในการแสดงการทำงานของวงจร บทที่ 4 จะเป็นการอธิบายเกี่ยวกับพฤติกรรมของวงจรเมื่อเกิดข้อผิดพร่องและวิธีการทดสอบวงจรแบบอสมวาร บทที่ 5 จะนำเสนอถึงวิธีการเติมจุดทดสอบลงในวงจรอสมวารที่ทดสอบไม่ได้ โดยใช้วิธีการที่ออกแบบไว้ และบทที่ 6 จะเป็นบทสรุปการวิจัยรวมถึงข้อเสนอแนะต่าง ๆ ซึ่งควรนำไปปรับปรุง

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงทฤษฎีและงานวิจัยต่าง ๆ ที่นำมาประยุกต์ใช้ เพื่อการออกแบบขั้นตอนการเติมจุดทดสอบในวงจรอสมวารแบบควอไซดีเลย์อินเซนซิทีฟ โดยจะกล่าวถึงรูปแบบข้อผิดพร่องที่ใช้ในวงจร[4], การทดสอบข้อผิดพร่องของวงจรสมวาร [1,4], โมเดลความหน่วง[7], ข้อกำหนดของวงจรอสมวาร[2] และคุณสมบัติการตอบรับ (Acknowledgement Property)[5,7]

2.1 ข้อผิดพร่อง (Fault)

ข้อผิดพร่องแบ่งเป็น 3 ชนิด คือ

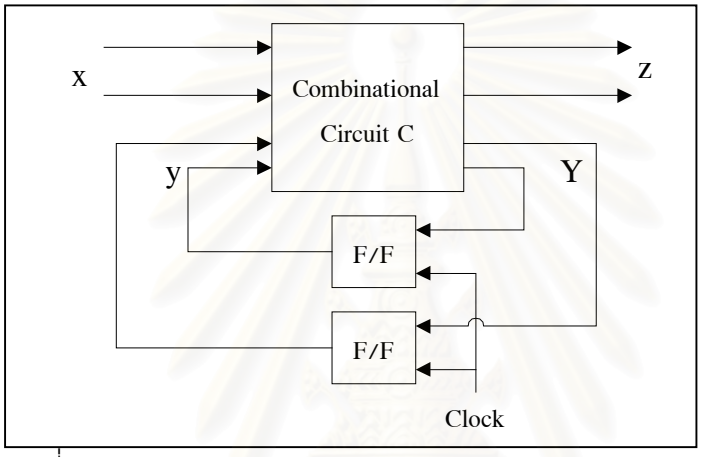
1. ถาวร (Permanent) หมายถึงข้อผิดพร่องที่ทำงานตลอดเวลาหลังจากที่ข้อผิดพร่องเกิดขึ้น
2. อินเตอร์มิแตนท์ (Intermittent) หมายถึงข้อผิดพร่องที่เกิดขึ้นแบบชั่วครั้งชั่วคราวเช่น การเกิดข้อผิดพร่องเนื่องจากอุณหภูมิของอุปกรณ์ เป็นต้น
3. ทรานเสียน (Transient) หมายถึงข้อผิดพร่องที่เกิดขึ้นชั่วระยะเวลาสั้น ๆ และอาจจะไม่เกิดขึ้นอีกเลย เช่น การถูกรบกวนจากสนามแม่เหล็ก เป็นต้น

สำหรับงานวิจัยนี้จะใช้ โมเดลข้อผิดพร่องแบบ Stuck-at ซึ่งเป็นโมเดลที่มีการศึกษาและใช้งานกันอย่างแพร่หลายมากและสามารถนำไปประยุกต์ใช้กับโมเดลอื่น ๆ ได้ โมเดลข้อผิดพร่องแบบ Stuck-at มีอยู่ 2 ชนิดคือ Stuck-at 0 และ Stuck-at 1 ซึ่งหมายถึงเส้นสัญญาณของวงจรมีค่าของสัญญาณคงที่ตลอดเวลาโดยที่ Stuck-at 0 จะมีค่าเป็น 0 และ Stuck-at 1 จะมีค่าเป็น 1 ในวงจรหนึ่ง ๆ อาจจะเกิดข้อผิดพร่องได้จุดเดียว (single fault) หรือหลายจุด(Multiple fault) ก็ได้ แต่ในงานวิจัยนี้จะสนใจเพียงโมเดลข้อผิดพร่อง Stuck-at แบบจุดเดียวเท่านั้น

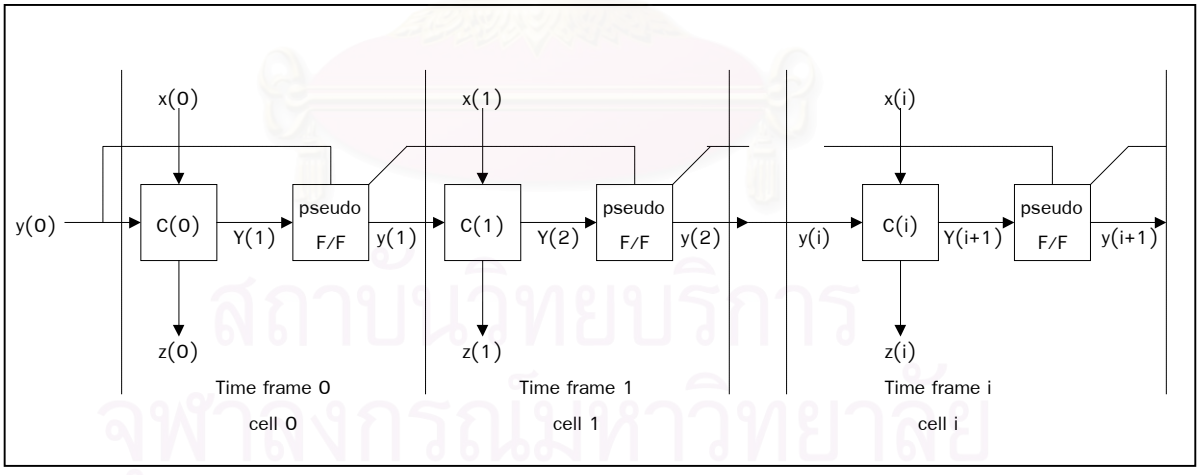
2.2 การทดสอบวงจรสมวาร

ในการทดสอบวงจรเชิงผสม (Combinational) มีวิธีการตรวจสอบหาข้อผิดพร่องแบบ Stuck-at อยู่หลายวิธีเช่น Path sensitization, D-algorithm ฯลฯ ซึ่งส่วนใหญ่จะทำการหาค่าอินพุตที่ทำให้ข้อผิดพร่องแพร่ขยาย (Propagate) ออกไปสู่เอาต์พุตของวงจร และเปรียบเทียบกับเอาต์พุตที่ควรจะได้ ซึ่งเราเรียกค่าอินพุตนี้ว่า เวกเตอร์ทดสอบ (Test Vector) โดยค่าเหล่านี้จะเป็นค่าสำหรับทดสอบข้อผิดพลาดแบบ Single stuck-at ชนิด stuck-at 0 หรือ stuck-at 1 อย่างใดอย่างหนึ่งเท่านั้น และทดสอบได้เพียงเส้นทางที่ออกไปสู่เอาต์พุตเพียงเส้นทางเดียวในวงจร ถ้าต้องการที่จะหา fault ชนิดอื่นหรือเส้นทางอื่น จะต้องทำการหาค่านี้ใหม่อีกครั้งหนึ่ง

สำหรับวงจรเชิงลำดับ (Sequential) นั้นไม่สามารถที่จะนำวิธีการเหล่านี้มาใช้ได้เนื่องจากวงจรประเภทนี้จะมีสัญญาณย้อนกลับ (feedback) คือสัญญาณเอาต์พุตของวงจรกลับมาเป็นอินพุตของวงจรอีกครั้ง การทดสอบวงจรประเภทนี้จะต้องหาค่าอินพุตของวงจรเป็นลำดับ (Test Sequence) และจะต้องเป็นการใส่ค่าอินพุตเป็นลำดับเข้าสู่วงจรตามลำดับและเวลาที่เหมาะสม ในการทดสอบวงจรเชิงลำดับแบบสมวาร (Synchronous Sequential Circuit) ซึ่งแสดงดังรูปที่ 2.1 จะทำโดยการสร้าง Pseudocombinational iterative array [1] ดังรูปที่ 2.2 โดยจะมีลักษณะเป็นวงจรเชิงผสมต่อซ้ำ ๆ กัน วงจรเชิงผสมแต่ละตัวจะเรียกว่า Time Frame และทำการทดสอบโดยใช้วิธีของวงจรเชิงผสม [5]



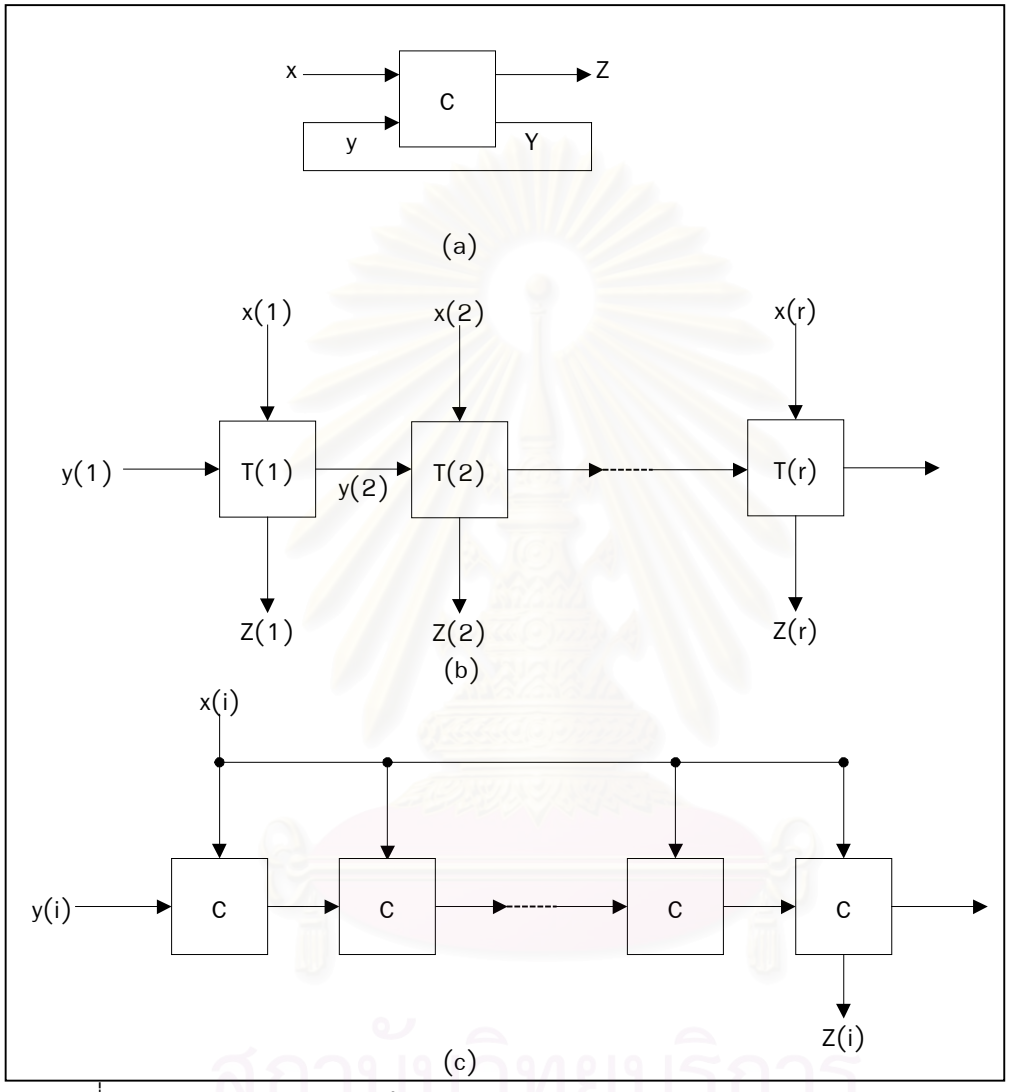
รูปที่ 2.1 รูปแบบของวงจรเชิงลำดับแบบสมวาร



รูปที่ 2.2 Pseudocombinational iterative array **ของวงจรเชิงลำดับแบบสมวาร**

แต่ในการทดสอบวงจรแบบอสมวารนั้นเราไม่สามารถใช้วิธีทดสอบสำหรับวงจรแบบสมวารได้ เนื่องจากเราจะสามารถตรวจสอบเอาต์พุตหลัก (Primary Output) ได้เมื่อวงจรอยู่ในสถานะคงที่เท่านั้นและอินพุตหลัก (Primary Input) ของวงจรจะเปลี่ยนแปลงได้เมื่อวงจรอยู่ในสถานะคงที่ ดังนั้นทำให้ Time Frame ของวงจรแบบอสมวารอาจจะประกอบไปด้วยส่วนย่อย ๆ

(Phase) มากกว่า 1 ส่วน[1] จนกระทั่งวงจรอยู่ในสถานะคงที่ดังรูปที่ 2.3 อีกทั้งในการทดสอบวงจรยังต้องคำนึงถึงโมเดลความหน่วงที่ใช้ในวงจรอีกด้วย ซึ่งวิธีการทดสอบวงจรของวงจรสมวารมักจะไม่คำนึงถึง ด้วยเหตุผลนี้เองทำให้วิธีการทดสอบวงจรแบบอสมวารไม่สามารถนำมาใช้กับวงจรแบบอสมวารได้ ดังนั้นในการศึกษาวงจรอสมวารนั้นจะต้องเข้าใจถึงโมเดลความหน่วงก่อน



รูปที่ 2.3 (a) รูปแบบวงจรเชิงลำดับแบบอสมวาร (b) pseudocombinational iterative array ของวงจรเชิงลำดับแบบอสมวาร (c) รูปแบบ Time Frame ขณะเวลา T(i) [1]

2.3 โมเดลความหน่วง (Delay Model)

ในการออกแบบวงจรอสมวารเราจำเป็นจะต้องกำหนดโมเดลความหน่วงเพื่อจะตั้งสมมติฐานเกี่ยวกับค่าความหน่วงของเกต (gate delay) และค่าความหน่วงของสัญญาณ (wire delay) สำหรับใช้ในการจำลองการทำงานของวงจร โมเดลความหน่วงที่นิยมใช้ในการออกแบบวงจรมีอยู่ 4 แบบ [7] คือ

- โมเดลความหน่วงแบบฮัฟแมน (Huffman Delay Model)
- โมเดลความหน่วงแบบดีเลย์อินเซนซิทีฟ (Delay-Insensitive (DI) Model)
- โมเดลความหน่วงแบบสปีดอินดิเพนเดนท์ (Speed Independent (SI) Model)
- โมเดลความหน่วงแบบควอไซดีเลย์อินเซนซิทีฟ (Quasi Delay Insensitive (QDI) Model)

โมเดลความหน่วงที่มีความเหมาะสมกับวงจรอสมวารมากที่สุดคือโมเดลความหน่วงแบบดีเลย์อินเซนซิทีฟ เนื่องจากมีข้อจำกัดทางด้านความหน่วงน้อยที่สุด ซึ่งมีนิยามดังนี้ [7]

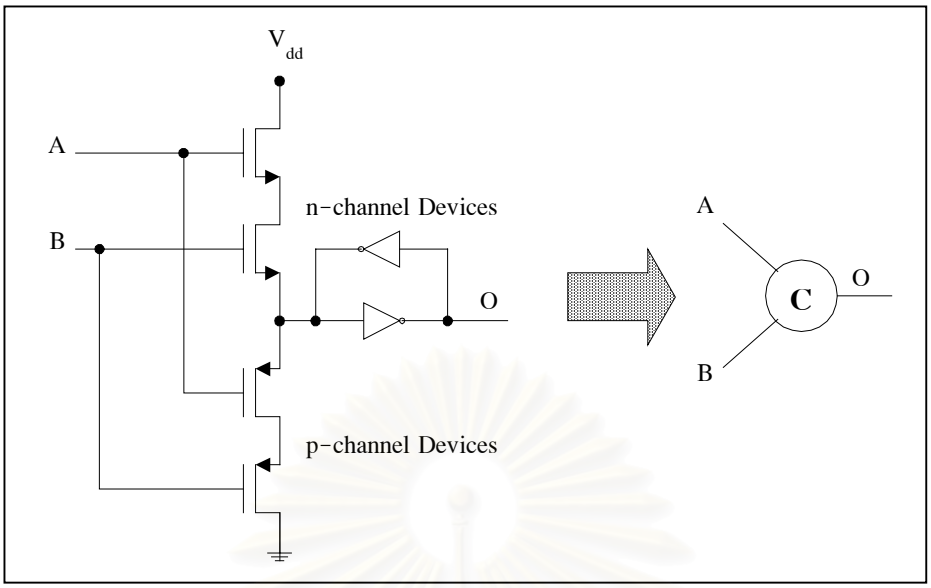
"ค่าความหน่วงของเกตและค่าความหน่วงของสัญญาณนั้นไม่ทราบค่า แต่มีขอบเขต"

แต่ในการสังเคราะห์วงจรที่มีโมเดลความหน่วงแบบดีเลย์อินเซนซิทีฟนั้น สามารถสังเคราะห์ได้ในขอบเขตที่จำกัด ทำให้ไม่สามารถประยุกต์ใช้กับวงจรทั่วไปได้ จึงได้มีการเพิ่มข้อจำกัดเกี่ยวกับความหน่วงขึ้นคือ กรณีที่มีการกระจายของสัญญาณ (fork) จะถือว่าค่าความหน่วงของสัญญาณมีค่าเท่ากัน (Isochronic fork)[7] ซึ่งถือว่าเป็นรูปแบบของโมเดลความหน่วงแบบควอไซดีเลย์อินเซนซิทีฟ

นอกจากนี้ในวงจรอสมวารที่มีโมเดลความหน่วงแบบดีเลย์อินเซนซิทีฟ และควอไซดีเลย์อินเซนซิทีฟ ยังมีเกตพิเศษเพิ่มขึ้นอีกตัวหนึ่งคือ เกต C-element ซึ่งเขียนแทนด้วยสัญลักษณ์ Ⓒ โดย C-Element มีหน้าที่ตรวจสอบสัญญาณอินพุตว่ามีค่าตรงกันหรือไม่ ถ้ามีค่า ตรงกันทั้งหมดก็จะให้ค่าอินพุตนั้นเป็นเอาต์พุต สำหรับกรณีอื่นนั้นเอาต์พุตจะคงสภาพเดิมโดยจะแสดงไว้ในตารางที่ 1.1 และมีวงจรดังรูปที่ 2.4

Input	Output
ทั้งหมดเป็น "0"	0
ทั้งหมดเป็น "1"	1
อื่น ๆ	สภาวะเดิม

ตารางที่ 1.1 ตารางแสดงสถานะของ C-Element



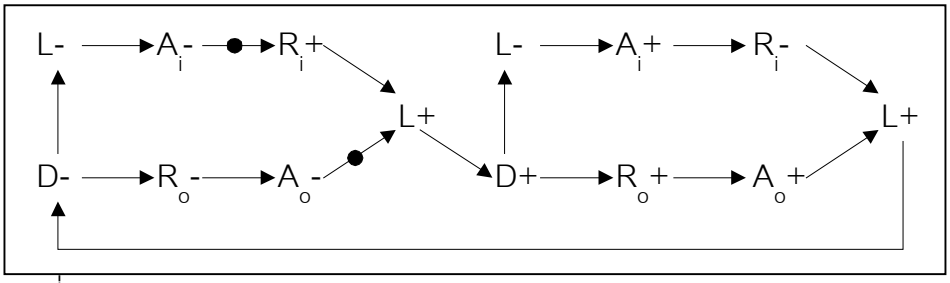
รูปที่ 2.4 C-Element

ในงานวิจัยนี้จะใช้วงจรที่มีโมเดลความหน่วงแบบควอไซดีเลย์อินเซนซิทีฟเป็นหลัก เนื่องจากโมเดลนี้มีความเหมาะสมกับวงจรอสมวารมาก เพราะมีค่าความเชื่อถือด้านเวลามาก [10] และได้ลดข้อจำกัดของวงจรที่มีโมเดลความหน่วงแบบดีเลย์อินเซนซิทีฟ ทำให้มีแนวโน้มการใช้โมเดลความหน่วงนี้กับวงจรอสมวารมาก

2.4 ข้อกำหนดของวงจรอสมวาร

ในการแสดงถึงลักษณะของวงจรอสมวารจะต้องมีข้อกำหนดของวงจร (Specification) ซึ่งส่วนใหญ่จะใช้กราฟ (Graph) เป็นตัวกำหนด และกราฟที่ใช้กันมากนั้นคือ ซิกแนลทรานสิชันกราฟ ซึ่งถูกพัฒนาโดย Tam-Anh Chu

ซิกแนลทรานสิชันกราฟเป็นกราฟแบบมีทิศทาง (Directed Graph) ซึ่งจะแสดงการเปลี่ยนแปลงของสัญญาณอินพุต, เอาต์พุตและสัญญาณภายในวงจรบางเส้นสัญญาณ ซึ่งเรียกว่า Transition แบ่งเป็น Transition ขาขึ้น (rising) หมายถึงการเปลี่ยนแปลงค่าของสัญญาณจาก 0 เป็น 1 และ Transition ขาลง (falling) หมายถึงการเปลี่ยนแปลงค่าของสัญญาณจาก 1 เป็น 0 โดยมีตัวอย่างดังรูปที่ 2.5



รูปที่ 2.5 ซิกแนลทรานสิชันกราฟของวงจร fifo [9]

เราสามารถแสดงพฤติกรรมของวงจรด้วยเซตลำดับของการเปลี่ยนแปลงสัญญาณ (Signal Transition) โดยการเปลี่ยนแปลงที่เกิดขึ้นระหว่างการเปลี่ยนแปลงสัญญาณจะมีความสัมพันธ์ที่เรียกว่า Causal Relation ซึ่งเขียนแทนด้วย t_1Rt_2 หมายถึง การเกิดสัญญาณ t_1 ทำให้เกิดสัญญาณ t_2 และซิกแนลทรานสิชันกราฟจะเขียนแทนด้วย “⟶” ตัวอย่างเช่น L+⟶ D+ หมายถึง L+ จะทำให้เกิดสัญญาณ D+ เรียกว่าเกิดการ Firing จากรูปจะเห็นว่าซิกแนลทรานสิชันกราฟจะสามารถแสดงการเกิดการเปลี่ยนแปลงสัญญาณได้ทั้งรูปแบบลำดับ (Sequential) และพร้อมกัน (Concurrent) สำหรับรายละเอียดของซิกแนลทรานสิชันกราฟขอกล่าวอย่างละเอียดในบทที่ 3 ต่อไป

2.5 **คุณสมบัติการตอบรับ**

ในวงจรอสมวารแบบดีเลย์อินเซนซิทีฟ วงจรจะมีคุณสมบัติที่เรียกว่าคุณสมบัติการตอบรับ ซึ่งหมายถึงเมื่อเกิดการเปลี่ยนแปลงสัญญาณขึ้นที่สัญญาณภายในใด ๆ ของวงจรจะเป็นผลทำให้เกิดการเปลี่ยนแปลงสัญญาณต่อเนื่องไปยังสัญญาณเอาต์พุตหลักของสัญญาณนั้น ๆ ดังนั้นจึงทำให้วงจรประเภทนี้ไม่มีสัญญาณไม่ตอบรับ (Non-Acknowledgement)

สำหรับวงจรอสมวารแบบควอไซดีเลย์อินเซนซิทีฟนั้นก็มีคุณสมบัตินี้เช่นเดียวกันเพียงแต่จะแตกต่างกันเมื่อเกิดการเปลี่ยนแปลงของสัญญาณที่สัญญาณที่เกิดการกระจายของสัญญาณนั้นจะต้องเกิดการเปลี่ยนแปลงของสัญญาณต่อเนื่องไปยังสัญญาณเอาต์พุทของสัญญาณนั้น ๆ อย่างน้อย 1 สัญญาณ ซึ่งไม่จำเป็นที่จะต้องเกิดทุกเส้นสัญญาณที่เกิดการกระจายและเราจะเรียกสัญญาณที่ไม่เกิดการเปลี่ยนแปลงสัญญาณต่อเนื่องไปยังสัญญาณเอาต์พุตว่าสัญญาณไม่ตอบรับ

2.6 **สรุป**

ในบทนี้ได้กล่าวถึงทฤษฎีที่จำเป็นต้องทราบก่อนที่จะต้องทำการทดสอบวงจรอสมวาร เช่น โมเดลข้อผิดพร่องเพื่อที่จะได้ทราบถึงข้อผิดพร่องที่เกิดขึ้นในวงจร, โมเดลความหน่วงเพื่อที่จะได้

ทราบของพฤติกรรมการทำงานของวงจร, ข้อกำหนดของวงจรอสมวารและคุณสมบัติของวงจรอสมวารเพื่อที่จะใช้เป็นแนวทางในการทดสอบวงจรต่อไป

นอกจากนี้ยังได้กล่าวถึงการทดสอบวงจรสมวารที่ใช้อยู่ในปัจจุบัน และข้อจำกัดของวิธีการเหล่านี้ในการนำไปประยุกต์ใช้กับวงจรอสมวารด้วย

ซึ่งความรู้ที่ได้กล่าวไปทั้งหมดนี้ นั้นจะเป็นความรู้พื้นฐานที่ทำให้ผู้อ่านทำความเข้าใจกับงานวิจัย และจะถูกประยุกต์ใช้ในงานวิจัยที่จะนำเสนอในบทต่อ ๆ ไป

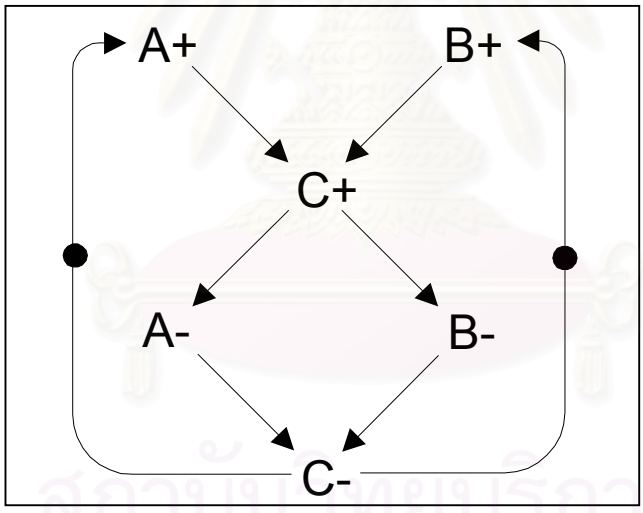
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 3
ซิกแนลทรานสิชันกราฟ

ในบทนี้จะกล่าวถึงรายละเอียดซิกแนลทรานสิชันกราฟ (Signal Transition Graph) [2] ซึ่งเป็นเครื่องมือที่ใช้เป็นข้อกำหนดของวงจรอสมวาร, state graph [2] ซึ่งเป็นเครื่องมือแสดงสถานะของ วงจรและใช้ในการสังเคราะห์วงจร, ขั้นตอนการสังเคราะห์วงจรจาก state graph [2] และ แผนภาพคล้ายซิกแนลทรานสิชัน (STG-like diagram) [10] ซึ่งเป็นเครื่องมือในการแสดงการทำงานของวงจร

3.1 ซิกแนลทรานซิกชันกราฟ

ซิกแนลทรานสิชันกราฟเป็นกราฟที่เครื่องมือที่ใช้ข้อกำหนดของวงจรอสมวาร ถูกพัฒนาโดย Tam-Anh Chu ซิกแนลทรานสิชันกราฟเป็นกราฟที่มีลักษณะเป็นกราฟแบบมีทิศทาง (direct graph) ซึ่งจะแสดงการเปลี่ยนแปลงของสัญญาณอินพุต, เอาต์พุตและสัญญาณภายในวงจรบางเส้นสัญญาณโดยมีตัวอย่างดังรูปที่ 3.1



รูปที่ 3.1 ตัวอย่างซิกแนลทรานสิชันกราฟ

ซิกแนลทรานสิชันกราฟประกอบด้วย

1) **เซตของสัญญาณ** (J) ประกอบไปด้วยเซตของสัญญาณอินพุต (J_I), และเซตของสัญญาณที่ไม่ใช่อินพุต (J_{NI}) ซึ่งประกอบด้วยเซตของสัญญาณเอาต์พุต (J_O) และเซตของสัญญาณภายในวงจร (Internal Signal) (J_N) จากรูปที่ 4 $J = \{ A,B,C \}$, $J_I = \{ A,B \}$, $J_{NI} = \{ C \}$

2) **เซตของ** Signal Transition (T) ถูกกำหนดโดย T = J $\mathbf{X}$ {+,-} หมายถึงทุก ๆ สัญญาณ j ∈ J จะสามารถมี Signal Transition ได้ 2 อย่างคือ {j+,j-} (j+ (rising) หมายถึง j มีการเปลี่ยนแปลงสัญญาณจาก 0 เป็น 1 และ j- (falling) หมายถึง j มีสัญญาณเปลี่ยนแปลงสัญญาณจาก 1 เป็น 0) และซิกแนลทรานสิชันกราฟจะแสดงการเปลี่ยนแปลงของสัญญาณอินพุตของวงจรด้วยการขีดเส้นใต้สัญญาณเพื่อแสดงให้เห็นความแตกต่างจากการเปลี่ยนแปลงของสัญญาณที่ไม่ใช่อินพุต

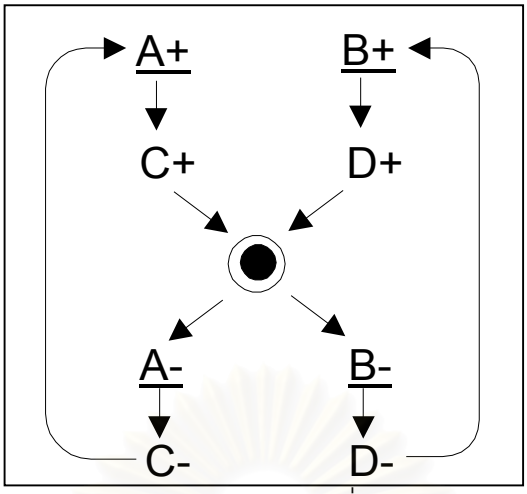
ซิกแนลทรานสิชันกราฟ จะใช้สัญลักษณ์ T/n เพื่อแสดงลำดับครั้งของการเปลี่ยนแปลงของสัญญาณ T ในซิกแนลทรานสิชันกราฟ ตัวอย่างเช่น A+/1 หมายถึงสัญญาณ A เกิดการเปลี่ยนแปลงสัญญาณครั้งที่ 1 และ A+/2 หมายถึงสัญญาณ A เกิดการเปลี่ยนแปลงสัญญาณครั้งที่ 2 เป็นต้น

3) **จุดดำ** (Marking) หรือที่เรียกว่า Token ถูกแทนด้วยสัญลักษณ์วงกลมทึบสีดำจะแสดงสถานะ (state) ของวงจร ณ เวลาใดเวลาหนึ่ง สำหรับทุก ๆ STG จะต้องมี Initial Marking เพื่อแสดงสถานะเริ่มต้นของวงจร

3.2 คุณสมบัติของซิกแนลทรานสิชันกราฟ

1. Causal relation เขียนแทนด้วยสัญลักษณ์ $t_1 R t_2$ หมายถึง การเปลี่ยนแปลงสัญญาณ t_1 ทำให้เกิดการเปลี่ยนแปลงสัญญาณ t_2 และในซิกแนลทรานสิชันกราฟจะเขียนแทนความสัมพันธ์นี้ด้วย " ⟶ " ตัวอย่างจากรูป 3.1 A+ ⟶ C+ หมายถึงการเปลี่ยนแปลงสัญญาณ A+ จะมีผลทำให้เกิดการเปลี่ยนแปลงสัญญาณ C+ และการเปลี่ยนแปลงสัญญาณ C+ จะเกิดขึ้นได้ก็ต่อเมื่อเกิดการเปลี่ยนแปลงของสัญญาณ A+ ด้วย

2. Input choice หรือเรียกอีกอย่างว่า Free choice จะใช้สัญลักษณ์ ◯ หรือเรียกว่า place ซึ่งหมายถึงเมื่อ marking อยู่ที่ place การเกิด Firing ของการเปลี่ยนแปลงสัญญาณที่เป็นเอาต์พุตของ place สามารถเลือกได้เพียง 1 สัญญาณเท่านั้น ซึ่งการเปลี่ยนแปลงที่เลือกนั้นจะไป Disable การเปลี่ยนแปลงของสัญญาณอื่น ๆ เพื่อไม่ให้เกิดการเปลี่ยนแปลงขึ้น จากรูป 3.2 เมื่อ marking อยู่ที่ place การเปลี่ยนแปลงสัญญาณที่สามารถเกิดได้คือ A- หรือ B- เพียงสัญญาณเดียวเท่านั้น



รูปที่ 3.2 ตัวอย่างซิกแนลทรานสิชันกราฟที่มีคุณสมบัติ input choice

นอกจากนี้ในขั้นตอนการสังเคราะห์วงจรจากซิกแนลทรานสิชันกราฟเพื่อที่จะทำให้วงจรสามารถทำงานได้ถูกต้องและไม่มี hazard นั้น ซิกแนลทรานสิชันกราฟจะต้องมีคุณสมบัติดังต่อไปนี้

Liveness

ซิกแนลทรานสิชันที่มีคุณสมบัติ Liveness หมายถึงทุก ๆ จุดที่ marking สามารถไปถึงได้จะต้องสามารถทำให้เกิดการเปลี่ยนแปลงของสัญญาณได้อีกเมื่อเกิดการเปลี่ยนแปลงของสัญญาณไปแล้ว ซิกแนลทรานสิชันกราฟจากรูป 3.3(a) ไม่มีคุณสมบัติ liveness เนื่องจากการเปลี่ยนแปลงของสัญญาณ “r+” จะเกิดขึ้นเพียงครั้งเดียวและจะไม่เกิดขึ้นอีกเลย

Safety

ซิกแนลทรานสิชันที่มีคุณสมบัติ Safety หมายถึงไม่มี place หรือ ลูกศรใด ๆ ในซิกแนลทรานสิชันกราฟสามารถมี marking ได้มากกว่า 1 marking ซิกแนลทรานสิชันกราฟจากรูป 3.3(b) ไม่มีคุณสมบัติ safety เนื่องจาก r+ $\longrightarrow$ y+ สามารถมี marking ได้มากกว่า 1 marking

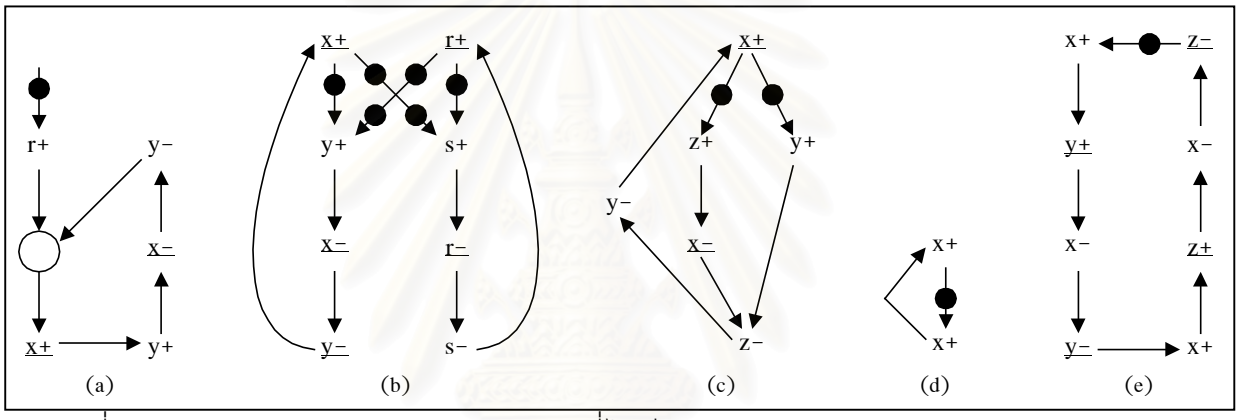
Persistency

ซิกแนลทรานสิชันที่มีคุณสมบัติ Persistency หมายถึงทุก ๆ ความสัมพันธ์ A* $\longrightarrow$ B* ในซิกแนลทรานสิชันกราฟ (t* หมายถึงการเปลี่ยนของสัญญาณ t+ หรือ t-) การเปลี่ยนแปลงของสัญญาณ B* จะต้องเกิดก่อนการเปลี่ยนแปลงของสัญญาณ A* ในทิศทางตรงข้าม ซิกแนลทรานสิชันกราฟจากรูป 3.3(c) ไม่มีคุณสมบัติ Persistency ในส่วนของ x+ $\longrightarrow$ y+ เนื่องจาก x- สามารถเกิดการเปลี่ยนแปลงของสัญญาณได้ก่อนที่ y+ จะเกิดขึ้น

คุณสมบัตินี้มีข้อยกเว้นสำหรับสัญญาณอินพุต เช่น A* ⟶ B* ถึงแม้ว่าสัญญาณ A* ในทิศทางตรงข้ามจะต้องเกิดก่อนการปลี่ยนแปลงของสัญญาณ B* ซิกแนลทรารสิชันนั้นก็ยังมีคุณสมบัติ Persistency

Consistency

ซิกแนลทรานสิชันที่มีคุณสมบัติ Consistency หมายถึงเมื่อมีการเปลี่ยนแปลงของทุก ๆ สัญญาณในซิกแนลทรานสิชันกราฟ การเปลี่ยนแปลงสัญญาณครั้งต่อไปของสัญญาณนั้น ๆ จะต้องเปลี่ยนแปลงในทิศทางตรงข้ามเท่านั้น ซิกแนลทรานสิชันกราฟจากรูป 3.3(d) ไม่มีคุณสมบัติ Consistency เนื่องจากมีการเกิดการเปลี่ยนแปลงของสัญญาณ x+ ต่อกันโดยที่ไม่มีการเปลี่ยนแปลงของสัญญาณ x- มาคั่น



รูปที่ 3.3 ตัวอย่างซิกแนลทรานสิชันกราฟที่ไม่มีคุณสมบัติ (a) liveness (b) Safety (c) Persistency (d) Consistency (e)Unique state assignment และ Single-cycle transitions

Unique state assignment

ซิกแนลทรานสิชันที่มีคุณสมบัติ Unique state assignment หมายถึงทุก ๆ จุดในซิกแนลทรานสิชันกราฟจะต้องมีค่าของสัญญาณทุกสัญญาณเพียงค่าเดียวและค่านั้น ๆ จะต้องไม่มีค่าซ้ำกับจุดอื่น ๆ ในซิกแนลทรานสิชันกราฟ ซิกแนลทรานสิชันกราฟจากรูป 3.3(e) ไม่มีคุณสมบัติ Unique state assignment เนื่องจากค่าของสัญญาณมีค่าซ้ำกันในสถานะเริ่มต้นและเมื่อ marking อยู่ที่จุด y- ⟶ x+

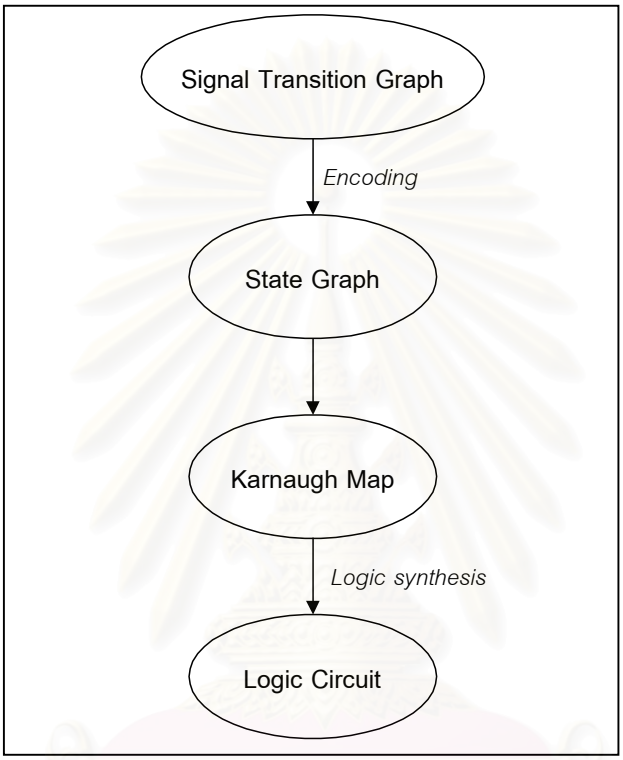
Single-cycle transitions

ซิกแนลทรานสิชันที่มีคุณสมบัติ Single-cycle transition หมายถึงการเปลี่ยนแปลงของแต่ละสัญญาณใน cycle ใดของซิกแนลทรานสิชันกราฟนั้น จะต้องเกิดการเปลี่ยนแปลงของ

สัญญาณนั้น ๆ 2 ครั้งคือขาขึ้น (+) และขาลง (-) อย่างละครั้งเท่านั้น ซิกแนลทรานสิชันกราฟจากรูป 3.3(e) ไม่มีคุณสมบัติ single-cycle transitions เนื่องจากเกิดการเปลี่ยนแปลงของสัญญาณ x+ และ x- มากกว่า 1 ครั้งใน cycle ของซิกแนลทรานสิชันกราฟ

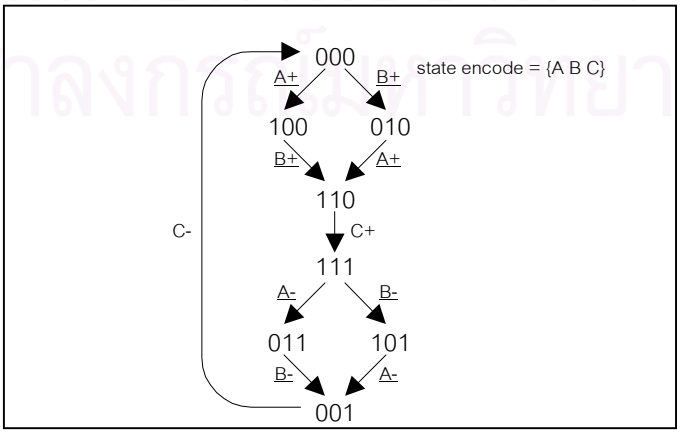
3.3 ขั้นตอนการสังเคราะห์วงจรจากซิกแนลทรานสิชันกราฟ

เราสามารถนำซิกแนลทรานสิชันไปสังเคราะห์วงจรได้ตามขั้นตอนดังรูป 3.4



รูปที่ 3.4 ขั้นตอนการสังเคราะห์วงจรจากซิกแนลทรานสิชันกราฟ

จากรูปที่ 3.4 จะเห็นว่าการสังเคราะห์วงจรจะต้องมี state graph จากซิกแนลทรานสิชันกราฟโดยผ่านขบวนการ encoding หรือ state assignment

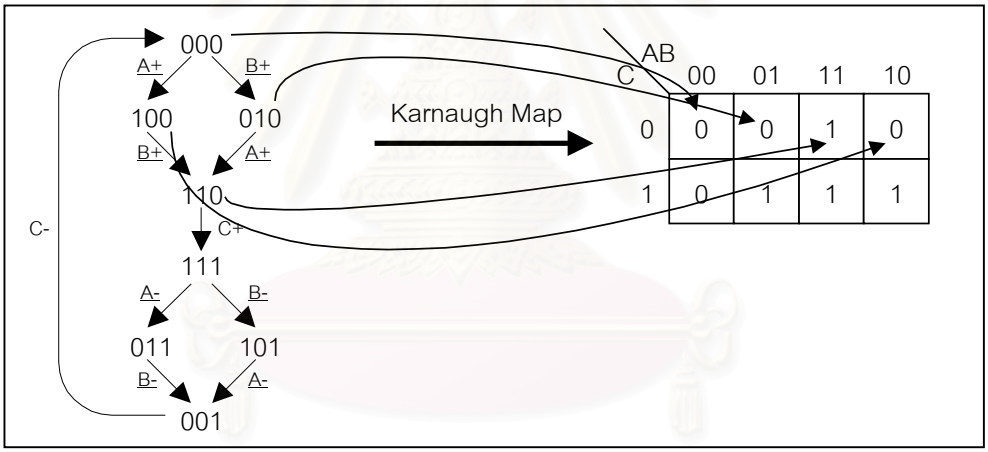


รูปที่ 3.5 State Graph ของซิกแนลทรานสิชันกราฟรูป 3.1

State graph (SG) เป็นกราฟที่จะแสดงสถานะต่าง ๆ ของกราฟซึ่งจะถูกแทนที่ด้วยเวกเตอร์ของเลขฐานสอง (binary vector) ซึ่งจะแสดงค่าของสัญญาณทั้งหมดในซิกแนลทรานสิชันกราฟ และจะแสดงถึงความสัมพันธ์ระหว่างสถานะต่าง ๆ ที่ไม่สามารถดูได้จากซิกแนลทรานสิชันกราฟ ดังตัวอย่างในรูป 3.5

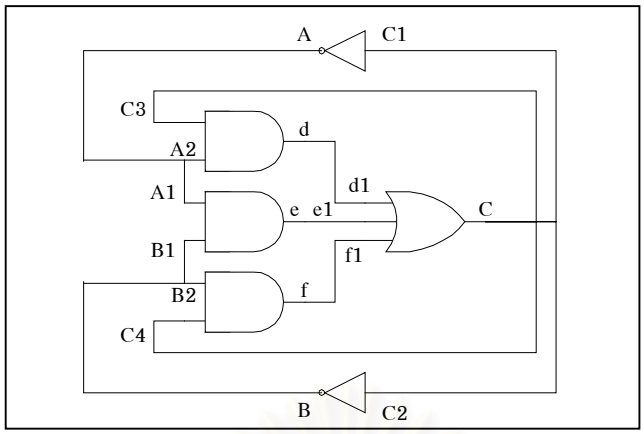
ในแต่ละสถานะใน state graph ยังแบ่งเป็นสถานะคงที่ซึ่งหมายถึงสถานะที่ไม่สามารถเปลี่ยนสถานะได้ถ้าไม่มีการเปลี่ยนแปลงของสัญญาณอินพุต จากรูป 3.5 สถานะที่เป็นสถานะคงที่คือ {000},{100},{010},{011},{101} และ {111} และสถานะไม่คงที่ซึ่งหมายถึงสถานะที่สามารถเปลี่ยนแปลงสถานะไปยังสถานะต่อไปได้โดยไม่ต้องรอการเปลี่ยนแปลงของสัญญาณอินพุต จากรูป 3.5 สถานะที่เป็นสถานะไม่คงที่คือ {110} และ {001} โดยในการทำงานของวงจรจะเริ่มต้นการทำงานที่สถานะเริ่มต้นซึ่งถือว่าเป็นสถานะคงที่เสมอ จากรูป 3.5 สถานะเริ่มต้นคือ {000}

ขั้นตอนต่อไปคือการนำ state graph มาสร้างเป็น Karnaugh Map เพื่อนำไปสังเคราะห์วงจรต่อไป โดยมีตัวอย่างการสร้างดังรูป 3.6



รูปที่ 3.6 ขั้นตอนการสร้าง Karnaugh Map จาก State Graph

จากรูป 3.6 พิจารณา 4 state ด้านบนจะพบว่าสถานะ {000},{100},{010} ค่าของ C จะถูกกำหนดเป็น 0 ใน Karnaugh Map เนื่องจาก C ยังคงมีค่าเป็นศูนย์ในสถานะนี้ ส่วนสถานะ {110} ค่าของ C จะถูกกำหนดให้เป็น 1 ใน Karnaugh Map เนื่องจาก การเปลี่ยนแปลงสัญญาณ C+ สามารถที่จะเกิดขึ้นได้ ในทำนองเดียวกันเมื่อพิจารณา 4 สถานะด้านล่างก็จะได้ Karnaugh Map ดังรูป 3.6 เมื่อได้ Karnaugh Map มาแล้วก็สามารถทำเป็นวงจรได้คือ C = AB + AC +BC ดังรูป 3.7



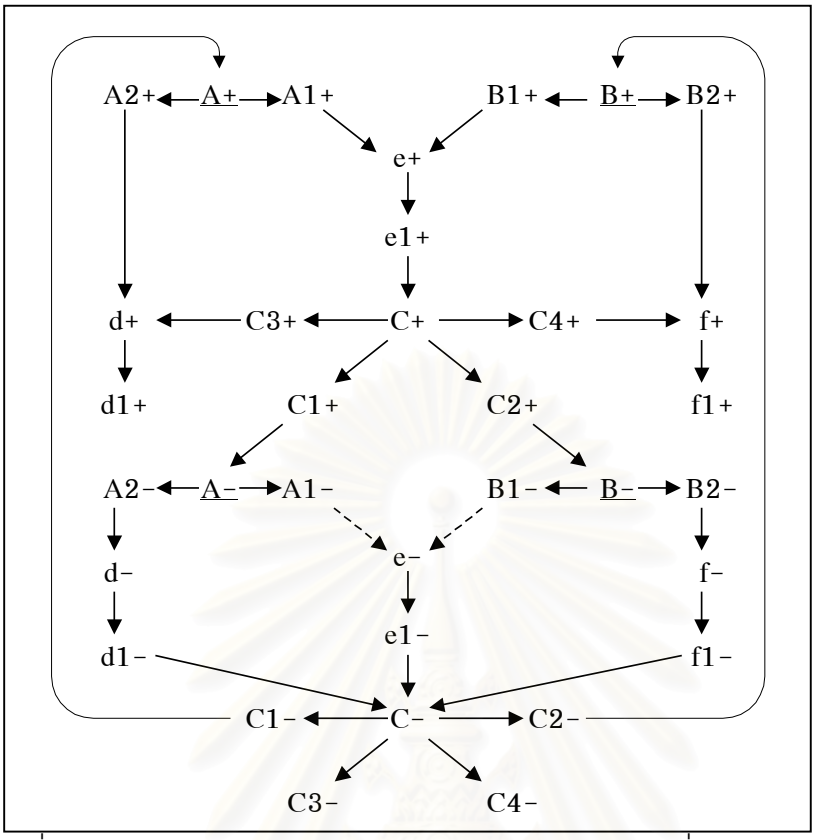
รูปที่ 3.7 วงจรที่สังเคราะห์ได้จาก state graph

3.4 แผนภาพคล้ายซิกแนลทรานสิชันกราฟ (STG-like diagram)

เมื่อเราสังเคราะห์วงจรจากซิกแนลทรานสิชันกราฟแล้ว เราจะพบว่าซิกแนลทรานสิชันกราฟจะไม่สามารถแสดงรายละเอียดการเปลี่ยนแปลงของสัญญาณทั้งหมดที่เกิดขึ้นในวงจรเช่น OR causality และการเปลี่ยนของสัญญาณที่ไม่ได้ระบุในซิกแนลทรานสิชันกราฟ ดังนั้นจึงต้องมีการสร้างแผนภาพคล้ายซิกแนลทรานสิชันกราฟเพื่อที่จะแสดงรายละเอียดของการเกิดการเปลี่ยนแปลงของสัญญาณทั้งหมด แผนภาพคล้ายซิกแนลทรานสิชันกราฟจะแสดงการเปลี่ยนแปลงของสัญญาณทั้งหมดในวงจรและยังมีคุณสมบัติ causal relation เช่นเดียวกับซิกแนลทรานสิชันกราฟ และยังใช้เส้นปะแสดงความสัมพันธ์แบบ OR-causality ดังตัวอย่างในรูป 3.8

จากรูปที่ 3.8 จะพบว่าแผนภาพคล้ายซิกแนลทรานสิชันกราฟจะสามารถแสดงรายละเอียดการเปลี่ยนแปลงของสัญญาณทั้งหมดในวงจรได้ ซึ่งจำเป็นอย่างยิ่งในขั้นตอนการทดสอบวงจร เนื่องจากรายละเอียดของซิกแนลทรานสิชันกราฟมีจำกัด

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 3.8 แผนภาพคล้ายซิกแนลทรานสิชันกราฟของวงจรรูปที่ 3.7

3.5 สรุป

ในบทนี้ได้กล่าวถึงรายละเอียดคุณสมบัติที่จำเป็นของซิกแนลทรานสิชันกราฟที่สามารถนำไปสังเคราะห์วงจรได้, ขั้นตอนการสังเคราะห์วงจรเพื่อให้รู้ถึงความสัมพันธ์ระหว่างซิกแนลทรานสิชันกราฟกับวงจรที่สังเคราะห์ได้ รวมทั้งยังได้กล่าวถึงแผนภาพคล้ายซิกแนลทรานสิชันกราฟที่ใช้ในการแสดงการทำงานของวงจรโดยละเอียด ซึ่งมีความจำเป็นเป็นอย่างมากในขั้นตอนการทดสอบวงจรอสมวาร

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 4
การทดสอบวงจรอสมวาร

ในบทนี้จะเป็นการนำเสนอวิธีการทดสอบวงจรอสมวารแบบควอไซดีเลย์อินเซนซิฟ ซึ่งเป็นวิธีที่ประยุกต์มาจากวิธีการทดสอบวงจรอสมวารที่ใช้โมเดลความหน่วงชนิดดีเลย์อินเซนซิทีฟ [4]และฮัฟแมนโดยใช้ Stable State Graph (SSG) [11] และ Extend Stable State Graph [11]

4.1 เงื่อนไขในการทดสอบวงจรอสมวาร

เนื่องจากวงจรแบบอสมวารมีข้อแตกต่างจากวงจรแบบสมวาร ดังนั้นวิธีการทดสอบวงจรชนิดนี้จึงต้องมีสิ่งพิจารณาเพิ่มเติมดังนี้

1. วงจรอสมวารที่มีโมเดลความหน่วงแบบควอไซดีเลย์อินเซนซิทีฟ จะสามารถทำงานได้ตามลำดับของซิกแนลทรานสิชันกราฟ ดังนั้นในขณะทดสอบวงจรเราจึงสนใจเส้นทางที่เป็นไปตามลำดับของซิกแนลทรานสิชันกราฟเท่านั้น
2. วงจรแบบอสมวารจะไม่มีสัญญาณนาฬิกาในการควบคุมการทำงาน ดังนั้นในขณะทดสอบเราจะต้องรอให้วงจรอยู่ในสถานะคงที่เสียก่อน แล้วจึงทำให้เกิดการเปลี่ยนแปลงสัญญาณอินพุตได้ จึงทำให้เราทดสอบได้เพียงบางเส้นทางของ state graph เท่านั้น
3. การหาสถานะวิกฤต (Excitation state) ในวงจรอสมวารที่มีโมเดลความหน่วงแบบควอไซดีเลย์อินเซนซิทีฟ เราไม่สามารถมองเป็นวงจรย่อยที่เกี่ยวข้องกับข้อผิดพลาดได้ ดังนั้นเราจึงต้องพิจารณาเกตที่เกี่ยวข้องกับข้อผิดพลาด
4. ต้องมีการจำลองข้อผิดพร่อง (Fault Simulation) เพื่อเปรียบเทียบและวิเคราะห์ว่าข้อผิดพร่องนั้น ๆ มีผลกระทบกับการทำงานของวงจรอย่างไร

จากเงื่อนไขดังกล่าวทำให้เราจำเป็นจะต้องหาสถานะเริ่มต้นและสถานะคงที่ของวงจร ซึ่งเราสามารถหาได้จากซิกแนลทรานสิชันกราฟ

4.2 การหาสถานะเริ่มต้นของวงจร

การหาสถานะเริ่มต้นเป็นขั้นตอนเริ่มต้นที่จะทำการทดสอบ ซึ่งมีขั้นตอนดังนี้

Algorithm 1 Find_ InitialState

Input STG

Output Initial state

Si is set of signal in STG

Step 1.1 Set_unknown(Si)

Step 1.2 Do while (Some of Si is unknown value)

Step 1.3 move_marking(STG)

Step 1.4 if Si+ and Si is unknown then

Si = 0

Else

Si=1

Step 1.5 Return (Si)

จากขั้นตอนที่ 1 (algorithm 1) เราสามารถอธิบายได้ดังนี้

Step 1.1 เป็นขั้นตอนในการกำหนดค่าเริ่มต้นให้กับสัญญาณต่าง ๆ ในวงจร โดยกำหนดให้เป็นค่าที่ไม่ทราบค่า

Step 1.2 เป็นขั้นตอนการสร้างลูปสำหรับหาค่าสถานะเริ่มต้นของวงจรโดยจะทำจนกระทั่งค่าของสัญญาณทุกสัญญาณไม่ใช่ค่าที่ไม่ทราบค่า

Step 1.3 เป็นขั้นตอนการเลื่อน marking บนซิกแนลทรานสิชันกราฟ โดยใช้คุณสมบัติ causal relation

Step 1.4 เป็นขั้นตอนการตรวจสอบการเปลี่ยนแปลงสัญญาณที่เกิดขึ้นเมื่อมีการเลื่อน marking โดยถ้าสัญญาณใดที่มีค่าเป็นที่ไม่ทราบค่าเกิดการเปลี่ยนแปลงสัญญาณแบบขาขึ้น (+) ให้กำหนดค่าสัญญาณนั้น ๆ เป็น 0 และสัญญาณใดที่มีค่าเป็ที่ไม่ทราบค่า เกิดการเปลี่ยนแปลงสัญญาณแบบขาลง (-) ให้กำหนดค่าสัญญาณนั้น ๆ เป็น 1

Step 1.5 เมื่อทำจนเสร็จสิ้นแล้ว ค่าที่ได้เป็นสถานะเริ่มต้นของวงจร

จากขั้นตอนนี้ทำให้เราสามารถหาค่าสถานะเริ่มต้นของวงจรเพื่อที่จะนำไปกำหนดเป็นค่าเริ่มต้นในขั้นตอนอื่น ๆ ต่อไป

4.3 การหาค่าสถานะคงที่ของวงจรจากซิกแนลทรานสิชันกราฟ

จากเงื่อนไขในการทดสอบวงจรอสมวารแบบควอไซดีเลย์อินเซนซิทีฟ พบว่าในขั้นตอนการทดสอบวงจรจะต้องรอให้วงจรอยู่ในสถานะคงที่เสียก่อนจึงสามารถทดสอบวงจรได้ ดังนั้นเราจึงต้องหาสถานะคงที่ทั้งหมดของวงจรได้โดยขั้นตอนต่อไปนี้

Algorithm 2 Find_ StableState

Input STG

Output Set of stable state

Si is set of signal in STG

Ii is set of input signal in STG

Stable_state is set of stable state

Step 2.1 InitialState = Find_ InitialState(STG)

Step 2.2 Set_signal(Initial_State)

Step 2.3 n = count_marking(STG,Si)

Step 2.4 if n = 1 then

Step 2.5 Stable_state = Fire_input_signal(STG,Si)

Else

Step 2.6 Stable_state = Fire_each_input_signal(STG,Si)

Step 2.7 repeat step 2.2 until found initial state in Stable_state

Step 2.8 return Stable_state

จากขั้นตอนที่ 2 (algorithm 2) เราสามารถอธิบายได้ดังนี้

Step 2.1 หาค่าสถานะเริ่มต้นของวงจร

Step 2.2 กำหนดให้วงจรอยู่ในสถานะเริ่มต้น

Step 2.3 นับจำนวน marking ที่รอการเปลี่ยนแปลงสัญญาณอินพุตในซิกแนลทรานสิชันกราฟในสถานะปัจจุบัน

Step 2.4 ตรวจสอบจำนวน marking

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

Step 2.5 ถ้ามี marking เพียง 1 ตัว ให้ทำการเปลี่ยนแปลงสัญญาณอินพุต โดยให้กำหนดค่าของสัญญาณเป็น 1 เมื่อพบการเปลี่ยนแปลงสัญญาณแบบขาขึ้น(+) และ กำหนดค่าของสัญญาณเป็น 0 เมื่อพบการเปลี่ยนแปลงสัญญาณแบบขาลง (-) ทำการเปลี่ยนแปลงสัญญาณตามซิกแนลทรานสิชันกราฟจนกระทั่ง marking พบการเปลี่ยนแปลงของสัญญาณอินพุตหรือไม่สามารถทำให้เกิดการเปลี่ยนแปลงสัญญาณตามคุณสมบัติ causal relation ได้ ค่าของสัญญาณที่ได้จะเป็นค่าสถานะเริ่มต้น

ในกรณีที่ marking อยู่ที่ place ให้ทำการเปลี่ยนแปลงสัญญาณอินพุตทีละกรณีจนครบทุกกรณี

Step 2.6 ถ้ามี marking มากกว่า 1 ตัวในซิกแนลทรานสิชันกราฟในสถานะปัจจุบัน

1. ให้ทำการเปลี่ยนแปลงสัญญาณอินพุตทีละตัวไปในซิกแนลทรานสิชันกราฟ โดยให้กำหนดค่าของสัญญาณเป็น 1 เมื่อพบการเปลี่ยนแปลงสัญญาณแบบขาขึ้น(+) และ กำหนดค่าของสัญญาณเป็น 0 เมื่อพบการเปลี่ยนแปลงสัญญาณแบบขาลง (-) ทำการเปลี่ยนแปลงสัญญาณตามซิกแนลทรานสิชันกราฟจนกระทั่ง marking พบการเปลี่ยนแปลงของสัญญาณอินพุตหรือไม่สามารถทำให้เกิดการเปลี่ยนแปลงสัญญาณตามคุณสมบัติ causal relation ได้ ค่าของสัญญาณที่ได้จะเป็นสถานะคงที่
2. ถ้า marking ที่เปลี่ยนแปลงตามข้อ 1 พบการเปลี่ยนแปลงที่เป็นสัญญาณอินพุต ให้ทำการเปลี่ยนแปลงสัญญาณอินพุตของ marking ตัวนั้น แล้วทำตามเงื่อนไขข้อ 5.1 จนกระทั่งไม่สามารถทำให้เกิดการเปลี่ยนแปลงสัญญาณตามคุณสมบัติ causal relation ได้
3. ย้อนกลับไปสถานะเดิมแล้วเลือกการเปลี่ยนแปลงของ marking ตัวต่อไป ทำตามข้อ 1 และ 2 จนครบทุกกรณี

Step 2.7 กลับไปทำ Step 2.3 จนกระทั่งพบสถานะเริ่มต้นในกลุ่มของสถานะคงที่ที่หาได้

Step 2.8 กลุ่มของสถานะคงที่ที่หาได้จะเป็นสถานะคงที่ทั้งหมดของวงจร

4.4 Stable State Graph (SSG)

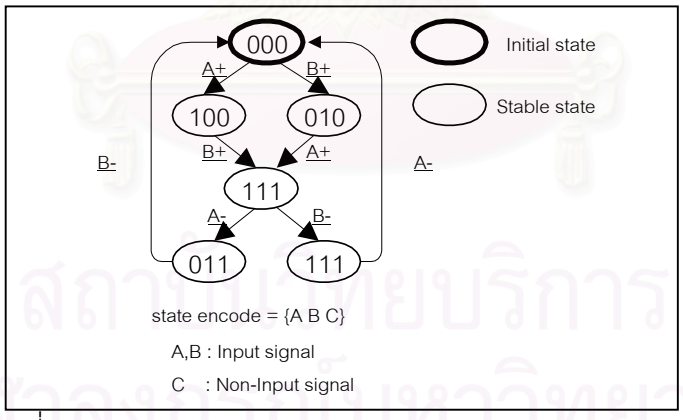
เมื่อเราหาสถานะคงที่ทั้งหมดของวงจรออกมาแล้วเราสามารถสร้างความสัมพันธ์ของสถานะคงที่ต่าง ๆ ได้โดยใช้ SSG ซึ่งสามารถสร้างได้จากซิกแนลทรานสิชันกราฟ ดังรูปที่ 4.1 โดยมีหลักการดังนี้

ให้ J = {J_1, J_2,..., J_n} และ J_I = {J_{I1}, J_{I2},..., J_{Il}}, $J_I \subseteq J$ โดยที่เป็นเซตของสัญญาณและเซตของสัญญาณอินพุตของซิกแนลทรานสิชันกราฟ

ส่วนประกอบของ SSG

1. เซตของสถานะคงที่ (S) ประกอบไปด้วยเซตของสถานะคงที่ ซึ่งเป็นเวกเตอร์ของเลขฐานสอง ซึ่งแทนค่าของสัญญาณต่าง ๆ ใน สถานะ s โดย s_0 เป็นสถานะเริ่มต้นของวงจร
2. เซตของการเปลี่ยนแปลงของสัญญาณอินพุต (T) ถูกกำหนดโดย T = J_I X {+,-} หมายถึงทุก ๆ สัญญาณอินพุต j $\in$ J_I จะสามารถเกิดการเปลี่ยนแปลงสัญญาณได้ 2 อย่างคือ {j+,j-} (j+ (ขาขึ้น) หมายถึง j มีการเปลี่ยนแปลงสัญญาณจาก 0 เป็น 1 และ j- (ขาลง) หมายถึง j มีสัญญาณเปลี่ยนแปลงสัญญาณจาก 1 เป็น 0)

ความสัมพันธ์ระหว่างสถานะคงที่สามารถเขียนได้โดย s_jTs_k หมายถึงถ้าวงจรอยู่ในสถานะ s_j ทำให้เกิดการเปลี่ยนแปลงสัญญาณอินพุต T แล้ววงจรจะเปลี่ยนไปอยู่ในสถานะ s_k และใน SSG จะเขียนแทนความสัมพันธ์นี้ด้วยการเปลี่ยนแปลงของสัญญาณอินพุต ตัวอย่างเช่น <s_j,a+,s_k>หมายถึงเมื่อมีการเปลี่ยนแปลงสัญญาณอินพุต a+ แล้ววงจรจะเปลี่ยนจากสถานะ s_j ไปยัง สถานะ s_k หรือเขียนแทนด้วย $s_j \xrightarrow{a+} s_k$



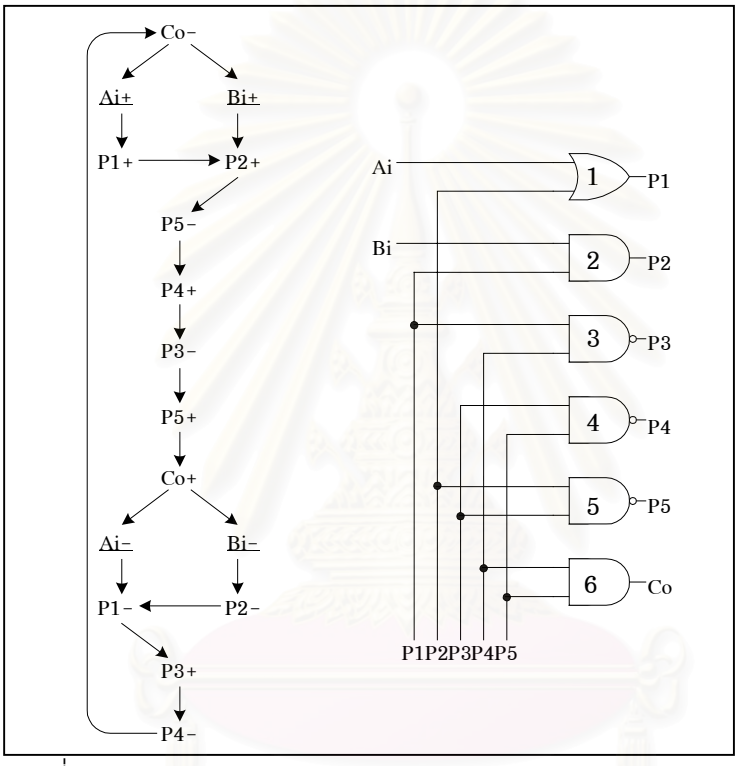
รูปที่ 4.1 ตัวอย่าง Stable state graph ของ C-element

4.5 การทดสอบวงจรอสมวาร

ค่าของสถานะใน SSG ประกอบไปด้วยค่าของสัญญาณอินพุตและสัญญาณที่ไม่ใช่สัญญาณอินพุต ถ้าการทำงานปกติของวงจรทำให้เกิดการเปลี่ยนแปลงสถานะ <s1,t,s2> ใน SSG แล้วเมื่อเกิดข้อผิดพร่องในวงจรจะทำให้วงจรไปหยุดในสถานะคงที่ s3 โดยที่ s2 $\neq$ s3 ถ้าข้อ

ผิดพร่องนี้สามารถทดสอบได้ s2 และ s3 จะต้องมีค่าของสัญญาณอินพุตเหมือนกันและค่าของสัญญาณเอาต์พุตแตกต่างกัน

จากรูปที่ 4.2 การทำงานปกติของวงจรจะต้องเกิดการเปลี่ยนแปลงของสถานะ { Ai Bi P1 P2 P3 P4 P5 C0} ดังนี้ <s1{0 1 0 0 1 0 1 0},Ai+,s2{1 1 1 1 0 1 1 1}> เมื่อเกิดข้อผิดพร่องแบบ stuck-at 0 ที่สัญญาณ Ai การทำงานของวงจรก็เปลี่ยนไปดังนี้ <s1{0 1 0 0 1 0 1 0}, Ai+,s3{1 1 0 0 1 0 1 0}> ซึ่งจะเห็นได้ว่าค่าของสัญญาณเอาต์พุตของ s2 และ s3 มีค่าต่างกันดังนั้นเมื่อเกิดข้อผิดพร่อง stuck-at 0 ที่สัญญาณ Ai จะสามารถทดสอบได้



รูปที่ 4.2 วงจรและซิกแนลทรานสิชันของวงจร C-element

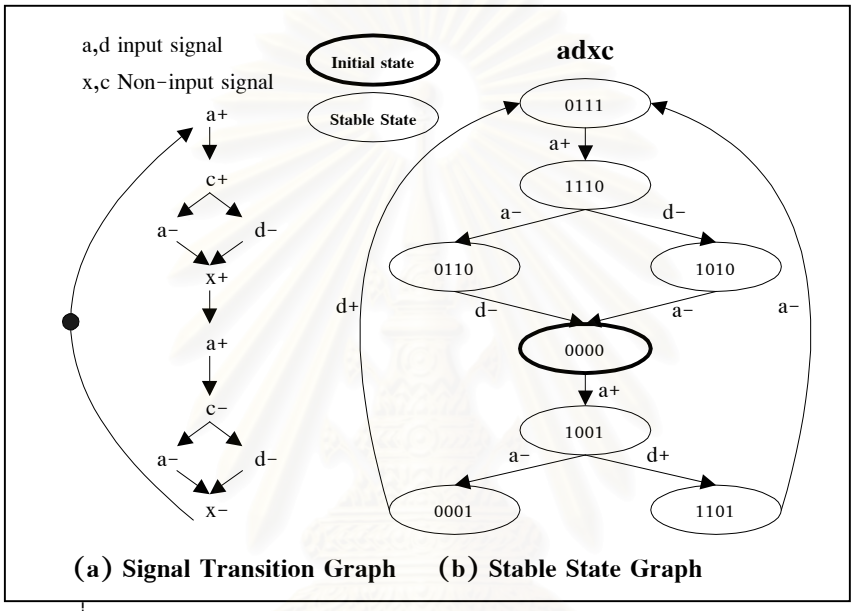
จากรูป 4.3 เมื่อเกิดข้อผิดพร่องแบบ stuck-at 0 ที่สัญญาณ Lr เราไม่สามารถหาการเปลี่ยนแปลงสถานะที่ทำให้วงจรทดสอบได้แต่ถ้าวงจรอยู่ในสถานะ {La Za Da X Y Zr Dr Lr} ที่ s0 {0 0 0 0 0 0 0 1} แล้วเราสามารถทำให้เกิดการเปลี่ยนแปลงสัญญาณตามลำดับของซิกแนลทรานสิชันกราฟได้เพียงสัญญาณเดียวคือ La+ ซึ่งเมื่อเปลี่ยนแปลงสัญญาณนี้จะทำให้เกิดการทำงาน <s0{0 0 0 0 0 0 0 1},La+,s1{1 0 0 0 1 0 1 0}> ซึ่งเมื่อเกิดข้อผิดพร่อง stuck-at 0 ที่สัญญาณ Lr แล้วการทำงานก็จะได้เป็น <s0{0 0 0 0 0 0 0 1},La+,s2{1 0 0 0 1 0 1 0}> ซึ่งมีค่าเอาต์พุตเหมือนกันทำให้เราไม่สามารถทดสอบได้

จากตัวอย่างข้างต้นจะเห็นว่าเราไม่สามารถทำการเปลี่ยนแปลงสถานะที่ทำให้ $s2 \neq s3$ ได้ ดังนั้นจึงมีการปรับปรุงวิธีการโดยการเพิ่มเซตของการเปลี่ยนแปลงสัญญาณอินพุต โดยนำการ

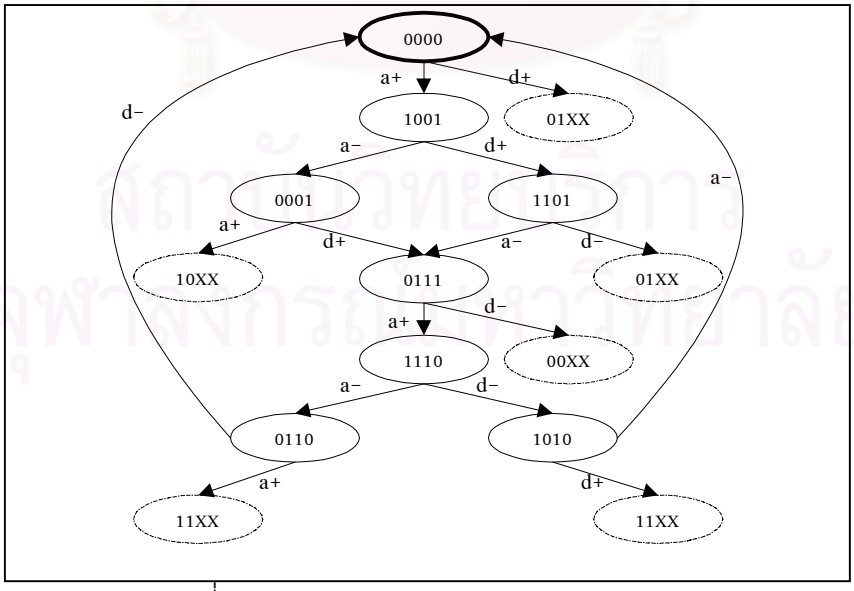
เปลี่ยนแปลงของสัญญาณอินพุตที่ไม่ได้เกิดตามลำดับของซิกแนลทรานสิชันกราฟทำให้ได้กราฟใหม่ที่เรียกว่า ESSG (Extend Stable State Graph)

4.6 Extend Stable State Graph (ESSG)

เป็นกราฟที่แสดงความสัมพันธ์เช่นเดียวกับ SSG แต่ได้เพิ่มการเปลี่ยนแปลงของสัญญาณอินพุตที่ไม่ได้เกิดตามลำดับของซิกแนลทรานสิชันกราฟเข้าไปในเซตของการเปลี่ยนแปลงสัญญาณอินพุต ซึ่งทำให้เซตของสถานะคงที่มีขนาดเพิ่มขึ้นด้วย



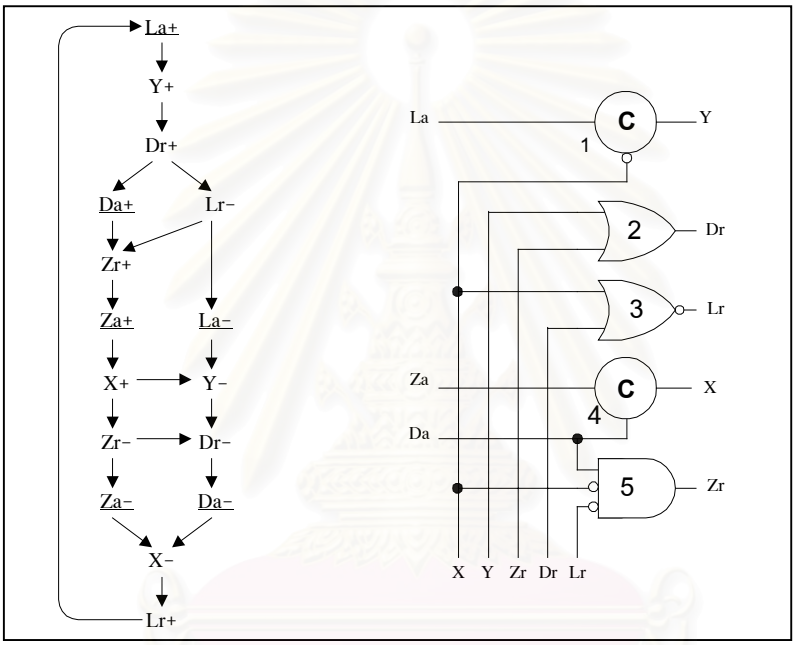
รูปที่ 4.3 ตัวอย่าง ซิกแนลทรานสิชันกราฟและStable State Graph



รูปที่ 4.4 ตัวอย่าง Extend Stable State Graph

จากรูปที่ 4.4 จะเป็น ESSG ที่ได้จาก SSG รูปที่ 4.3 และได้ใช้วงรีที่เป็นเส้นปะแสดงสถานะคงที่ที่เพิ่มขึ้นจากการเพิ่มเซตของการเปลี่ยนแปลงของสัญญาณอินพุต

ดังนั้นเมื่อเกิดข้อผิดพร่องแบบ stuck-at 0 ที่สัญญาณ Lr เราสามารถใช้ ESSG ทำให้ทดสอบได้โดยทำให้เกิดการเปลี่ยนแปลงสัญญาณ Da+ ซึ่งไม่เป็นตามลำดับของซิกแนลทรานสิชันกราฟจะทำให้เกิดการทำงานของวงจรดังนี้ <s0{0 0 0 0 0 0 0 1},Da+,s1{0 0 1 0 0 0 0 1}> แต่ถ้าวงจรเกิดข้อผิดพร่อง stuck-at 0 ที่สัญญาณ Lr การทำงานก็จะเปลี่ยนไปเป็น <s0{0 0 0 0 0 0 0 1},Da+,s2{0 0 1 0 0 1 1 0} ซึ่งจะเห็นว่าค่าของสัญญาณเอาต์พุตของ s1 และ s2 มีค่าต่างกันดังนั้นเมื่อเกิดข้อผิดพร่อง stuck-at 0 ที่สัญญาณ Ai จะสามารถทดสอบได้



รูปที่ 4.5 วงจรและซิกแนลทรานสิชันของวงจร Chu133

เมื่อเกิดข้อผิดพร่องชนิด stuck-at ในวงจรอสมวารสามารถเกิดเหตุการณ์ได้ 2 อย่างคือข้อผิดพร่องจะไปป้องกันไม่ให้การเกิดการเปลี่ยนแปลงสัญญาณตามลำดับของซิกแนลทรานสิชันกราฟทำให้วงจรไปหยุดในสถานะไม่คงที่ซึ่งจะเรียกเหตุการณ์นี้ว่า Inhibit Transition และข้อผิดพร่องจะไปเป็นสาเหตุให้เกิดการเปลี่ยนแปลงที่ไม่ได้กำหนดในซิกแนลทรานสิชันกราฟ ซึ่งจะเรียกเหตุการณ์นี้ว่า Premature Transition การเกิด Premature Transition อาจเป็นสาเหตุทำให้เกิด Hazard ซึ่งจะทำให้ Fault coverage ที่ได้น้อยลง ดังนั้นเราจึงจำเป็นจะต้องแสดงพฤติกรรมของวงจรขณะที่ทำการเปลี่ยนแปลงสถานะแล้วเกิด Hazard โดยละเอียดโดยใช้แผนภาพคล้ายซิกแนลทรานซิชันกราฟแสดง เพื่อนำมาวิเคราะห์และเติมจุดทดสอบเพื่อให้วงจรทดสอบได้

จากรูปที่ 4.2 การทำงานปกติของวงจรจะต้องเกิดการเปลี่ยนแปลงของสถานะ { Ai Bi P1 P2 P3 P4 P5 C0} ดังนี้ <s1{0 1 0 0 1 0 1 0},Ai+,s2{1 1 1 1 0 1 1 1}> เมื่อเกิดข้อผิดพร่อง

แบบ stuck-at 1 ที่สัญญาณ P3 ทีเกิดการกระจายของสัญญาณเส้นที่ 1 (P31) การทำงานของวงจรเกิด hazard ขึ้น ซึ่งทำให้เราไม่สามารถทราบถึงสถานะต่อไปได้ทำให้ไม่สามารถทดสอบข้อผิดพร่องนี้ได้

ในการทดสอบข้อผิดพร่องที่ไม่สามารถทดสอบได้นั้น เราจำเป็นที่จะต้องเพิ่มจุดทดสอบในวงจรเพื่อให้ข้อผิดพร่องนั้นทดสอบได้ เพื่อที่จะให้จุดทดสอบที่เพิ่มในวงจรมีจำนวนน้อย ดังนั้นจึงจำเป็นที่จะต้องรู้ถึงพฤติกรรมของวงจรเมื่อวงจรเกิดข้อผิดพร่องโดยใช้แผนภาพคล้ายซิกแนลทรานสิชันกราฟ และนำมาวิเคราะห์พฤติกรรมเพื่อให้ทราบถึงสัญญาณที่จำเป็นต้องใช้จุดทดสอบ

4.7 การหาสถานะวิกฤติ

เพื่อที่จะได้รู้ถึงพฤติกรรมของวงจรเมื่อเกิดข้อผิดพร่องเราจึงต้องแสดงพฤติกรรมของวงจรที่เกิดข้อผิดพร่องโดยการนำแผนภาพคล้ายซิกแนลทรานสิชันกราฟมาแสดงการเปลี่ยนแปลงของสถานะคงที่ใน SSG ในขั้นตอนนี้จะต้องหาค่าของสัญญาณที่ทำให้ข้อผิดพร่องทำงานซึ่งเราเรียกว่าสถานะวิกฤติเพื่อนำมาตรวจสอบว่าอยู่ในช่วงการเปลี่ยนแปลงสถานะใด เราสามารถหาค่าสถานะวิกฤตได้โดยขั้นตอนต่อไปนี้

1. ให้กำหนดค่าของสัญญาณที่เกิดข้อผิดพร่องเป็นค่าตรงข้ามกับค่าที่เป็น เช่น ถ้าสัญญาณ a เกิด stuck-at 1 ให้กำหนดสัญญาณ a เป็น 0 และถ้าสัญญาณ a เกิด stuck-at 0 ให้กำหนดสัญญาณ a เป็น 1
2. ให้กำหนดค่าสัญญาณที่เป็นสัญญาณอินพุตร่วมกับสัญญาณที่เกิดข้อผิดพร่องโดยมีเงื่อนไขดังนี้

    2.1 เกต And และ Nand

    กำหนดค่าสัญญาณเป็น 1

    2.2 เกต Or และ Nor

    กำหนดค่าสัญญาณเป็น 0

    2.3 เกต C-element และ X-Or

    ไม่จำเป็นต้องกำหนดเพราะข้อผิดพร่องจะทำงานเมื่อมีการเปลี่ยนแปลงสัญญาณที่เกิดข้อผิดพร่องหรือสัญญาณที่เป็นสัญญาณอินพุตร่วม

จากนั้นก็นำค่าสถานะวิกฤติมาตรวจสอบว่าอยู่ที่ช่วงใดใน SSG แล้วนำช่วงที่เกิดสัญญาณวิกฤติไปสร้างเป็นแผนภาพคล้ายซิกแนลทรานสิชันกราฟต่อไป

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

4.8 การจำลองข้อผิดพร่องโดยใช้แผนภาพคล้ายซิกแนลทรานสิชันกราฟ

เราสามารถสร้างแผนภาพคล้ายซิกแนลทรานสิชันกราฟเพื่อแสดงพฤติกรรมของวงจรที่เกิดข้อผิดพร่องแบบ Stuck-at ได้โดยขั้นตอนวิธีต่อไปนี้

Algorithm 3 Create_stg_like_diagram

Input STG,Stable_state,fault

Output STG-like diagram,Result of fault

Step 3.1 excite_state = Find_ExciteState(STG,fault)

Step 3.2 Set_signal(Initial_State)

Step 3.3 If Circuit(Initial_State) != initial_state

Step 3.3.1 If circuit is loop

Create_stg_like_diagram (STG,fault)

return (stg-like diagram,loop)

Step 3.3.2 If circuit has hazard

Create_stg_like_diagram (STG,fault)

return (stg-like diagra,loop)

Step 3.3.3 If output_signal(initial_state) != output_signal(current_state)

Result = can test

Return (Result)

Step 3.4 Stable state = Find_stablestate(excite_state)

Step 3.5 Circuit(Stable_state)

Step 3.6 Nextstate = Fire_transition

Step 3.7 Case Nextstate

Step 3.7.1 Loop :

Create_stg_like_diagram (STG,fault)

return (stg-like diagram,loop)

Step 3.7.2 Hazard :

Create_stg_like_diagram (STG,fault)

return (stg-like diagra,loop)

Step 3.7.3 output_signal(Nextstate) != output_signal(expected_state)

Result = can test

Return (Result)

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

Step 3.8 If Stable_state is not last Stable state

Step 3.8.1 Skip to next Stable_state and go to step 3.5

Step 3.8.2 Else

Create_stg_like_diagram (STG,fault)

return (stg-like diagra,loop)

จากขั้นตอนที่ 3 (algorithm 3) เราสามารถอธิบายได้ดังนี้

Step 3.1 ให้หาสถานะวิกฤติที่ทำให้วงจรทำงานผิดพลาด

Step 3.2 ให้เริ่มการทำงานของวงจรที่สถานะเริ่มต้นและเริ่มจำลองการทำงาน

Step 3.3 กรณีที่เริ่มการทำงานที่สถานะเริ่มต้นแล้ววงจรไม่อยู่ในสถานะเริ่มต้น

Step 3.3.1 ถ้าวงจรทำงานต่อไปไม่หยุดอยู่ในสถานะใด ๆ (ลูป) ตรวจสอบโดยการเก็บสถานะต่าง ๆ ของวงจรขณะจำลองการทำงานแล้วพบว่ามีสถานะซ้ำกับสถานะก่อนหน้าแสดงว่าวงจรเกิดลูปให้หยุดการทำงานทันที

Step 3.3.2 ถ้าพบว่าวงจรเกิด hazard ให้หยุดการทำงานทันที

Step 3.3.3 ถ้าวงจรหยุดในสถานะ อื่น แล้วสัญญาณเอาต์พุตของสถานะนั้นมีค่าเท่ากับสถานะเริ่มต้นแสดงว่าวงจรทดสอบได้ให้หยุดการทำงาน

ถ้าสัญญาณเอาต์พุตมีค่าเท่ากันให้เริ่มการทำงานขั้นตอนต่อไปที่สถานะนั้น ๆ

Step 3.4 ตรวจสอบว่าสถานะวิกฤติที่พบอยู่ระหว่างสถานะคงที่ใดบ้าง

Step3.5 ถ้าพบว่ามีสถานะวิกฤติอยู่ในหลาย ๆ ช่วงของสถานะคงที่ให้เลือกช่วงของสถานะคงที่ที่เกิดขึ้นก่อนเป็นสถานะเริ่มเมื่อทำงานตามลำดับของซิกแนลทรานสิชันกราฟ

Step 3.6 เริ่มการป้อนสัญญาณตามลำดับของซิกแนลทรานสิชันกราฟโดยเริ่มจากสถานะที่ได้ตามข้อ 2 และจำลองการทำงานของวงจร

Step 3.6.1 ตรวจสอบการเกิดลูปโดยการเก็บสถานะต่าง ๆ ของวงจรขณะจำลองการทำงานแล้วพบว่ามีสถานะซ้ำกับสถานะก่อนหน้าแสดงว่าวงจรเกิดลูปให้หยุดการทำงานทันที

Step 3.6.2 เมื่อพบว่ามีการเกิด hazard ให้หยุดการทำงาน

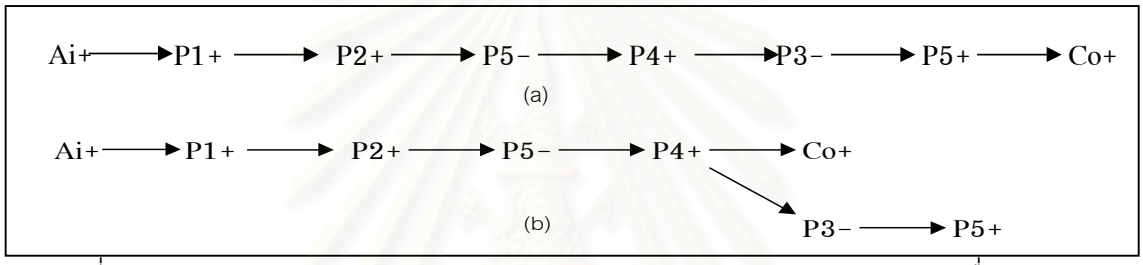
Step 3.6.3 ถ้าวงจรหยุดในลถานะอื่น แล้วสัญญาณเอาต์พุตของสถานะนั้นมีค่าเท่ากับสถานะเริ่มต้นแสดงว่าวงจรทดสอบได้ให้หยุดการทำงาน

Step 3.8 ถ้าพบว่าสถานะวิกฤติอยู่ในช่วงสถานะคงที่อื่น

Step 3.8.1 จะต้องจำลองการทำงานไปจนถึงช่วง สถานะคงที่นั้น ๆ แล้วทำการทำขั้นตอนที่ 3.5

Step 3.8.2 เมื่อทำงานครบทุกช่วงของสถานะคงที่ ถ้าวงจรไม่พบช่วงสถานะคงที่ที่ทำให้ข้อผิดพร่องนี้ทดสอบได้แสดงว่าข้อผิดพร่องชนิดนี้ทดสอบไม่ได้

จากรูป 4.2 การทำงานปกติของวงจรจะต้องเกิดการเปลี่ยนแปลงของสถานะ { Ai Bi P1 P2 P3 P4 P5 C0} ดังนี้ <s1{0 1 0 0 1 0 1 0},Ai+,s2{1 1 1 1 0 1 1 1}> เมื่อเกิดข้อผิดพร่องแบบ stuck-at 1 ที่สัญญาณ P5 ทีเกิดการกระจายของสัญญาณเส้นที่ 2 (P52) แล้วหาค่าสถานะวิกฤติพบว่าอยู่ในช่วงการทำงานในสถานะคงที่นี้ แต่ผลที่ได้จากการเปลี่ยนแปลงก็ยังคงเหมือนกับการเปลี่ยนแปลงของการทำงานของวงจรปกติของวงจรทำให้เราไม่สามารถทดสอบวงจรนี้ได้ ดังนั้นเราจึงจำเป็นต้องสร้างแผนภาพซิกแนลทรานสิชันกราฟเพื่อแสดงพฤติกรรมของวงจรซึ่งได้ดังรูป 4.6



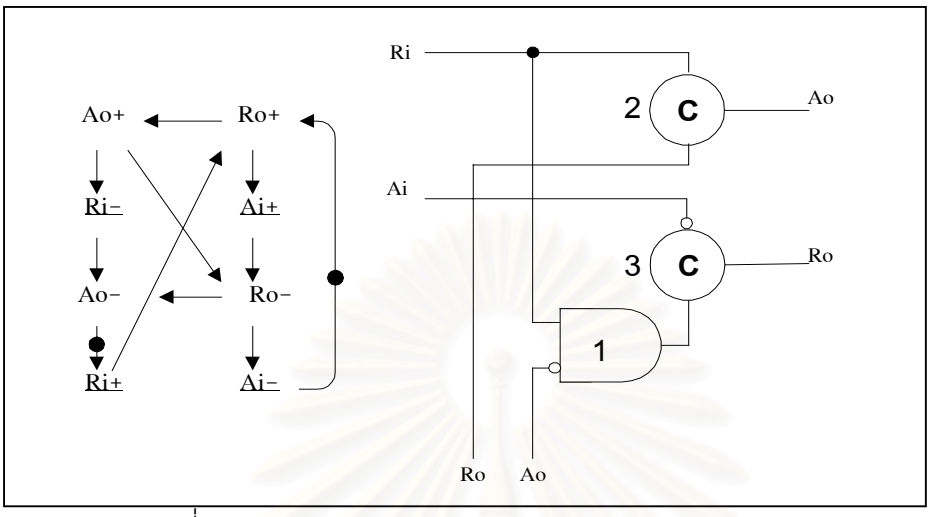
รูปที่ 4.6 แผนภาพคล้ายซิกแนลทรานสิชันของวงจรรูป 4.2 (a) วงจรปกติ (b) เมื่อวงจรเกิดข้อผิดพร่อง stuck-at 1 ที่สัญญาณ P52

จะเห็นได้ว่าพฤติกรรมของวงจรที่ได้จะแตกต่างกันแต่ทำให้เกิดการเปลี่ยนแปลงของสัญญาณเอาต์พุตเหมือนกันซึ่งทำให้ไม่สามารถทดสอบได้ ดังนั้นจึงต้องนำเอาพฤติกรรมของวงจรนี้ไปวิเคราะห์เพื่อเติมจุดทดสอบต่อไป

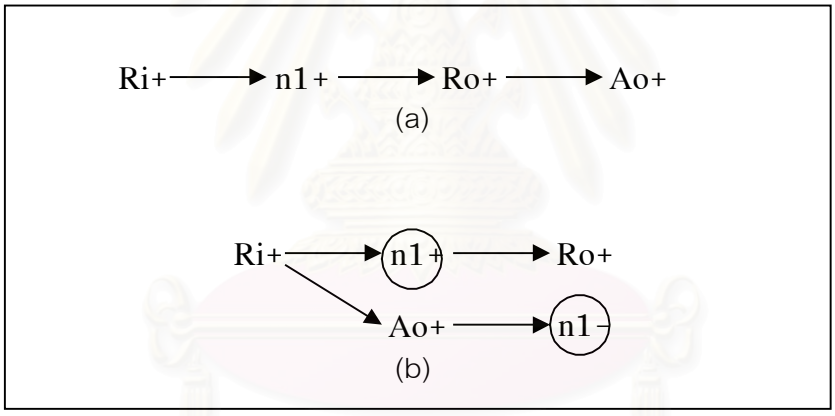
การตรวจสอบการเกิด hazard เราสามารถตรวจสอบได้โดยสังเกตุจากพฤติกรรมของวงจรจากแผนภาพคล้ายซิกแนลทรานสิชันกราฟ ถ้าพฤติกรรมของวงจรในแผนภาพคล้ายซิกแนลทรานสิขันกราฟแสดงถึงเกตใด ๆ ที่มีสัญญาณอินพุตร่วมเกิดการเปลี่ยนแปลงในลักษณะในทิศทางตรงข้ามกัน ในช่วงที่เป็นการเปลี่ยนแปลงในลักษณะพร้อมกัน (concurrent transition)

จากรูป 4.7 เมื่อวงจรเกิดข้อผิดพร่องแบบ stuck-at 1 ที่สัญญาณ Ro จะทำให้วงจรมีพฤติกรรมดังรูป 4.8 ซึ่งเห็นได้ว่าเกิด hazard ที่สัญญาณ n1 ซึ่งเป็นเอาต์พุตของเกต 1 เนื่องจากสัญญาณอินพุตของเกต 1 ซึ่งประกอบด้วยสัญญาณ Ri และ Ao เกิดการเปลี่ยนแปลงสัญญาณในลักษณะตรงกันข้ามกันคือ Ri เกิดการเปลี่ยนแปลงสัญญาณแบบขาขึ้น(+) ส่วน Ao เกิดการเปลี่ยนแปลงสัญญาณแบบขาขึ้น (+) แต่สัญญาณ Ao ผ่าน Inverter ทำให้สัญญาณที่เข้าเกต 1 มีลักษณะตรงข้ามคือ ขาลง (-) ซึ่งตรงกันข้ามกับสัญญาณ Ri ในช่วงการเปลี่ยนแปลงที่เกิดขึ้นพร้อมกันเนื่องจากเราไม่สามารถรู้ว่าสัญญาณ n1 เกิดการเปลี่ยนแปลงชนิด ขาขึ้น (+) เสร็จสิ้น

เมื่อใดเพราะเราไม่สามารถรู้ค่าความหน่วงของเกต 1 ได้ ดังนั้นจึงทำให้สัญญาณที่เป็นเอาต์พุตของเกต 1 คือสัญญาณ n1 มีโอกาสเกิด hazard ได้



รูปที่ 4.7 วงจรและซิกแนลทรานสิชันของวงจร half



รูปที่ 4.8 แผนภาพคล้ายซิกแนลทรานสิชันกราฟแสดงพฤติกรรมของวงจร half (a) ปกติ (b) เมื่อเกิดข้อผิดพร่องแบบ stuck –at 1 ที่สัญญาณ Ro

จากการที่เราใช้แผนภาพคล้ายซิกแนลทรานสิชันกราฟมาแสดงพฤติกรรมของวงจรระหว่างสถานะคงที่พบว่าแผนภาพของวงจรปกติจะมีลักษณะดังนี้

- มีการเปลี่ยนแปลงของสัญญาณอินพุตเพียงสัญญาณเดียวและครั้งเดียวเท่านั้น
- มีการเปลี่ยนแปลงของสัญญาณเอาต์พุตเพียงครั้งเดียวเท่านั้น แต่อาจมีได้หลายสัญญาณ
- การเปลี่ยนแปลงของสัญญาณภายในที่แสดงในซิกแนลทรานสิชันกราฟสามารถเกิดการเปลี่ยนแปลงได้เพียง 2 ครั้งคือการเปลี่ยนแปลงแบบขาลง (-) และ ขาขึ้น (+)

อย่างละ 1 ครั้งเท่านั้น สำหรับสัญญาณภายในที่ไม่พบในซิกแนลทรานสิชันกราฟ สามารถเกิดการเปลี่ยนแปลงได้หลายครั้ง แต่การเปลี่ยนแปลงจะต้องเป็นแบบขาลง (-) หรือขาขึ้น (+) เท่านั้น

4.9 พฤติกรรมของวงจรเมื่อวงจรเกิดข้อผิดพร่อง

เมื่อนำวิธีการทดสอบที่ได้นำเสนอในงานวิจัยฉบับนี้ไปใช้ทดสอบกับวงจรเกณฑ์เปรียบเทียบสมรรถนะจำนวน 29 วงจรดังแสดงไว้ในภาคผนวก ก. ซึ่งจะได้ผลการทดลองดังตารางที่ 6.2 พบว่าลักษณะพฤติกรรมของวงจรอสมวารที่เกิดข้อผิดพร่องแบบ single stuck-at มีดังนี้

1. ทดสอบได้ ได้แก่วงจรที่มีค่าของสัญญาณเอาต์พุตของสถานะคงที่แตกต่างกับสถานะคงที่ปกติ ซึ่งสามารถพบได้มากที่สุด
2. ทดสอบไม่ได้

   2.1 วงจรเกิด Hazard

   2.2 วงจรที่เกิดข้อผิดพร่องมีค่าของสัญญาณเอาต์พุตของสถานะคงที่ค่าเดียวกับสถานะคงที่ปกติของวงจร

   2.3 วงจรเกิดลูป

จากผลการทดลองพบว่าพฤติกรรมของวงจรที่ทดสอบไม่ได้ทั้งสิ้น 41 จุด มีพฤติกรรมในรูปแบบที่ 2.1 จะพบได้มากที่สุดถึง 24 จุด ส่วนพฤติกรรมในรูปแบบที่ 2.2 จะพบได้ 16 จุดและพฤติกรรมในรูปแบบที่ 2.3 จะพบได้ 1 จุด

เราสามารถวิเคราะห์ความซับซ้อนของขั้นตอนวิธีการทดสอบวงจรได้โดยแบ่งเป็นขั้นตอนต่าง ๆ ดังนี้

1. ขั้นตอนการสร้าง SSG

   ในขั้นตอนนี้จะมีการรับอินพุตเป็นซิกแนลทรานสิชันกราฟและจะทำการสร้าง SSG จากซิกแนลทรานสิชันกราฟ โดยให้จำนวนครั้งในการเกิดการเปลี่ยนแปลงสัญญาณเป็นเวลา

   สำหรับตัวอย่างที่มีซิกแนลทรานสิชันที่มีเซตของการเปลี่ยนแปลงสัญญาณขนาด S ตัวโดยแบ่งเป็นสัญญาณอินพุต i ตัว, สัญญาณที่ไม่ใช่อินพุต S-i ตัวและมีเซตการเปลี่ยนแปลงขนาด T ตัว

   ในการทำงานของขั้นตอนวิธีจะพบว่าการสร้าง SSG จะต้องทำการเปลี่ยนแปลงสัญญาณอย่างน้อยที่สุดเท่ากับจำนวนของ T ดังนั้นเราพบว่าเวลาที่ดีที่สุดของขั้นตอนนี้คือ T

ในบางครั้งเราพบว่าการเปลี่ยนแปลงสัญญาณบางตัวอาจเกิดขึ้นได้มากกว่า 1 ครั้ง เนื่องจากจะต้องทำการเลือกการเปลี่ยนแปลงสัญญาณอินพุต ดังนั้นเวลาที่มากที่สุดของการทำงานขั้นตอนนี้คือ i * T

2. ขั้นตอนการสร้าง ESSG

ในขั้นตอนนี้จะมีการรับอินพุตเป็น SSG และสัญญาณอินพุตของวงจร แล้วจะทำการสร้าง ESSG โดยให้จำนวนครั้งในการหาสถานะคงที่และสร้างสถานะคงที่เป็นเวลา

สำหรับตัวอย่างที่มี SSG ที่มีเซตของสถานะคงที่ขนาด S ตัวและเซตของการเปลี่ยนแปลงสัญญาณอินพุต T ตัว และมีจำนวนของสัญญาณอินพุต i ตัว

ในการทำงานของขั้นตอนวิธีจะพบว่าต้องทำการหาสถานะคงที่ใน SSG เป็นจำนวน S ครั้งดังนั้นกรณีที่ดีที่สุดของขั้นตอนนี้คือทุก ๆ สถานะคงที่ไม่จำเป็นต้องสร้างสถานะคงที่ใหม่จึงทำให้เวลาที่ดีที่สุดเท่ากับ S

ในบางครั้งเราพบว่าอาจจะต้องทำการสร้างสถานะคงที่ใหม่ในทุก ๆ สถานะคงที่ซึ่งอาจจะต้องสร้างเป็นจำนวน i – 1 สถานะดังนั้นเวลาที่มากที่สุดจะเท่ากับ S(i-1) หรือมีค่า O(S * I)

3. ขั้นตอนการทดสอบวงจร

ในขั้นตอนนี้จะมีการรับอินพุตเป็น SSG และ ESSG แล้วจะทำการทดสอบวงจร โดยให้จำนวนครั้งในการเปรียบเทียบสถานะคงที่เป็นเวลา

สำหรับตัวอย่างที่มี SSG ที่มีเซตของสถานะคงที่ขนาด s ตัวและเซตของการเปลี่ยนแปลงสัญญาณอินพุต t ตัว และ ESSG ที่มีเซตของสถานะคงที่ขนาด S ตัวและเซตของการเปลี่ยนแปลงสัญญาณอินพุต T ตัว

ในการทำงานของขั้นตอนวิธีจะพบว่าในกรณีที่ดีที่สุดของการทดสอบวงจรคือสัญญาณเอาต์พุตแตกต่างกันตั้งแต่สถานะเริ่มต้นทำให้เวลาที่ดีที่สุดเท่ากับ 1

แต่ในกรณีที่ข้อผิดพร่องที่ทดสอบไม่ได้เราพบว่าอาจจะต้องทำการเปรียบเทียบจนครบทุกสถานะดังนั้นเวลาที่แย่ที่สุดเท่ากับ s + S แต่ค่าของ S มีมากกว่า s ดังนั้นเวลามากที่สุดเท่ากับ S

จากการวิเคราะห์ทั้ง 3 ขั้นตอนพบว่าเวลาที่ดีที่สุดของการทดสอบจะเท่ากับ T หรือจำนวนของเซตของการเปลี่ยนแปลงสัญญาณ

ส่วนในกรณีที่ใช้เวลามากที่สุดเท่ากับ iT + Si +S หรือมีค่าเป็น O(iT + Si) ซึ่งหมายความว่าเวลาที่ใช้จะขึ้นอยู่กับขนาดของสัญญาณอินพุต, ขนาดของเซตการเปลี่ยนแปลงสัญญาณและจำนวนของสถานะคงที่

4.10 สรุป

ในบทนี้กล่าวถึงเงื่อนไขการทดสอบวงจรอสมวารและขั้นตอนวิธีการทดสอบวงจรอสมวารแบบควอไซดีเลย์อินเซนซิทีฟ โดยในขั้นตอนการทดสอบจะแบ่งการทำงานของวงจรเป็นช่วงของสถานะคงที่โดยในแต่ละช่วงของสถานะคงที่จะมีความสัมพันธ์กันด้วยการเปลี่ยนแปลงของสัญญาณอินพุตและนำค่าของสถานะคงที่ของวงจรปกติมาเปรียบเทียบกับสถานะคงที่ของวงจรที่มีข้อผิดพร่อง ซึ่งวิธีการนี้ได้ทำการปรับปรุงมาจากวิธีการทดสอบวงจรอสมวารเพื่อลดความซับซ้อนของพฤติกรรมของวงจร อีกทั้งยังแสดงให้เห็นว่าอุปสรรคที่สำคัญในการทดสอบวงจรคือการเกิด hazard ดังนั้นจึงต้องมีการแสดงพฤติกรรมของวงจรที่เกิดข้อผิดพร่องโดยใช้แผนภาพคล้ายซิกแนลทรานสิชันกราฟให้สามารถนำแผนภาพที่ได้มาวิเคราะห์เติมจุดทดสอบในวงจรได้ในบทต่อไป

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

วงจร	จำนวนข้อผิดพร่อง	จำนวนที่ทดสอบได้	จำนวนทดสอบไม่ได้	Fault Coverage
1. celement	24	22	2	91.6667
2. converta	48	41	7	64.5833
3. ebergen	30	26	4	86.6667
4. half	12	12	0	100
5. hybridf	22	22	0	100
6. nowich	26	25	1	96.1538
7. trimos-send	72	60	12	83.3333
8. vbe5b	24	23	1	95.8333
9. vbe5c	20	20	0	100
10. vbe6a	48	48	0	100
11. vbe10b	56	54	2	96.4286
12. wrdata	22	21	1	95.4545
13. wrdatab	80	73	7	91.25
14. mmu	44	44	0	100
15. chu133	22	21	1	95.4545
16. sendr-done	8	8	0	100
17. alloc-outbound	40	40	0	100
18. rlm	12	12	0	100
19. full	8	8	0	100
20. input	8	8	0	100
21. master-read	60	59	1	98.3333
22. ram-read-sbuf	54	54	0	100
23. sbuf-ram-write	46	46	0	100
24. sbuf-read-ctl	28	28	0	100
25. sbuf-send-ctl	32	31	1	96.875
26. sbuf-send-pkt2	34	34	0	100
27. rcv-setup	16	15	1	93.75
28. nak-pa	40	40	0	100
29. mp-forward-pkt	32	32	0	100
รวม	968	927	41	

ตารางที่ 4.1 แสดงผลการทดสอบวงจรเกณฑ์เปรียบสมรรถนะ

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 5
การเติมจุดทดสอบวงจรอสมวาร

ในบทนี้จะกล่าวถึงการวิเคราะห์พฤติกรรมของวงจรเมื่อเกิดข้อผิดพร่องแบบ single stuck-at ขึ้นในวงจร, ชนิดของจุดทดสอบที่ใช้ในวงจร[13], เงื่อนไขในการเติมจุดทดสอบ, ขั้นตอนวิธีการเติมจุดทดสอบ, แนวทางการลดจุดทดสอบ[14] และแสดงผลการทดลองเมื่อนำขั้นตอนวิธีที่นำเสนอในบทนี้ไปทดลองกับวงจรเกณฑ์เปรียบเทียบสมรรถนะ

5.1 พฤติกรรมของวงจรที่เกิดข้อผิดพร่องและไม่สามารถทดสอบได้

จากบทที่ 4 ทำให้ทราบว่าพฤติกรรมของวงจรที่เกิดข้อผิดพร่องแบบ single stuck-at และไม่สามารถทดสอบได้ ว่ามีพฤติกรรมดังนี้

1. วงจรเกิด hazard สามารถเกิดได้ 2 กรณีคือ
   1.1 hazard เกิดจากการเปลี่ยนแปลงของสัญญาณที่มีผลจากการเกิด Premature transition เพียงอย่างเดียว
   1.2 hazard เกิดจากการเปลี่ยนแปลงสัญญาณตามลำดับของซิกแนลทรานสิชันกราฟปกติและการเปลี่ยนแปลงของสัญญาณที่มีผลจากการเกิด Premature transition
2. วงจรที่เกิดข้อผิดพร่องมีค่าของสัญญาณเอาต์พุตของสถานะคงที่ค่าเดียวกับสถานะคงที่ปกติของวงจร
   2.1 วงจรที่มีข้อผิดพร่องมีสถานะคงที่มีค่าของสัญญาณเดียวกันกับวงจรปกติ
   2.2 วงจรที่มีข้อผิดพร่องมีสถานะคงที่มีค่าของสัญญาณต่างกันกับวงจรปกติ
3. วงจรเกิดลูป

จะเห็นได้ว่าปัญหาสำคัญในการทดสอบวงจรอสมวารแบบควอไซดีเลย์อินเซนซิทีฟที่มีข้อผิดพร่องแบบ stuck-at นั้นคือ hazard ซึ่งเกิดขึ้นโดยการเกิดการเปลี่ยนแปลงสัญญาณแบบ non-monotonous ซึ่งหมายถึงการเกิดการเปลี่ยนแปลงสัญญาณอินพุตของเกตใด ๆ ในทิศทางตรงกันข้ามกัน แนวทางที่นิยมในการแก้ปัญหา hazard คือ

1. การ Disable เกตซึ่งหมายถึงการเติมสัญญาณเพื่อให้เกตที่เกิด hazard อยู่ในสถานะที่ไม่พร้อมจะทำงาน ซึ่งสามารถทำได้ถ้าเกตนั้นเป็น and,nand,or และ nor โดยมีหลักการดังนี้
   - เกต and และ nand สามารถทำให้เกตไม่ทำงานได้โดยเติมสัญญาณที่มีค่าเป็น 0 ให้เป็นสัญญาณอินพุตของวงจร ซึ่งจะเห็นว่าไม่ว่าเกิดการเปลี่ยนแปลงของ

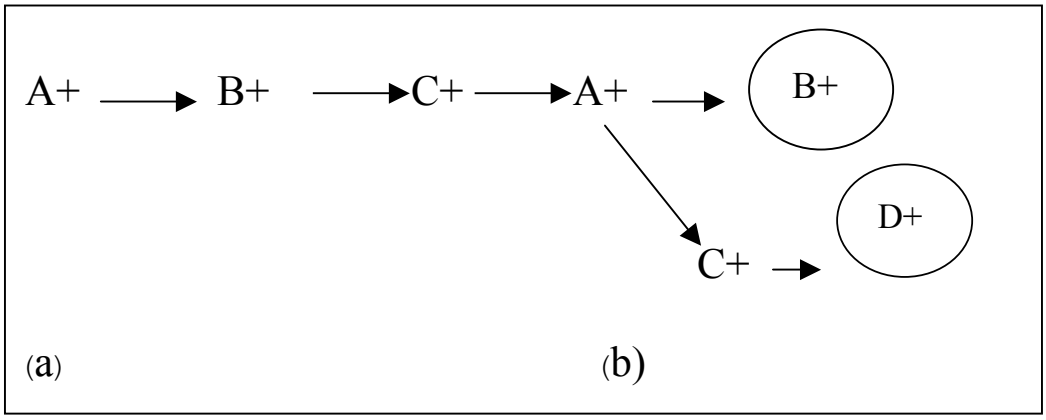
สัญญาณอินพุตของเกตนี้อย่างไรก็ไม่ทำให้เกิดการเปลี่ยนแปลงสัญญาณเอาต์พุตของเกตนี้ และสามารถทำให้วงจรทำงานปกติโดยเปลี่ยนค่าของสัญญาณที่เติมเป็น 1

- เกต or และ nor สามารถทำให้เกตไม่ทำงานได้โดยเติมสัญญาณที่มีค่าเป็น 1 ให้เป็นสัญญาณอินพุตของวงจร ซึ่งจะเห็นว่าไม่ว่าเกิดการเปลี่ยนแปลงของสัญญาณอินพุตของเกตนี้อย่างไรก็ไม่ทำให้เกิดการเปลี่ยนแปลงสัญญาณเอาต์พุตของเกตนี้ และสามารถทำให้วงจรทำงานปกติโดยเปลี่ยนค่าของสัญญาณที่เติมเป็น 1

2. Delay padding หมายถึงการเพิ่มค่าความหน่วงของสัญญาณอินพุตเพื่อให้เกตทำงานเสร็จก่อนที่จะเกิดการเปลี่ยนแปลงของสัญญาณอินพุตอื่น แต่ในวงจรแบบควอไซดีเลย์อินเซนซิทีฟนั้นเราไม่สามารถทราบค่าความหน่วงได้จึงได้ใช้จุดทดสอบเพื่อควบคุมการเกิดการเปลี่ยนแปลงสัญญาณไม่ให้เกิดขึ้น

พฤติกรรมที่เกิดขึ้นในข้อ 1.1 สามารถทำให้ทดสอบได้โดยการควบคุมเกตแต่สำหรับเกต X-or และ C-element ไม่สามารถใช้การควบคุมเกตได้เนื่องจากไม่ว่าเติมสัญญาณที่มีค่าใดสัญญาณเอาต์พุตก็สามารถเปลี่ยนแปลงได้ดังนั้นจึงต้องใช้วิธีควบคุมสัญญาณที่เป็นอินพุตของเกตแทนโดยควบคุมไม่ให้เกิดการเปลี่ยนแปลงของสัญญาณอินพุตของเกตนั้นที่เกิดขึ้นตามลำดับในแผนภาพคล้ายซิกแนลทรานสิชันกราฟของวงจรปกติ และปล่อยให้มีการเกิดการเปลี่ยนแปลงสัญญาณ Premature จะทำให้ไม่เกิด Hazard และสามารถทดสอบได้

สำหรับพฤติกรรมข้อ 1.2 จะมีรูปแบบพฤติกรรมดังรูป 5.1(b) คือการเกิดการเปลี่ยนแปลงของสัญญาณ B+ ซึ่งอยู่ในลำดับของการเปลี่ยนแปลงสัญญาณปกติและการเกิดการเปลี่ยนแปลงสัญญาณ D+ ซึ่งเกิดจากการเปลี่ยนแปลงของสัญญาณ Premature transition เราสามารถทำให้ทดสอบได้โดยควบคุมการเกิดการเปลี่ยนแปลงสัญญาณ B+ ซึ่งทำให้วงจรปกติไม่เกิดการเปลี่ยนแปลงของสัญญาณ B+, C- และ D+ แต่วงจรที่เกิดข้อผิดพร่องจะเกิดการเปลี่ยนแปลง C- และ D+ และจะไม่ทำให้เกิด hazard เพราะไม่เกิด transition B+ ซึ่งทำให้เราสามารถทดสอบวงจรได้ แต่บางกรณีอาจพบว่าการเกิดการเปลี่ยนแปลง C- อาจทำให้เกิด hazard ได้แต่เราก็สามารถทำให้ทดสอบได้โดยใช้วิธีการเดียวกับข้อ 1.1



รูปที่ 5.1 รูปแบบการเกิดการเปลี่ยนแปลงสัญญาณ (a) ปกติ (b) เกิดข้อผิดพร่อง

พฤติกรรมในข้อ 2.1 จะมีลักษณะคล้ายกับรูป 5.1 (a) แต่จะแตกต่างกันที่ไม่เกิด hazard ในวงจร ดังนั้นเราเติมจุดควบคุมไม่ให้เกิดการเปลี่ยนแปลงสัญญาณ B+ ก็จะทำให้ทดสอบวงจรได้

ส่วนพฤติกรรมในข้อ 2.2 นั้นจะเราสามารถทำให้ทดสอบได้โดยเติมจุดสังเกตค่าของสัญญาณในสถานะคงที่ของวงจรที่เกิดข้อผิดพร่องที่มีค่าต่างกับค่าของสัญญาณในสถานะคงที่ของวงจรปกติก็จะทำให้วงจรทดสอบได้

และพฤติกรรมในข้อ 3 นั้นเราไม่สามารถใช้วิธีการปกติทดสอบได้จะต้องมีขั้นตอนเพิ่มเติมเพื่อทดสอบซึ่งจะกล่าวในภายหลัง

5.2 จุดทดสอบ

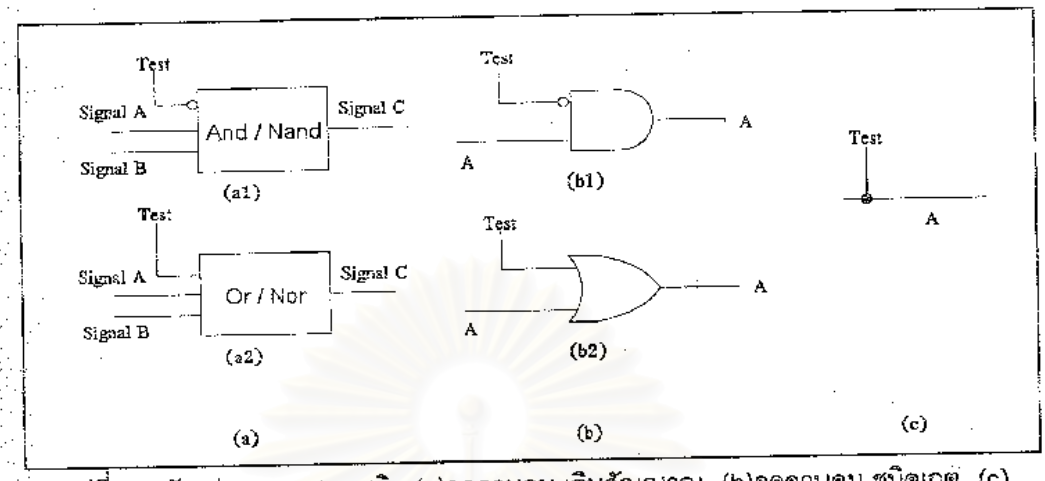
จุดทดสอบที่ใช้ในการเติมในวงจรมี 2 ประเภทคือ

1. จุดควบคุม หมายถึงการเติมสัญญาณเพื่อที่จะควบคุมสัญญาณที่อยู่ในวงจรมีค่าตามกำหนด แบ่งเป็น 2 ประเภทคือ
   1.1 จุดควบคุม 0 (C-0) หมายถึงควบคุมสัญญาณให้เป็น 1
   1.2 จุดควบคุม 1 (C-1) หมายถึงควบคุมสัญญาณให้เป็น 0
2. จุดสังเกต หมายถึงการเติมสัญญาณเพื่อที่ทราบค่าของสัญญาณที่อยู่ในวงจร

การเติมจุดควบคุมที่ใช้ในการวิจัยนี้มี 2 วิธีคือ

1. การเติมสัญญาณเพื่อควบคุมค่าเอาต์พุตของเกตหรือเรียกว่าการ disable การทำงานของเกต โดยสามารถใช้ได้กับสัญญาณที่เป็นเอาต์พุตของเกต and,nand,or และ nor เท่านั้น ตัวอย่างดังรูป 5.2 (a)
2. การเติมเกต and สำหรับควบคุมสัญญาณให้เป็น 0 หรือ or เพื่อควบคุมสัญญาณให้เป็น 1 ซึ่งมีลักษณะดังรูป 5.2 (b)

สำหรับจุดสังเกตสามารถเติมสัญญาณไปที่สัญญาณดังกล่าวโดยตรงเพื่อทำให้สัญญาณนั้นเสมือนเอาต์พุตหลักของวงจรดังรูป 5.2(c)



รูปที่ 5.2 ตัวอย่างจุดทดสอบชนิด (a)จุดควบคุม เติมสัญญาณ (b)จุดควบคุม ชนิดเกต (c) จุดสังเกต

5.3 เงื่อนไขในการเติมจุดทดสอบ

1. การเติมจุดทดสอบจะต้องไม่ไปรบกวนคุณสมบัติ Isochronic fork ของวงจร นั้นหมายความว่าเราไม่สามารถเติมจุดทดสอบในสัญญาณย่อยที่เกิดการกระจายของสัญญาณได้
2. การเติมจุดทดสอบจะต้องสามารถทำให้วงจรทำงานได้โดยปกติ และไม่ทำให้วงจรเกิด hazard
3. การเติมจุดควบคุมเป็นการเติมเพื่อที่จะควบคุมไม่เกิดการเปลี่ยนแปลงของสัญญาณดังนั้นในการเติมเพื่อไม่ให้การเกิดการเปลี่ยนแปลงสัญญาณแบบ falling (-) คือการเติมจุดควบคุม 1 ในสัญญาณนั้น ๆ และการเติมเพื่อไม่ให้การเกิดการเปลี่ยนแปลงสัญญาณแบบ rising คือการเติมจุดควบคุม 0 ในสัญญาณนั้น ๆ

5.4 ขั้นตอนวิธีการเติมจุดทดสอบ

ขั้นตอนวิธีการเติมจุดทดสอบในวงจรอสมสารที่มีโมเดลความหน่วงแบบควอไซดีเลย์อินเซนซิทีฟที่ไม่สามารถทดสอบได้ในขั้นตอนปกติ มีขั้นตอนดังนี้

Algorithm 4 testpoint_insertion1

Input Fault_STG-like_diagram,Normal_STG-like_diagram

Output Testpoint

Step 4.1 **Analyze_Transition(Fault_STG-like_diagram,Normal_STG-like_diagram)**
Case
Step 4.2 : Hazard
Step 4.2.1 if normal transition is critical signal
Testpoint = control point at normal signal transition that cause of hazard
If still has hazard
Find critical signal
Testpoint = control point at critical signal
Step 4.2.2 else
Testpoint=control point at critical signal
Step 4.3 : Stable state has Same output signal
If normal_stable state = fault_stable_state
Testpoint = control point at the concurrent signal transition with premature signal
If transition after premature have not output transition
Testpoint = observe point at premature transition
Step 4.4 : Loop
Testpoint = Algorithm 5
Step 4.5 return (testpoint)

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

จากขั้นตอนที่ 4 (algorithm 4) เราสามารถอธิบายได้ดังนี้

Step 4.1 เป็นขั้นตอนการตรวจสอบพฤติกรรมของวงจรว่าพฤติกรรมของวงจรที่เกิดข้อผิดพร่องมีลักษณะใด

Step 4.2 พบว่าเกิด hazard ในวงจรที่เกิดข้อผิดพร่อง

Step 4.2.1 ถ้าพบว่าเกิด hazard ขึ้นแล้วพบว่า hazard มีสาเหตุจากสัญญาณที่เกิดการเปลี่ยนแปลงที่เกิดขึ้นตามลำดับของแผนภาพคล้ายซิกแนลทรานสิชันกราฟปกติ ให้เติมจุดควบคุมที่สัญญาณที่เกิดการเปลี่ยนแปลงสัญญาณที่เป็นสาเหตุให้เกิด hazard เพื่อไม่ให้เกิดการเปลี่ยนแปลงนั้น

เมื่อเติมแล้วตรวจสอบว่า hazard ยังคงเกิดขึ้นหรือไม่ ถ้าเกิดขึ้นอีกให้หาสัญญาณที่ทำให้เกิด hazard แล้วเติมจุดควบคุมที่สัญญาณนั้น ๆ

Step 4.2.2 ถ้าพบว่าเกิด hazard ขึ้นแล้ว hazard นั้นไม่ได้มีสาเหตุจากการเปลี่ยนแปลงสัญญาณที่เกิดการเปลี่ยนแปลงสัญญาณแบบ premature transition ให้เติมจุดควบคุมที่สัญญาณที่ทำให้เกิด hazard

Step 4.3 พบว่าวงจรไม่ได้เกิด hazard แต่วงจรที่เกิดข้อผิดพร่องมีสถานะคงที่ที่มีสัญญาณเอาต์พุตเหมือนกับสถานะคงที่ปกติ

ตรวจสอบว่าสถานะคงที่ทั้งสองเหมือนกันหรือไม่ถ้าเหมือนกันเติมจุดควบคุมที่สัญญาณที่สัญญาณที่เกิดการเปลี่ยนแปลงในลักษณะพร้อมกัน (concurrent) กับสัญญาณที่เกิด Premature

ถ้าพบว่าสัญญาณเอาต์พุตของทั้งสองเหมือนกันหรือไม่ ให้เติมจุดสังเกตที่สัญญาณที่เกิด Premature transition

Step 4.4 กรณีที่เกิดลูปให้ทำงานในขั้นตอนวิธีที่ 5 ต่อไป

ถ้าพบว่าวงจรเกิดลูปขึ้นทำให้ไม่สามารถทดสอบด้วยการเปลี่ยนแปลงของสัญญาณอินพุตปกติได้ จะต้องทำให้ข้อผิดพร่องทำงานโดยเติมจุดทดสอบขั้นตอนในการเติมจุดทดสอบในสัญญาณที่เกิดข้อผิดพร่องโดยเป็นจุดควบคุม 1 ถ้าข้อผิดพร่องเป็นแบบ stuck-at 0 และเป็น 0 ถ้าข้อผิดพร่องเป็นแบบ stuck-at 1 และเปลี่ยนลำดับการเปลี่ยนแปลงของสัญญาณอินพุตเดิมเป็นให้เกิดการเปลี่ยนแปลงสัญญาณที่จุดทดสอบจะทำให้วงจรทดสอบได้

ขั้นตอนเพิ่มเติมมีขั้นตอนดังนี้

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

Algorithm 5 Testpoint_insertion2

Input STG,fault,normal_STG-like_diagram

Output Testpoint

Step 5.1 Testpoint = control point at Fault signal

Step 5.2 Active testpoint

Step 5.3 If this fault cannot test

Generate_STG-like diagram

Do algorithm 5

Step 5.4 Return testpoint

จากขั้นตอนวิธีที่ 5 (algorithm 5) เราสามารถอธิบายการทำงานได้ดังนี้

Step 5.1 จะต้องเติมจุดควบคุมในสัญญาณที่เกิดข้อผิดพร่องโดยเติมจุดควบคุมชนิด Control 0 เมื่อเป็นข้อผิดพร่องเป็นแบบ Stuck-at-1 และเติมจุดควบคุมชนิด Control 1 เมื่อเป็นข้อผิดพร่องเป็นแบบ Stuck-at-0 และในการเติมจะต้องเติมก่อนจุดที่เกิดการกระจายของสัญญาณ

Step 5.2 ให้จุดทดสอบทำงาน

Step 5.3 ตรวจสอบว่าวงจรทดสอบได้หรือไม่ถ้าวงจรทดสอบไม่ได้ให้สร้างแผนภาพคล้ายซิกแนลทรานสิชันกราฟของวงจรปกติและวงจรที่เกิดข้อผิดพร่องแล้วนำไปทำในขั้นตอนที่ 4 ต่อไป

5.5 ผลการทดลองการเติมจุดทดสอบ

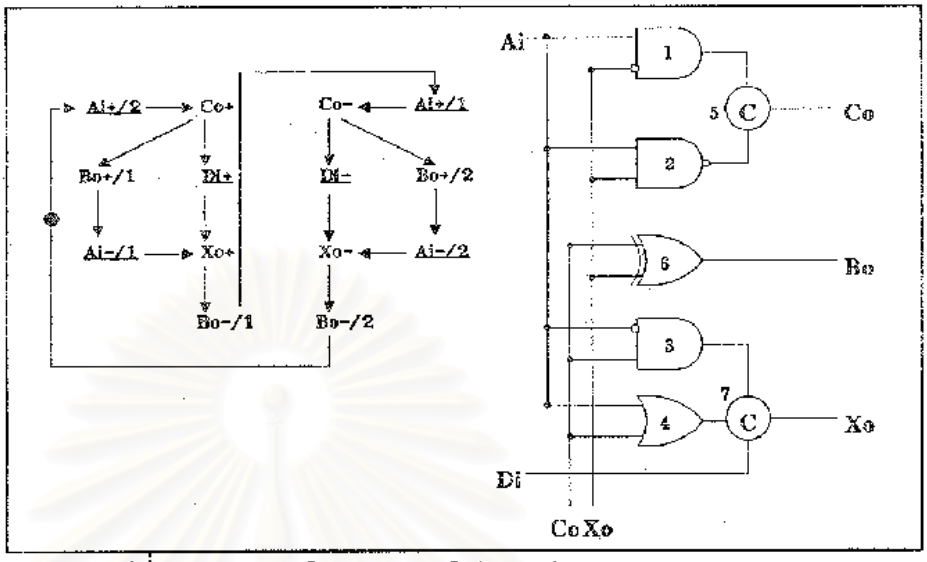
เมื่อได้ทำการทดสอบขั้นตอนวิธีที่ได้ออกแบบกับวงจรเกณฑ์เปรียบเทียบสมรรถนะจำนวน 29 วงจร ดังที่แสดงไว้ในภาคผนวก ก พบว่าสามารถเติมจุดทดสอบได้ในแต่ละกรณีดังนี้

กรณีที่ 1. วงจรเกิด hazard แบ่งเป็น 2 ประเภทได้แก่

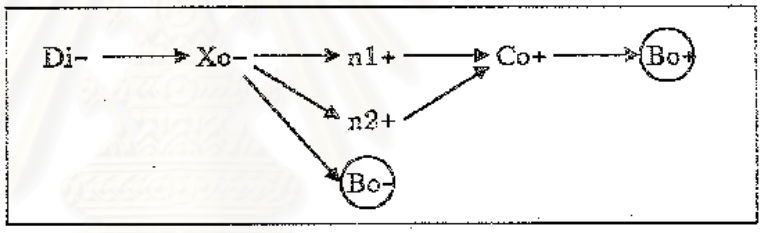
1.1 hazard เกิดจากการเปลี่ยนแปลงของสัญญาณ Premature transition

จากรูปที่ 5.3 เมื่อทำให้เกิดการเปลี่ยนแปลงสัญญาณอินพุต Di– ขณะวงจรอยู่ในสถานะ Ai Di Co Xo Bo = 1 1 0 1 1 แล้ววงจรจะทำให้วงจรไม่เกิดการเปลี่ยนแปลง แต่เมื่อวงจรเกิดข้อผิดพร่องแบบ stuck-at 0 ที่สัญญาณ Ai4 แล้วทำให้เกิดการเปลี่ยนแปลงสัญญาณ Di- วงจรจะมีพฤติกรรมดังรูป 5.4 ซึ่งเห็นได้ว่าเกิด hazard ขึ้นที่สัญญาณ Bo เราสามารถสามารถเติมจุดทดสอบได้โดยการควบคุมไม่ให้เกิดการเปลี่ยนแปลงสัญญาณ Bo นั่นหมายถึงการเติมจุดควบคุม 1 (C-1) ที่จุด Bo นั้นเองซึ่งทำให้วงจรทดสอบได้ โดยจะเห็นว่าถ้าวงจรที่มีข้อผิด

พร่องเกิดขึ้นจะพบว่า เมื่อเปลี่ยนแปลงสัญญาณ Di- จะทำให้เกิด Xo- และ Co+ และไม่ทำให้เกิด Hazard ดังนั้นจะทำให้ทดสอบได้



รูปที่ 5.3 วงจรและซิกแนลทรานสิชันกราฟของวงจร ebergen

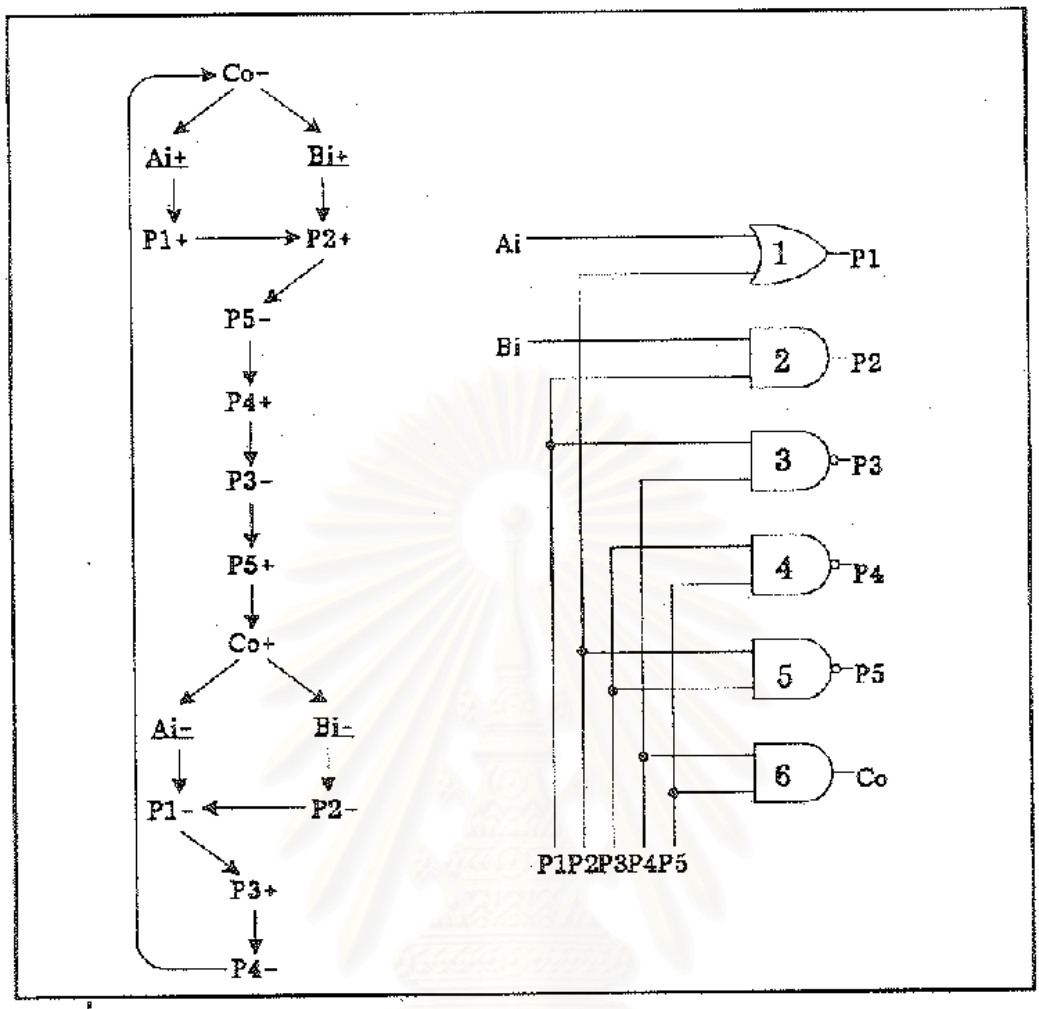


รูปที่ 5.4 แผนภาพคล้ายซิกแนลทรานสิชันกราฟแสดงพฤติกรรมของวงจร ebergen เมื่อเกิดข้อผิดพร่องแบบ stuck –at 0 ที่สัญญาณ Ai4

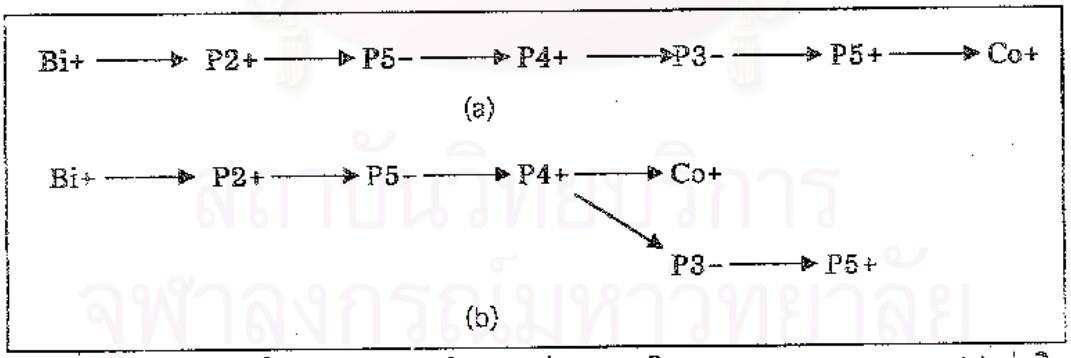
กรณีที่ 2 วงจรที่เกิดข้อผิดพร่องมีค่าของสัญญาณเอาต์พุตของสถานะคงที่ค่าเดียวกับสถานะคงที่ปกติของวงจร แบ่งเป็น 2 ประเภทได้แก่

3.1 วงจรที่มีข้อผิดพร่องมี stable state มีค่าของสัญญาณเดียวกันกับวงจรปกติ

จากรูป 5.5 เมื่อวงจรเกิดข้อผิดพร่องแบบ stuck-at 1 ที่สัญญาณ P52 พบว่าวงจรจะมีลักษณะการทำงานดังรูปที่ 5.6 ซึ่งพบว่าวงจรที่เกิดข้อผิดพร่องมี stable state เดียวกับวงจรปกติ แต่เมื่อเติมจุดควบคุมไม่ให้เกิดการเปลี่ยนแปลงหรือจุดควบคุม 1 (C-1) ในสัญญาณ P3 ก็จะทำให้วงจรปกติไม่เกิดการเปลี่ยนแปลงของสัญญาณ Co แต่วงจรที่เกิดข้อผิดพร่องเกิดการเปลี่ยนแปลงสัญญาณ Co ดังนั้นจึงสามารถทดสอบวงจรนี้ได้



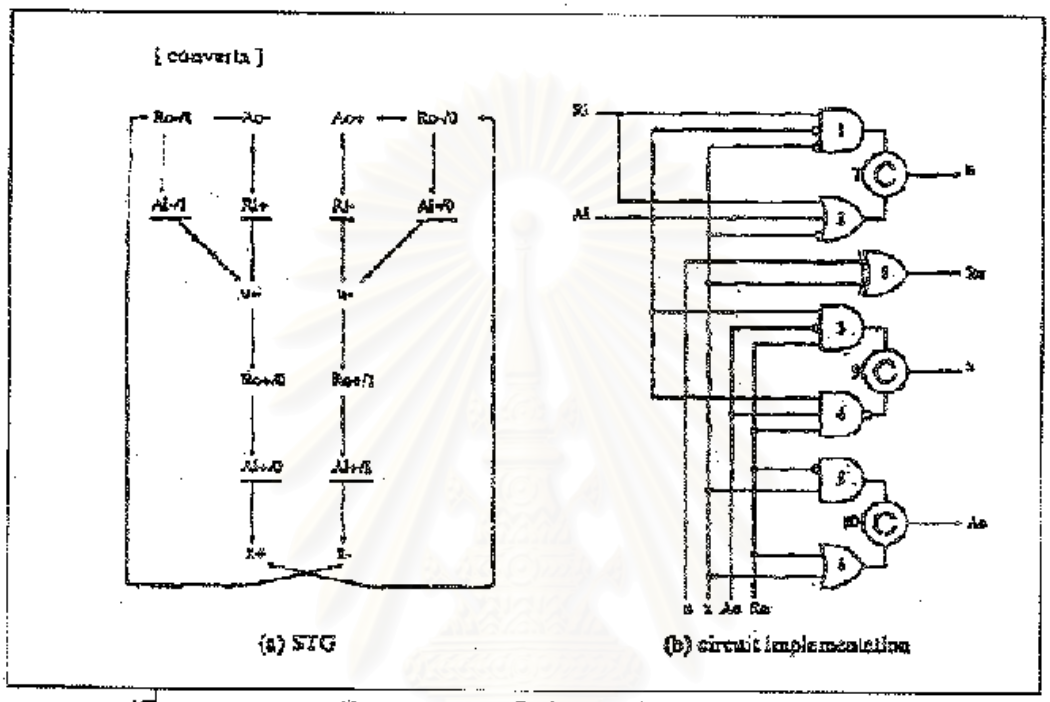
รูปที่ 5.5 วงจรและซิกแนลทรานสิชันกราฟของวงจร C-element



รูปที่ 5.6 แผนภาพคล้ายซิกแนลทรานสิชันกราฟแสดงพฤติกรรมของวงจร C-element (a) ปกติ (b) เกิดข้อผิดพร่องแบบ stuck –at 1 ที่สัญญาณ P52

3.2 วงจรมีค่าของสัญญาณที่สถานะคงที่ต่างกันกับวงจรปกติ

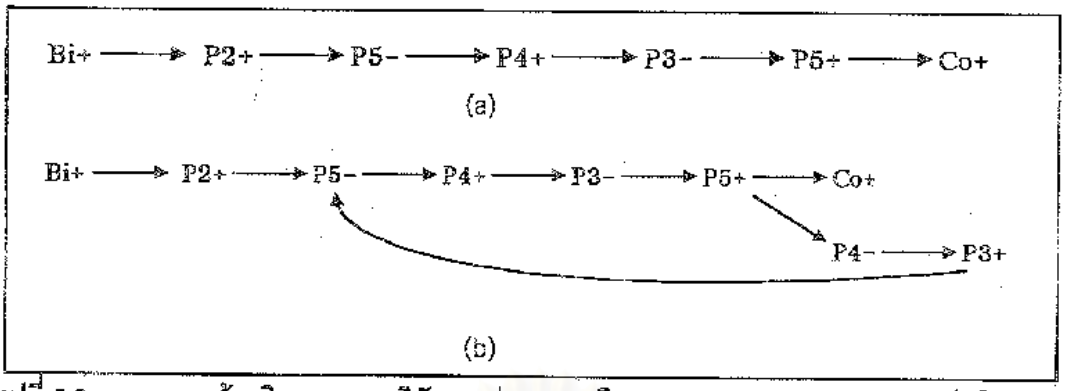
จากรูปที่ 5.7 เมื่อวงจรเกิดข้อผิดพร่องแบบ stuck-at 1 ที่สัญญาณ x2 พบว่าเมื่อวงจรอยู่ในสถานะเริ่มต้นค่าของสถานะเริ่มต้นของสัญญาณที่เกิดข้อผิดพร่องมีสัญญาณภายในที่มีค่าต่างกันคือสัญญาณ n2 ดังนั้นเมื่อเติมจุดสังเกตที่สัญญาณ n2 ก็จะทำให้สามารถทดสอบวงจรได้



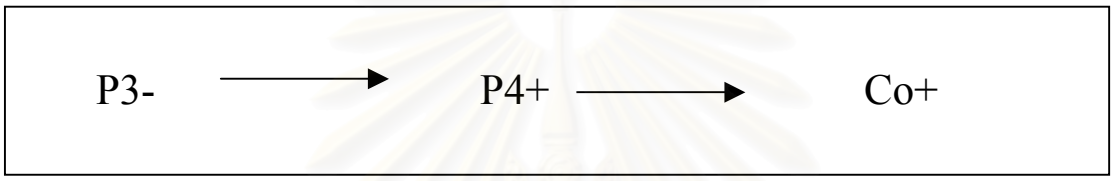
รูปที่ 5.7 วงจรและชีกแนพทรานสิชั่นกราฟของวงจร Converta

กรณีที่ 3 วงจรเกิดลูป

จากรูปที่ 5.5 เมื่อวงจรเกิดข้อผิดพร่องแบบ stuck-at 1 ที่สัญญาณ P31 จะทำให้พฤติกรรมดังรูป 5.8 ซึ่งพบว่าวงจรเกิด loop ขึ้นทำให้วงจรไม่สามารถทดสอบได้ในขั้นตอนปกติ จากขั้นตอนเพิ่มเติมพบว่าถ้าเติมจุดควบคุม 0 ที่สัญญาณ P3 และเมื่อวงจรอยู่ในสถานะคงที่ให้จุดควบคุมทำงานพบว่าจะเกิดการเปลี่ยนแปลงสัญญาณดังรูป 5.9 ซึ่งพบว่าเกิดการเปลี่ยนแปลงที่สัญญาณเอาต์พุตแต่ถ้าวงจรเกิดข้อผิดพร่องเกิดขึ้นจะไม่พบการเปลี่ยนแปลงใด ๆ จึงทำให้วงจรสามารถทดสอบได้



รูปที่ 5.8 แผนภาพคลื่นซิกแนลทรานสิชันกราฟแสดงพฤติกรรมของวงจร C-element (a) ปกติ (b) เกิดข้อผิดพร่องแบบ stuck –at 1 ที่สัญญาณ P31



รูปที่ 5.9 แผนภาพคลื่นซิกแนลทรานสิชันกราฟของวงจร C-element เมื่อเติมจุดควบคุม 0 ในสัญญาณ P3

5.6 แนวทางการลดจุดทดสอบ

ในส่วนนี้จะอธิบายถึงขั้นตอนวิธีการลดจำนวนจุดทดสอบ โดยมีเป้าหมายเพื่อที่จะใช้จำนวนจุดทดสอบให้เหมาะสม แนวทางเบื้องต้นที่ใช้ในการลดจุดทดสอบคือการเพิ่ม Fault Coverage ของวงจรโดยใช้ ESSG มาช่วยในขั้นตอนการทดสอบของวงจร ซึ่งสามารถช่วยลดจุดทดสอบได้ เนื่องจากจำนวนข้อผิดพร่องที่ทดสอบไม่ได้น้อยลง แต่เพื่อที่จะทำให้จุดทดสอบน้อยลงจึงได้นำเอาวิธีการลดจุดทดสอบของวงจรสมวารมาประยุกต์ใช้

ซึ่งในขั้นตอนการลดจำนวนจุดทดสอบนี้จะใช้หลักการของขั้นตอนวิธีแบบตระกราม (Greedy algorithm) และจะแบ่งเป็น 2 ขั้นตอนคือขั้นตอนการให้น้ำหนักจุดทดสอบและขั้นตอนการเลือกจุดทดสอบ

ในขั้นตอนที่ 1 เราก็จะใช้วิธีการให้น้ำหนักของจุดทดสอบแต่ละจุดเพื่อที่จะแสดงถึงลำดับความสำคัญของจุดทดสอบแต่ละจุดที่เพิ่มลงในวงจรแล้วทำให้ข้อผิดพร่องของวงจรทดสอบได้ โดยจะให้ P เป็นเซตของข้อผิดพร่องทั้งหมดที่ทดสอบไม่ได้และมี p เป็นสมาชิกของ P เราก็จะให้น้ำหนักของจุดทดสอบมีค่ามากเมื่อจุดทดสอบนั้น ๆ สามารถทำให้ข้อผิดพร่อง p ทดสอบได้มากกว่าจุดทดสอบที่มีน้ำหนักน้อย

สำหรับทุก ๆ จุดทดสอบที่ทำให้ p ทดสอบได้ เราจะพิจารณาการให้น้ำหนักดังนี้

1. ถ้าจุดทดสอบจุดเดียวทำให้ข้อผิดพร่อง p ทดสอบได้ให้เพิ่มน้ำหนักจุดทดสอบนั้น 1

2. ถ้าจุดทดสอบมากกว่า 1 จุดทำให้ข้อผิดพร่อง p ทดสอบได้ให้เพิ่มน้ำหนักจุดทดสอบทุกจุดที่ทำให้ p ทดสอบได้ 1

เราสามารถแสดงขั้นตอนการให้น้ำหนักจุดทดสอบได้ดังนี้ โดยให้น้ำหนักของจุดทดสอบ e_l มีค่าเป็น $W(e_l)$

Algorithm 6 Testpoint_Weight

Input P,e

Output W(e)

Step 6.1 Initialize $W(e_l) = 0$

Step 6.2 For every $p \in P$ do

Step 6.2.1 If a single test point (e_l) placed makes p testable

$W(e_l) = W(e_l) + 1$

Step 6.2.2 If more than one test point (e_l) is required to makes p testable

For i = 1 to n do

$W(e_l) = W(e_l) + 1$

Step 6.3 Return W(e)

เมื่อเราให้น้ำหนักของจุดทดสอบแต่ละจุดแล้วขั้นตอนต่อไปจะเป็นขั้นตอนการเลือกจุดทดสอบ โดยเราพิจารณาแต่ละข้อผิดพร่อง p โดยจะเลือกจุดทดสอบที่มีน้ำหนักมากที่สุดที่ทำให้ข้อผิดพร่อง p ทดสอบได้ หลังจากนั้นก็จะทำการตัดเอาทุกข้อผิดพร่อง p ที่ถูกทำให้ทดสอบได้จากจุดทดสอบนี้ออกจาก P แล้วทำการปรับปรุงน้ำหนักของจุดทดสอบอีกครั้ง จากนั้นจะทำการเลือกจุดทดสอบจนกระทั่งครบทุกข้อผิดพร่อง

เราสามารถแสดงขั้นตอนการเลือกจุดทดสอบได้ดังนี้

Algorithm 7 Testpoint_Selection

Input P,e,W(e)

Output Test point

Step 7.1 Repeat until P is empty

Step 7.1.1 Select test point with the mazimum weight

Step 7.1.2 Remove all the fault from P that Testable due to this test point

Step 7.1.3 Update the test point weights

Step 7.2 Return Test point

เมื่อเรานำเอาขั้นตอนการลดจุดทดสอบไปใช้กับผลการทดลองพบว่าวงจรที่สามารถลดจุดทดสอบได้จะต้องมีข้อผิดพร่องที่ทดสอบไม่ได้มากกว่า 1 จุดและจุดทดสอบที่ทำให้ข้อผิดพร่องต่าง ๆ ทดสอบได้มีค่าซ้ำกันเช่นวงจร converta และ วงจร trimos-send แต่เมื่อนำขั้นตอนวิธีมาใช้พบว่าไม่สามารถลดได้เนื่องจากจุดทดสอบที่ได้จากขั้นตอนการเติมจุดทดสอบมีจำนวนไม่มากและวงจรอสมวารมักจะมีข้อผิดพร่องที่ทดสอบไม่ได้น้อยมาก ดังนั้นการนำขั้นตอนการลดจุดทดสอบมาใช้อาจจะไม่ได้ประโยชน์มากนัก

เราสามารถวิเคราะห์ความซับซ้อนของขั้นตอนวิธีในการเติมจุดทดสอบแบ่งเป็นขั้นตอนดังนี้

1. ขั้นตอนการสร้างแผนภาพคล้ายซิกแนลทรานสิชันกราฟ

ในขั้นตอนนี้จะมีการรับอินพุตเป็นเซตการเปลี่ยนแปลงสัญญาณอินพุตและวงจร แล้วทำการจำลองการทำงานของวงจรจากลำดับการเปลี่ยนแปลงของสัญญาณอินพุต โดยให้จำนวนครั้งการทำงานของเกตเป็นเวลา

สำหรับตัวอย่างวงจรที่มีจำนวนเกตในวงจรทั้งสิ้น n เกตและจำนวนสัญญาณ m สัญญาณแบ่งเป็นสัญญาณอินพุต i สัญญาณ, สัญญาณเอาต์พุต o สัญญาณ, สัญญาณย้อนกลับ j สัญญาณ, และสัญญาณภายใน k สัญญาณ โดยที่ m=(i + o + j + k) และมีเซตการเปลี่ยนแปลงของสัญญาณอินพุต t ตัว ซึ่งจะมีค่าไม่เกิน 2i ตัว เนื่องจากคุณสมบัติ Single cycle transitions ของซิกแนลทรานสิชันกราฟ

ในการทำงานเราพบว่ากรณีที่ดีที่สุดของขั้นตอนการทำงานคือเมื่อวงจรอยู่ในสถานะเริ่มต้นในการทำงานของวงจรปกติก็จะไม่มีการทำงาน ส่วนวงจรที่เกิดข้อผิดพร่องเกิดการทำงานของเกต 1 เกต ดังนั้นจะได้ว่าเวลาการทำงานของขั้นตอนนี้ก็จะเท่ากับ 1

เนื่องจากคุณสมบัติการตอบรับของซิกแนลทรานสิชันกราฟ ในแต่ละรอบจะต้องเกิดการเปลี่ยนแปลงของสัญญาณเอาต์พุตหรือสัญญาณย้อนกลับ และสัญญาณเอาต์พุตและสัญญาณย้อนกลับเกิดการเปลี่ยนแปลงได้ไม่เกิน 2 ครั้ง ดังนั้นจำนวนรอบของวงจรทั้งหมดจะเท่ากับ 2*(o+j) และในแต่ละรอบจะเกิดการทำงานของเกตได้ไม่เกิน n เกตดังนั้นเวลาการทำงานที่มากที่สุดของขั้นตอนนี้มีค่าเท่ากับ 2(o+j) * n หรือมีค่าเป็น O(n(o+j))

2. ขั้นตอนการเติมจุดทดสอบ

ในขั้นตอนนี้จะมีการรับอินพุตเป็นแผนภาพคล้ายซิกแนลทรานสิชันกราฟและชนิดของพฤติกรรมของวงจรที่เกิดข้อผิดพร่อง โดยให้จำนวนครั้งในการเดินทางผ่านการเปลี่ยนแปลงของสัญญาณบนแผนภาพคล้ายซิกแนลทรานสิชันกราฟเป็นเวลา แล้วทำการตรวจสอบแต่ละกรณีดังนี้

2.1 กรณี hazard

สำหรับตัวอย่างที่มีแผนภาพคล้ายซิกแนลทรานสิชันกราฟขนาดเซตการเปลี่ยนแปลงสัญญาณ n ตัวและเซตของการเปลี่ยนแปลง m ตัว

ในกรณีที่การเกิด hazard เกิดจากการเปลี่ยนแปลงสัญญาณที่เกิดก่อนข้อกำหนดเพียงอย่างเดียว การทำงานเราจะต้องทำการหาการเปลี่ยนแปลงสัญญาณที่ทำให้เกิด hazard จากแผนภาพคล้ายซิกแนลทรานสิชันกราฟแล้วทำการเติมจุดทดสอบที่สัญญาณนั้น ๆ ในกรณีที่ดีที่สุดพบว่าเราจะต้องใช้เวลาทั้งสิ้น n

ในการเติมจุดทดสอบจะต้องมีการตรวจสอบเงื่อนไขว่าจุดทดสอบที่เติมนั้นผิดเงื่อนไขหรือไม่ ดังนั้นในกรณีที่แย่ที่สุดคือการหาสัญญาณ hazard 2 ครั้งแล้วการเติมจุดควบคุมที่สัญญาณที่เกิด hazard ไม่ทำให้ทดสอบได้จะต้องทำการเติมจุดสังเกตุควบคู่ไปด้วย จึงทำให้เวลาที่มากที่สุดเท่ากับ 3n

สำหรับกรณีที่การเกิด hazard เกิดจากการเปลี่ยนแปลงสัญญาณที่เกิดก่อนข้อกำหนดและการเปลี่ยนแปลงสัญญาณตามข้อกำหนด การทำงานเราจะต้องทำการหาการเปลี่ยนแปลงในเส้นทางปกติที่ทำให้เกิด hazard แล้วทำการเติมจุดทดสอบเพื่อหยุดการเปลี่ยนแปลงสัญญาณในเส้นทางปกติ ในกรณีที่ดีที่สุดเราจะต้องใช้เวลาทั้งสิ้น n

แต่การเติมจุดทดสอบจะต้องมีการตรวจสอบเงื่อนไขว่าจุดทดสอบที่เติมนั้นผิดเงื่อนไขหรือไม่ จึงทำให้กรณีที่ใช้เวลามากที่สุดคือการเติมจุดควบคุมและการเติมจุดสังเกตุเพื่อให้วงจรทดสอบได้ ซึ่งกรณีนี้ใช้เวลาทั้งสิ้น 2n

2.2 กรณีวงจรมีค่าของสถานะคงที่เท่ากัน

สำหรับตัวอย่างที่มีแผนภาพคล้ายซิกแนลทรานสิชันกราฟขนาดเซตการเปลี่ยนแปลงสัญญาณ n ตัวและเซตของการเปลี่ยนแปลง m ตัว

ในกรณีที่สถานะคงที่มีค่าเหมือนกันเราพบว่าจะมีการทำงานเหมือนกับกรณีการเกิด hazard ที่เกิดจากการเปลี่ยนแปลงสัญญาณก่อนข้อกำหนดและการเปลี่ยนแปลงสัญญาณตามข้อกำหนด ดังนั้นเวลาที่ใช้ในการทำงานจะมีค่าเท่ากันคือ n ในกรณีที่ดีที่สุดและ O(2n) ในกรณีที่ใช้เวลามากที่สุด

สำหรับกรณีที่สถานะคงที่มีสัญญาณภายในต่างกันในขั้นตอนวิธีการเติมจุดทดสอบจะมีการใช้เพียงแค่จุดควบคุมเพียงอย่างเดียว ดังนั้นเวลาที่ดีที่สุดคือ 1 ส่วนในกรณีที่ใช้เวลามากที่สุดคือ O(n)

2.3 กรณีวงจรเกิดลูป

ในกรณีนี้จะต้องมีการสร้างแผนภาพคล้ายซิกแนลทรานสิชันกราฟใหม่ ดังนั้นเวลาที่ใช้ก็จะมีค่าเหมือนกับขั้นตอนการสร้างแผนภาพคล้ายซิกแนลทรานสิชันกราฟนั้นคือ 1 ในกรณีที่ดีที่สุดและมีค่าเป็น O(n(o+j)) ในกรณีที่ใช้เวลามากที่สุด โดย n มีค่าเป็นจำนวนเกตของวงจร o มีค่าเป็นจำนวนสัญญาณเอาต์พุตของวงจรและ j มีค่าเป็นจำนวนสัญญาณย้อนกลับของวงจร

5.7 ผลการทดลองและสรุปผลการทดลอง

จากผลการทดลองในตารางที่ 5.1 เมื่อนำขั้นตอนวิธีการเติมจุดทดสอบมาใช้กับวงจรเปรียบเทียบสมรรถนะพบว่าจะต้องเติมจุดทดสอบทั้งสิ้น 36 จุดแบ่งเป็น

จำนวนของจุดควบคุม 0	17	จุด
จำนวนของจุดควบคุม 1	7	จุด
จำนวนจุดสังเกต	12	จุด

จากผลการทดลองในตารางที่ 5.2 จะแสดงรายละเอียดของข้อผิดพร่องที่ทดสอบไม่ได้และรายละเอียดของจุดทดสอบที่เติม ที่ได้จากขั้นตอนวิธีการเติมจุดทดสอบกับวงจรเปรียบเทียบสมรรถนะ ผลปรากฏว่าจำนวนจุดทดสอบที่ใช้ทั้งสิ้นมีจำนวน 36 จุดจากข้อผิดพร่องที่ทดสอบไม่ได้ทั้งสิ้น 41 จุด โดยแบ่งเป็น

จำนวนของข้อผิดพร่องที่ต้องใช้จุดทดสอบ 1 จุด	36	จุด
จำนวนของข้อผิดพร่องที่ต้องใช้จุดทดสอบมากกว่า 1 จุด	5	จุด

จะเห็นได้ว่าจำนวนจุดทดสอบที่ต้องใช้ทั้งสิ้นมีทั้งหมด 46 จุด แต่เป็นจุดที่สามารถใช้ร่วมกันได้ถึง 10 จุดซึ่งคิดเป็น 21.74% จากจำนวนจุดทดสอบที่ต้องใช้

5.8 สรุป

ในบทนี้ได้กล่าวถึงขั้นตอนวิธีการเติมจุดทดสอบที่ได้ออกแบบมา โดยอาศัยการวิเคราะห์พฤติกรรมของวงจรที่เกิดข้อผิดพร่องและแบ่งพฤติกรรมของวงจรที่เกิดข้อผิดพร่องชนิดต่าง ๆ ออกเป็นกลุ่ม ๆ ซึ่งสามารถแบ่งออกมาได้ 3 กลุ่มใหญ่ ๆ คือ เกิด hazard, มีค่าสัญญาณเอาต์พุตของสถานะคงที่ค่าเดียวกันและเกิดลูป จากนั้นก็นำเอาจุดทดสอบมาเติมลงในวงจรเพื่อแก้ไขปัญหาที่เกิดขึ้น

นอกจากนี้ยังได้แสดงผลการทดลองขั้นตอนวิธีการเติมจุดทดสอบกับวงจรเกณฑ์เปรียบเทียบสมรรถนะ ซึ่งจะเห็นว่าเมื่อเพิ่มจุดทดสอบสามารถทำให้วงจรสามารถทดสอบได้ทุกกรณี

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

วงจร	Fault Coverage ก่อนเติมจุดทดสอบ	จำนวนข้อผิดพร่องที่ทดสอบไม่ได้	จำนวนจุดทดสอบ		
			C-0	C-1	Obs
1. celement	91.6667	2	1	1	0
2. converta	64.5833	7	2	1	4
3. ebergen	86.6667	4	2	0	2
6. nowich	96.1538	1	0	1	0
7. trimos-send	83.3333	12	4	1	4
8. vbe5b	95.8333	1	1	0	0
11. vbe10b	96.4286	2	2	0	0
12. wrdata	95.4545	1	1	0	0
13. wrdatab	91.25	7	1	2	2
15. chu133	95.4545	1	1	0	0
21. master-read	98.3333	1	0	1	0
25. sbuf-send-ctl	96.875	1	1	0	0
27. rcv-setup	87.5	1	1	0	0
รวม		41	17	7	12

ตารางที่ 5.1 แสดงการเติมจุดทดสอบโดยใช้ขั้นตอนวิธีการเติมจุดทดสอบ

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

วงจร	สัญญาณ	ประเภทของข้อผิดพร่อง	เติมจุดทดสอบที่สัญญาณ	ประเภทของจุดทดสอบ
1. celement	P3-1	s-a-1	P3	C-0
	P5-2	s-a-1	P3	C-1
2. converta	x2	s-a-1	n2	Obs
	x4	s-a-1	n5	Obs
	x5	s-a-0	n6	Obs
	Ao2	s-a-1	n3	C-0
			n4	Obs
	Ro2	s-a-1	n6	C-0
	Ro3	s-a-0	n4	C-1
	Ro4	s-a-0	n6	Obs
3. ebergen	Ai2	s-a-1	Co	C-1
	Ai3	s-a-0	n3	Obs
	Ai4	s-a-0	n1	C-0
	Xo2	s-a-1	n1	C-0
			n2	Obs
6. nowich	Ci1	s-a-0	n1	C-1
7. trimos-send	Ao1	s-a-0	n4	C-1
	Ao3	s-a-0	n6	Obs
	Ao4	s-a-1	n2	C-0
	Bo1	s-a-0	n6	C-1
	Bo3	s-a-1	n3	C-0
	Bo5	s-a-0	n8	Obs
	Co2	s-a-0	n3	Obs
	Co3	s-a-0	n4	Obs
	Co4	s-a-1	n9	C-0
	T12	s-a-1	n1	C-0
	T21	s-a-1	n2	C-0
	T31	s-a-1	n3	C-0
8. vbe5b	La1	s-a-0	Zr	C-0

ตารางที่ 5.2 ผลการทดลองแสดงจุดทดสอบที่เติมในวงจรเมื่อเกิดข้อผิดพร่องขึ้นในวงจร

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

วงจร	สัญญาณ	ประเภทของ Fault	เติม Test point ที่สัญญาณ	ประเภทของ Test point
11. vbe10b***	A23	s-a-0	X1	C-0
			n1	Obs
	A33	s-a-0	X2	C-0
			n1	Obs
12. wrdata	T2	s-a-1	n1	C-0
13. wrdatab	Ack4	s-a-0	n3	C-1
	ST2	s-a-0	n10	C-1
	ST3	s-a-0	n9	C-0
	X1	s-a-1	n2	Obs
	X5	s-a-0	n10	C-1
	X6	s-a-1	n9	C-0
			n10	Obs
	Y2	s-a-0	n2	Obs
15. chu133	Zr	s-a-0	Lr	C-0
21. master-read**	Bs2	s-a-0	n1	C-1
25. sbuf-send-ctl	Rp1	s-a-0	La	C-0
27. rcv-setup	Sd2	s-a-0	n1	C-0

ตารางที่ 5.2 (ต่อ)

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 6
สรุปผลการวิจัยและข้อเสนอแนะ

งานวิจัยนี้เป็นงานวิจัยเกี่ยวกับการทดสอบวงจรอสมวาร โดยได้ประยุกต์การทดสอบวงจรสมวารมาใช้คือการหาค่าลำดับอินพุตมาเพื่อทดสอบวงจรโดยให้วงจรแสดงข้อผิดพร่องมาทางเอาต์พุตของวงจร แต่กรณีของวงจรอสมวารมีข้อจำกัดคือวงจรไม่มีสัญญาณนาฬิกามาควบคุมให้วงจรอยู่ในสถานะที่จะป้อนอินพุตได้ ดังนั้นจึงทำให้การทดสอบวงจรอสมวารมีความยุ่งยากกว่าการทดสอบวงจรแบบสมวาร

6.1 สรุปผลการวิจัย

งานวิจัยนี้มีวัตถุประสงค์เพื่อออกแบบขั้นตอนวิธีการเติมจุดทดสอบสำหรับวงจรอสมวารแบบควอไซดีเลย์อินเซนซิทีฟ โดยได้มีการดำเนินการตามขั้นตอนหลัก ๆ ต่อไปนี้

1. จำลองพฤติกรรมของวงจรเมื่อวงจรเกิดข้อผิดพร่องโดยได้พัฒนาโปรแกรมมาช่วยในการจำลองพฤติกรรมของวงจร
2. ปรับปรุง Fault Coverage ของวงจรโดยใช้ ESSG มาช่วยในการทดสอบวงจร
3. แบ่งพฤติกรรมของวงจรที่ทดสอบไม่ได้ โดยแบ่งพฤติกรรมเป็น 3 กลุ่มคือ
   3.1 วงจรเกิด hazard
   3.2 วงจรที่เกิดข้อผิดพร่องมีค่าของสัญญาณเอาต์พุตของสถานะคงที่ค่าเดียวกับสถานะคงที่ปกติของวงจร
   3.3 วงจรเกิดลูป
4. ออกแบบขั้นตอนวิธีการเติมจุดทดสอบ และนำไปทดสอบกับวงจรเปรียบเทียบสมรรถนะ

จากขั้นตอนการวิจัยสามารถสรุปผลในแต่ละขั้นตอนดังนี้

1. ซิกแนลทรานสิชั่นกราฟมีข้อจำกัดคือไม่สามารถแสดงพฤติกรรมของวงจรได้อย่างละเอียดทำให้เป็นจำเป็นต้องใช้แผนภาพคล้ายซิกแนลทรานสิชันกราฟช่วยในการแสดงพฤติกรรมของวงจร
2. เพื่อให้จำนวนจุดทดสอบที่ใช้ในวงจรมีจำนวนน้อย ดังนั้นจึงจำเป็นที่จะต้องวิเคราะห์พฤติกรรมของวงจรจากแผนภาพคล้ายซิกแนลทรานสิชันกราฟ เพื่อให้ทราบถึงสัญญาณที่ควรจะเพิ่มจุดทดสอบ

3. การจำลองพฤติกรรมการทำงานของวงจรอสมวารที่เกิดข้อผิดพร่องมีความซับซ้อน ดังนั้นจึงได้พัฒนาโปรแกรมเพื่อช่วยในการจำลองพฤติกรรมของวงจรและได้พบว่า Fault Coverage ที่ได้มีค่ามากกว่า 50 % เนื่องจากข้อผิดพร่องที่ทำให้ค่าเอาต์พุตของเกตใด ๆ ในวงจรมีค่าคงที่สามารถทดสอบได้

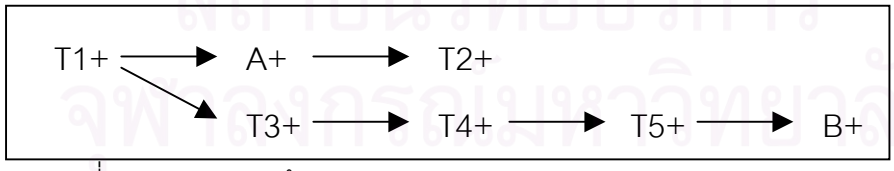
4. การใช้แผนภาพคล้ายซิกแนลทรานสิชันกราฟแสดงพฤติกรรมทั้งหมดจะทำให้กราฟที่ได้มีขนาดใหญ่มากดังนั้นจึงต้องใช้ SSG มาช่วยแสดงพฤติกรรมของวงจรขณะที่ข้อผิดพร่องยังไม่มีผลกระทบกับวงจร

5. พฤติกรรมของวงจรอสมวารที่เกิดข้อผิดพร่องที่พบได้มากที่สุดคือข้อ 2.1 รูปแบบที่ 2.2 จะพบได้น้อยส่วนรูปแบบที่ 2.3 มักจะไม่ค่อยพบ

6. เมื่อนำเอาขั้นตอนวิธีการเติมจุดทดสอบไปทดสอบกับวงจรเปรียบเทียบสมรรถนะสามารถทำให้วงจรทดสอบได้ทุกข้อผิดพร่องและมีค่า Fault Coverage เป็น 100 %

6.2 ข้อเสนอแนะ

เราพบว่าพฤติกรรมของวงจรที่เกิดข้อผิดพร่องขึ้นและไม่สามารถทดสอบได้คือเกิด hazard ซึ่งเกิดจากการเกิดการเปลี่ยนสัญญาณอินพุตที่เป็นในทิศทางตรงกันข้ามกัน (non-monotonous) ในการเปลี่ยนแปลงของคู่สัญญาณนี้อาจจะเกิดช้ามากหรืออาจไม่เกิดขึ้นเลยในการใช้งานจริง เช่นการเปลี่ยนแปลงสัญญาณ A+ และ B+ เป็นการเปลี่ยนแปลงสัญญาณอินพุตที่เป็นในทิศทางตรงข้าม แต่จากแผนภาพคล้ายซิกแนลทรานสิชันกราฟพบว่าการเปลี่ยนแปลงสัญญาณ A+ เกิดก่อนมากจนแน่ใจว่าสัญญาณเอาต์พุตเกิดการเปลี่ยนแปลงเสร็จสิ้นแล้ว และการเปลี่ยนแปลงสัญญาณ B+ นั้นไม่มีผลกับสัญญาณเอาต์พุต พฤติกรรมนั้น ๆ ดังรูป 6.1 ในกรณีเช่นนี้เราอาจจะพิจารณาไม่ต้องเติมจุดทดสอบสำหรับกรณีนี้ เพื่อให้วงจรมีจุดทดสอบที่น้อยลง



รูปที่ 6.1 แผนภาพคล้ายซิกแนลทรานสิชันกราฟ

รายการอ้างอิง

[1] Abramovici M.,Breuer A.,M.,and Friedman D.A. Digital Systems Testing and Testable Design. U.S.A.:Computer Science Press, 1990.

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

[3] Hauck S. Asynchronous Design Methodologies: An Overview. Proceeding of the IEEE 1995:69-93.

[4] Lada P.K. Fault Tolerant and Fault Testable hardware design. Prentice hall, 1985

[5] Martin A.J. and Hazewindus P.J. Testing Delay-Insensitive Circuit. Advance Research in VLSI UC Santa Cruz 1991.

[6] Micheli G.D. Synthesis and Optimization of digital circuit. McGraw-hill, 1994

[7] Nanya T.,Ueno Y.,Kagotani H.,Kuwako M. and Takamura A. TITAC : Design of a Quasi-Delay-Insensitive Microprocessor. IEEE Design & Test of Computers 1994:50-63

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

[9] Thongtak A. Testing Asynchronous Circuit. Doctoral dissertation, Tokyo Institute of Technology, March 1996.

[10] Kuwako M. and Nanya T. Timing-Reliability Evaluation of Asynchronous Circuits Based on Different Delay Models. Proceedings of the International Symposium on Advanced Reserch in Asynchronous Circuits and Systems 1994:22-31.

[11] Eunjung Oh,Soo-Hyun Kim,Ho-Yonh Choi and Dong-Ik Lee. High Level Test Generation for Asynchronous Circuits using Signal Transition Graph. Concurrent system research Lab .

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

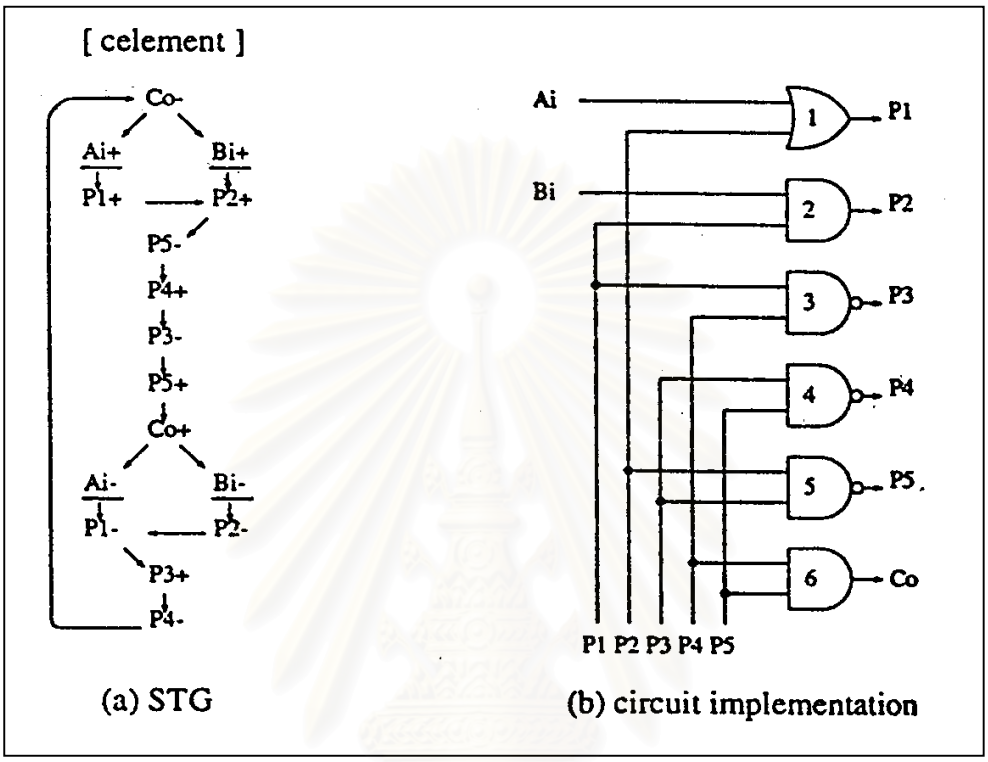
[13] Nur A. Touba and Edward J. McCluskey Test Point Insertion Based on Path Tracing. Transactions on Computer-Aided Design. 1996.

[14] Uppaluri P.,Sparmann U. and Pomeranz I. On Minimizing the Number of Test Points Needed to Achieve Complete Robust Path Delay Fault Testability. University of Iowa 1995.

ภาคผนวก ก.

วงจรเปรียบเทียบสมรรถนะและแผนภาพคล้ายซิกแนลทรานสิชันกราฟแสดงพฤติกรรมของวงจรที่เกิดข้อผิดพร่องที่ไม่สามารถทดสอบได้

1. วงจร celement



State Encoder Ai Bi P1 P2 P3 P4 P5 Co

สถานะเริ่มต้น 0 0 0 0 1 0 1 0

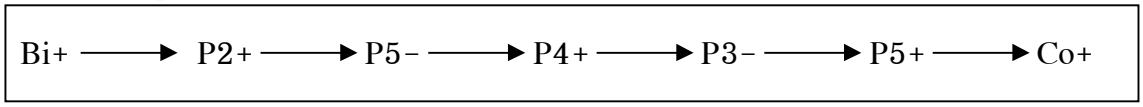
จำนวนสถานะคงที่ 6

1.1 P31 s-a-1

สถานะคงที่ต้นทาง 1 0 1 0 1 0 1 0

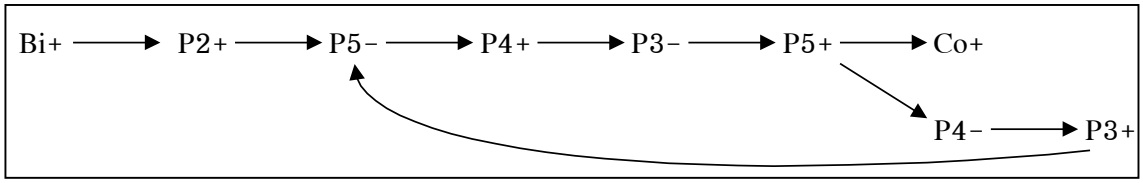
สถานะคงที่ปลายทาง 1 1 1 1 0 1 1 1

ลำดับอินพุต Bi+



- พฤติกรรมปกติ

- พฤติกรรมเมื่อเกิดข้อผิดพร่อง

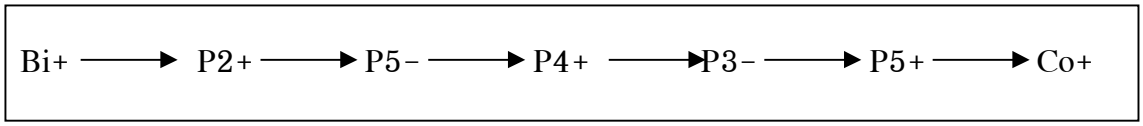


1.2 P52 s-a-1

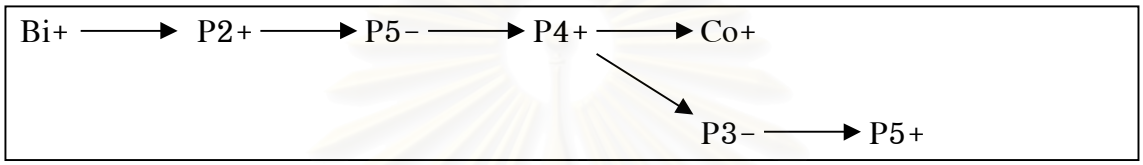
สถานะคงที่ต้นทาง 1 0 1 0 1 0 1 0

สถานะคงที่ปลายทาง 1 1 1 1 0 1 1 1

❑ พฤติกรรมปกติ

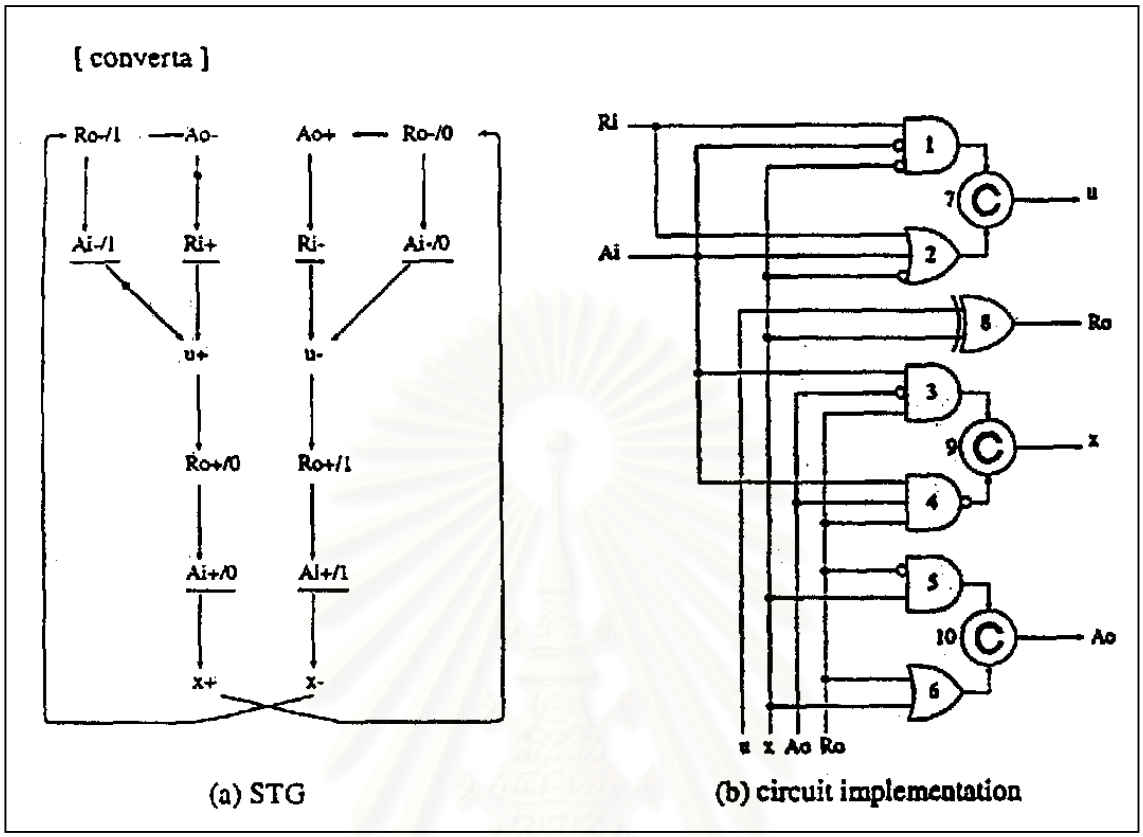


❑ พฤติกรรมเมื่อเกิดข้อผิดพร่อง



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

2. วงจร converta



State Encoder Ri Ai u x Ao Ro

สถานะเริ่มต้น 0 0 0 0 0 0

จำนวนสถานะคงที่ 8

2.1 x2 s-a-1

สถานะคงที่ต้นทาง 0 0 0 0 0 0

สถานะคงที่ปลายทาง ไม่มี

ลำดับอินพุต ไม่มี

- พฤติกรรมปกติ

  สถานะเริ่มต้น

- พฤติกรรมเมื่อเกิดข้อผิดพร่อง

n2 –

2.2 x4 s-a-1

สถานะคงที่ต้นทาง 0 0 0 0 0 0

สถานะคงที่ปลายทาง ไม่มี

ลำดับอินพุต ไม่มี

- พฤติกรรมปกติ

  สถานะเริ่มต้น

- พฤติกรรมเมื่อเกิดข้อผิดพร่อง

n5 –

2.3 x5 s-a-0

สถานะคงที่ต้นทาง 0 0 0 0 0 0 0

สถานะคงที่ปลายทาง ไม่มี

ลำดับอินพุต ไม่มี

- พฤติกรรมปกติ

  สถานะเริ่มต้น

- พฤติกรรมเมื่อเกิดข้อผิดพร่อง

n6 –

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

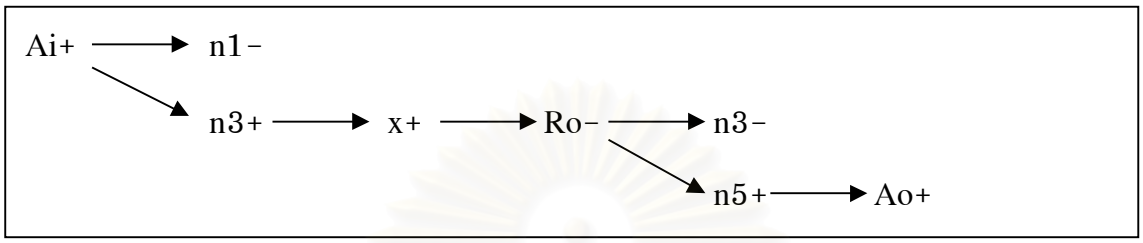
2.4 Ao2 s-a-1

สถานะคงที่ต้นทาง 1 0 1 0 0 1

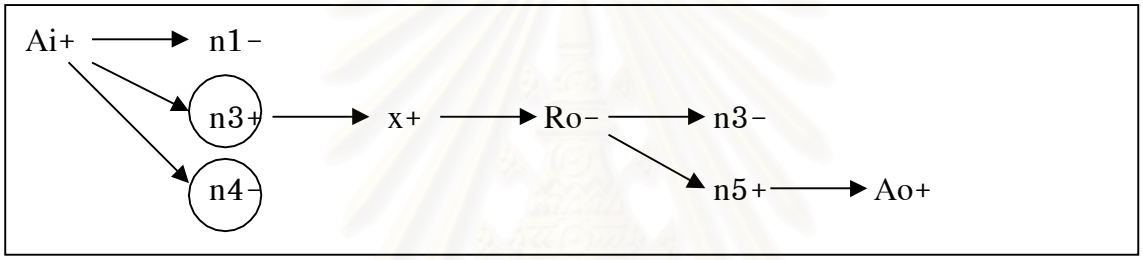
สถานะคงที่ปลายทาง 1 1 1 1 1 0

ลำดับอินพุต Ai+

❑ พฤติกรรมปกติ



❑ พฤติกรรมเมื่อเกิดข้อผิดพร่อง



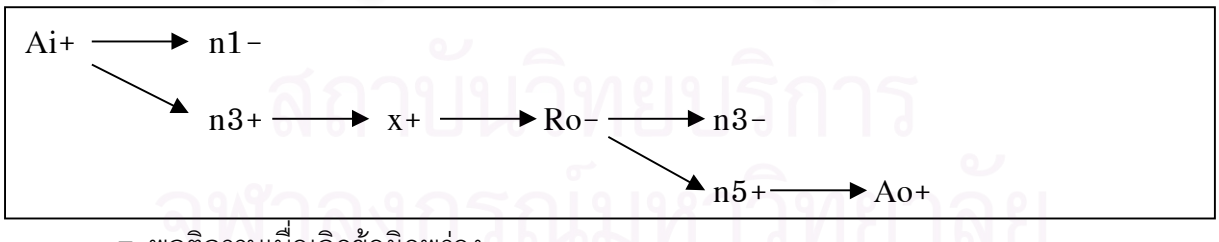
2.5 Ro2 s-a-1

สถานะคงที่ต้นทาง 1 0 1 0 0 1

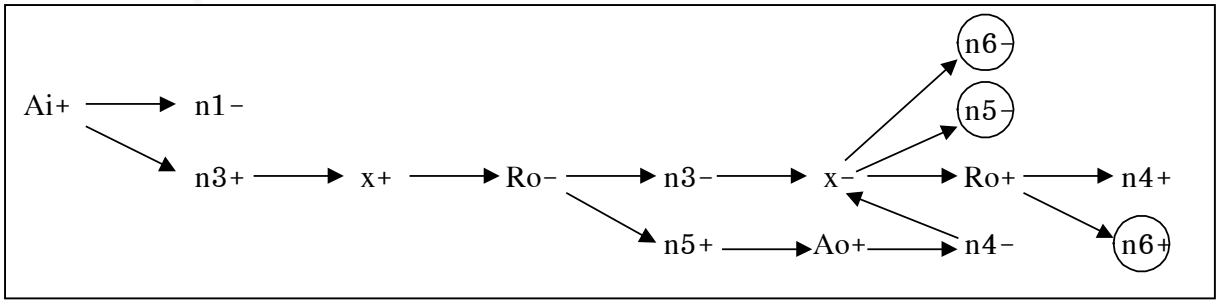
สถานะคงที่ปลายทาง 1 1 1 1 1 0

ลำดับอินพุต Ai+

❑ พฤติกรรมปกติ



❑ พฤติกรรมเมื่อเกิดข้อผิดพร่อง



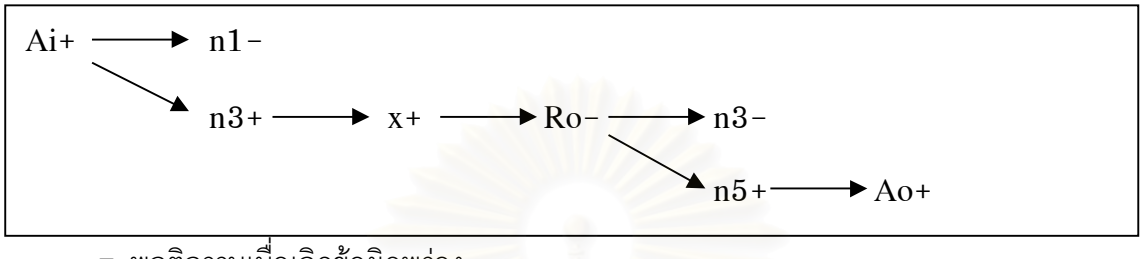
2.6 Ro3 s-a-0

สถานะคงที่ต้นทาง 1 0 1 0 0 1

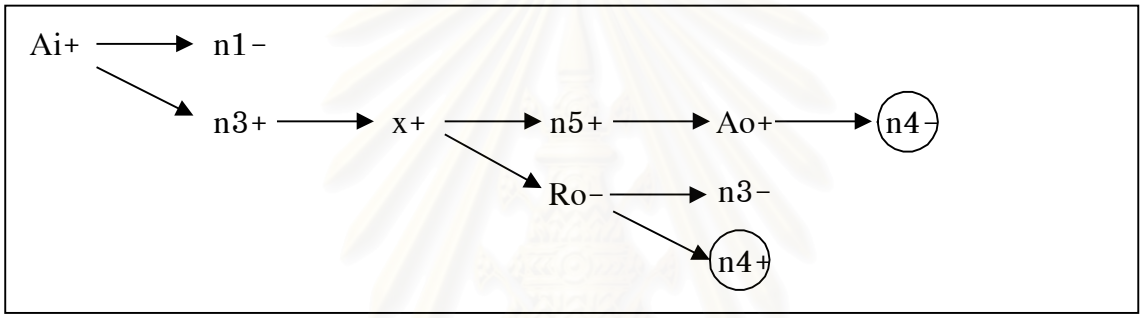
สถานะคงที่ปลายทาง 1 1 1 1 1 0

ลำดับอินพุต Ai+

❑ พฤติกรรมปกติ



❑ พฤติกรรมเมื่อเกิดข้อผิดพร่อง



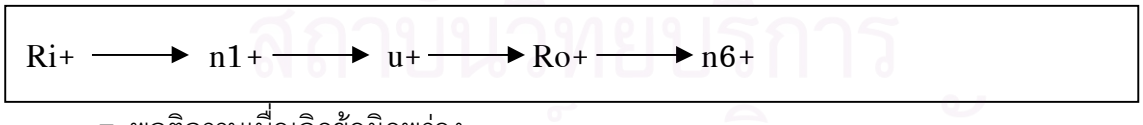
2.7 Ro4 s-a-0

สถานะคงที่ต้นทาง 0 0 0 0 0 0

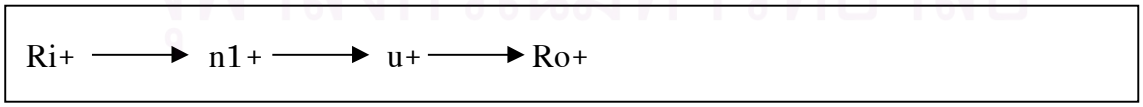
สถานะคงที่ปลายทาง 1 0 1 0 0 1

ลำดับอินพุต Ri+

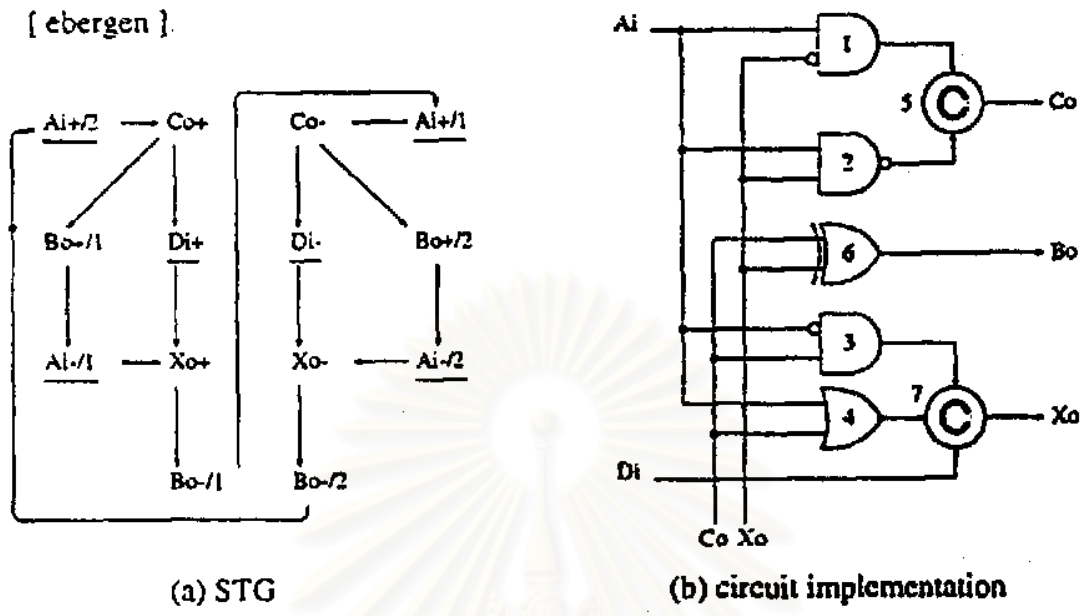
❑ พฤติกรรมปกติ



❑ พฤติกรรมเมื่อเกิดข้อผิดพร่อง



3. วงจร ebergen



State Encoder Ai Di Co Xo Bo

สถานะเริ่มต้น 0 0 0 0 0

จำนวนสถานะคงที่ 8

3.1 Ai2 s-a-1

สถานะคงที่ต้นทาง 0 0 1 0 1

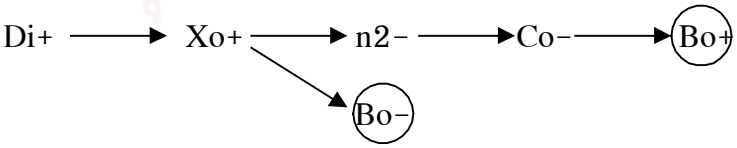
สถานะคงที่ปลายทาง 0 1 1 1 0

ลำดับอินพุต Di+

- พฤติกรรมปกติ

Di+ ⟶ Xo+ ⟶ Bo−

- พฤติกรรมเมื่อเกิดข้อผิดพร่อง



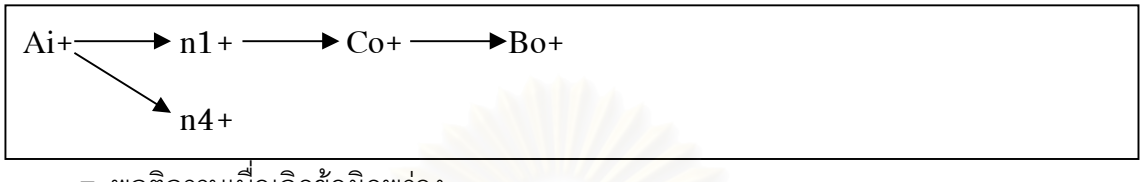
3.2 Ai3 s-a-0

สถานะคงที่ต้นทาง 0 0 0 0 0

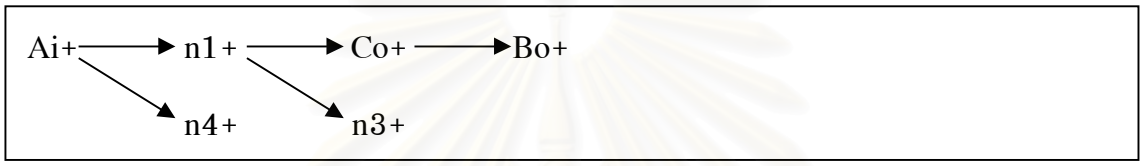
สถานะคงที่ปลายทาง 1 0 1 0 1

ลำดับอินพุต Ai+

- พฤติกรรมปกติ



- พฤติกรรมเมื่อเกิดข้อผิดพร่อง



3.3 Ai4 s-a-0

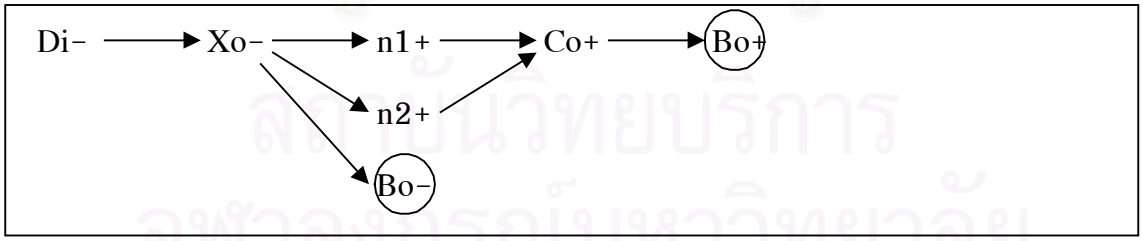
สถานะคงที่ต้นทาง 1 1 0 1 1

สถานะคงที่ปลายทาง 1 0 0 1 1

ลำดับอินพุต Di-

- พฤติกรรมปกติ

- พฤติกรรมเมื่อเกิดข้อผิดพร่อง



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

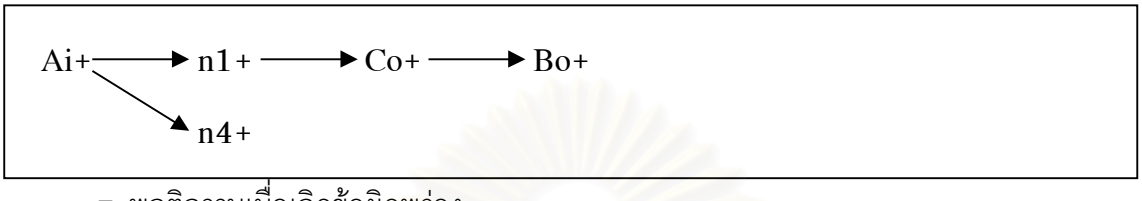
3.4 Xo2 s-a-1

สถานะคงที่ต้นทาง 0 0 0 0 0

สถานะคงที่ปลายทาง 1 0 1 0 1

ลำดับอินพุต Ai+

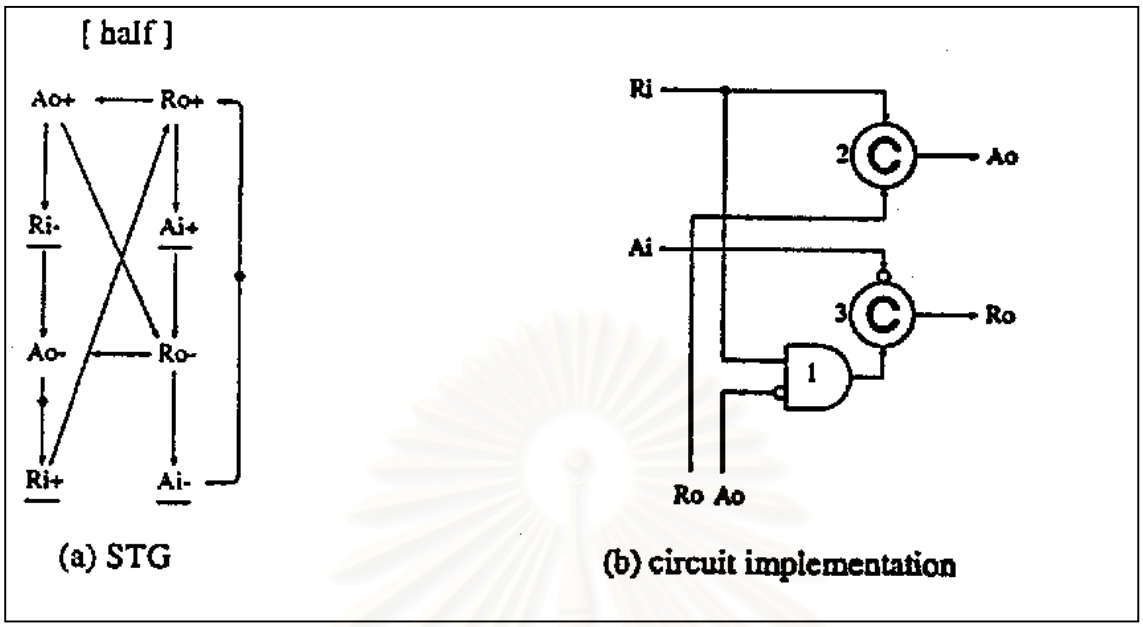
- พฤติกรรมปกติ



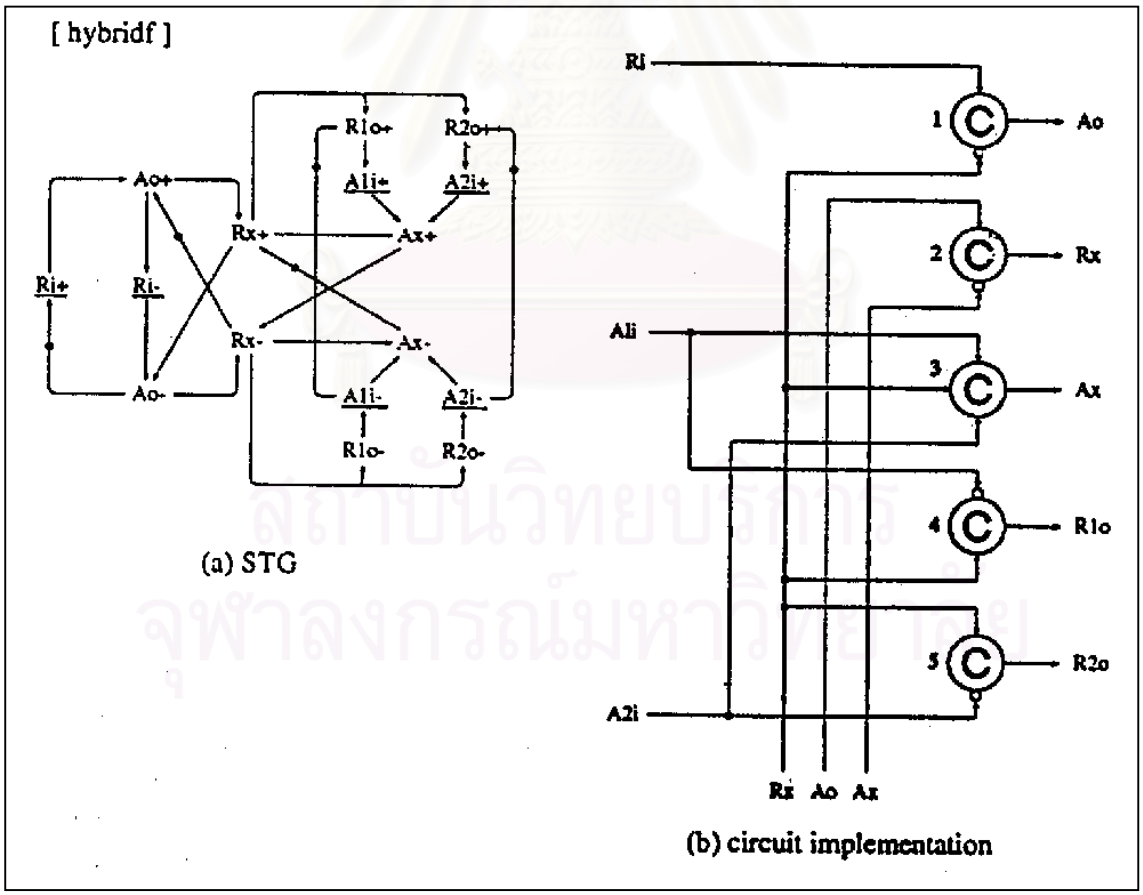
- พฤติกรรมเมื่อเกิดข้อผิดพร่อง

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

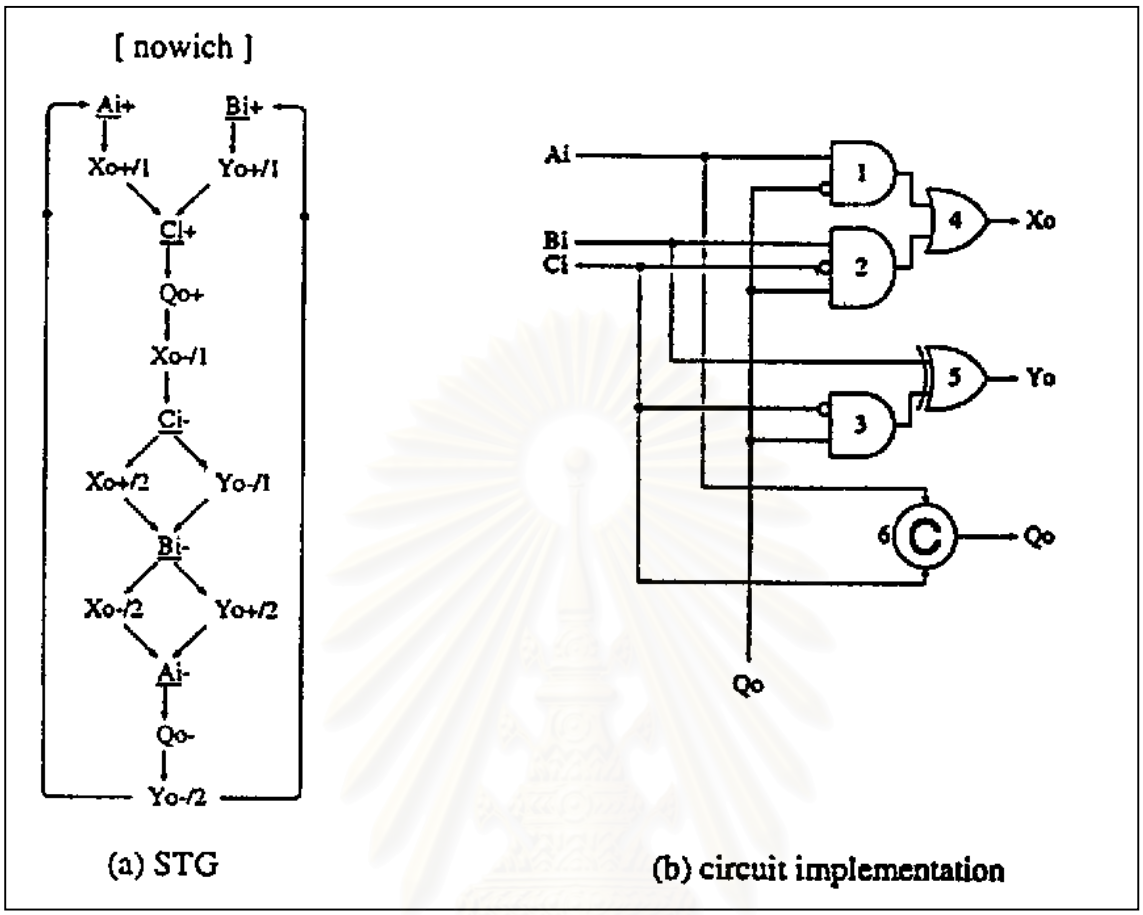
4. วงจร half



5. วงจร hybridf



6. วงจร nowich



State Encoder Ai Bi Ci Qo X0 Yo

สถานะเริ่มต้น 0 0 0 0 0 0

จำนวนสถานะคงที่ 7

6.1 Ci1 s-a-0

สถานะคงที่ต้นทาง 1 1 0 0 1 1

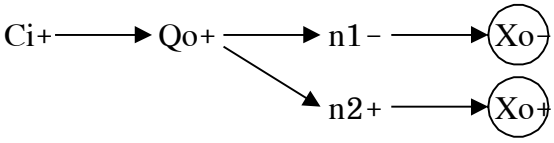
สถานะคงที่ปลายทาง 1 1 1 1 0 1

ลำดับอินพุต Ci+

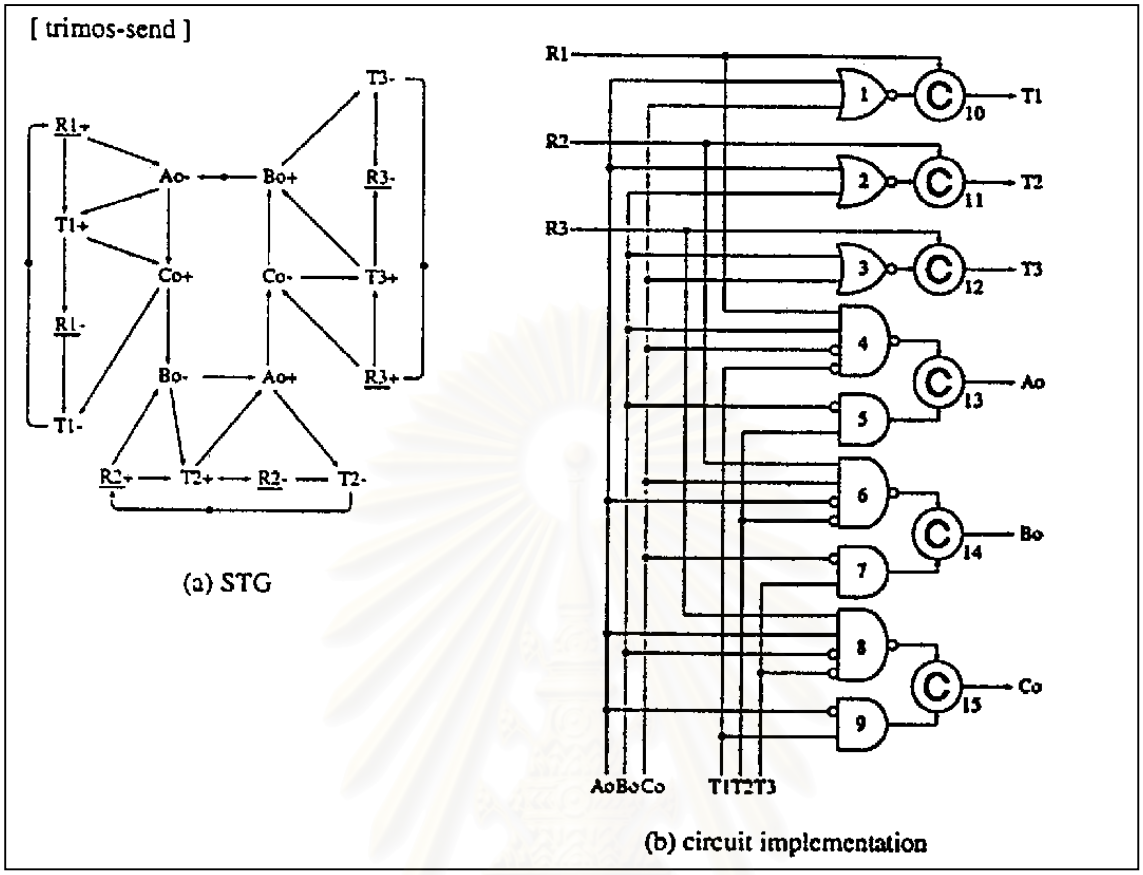
- พฤติกรรมปกติ

Ci+ ⟶ Qo+ ⟶ n1− ⟶ Xo−

- พฤติกรรมเมื่อเกิดข้อผิดพร่อง



7. วงจร trimos-send



State Encoder R1 R2 R3 Ao Bo Co T1 T2 T3

สถานะเริ่มต้น 0 0 0 1 1 0 0 0 0

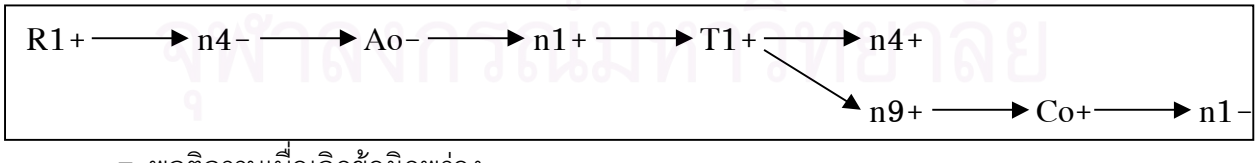
7.1 Ao1 s-a-0

สถานะคงที่ต้นทาง 0 0 0 1 1 0 0 0 0

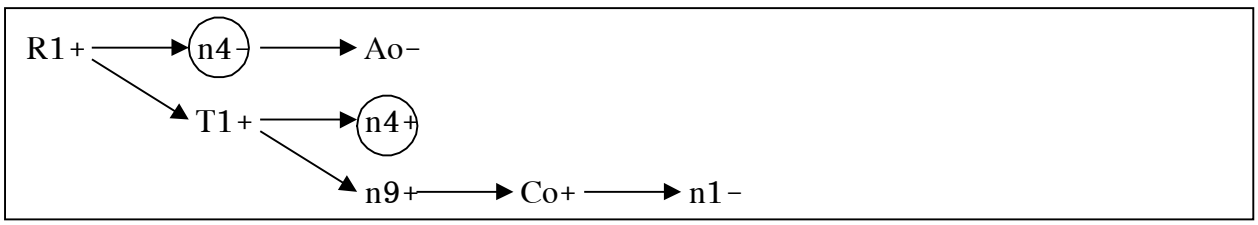
สถานะคงที่ปลายทาง 1 0 0 0 1 1 1 0 0

ลำดับอินพุต Ai+

- พฤติกรรมปกติ



- พฤติกรรมเมื่อเกิดข้อผิดพร่อง



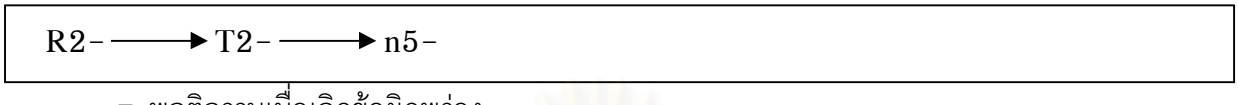
7.2 Ao3 s-a-0

สถานะคงที่ต้นทาง 0 1 0 1 0 1 0 1 0

สถานะคงที่ปลายทาง 0 0 0 1 0 1 0 0 0

ลำดับอินพุต R2-

❑ พฤติกรรมปกติ



❑ พฤติกรรมเมื่อเกิดข้อผิดพร่อง

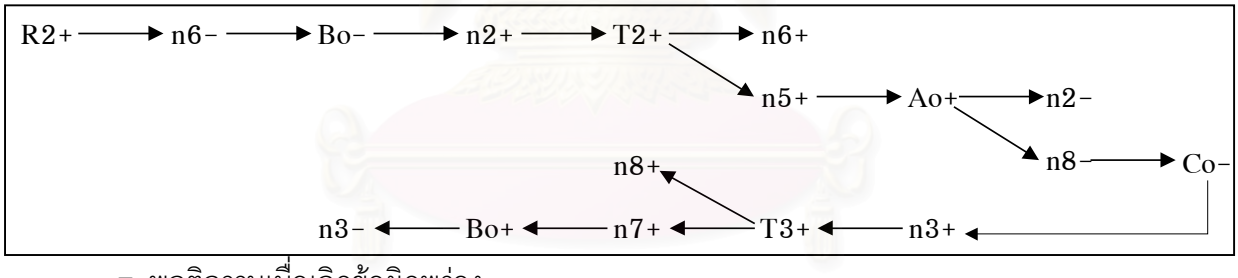
7.3 Ao4 s-a-1

สถานะคงที่ต้นทาง 0 0 1 0 1 1 0 0 0

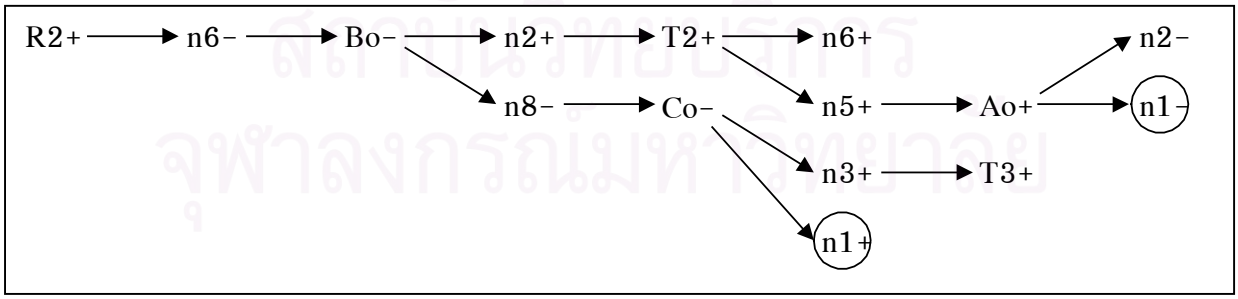
สถานะคงที่ปลายทาง 0 1 1 1 1 0 0 1 1

ลำดับอินพุต R2+

❑ พฤติกรรมปกติ



❑ พฤติกรรมเมื่อเกิดข้อผิดพร่อง



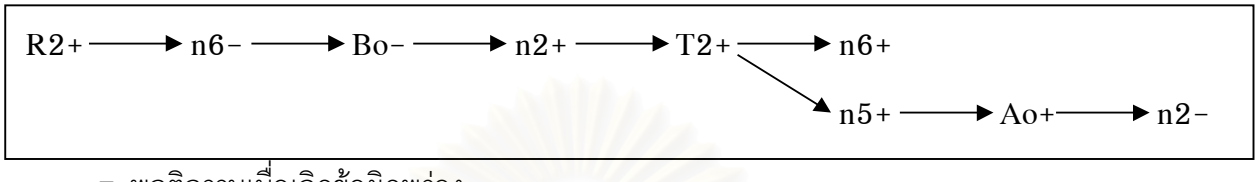
7.4 Bo1 s-a-0

สถานะคงที่ต้นทาง 0 0 0 0 1 1 0 0 0

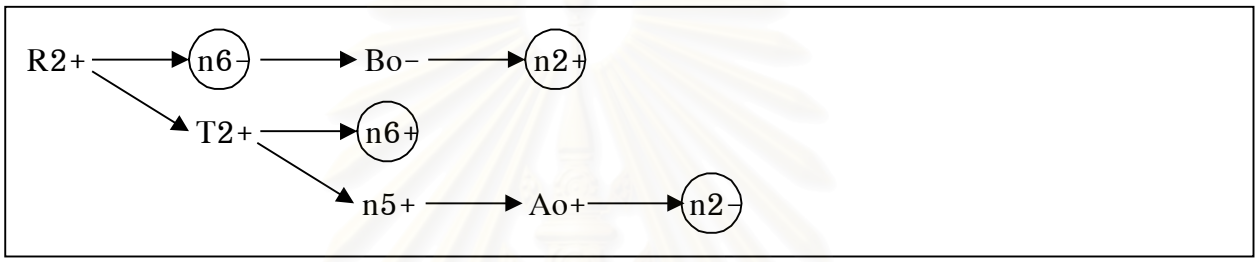
สถานะคงที่ปลายทาง 0 1 0 1 0 1 0 1 0

ลำดับอินพุต R2+

❑ พฤติกรรมปกติ



❑ พฤติกรรมเมื่อเกิดข้อผิดพร่อง



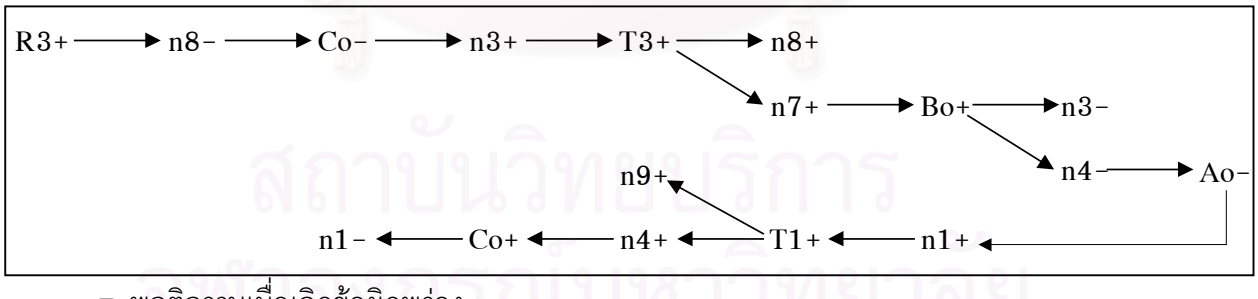
7.5 Bo3 s-a-1

สถานะคงที่ต้นทาง 1 0 0 1 0 1 0 0 0

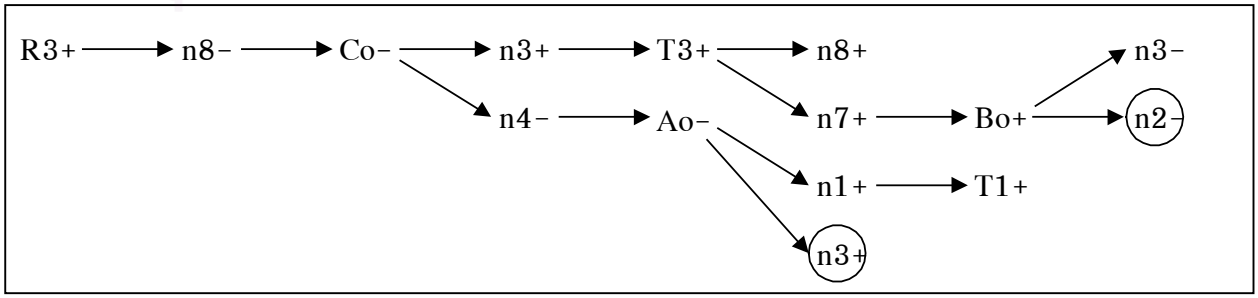
สถานะคงที่ปลายทาง 1 0 1 0 1 1 1 0 1

ลำดับอินพุต R3+

❑ พฤติกรรมปกติ



❑ พฤติกรรมเมื่อเกิดข้อผิดพร่อง



7.6 Bo5 s-a-0

สถานะคงที่ต้นทาง 0 0 0 1 1 0 0 0 0

สถานะคงที่ปลายทาง 0 0 1 1 1 0 0 0 0

ลำดับอินพุต R3+

- พฤติกรรมปกติ

- พฤติกรรมเมื่อเกิดข้อผิดพร่อง

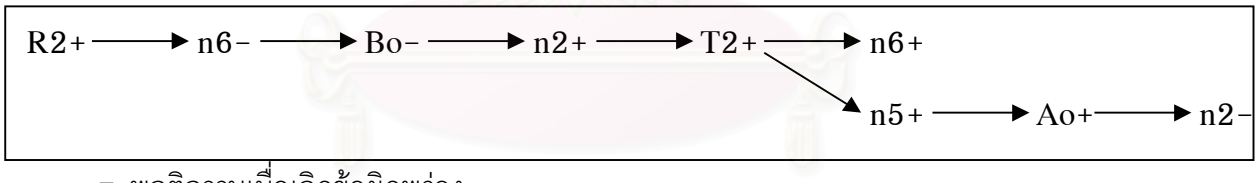
7.7 Co2 s-a-0

สถานะคงที่ต้นทาง 0 0 0 0 1 1 0 0 0

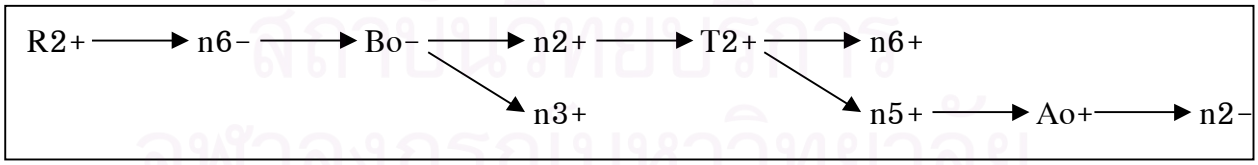
สถานะคงที่ปลายทาง 0 1 0 1 0 1 0 1 0

ลำดับอินพุต R2+

- พฤติกรรมปกติ



- พฤติกรรมเมื่อเกิดข้อผิดพร่อง



7.8 Co3 s-a-0

สถานะคงที่ต้นทาง 0 0 0 0 1 1 0 0 0

สถานะคงที่ปลายทาง 1 0 0 0 1 1 0 0 0

ลำดับอินพุต R1+

- พฤติกรรมปกติ

R1+

- พฤติกรรมเมื่อเกิดข้อผิดพร่อง

R1+ → n4−

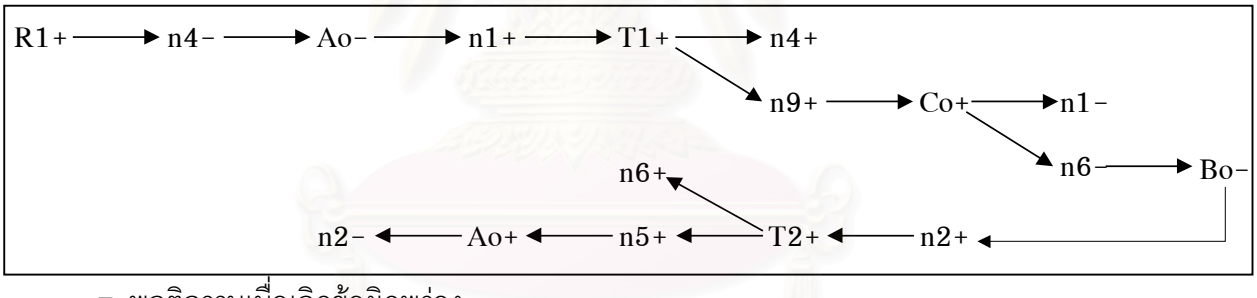
7.9 Co4 s-a-1

สถานะคงที่ต้นทาง 0 1 0 1 1 0 0 0 0

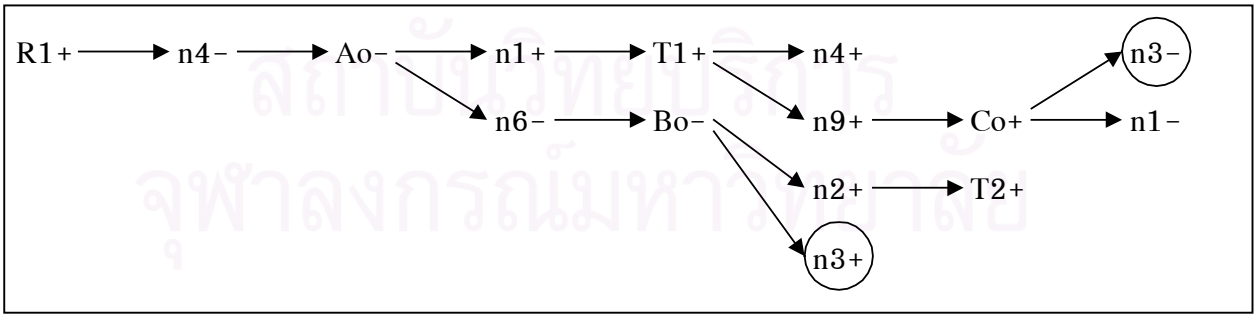
สถานะคงที่ปลายทาง 1 1 0 1 0 1 1 1 0

ลำดับอินพุต R1+

- พฤติกรรมปกติ



- พฤติกรรมเมื่อเกิดข้อผิดพร่อง



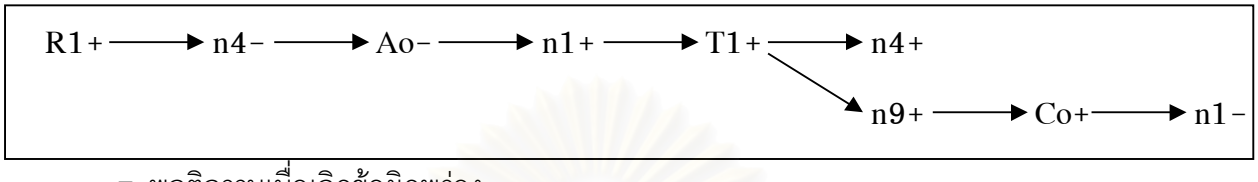
7.10 T12 s-a-1

สถานะคงที่ต้นทาง 0 0 0 1 1 0 0 0 0

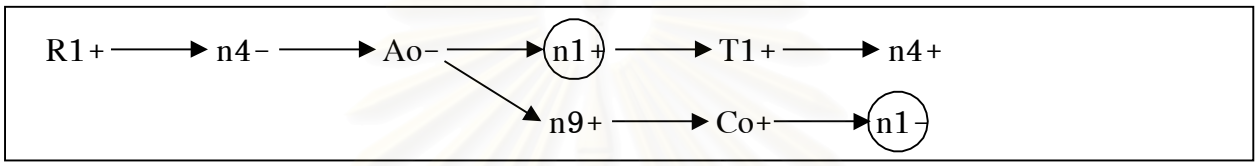
สถานะคงที่ปลายทาง 1 0 0 0 1 1 1 0 0

ลำดับอินพุต R1+

- พฤติกรรมปกติ



- พฤติกรรมเมื่อเกิดข้อผิดพร่อง



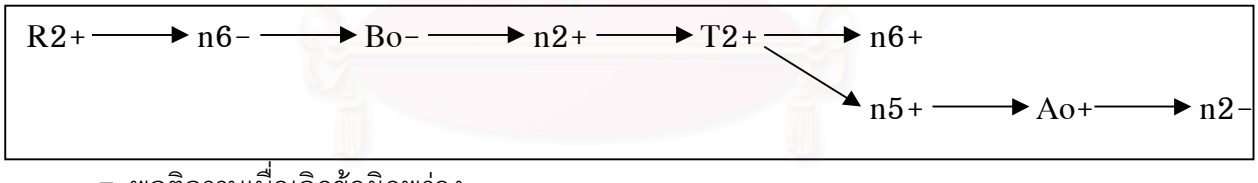
7.11 T21 s-a-1

สถานะคงที่ต้นทาง 0 0 0 0 1 1 0 0 0

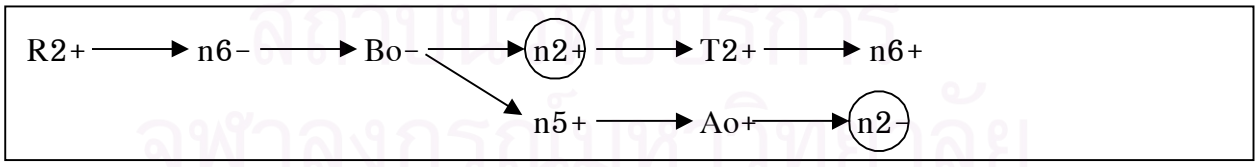
สถานะคงที่ปลายทาง 0 1 0 1 0 1 0 1 0

ลำดับอินพุต R2+

- พฤติกรรมปกติ



- พฤติกรรมเมื่อเกิดข้อผิดพร่อง



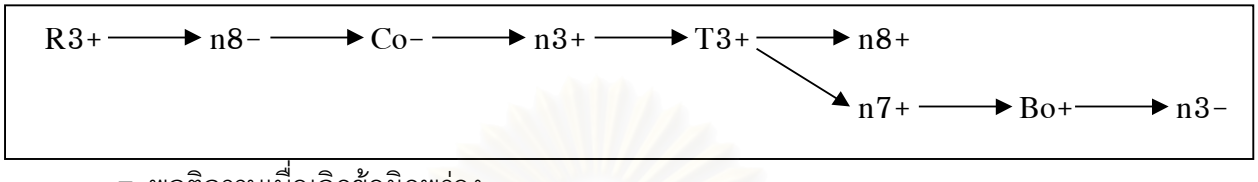
7.12 T31 s-a-1

สถานะคงที่ต้นทาง 0 0 0 1 0 1 0 0 0

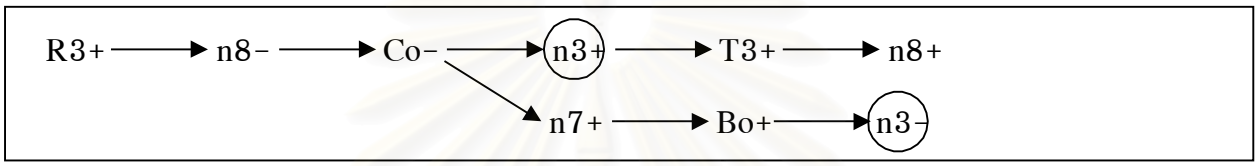
สถานะคงที่ปลายทาง 0 0 1 1 1 0 0 0 1

ลำดับอินพุต R3+

- พฤติกรรมปกติ

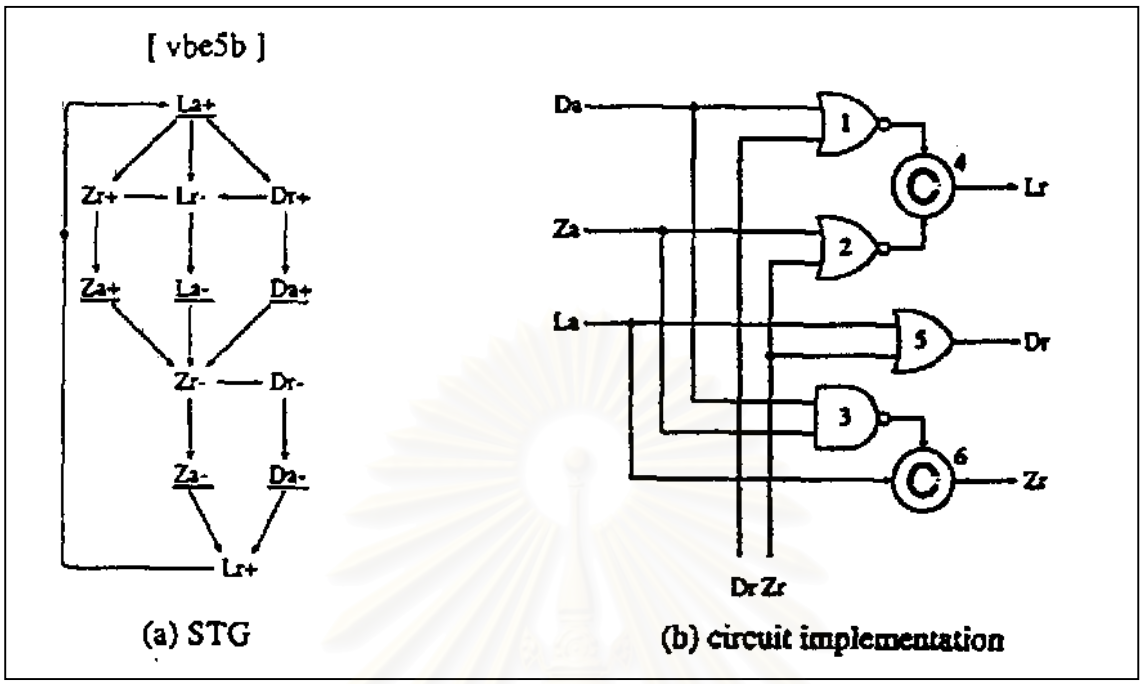


- พฤติกรรมเมื่อเกิดข้อผิดพร่อง



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

8. วงจร vbe5b



State Encoder Da Za La Dr Zr Lr

สถานะเริ่มต้น 0 0 0 0 0 1

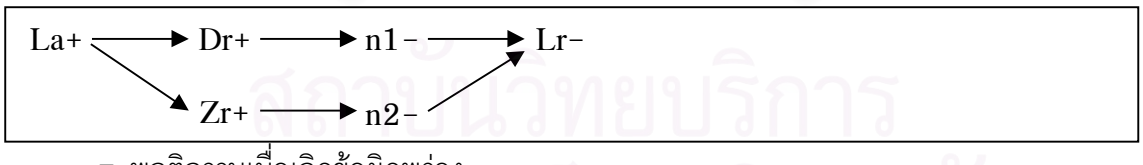
8.1 La1 s-a-0

สถานะคงที่ต้นทาง 0 0 0 0 0 1

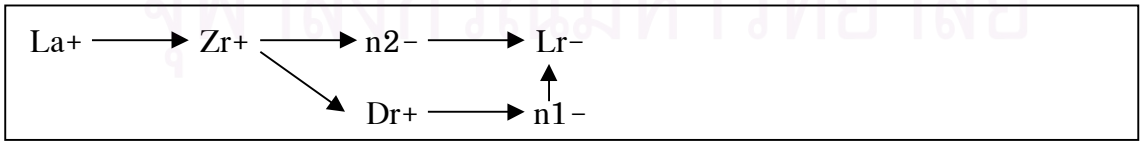
สถานะคงที่ปลายทาง 0 0 1 1 1 0

ลำดับอินพุต La+

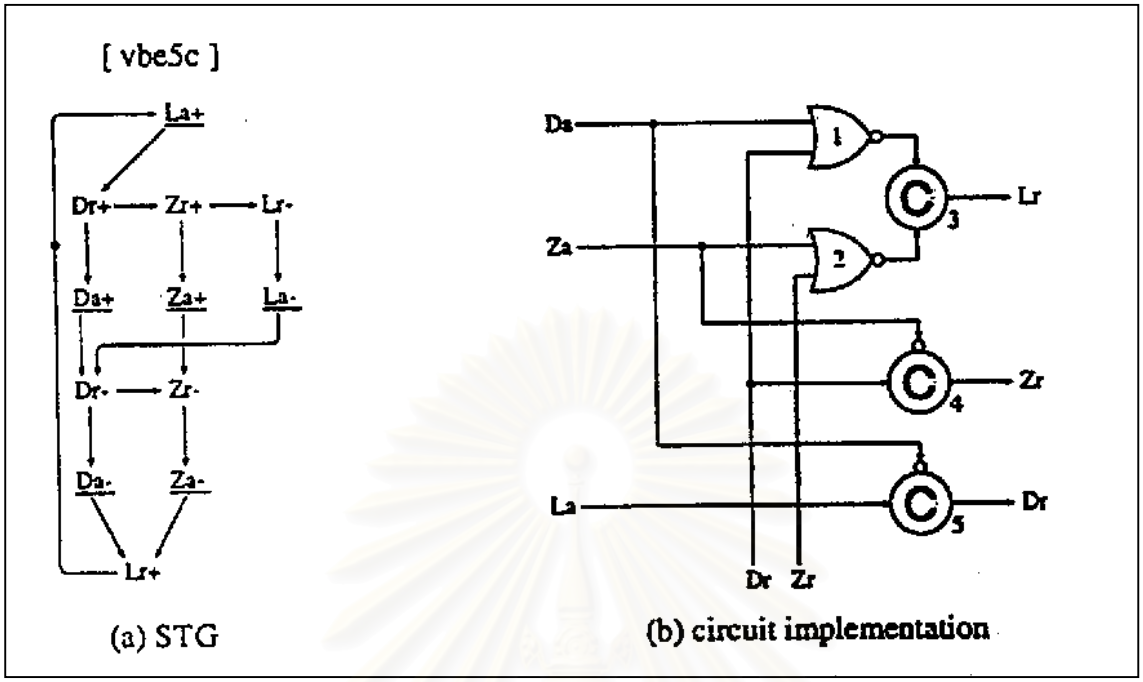
- พฤติกรรมปกติ



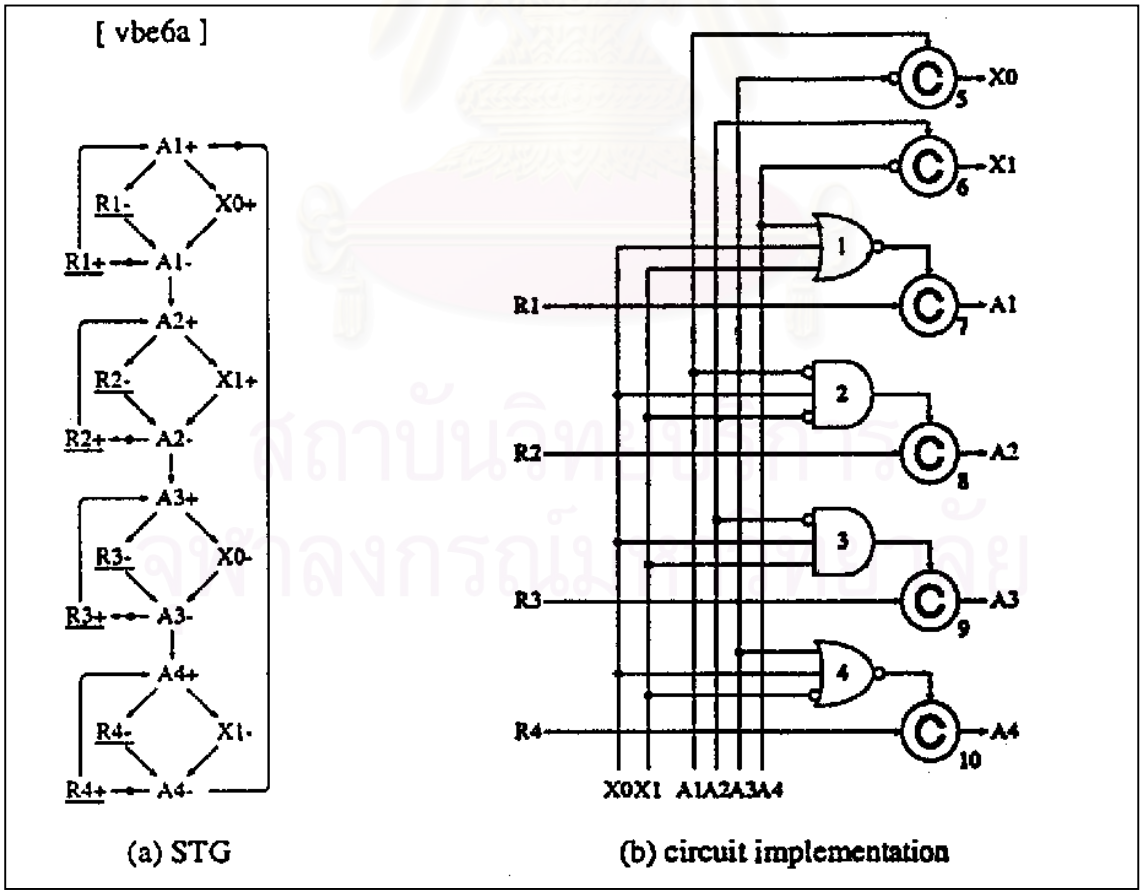
- พฤติกรรมเมื่อเกิดข้อผิดพร่อง



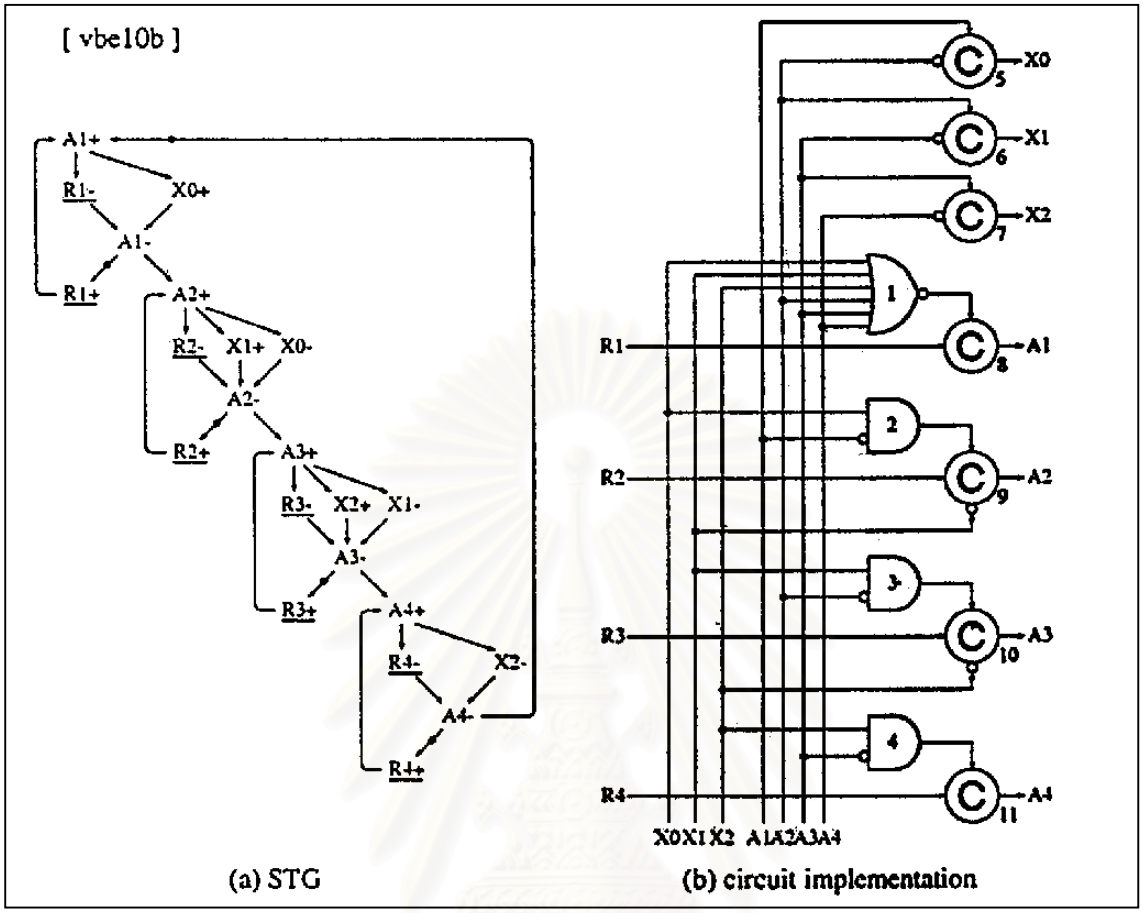
9. วงจร vbe5c



10. วงจร vbe6a



11. วงจร vbe10b



State Encoder R1 R2 R3 R4 X0 X1 X2 A1 A2 A3 A4

สถานะเริ่มต้น 0 0 0 0 0 0 0 0 0 0 0

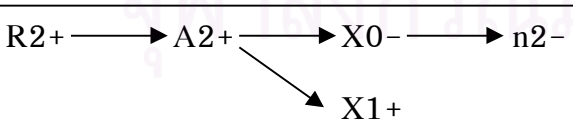
11.1 A23 s-a-0

สถานะคงที่ต้นทาง 0 0 0 0 1 0 0 0 0 0 0

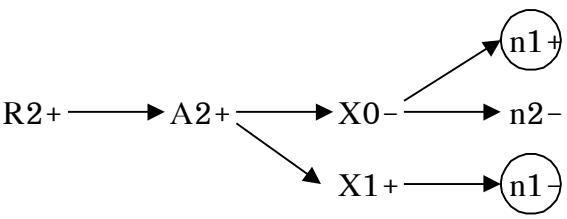
สถานะคงที่ปลายทาง 0 1 0 0 0 1 0 0 1 0 0

ลำดับอินพุต R2+

- พฤติกรรมปกติ



- พฤติกรรมเมื่อเกิดข้อผิดพร่อง



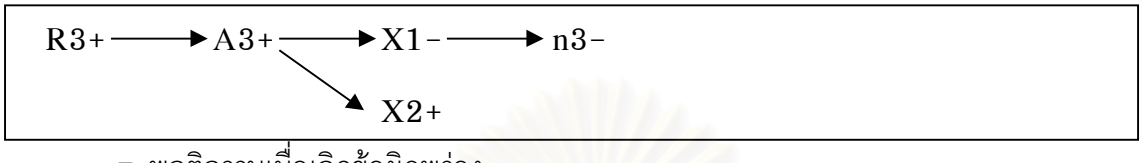
11.2 A33 s-a-0

สถานะคงที่ต้นทาง 0 0 0 0 0 1 0 0 0 0 0

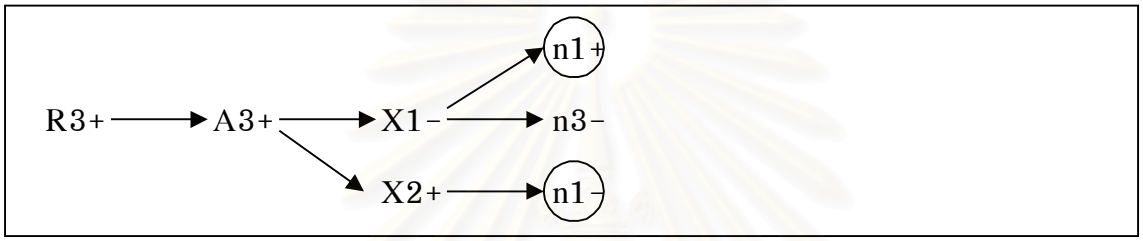
สถานะคงที่ปลายทาง 0 0 1 0 0 0 1 0 0 1 0

ลำดับอินพุต R3+

- พฤติกรรมปกติ

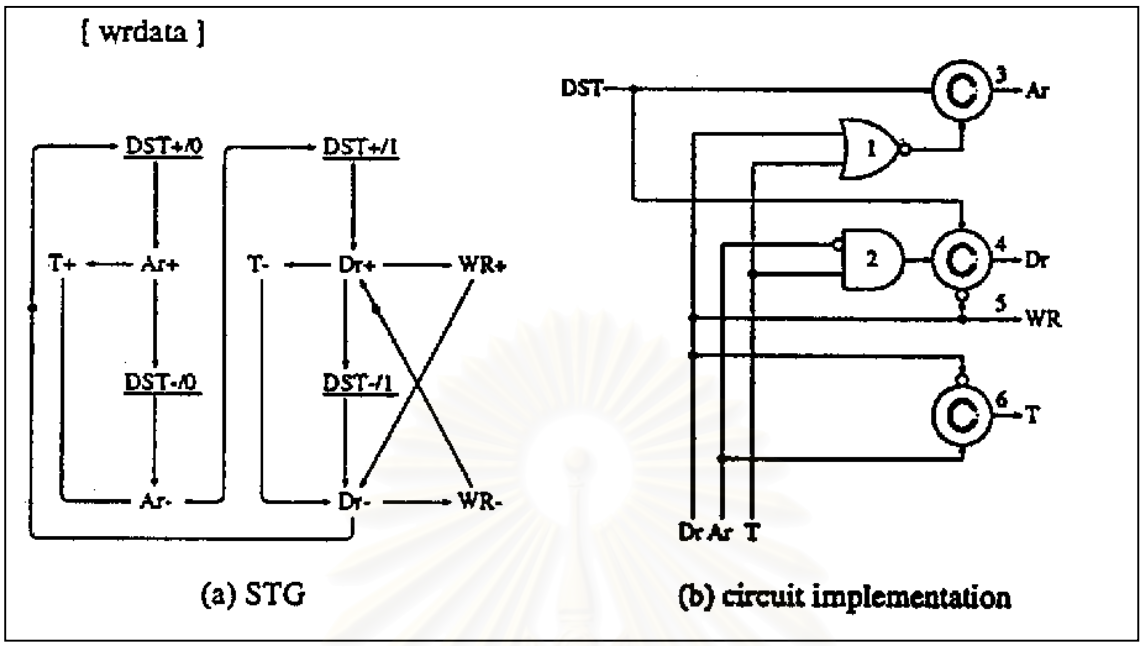


- พฤติกรรมเมื่อเกิดข้อผิดพร่อง



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

12. วงจร wrdata



State Encoder DST Dr Ar T WR

สถานะเริ่มต้น 0 0 0 0 0

จำนวนสถานะคงที่ 4

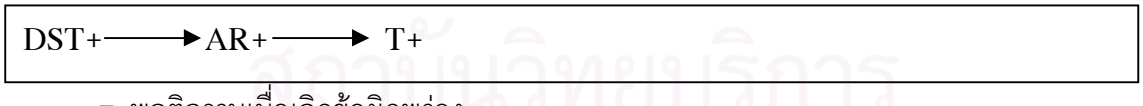
12.1 T2 s-a-1

สถานะคงที่ต้นทาง 0 0 0 0 0

สถานะคงที่ปลายทาง 1 0 1 1 0

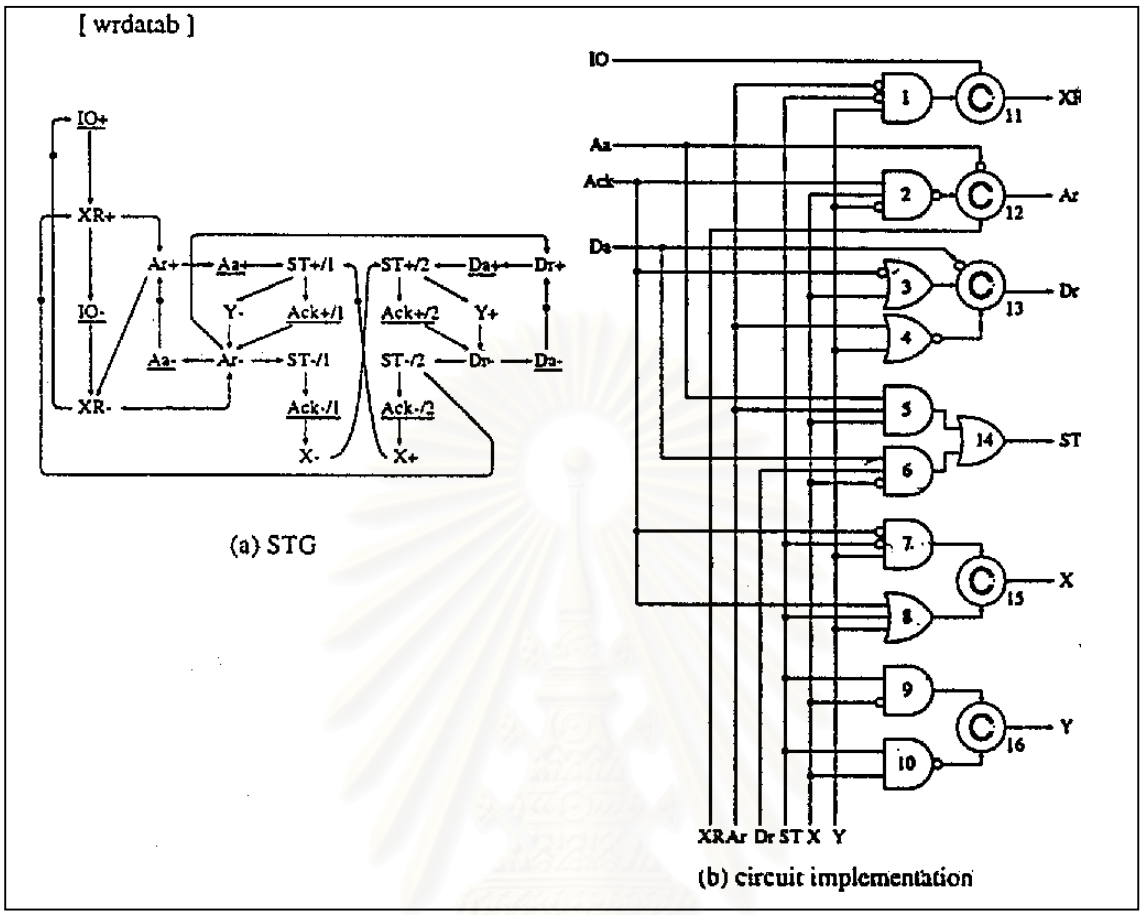
ลำดับอินพุต DST+

- พฤติกรรมปกติ



- พฤติกรรมเมื่อเกิดข้อผิดพร่อง

13. วงจร wrdatab



State Encoder IO Aa Ack Da XR Ar Dr ST X Y

สถานะเริ่มต้น 0 0 0 0 0 0 0 0 0 1 1

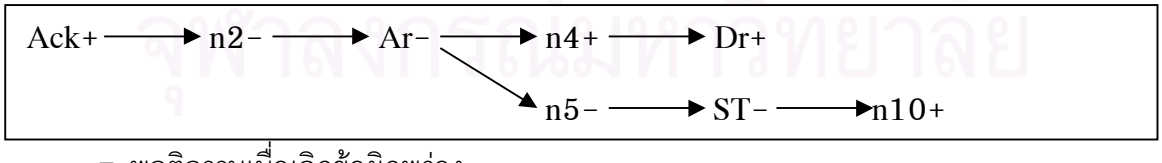
13.1 Ack4 s-a-0

สถานะคงที่ต้นทาง 1 1 0 0 1 1 0 1 1 0

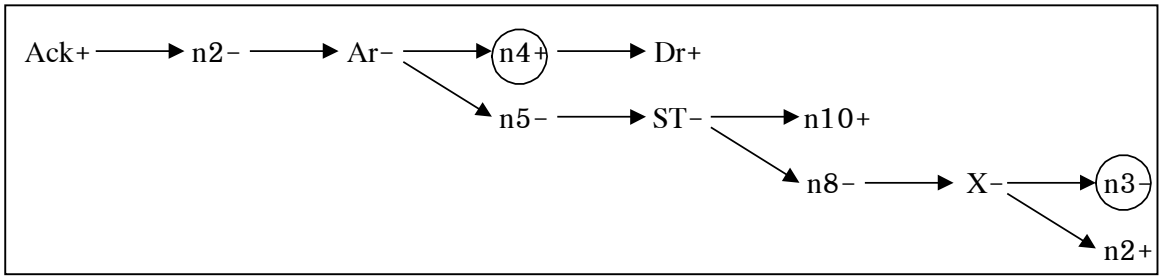
สถานะคงที่ปลายทาง 1 1 1 0 1 0 1 0 1 0

ลำดับอินพุต Ack+

- พฤติกรรมปกติ



- พฤติกรรมเมื่อเกิดข้อผิดพร่อง



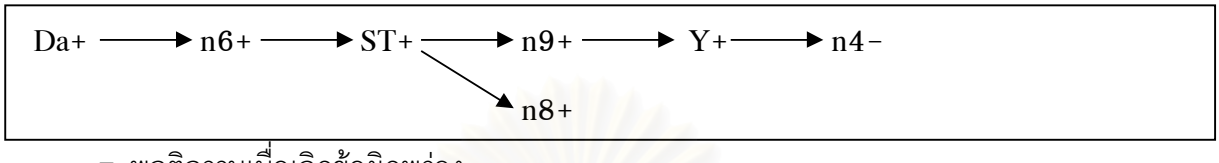
13.2 ST2 s-a-0

สถานะคงที่ต้นทาง 0 1 0 0 0 0 1 0 0 0

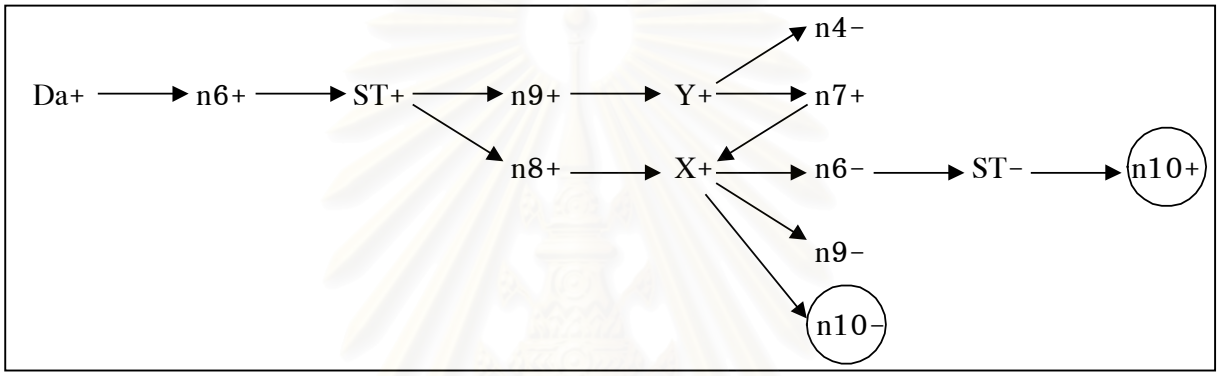
สถานะคงที่ปลายทาง 0 1 0 1 0 0 1 1 0 1

ลำดับอินพุต Da+

- พฤติกรรมปกติ



- พฤติกรรมเมื่อเกิดข้อผิดพร่อง



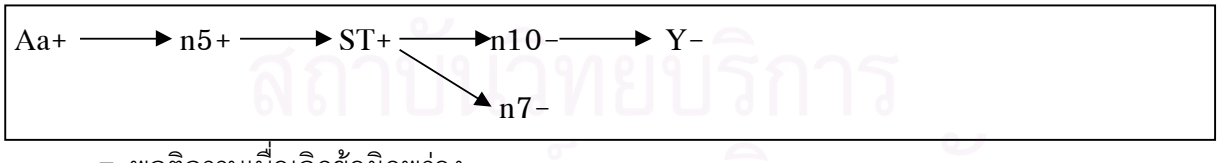
13.3 ST3 s-a-0

สถานะคงที่ต้นทาง 1 0 0 0 1 1 0 0 1 1

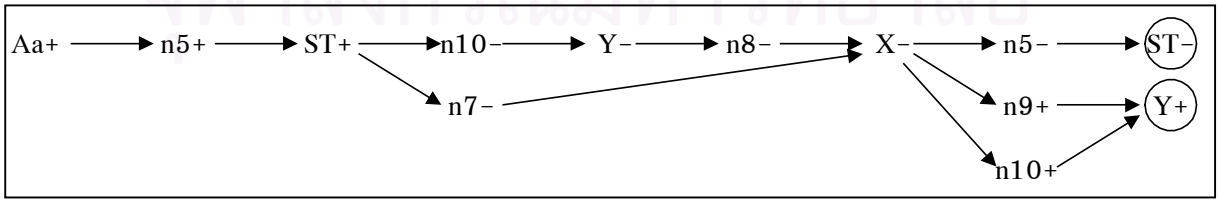
สถานะคงที่ปลายทาง 1 1 0 0 1 1 0 1 1 0

ลำดับอินพุต Aa+

- พฤติกรรมปกติ



- พฤติกรรมเมื่อเกิดข้อผิดพร่อง



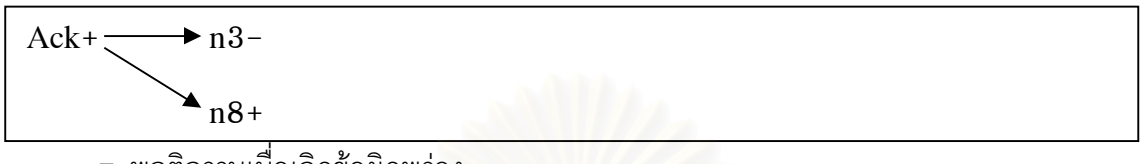
13.4 X1 s-a-1

สถานะคงที่ต้นทาง 0 1 0 0 0 0 1 0 0 0

สถานะคงที่ปลายทาง 0 1 1 0 0 0 1 0 0 0

ลำดับอินพุต Ack+

- พฤติกรรมปกติ



- พฤติกรรมเมื่อเกิดข้อผิดพร่อง

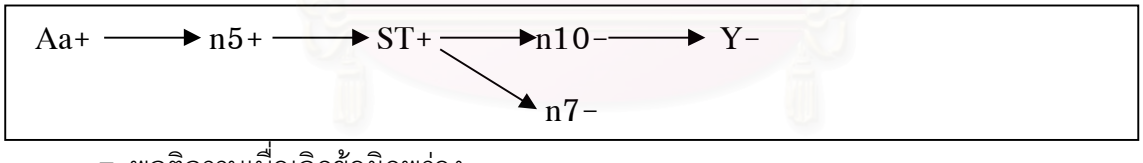
13.5 X5 s-a-0

สถานะคงที่ต้นทาง 1 0 0 0 1 1 0 0 1 1

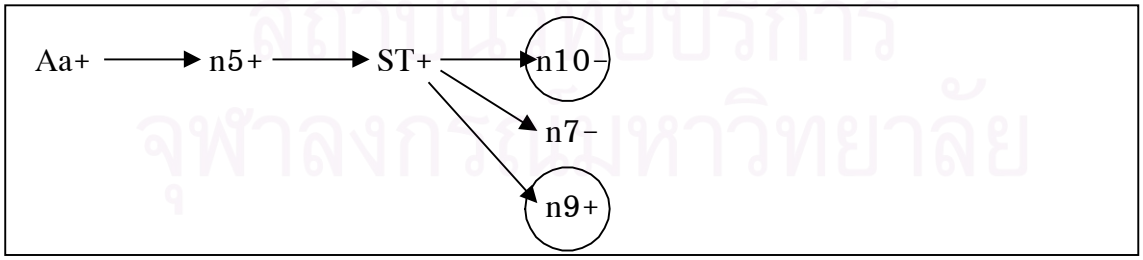
สถานะคงที่ปลายทาง 1 0 0 0 1 1 0 1 1 0

ลำดับอินพุต Aa+

- พฤติกรรมปกติ



- พฤติกรรมเมื่อเกิดข้อผิดพร่อง



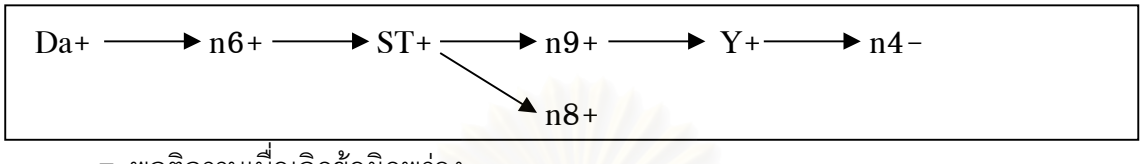
13.6 x6 s-a-1

สถานะคงที่ต้นทาง 0 1 0 0 0 0 1 0 0 0

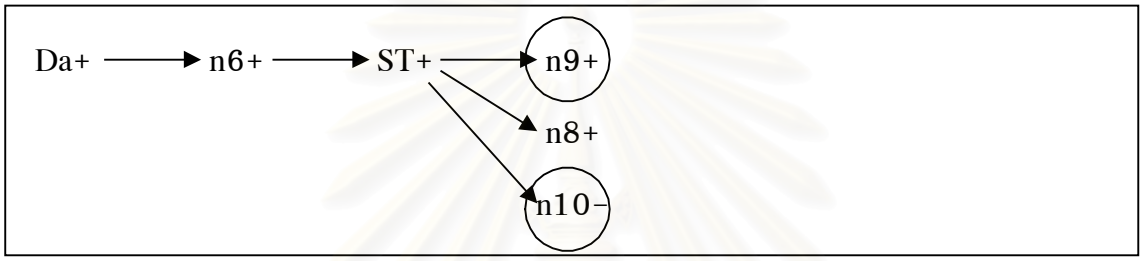
สถานะคงที่ปลายทาง 0 1 0 1 0 0 1 1 0 1

ลำดับอินพุต Da+

- พฤติกรรมปกติ



- พฤติกรรมเมื่อเกิดข้อผิดพร่อง



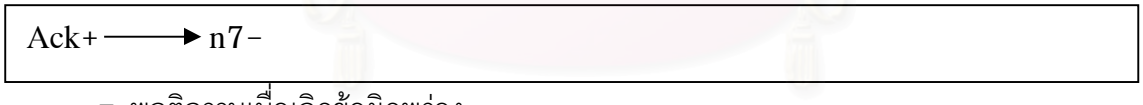
13.7 Y2 s-a-0

สถานะคงที่ต้นทาง 1 0 0 0 1 1 0 0 1 1

สถานะคงที่ปลายทาง 1 0 1 0 1 1 0 0 1 1

ลำดับอินพุต Ack+

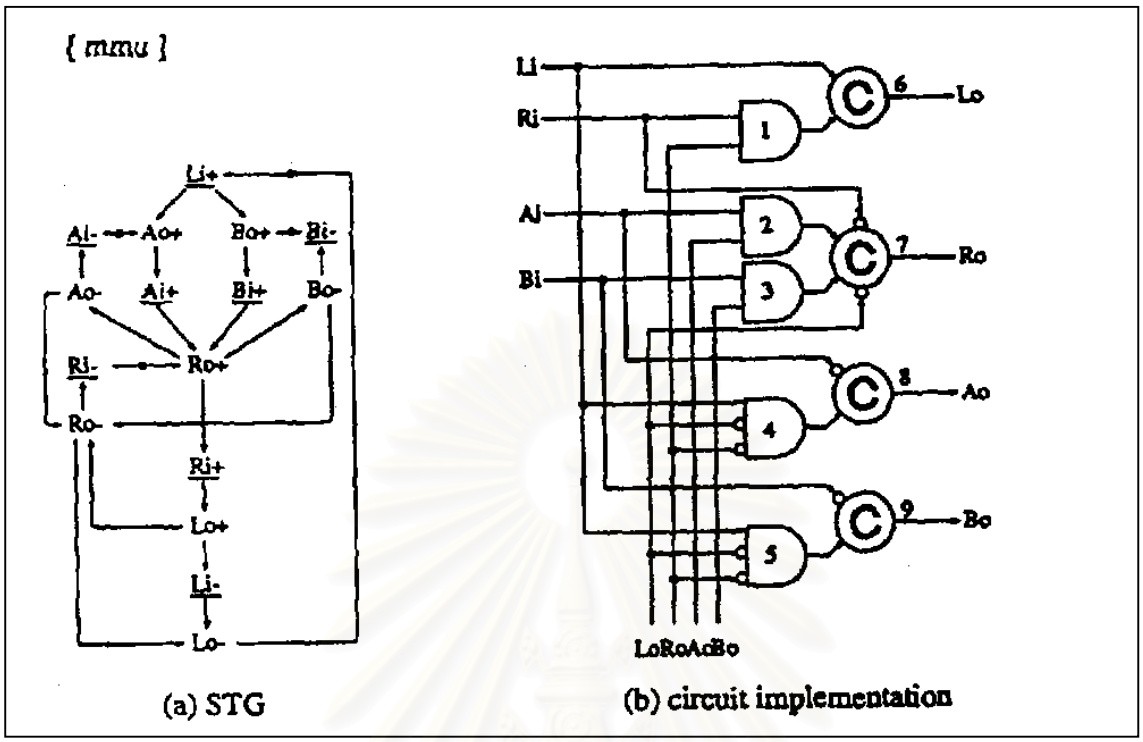
- พฤติกรรมปกติ



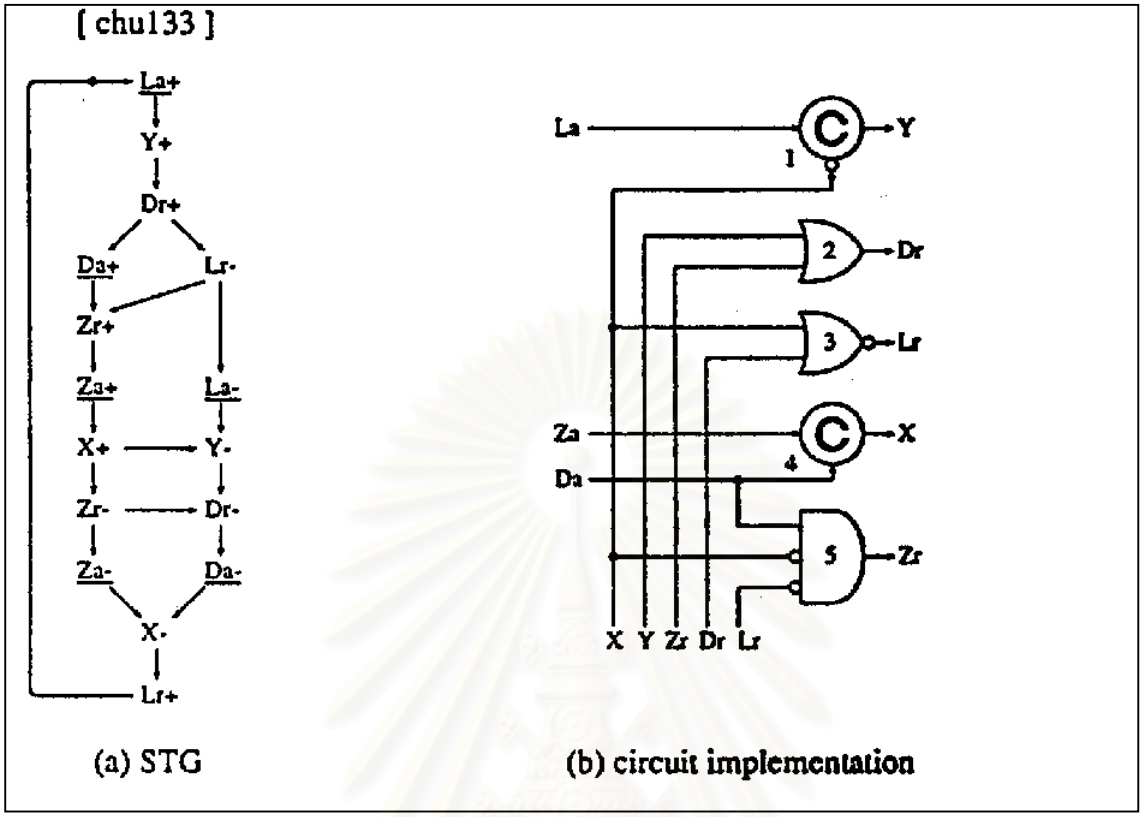
- พฤติกรรมเมื่อเกิดข้อผิดพร่อง

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

14. วงจร mmu


{ mmu }
Li+
Ai- Ao+ Bo+ Bi-
Ao- Ai+ Bi+ Bo-
Ri- Ro+
Ro-
Ri+
Lo+
Li-
Lo-
(a) STG
Li
Ri
Ai
Bi
1
2
3
4
5
6
7
8
9
Lo
Ro
Ao
Bo
LoRoAoBo
(b) circuit implementation

วงจร chu133



State Encoder La Za Da X Y Zr Dr Lr

สถานะเริ่มต้น 0 0 0 0 0 0 1

จำนวนสถานะคงที่ 10

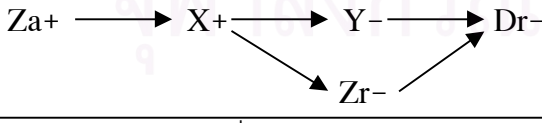
15.1 Zr s-a-1

สถานะคงที่ต้นทาง 1 0 1 0 1 1 1 0

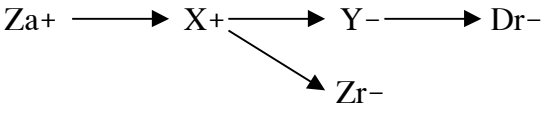
สถานะคงที่ปลายทาง 1 1 1 1 0 0 0 0

ลำดับอินพุต Za+

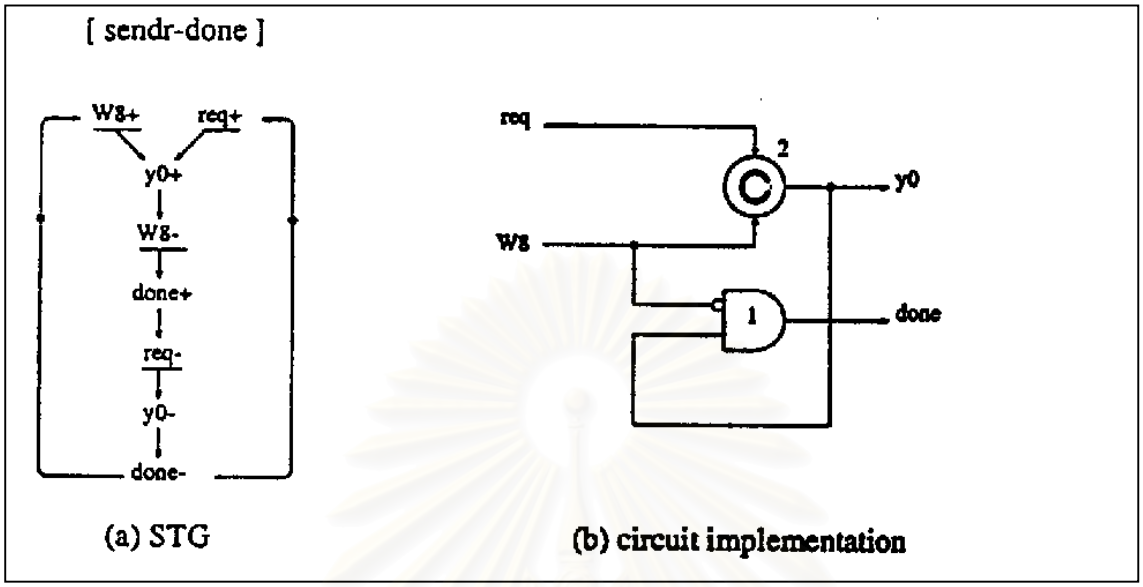
- พฤติกรรมปกติ



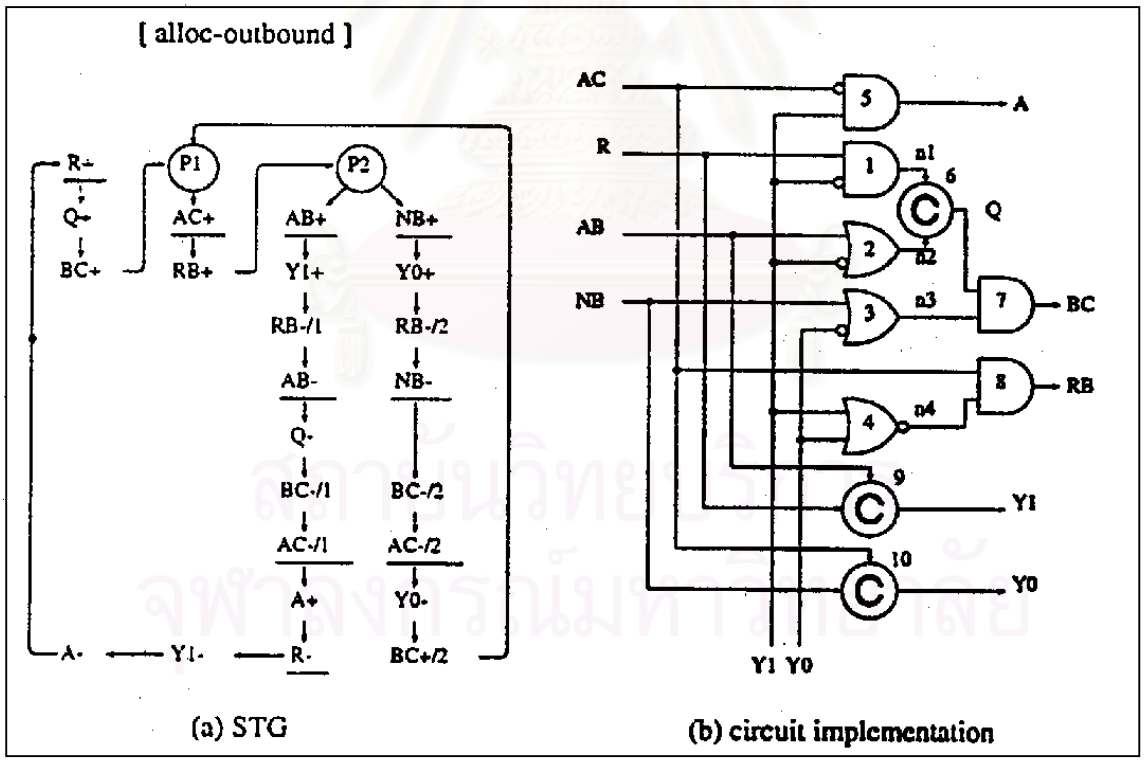
- พฤติกรรมเมื่อเกิดข้อผิดพร่อง



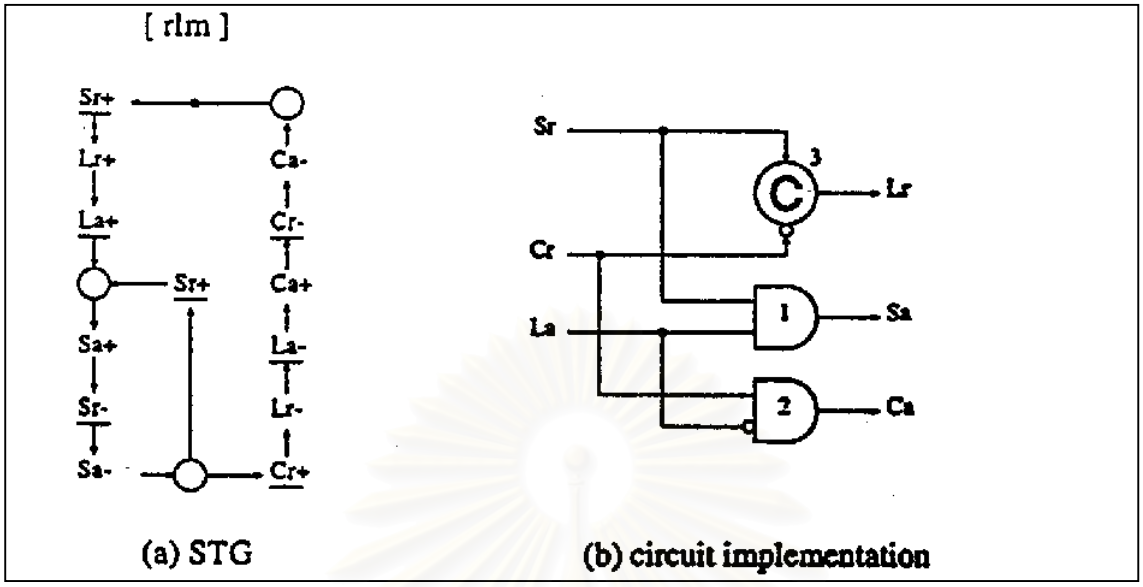
16. วงจร sendr-done



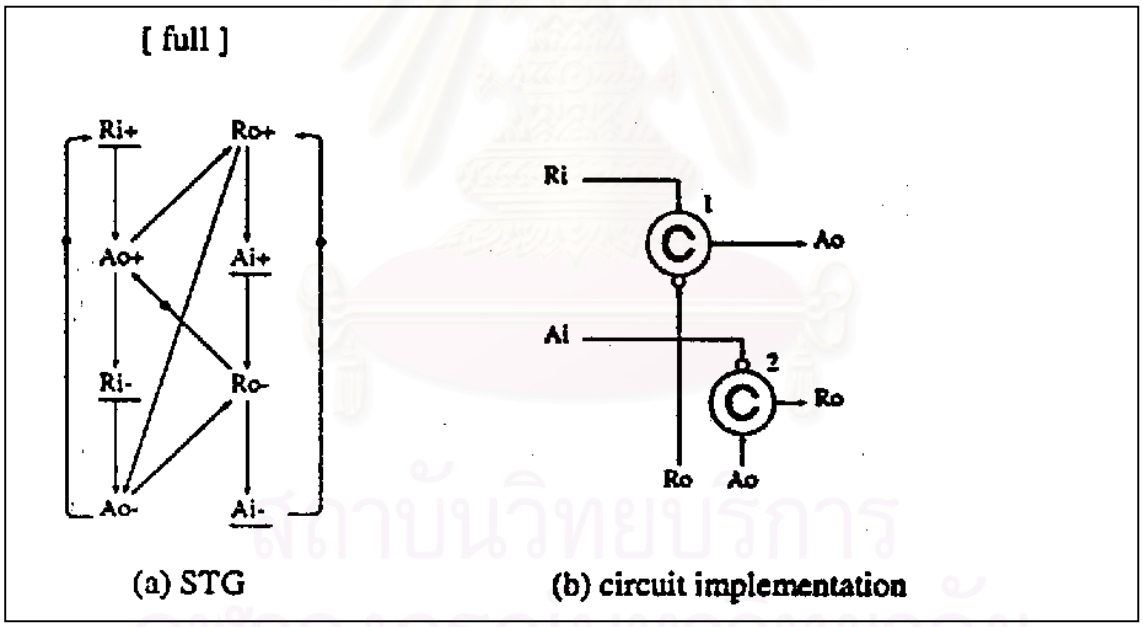
17. วงจร alloc-outbound



18. วงจร rlm

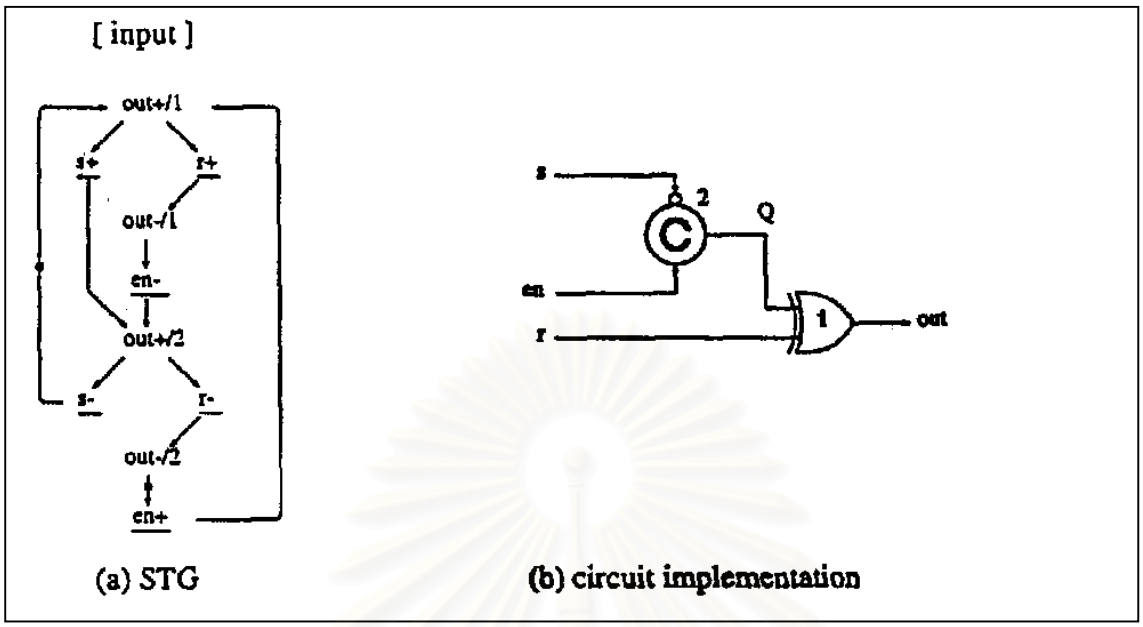


19. วงจร full



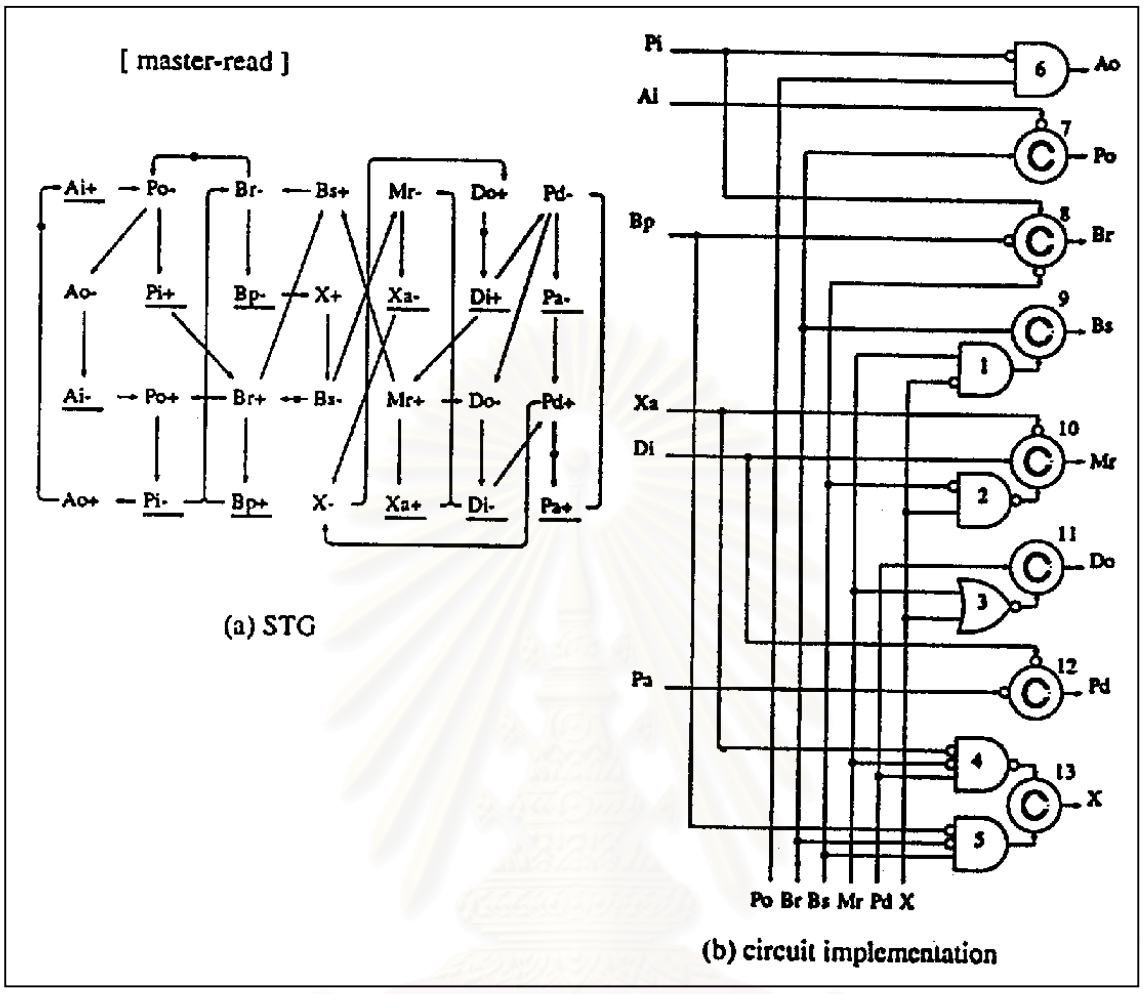
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

20. วงจร input



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

21. วงจร master-read



State Encoder Pi Ai Bp Xa Di Pa Po Br Bs Mr Pd X Ao Do

สถานะเริ่มต้น 0 0 0 0 0 0 1 0 0 0 1 0 1 1

21.1 Bs2 s-a-0

สถานะคงที่ต้นทาง 0 0 1 1 0 1 1 1 1 1 0 0 1 0

สถานะคงที่ปลายทาง 0 0 0 1 0 1 1 1 0 0 0 1 1 0

ลำดับอินพุต Bp-

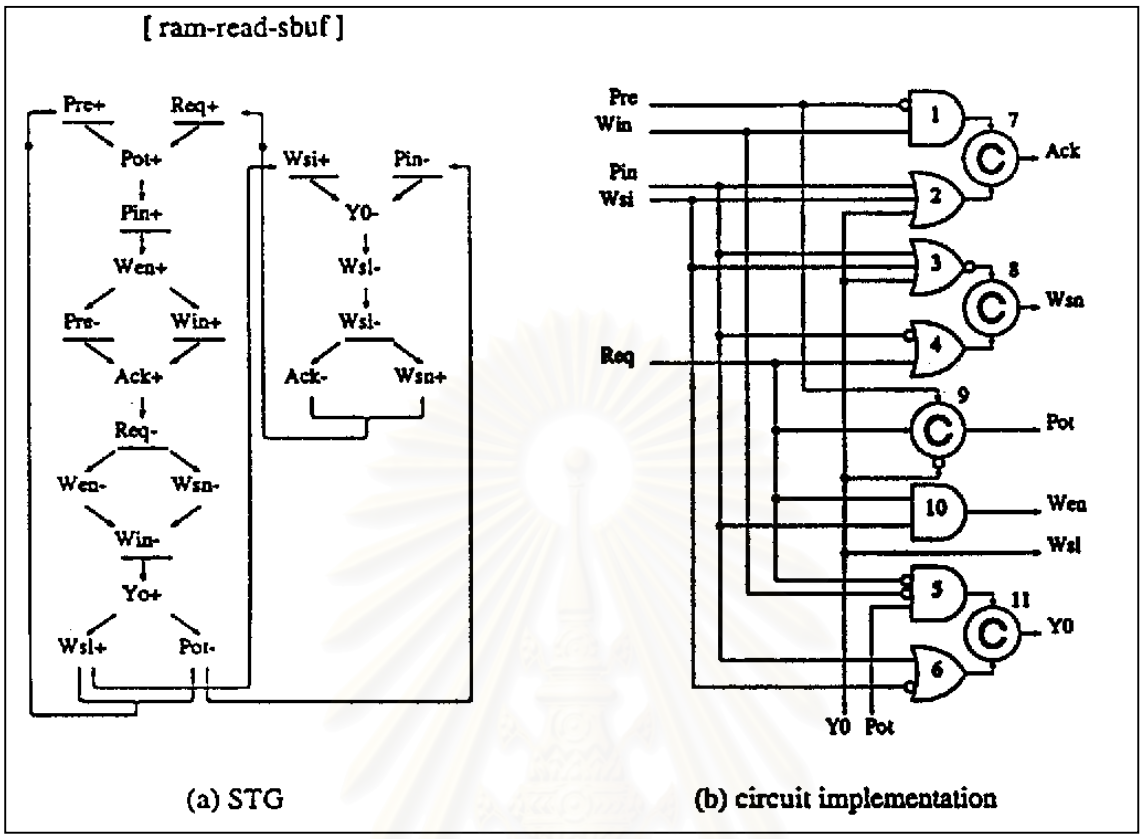
❑ พฤติกรรมปกติ

Bp− ⟶ n5+ ⟶ X+ ⟶ n1− ⟶ Bs− ⟶ n2− ⟶ Mr−
Bs− ⟶ n5−

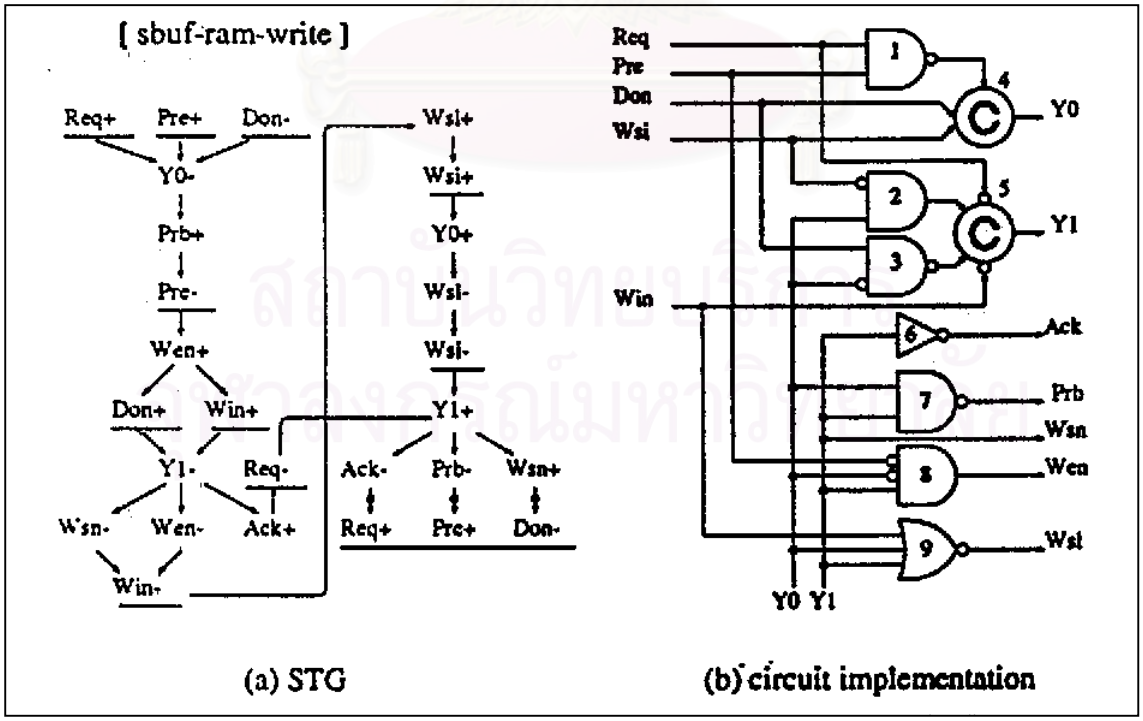
❑ พฤติกรรมเมื่อเกิดข้อผิดพร่อง

Bp− ⟶ n5+ ⟶ X+ ⟶ n1− ⟶ Bs− ⟶ n5−
X+ ⟶ n2− ⟶ Mr−

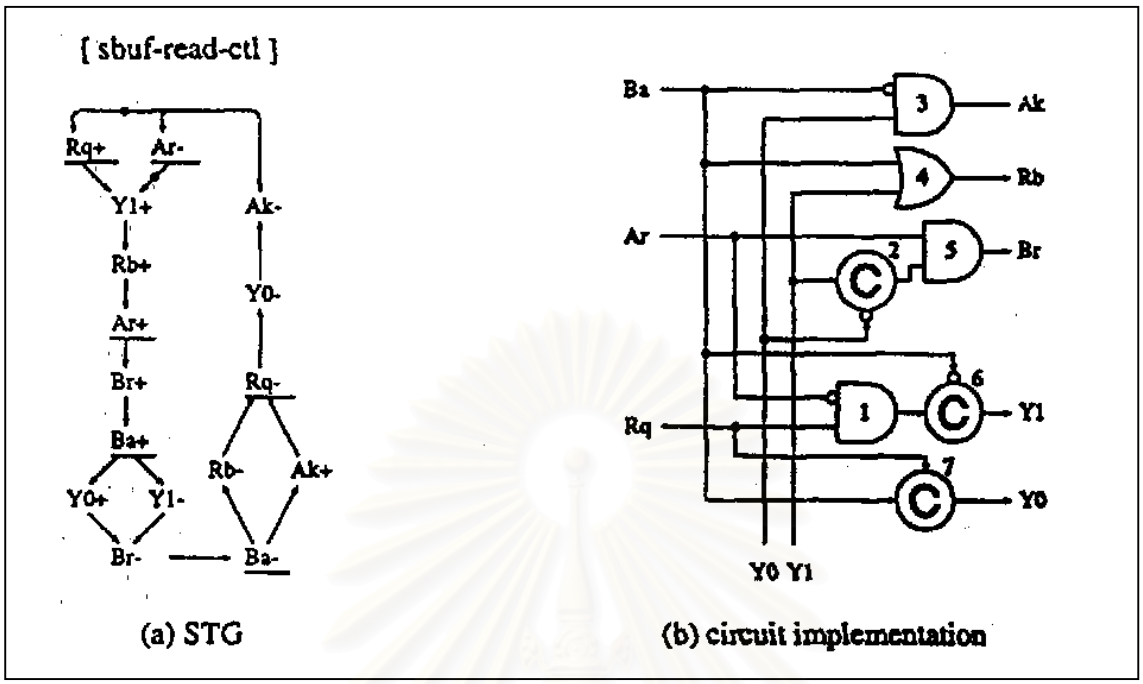
22. วงจร ram-read-sbuf



23. วงจร sbuf-ram-write

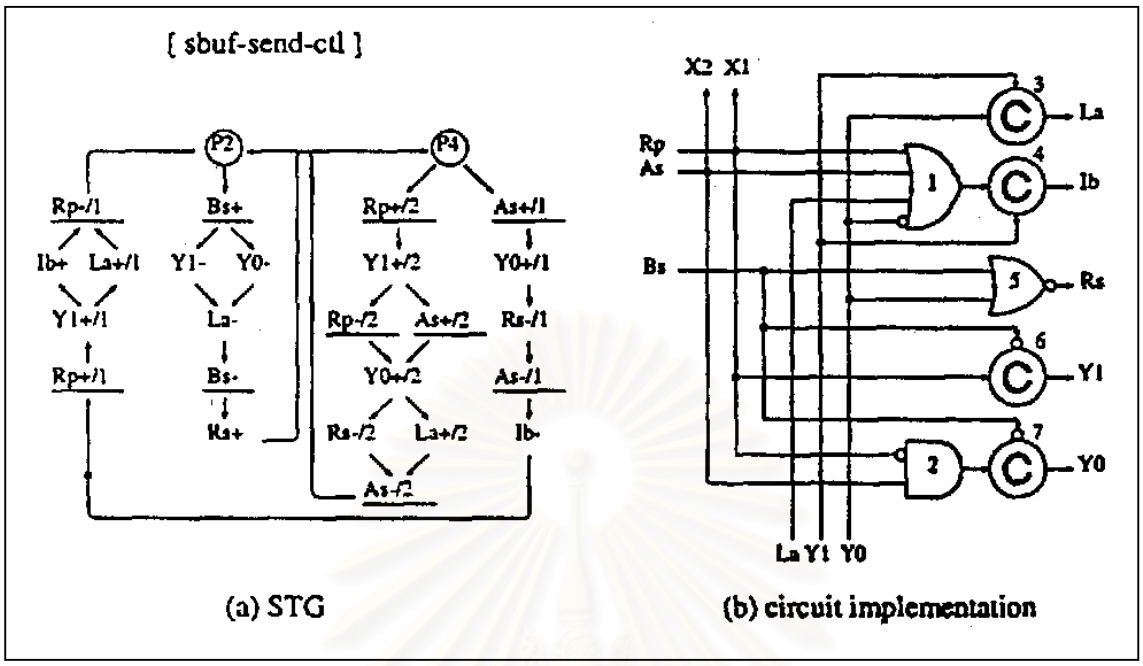


24. วงจร sbuf-read-ctl



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

25. วงจร sbuf-send-ctl



State Encoder Rp As Bs La Y1 Y0 Ib Rs

สถานะเริ่มต้น 0 0 0 0 0 1 0 0

จำนวน สถานะคงที่ 10

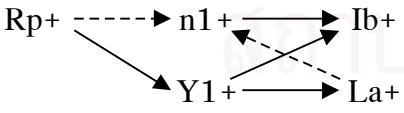
25.1 Rp1 s-a-0

สถานะคงที่ต้นทาง 0 0 0 0 0 1 0 0

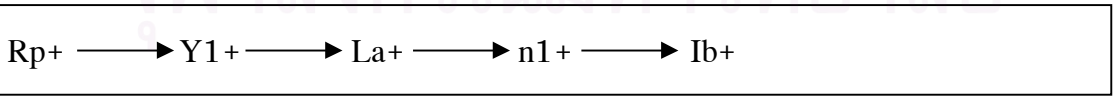
สถานะคงที่ปลายทาง 1 0 0 1 1 0 1 0

ลำดับอินพุต Rp+

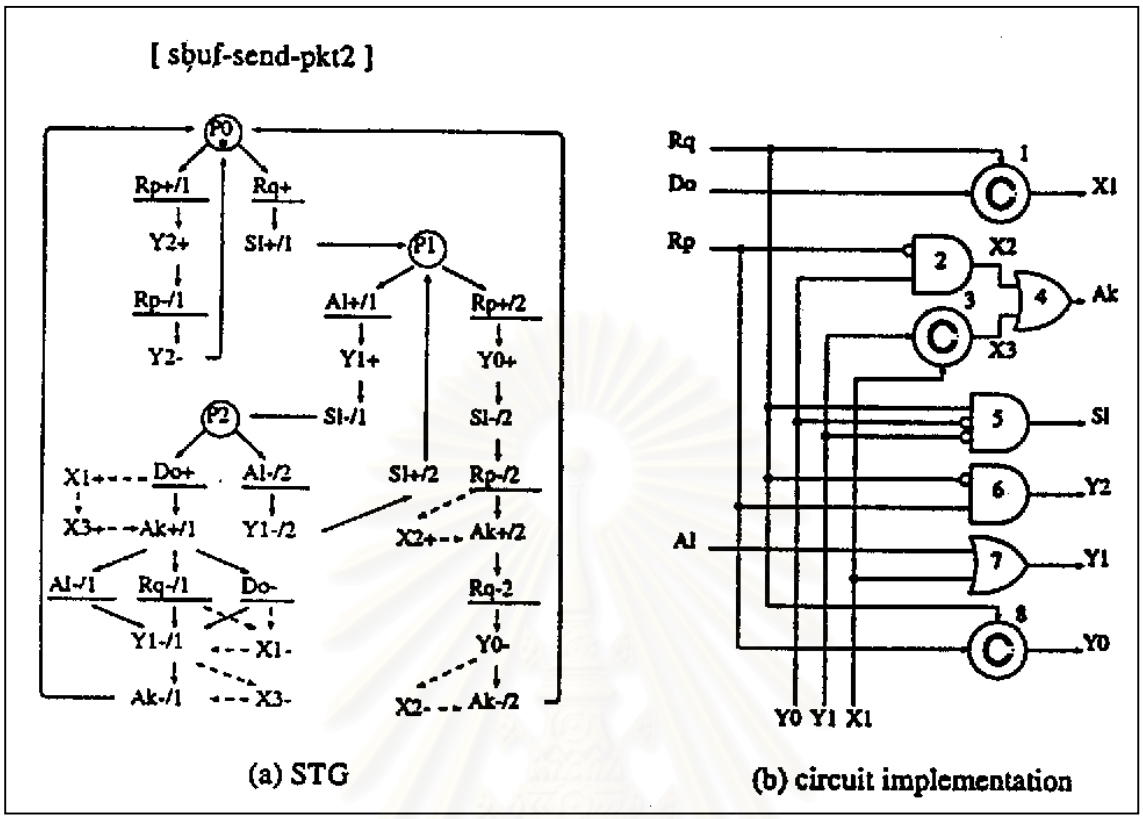
- พฤติกรรมปกติ



- พฤติกรรมเมื่อเกิดข้อผิดพร่อง

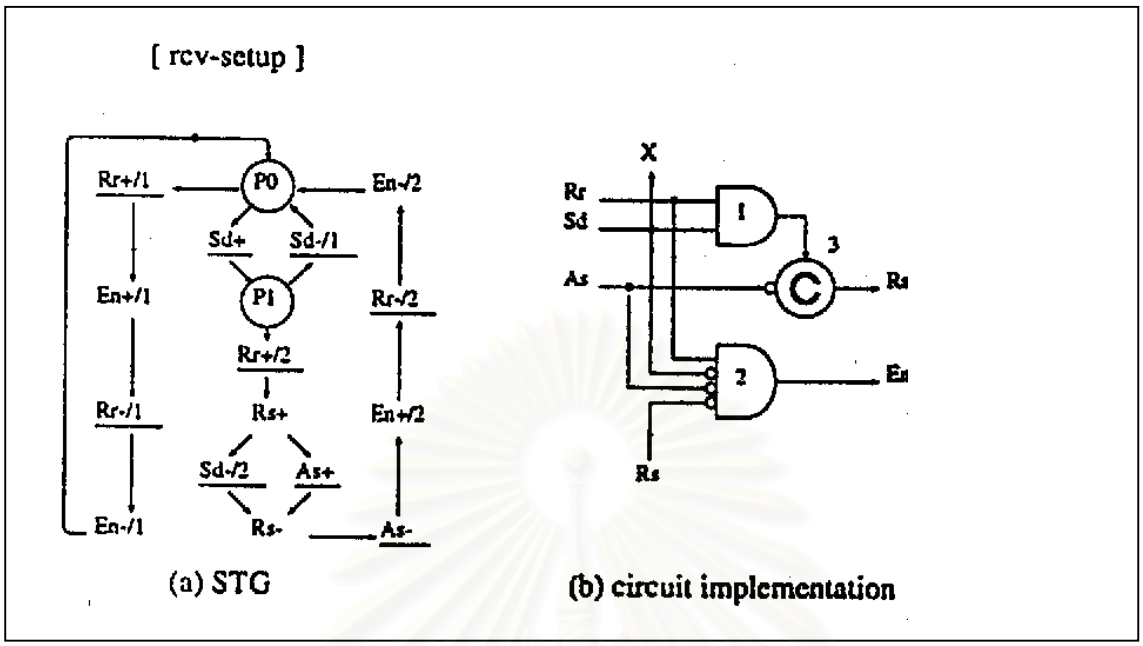


26. วงจร sbuf-send-pkt2



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

27. วงจร rcv-setup



State Encoder Rr Sd As Rs En

สถานะเริ่มต้น 0 0 0 0 0

จำนวน สถานะคงที่ 7

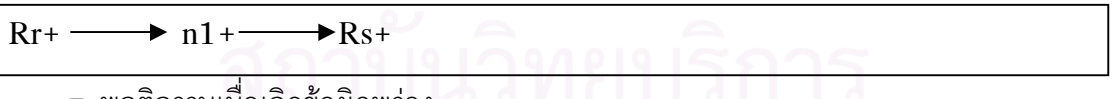
27.1 Sd2 s-a-0

สถานะคงที่ต้นทาง 0 1 0 0 0

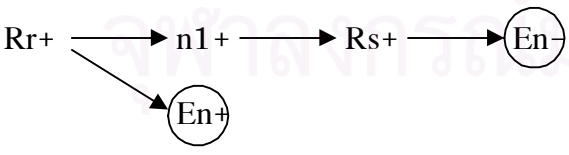
สถานะคงที่ปลายทาง 1 1 0 1 0

ลำดับอินพุต Rr+

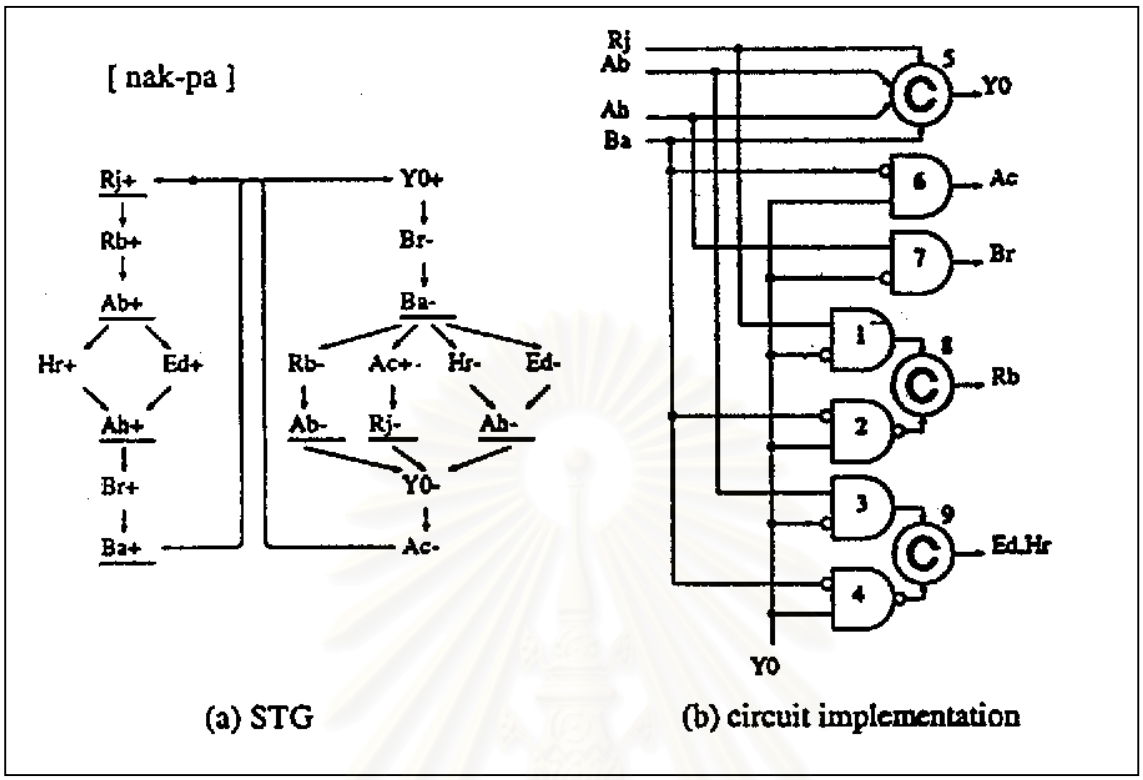
- พฤติกรรมปกติ



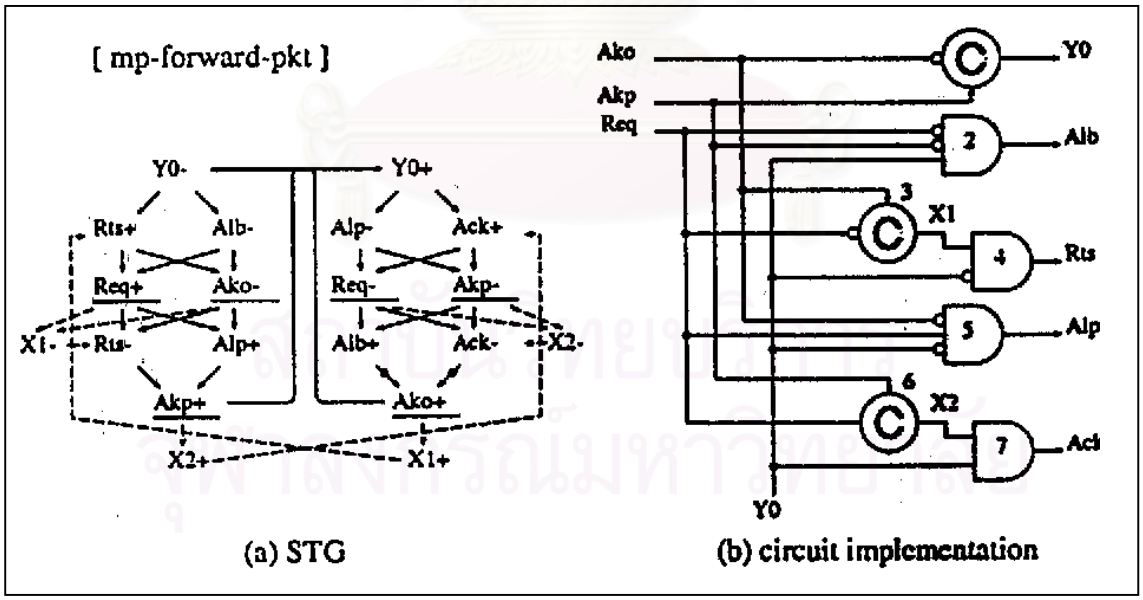
- พฤติกรรมเมื่อเกิดข้อผิดพร่อง



28. วงจร nak-pa



29. วงจร mp-forward-pkt



ประวัติผู้วิจัย

นายวุฒิชัย เลิศศิริสัมพันธ์ เกิดวันที่ 1 สิงหาคม พ.ศ. 2516 ที่อำเภอบางกอกน้อย จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาปริญญาตรีวิทยาศาสตร์บัณฑิต ภาควิชาคณิตศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าธนบุรี ในปีการศึกษา 2537 และเข้าศึกษาต่อในหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย เมื่อ พ.ศ. 2540

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย