การทวนสอบเชิงฟังก์ชันสำหรับแบบจำลองประกอบของการออกแบบบนพื้นฐานของกระแสด้วยภาวะนามธรรมประวัติ

นายนนทศักดิ์ จันทร์ชุม

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2551

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

FUNCTIONAL VERIFICATION FOR COMPOSITE MODEL OF STREAM-BASED DESIGN ON HISTORY ABSTRACTION

Mr. Nontasak Janchum

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

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

Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University

Academic Year 2008

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	การทวนสอบเชิงฟังก์ชันสำหรับแบบจำลองประกอบของการออกแบบบนพื้นฐานของกระแสด้วยภาวะนามธรรมประวัติ
โดย	นายนนทศักดิ์ จันทร์ชุม
สาขาวิชา	วิทยาศาสตร์คอมพิวเตอร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	รองศาสตราจารย์ ดร.พรศิริ หมื่นไชยศรี

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์
(รองศาสตราจารย์ ดร.บุญสม เลิศหิรัญวงศ์)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(รองศาสตราจารย์ ดร.ธาราทิพย์ สุวรรณศาสตร์)

..... อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(รองศาสตราจารย์ ดร.พรศิริ หมื่นไชยศรี)

..... กรรมการภายนอกมหาวิทยาลัย
(ผู้ช่วยศาสตราจารย์ ดร.ทรงศักดิ์ รองวิริยะพานิช)

..... กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.วิวัฒน์ วัฒนาวุฒิ)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

นนทศักดิ์ จันทร์ชุม : การทวนสอบเชิงฟังก์ชันสำหรับแบบจำลองประกอบของการออกแบบบนพื้นฐานของกระแสด้วยภาวะนามธรรมประวัติ. (Functional Verification for Composite Model of Stream-Based Design on History Abstraction) อ. ที่ปรึกษาวิทยานิพนธ์หลัก : รศ. ดร. พรศิริ หมื่นไชยศรี, 128 หน้า.

วิทยานิพนธ์นี้มีวัตถุประสงค์เพื่อนำเสนอวิธีการสำหรับการทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบด้วยการตรวจสอบแบบจำลอง แบบจำลองประกอบในวิทยานิพนธ์นี้ได้รับจากการประกอบแบบจำลองบนพื้นฐานของกระแสเฉพาะรายด้วยตัวดำเนินการประกอบแบบลำดับ ด้วยตัวดำเนินการประกอบแบบลำดับนี้แบบจำลองประกอบสามารถจำลองด้วยฟังก์ชันการประมวลผลกระแส ฟังก์ชันการประมวลผลกระแสสามารถแปลงเป็นระบบเปลี่ยนสถานะนำเข้า/นำออกที่เป็นแบบจำลองบนพื้นฐานสถานะที่มีความเหมาะสมสำหรับการทวนสอบสมบัตินำเข้า/นำออกที่คาดหมายด้วยการตรวจสอบแบบจำลอง

ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบสามารถพิจารณาได้จากฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองบนพื้นฐานของกระแสเฉพาะรายที่นำมาประกอบ และสมบัตินำเข้า/นำออกที่คาดหมายของแบบจำลองประกอบซึ่งได้จากการเปลี่ยนนำออกของสมบัตินำเข้า/นำออกที่คาดหมายของส่วนประกอบแรกโดยการประยุกต์ฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่สองไปยังนำออกของส่วนประกอบแรก และเปลี่ยนนำเข้าของสมบัตินำเข้า/นำออกที่คาดหมายของส่วนประกอบที่สอง โดยการประยุกต์ผกผันของฟังก์ชันการประมวลผลกระแสของส่วนประกอบแรก (ถ้ามี) ไปยังนำเข้าของส่วนประกอบที่สอง

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ภาควิชา.....วิศวกรรมคอมพิวเตอร์..... ลายมือชื่อนิสิต.....นนทศักดิ์ จันทร์ชุม.....

สาขาวิชา.....วิทยาศาสตร์คอมพิวเตอร์..... ลายมือชื่ออ.ที่ปรึกษาวิทยานิพนธ์หลัก.....

ปีการศึกษา 2551

4970370621 : MAJOR COMPUTER SCIENCE

KEY WORD: STREAM PROCESSING FUNCTION/ TEMPORAL LOGIC/ MODEL CHECKING/ MOORE-BASED STATE TRANSITION MACHINE/ INPUT/OUTPUT TRANSITION SYSTEM/ COMPOSITE MODEL

NONTASAK JANCHUM : FUNCTIONAL VERIFICATION FOR COMPOSITE MODEL OF STREAM-BASED DESIGN ON HISTORY ABSTRACTION. THESIS PRINCIPAL ADVISOR : ASSOC. PROF. PORNSIRI MUENCHAISRI, Ph.D., 128 pp.

The objective of this thesis is to present a method for verifying input/output properties of a composite model with model checking. The composite model in this thesis is derived from composing individual stream-based models with a sequential composition operator. With the sequential composition operator, composite model can be modeled with a stream processing function. The stream processing function can be transformed into an input/output transition systems, a state-based model which is suitable for verifying expected input/output properties with model checking.

A history abstraction for the Moore machine, which is a state-based model for implementation, plays a crucial role in reducing the number of states of an input/output transition system. With a finite state space, a stream processing function is used to verified input/output properties with model checking. A history abstraction for the Moore machine of a composite model can be considered from history abstractions for the Moore machine of individual stream-based models. Expected input/output properties of a composite model can be obtained from changing output of expected input/output properties of the first component by applying the stream processing function of the second component to output of the first component and/or changing input of expected input/output properties of the second component by applying the inverse stream processing function of the first component (if exists) to input of the second component.

Department.....Computer Engineering..... Student's Signature Nontasak Janchum

Field of Study....Computer Science..... Principal Advisor's Signature P. Muenchaisri

Academic Year 2008

กิตติกรรมประกาศ

ขอขอบพระคุณอาจารย์ที่ปรึกษา รองศาสตราจารย์ ดร.พรศิริ หมื่นไชยศรี ผู้ซึ่งเสียสละเวลาช่วยเหลือทั้งทางด้านวิชาการและทางด้านอื่นๆอีกมากมาย รวมถึงให้แนวทางและโอกาสในการสร้างประสบการณ์ทางด้านงานวิจัยที่ดีมาก ทำให้งานวิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี

ขอขอบพระคุณคณะกรรมการสอบวิทยานิพนธ์ทุกท่าน รองศาสตราจารย์ ดร.ธาราทิพย์ สุวรรณศาสตร์ ผู้ช่วยศาสตราจารย์ ดร.วิวัฒน์ วัฒนาวุฒิ และผู้ช่วยศาสตราจารย์ ดร.ทรงศักดิ์ รองวิริยะพานิช ที่กรุณาสละเวลาในการให้คำแนะนำเกี่ยวกับงานวิจัย และตรวจสอบความถูกต้องสมบูรณ์ของวิทยานิพนธ์ฉบับนี้

ขอขอบพระคุณพี่วุฒิพงษ์ เรือนทอง ผู้ริเริ่มแนะแนวทางปัญหาทางด้านการวิจัย รวมถึงให้คำปรึกษาและความช่วยเหลือทางด้านเอกสารทางวิชาการต่างๆ ที่มีประโยชน์อย่างมากที่ทำให้งานวิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี

ขอขอบพระคุณ Professor Dr. Walter Dosch ที่ให้คำปรึกษา ตอบคำถาม และให้คำชี้แนะ รวมทั้งให้เอกสารทางวิชาการมีประโยชน์อย่างมาก และ Dr. Annete Stumpel ที่ส่งเอกสารวิทยานิพนธ์ของตนเองและนักศึกษาในความดูแล

ขอกราบขอบพระคุณสำนักงานกองทุนสนับสนุนการวิจัย และจุฬาลงกรณ์มหาวิทยาลัย ที่กรุณาให้ทุนอุดหนุนสำหรับการทำวิทยานิพนธ์ ภายใต้โครงการทุนวิจัยมหาบัณฑิต สกว. สาขาวิชาวิทยาศาสตร์และเทคโนโลยี และความเห็นในรายงานผลการวิจัยเป็นของผู้รับทุน สำนักงานกองทุนสนับสนุนการวิจัย และจุฬาลงกรณ์มหาวิทยาลัย ไม่จำเป็นต้องเห็นด้วยเสมอไป

ขอขอบพระคุณคณาจารย์ทุกท่านและเพื่อนๆทุกคนในภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ที่ให้ความรู้ คำแนะนำในการเรียน และการทำวิจัย

สุดท้ายนี้ ขอกราบขอบพระคุณบิดา มารดา และสมาชิกในครอบครัวทุกท่านที่ให้การสนับสนุน ให้กำลังใจ ในการทำงานที่ต้องใช้ความมานะและอดทน แก่ผู้วิจัยตลอดมาจนสำเร็จการศึกษา

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญ

หน้า

บทคัดย่อภาษาไทย..... ง

บทคัดย่อภาษาอังกฤษ..... จ

กิตติกรรมประกาศ..... ฉ

สารบัญ..... ช

สารบัญตาราง..... ฎ

สารบัญภาพ..... ฐ

บทที่

1 บทนำ..... 1

1.1 ความเป็นมาและความสำคัญของปัญหา..... 1

1.2 วัตถุประสงค์ของการวิจัย..... 3

1.3 ขอบเขตของการวิจัย..... 3

1.4 ประโยชน์ที่คาดว่าจะได้รับ..... 4

1.5 วิธีดำเนินการวิจัย..... 4

1.6 โครงสร้างของเนื้อหางานวิจัย..... 5

1.7 นิยาม..... 5

2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง..... 6

2.1 แนวคิดและทฤษฎี..... 6

2.1.1 กระแสและการพัฒนาบนพื้นฐานของกระแส..... 6

2.1.1.1 กระแส..... 6

2.1.1.2 ฟังก์ชันการประมวลผลกระแส..... 9

2.1.1.2.1 ตัวอย่างฟังก์ชันการประมวลผลกระแส..... 10

2.1.1.3 การประกอบของฟังก์ชันการประมวลผลกระแส..... 13

2.1.1.3.1 การประกอบแบบลำดับ..... 13

2.1.1.3.2 การประกอบแบบขนาน..... 14

2.1.1.3.3 ผลป้อนกลับ..... 14

2.1.1.4 เครื่องเปลี่ยนสถานะ..... 15

2.1.2 การตรวจสอบแบบจำลอง..... 19

2.1.2.1 โครงสร้างคริปเก..... 19

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ หน้า

2.1.2.2 ตรรกศาสตร์เชิงกาลเวลา..... 20
2.1.2.2.1 ตรรกซีทีแอลสตาร์..... 20
2.1.2.2.2 ตรรกซีทีแอล..... 23
2.1.2.2.3 ตรรกแอลทีแอล..... 23
2.1.3 ระบบเปลี่ยนสถานะนำเข้า/นำออก..... 23
2.2 งานวิจัยที่เกี่ยวข้อง..... 24
2.2.1 งานวิจัยที่เกี่ยวข้องกับกระแสและการพัฒนาบนพื้นฐานของกระแส... 24
2.2.2 งานวิจัยที่เกี่ยวข้องกับการตรวจสอบแบบจำลอง..... 28
2.2.3 งานวิจัยที่เกี่ยวข้องกับการทวนสอบสำหรับสมบัตินำเข้า/
นำออกของแบบจำลองกล่องดำ..... 29
2.2.3.1 การแปลงจากฟังก์ชันการประมวลผลกระแสเป็น
เครื่องมัวร์แบบบัญญัติ..... 29
2.2.3.2 การแปลงจากเครื่องมัวร์แบบบัญญัติเป็นระบบ
เปลี่ยนสถานะนำเข้า/นำออก..... 29
3 วิธีดำเนินการวิจัย..... 31
3.1 แนวคิดและวิธีการวิจัย..... 31
3.2 ระเบียบวิธีการทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบ..... 34
3.2.1 การประกอบฟังก์ชันการประมวลผลกระแสเป็น
แบบจำลองประกอบ..... 34
3.2.2 การแปลงแบบจำลองประกอบเป็นระบบเปลี่ยน
สถานะนำเข้า/นำออก..... 36
3.2.3 การกำหนดสมบัตินำเข้า/นำออกที่คาดหมายในรูปแบบ
ของตรรกแอลทีแอล..... 38
3.2.4 การทวนสอบแบบจำลองประกอบโดยใช้การตรวจสอบ
แบบจำลอง..... 38
3.3 สรุป..... 38
4 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองบน
พื้นฐานของกระแสเฉพาะราย..... 39
4.1 ฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบที่อิสระจากประวัติ..... 39

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ หน้า

4.1.1 ลักษณะของส่วนประกอบที่อิสระจากประวัติ..... 39
4.1.2 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของ
ส่วนประกอบที่อิสระจากประวัติ..... 40
4.1.3 นิยามของส่วนประกอบที่อิสระจากประวัติ..... 40
4.1.4 ตัวอย่างการแปลงส่วนประกอบที่อิสระจากประวัติเป็นระบบ
เปลี่ยนสถานะนำเข้า/นำออก..... 41
4.2 ฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบ
ที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต..... 43
4.2.1 ลักษณะของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบ
มีขอบเขต..... 43
4.2.2 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของ
ส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต..... 43
4.2.3 นิยามของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมี
ขอบเขต..... 44
4.2.4 ตัวอย่างการแปลงส่วนประกอบที่มีหน่วยความจำพจน์สั้น
แบบมีขอบเขตเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก..... 44
4.3 ฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบ
แบบมีคำร้องขอ..... 47
4.3.1 ลักษณะของส่วนประกอบแบบมีคำร้องขอ..... 47
4.3.2 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของ
ส่วนประกอบแบบมีคำร้องขอ..... 47
4.3.3 นิยามของส่วนประกอบแบบมีคำร้องขอ..... 48
4.3.4 ตัวอย่างการแปลงส่วนประกอบแบบมีคำร้องขอเป็นระบบ
เปลี่ยนสถานะนำเข้า/นำออก..... 48
4.4 ฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบคงที่..... 51
4.4.1 ลักษณะของส่วนประกอบคงที่..... 51
4.4.2 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของ
ส่วนประกอบคงที่..... 51
4.4.3 นิยามของส่วนประกอบที่คงที่..... 52

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ หน้า

4.4.4 ตัวอย่างการแปลงส่วนประกอบคงที่เป็นระบบเปลี่ยนสถานะนำเข้า/นำออก..... 52
4.5 ฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบสแกน..... 54
4.5.1 ลักษณะของส่วนประกอบสแกน..... 55
4.5.2 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบสแกน..... 55
4.5.3 นิยามของส่วนประกอบสแกน..... 55
4.5.4 ตัวอย่างการแปลงส่วนประกอบสแกนเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก..... 55
4.6 ฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบที่ต้องการประวัติ..... 58
4.6.1 ลักษณะของส่วนประกอบที่ต้องการประวัติ..... 58
4.6.2 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบที่ต้องการประวัติ..... 58
4.6.3 นิยามของส่วนประกอบที่ต้องการประวัติ..... 59
4.6.4 ตัวอย่างการแปลงส่วนประกอบที่ต้องการประวัติเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก..... 59
4.7 สรุป..... 61
5 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่อวมัวร์ของแบบจำลองประกอบ..... 62
5.1 การกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบ..... 62
5.1.1 การกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบ..... 63
5.1.2 ตัวอย่างการกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบ และการแปลงแบบจำลองประกอบเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก..... 78
5.1.3 ความสัมพันธ์ของฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบกับแบบจำลองบนพื้นฐานของกระแสเฉพาะรายที่นำมาประกอบ..... 98

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ หน้า

5.2 การทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบด้วยการตรวจสอบแบบจำลอง..... 99
5.2.1 วิธีการในการกำหนดสมบัตินำเข้า/นำออกของแบบจำลองประกอบ..... 99
5.2.2 การทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบด้วยการตรวจสอบแบบจำลอง..... 101
5.2.3. ตัวอย่างการกำหนดสมบัตินำเข้า/นำออกของแบบจำลองประกอบ และวิธีการทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบด้วยการตรวจสอบแบบจำลอง..... 101
5.3 สรุป..... 103
6 บทสรุปและข้อเสนอแนะ..... 105
6.1 บทสรุป..... 105
6.2 ข้อจำกัด..... 106
6.3 ข้อเสนอแนะ..... 107
6.4 ผลงานตีพิมพ์..... 107
รายการอ้างอิง..... 109
ภาคผนวก..... 112
ภาคผนวก ก การทดสอบสมบัติการเปลี่ยนกลุ่มของตัวดำเนินการประกอบแบบลำดับ..... 113
ภาคผนวก ข รหัสต้นฉบับภาษาเอสเอ็มวี..... 115
ข.1 รหัสต้นฉบับภาษาเอสเอ็มวีที่ใช้ในการทวนสอบสมบัตินำเข้า/นำออกของส่วนประกอบเรจิสเตอร์การเลื่อน..... 115
ข.2 รหัสต้นฉบับภาษาเอสเอ็มวีที่ใช้ในการทวนสอบสมบัตินำเข้า/นำออกของส่วนประกอบสำเนา..... 117
ข.3 รหัสต้นฉบับภาษาเอสเอ็มวีที่ใช้ในการทวนสอบสมบัตินำเข้า/นำออกของส่วนประกอบตัวหน่วงการสำเนา..... 118
ภาคผนวก ค ผลงานตีพิมพ์..... 122
ประวัติผู้เขียนวิทยานิพนธ์..... 128

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญตาราง

ตาราง หน้า

4.1 ตารางการเปลี่ยนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกของส่วนประกอบสำเนา..... 42
4.2 ตารางการเปลี่ยนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกของส่วนประกอบเรจิสเตอร์การเลื่อน..... 46
4.3 ตารางการเปลี่ยนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกของส่วนประกอบหน่วยความจำ..... 50
4.4 ตารางการเปลี่ยนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกของส่วนประกอบตัวกรองกำหนดเอง..... 54
4.5 ตารางการเปลี่ยนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกของส่วนประกอบผลรวม..... 57
4.6 ตารางการเปลี่ยนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกของส่วนประกอบตำแหน่ง..... 60
5.1 ตารางการเปลี่ยนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกของส่วนประกอบค่าเฉลี่ย..... 82
5.2 ตารางการเปลี่ยนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกของส่วนประกอบเรจิสเตอร์การเลื่อนคู่..... 86
5.3 ตารางการเปลี่ยนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกของส่วนประกอบตัวหน่วงการสำเนา..... 89
5.4 ตารางการเปลี่ยนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกของส่วนประกอบตัวนับตัวเลข..... 94
5.5 ตารางการเปลี่ยนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกของส่วนประกอบตัวหน่วงผลบวก..... 97

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญภาพ

ภาพประกอบ		หน้า
1.1	กระบวนการพัฒนาบนพื้นฐานของกระแส	2
1.2	แบบจำลองบนพื้นฐานของกระแสเฉพาะราย	3
1.3	แบบจำลองประกอบ	3
2.1	ส่วนประกอบจำลองด้วยฟังก์ชันการประมวลผลกระแส f	9
2.2	การประกอบแบบลำดับของ f และ g	13
2.3	การประกอบแบบขนานของ f และ g	14
2.4	ผลป้อนกลับของ k ช่อง	15
2.5	เครื่องประมวลผลกระแสนำเข้า	17
2.6	ภาคขยายนำออกของการประมวลผลกระแส f	17
2.7	ตัวอย่างของโครงสร้างคริปเก	20
2.10(a)	ต้นไม้การคำนวณที่ได้จากโครงสร้างคริปเก	21
2.10(b)	โครงสร้างคริปเก	20
3.1	แบบจำลองประกอบที่ประกอบจากแบบจำลองบนพื้นฐานของกระแสเฉพาะราย	31
3.2	แสดงการทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองบนพื้นฐานของกระแสเฉพาะรายด้วยการตรวจสอบแบบจำลอง	32
3.3	แสดงแนวคิดการทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบด้วยการตรวจสอบแบบจำลอง	32
3.4	แสดงการประกอบแบบจำลองบนพื้นฐานของกระแสเฉพาะรายเป็นแบบจำลองประกอบที่สามารถทวนสอบสมบัตินำเข้า/นำออกด้วยการตรวจสอบแบบจำลอง	33
3.5	แผนภาพแสดงขั้นตอนการดำเนินการวิจัย	35
3.6	การประกอบแบบจำลอง	37
4.1	ส่วนประกอบที่อิสระจากประวัติ	40
4.2	ส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต	43
4.3	ส่วนประกอบแบบมีคำร้องขอ	47
4.4	ส่วนประกอบคงที่	51
4.5	ส่วนประกอบสแกน	55

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ภาพประกอบ หน้า

4.6 ส่วนประกอบที่ต้องการประวัติ..... 58

5.1 แบบจำลองประกอบ..... 63

5.2 ส่วนประกอบค่าเฉลี่ย..... 79

5.3 ส่วนประกอบผลรวม..... 79

5.4 ส่วนประกอบผลหาร..... 80

5.5 ส่วนประกอบเรจิสเตอร์การเลื่อนคู่..... 83

5.6 ส่วนประกอบหน่วงการสำเนา..... 87

5.7 ส่วนประกอบตัวนับตัวเลข..... 90

5.8 ส่วนประกอบตัวกรองตัวเลข..... 90

5.9 ส่วนประกอบตัวนับ..... 91

5.10 ส่วนประกอบหน่วงผลบวก..... 95

ก.1 ส่วนประกอบตัวกรองค่าเฉลี่ย..... 113

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 1
บทนำ

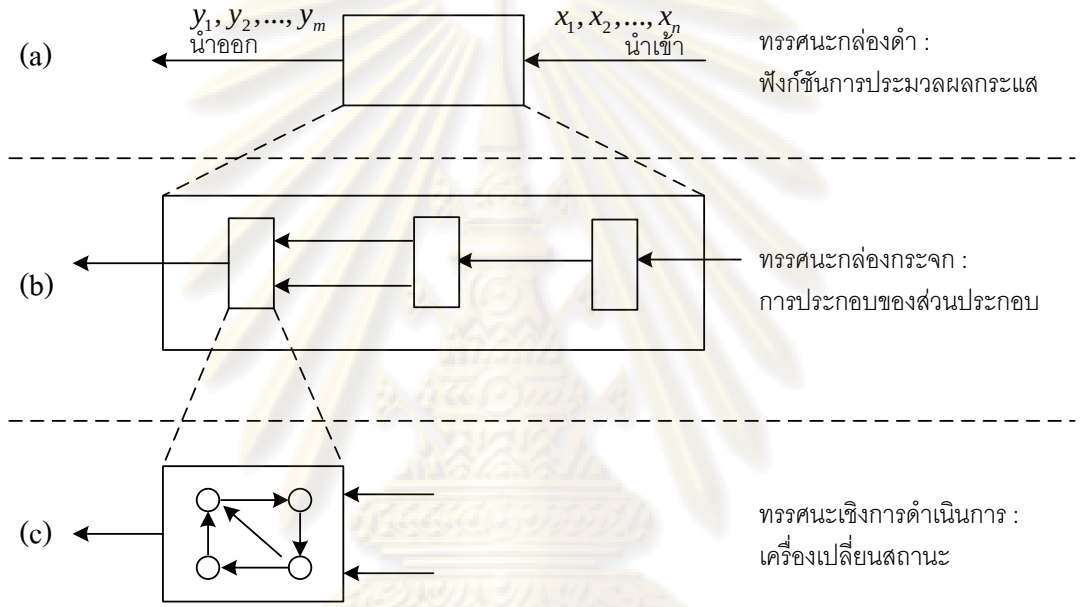
ในบทนี้ จะกล่าวถึงแนวคิดหลักของงานวิจัย อันประกอบไปด้วย ที่มาและความสำคัญ วัตถุประสงค์ ขอบเขต ประโยชน์ ขั้นตอนและโครงสร้างของเนื้อหางานวิจัย ซึ่งมีเนื้อหาดังต่อไปนี้

1.1 ความเป็นมาและความสำคัญของปัญหา

วิธีการพัฒนาบนพื้นฐานของกระแส (Stream-based development approach) [1, 2, 3, 4] เป็นวิธีการเชิงรูปนัย (Formal method) สำหรับการพัฒนาระบบที่ได้รับข้อมูลนำเข้า (Input) และปล่อยข้อมูลนำออก (Output) ในรูปแบบของกระแส (Stream) ดังแสดงในรูปที่ 1.1 วิธีการนี้เริ่มจากการบรรยายระดับสูงของพฤติกรรมนำเข้า/นำออกของระบบโดยไม่แสดงโครงสร้างการประกอบเรียกว่า ทรรศนะกล่องดำ (Black-box view) ดังแสดงในรูปที่ 1.1 (a) ซึ่งส่วนประกอบจำลองพฤติกรรมเชิงฟังก์ชัน (Functional behavior) จะแทนด้วยแบบจำลองบนพื้นฐานของกระแส เรียกว่า ฟังก์ชันการประมวลผลกระแส (Stream processing function) และพฤติกรรมเชิงฟังก์ชันของส่วนประกอบที่สัมพันธ์ตามการประมวลผลกระแสเรียกว่า พฤติกรรมนำเข้า/นำออก (Input/Output behavior) ขั้นตอนต่อไปของการพัฒนาคือ การจำลองสถาปัตยกรรมเชิงการประกอบ (Compositional architecture) ของส่วนประกอบที่บรรยายโครงสร้างการสื่อสารระหว่างส่วนประกอบ เรียกว่าทรรศนะกล่องกระจก (Glass-box view) ดังแสดงในรูปที่ 1.1 (b) ซึ่งแต่ละส่วนประกอบภายในระบบจะจำลองด้วยฟังก์ชันการประมวลผลกระแส ส่วนวิธีการพัฒนาบนพื้นฐานของกระแส กำหนดขั้นตอนการแบ่งละเอียด (Refinement) จากการออกแบบระดับสูงที่จำลองด้วยฟังก์ชันการประมวลผลกระแส ไปยังการออกแบบระดับต่ำที่จำลองด้วยเครื่องเปลี่ยนสถานะ (State transition machine) ดังแสดงในรูปที่ 1.1 (c) ซึ่งถูกใช้ในฐานะที่เป็นข้อกำหนดสำหรับการทำให้เกิดผล (Implementation) ขั้นตอนการแบ่งละเอียดนี้เป็นรูปนัยนิยม (Formalism) ซึ่งสามารถรับรองความถูกต้อง (Correctness) ของการแปลง (Transformation) ที่บ่งชี้การคงสภาพ (Preservation) ของพฤติกรรม นำเข้า/นำออก ระหว่างแบบจำลองบนพื้นฐานของกระแส (Stream-based model) และแบบจำลองบนพื้นฐานของสถานะ (State-based model)

การตรวจสอบแบบจำลอง (Model checking) [5, 6] คือ วิธีสำหรับการทวนสอบสมบัติเชิงกาลเวลา (Temporal property) โดยอัตโนมัติ ซึ่งได้แก่ สมบัติความปลอดภัย (Safety property) หรือ สมบัติความมีชีวิต (Liveness property) กระบวนการทวนสอบกระทำโดยเครื่องมือการทวนสอบที่เรียกว่า ตัวตรวจสอบแบบจำลอง (Model checker) การตรวจสอบ

แบบจำลองต้องการข้อกำหนดสำหรับกระบวนการทวนสอบในรูปของตรรกศาสตร์เชิงกาลเวลา และแบบจำลองบนพื้นฐานของสถานะ เรียกว่าโครงสร้างคริปเก (Kripke structure) ซึ่งใช้เป็นฐานหลัก (Basis) สำหรับการทวนสอบ ในกรณีที่แบบจำลองที่สร้างขึ้นไม่สอดคล้องกับสมบัติเชิงเวลาที่กำหนด ตัวตรวจสอบแบบจำลองจะสร้างตัวอย่างกรณีที่เกิดปัญหาให้ด้วยซึ่งจะเป็นประโยชน์ในการแก้ไขแบบจำลอง ค่าใช้จ่ายในการทวนสอบจะลดลงถ้าเครื่องมือการทวนสอบโดยอัตโนมัติสามารถประยุกต์กับกระบวนการพัฒนาระบบได้

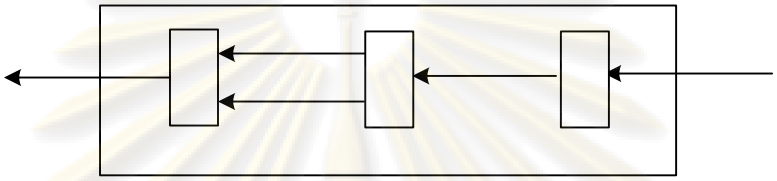


รูปที่ 1.1 กระบวนการพัฒนาบนพื้นฐานของกระแส [4]

เนื่องจากฟังก์ชันการประมวลผลกระแส มีความสัมพันธ์ของกระแสนำเข้าและกระแสนำออกเท่านั้นแต่ไม่มีโครงสร้างการเปลี่ยนสถานะ (State-transition structure) ดังนั้นการทวนสอบสมบัตินำเข้า/นำออก (Input/Output property) ไม่สามารถทำได้โดยตรง Dosch, Muenchaisri, Ruanthong และStumpel [7] ได้นำเสนอวิธีการทวนสอบสมบัตินำเข้า/นำออกของส่วนประกอบของแบบจำลองกล่องดำ (Black-box model) โดยการสร้างระบบเปลี่ยนสถานะนำเข้า/นำออก (Input/Output transition system) ซึ่งคงสภาพนำเข้า/นำออกเดิมของส่วนประกอบของแบบจำลองกล่องดำที่กำหนด โดยที่รูปแบบของระบบเปลี่ยนสถานะนำเข้า/นำออกเป็นฐานหลักที่เหมาะสมสำหรับการทวนสอบด้วยการตรวจสอบแบบจำลอง ผลที่ได้รับคือสามารถทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองกล่องดำที่สามารถแปลงเป็นระบบเปลี่ยนสถานะนำเข้า/นำออกที่มีสถานะจำนวนจำกัดด้วยการตรวจสอบแบบจำลอง แต่วิธีการนี้สามารถทวนสอบแบบจำลองบนพื้นฐานของกระแสเฉพาะราย (Individual stream-based model)

เท่านั้น ดังแสดงในรูปที่ 1.2 ไม่รวมถึงแบบจำลองประกอบ (Composite model) ดังแสดงในรูปที่ 1.3 ที่แสดงการสื่อสารกันระหว่างส่วนประกอบในระบบ

รูปที่ 1.2 แบบจำลองบนพื้นฐานของกระแสเฉพาะราย



รูปที่ 1.3 แบบจำลองประกอบ

งานวิจัยนี้จึงนำเสนอระเบียบวิธีการทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบ ที่แสดงโครงสร้างการสื่อสารของส่วนประกอบด้วยการตรวจสอบแบบจำลอง สำหรับการทวนสอบแบบจำลองประกอบที่ได้รับมาจากการประกอบแบบจำลองบนพื้นฐานของกระแสเฉพาะรายด้วยตัวดำเนินการประกอบ (Composition operator) การทวนสอบแบบจำลองประกอบด้วยการตรวจสอบแบบจำลอง สามารถบ่งชี้ความถูกต้องของการสร้างแบบจำลองตามสมบัตินำเข้า/นำออกที่กำหนดไว้

1.2 วัตถุประสงค์ของการวิจัย

เพื่อออกแบบระเบียบวิธีสำหรับการประกอบแบบจำลองบนพื้นฐานของกระแสด้วยตัวดำเนินการประกอบแบบลำดับเป็นแบบจำลองประกอบ และกำหนดภาวะนามธรรมประวัติที่สามารถทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบด้วยวิธีการตรวจสอบแบบจำลอง

1.3 ขอบเขตของการวิจัย

1) ระเบียบวิธีที่นำเสนอสำหรับการทวนสอบของแบบจำลองประกอบที่เกิดจากการประกอบแบบจำลองบนพื้นฐานของกระแสโดยใช้ตัวดำเนินการประกอบชนิด การประกอบแบบลำดับ เท่านั้น

2) แบบจำลองประกอบที่สามารถแปลงเป็นแบบจำลองบนพื้นฐานของสถานะที่มีสถานะจำกัดเท่านั้นที่สามารถทวนสอบได้ด้วยระเบียบวิธีที่นำเสนอ

3) แบบจำลองประกอบที่ใช้สำหรับการแสดงในวิทยานิพนธ์นี้คือ แบบจำลองประกอบที่ประกอบด้วยแบบจำลองบนพื้นฐานของกระแสเฉพาะรายครั้งละสองส่วนประกอบ

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

4) ข้อกำหนดสมบัตินำเข้า/นำออกของส่วนประกอบที่ยังไม่ประกอบ และส่วนประกอบที่ประกอบแล้วอยู่ในรูปของสูตรตรรกแอลทีแอลเท่านั้น

5) การทวนสอบที่นำเสนอใช้ตัวตรวจสอบแบบจำลองเอ็นยูเอสเอ็มวี

6) ทำการทดลองทวนสอบและกำหนดฟังก์ชันภาวะนามธรรมประวัติกับแบบจำลองประกอบที่ได้รับจากการประกอบแบบจำลองบนพื้นฐานของกระแสด้วยตัวดำเนินการประกอบชนิดการประกอบแบบลำดับรูปแบบต่างๆ อย่างน้อยรูปแบบละ 1 ระบบ

7) ทำการทดสอบสมบัติการเปลี่ยนหมู่ (Associative property) ของตัวดำเนินการประกอบแบบลำดับอย่างน้อย 1 ระบบ

8) ระเบียบวิธีที่นำเสนอสามารถทวนสอบแบบจำลองประกอบที่มีหนึ่งช่องทางนำเข้าและหนึ่งช่องทางนำออกเท่านั้น

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1) ได้วิธีการประกอบแบบจำลองบนพื้นฐานของกระแสเป็นแบบจำลองประกอบด้วยตัวดำเนินการประกอบแบบลำดับ ซึ่งสามารถนำไปทวนสอบสมบัตินำเข้า/นำออกด้วยการตรวจสอบแบบจำลองได้

2) ได้วิธีการกำหนดฟังก์ชันภาวะนามธรรมประวัติของแบบจำลองประกอบ ซึ่งทำให้สามารถลดจำนวนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกได้

3) เป็นแนวทางสำหรับการทวนสอบแบบจำลองกล่องดำที่ได้จากการออกแบบระบบโดยวิธีการอื่น

1.5 วิธีดำเนินการวิจัย

1) ศึกษาข้อมูลและเอกสารการวิจัยที่เกี่ยวกับวิธีการพัฒนาบนพื้นฐานของกระแส

2) ศึกษาการทวนสอบโดยใช้วิธีการตรวจสอบแบบจำลอง

3) ศึกษาการทวนสอบแบบจำลองบนพื้นฐานของกระแสเฉพาะราย

4) ศึกษาการประกอบแบบจำลองประกอบของแบบจำลองบนพื้นฐานของกระแส

5) ศึกษาตัวตรวจสอบแบบจำลองเอ็นยูเอสเอ็มวี

6) เสนอระเบียบวิธีสำหรับการทวนสอบแบบจำลองประกอบของแบบจำลองบนพื้นฐานของกระแส

7) ทำการทดลองและสรุปผลการวิจัย

8) เขียนผลงานวิจัยเพื่อตีพิมพ์

9) เรียบเรียงและเขียนวิทยานิพนธ์

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

1.6 โครงสร้างของเนื้อหางานวิจัย

เนื้อหาโครงสร้างของวิทยานิพนธ์ฉบับนี้ถูกแบ่งออกเป็น 6 บทดังนี้ คือ บทที่ 1 เป็นบทนำ บทที่ 2 กล่าวถึงทฤษฎีและงานวิจัยต่างๆ ที่เกี่ยวข้อง บทที่ 3 นำเสนอระเบียบวิธีการทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบด้วยวิธีการตรวจสอบแบบจำลอง บทที่ 4 นำเสนอฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองบนพื้นฐานของกระแสเฉพาะราย ซึ่งจะนำไปใช้ในการกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบ บทที่ 5 นำเสนอวิธีการกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบ พิจารณาความสัมพันธ์ของฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบกับฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองบนพื้นฐานของกระแสเฉพาะราย และนำเสนอวิธีการกำหนดสมบัตินำเข้า/นำออกที่คาดหมายของแบบจำลองประกอบจากสมบัตินำเข้า/นำออกที่คาดหมายของแบบจำลองบนพื้นฐานของกระแสเฉพาะราย และบทที่ 6 ซึ่งเป็นบทสุดท้ายจะเป็นบทสรุปของงานวิจัย ข้อจำกัด รวมทั้งข้อเสนอแนะงานวิจัยในอนาคต และบทความวิชาการที่ตีพิมพ์

1.7 นิยาม

แบบจำลองบนพื้นฐานของกระแสเฉพาะราย หมายถึง ส่วนประกอบเดี่ยวที่สามารถจำลองพฤติกรรมด้วยฟังก์ชันการประมวลผลกระแส

แบบจำลองประกอบ หมายถึง ส่วนประกอบที่เกิดจากการประกอบแบบจำลองบนพื้นฐานของกระแสเฉพาะรายตั้งแต่ 2 ส่วนประกอบย่อยขึ้นไปด้วยตัวดำเนินการระกอบ

ส่วนประกอบ หมายถึง แบบจำลองบนพื้นฐานของกระแสเฉพาะราย ในกรณีที่ส่วนประกอบไม่สามารถแบ่งเป็นส่วนประกอบย่อย หรือแบบจำลองประกอบ ในกรณีที่ส่วนประกอบสามารถแบ่งเป็นส่วนประกอบย่อย

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

บทนี้ จะกล่าวถึงทฤษฎีที่สำคัญ ที่นำมาประยุกต์ สนับสนุน และใช้อ้างอิงในการทำวิทยานิพนธ์ รวมถึงข้อดีและข้อจำกัดของงานวิจัยต่างๆ ที่เกี่ยวข้อง โดยมีเนื้อหาดังต่อไปนี้

2.1 แนวคิดและทฤษฎี

ตอนนี้นำเสนอทฤษฎีสามตอนหลักคือ ตอนที่ 2.1.1 ทฤษฎีที่เกี่ยวข้องกับกระแสและการพัฒนาบนพื้นฐานของกระแส ตอนที่ 2.1.2 ทฤษฎีที่เกี่ยวข้องกับการตรวจสอบแบบจำลอง และตอนที่ 2.1.3 ทฤษฎีที่เกี่ยวข้องกับระบบเปลี่ยนสถานะนำเข้า/นำออก ซึ่งเป็นความสัมพันธ์ระหว่างการพัฒนาบนพื้นฐานของกระแสและการตรวจสอบแบบจำลอง

2.1.1 กระแสและการพัฒนาบนพื้นฐานของกระแส

ทฤษฎีที่เกี่ยวกับการพัฒนาบนพื้นฐานของกระแส แบ่งเป็นสี่ตอนคือ ตอนที่ 2.1.1.1 ทฤษฎีที่เกี่ยวข้องกับกระแส ตอนที่ 2.1.1.2 ทฤษฎีที่เกี่ยวข้องกับฟังก์ชันการประมวลผลกระแส ซึ่งเป็นแบบจำลองพื้นฐานของกระแสเฉพาะราย ตอนที่ 2.1.1.3 ทฤษฎีที่เกี่ยวข้องกับการประกอบของฟังก์ชันการประมวลผลกระแส ซึ่งเป็นการสร้างแบบจำลองประกอบ และตอนที่ 2.1.1.4 ทฤษฎีที่เกี่ยวข้องกับเครื่องเปลี่ยนสถานะ ซึ่งเป็นแบบจำลองเป้าหมายของการพัฒนาบนพื้นฐานของกระแส

2.1.1.1 กระแส

กระแส แทนลำดับของข้อมูลที่สืบต่อเชิงเวลาในเครือข่ายที่ส่วนประกอบสื่อสารกัน [7, 4]

นิยาม 2.1 กระแสจำกัด (Finite stream) กระแสว่าง (Empty stream) และกระแสไม่ว่าง (Non-empty stream) [7, 4]

กำหนดให้ $\mathcal{A}$ เป็นเซตข้อมูลไม่ว่าง (Non-empty data set) นั่นคือ $\mathcal{A}$ ต้องมีสมาชิกอย่างน้อยหนึ่งตัว และ $\langle\rangle$ แทนกระแสว่างคือ กระแสที่ไม่มีสมาชิก

กระแสจำกัดคือ กระแสที่มีจำนวนสมาชิกเป็นจำนวนเต็มบวก หรือศูนย์

เซตของกระแสจำกัดบน $\mathcal{A}$ นิยามโดย

$$\mathcal{A}^* = \{\langle\rangle\} \cup \mathcal{A} \times \mathcal{A}^*$$

กระแสจำกัด $X \in \mathcal{A}^*$ สามารถแจกแจงสมาชิกของ X ได้ดังนี้

$X = \langle x_1, x_2, ..., x_m \rangle$ โดยที่ความยาวของกระแส $|X| = m$ เมื่อ $m \geq 0$ และ $x_i \in \mathcal{A} (i \in [1, m])$

กระแสไม่ว่างคือ กระแสที่มีจำนวนสมาชิกเป็นจำนวนเต็มบวก

เซตของกระแสไม่ว่าง บน $\mathcal{A}$ นิยามโดย

$$\mathcal{A}^{+} = \mathcal{A}^{*} \setminus \{\langle\rangle\}$$

นิยาม 2.2 การต่อกระแส (Stream concatenation) [7]

การต่อกระแสคือ การนำกระแสสองกระแสมาเชื่อมต่อกัน

การต่อกระแส นิยามโดย

$$.\&.: \mathcal{A}^{*} \times \mathcal{A}^{*} \to \mathcal{A}^{*}$$

เมื่อ กำหนดให้ $X = \langle x_1, x_2, ..., x_m \rangle$ และ $Y = \langle y_1, y_2, ..., y_n \rangle$ การต่อกระแสจำกัด X และกระแสจำกัด Y คือ $X \,\&\, Y = \langle x_1, x_2, ..., x_m, y_1, y_2, ..., y_n \rangle$

นิยาม 2.3 การตัดกระแส (Stream deletion) [8]

การตัดกระแสคือ การลบสมาชิกที่ด้านหน้าของกระแสเท่ากับจำนวนเต็มบวกที่กำหนด

การตัดกระแส นิยามโดย

$$.-.: \mathcal{A}^{*} \times \mathbb{N} \to \mathcal{A}^{*}$$

เมื่อ $\langle x_1, x_2, ..., x_{k+l} \rangle - k = \langle x_{k+1}, x_{k+2}, ..., x_{k+l} \rangle$

โดยที่ $\mathbb{N}$ คือ จำนวนธรรมชาติ

นิยาม 2.4 เติมหน้า (Prefix) [9, 7]

เติมหน้าคือ ความสัมพันธ์ของกระแสสองกระแส โดยที่กระแสแรกมีสมาชิกเหมือนกับกระแสหลังเมื่อเปรียบเทียบที่ตำแหน่งเดียวกัน และกระแสแรกจะต้องมีความยาวไม่มากกว่ากระแสหลัง

นิยามโดย กระแสจำกัด X เรียกว่าเติมหน้าของกระแส Y แทนด้วย $X \sqsubseteq Y$ ก็ต่อเมื่อมีกระแส Z โดยที่ $X \,\&\, Z = Y$

เซต $\mathcal{A}^{*}$ ของกระแสจำกัดบน $\mathcal{A}$ เป็นลำดับบางส่วน (Partial order) ภายใต้ความสัมพันธ์เติมหน้า (Prefix relation) $(\mathcal{A}^{*}, \sqsubseteq)$ โดยมี $\langle\rangle$ เป็นสมาชิกที่น้อยที่สุด

นิยาม 2.5 การลบกระแส (Stream subtraction) [7]

การลบกระแสคือ การลบกระแสส่วนที่เป็นเติมหน้าออกจากกระแสที่เป็นภาคขยาย

การลบกระแส นิยามโดย

$$.\ominus.: \mathcal{A}^{*} \times \mathcal{A}^{*} \to \mathcal{A}^{*}$$

เมื่อ $Z \ominus X = Y$ ก็ต่อเมื่อ $X \,\&\, Y = Z$ เป็นจริง

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

นิยาม 2.6 การดำเนินการเติมหน้า (Prefixing operation) และการดำเนินการเติมหลัง (Postfixing operation) [7]

การดำเนินการเติมหน้าคือ การเติมสมาชิกเข้าไปด้านหน้าของกระแส

การดำเนินการเติมหน้า นิยามโดย

$$.\lhd .: \mathcal{A} \times \mathcal{A}^* \to \mathcal{A}^*$$

เมื่อ $x \lhd X = \langle x \rangle \,\&\, X$

การดำเนินการเติมหลังคือ การเติมสมาชิกเข้าไปด้านหลังของกระแส

การดำเนินการเติมหลัง นิยามโดย

$$.\rhd .: \mathcal{A}^* \times \mathcal{A} \to \mathcal{A}^*$$

เมื่อ $X \rhd x = X \,\&\, \langle x \rangle$

นิยาม 2.7 การดำเนินการตัวแรก (First operation) การดำเนินการตัวหลัง (Last operation) การดำเนินการยกเว้นตัวแรก (Rest operation) และการดำเนินการยกเว้นตัวหลัง (Lead operation) [4]

การดำเนินการตัวแรกคือ การตัดให้เหลือเฉพาะสมาชิกตัวแรกของกระแสไม่ว่าง

การดำเนินการตัวแรก นิยามโดย

$$first: \mathcal{A}^+ \to \mathcal{A}$$

เมื่อ $first(x \lhd X) = x$

การดำเนินการตัวหลังคือ การตัดให้เหลือเฉพาะสมาชิกตัวหลังของกระแสไม่ว่าง

การดำเนินการตัวหลัง นิยามโดย

$$last: \mathcal{A}^+ \to \mathcal{A}$$

เมื่อ $last(X \rhd x) = x$

การดำเนินการยกเว้นตัวแรกคือ การตัดสมาชิกตัวแรกของกระแสไม่ว่างออก

การดำเนินการยกเว้นตัวแรก นิยามโดย

$$rest: \mathcal{A}^+ \to \mathcal{A}^*$$

เมื่อ $rest(x \lhd X) = X$

การดำเนินการยกเว้นตัวหลังคือ การตัดสมาชิกตัวหลังของกระแสไม่ว่างออก

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

การดำเนินการยกเว้นตัวหลัง นิยามโดย

$$lead : \mathcal{A}^{+} \to \mathcal{A}^{*}$$

เมื่อ $lead(X \triangleright x) = X$

2.1.1.2 ฟังก์ชันการประมวลผลกระแส

ประวัติของข้อมูลที่ส่งผ่านช่องการสื่อสาร (Communication channel) ระหว่างส่วนประกอบที่กำหนดและส่วนประกอบอื่น หรือระหว่างส่วนประกอบที่กำหนดและสิ่งแวดล้อม สามารถอธิบายด้วยแนวคิดของกระแส ส่วนประกอบเชิงกำหนดซึ่งประมวลผลข้อมูลอย่างต่อเนื่องจากช่องทางนำเข้า ไปยังช่องทางนำออก สามารถพิจารณาเหมือนกับฟังก์ชันการประมวลผลกระแสที่ส่งประวัติของนำเข้าไปยังประวัติของนำออก

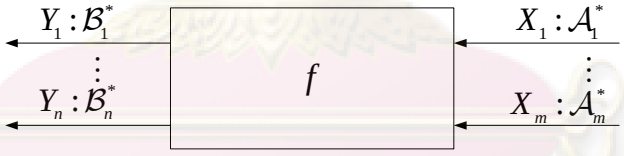
นิยาม 2.8 ฟังก์ชันการประมวลผลกระแส [4]

ฟังก์ชันการประมวลผลกระแสคือ ฟังก์ชันที่ส่งจากทูเพิล (Tuple) ของกระแสนำเข้าไปยังทูเพิลของกระแสนำออก

ฟังก์ชันการประมวลผลกระแส นิยามโดย

$$f : \mathcal{A}_1^{*} \times ... \times \mathcal{A}_m^{*} \to \mathcal{B}_1^{*} \times ... \times \mathcal{B}_n^{*}$$

เมื่อ f เป็นฟังก์ชันการประมวลผลกระแสที่กำหนดด้วย m ช่องนำเข้า และ n ช่องนำออก



รูปที่ 2.1 ส่วนประกอบจำลองด้วยฟังก์ชันการประมวลผลกระแส f [4]

ในรูปที่ 2.1 แสดงส่วนประกอบที่กำหนดด้วย m ช่องนำเข้า และ n ช่องนำออก สามารถจำลองด้วยฟังก์ชันการประมวลผลกระแส f ที่มีแบบชนิดนำเข้า $(\mathcal{A}_1, ..., \mathcal{A}_m)(m \geq 0)$ และแบบชนิดนำออก $(\mathcal{B}_1, ..., \mathcal{B}_n)$ $(n \geq 0)$ เป็นตัวกำหนดลักษณะตัวต่อประสาน (Interface) ของส่วนประกอบ

นิยาม 2.9 ทางเดียวเติมหน้า (Prefix monotonic) [7]

ทางเดียวเติมหน้าคือ ฟังก์ชันการประมวลผลกระแสที่เมื่อได้รับนำเข้าใหม่ แล้วนำออกจะต้องเป็นภาคขยายของนำออกเดิม หรือเท่ากับนำออกเดิมเท่านั้น

นิยามโดย ฟังก์ชันการประมวลผลกระแส $f : \mathcal{A}^{*} \to \mathcal{B}^{*}$ เป็นฟังก์ชันทางเดียวเติมหน้า ถ้า $X \sqsubseteq Y$ แล้ว $f(X) \sqsubseteq f(Y)$ สำหรับทุก $X, Y \in \mathcal{A}^{*}$

การจำลองบนพื้นฐานของกระแส (Stream-based modeling) สำหรับส่วนประกอบต้องมีฟังก์ชันการประมวลผลกระแสทางเดียวเติมหน้าที่ใช้แนวคิดของเหตุภาพ (Causality) ดังนั้นนำเข้าในอนาคตไม่สามารถยกเลิกหรือเปลี่ยนนำออกก่อนหน้า การขยายของประวัติของนำเข้า (Input history) จะมีผลสำหรับภาคขยายของประวัติของนำออก (Output history) เท่านั้น

นิยาม 2.10 ความต่อเนื่อง (Continuity) [4]

ฟังก์ชันต่อเนื่องคือ ฟังก์ชันการประมวลผลกระแสที่กระแสนำออกที่เกิดกระแสนำเข้าอนันต์ประมาณได้จากกระแสนำออกของเติมหน้าของกระแสนำเข้าอนันต์นั้น

นิยามโดย ฟังก์ชันทางเดียว (Monotonic function)

$f : \mathcal{A}_1^\infty \times ... \times \mathcal{A}_m^\infty \to \mathcal{B}_1^\infty \times ... \times \mathcal{B}_n^\infty$ บนกระแส เรียกว่าฟังก์ชันต่อเนื่อง ถ้า

$$f(\bigcup D) = \bigcup f(D)$$

สำหรับทุกเซตระบุทิศทาง (Directed set) $\mathcal{D} \subseteq \mathcal{A}_1^\infty \times ... \times \mathcal{A}_m^\infty$ เมื่อ $\bigcup$ แทน ขอบเขตบนน้อยสุด (Least upper bound)

2.1.1.2.1 **ตัวอย่างฟังก์ชันการประมวลผลกระแส**

ตัวอย่างที่ 2.1 ส่วนประกอบสำเนา (Duplicate component) [4]

ส่วนประกอบสำเนาเป็นส่วนประกอบที่สำเนาสมาชิกแต่ละตัวของกระแสนำเข้าสามารถเขียนแทนด้วยฟังก์ชันการประมวลผลกระแส นิยามดังนี้

$$duplicate : \mathcal{A}^* \to \mathcal{A}^*$$

โดยที่

$$duplicate(\langle\rangle) = \langle\rangle \quad (2.1)$$

$$duplicate(x \lhd X) = \langle x, x\rangle \,\&\, duplicate(X) \quad (2.2)$$

เมื่อ $x \in \mathcal{A}$ และ $X \in \mathcal{A}^*$

(2.1) หมายถึง กระแสนำเข้าว่างก่อกำเนิดกระแสนำออกว่าง

(2.2) หมายถึง สำเนาสมาชิกแต่ละตัวของกระแสนำเข้าเป็นสมาชิกของกระแสนำออกที่เหมือนกันสองตัว

ตัวอย่างที่ 2.2 ส่วนประกอบเรจิสเตอร์การเลื่อน (Shift register component) [4]

ส่วนประกอบเรจิสเตอร์การเลื่อนเป็นส่วนประกอบที่หน่วงกระแสนำเข้าด้วยการกำหนดจำนวนของขั้นตอนที่ต้องการหน่วง นิยามดังนี้

$$shift : \mathbb{N} \setminus \{0\} \to [\mathcal{B}^* \to \mathcal{B}^*]$$

โดยที่

$$shift(n)(\langle\rangle) = \langle\rangle \tag{2.3}$$

$$shift(n)(x \lhd X) \tag{2.4}$$

$$= \begin{cases} \langle\rangle & if\ |x \lhd X| \le n \\ x \lhd shift(n)(X)\ if\ |x \lhd X| > n \end{cases} \tag{2.5}$$

(2.3) หมายถึง ฟังก์ชัน *shift* รอคอยนำเข้า

(2.4) หมายถึง ไม่มีการปล่อยนำออก ถ้าจำนวนนำเข้าของฟังก์ชัน *shift* น้อยกว่าหรือเท่ากับ n

(2.5) หมายถึง ปล่อยนำเข้าตัวแรกเป็นนำออก ถ้าจำนวนนำเข้าที่เก็บไว้ของฟังก์ชัน *shift* มากกว่า n และส่วนนำเข้าที่เหลือยกเว้นสมาชิกตัวแรกนำไปประมวลผลผ่านฟังก์ชัน *shift* ต่อ

ตัวอย่างที่ 2.3 ส่วนประกอบหน่วยความจำ (Memory component) [4]

ส่วนประกอบหน่วยความจำมีพฤติกรรมคือ หน่วยความจำจะเก็บกระแสของข้อมูลที่ได้รับจากคำสั่งเขียนไว้และส่งเป็นนำออกเป็นกระแสของข้อมูลที่ได้รับจากคำสั่งเขียนหนึ่งครั้งเมื่อได้รับคำสั่งอ่านหนึ่งครั้ง สามารถเขียนแทนด้วยฟังก์ชันการประมวลผลกระแส นิยามดังนี้

$$memory : (w(\mathcal{D}) \cup \{r\})^* \to \mathcal{D}^*$$

โดยที่

$$memory(\langle\rangle) = \langle\rangle \tag{2.6}$$

$$memory(w(d)) = \langle\rangle \tag{2.7}$$

$$memory(r \lhd X) = memory(X) \tag{2.8}$$

$$memory(W \,\&\, r \lhd X) = W \,\&\, memory(W \,\&\, X) \tag{2.9}$$

เมื่อ $w(d) \in w(\mathcal{D})$, $W \in (w(\mathcal{D}))^*$ และ $X \in (w(\mathcal{D}) \cup \{r\})^*$

(2.6) หมายถึง กระแสนำเข้าว่างก่อกำเนิดกระแสนำออกว่าง

(2.7) หมายถึง กระแสนำเข้าเดี่ยวที่เป็นคำสั่งเขียนก่อกำเนิดกระแสนำออกว่าง

(2.8) หมายถึง เมื่อได้รับคำสั่งอ่านโดยที่ไม่มีข้อมูลในหน่วยความจำจะไม่มีนำออก

(2.9) หมายถึง ข้อมูลที่เก็บเอาไว้ดึงออกมาด้วยคำสั่งอ่าน

ตัวอย่างที่ 2.4 ส่วนประกอบตัวกรองกำหนดเอง (Self-initializing filter) [4]

ส่วนประกอบตัวกรองกำหนดเองมีพฤติกรรมคือ กรองกระแสนำเข้าที่ยกเว้นสมาชิกตัวแรกด้วยสมาชิกตัวแรกของกระแส สามารถเขียนแทนด้วยฟังก์ชันการประมวลผลกระแส นิยามดังนี้

$$filterinit : \mathcal{A}^* \to \mathcal{A}^*$$

โดยที่

$$filterinit(\langle\rangle) = \langle\rangle \tag{2.10}$$

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

$$filterinit(\langle x\rangle) = \langle\rangle \tag{2.11}$$

$$filterinit(x \lhd y \lhd X) = \begin{cases} y \lhd filterinit(x \lhd X) \, if \; x = y \\ filterinit(x \lhd X) \, if \; x \neq y \end{cases} \tag{2.12}$$

เมื่อ $x, y \in \mathcal{A}$ และ $X \in \mathcal{A}^*$

(2.10) หมายถึง กระแสนำเข้าว่างก่อกำเนิดกระแสนำออกว่าง

(2.11) หมายถึง กระแสนำเข้าเดี่ยวก่อกำเนิดกระแสนำออกว่าง

(2.12) หมายถึง สมาชิกแต่ละตัวของกระแสนำเข้ายกเว้นสมาชิกตัวแรกที่เหมือนกับสมาชิกตัวแรกจะถูกส่งไปเป็นสมาชิกของกระแสนำออก

ตัวอย่างที่ 2.5 ส่วนประกอบผลบวก (Summation component) [4]

ส่วนประกอบผลบวกมีพฤติกรรมคือ ทำการบวกสมาชิกแต่ละในกระแสนำเข้าที่เป็นจำนวนธรรมชาติสะสมไว้เพื่อก่อกำเนิดนำออกสามารถเขียนแทนด้วยฟังก์ชันการประมวลผลกระแส ดังนี้

$$sum : \mathbb{N} \times [\mathbb{N} \times \mathbb{N} \to \mathbb{N}] \to [\mathbb{N}^* \to \mathbb{N}^*]$$

โดยที่

$$sum(s, +)(\langle\rangle) = \langle s\rangle \tag{2.13}$$

$$sum(s, +)(x \lhd X) = (s + x) \lhd sum(s + x, +)(X) \tag{2.14}$$

เมื่อ $s, x \in \mathbb{N}$ $X \in \mathbb{N}^*$ และ กำหนดสถานะเริ่มต้น $s = 0$

(2.13) หมายถึง กระแสนำเข้าว่างก่อกำเนิดกระแสนำออกที่เป็นค่าเริ่มต้น

(2.14) หมายถึง สมาชิกแต่ละตัวของกระแสนำเข้าถูกบวกสะสมเก็บไว้เพื่อก่อกำเนิดนำออก

ตัวอย่างที่ 2.6 ส่วนประกอบตำแหน่ง (Positions component) [4]

ส่วนประกอบตำแหน่งรับกระแสนำเข้าที่บรรจุหมายเลขตำแหน่ง แต่ละตำแหน่งในกระแสนำเข้าจะนำออกค่าในนี้ในกระแสนำเข้าที่กำหนดถ้ามีอยู่จริง นิยามดังนี้

$$positions : \mathbb{N}^* \to \mathbb{N}^*$$

โดยที่

$$positions(X) = history(\langle\rangle)(X) \tag{2.15}$$

เมื่อ

$$history : \mathbb{N}^* \to [\mathbb{N}^* \to \mathbb{N}^*]$$

โดยที่

$$history(X)(\langle\rangle) = \langle\rangle \tag{2.16}$$

$$history(X)(x \lhd Y) = lookup(X, x) \,\&\, history(X \lhd X)(Y) \tag{2.17}$$

เมื่อ

$$lookup : \mathbb{N}^* \times \mathbb{N} \to \mathbb{N}^*$$

โดยที่

$$lookup(\langle\rangle, n) = \langle\rangle \tag{2.18}$$

$$lookup(x \lhd X, n) = \begin{cases} \langle x\rangle & if \; n = 0 \\ lookup(X, n-1) \; if \; n > 0 \end{cases} \tag{2.19}$$

เมื่อ $n, x, y \in \mathbb{N}$ และ $X, Y \in \mathbb{N}^*$

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

(2.15) หมายถึง พฤติกรรมของฟังก์ชัน *positions* เหมือนกับพฤติกรรมของฟังก์ชัน *history* ที่ส่งพารามิเตอร์ของฟังก์ชัน *positions* ให้กับพารามิเตอร์ตัวหลังของฟังก์ชัน *history* โดยกำหนดให้พารามิเตอร์ตัวแรกเป็นกระแสว่าง

(2.16) หมายถึง กระแสนำเข้าว่างของฟังก์ชัน *history* ก่อกำเนิดกระแสนำออกว่าง

(2.17) หมายถึง สมาชิกแต่ละตัวของกระแสนำเข้าของฟังก์ชัน *history* จะนำไปค้นหาตำแหน่งข้อมูลจากกระแสภายในที่เกิดจากสมาชิกของกระแสนำเข้าที่ค้นหาข้อมูลเสร็จแล้ว โดยฟังก์ชัน *lookup*

(2.18) หมายถึง กระแสนำเข้าว่างของฟังก์ชัน *lookup* ก่อกำเนิดกระแสนำออกว่าง

(2.19) หมายถึง ฟังก์ชัน *lookup* ค้นหาข้อมูลที่มีตำแหน่งเดียวกับที่กำหนดแล้วส่งเป็นนำออกหรือก่อกำเนิดนำออกว่างหากไม่มีข้อมูลในตำแหน่งนั้น

2.1.1.3 การประกอบของฟังก์ชันการประมวลผลกระแส

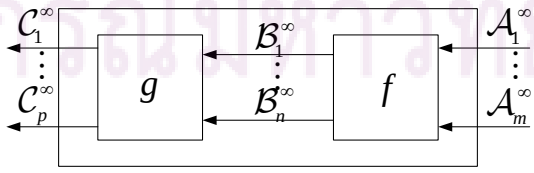
แบบจำลองประกอบ [10, 9, 3] สร้างจากโครงสร้างการสื่อสารของส่วนประกอบ โดยที่พฤติกรรมของแบบจำลองประกอบ สามารถอนุมานจากพฤติกรรมของส่วนประกอบ การประกอบฟังก์ชันการประมวลผลกระแสสามารถทำได้โดยใช้ตัวดำเนินการประกอบทั้งสามชนิดคือ การประกอบแบบลำดับ การประกอบแบบขนาน และผลป้อนกลับ

2.1.1.3.1 การประกอบแบบลำดับ

การประกอบแบบลำดับเกิดขึ้นเมื่อช่องทางนำออกของส่วนประกอบหนึ่งถูกเชื่อมต่อไปยังช่องทางนำเข้าทั้งหมดของส่วนประกอบอื่น โดยที่พฤติกรรมของระบบประกอบได้มาจากการประกอบฟังก์ชันการประมวลผลกระแสของทั้งสองส่วนประกอบ

นิยาม 2.11 การประกอบแบบลำดับ [4]

การประกอบแบบลำดับของส่วนประกอบ $f : \mathcal{A}_1^\infty \times ... \times \mathcal{A}_m^\infty \rightarrow \mathcal{B}_1^\infty \times ... \times \mathcal{B}_n^\infty$ และส่วนประกอบ $g : \mathcal{B}_1^\infty \times ... \times \mathcal{B}_n^\infty \rightarrow \mathcal{C}_1^\infty \times ... \times \mathcal{C}_p^\infty$ ด้วยตัวต่อประสานที่เข้ากันได้สามารถแทนโดย $f;g : \mathcal{A}_1^\infty \times ... \times \mathcal{A}_m^\infty \rightarrow \mathcal{C}_1^\infty \times ... \times \mathcal{C}_p^\infty$ โดยที่ $f;g = g \circ f$ เมื่อ ตัวดำเนินการ $\circ$ แทนการประกอบฟังก์ชัน $(g \circ f)(X_1,...,X_m) = g(f(X_1,...,X_m))$ แบบจำลองประกอบ $f;g$ สามารถพิจารณาในฐานะที่เป็นส่วนประกอบใหม่ในรูปที่ 2.2

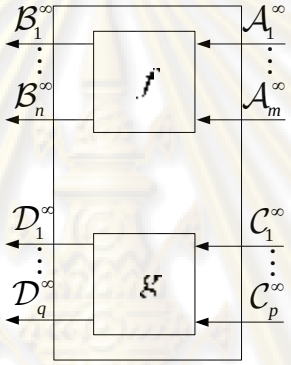


รูปที่ 2.2 การประกอบแบบลำดับของ f และ g [4]

การประกอบแบบลำดับของฟังก์ชันทางเดียว (Monotonic function) หรือฟังก์ชันต่อเนื่อง(Continuous function) จะได้ผลลัพธ์เป็นฟังก์ชันทางเดียว หรือ ฟังก์ชันต่อเนื่องตามลำดับ

2.1.1.3.2 การประกอบแบบขนาน

ส่วนประกอบที่ทำงานอย่างอิสระกัน ถูกประกอบโดยการประกอบแบบขนาน โดยการเรียงช่องนำเข้า (Input channel) และช่องนำออก (Output channel) ของทั้งสองส่วนประกอบดังแสดงในรูปที่ 2.3 และพฤติกรรมของแบบจำลองประกอบได้รับมาจากพฤติกรรมของทั้งสองส่วนประกอบ



รูปที่ 2.3 การประกอบแบบขนานของ f และ g [4]

นิยาม 2.12 การประกอบแบบขนาน [4]

การประกอบแบบขนานของส่วนประกอบ $f : \mathcal{A}_1^\infty \times ... \times \mathcal{A}_m^\infty \rightarrow \mathcal{B}_1^\infty \times ... \times \mathcal{B}_n^\infty$ และส่วนประกอบ $g : \mathcal{C}_1^\infty \times ... \times \mathcal{C}_p^\infty \rightarrow \mathcal{D}_1^\infty \times ... \times \mathcal{D}_q^\infty$ สามารถแทนโดย $f \parallel g : \mathcal{A}_1^\infty \times ... \times \mathcal{A}_m^\infty \times \mathcal{C}_1^\infty \times ... \times \mathcal{C}_p^\infty \rightarrow \mathcal{B}_1^\infty \times ... \times \mathcal{B}_n^\infty \times \mathcal{D}_1^\infty \times ... \times \mathcal{D}_q^\infty$ โดยที่ $(f \parallel g)(X_1, ..., X_{m+p}) = f(X_1, ..., X_m) \wedge g(X_{m+1}, ..., X_{m+p})$ เมื่อ ตัวดำเนินการ ^ แทนการต่อกันของทูเพิล (Tuple) $(R_1, ..., R_n) \wedge (S_1, ..., S_q) = (R_1, ..., R_n, S_1, ..., S_q)$

การประกอบแบบขนานของฟังก์ชันทางเดียว หรือ ฟังก์ชันต่อเนื่อง จะได้ผลลัพธ์เป็นฟังก์ชันทางเดียว หรือ ฟังก์ชันต่อเนื่อง ตามลำดับ

2.1.1.3.3 ผลป้อนกลับ

ตัวดำเนินการผลป้อนกลับในรูปที่ 2.4 ถูกนิยามด้วยตัวดำเนินการจุดตรึงน้อยที่สุด (Least fixpoint operator)

นิยาม 2.13 ผลป้อนกลับ [4]

ผลป้อนกลับเชื่อมต่อส่วนหลังจำนวน $k \geq 1$ ช่องของช่องนำออกของส่วนประกอบไปยังส่วนหลังของช่องนำเข้าของส่วนประกอบนั้น $f : \mathcal{A}_1^\infty \times ... \times \mathcal{A}_m^\infty \times$

$\mathcal{B}_1^\infty \times ... \times \mathcal{B}_k^\infty \to \mathcal{C}_1^\infty \times ... \times \mathcal{C}_n^\infty \times \mathcal{B}_1^\infty \times ... \times \mathcal{B}_k^\infty$ โดยที่พฤติกรรมของระบบนี้ถูกระบุโดยตัวดำเนินการผลป้อนกลับ (Feedback operator) $\circlearrowleft_k f : \mathcal{A}_1^\infty \times ... \times \mathcal{A}_m^\infty \to \mathcal{C}_1^\infty \times ... \times \mathcal{C}_n^\infty$ นิยามโดย

$$(\circlearrowleft_k f)(X_1, ..., X_m) = \Pi_{1,...,n} f(X_1, ..., X_m, Z_1, ..., Z_k)$$

โดยที่ $\Pi_{i,...,j}$ แทนการฉาย (Projection) บนส่วนประกอบ เริ่มต้นจากทูเพิล i ถึงทูเพิล j และโดยที่ $(Z_1, ..., Z_k)$ คือ จุดตรึงน้อยที่สุดของสมการ

$$(Z_1, ..., Z_k) = \Pi_{n+1,...,n+k} f(X_1, ..., X_m, Z_1, ..., Z_k)$$

เนื่องจากฟังก์ชันการประมวลผลกระแสเป็นฟังก์ชันต่อเนื่องบนลำดับบางส่วนแบบสมบูรณ์ ดังนั้นจุดตรึงน้อยที่สุด ซึ่งถูกทำซ้ำจากสมาชิกน้อยที่สุด (Least element) ภายใต้ลำดับบางส่วนสมบูรณ์ก็คือ k ทูเพิลของกระแสว่าง (Empty stream)

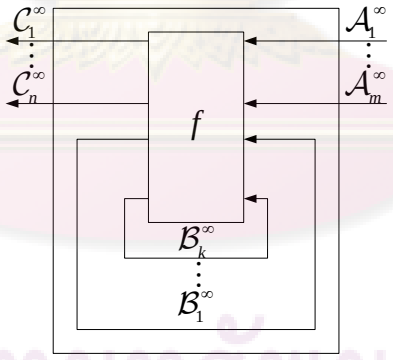
$$(\circlearrowleft_k f)(X_1, ..., X_m) = \bigcup_{i>0} (Y_1^i, ..., Y_n^i)$$

โดยที่

$$(Z_1^0, ..., Z_k^0) = (\langle\rangle, ..., \langle\rangle)$$

$$(Y_1^{i+1}, ..., Y_n^{i+1}, Z_1^{i+1}, ..., Z_k^{i+1}) = f(X_1, ..., X_m, Z_1^i, ..., Z_k^i)$$

ผลป้อนกลับของฟังก์ชันทางเดียว หรือ ฟังก์ชันต่อเนื่อง จะได้ผลลัพธ์เป็นฟังก์ชันทางเดียว หรือ ฟังก์ชันต่อเนื่อง ตามลำดับ



รูปที่ 2.4 ผลป้อนกลับของ k ช่อง [4]

2.1.1.4 **เครื่องเปลี่ยนสถานะ**

เครื่องเปลี่ยนสถานะเป็นแบบจำลองเป้าหมายสำหรับวิธีการพัฒนาบนพื้นฐานของกระแส เครื่องเปลี่ยนสถานะแบ่งเป็นสองชนิดคือ เครื่องเปลี่ยนสถานะแบบมิลลี (Mealy-based state transition machine) ซึ่งนำออกขณะที่เปลี่ยนสถานะ และเครื่องเปลี่ยนสถานะแบบมัวร์ (Moore-based state transition machine) ซึ่งนำออกที่สถานะ งานวิจัยนี้

เลือกใช้เครื่องเปลี่ยนสถานะแบบมัวร์ เพราะนำออกที่สถานะมีความเหมาะสมสำหรับการเปลี่ยนเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

นิยาม 2.14 โครงสร้างเครื่องเปลี่ยนสถานะแบบมัวร์ [1]

เครื่องเปลี่ยนสถานะแบบมัวร์โดยนำเข้าและนำออก $M = (Q, I, O, \delta, \varphi, q_0)$ หรือเรียกว่า เครื่องมัวร์ (Moore machine) ประกอบด้วย

- Q แทนเซตไม่ว่างของสถานะ
- I แทนเซตไม่ว่างของข้อมูลนำเข้า
- O แทนเซตไม่ว่างของข้อมูลนำออก
- $\delta : Q \times I \to Q$ แทนฟังก์ชันการเปลี่ยนสถานะขั้นเดี่ยว (Single-step state transition function)
- $\varphi : Q \to O*$ แทนฟังก์ชันนำออกสถานะเดี่ยว (Single-state output function)
- $q_0 \in Q$ แทนสถานะเริ่มต้น

นิยาม 2.15 ฟังก์ชันการเปลี่ยนสถานะหลายขั้น (Multi-step state transition function) [1]

ฟังก์ชันการเปลี่ยนสถานะหลายขั้น หมายถึงการบรรยายผลลัพธ์ของการเปลี่ยนสถานะทั้งหมดหลังจากการประมวลผลกระแสนำเข้าจำนวนจำกัด (Finite input stream)

ฟังก์ชันการเปลี่ยนสถานะหลายขั้น นิยามโดย

$$\delta* : Q \to I* \to Q$$

เมื่อ

$$\begin{aligned} \delta*(q)(\langle\rangle) &= q \\ \delta*(q)(x \lhd X) &= \delta*(\delta(q, x))(X) \end{aligned}$$

นิยาม 2.16 ฟังก์ชันนำออกหลายขั้น (Multi-state output function) [1]

ฟังก์ชันนำออกหลายขั้น หมายถึงการต่อส่วนนำออกจากทุกสถานะที่ผ่านเมื่อการประมวลผลกระแสนำเข้าจำกัด

ฟังก์ชันนำออกหลายขั้น นิยามโดย

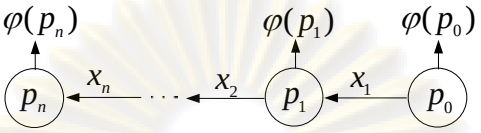
$$\varphi* : Q \to I* \to O*$$

เมื่อ

$$\begin{aligned} \varphi*(q)(\langle\rangle) &= \varphi(q) \\ \varphi*(q)(x \lhd X) &= \varphi(q) \,\&\, \varphi*(\delta(q, x))(X) \end{aligned}$$

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ในรูปที่ 2.5 แสดงการประมวลผลเครื่องมัวร์จากกระแสนำเข้า X เริ่มจากสถานะ q_0 ฟังก์ชันการเปลี่ยนสถานะหลายขั้น เปลี่ยนสถานะไปยังสถานะ q_n และฟังก์ชันนำออกหลายขั้น ผลิตกระแสนำออกโดยรวมทุกสถานะที่ผ่าน



$$\delta^*(p_0)(X) = p_n$$

$$\varphi^*(p_0)(X) = \varphi(p_0)\,\&\,\varphi(p_1)\,\&\ldots\&\,\varphi(p_n)$$

รูปที่ 2.5 เครื่องประมวลผลกระแสนำเข้า $X = \langle x_1,\ldots,x_n\rangle$ [1]

นิยาม 2.17 ภาคขยายนำออก (Output extension) [1]

ภาคขยายนำออก หมายถึงการแยกนำออกให้เหลือเฉพาะส่วนของนำออกที่เกิดจากนำเข้าปัจจุบัน

กำหนดให้ $f : \mathcal{A}^* \to \mathcal{B}^*$ เป็นฟังก์ชันการประมวลผลกระแส

ภาคขยายนำออก นิยามโดย

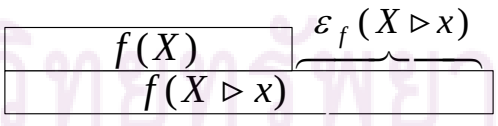
$$\varepsilon_f : \mathcal{A}^* \to \mathcal{B}^*$$

เมื่อ

$$\varepsilon_f(\langle\rangle) \quad = \quad f(\langle\rangle)$$

$$\varepsilon_f(x \lhd X) \quad = \quad f(X \rhd x) \ominus f(X)$$

ภาคขยายนำออกแยกส่วนนำออกที่ก่อกำเนิดโดยนำเข้าปัจจุบัน x หลังจากให้ประวัตินำเข้า X ดังแสดงในรูปที่ 2.6



รูปที่ 2.6 ภาคขยายนำออกของการประมวลผลกระแส f [1]

นิยาม 2.18 ฟังก์ชันภาวะนามธรรมประวัติ (History abstraction) [3]

ฟังก์ชันภาวะนามธรรมประวัติคือ ฟังก์ชันลดจำนวนสถานะของเครื่องเปลี่ยนสถานะโดยการจัดกลุ่มของสถานะที่มีนำออกและการเปลี่ยนสถานะเหมือนกัน

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

กำหนดให้ $f : \mathcal{A}^* \to \mathcal{B}^*$ เป็นฟังก์ชันการประมวลผลกระแส และ Q เป็นเซตของสถานะ

ฟังก์ชันภาวะนามธรรมประวัติ นิยามโดย

$$h : \mathcal{A}^* \to Q$$

เมื่อ นำออกเข้ากันได้ (Output compatible)

$$h(X) = h(Y) \Rightarrow \varepsilon_f(X) = \varepsilon_f(Y)$$

และการเปลี่ยนสถานะปิด (Transition closed)

$$h(X) = h(Y) \Rightarrow h(X \triangleright x) = h(Y \triangleright x)$$

นิยาม 2.19 เครื่องมัวร์แบบบัญญัติ [1]

ให้ฟังก์ชันการประมวลผลกระแส $f : \mathcal{A}^* \to \mathcal{B}^*$ เครื่องเปลี่ยนสถานะมัวร์แบบบัญญัติ (Canonical Moore state transition machine) หรือเรียกว่า เครื่องมัวร์แบบบัญญัติ (Canonical Moore machine) นิยามโดย

$$M_f = (\mathcal{A}^*, \mathcal{A}, \mathcal{B}^*, \delta, \varphi, q_0)$$

เมื่อ

$$q_0 = \langle\rangle \tag{2.20}$$

$$\delta(X, x) = X \triangleright x \tag{2.21}$$

$$\varphi_f(X) = \varepsilon(X) \tag{2.22}$$

$$\delta^*(X)(Y) = X \,\&\, Y \tag{2.23}$$

$$\varphi^*(q_0)(X) = f(X) \tag{2.24}$$

(2.20) หมายถึงกำหนดสถานะเริ่มต้นให้เป็นกระแสว่าง

(2.21) หมายถึงฟังก์ชันการเปลี่ยนสถานะขั้นเดี่ยวคือ การขยายประวัติด้วยหนึ่งสมาชิกนำเข้า

(2.22) หมายถึงฟังก์ชันนำออกสถานะเดี่ยวคือ ภาคขยายนำออกของฟังก์ชันการประมวลผลกระแส

(2.23) หมายถึงฟังก์ชันการเปลี่ยนสถานะหลายขั้นคือ การต่อกระแสนำเข้า

(2.24) หมายถึงฟังก์ชันนำออกหลายสถานะที่เริ่มต้นที่สถานะเริ่มต้นคือ ฟังก์ชันการประมวลผลกระแส

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

2.1.2 การตรวจสอบแบบจำลอง

วิธีการตรวจสอบแบบจำลองเชิงสัญลักษณ์ (Symbolic model checking) [6, 5] เป็นเทคนิคหนึ่งในกลุ่มวิธีที่ใช้พื้นฐานของแบบจำลอง โดยเทคนิคนี้เหมาะสมที่จะใช้ทวนสอบระบบสถานะจำกัด (Finite state system)

ในตอนนี้ทฤษฎีสามารถแบ่งเป็นสองตอนย่อยด้วยกัน ตอนที่ 2.1.2.1 ทฤษฎีที่เกี่ยวข้องกับโครงสร้างคริปเก ซึ่งเป็นแบบจำลองสำหรับการทวนสอบด้วยการตรวจสอบแบบจำลอง และตอนที่ 2.1.2.2 ทฤษฎีที่เกี่ยวข้องกับตรรกศาสตร์เชิงกาลเวลา ซึ่งใช้สำหรับกำหนดสมบัติเชิงเวลาสำหรับการทวนสอบด้วยการตรวจสอบแบบจำลอง

2.1.2.1 โครงสร้างคริปเก

การทวนสอบระบบจะทำโดยการแทนระบบที่จะทวนสอบให้อยู่ในรูปแบบที่เครื่องมือช่วยทวนสอบสามารถเข้าใจได้ โดยในที่นี้จะแทนระบบในลักษณะของโครงสร้างคริปเก ซึ่งเป็นระบบการเปลี่ยนสถานะชนิดหนึ่ง โดยโครงสร้างคริปเก ประกอบด้วยบัพ (Node) แทนสถานะของระบบ และเส้นเชื่อมแบบมีทิศทาง (Arc) แทนการเปลี่ยนระหว่างสถานะดังแสดงในรูปที่ 2.7 การตรวจสอบแบบจำลองจำกัดปริภูมิสถานะจำกัด (Finite state space) สำหรับการใช้เหตุผลเชิงตรรกะบนโครงสร้างของแบบจำลอง สถานะในโครงสร้างคริปเกเกี่ยวข้องกับประพจน์ของตรรกอันดับหนึ่ง (First-order logic) เรียกว่า ประพจน์เดี่ยว (Atomic proposition)

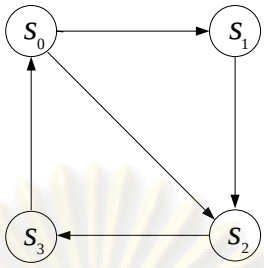
นิยาม 2.20 โครงสร้างคริปเก [5]

ให้ AP เป็นเซตของประพจน์เดี่ยว โครงสร้างคริปเก M บนเซต AP นิยามโดย $M = (S, S_0, R, L)$ ประกอบด้วย

- S แทนเซตของสถานะ
- $S_0 \subseteq S$ แทนเซตของสถานะเริ่มต้น
- $R \subseteq S \times S$ แทนความสัมพันธ์การเปลี่ยนสถานะ (State transition relation) ซึ่งต้องเป็นความสัมพันธ์แบบทั่วถึงในเซต S หรืออีกความหมายคือ สำหรับทุกสถานะ $s \in S$ จะต้องมี $s' \in S$ ซึ่งทำให้ความสัมพันธ์ $R(s, s')$ เป็นจริง
- $L : S \to 2^{AP}$ คือ ฟังก์ชันซึ่งกำหนดป้ายแต่ละสถานะกับเซตของประพจน์เดี่ยวที่เป็นจริงในสถานะนั้น

บางครั้งจะไม่พิจารณาเซตของสถานะเริ่มต้น S_0 ในกรณีนี้จะยกเว้นเซตของสถานะเริ่มต้น

วิถี (Path) ในโครงสร้าง M จากสถานะ s คือ ลำดับอนันต์ของสถานะ $\pi = s_0 s_1 s_2 \ldots$ เมื่อ $s_0 = s$ และ $R(s_i, s_{i+1})$ เป็นจริงสำหรับทุก $i \geq 0$



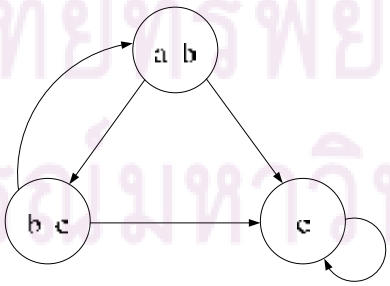
รูปที่ 2.7 ตัวอย่างของโครงสร้างคริปเก

2.1.2.2 **ตรรกศาสตร์เชิงกาลเวลา** [5]

ตรรกศาสตร์เชิงกาลเวลาคือ ข้อกำหนดซึ่งเป็นการกำหนดสมบัติเชิงเวลา สามารถแบ่งเป็นสามชนิดคือ ตรรกเชิงกาลเวลาแบบเชิงเส้นหรือตรรกแอลทีแอล (Linear-time temporal logic or LTL) ตรรกเชิงกาลเวลาแบบต้นไม้การคำนวณหรือตรรกซีทีแอล (Computational tree logic or CTL) และตรรกซีทีแอลสตาร์ (CTL*) โดยตรรกซีทีแอล และตรรกแอลทีแอลเป็นตรรกย่อยของตรรกซีทีแอลสตาร์ ข้อแตกต่างระหว่างตรรกซีทีแอลและตรรกแอลทีแอลคือ วิธีการจัดการกิ่งของต้นไม้การคำนวณ ในตรรกซีทีแอล ตัวดำเนินการเชิงเวลากำหนดปริมาณบนวิถีที่เป็นไปได้จากสถานะที่กำหนดให้ ในขณะที่ในตรรกแอลทีแอล ตัวดำเนินการถูกกำหนดสำหรับการบรรยายเหตุการณ์วิถีการคำนวณเดี่ยว

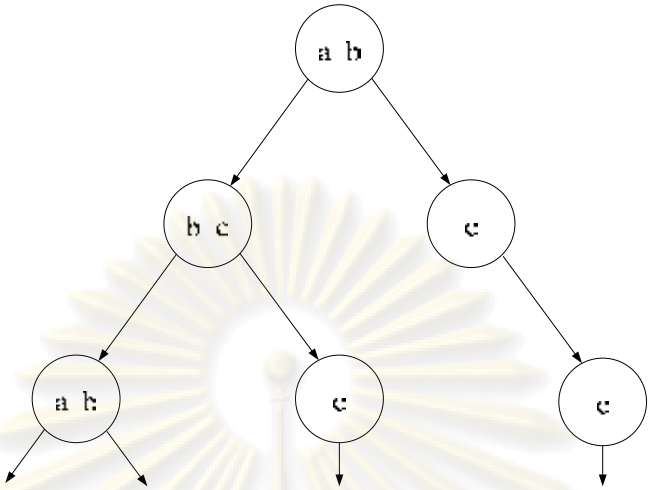
2.1.2.2.1 **ตรรกซีทีแอลสตาร์** [5]

สูตรตรรกซีทีแอลสตาร์ บรรยายสมบัติของต้นไม้การคำนวณ (Computational tree) ซึ่งสร้างได้จากโครงสร้างคริปเก โดยที่รากของต้นไม้คือ สถานะเริ่มต้นของโครงสร้างคริปเก และบัพลูก (Child node) คือ สถานะที่มีเส้นเชื่อมจากสถานะปัจจุบัน ตัวอย่างในรูปที่ 2.8 (b) เป็นต้นไม้การคำนวณที่สร้างจากโครงสร้างคริปเกในรูปที่ 2.8 (a) จากรูปจะเห็นได้ว่าต้นไม้การคำนวณแสดงทุกวิถีทางที่เป็นไปได้จากสถานะเริ่มต้นที่เป็นไปได้ของโครงสร้างคริปเก



รูปที่ 2.8 (b) โครงสร้างคริปเก [5]

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 2.8 (a) ต้นไม้การคำนวณที่ได้จากโครงสร้างคริปเก [5]

ตรรกซีทีแอลสตาร์ประกอบด้วยตัวบ่งปริมาณวิถี (Path quantifier) และตัวดำเนินการเชิงเวลา (Temporal operator) ตัวบ่งปริมาณวิถีใช้บรรยายกิ่งโครงสร้างในต้นไม้การคำนวณ ตัวบ่งปริมาณวิถีประกอบด้วยตัวบ่งปริมาณวิถีแบบทุกกรณี A (“สำหรับทุกวิถีการคำนวณ”) และตัวบ่งปริมาณวิถี E (“สำหรับบางวิถีการคำนวณ”) ตัวดำเนินการเชิงเวลาประกอบด้วยตัวดำเนินการพื้นฐานห้าตัวคือ

- $\mathrm{X}\,f$ (“เป็นจริงในสถานะถัดไป”) จะเป็นจริงเมื่อสมบัติ f เป็นจริงในสถานะถัดไปบนวิถีที่พิจารณาอยู่
- $\mathrm{F}\,f$ (“เป็นจริงบางเวลา”) จะเป็นจริงเมื่อสมบัติ f เป็นจริงในบางสถานะบนวิถีที่พิจารณาอยู่
- $\mathrm{G}\,f$ (“เป็นจริงตลอดเวลา”) จะเป็นจริงเมื่อสมบัติ f เป็นจริงทุกสถานะบนวิถีที่พิจารณาอยู่
- $f\,\mathrm{U}\,g$ (“เป็นจริงจนกระทั่ง”) จะเป็นจริงถ้ามีบางสถานะบนวิถีที่พิจารณาอยู่ซึ่งสมบัติ g เป็นจริง และทุกสถานะก่อนหน้านั้นสมบัติ f เป็นจริงตลอด
- $f\,\mathrm{R}\,g$ (“เป็นจริงเมื่อปล่อย”) จะเป็นจริงถ้าทุกสถานะบนวิถีที่พิจารณาอยู่ซึ่งสมบัติ g เป็นจริง รวมถึงสถานะแรกที่ f เป็นจริง อย่างไรก็ตาม f อาจเป็นจริงหลายสถานะในวิถีที่พิจารณาอยู่ได้

ไวยากรณ์ (Syntax) และความหมาย (Semantic) ของตรรกซีทีแอลสตาร์ สูตรในซีทีแอลสตาร์มีสองชนิดด้วยกันคือ สูตรสถานะ (State formula) ซึ่งเป็นจริงในสถานะ และสูตรวิถี (Path formula) ซึ่งเป็นจริงตลอดวิถี ให้ AP เป็นเซตของประพจน์เดี่ยว ไวยากรณ์ของสูตรสถานะมีดังนี้

- ถ้า $p \in AP$ แล้ว p เป็นสูตรสถานะ

- ถ้า f และ g คือ สูตรสถานะ แล้ว $\neg f, f \vee g$ และ $f \wedge g$ เป็นสูตรสถานะ
- ถ้า f เป็นสูตรวิถี แล้ว $\mathrm{E}\, f$ และ $\mathrm{A}\, f$ เป็นสูตรสถานะ

ไวยากรณ์ของสูตรวิถีมีดังนี้

- ถ้า f เป็นสูตรสถานะ แล้ว f เป็นสูตรวิถีด้วย
- ถ้า f และ g เป็นสูตรวิถี แล้ว $\neg f, f \vee g, f \wedge g, \mathrm{X}\, f, \mathrm{F}\, f, \mathrm{G}\, f, f\, \mathrm{U}\, g$ และ $f\, \mathrm{R}\, g$ เป็นสูตรวิถี

ตรรกซีทีแอลสตาร์คือ เซตของสูตรสถานะซึ่งสร้างจากไวยากรณ์ที่กล่าวมาแล้ว ต่อไปจะกล่าวถึงความหมายของตรรกซีทีแอลสตาร์ ถ้า f เป็นสูตรสถานะ สัญลักษณ์ $M, s \models f$ มีความหมายว่า f เป็นจริงที่สถานะ s ในโครงสร้างคริปเก $M = \langle S, R, L \rangle$ ทำนองเดียวกัน ถ้า f เป็นสูตรวิถี $M, \pi \models f$ มีความหมายว่า f เป็นจริงตลอดวิถี π ในโครงสร้างคริปเก M ในกรณีนี้โครงสร้าง M เป็นที่รู้จักกันเราสามารถละไม่เขียน M ก็ได้

กำหนดให้ f_1 และ f_2 เป็นสูตรสถานะ g_1 และ g_2 เป็นสูตรวิถี และนิยามว่า π^i หมายถึง วิถีที่เริ่มจากสถานะ s_i ของวิถี π เราสามารถนิยามความสัมพันธ์ $\models$ จากกฎดังนี้

- $M, s \models p \quad \Leftrightarrow \quad p \in L(s)$
- $M, s \models \neg f_1 \quad \Leftrightarrow \quad M, s \not\models \neg f_1$
- $M, s \models f_1 \vee f_2 \quad \Leftrightarrow \quad M, s \models f_1$ หรือ $M, s \models f_2$
- $M, s \models f_1 \wedge f_2 \quad \Leftrightarrow \quad M, s \models f_1$ และ $M, s \models f_2$
- $M, s \models \mathrm{E}\, g_1 \quad \Leftrightarrow \quad$ มีบางวิถี π ที่เริ่มต้นจากสถานะ s ซึ่ง $M, \pi \models g_1$
- $M, s \models \mathrm{A}\, g_1 \quad \Leftrightarrow \quad$ สำหรับทุกวิถี π ที่เริ่มต้นจากสถานะ s, $M, \pi \models g_1$
- $M, \pi \models f_1 \quad \Leftrightarrow \quad$ ให้สถานะ s คือ สถานะแรกของวิถี π และ $M, s \models f_1$
- $M, \pi \models \neg g_1 \quad \Leftrightarrow \quad M, \pi \not\models g_1$
- $M, \pi \models g_1 \vee g_2 \quad \Leftrightarrow \quad M, \pi \models g_1$ หรือ $M, \pi \models g_2$
- $M, \pi \models g_1 \wedge g_2 \quad \Leftrightarrow \quad M, \pi \models g_1$ และ $M, \pi \models g_2$
- $M, \pi \models \mathrm{X}\, g_1 \quad \Leftrightarrow \quad M, \pi^1 \models g_1$
- $M, \pi \models \mathrm{F}\, g_1 \quad \Leftrightarrow \quad$ มีบางค่า $k \geq 0$ ซึ่ง $M, \pi^k \models g_1$
- $M, \pi \models \mathrm{G}\, g_1 \quad \Leftrightarrow \quad$ สำหรับทุกค่า $i \geq 0, M, \pi^i \models g_1$

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

- $M,\pi \models g_1 \mathrm{U} g_2 \quad \Leftrightarrow$ มีบางค่า $k \geq 0$ ซึ่ง $M,\pi^k \models g_2$ และ
  สำหรับทุกค่า $0 \leq j < k, M,\pi^j \models g_1$
- $M,\pi \models g_1 \mathrm{R} g_2 \quad \Leftrightarrow$ สำหรับทุกค่า $j \geq 0$, ถ้าสำหรับทุก $i < j$ $M,\pi^i \not\models g_1$ แล้ว
  $M,\pi^j \models g_2$

2.1.2.2.2 **ตรรกซีทีแอล** [5]

ตรรกซีทีแอลเป็นเซตย่อยของตรรกซีทีแอลสตาร์ ซึ่งมีเงื่อนไขว่าแต่ละตัวดำเนินการเชิงเวลา $\mathrm{X},\mathrm{F},\mathrm{G},\mathrm{U}$ และ R จะต้องใช้คู่กับตัวบ่งปริมาณวิถีเสมอ นอกจากนั้นตรรกซีทีแอลสตาร์เพิ่มไวยากรณ์ด้วยดังนี้

- ถ้า f และ g เป็นสูตรสถานะ แล้ว $\neg f, f \vee g, f \wedge g, \mathrm{X} f, \mathrm{F} f, \mathrm{G} f, f \mathrm{U} g$ และ $f \mathrm{R} g$ เป็นสูตรวิถี

2.1.2.2.3 **ตรรกแอลทีแอล** [5]

ตรรกแอลทีแอลประกอบด้วยสูตรซึ่งมีรูปแบบ $\mathrm{A} f$ โดยที่ f เป็นสูตรวิถี ซึ่งสูตรย่อยสถานะอนุญาตให้เป็นประพจน์เดี่ยวเท่านั้น สูตรวิถีตรรกแอลทีแอลมีดังนี้

- ถ้า $p \in AP$ แล้ว p เป็นสูตรวิถี
- ถ้า f และ g เป็นสูตรวิถี แล้ว $\neg f, f \vee g, f \wedge g, \mathrm{X} f, \mathrm{F} f, \mathrm{G} f, f \mathrm{U} g$ และ $f \mathrm{R} g$ เป็นสูตรวิถี

2.1.3 **ระบบเปลี่ยนสถานะนำเข้า/นำออก**

ระบบเปลี่ยนสถานะนำเข้า/นำออกเป็นรูปแบบที่มีความเหมาะสมสำหรับการทวนสอบด้วยการตรวจสอบแบบจำลอง

นิยาม 2.21 ระบบเปลี่ยนสถานะนำเข้า/นำออก [7]

ระบบเปลี่ยนสถานะนำเข้า/นำออก $T = (I, O, Q, S, R, s_0)$ ประกอบด้วย

- I แทนชุดตัวอักษรนำเข้า (input alphabet)
- O แทนชุดตัวอักษรนำออก (output alphabet)
- Q แทนเซตของสถานะภายใน
- S แทนเซตของสถานะ โดยที่ $S \subseteq (I \cup \{\phi\}) \times Q \times O^*$
- R แทนความสัมพันธ์การเปลี่ยนสถานะ โดยที่ $R \subseteq S \times S$
- s_0 แทนสถานะเริ่มต้น

เมื่อ $\phi \notin I$ แทนนำเข้าว่าง (Empty input) ในสถานะเริ่มต้น

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

นิยาม 2.22 ฟังก์ชันการแปลง (Transforming function) [7]

ฟังก์ชันการแปลงคือ ฟังก์ชันที่เปลี่ยนสถานะของเครื่องมัวร์แบบบัญญัติไปเป็นสถานะของระบบการเปลี่ยนสถานะ

ฟังก์ชันการแปลง นิยามโดย

$$\alpha : \mathcal{A}^* \to S$$

เมื่อ

$$\alpha(\langle\rangle) = (\phi, h(\langle\rangle), \varphi_f(\langle\rangle))$$
$$\alpha(X \triangleright x) = (x, h(X \triangleright x), \varphi_f(X \triangleright x))$$

2.2 งานวิจัยที่เกี่ยวข้อง

ในตอนนี้งานวิจัยสามารถแบ่งเป็นสามตอนหลักด้วยกัน ตอนที่ 2.2.1 งานวิจัยที่เกี่ยวข้องกับกระแสและการพัฒนาบนพื้นฐานของกระแส ตอนที่ 2.2.2 งานวิจัยที่เกี่ยวข้องกับการตรวจสอบแบบจำลอง และตอนที่ 2.2.3 งานวิจัยที่เกี่ยวข้องกับการทวนสอบสำหรับสมบัตินำเข้า/นำออกของแบบจำลองกล่องดำ

2.2.1 งานวิจัยที่เกี่ยวข้องกับกระแสและการพัฒนาบนพื้นฐานของกระแส

Dosch, Ruanthong และ Stumpel [1] นำเสนอวิธีการกำหนดรายละเอียดฟังก์ชันการประมวลผลกระแสที่มีหนึ่งช่องทางนำเข้าและหนึ่งช่องทางนำออกเป็นเครื่องเปลี่ยนสถานะแบบมัวร์เพื่อใช้ในการทำให้เกิดผล โดยนำเสนอวิธีการเชิงรูปนัยสำหรับการแปลงฟังก์ชันการประมวลผลกระแสที่มีหนึ่งช่องทางนำเข้าและหนึ่งช่องทางนำออกเป็นเครื่องเปลี่ยนสถานะแบบมัวร์ ซึ่งฟังก์ชันการประมวลผลกระแสจะไม่แสดงสถานะภายใน เมื่อแปลงเป็นเครื่องเปลี่ยนสถานะมัวร์แล้วจะแสดงให้เห็นสถานะภายในโดยแทนสถานะด้วยกระแสนำเข้า การแปลงแบ่งเป็นสองขั้นตอนย่อยคือ การแปลงจากฟังก์ชันการประมวลผลกระแสไปเป็นเครื่องมัวร์แบบบัญญัติ ซึ่งแทนสถานะด้วยกระแสนำเข้า และนำออกสามารถพิจารณาที่สถานะ นั่นคือนำออกสัมพันธ์กับสถานะไม่ใช่การเปลี่ยนสถานะ ทำให้สามารถพิจารณาส่วนประกอบที่ปล่อยนำออกโดยที่ไม่ได้รับนำเข้า และฟังก์ชันการประมวลผลกระแสที่มีหนึ่งช่องทางนำเข้าและหนึ่งช่องทางนำออกทุกฟังก์ชันสามารถแปลงเป็นเครื่องมัวร์แบบบัญญัติได้ทั้งหมด จากนั้นทำการแปลงเครื่องมัวร์แบบบัญญัติไปเป็นเครื่องมัวร์แบบลดรูป โดยอาศัยฟังก์ชันภาวะนามธรรมประวัติในการลดจำนวนสถานะ ซึ่งขั้นตอนนี้ขึ้นอยู่กับฟังก์ชันภาวะนามธรรมประวัติที่ตัดสินใจเลือก เครื่องเปลี่ยนสถานะแบบมัวร์ที่ได้รับมีความเหมาะสมในการทำให้เกิดผล แต่เครื่องเปลี่ยนสถานะจากงานวิจัยนี้ไม่สามารถแทนฟังก์ชันการประมวลผลกระแสที่มีหลายช่องทางนำเข้าและหลายช่องทางนำออกได้

Dosch และ Stumpel [3] นำเสนอวิธีการกำหนดรายละเอียดฟังก์ชันการประมวลผลกระแสที่มีหนึ่งช่องทางนำเข้าและหนึ่งช่องทางนำออกเป็นเครื่องเปลี่ยนสถานะแบบมิลลีเพื่อใช้ในการทำให้เกิดผล โดยนำเสนอวิธีการเชิงรูปนัยสำหรับการแปลงฟังก์ชันการประมวลผลกระแสที่มีหนึ่งช่องทางนำเข้าและหนึ่งช่องทางนำออกซึ่งเป็นแบบจำลองบนพื้นฐานของกระแสเป็นเครื่องเปลี่ยนสถานะแบบมิลลีซึ่งเป็นแบบจำลองบนพื้นฐานของสถานะที่มีความเหมาะสมในการทำให้เกิดผล โดยที่วิธีการนี้อาศัยฟังก์ชันภาวะนามธรรมประวัติช่วยในการลดจำนวนสถานะในระบบ และฟังก์ชันการประมวลผลกระแสหนึ่งฟังก์ชัน สามารถแปลงเป็นเครื่องเปลี่ยนสถานะแบบมิลลีที่แตกต่างกันได้ ถ้าใช้ฟังก์ชันภาวะนามธรรมประวัติที่แตกต่างกัน นำออกของเครื่องเปลี่ยนสถานะแบบมิลลีสามารถพิจารณาขณะที่เปลี่ยนสถานะ แต่เครื่องเปลี่ยนสถานะจากงานวิจัยนี้ไม่สามารถแสดงนำออกที่เกิดที่สถานะเริ่มต้นซึ่งไม่ได้รับนำเข้า และไม่สามารถแทนฟังก์ชันการประมวลผลกระแสที่มีหลายช่องทางนำเข้าและหลายช่องทางนำออกได้

Dosch และ Stumpel [2] เสนอวิธีการเชิงรูปนัยสำหรับการแปลงฟังก์ชันการประมวลผลที่มีหลายช่องทางนำเข้าและช่องทางนำออก ไปเป็นเครื่องเปลี่ยนสถานะที่มีความเหมาะสมในการทำให้เกิดผล โดยที่เครื่องเปลี่ยนสถานะที่ได้รับยังคงสภาพสมบัตินำเข้า/นำออกเดิมของฟังก์ชันการประมวลผลกระแส และใช้สมบัติเพชร (Diamond property) ในการรับรองว่าเครื่องเปลี่ยนสถานะไม่ได้รับผลกระทบจากลำดับของทูเพิลที่รับนำเข้า การแปลงแบ่งเป็นสองขั้นตอนคือ การแปลงฟังก์ชันการประมวลผลกระแสเป็นเครื่องเปลี่ยนสถานะแบบบัญญัติ โดยที่ฟังก์ชันการประมวลผลกระแสทั้งหมดสามารถแปลงเป็นเครื่องเปลี่ยนสถานะแบบบัญญัติได้ ขั้นตอนที่สองคือ การแปลงเครื่องเปลี่ยนสถานะแบบบัญญัติเป็นเครื่องเปลี่ยนสถานะ ซึ่งยังคงสภาพของสมบัตินำเข้า/นำออกเดิมของเครื่องเปลี่ยนสถานะแบบบัญญัติ โดยอาศัยฟังก์ชันนามธรรมประวัติในการลดจำนวนสถานะในระบบ และฟังก์ชันการหาอนุพันธ์ (Differentiation) ในการแสดงผลที่เกิดจากนำเข้าเดี่ยวจากหนึ่งช่องนำเข้า แต่เครื่องเปลี่ยนสถานะจากงานวิจัยนี้ไม่สามารถแสดงนำออกที่เกิดที่สถานะเริ่มต้นที่ไม่ได้รับนำเข้า

Broy, Dederichs และ Dendorfer [9] นำเสนอวิธีการพัฒนาซอฟต์แวร์ที่ชื่อว่าโฟกัส (Focus) ซึ่งเป็นวิธีการเชิงรูปนัยสำหรับการพัฒนาระบบแบบกระจาย เริ่มจากการกำหนดข้อกำหนดความต้องการ (Requirements specification) ซึ่งใช้เทคนิคข้อกำหนดติดตาม (Trace specification) แล้วกำหนดข้อกำหนดการออกแบบ (Design specification) โดยใช้เทคนิคข้อกำหนดเชิงฟังก์ชัน (Functional specification) หลังจากนั้นทำการแปลงจากข้อกำหนดการออกแบบ เป็นการทำให้เกิดผลนามธรรม (Abstract implementation) ซึ่งสามารถใช้เทคนิคการโปรแกรมเชิงฟังก์ชัน (Functional programming) คือ ภาษาเชิงหน้าที่ (Applicative language)

และทำการแปลงการทำให้เกิดผลนามธรรม เป็นการทำให้เกิดผลรูปธรรม (Concrete implementation) ซึ่งสามารถใช้เทคนิคการโปรแกรมเชิงกระบวนคำสั่ง (Procedural programming) คือ ภาษาเชิงกระบวนคำสั่ง (Procedural language) โดยที่การแปลงจากข้อกำหนดระดับสูงไปยังข้อกำหนดระดับต่ำไม่สามารถทำได้โดยอัตโนมัติทั้งหมด มีบางขั้นตอนที่ต้องอาศัยการตัดสินใจของผู้พัฒนาระบบ

Stumpel [4] นำเสนอฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมิลลีของฟังก์ชันการประมวลผลกระแส โดยแบ่งฟังก์ชันการประมวลผลกระแสเป็นฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบ 6 กลุ่มดังต่อไปนี้

1) กลุ่มส่วนประกอบที่อิสระจากประวัติ (History independent components) คือ กลุ่มส่วนประกอบที่นำออกขึ้นอยู่กับสมาชิกตัวปัจจุบันของกระแสนำเข้าเพียงตัวเดียวเท่านั้นไม่ขึ้นกับสมาชิกก่อนหน้าของกระแสนำเข้า ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมิลลีของฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบนี้คือ ฟังก์ชัน $const$ ที่ส่งกระแสไปเป็นกระแสเดี่ยวที่กำหนด นิยามดังนี้

$$const \quad : \mathcal{A}^* \to \{q_0\}$$

โดยที่

$$const(X) \quad = \{q_0\} \tag{4.20}$$

(4.20) หมายถึง ส่งกระแสนำเข้าไปเป็นสถานะเดี่ยวที่กำหนด

2) กลุ่มส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต (Components with a bounded short-term memory) คือ กลุ่มส่วนประกอบที่ภาคขยายนำออกขึ้นอยู่กับเติมหลังของกระแสนำเข้าที่มีความยาวคงตัวหรือขึ้นอยู่กับกระแสนำเข้าเท่าที่มีอยู่ในกรณีที่กระแสนำเข้ามีความยาวน้อยกว่าค่าคงตัวที่กำหนด ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมิลลีของฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบนี้คือ ฟังก์ชัน $last(n)$ ที่เก็บกระแสนำเข้าเฉพาะสมาชิก n ตัวหลังสุดในกรณีที่กระแสนำเข้ามีความยาวมากกว่า n หรือเก็บกระแสนำเข้าทั้งหมดในกรณีที่กระแสนำเข้ามีความยาวน้อยกว่าหรือเท่ากับ n นิยามดังนี้

$$last : \mathbb{N} \to [\mathcal{A}^* \to \mathcal{A}^*]$$

โดยที่

$$last(n')(\langle\rangle) = \langle\rangle \tag{2.25}$$

$$last(0)(X) = \langle\rangle \tag{2.26}$$

$$last(n')(X \triangleright x) = last(n'-1)(X) \triangleright x \tag{2.27}$$

สำหรับทุก $n' \in \mathbb{N}$

(2.25) หมายถึง กระแสนำเข้าว่างก่อกำเนิดกระแสนำออกว่าง

(2.26) หมายถึง ค่าของพารามิเตอร์เป็นศูนย์ก่อกำเนิดกระแสนำออกว่าง

(2.27) หมายถึง ฟังก์ชัน $last(n')$ ก่อกำเนิดกระแสนำออกเป็นกระแสนำเข้าย่อยที่ประกอบด้วยสมาชิก n' ตัวสุดท้าย

3) กลุ่มส่วนประกอบแบบมีคำร้องขอ (Components with requests) คือ กลุ่มส่วนประกอบที่มีชนิดข้อมูลนำเข้าสามารถจำแนกได้เป็นเซตย่อยของข้อมูลที่มีผลต่อนำออกที่จะเกิดขึ้นในอนาคต และเซตย่อยของข้อมูลที่ปราศจากผลกระทบต่อนำออกที่จะเกิดขึ้นในอนาคต ชนิดข้อมูลที่มีผลต่อนำออกที่จะเกิดขึ้นในอนาคตเรียกว่า อัพเดท (Update) และชนิดข้อมูลที่มีไม่ผลต่อนำออกที่จะเกิดขึ้นในอนาคตเรียกว่า คำร้องขอ (Request) ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมิลลีของฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบนี้คือ ฟังก์ชัน $filterU$ ที่จำแนกเซตของชนิดข้อมูลนำเข้าเป็นเซตย่อยของข้อมูลอัพเดท และเซตย่อยของข้อมูลคำร้องขอ นิยามดังนี้

$$filterU \ : \mathcal{A}^* \to \mathcal{A}^*$$

โดยที่

$$filterU(\langle\rangle) = \langle\rangle \tag{2.28}$$

$$filterU(x \lhd X) = \begin{cases} \langle x \rangle \lhd filterU(x \lhd X) & if \ x \in U \\ filterU(x \lhd X) & if \ x \notin U \end{cases} \tag{2.29}$$

(2.28) หมายถึง กระแสนำเข้าว่างก่อกำเนิดกระแสนำออกว่าง

(2.29) หมายถึง เก็บเฉพาะสมาชิกของกระแสนำเข้าที่เป็นข้อมูลอัพเดทเท่านั้น

4) กลุ่มส่วนประกอบคงที่ (Stationing components) คือ กลุ่มส่วนประกอบที่มีพฤติกรรมคงที่เมื่อได้รับข้อมูลนำเข้าที่ไม่คาดหมาย เป็นสาเหตุของการหยุดพฤติกรรมของส่วนประกอบยังคงเป็นเช่นนี้บนการขยายของกระแสนำเข้านี้ทั้งหมด ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมิลลีของฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบนี้คือ ฟังก์ชัน $break$ ที่จำแนกกระแสนำเข้าเป็นกระแสนำเข้าปกติ และกระแสนำเข้าคงที่ นิยามดังนี้

$$break : \mathcal{A}^* \to R \cup \{s_i \mid i \in I\}$$

โดยที่

$$break(X) = \begin{cases} s_i \ if \ X \in S_i \\ X \ if \ X \in R \end{cases} \tag{2.30}$$

เมื่อ $s_i \neq s_j$ สำหรับ $i \neq j$ และ R คือ กระแสปกติ

(2.30) หมายถึง จำแนกกระแสนำเข้าเป็นกระแสนำเข้าปกติ และกระแสนำเข้าคงที่

5) กลุ่มส่วนประกอบสแกน (Scan components) คือ กลุ่มส่วนประกอบที่สะสมนำเข้าของส่วนประกอบด้วยการดำเนินการทวิภาค (Binary operation) เพื่อใช้ในการก่อกำเนิดนำออก ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมิลลีของฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบนี้คือ ฟังก์ชัน $reduce(s,\oplus)$ ที่ดำเนินการสมาชิกของกระแสนำเข้าด้วยตัวดำเนินการทวิภาคแล้วเก็บสะสมไว้ที่ตัวแปรภายใน นิยามดังนี้

$$reduce : \mathcal{B} \times [\mathcal{B} \times \mathcal{A} \to \mathcal{B}] \to [\mathcal{A}^* \to \mathcal{B}]$$

โดยที่

$$reduce(s,\oplus)(\langle\rangle) = s \tag{2.31}$$

$$reduce(s,\oplus)(x \lhd X) = reduce(s \oplus x,\oplus)(X) \tag{2.32}$$

เมื่อ $x \in \mathcal{A}, s \in \mathcal{B}$ และ $X \in \mathcal{A}^*$

(2.31) หมายถึง นำเข้าว่างก่อกำเนิดนำออกที่เป็นค่าที่เก็บไว้ในสถานะภายใน s

(2.32) หมายถึง สมาชิกแต่ละตัวของกระแสนำเข้าถูกดำเนินการด้วยตัวดำเนินการทวิภาคแล้วเก็บสะสมไว้ในตัวแปรภายใน s

6) กลุ่มส่วนประกอบที่ต้องการประวัติ (History requiring components) คือ กลุ่มส่วนประกอบที่ไม่มีประวัตินำเข้าที่ทำให้นำออกเหมือนกัน ดังนั้นจึงไม่สามารถลดจำนวนสถานะในระบบได้

ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมิลลีเป็นฟังก์ชันที่ลดจำนานของสถานะที่เกิดจากการแทนสถานะด้วยกระแสนำเข้ายกเว้นสมาชิกตัวปัจจุบันเนื่องจากนำออกของเครื่องมิลลีเกิดที่การเปลี่ยนสถานะ แตกต่างจากสถานะของเครื่องมัวร์ที่แทนด้วยกระแสนำเข้าทั้งหมดเนื่องจากนำออกของเครื่องมัวร์เกิดที่สถานะ

ลักษณะของกลุ่มส่วนประกอบนำไปใช้ในการนำเสนอฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ที่มีความเหมาะสมในการทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองบนพื้นฐานของกระแสด้วยการตรวจสอบแบบจำลองในบทที่ 4

2.2.2 งานวิจัยที่เกี่ยวข้องกับการตรวจสอบแบบจำลอง

Cimatti และคณะ [11] นำเสนอตัวตรวจสอบแบบจำลองชื่อเอ็นยูเอสเอ็มวี ซึ่งรวมเทคนิคการตรวจสอบแบบจำลองแบบบีดีดี (BDD-based model checking) และเทคนิคการตรวจสอบแบบจำลองแบบแซท (SAT-based model checking) ซึ่งเทคนิคทั้งสองแบบเป็นส่วนเติมเต็มซึ่งกันและกัน โดยการตรวจสอบแบบจำลองเป็นกระบวนการอัตโนมัติ สมบัติเชิงเวลาที่กำหนดไว้จะถูกตรวจสอบว่าสอดคล้องกับแบบจำลองหรือไม่ ถ้าไม่สอดคล้องกับสมบัติเชิงเวลาที่กำหนด โปรแกรมจะสร้างตัวอย่างกรณีที่เกิดปัญหาให้ด้วยซึ่งมีประโยชน์ในการแก้ไขแบบจำลอง

เอ็นยูเอสเอ็มวีเป็นซอฟต์แวร์แบบโอเพนซอร์ส (Open source) จึงทำให้ทุกคนที่สนใจสามารถใช้และพัฒนาร่วมกันอย่างอิสระ

2.2.3 งานวิจัยที่เกี่ยวข้องกับการทวนสอบสำหรับสมบัตินำเข้า/นำออกของแบบจำลองกล่องดำ

Dosch, และคณะ [7] นำเสนอวิธีการทวนสอบสำหรับฟังก์ชันการประมวลผลกระแสด้วยวิธีการตรวจสอบแบบจำลอง โดยการแปลงฟังก์ชันการประมวลผลกระแสไปเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก ซึ่งมีความเหมาะสมสำหรับการทวนสอบด้วยวิธีการตรวจสอบแบบจำลอง การแปลงแบ่งเป็นสองขั้นตอนคือ การแปลงจากฟังก์ชันการประมวลผลกระแสไปเป็นเครื่องมัวร์แบบบัญญัติ และแปลงจากเครื่องมัวร์แบบบัญญัติไปเป็นระบบเปลี่ยนสถานะนำเข้า/นำออกที่สามารถคงสภาพสมบัตินำเข้า/นำออกของฟังก์ชันการประมวลผลกระแสเดิม ซึ่งงานวิจัยนี้ที่ทำให้เกิดแนวความคิดริเริ่มของงานวิจัยที่จะนำเสนอ

ต่อไปนี้เป็นวิธีการแปลงฟังก์ชันการประมวลผลกระแสที่มีหนึ่งช่องทางนำเข้า และหนึ่งช่องทางนำออก เป็นระบบเปลี่ยนสถานะนำเข้า/นำออก ที่ใช้เป็นแบบจำลองสำหรับการทวนสอบด้วยการตรวจสอบแบบจำลอง 2.2.3.1 แสดงการแปลงจากฟังก์ชันการประมวลผลกระแสเป็นเครื่องมัวร์แบบบัญญัติ และ 2.2.3.2 แสดงการแปลงจากเครื่องมัวร์แบบบัญญัติเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

2.2.3.1 การแปลงจากฟังก์ชันการประมวลผลกระแสเป็นเครื่องมัวร์แบบบัญญัติ

สำหรับฟังก์ชันการประมวลผลกระแส $f:\mathcal{A}^*\to\mathcal{B}^*$ สามารถแปลงเป็นเครื่องมัวร์แบบบัญญัติ $M_f=(\mathcal{A}^*,\mathcal{A},\mathcal{B},\delta,\varphi_f,\langle\rangle)$ โดยที่ฟังก์ชันการเปลี่ยนสถานะขั้นเดี่ยว $\delta:\mathcal{A}^*\times\mathcal{A}\to\mathcal{A}^*$ ขยายสถานะโดยนำเข้าปัจจุบัน

$$\delta(X,x)=X\triangleright x$$

ฟังก์ชันนำออกสถานะเดี่ยว $\varphi_f:\mathcal{A}^*\to\mathcal{B}^*$ สอดคล้องตามฟังก์ชันภาคขยายนำออก

$$\varphi_f(X)=\varepsilon_f(X)$$

2.2.3.2 การแปลงจากเครื่องมัวร์แบบบัญญัติเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

สำหรับเครื่องมัวร์แบบบัญญัติ $M_f=(\mathcal{A}^*,\mathcal{A},\mathcal{B},\delta,\varphi_f,\langle\rangle)$ สามารถแปลงเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก $T_{f,\alpha}=(\mathcal{A},\mathcal{B}^*,Q,S,R,s_0)$ โดยที่

$$S=(\mathcal{A}\cup\{\phi\})\times Q\times\mathcal{B}^*,$$

$$R=\{(\alpha(X),\alpha(X\triangleright x))\mid X\in\mathcal{A}^*,x\in\mathcal{A}\},$$

$$s_0=(\phi,h(\langle\rangle),\varphi_f(\langle\rangle))$$

เมื่อฟังก์ชันการแปลง $\alpha : \mathcal{A}^* \to S$ ส่งสถานะของเครื่องมัวร์แบบบัญญัติไปยังสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออก นิยามโดย

$$\alpha(\langle\rangle) = (\phi, h(\langle\rangle), \varphi_f(\langle\rangle))$$

$$\alpha(X \triangleright x) = (x, h(X \triangleright x), \varphi_f(X \triangleright x))$$

ฟังก์ชันการแปลง $\alpha : \mathcal{A}^* \to S$ เป็นความสัมพันธ์ระหว่างสถานะของเครื่องมัวร์แบบบัญญัติและสถานะของระบบการเปลี่ยนสถานะ ความสัมพันธ์ $H = \{(X, \alpha(X)) \mid X \in \mathcal{A}^*\}$ บน $\mathcal{A}^* \times S$ เรียกว่าความสัมพันธ์การจำลองสองทาง (Bisimulation relation) ระหว่างปริภูมิสถานะ $\mathcal{A}^*$ ของเครื่องมัวร์แบบบัญญัติ M_f และปริภูมิสถานะ S ของระบบการเปลี่ยนสถานะ $T_{f,\alpha}$ สำหรับทุก $X \in \mathcal{A}^*$ และ $s \in S$

ประพจน์ $H(X, s)$ หมายถึง $\forall X' \exists s' : X' = X \triangleright x \Rightarrow (s, s') \in R \wedge H(X', s')$

และ $\forall s' \exists X' : (s, s') \in R \Rightarrow X' = X \triangleright x \wedge H(X', s')$

ความสัมพันธ์การจำลองสองทางกำหนดสถานะเริ่มต้นด้วยความสัมพันธ์ $H(\langle\rangle, s_0)$ ทำให้เครื่องมัวร์แบบบัญญัติ M_f สมมูลการจำลองสองทาง (Bisimulation equivalent) กับปริภูมิสถานะ S ของระบบการเปลี่ยนสถานะ $T_{f,\alpha}$ ดังนั้นพฤติกรรมนำเข้า/นำออกของระบบการเปลี่ยนสถานะที่สร้างสมนัยกับพฤติกรรมนำเข้า/นำออกของเครื่องมัวร์แบบบัญญัติ ผลที่ตามมาคือ พฤติกรรมนำเข้า/นำออกของระบบการเปลี่ยนสถานะที่สร้างสมนัยกับพฤติกรรมนำเข้า/นำออกของเครื่องมัวร์แบบบัญญัติ

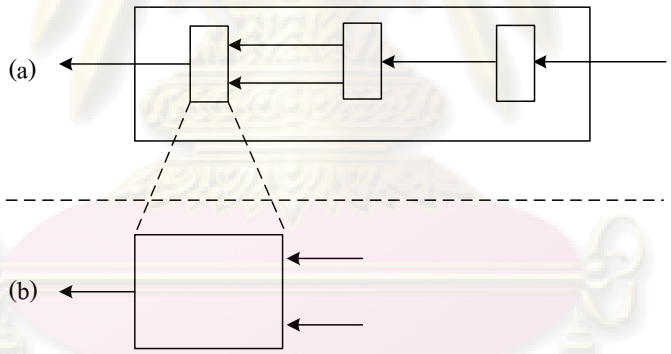
วิธีการแปลงจากฟังก์ชันการประมวลผลกระแสเป็นระบบเปลี่ยนสถานะนำเข้า/นำออกเป็นรูปนัยนิยมทำให้ยังคงสภาพของสมบัตินำเข้า/นำออกเดิมของส่วนประกอบ ดังนั้นผลลัพธ์ที่ได้จากการทวนสอบสมบัตินำเข้า/นำออกของระบบเปลี่ยนสถานะนำเข้า/นำออกจึงเป็นผลลัพธ์เดียวกับการทวนสอบสมบัตินำเข้า/นำออกของฟังก์ชันการประมวลผลกระแสที่กำหนด

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 3
วิธีดำเนินการวิจัย

3.1 แนวคิดและวิธีการวิจัย

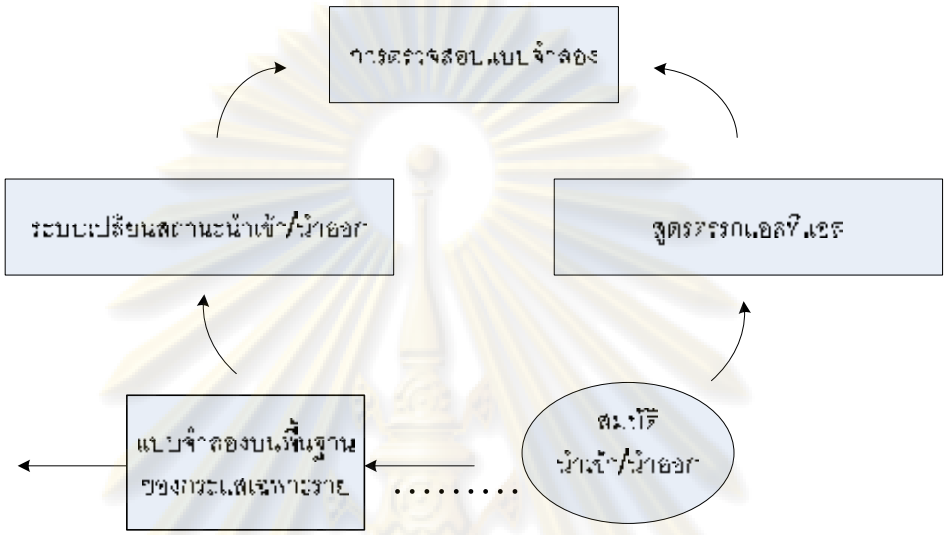
การออกแบบซอฟต์แวร์บนพื้นฐานของกระแสจำลองซอฟต์แวร์ด้วยแบบจำลองประกอบที่แสดงโครงสร้างการสื่อสารกันของแบบจำลองบนพื้นฐานของกระแสเฉพาะราย ดังแสดงในรูปที่ 3.1 โดยที่ 3.1 (a) แสดงแบบจำลองประกอบ และรูปที่ 3.1 (b) แสดงแบบจำลองบนพื้นฐานของกระแสเฉพาะราย วิศวกรซอฟต์แวร์สามารถตรวจสอบสมบัติที่คาดหมายของแบบจำลองประกอบได้ด้วยมือก่อนที่จะทำการแบ่งละเอียดเพื่อการทำให้เกิดผลต่อไป โดยการทวนสอบแบบจำลองประกอบนี้ต้องใช้แรงงานอย่างมากจากผู้เชี่ยวชาญ และเป็นภาระหนักถ้าทวนสอบโดยปราศจากระเบียบวิธีสนับสนุน วิทยานิพนธ์นี้มีจุดประสงค์เพื่อที่จะนำเสนอระเบียบวิธีสำหรับการทวนสอบสมบัติของแบบจำลองประกอบ โดยการประยุกต์การตรวจสอบแบบจำลองไปยังการทวนสอบสมบัตินำเข้า/นำออกที่คาดหมายของแบบจำลองประกอบ ซึ่งทำให้การทวนสอบสามารถทำได้อย่างอัตโนมัติและรวดเร็ว ทำให้ค่าใช้จ่ายในการทวนสอบลดลง



รูปที่ 3.1 แบบจำลองประกอบที่ประกอบด้วยแบบจำลองบนพื้นฐานของกระแสเฉพาะราย

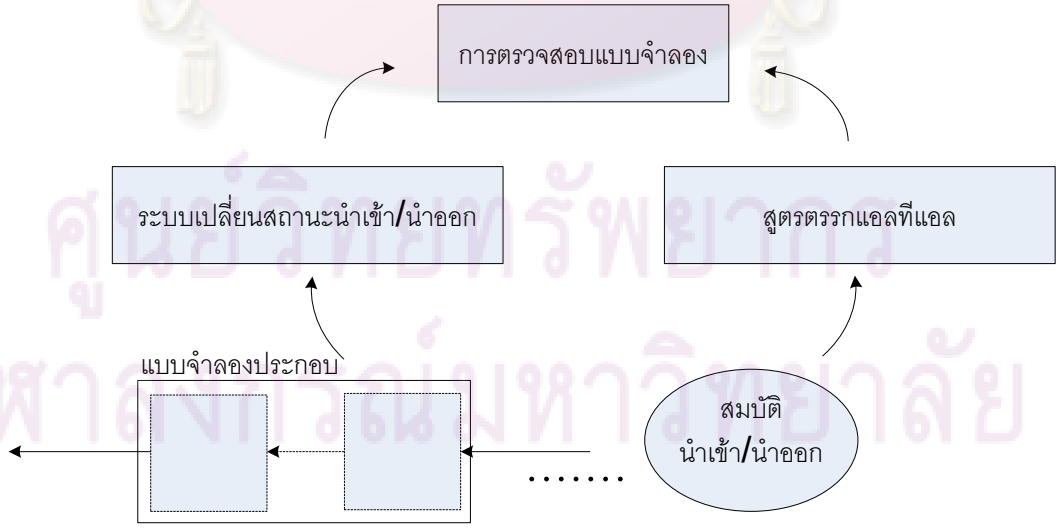
รูปที่ 3.2 แสดงการทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองบนพื้นฐานของกระแสเฉพาะรายด้วยการตรวจสอบแบบจำลอง โดยที่การตรวจสอบแบบจำลองต้องการแบบจำลองบนพื้นฐานของสถานะและข้อกำหนดของสมบัติที่ต้องการทวนสอบในรูปของตรรกศาสตร์เชิงกาลเวลา เนื่องจากส่วนประกอบที่จำลองพฤติกรรมนำเข้า/นำออกด้วยฟังก์ชันการประมวลผลกระแสเป็นแบบจำลองกล่องดำที่ไม่แสดงโครงสร้างสถานะ ดังนั้นจึงไม่สามารถนำไปทวนสอบได้โดยตรง ทำให้ต้องแปลงฟังก์ชันการประมวลผลกระแสเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก ที่เป็นแบบจำลองบนพื้นฐานของสถานะชนิดหนึ่งที่มีความเหมาะสมต่อการทวนสอบด้วยการทวนสอบแบบจำลอง และเขียนสมบัตินำเข้า/นำออกที่คาดหมายให้อยู่ในรูปของสูตรตรรกแอลทีแอล

ซึ่งเป็นตรรกศาสตร์เชิงกาลเวลาชนิดหนึ่งที่สามารถทวนสอบได้ด้วยการทวนสอบแบบจำลอง ดังนั้นด้วยระเบียบวิธีดังกล่าวจึงสามารถทวนสอบสมบัตินำเข้า/นำออกที่คาดหมายของแบบจำลองบนพื้นฐานของกระแสเฉพาะรายด้วยการตรวจสอบแบบจำลองได้



รูปที่ 3.2 การทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองบนพื้นฐานของกระแสเฉพาะรายด้วยการตรวจสอบแบบจำลอง

ในบทนี้นำเสนอระเบียบวิธีการทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบด้วยการตรวจสอบแบบจำลอง โดยการประยุกต์วิธีการทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองบนพื้นฐานของกระแสเฉพาะราย ดังแสดงในรูปที่ 3.3

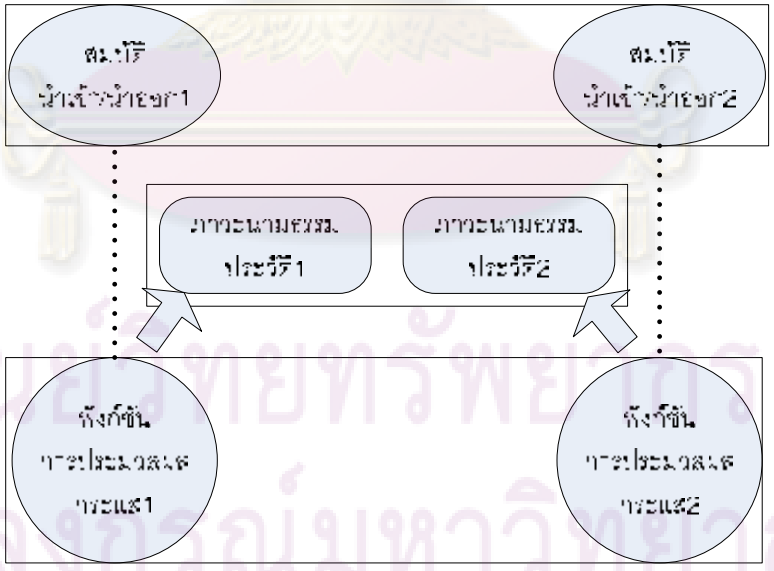


รูปที่ 3.3 แนวคิดการทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบด้วยการตรวจสอบแบบจำลอง

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 3.3 แสดงแนวคิดการทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบด้วยการตรวจสอบแบบจำลอง ซึ่งกระทำได้ก็ต่อเมื่อสามารถประกอบแบบจำลองบนพื้นฐานของกระแสเฉพาะรายที่จำลองด้วยฟังก์ชันการประมวลผลกระแสเป็นแบบจำลองประกอบที่ยังคงสามารถจำลองได้ด้วยฟังก์ชันการประมวลผลกระแส และกำหนดสมบัตินำเข้า/นำออกของแบบจำลองประกอบให้อยู่ในรูปของสูตรตรรกแอลทีแอล

การทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองบนพื้นฐานของกระแสเฉพาะรายด้วยการตรวจสอบแบบจำลอง ที่พฤติกรรมนำเข้า/นำออกของแบบจำลองบนพื้นฐานของกระแสเฉพาะรายสามารถแทนด้วยฟังก์ชันการประมวลผลกระแสจะต้องกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ที่เหมาะสมสำหรับการแปลงฟังก์ชันการประมวลผลกระแสเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก และกำหนดสมบัตินำเข้า/นำออกที่ต้องการทวนสอบ ดังนั้นในการประยุกต์ระเบียบวิธีการทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบบนพื้นฐานของกระแสเฉพาะราย มาใช้ทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบ “การประกอบแบบจำลองบนพื้นฐานของกระแสเฉพาะรายที่แทนพฤติกรรมนำเข้า/นำออกด้วยฟังก์ชันการประมวลผลกระแสเป็นแบบจำลองประกอบที่ยังคงสามารถแทนพฤติกรรมนำเข้า/นำออกด้วยฟังก์ชันการประมวลผลกระแส ย่อมสามารถทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบด้วยการตรวจสอบแบบจำลองได้”



รูปที่ 3.4 การประกอบแบบจำลองบนพื้นฐานของกระแสเฉพาะรายเป็นแบบจำลองประกอบที่สามารถทวนสอบสมบัตินำเข้า/นำออกด้วยการตรวจสอบแบบจำลอง

จากแนวคิดดังกล่าว ทำให้งานวิจัยนี้สนใจพิจารณาความสัมพันธ์ของลักษณะ 3 ลักษณะของแบบจำลองประกอบกับแบบจำลองบนพื้นฐานของกระแสเฉพาะรายคือ การประกอบแบบจำลองบนพื้นฐานของกระแสเฉพาะรายที่แทนพฤติกรรมนำเข้า/นำออกด้วยฟังก์ชันการประมวลผลกรแสเป็นแบบจำลองประกอบที่ยังคงสามารถแทนพฤติกรรมนำเข้า/นำออกได้ด้วยฟังก์ชันการประมวลผลกระแส ความสัมพันธ์ระหว่างฟังก์ชันภาวะนามธรรมประวัติของแบบจำลองประกอบกับฟังก์ชันภาวะนามธรรมประวัติของแบบจำลองบนพื้นฐานของกระแสเฉพาะราย และความสัมพันธ์ของสมบัตินำเข้า/นำออกของแบบจำลองประกอบกับสมบัตินำเข้า/นำออกของแบบจำลองบนพื้นฐานของกระแสเฉพาะรายดังแสดงในรูปที่ 3.4

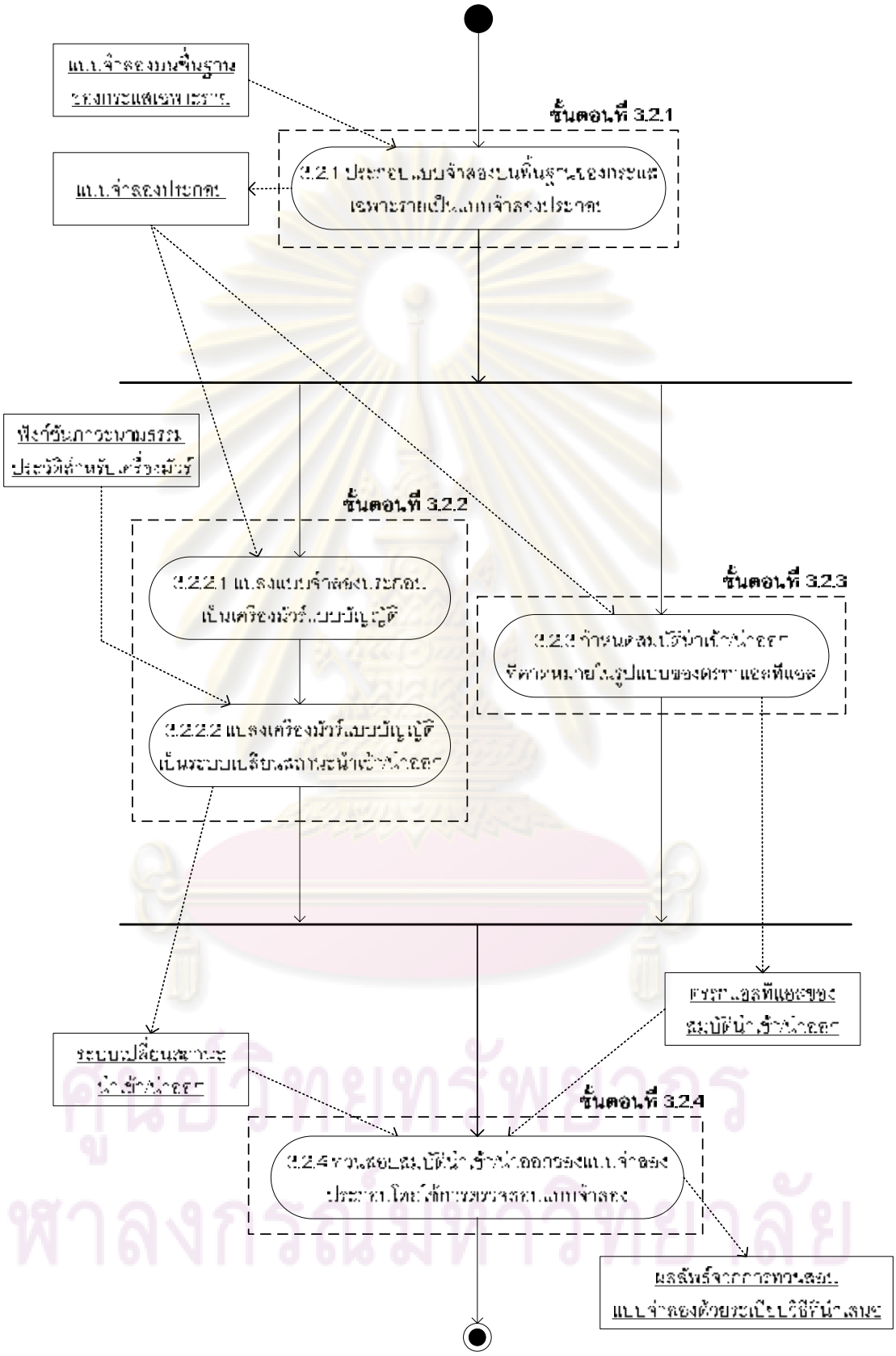
3.2 ระเบียบวิธีการทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบ

ระเบียบวิธีการทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบด้วยการตรวจสอบแบบจำลอง ซึ่งประกอบด้วย 4 ขั้นตอนคือ ขั้นตอน 3.2.1. แสดงการประกอบฟังก์ชันการประมวลผลกระแสเป็นแบบจำลองประกอบ ขั้นตอน 3.2.2. แสดงการแปลงแบบจำลองประกอบเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก ขั้นตอน 3.2.3. แสดงการกำหนดสมบัตินำเข้า/นำออกที่คาดหมายในรูปแบบของตรรกแอลทีแอล และขั้นตอน 3.2.4. แสดงการทวนสอบแบบจำลองประกอบโดยใช้การตรวจสอบแบบจำลอง โดยที่ขั้นตอนที่ 3.2.2 ประกอบด้วย 2 ขั้นตอนย่อยคือ ขั้นตอน 3.2.2.1 แสดงการแปลงแบบจำลองประกอบเป็นเครื่องมัวร์แบบบัญญัติ และขั้นตอน 3.2.2.2 แสดงการแปลงเครื่องมัวร์แบบบัญญัติเป็นระบบการเปลี่ยนสถานะนำเข้า/นำออก ดังแสดงในรูปที่ 3.5 รายละเอียดของแต่ละขั้นตอนมีดังต่อไปนี้

3.2.1 การประกอบฟังก์ชันการประมวลผลกระแสเป็นแบบจำลองประกอบ

แบบจำลองประกอบ แสดงโครงสร้างการสื่อสารกันระหว่างส่วนประกอบดังแสดงในรูปที่ 3.1 (a) ดังนั้นหากนำส่วนประกอบย่อยที่จำลองด้วยฟังก์ชันการประมวลผลกระแสมาประกอบตามโครงสร้างการสื่อสาร ก็จะได้แบบจำลองประกอบที่ต้องการจากส่วนประกอบย่อย

เนื่องจากฟังก์ชันการประมวลผลกระแส เป็นฟังก์ชันที่มีสมบัติที่สำคัญคือ เป็นฟังก์ชันทางเดียว และต่อเนื่อง และตัวดำเนินการประกอบแบบลำดับ ตัวดำเนินการประกอบแบบขนานและผลป้อนกลับ มีสมบัติการคงสภาพสมบัติฟังก์ชันทางเดียว และต่อเนื่อง จากการประกอบฟังก์ชันทางเดียว และต่อเนื่องนั่นคือ เมื่อประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบย่อยด้วยตัวดำเนินการทั้ง 3 ชนิด ผลลัพธ์ที่ได้คือ ฟังก์ชันการประมวลผลกระแสของแบบจำลองประกอบที่สามารถทวนสอบสมบัตินำเข้า/นำออกด้วยการตรวจสอบแบบจำลอง



รูปที่ 3.5 แผนภาพแสดงขั้นตอนการดำเนินการวิจัย

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

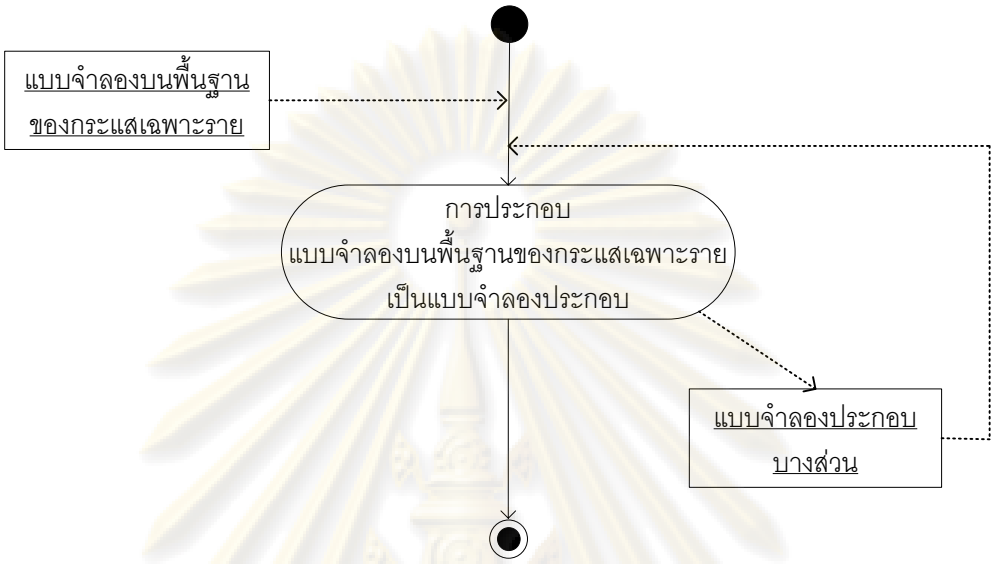
ตัวดำเนินการประกอบแบบลำดับ ตัวดำเนินการประกอบแบบขนาน และผลป้อนกลับ มีระดับการทำก่อน (Precedence) ที่ต่างกันคือ ผลป้อนกลับ ตัวดำเนินการประกอบแบบลำดับและตัวดำเนินการประกอบแบบขนาน มีระดับการทำก่อนเรียงจากน้อยไปหามากตามลำดับ และด้วยส่วนประกอบการจัดเส้นทาง (Routing component) ได้แก่ส่วนประกอบการไขว้ (Cross component) ส่วนประกอบเอกลักษณ์ (Identity component) ส่วนประกอบสำเนา (Copy component) ส่วนประกอบต้นทาง (Source component) และส่วนประกอบปลายทาง (Sink component) และตัวดำเนินการทั้ง 3 ชนิด สามารถเขียนแทนแบบจำลองตามโครงสร้างการสื่อสารทุกรูปแบบ

ผู้วิจัยใช้วิธีการทวนสอบแบบจำลองประกอบโดยการแทนแบบจำลองประกอบด้วยฟังก์ชันการประมวลผลกระแสที่ได้รับจากประกอบฟังก์ชันการประมวลผลกระแสของแบบจำลองบนพื้นฐานของกระแสเฉพาะรายด้วยตัวดำเนินการประกอบ ด้วยวิธีการนี้ทำให้สามารถแบ่งแบบจำลองประกอบเป็นแบบจำลองประกอบบางส่วน ซึ่งแต่ละแบบจำลองประกอบบางส่วนสามารถแทนด้วยฟังก์ชันการประมวลผลกระแสที่สามารถทวนสอบด้วยวิธีการตรวจสอบแบบจำลอง และจะทำการประกอบแบบจำลองประกอบบางส่วนไปจนกระทั่งได้แบบจำลองประกอบดังแสดงในรูปที่ 3.6 ซึ่งแสดงการประกอบแบบจำลองประกอบจากฟังก์ชันการประมวลผลกระแส โดยการประกอบแบบจำลองประกอบนี้จะเกิดจากการประกอบฟังก์ชันการประมวลผลกระแสจำนวนครั้งละสองฟังก์ชัน ผลลัพธ์ที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสด้วยตัวดำเนินการประกอบแบบลำดับคือ แบบจำลองประกอบบางส่วน ซึ่งสามารถจำลองด้วยฟังก์ชันการประมวลผลกระแส และนำกลับไปประกอบด้วยตัวดำเนินการประกอบแบบลำดับ จนกระทั่งได้แบบจำลองประกอบที่ต้องการ

3.2.2. การแปลงแบบจำลองประกอบเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

ผลลัพธ์ที่ได้รับจากการประกอบส่วนประกอบตามโครงสร้างการสื่อสารคือ แบบจำลองประกอบ หรือแบบจำลองประกอบบางส่วน สามารถพิจารณาได้ในฐานะที่เป็นส่วนประกอบหนึ่งซึ่งสามารถจำลองได้ด้วยฟังก์ชันการประมวลผลกระแสที่มีสมบัติทางเดียว และต่อเนื่อง ดังนั้นจึงสามารถแปลงแบบจำลองประกอบเป็นระบบเปลี่ยนสถานะนำเข้า/นำออกได้โดยสามารถแบ่งการแปลงเป็นสองขั้นตอนคือ ขั้นตอนแรกทำการแปลงฟังก์ชันการประมวลผลกระแสที่ได้รับจากการประกอบเป็นเครื่องมัวร์แบบบัญญัติ โดยการแทนประวัตินำเข้าของฟังก์ชันการประมวลผลกระแสเป็นสถานะของเครื่องมัวร์แบบบัญญัติ หลังจากนั้นทำการแปลงเครื่องมัวร์แบบบัญญัติที่ได้รับเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก โดยอาศัยฟังก์ชันภาวะนามธรรมประวัติเพื่อลดจำนวนสถานะในระบบ และสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกได้รับการแปลงสถานะของเครื่องมัวร์

แบบบัญญัติด้วยฟังก์ชันการแปลง โดยฟังก์ชันการแปลงทำการรวมนำเข้าและนำออกกับสถานะภายในที่ได้รับจากการแปลงสถานะของเครื่องมัวร์แบบบัญญัติด้วยฟังก์ชันนามธรรมประวัติ



รูปที่ 3.6 การประกอบแบบจำลอง

จากการที่สถานะของเครื่องมัวร์แบบบัญญัติเกิดจากการแทนด้วยประวัตินำเข้า และนำออกคำนวณได้จากสถานะ ดังนั้นพฤติกรรมนำเข้า/นำออกของส่วนประกอบสามารถอนุมานได้จากโครงสร้างการเปลี่ยนสถานะของเครื่องมัวร์แบบบัญญัติ และจากการที่สถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกเกิดจากการรวมนำเข้าและนำออกกับสถานะภายในที่ได้รับจากการแปลงสถานะของเครื่องมัวร์แบบบัญญัติด้วยฟังก์ชันนามธรรมประวัติ ดังนั้นโครงสร้างการเปลี่ยนสถานะของเครื่องมัวร์แบบบัญญัติสามารถอนุมานได้จากโครงสร้างการเปลี่ยนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออก นั่นคือความถูกต้องของสมบัตินำเข้า/นำออกที่คาดหมายของระบบเปลี่ยนสถานะนำเข้า/นำออก หมายถึงความถูกต้องของสมบัตินำเข้า/นำออกที่คาดหมายของพฤติกรรมนำเข้า/นำออกเดิมของส่วนประกอบ

ฟังก์ชันภาวะนามธรรมประวัติมีบทบาทสำคัญมากในการแปลงฟังก์ชันการประมวลผลกระแสเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก โดยฟังก์ชันการประมวลผลกระแสเดียวกันสามารถแปลงเป็นระบบเปลี่ยนสถานะนำเข้า/นำออกได้หลายแบบขึ้นอยู่กับฟังก์ชันภาวะนามธรรมประวัติ งานวิจัยนี้ผู้วิจัยวิเคราะห์ความสัมพันธ์ของฟังก์ชันภาวะนามธรรมประวัติของแบบจำลองประกอบกับฟังก์ชันภาวะนามธรรมประวัติของแบบจำลองบนพื้นฐานของกระแสเฉพาะรายที่นำมาประกอบเป็นแบบจำลองประกอบด้วยตัวดำเนินการประกอบแบบลำดับ รายละเอียดดูได้ในบทที่ 5

3.2.3 การกำหนดสมบัตินำเข้า/นำออกที่คาดหมายในรูปแบบของตรรกแอลทีแอล

การกำหนดสมบัตินำเข้า/นำออกที่คาดหมายให้อยู่ในรูปที่สามารถทวนสอบด้วยการตรวจสอบแบบจำลอง สามารถทำได้โดยเขียนสมบัตินำเข้า/นำออกที่คาดหมายให้อยู่ในรูปตรรกศาสตร์เชิงกาลเวลา งานวิจัยนี้ได้กำหนดสมบัตินำเข้า/นำออกที่คาดหมายในรูปของตรรกแอลทีแอลซึ่งเป็นตรรกศาสตร์เชิงกาลเวลาชนิดหนึ่งที่สามารถทวนสอบได้ด้วยการตรวจสอบแบบจำลอง

3.2.4 การทวนสอบแบบจำลองประกอบโดยใช้การตรวจสอบแบบจำลอง

การทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบด้วยการตรวจสอบแบบจำลองสามารถทำได้จากการนำระบบเปลี่ยนสถานะนำเข้า/นำออกซึ่งเป็นแบบจำลองบนพื้นฐานของสถานะและสมบัตินำเข้า/นำออกที่คาดหมายที่อยู่ในรูปตรรกแอลทีแอลด้วยตัวตรวจสอบแบบจำลอง ผลการตรวจสอบสามารถบอกได้ว่าแบบจำลองสอดคล้องกับสมบัติเชิงการเวลาที่กำหนดหรือไม่ ในกรณีที่ไม่สอดคล้อง ตัวตรวจสอบแบบจำลองจะแสดงตัวอย่างวิถีที่เกิดข้อผิดพลาดในแบบจำลอง ทำให้ง่ายต่อการปรับปรุงแก้ไขต่อไป

3.3 สรุป

ระเบียบวิธีที่นำเสนอสามารถทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบ โดยการประยุกต์ระเบียบวิธีสำหรับการทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองบนพื้นฐานของกระแสเฉพาะราย ไปใช้ในการทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบที่ได้เกิดจากการประกอบแบบจำลองบนพื้นฐานของกระแสเฉพาะรายด้วยตัวดำเนินการประกอบ ทำให้ยังคงสามารถแทนแบบจำลองประกอบด้วยฟังก์ชันการประมวลผลกระแสที่สามารถทวนสอบสมบัติได้ด้วยการตรวจสอบแบบจำลอง

ด้วยแนวความคิดนี้ผู้วิจัยได้นำเสนอฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองบนพื้นฐานของกระแสเฉพาะรายในบทที่ 4 ที่ใช้เพื่อที่จะลดจำนวนสถานะในระบบเปลี่ยนสถานะนำเข้า/นำออก ซึ่งมีบทบาทที่สำคัญมากในการทวนสอบสมบัตินำเข้า/นำออก และนำเสนอวิธีการกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบในบทที่ 5

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 4
ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองบนพื้นฐานของกระแสเฉพาะราย

บทนี้นำเสนอฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองบนพื้นฐานของกระแสเฉพาะราย เพื่อที่จะใช้ในการลดจำนวนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออก ซึ่งเป็นแบบจำลองบนพื้นฐานของสถานะที่มีความเหมาะสมสำหรับการทวนสอบสมบัตินำเข้า/นำออกที่คาดหมายด้วยการตรวจสอบแบบจำลอง โดยนำเสนอฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ตามกลุ่มฟังก์ชันการประมวลผลกระแส 6 กลุ่ม ได้แก่ ฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบที่อิสระจากประวัติ ฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต ฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบคงที่ ฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบแบบมีคำร้องขอ ฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบสแกน และฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบที่ต้องการประวัติ

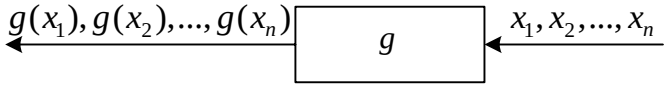
ต่อไปนี้เป็นการนำเสนอฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของฟังก์ชันการประมวลผลกระแสในแต่ละกลุ่ม ซึ่งแต่ละกลุ่มเนื้อหาที่นำเสนอประกอบด้วย ลักษณะของส่วนประกอบ ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของฟังก์ชันการประมวลผลกระแสของส่วนประกอบดังกล่าว นิยามของกลุ่มของส่วนประกอบโดยอาศัยฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ที่นำเสนอ และตัวอย่างการแปลงฟังก์ชันการประมวลผลกระแสเป็นระบบเปลี่ยนสถานะนำเข้า/นำออกโดยอาศัยฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ที่นำเสนอ

4.1 ฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบที่อิสระจากประวัติ

ตอนย่อยนี้นำเสนอลักษณะของกลุ่มส่วนประกอบที่อิสระจากประวัติ ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบที่อิสระจากประวัติ นิยามของกลุ่มส่วนประกอบที่อิสระจากประวัติโดยอาศัยฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ที่นำเสนอ และตัวอย่างการแปลงฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบที่อิสระจากประวัติเป็นระบบเปลี่ยนสถานะนำเข้า/นำออกโดยอาศัยฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ที่นำเสนอ

4.1.1 ลักษณะของส่วนประกอบที่อิสระจากประวัติ

จากลักษณะของส่วนประกอบที่อิสระจากประวัติที่แสดงใน 2.2.1 สามารถแสดงได้ดังรูปที่ 4.1



รูปที่ 4.1 ส่วนประกอบที่อิสระจากประวัติ

4.1.2 **ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบที่อิสระจากประวัติ**

ภาคขยายนำออกของส่วนประกอบกลุ่มนี้คือ $\varphi_{map'(g)}(X \triangleright x) = g(x)$ นั่นคือ ภาคขยายนำออกขึ้นอยู่กับนำเข้าตัวปัจจุบันเท่านั้น เนื่องจากนำออกอิสระจากประวัตินำเข้าที่เกิดขึ้นก่อนหน้า ดังนั้นฟังก์ชันภาวะนามธรรมประวัติไม่จำเป็นต้องเก็บนำเข้าก่อนหน้า

ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบกลุ่มนี้คือ ฟังก์ชัน $current$ ที่เก็บสมาชิกตัวปัจจุบันไว้ โดยไม่สนใจสมาชิกส่วนที่เหลือ นิยามดังนี้

$$current : \mathcal{A}^* \to \mathcal{A}^{\leq 1}$$

โดยที่

$$current(\langle\rangle) = \langle\rangle \tag{4.1}$$

$$current(X \triangleright x) = \langle x \rangle \tag{4.2}$$

(4.1) หมายถึง ไม่มีนำออกเมื่อไม่มีนำเข้า

(4.2) หมายถึง ส่งกระแสนำเข้าไปเป็นกระแสของสมาชิกตัวหลังสุด

ฟังก์ชัน $current$ เก็บสมาชิกตัวปัจจุบันไว้ โดยไม่สนใจสมาชิกส่วนที่เหลือดังนั้น $current(X) = current(Y) \Rightarrow \varphi_{map'}(X) = \varphi_{map'}(Y)$ เพราะ $current(X) = current(Y)$ หมายถึง สมาชิกตัวสุดท้ายของกระแส X และ Y เหมือนกัน ทำให้ภาคขยายนำออกเหมือนกันด้วย นั่นคือฟังก์ชัน $current$ มีสมบัตินำออกเข้ากันได้ และ $current(X) = current(Y)$ $\Rightarrow current(X \triangleright x) = current(Y \triangleright x)$ เพราะสมาชิกตัวสุดท้ายของกระแส $X \triangleright x$ เหมือนกับสมาชิกตัวสุดท้ายของกระแส $Y \triangleright x$ นั่นคือฟังก์ชัน $current$ มีสมบัติการเปลี่ยนสถานะปิด ดังนั้นฟังก์ชัน $current$ เป็นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบกลุ่มนี้

4.1.3 **นิยามของส่วนประกอบที่อิสระจากประวัติ**

นิยาม 4.1 ส่วนประกอบที่อิสระจากประวัติ

ส่วนประกอบที่จำลองพฤติกรรมด้วยฟังก์ชันการประมวลผลกระแสที่มีฟังก์ชัน $current$ เป็นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ เรียกว่า ส่วนประกอบที่อิสระจากประวัติ

4.1.4 ตัวอย่างการแปลงส่วนประกอบที่อิสระจากประวัติเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

ตัวอย่างส่วนประกอบที่อิสระจากประวัติคือ ส่วนประกอบสำเนาดังแสดงในตัวอย่าง 2.1 ต่อไปนี้ผู้วิจัยนำเสนอระบบเปลี่ยนสถานะนำเข้า/นำออกที่ได้รับจากฟังก์ชันภาวะนามธรรมประวัติ *current*

4.1.4.1 การแปลงส่วนประกอบสำเนาเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

ต่อไปนี้เป็นการแปลงส่วนประกอบสำเนาเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก โดยแปลงส่วนประกอบสำเนาผ่านทางเครื่องมัวร์แบบบัญญัติ จากนั้นจึงแปลงเครื่องมัวร์แบบบัญญัติที่ได้รับเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

4.1.4.1.1 การแปลงส่วนประกอบสำเนาเป็นเครื่องมัวร์แบบบัญญัติ

ส่วนประกอบสำเนาสามารถแปลงเป็นเครื่องมัวร์แบบบัญญัติได้ดังนี้

$$M_{duplicate} = (\mathcal{A}^*, \mathcal{A}, \mathcal{A}^*, \delta,\ \varphi_{duplicate}, \langle\rangle)$$

โดยที่

ฟังก์ชันนำออกเดี่ยว $\varphi_{duplicate}$ คือ

$$\varphi_{duplicate}(\langle\rangle) = \langle\rangle \tag{4.3}$$

$$\varphi_{duplicate}(X \lhd x) = \langle x, x\rangle \tag{4.4}$$

(4.3) หมายถึง ไม่มีนำออกเมื่อไม่มีนำเข้า

(4.4) หมายถึง สำเนาสมาชิกตัวสุดท้ายของกระแสนำเข้า

4.1.4.1.2 การแปลงเครื่องมัวร์แบบบัญญัติเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

เครื่องมัวร์แบบบัญญัติ $M_{duplicate}$ สามารถแปลงเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก $T_{duplicate,\alpha}$ ได้ดังนี้

$$T_{duplicate,\alpha} = (\mathcal{A}, \mathcal{A}^*, Q, S, R, s_0)$$

โดยที่

$$Q = \mathcal{A}^{\leq 1} \tag{4.5}$$

$$S, = (\mathcal{A} \cup \{\phi\}) \times \mathcal{A}^{\leq 1} \times \mathcal{A}^* \tag{4.6}$$

$$R, = \{(\alpha(X), \alpha(X \rhd x)) \mid X \in \mathcal{A}^*, x \in \mathcal{A}\} \tag{4.7}$$

$$s_0 = (\phi, current(\langle\rangle), \varphi_{duplicate}(\langle\rangle)) = (\phi, \langle\rangle, \langle\rangle) \tag{4.8}$$

เมื่อ

$$\alpha(\langle\rangle) = (\phi, current(\langle\rangle), \varphi_{duplicate}(\langle\rangle))$$

$$\alpha(X \rhd x) = (x, current(X \rhd x), \varphi_{duplicate}(X \rhd x))$$

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

(4.5) หมายถึง เซตของสถานะภายในคือ กระแส $\mathcal{A}^{\leq 1}$ ซึ่งได้รับจากการลดจำนวนสถานะที่ได้รับจากกระแสนำเข้าด้วยฟังก์ชันภาวะนามธรรมประวัติ *current*

(4.6) หมายถึง ปริภูมิสถานะ S ที่เกิดจากการรวมเซตข้อมูลนำเข้า สถานะภายใน และเซตข้อมูลนำออก ด้วยฟังก์ชันการแปลง α สามารถสร้างสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกภายใต้ความสัมพันธ์การจำลองสองทางกับสถานะของเครื่องมัวร์แบบบัญญัติได้ดังนี้

สร้างสถานะจากสถานะเริ่มต้นที่เป็นกระแสว่าง

$$\langle\rangle \leftrightarrow_H \{(\phi, \langle\rangle, \langle\rangle)\}$$

สร้างสถานะจากสถานะภายในไม่เป็นกระแสว่าง

$$\langle x\rangle \leftrightarrow_H \{(x, \langle x\rangle, \langle x, x\rangle) \mid x \in \mathcal{A}\}$$

(4.7) หมายถึง ความสัมพันธ์การเปลี่ยนสถานะ R แสดงความสัมพันธ์ของสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออก

การเปลี่ยนสถานะจากสถานะเริ่มต้น เมื่อได้รับนำเข้า $x \in \mathcal{A}$

$$(\phi, \langle\rangle, \langle\rangle) \rightarrow (x, \langle x\rangle, \langle x, x\rangle)$$

การเปลี่ยนสถานะจากสถานะ $x \in \mathcal{A}$ เมื่อได้รับนำเข้า $y \in \mathcal{A}$

$$(x, \langle x\rangle, \langle x, x\rangle) \rightarrow (y, \langle y\rangle, \langle y, y\rangle)$$

เมื่อ $x, y \in \mathcal{A}$

(4.8) หมายถึง กำหนดสถานะเริ่มต้นของระบบเปลี่ยนสถานะนำเข้า/นำออก s_0 คือ $(\phi, \langle\rangle, \langle\rangle)$

จากระบบเปลี่ยนสถานะนำเข้า/นำออก $T_{duplicate,\alpha}$ สามารถเขียนให้อยู่ในรูปตารางการเปลี่ยนสถานะ ดังตารางที่ 4.1

ตารางที่ 4.1 ตารางการเปลี่ยนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกของส่วนประกอบสำเนา

State			*State′*		
input	*internal state*	*output*	*input′*	*internal state′*	*output′*
ϕ	$\langle\rangle$	$\langle\rangle$	x	$\langle x\rangle$	$\langle x, x\rangle$
x	$\langle x\rangle$	$\langle x, x\rangle$	y	$\langle y\rangle$	$\langle y, y\rangle$
$x, y \in \mathcal{A}$					

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

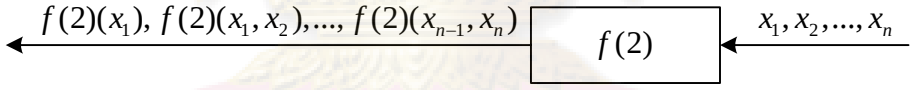
ฟังก์ชันภาวะนามธรรมประวัติ *current* ที่นำเสนอนี้มีความเหมาะสมสำหรับการทวนสอบสมบัตินำเข้า/นำออกด้วยการตรวจสอบแบบจำลอง เนื่องจากสามารถลดปริภูมิสถานะอนันต์ของเครื่องมัวร์แบบบัญญัติเป็นปริภูมิสถานะจำกัดของระบบเปลี่ยนสถานะนำเข้า/นำออก

4.2 ฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต

ตอนย่อยนี้นำเสนอลักษณะของกลุ่มส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต นิยามของกลุ่มส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขตโดยอาศัยฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ที่นำเสนอ และตัวอย่างการแปลงฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขตเป็นระบบเปลี่ยนสถานะนำเข้า/นำออกโดยอาศัยฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ที่นำเสนอ

4.2.1 ลักษณะของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต

จากลักษณะของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขตที่แสดงใน 2.2.1 ในกรณีที่แทนค่าความยาวที่กำหนดด้วย 2 สามารถแสดงได้ดังรูปที่ 4.2



รูปที่ 4.2 ส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต

4.2.2 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต

ภาคขยายนำออกของส่วนประกอบกลุ่มนี้คือ ภาคขยายนำออกของกระแสย่อย $n+1$ ตัวสุดท้ายของกระแสนำเข้าในกรณีที่กระแสนำเข้ามีความยาวมากกว่าหรือเท่ากับค่าคงตัวที่กำหนด หรือภาคขยายนำออกของกระแสย่อยที่มีสมาชิกเท่าที่มีอยู่ในกรณีที่กระแสนำเข้ามีความยาวน้อยกว่าค่าคงตัวที่กำหนด

ฟังก์ชันภาวะนามธรรมสำหรับเครื่องมัวร์ของส่วนประกอบกลุ่มนี้คือ ฟังก์ชัน $last(n+1)$ โดยที่ฟังก์ชัน $last(n)$ มีพฤติกรรมดังนิยามใน 2.2.1 ให้ผลลัพธ์คือ สมาชิก $n+1$ ตัวหลังสุดของกระแสนำเข้าเท่าที่เป็นไปได้ นั่นก็คือสมาชิก $n+1$ ตัวสุดท้ายของกระแสนำเข้าในกรณีที่กระแสนำเข้ามีความยาวมากกว่าหรือเท่ากับค่าคงตัวที่กำหนด หรือกระแสนำเข้าเดิมในกรณีที่กระแสนำเข้ามีความยาวน้อยกว่าค่าคงตัวที่กำหนด ดังนั้น $last(n+1)(X) = last(n+1)(Y)$

$\Rightarrow \varphi_f(X) = \varphi_f(Y)$ เพราะ $last(n+1)\ (X) = last(n+1)(Y)$ หมายถึง $X = Y$ ในกรณีที่กระแสนำเข้ามีความยาวน้อยกว่าค่าคงตัวที่กำหนด หรือสมาชิก $n+1$ ตัวหลังสุดของกระแสนำเข้า X และกระแสนำเข้า Y เหมือนกันตำแหน่งต่อตำแหน่ง ทำให้ $\varphi_f(X) = \varphi_f(Y)$ นั่นคือ ฟังก์ชัน $last(n+1)$ มีสมบัตินำออกเข้ากันได้ และ $last(n+1)(X) = last(n+1)(Y)$ $\Rightarrow last(n+1)(X \triangleright x) = last(n+1)(Y \triangleright x)$ เนื่องจาก $last(n+1)(X) = last(n+1)(Y)$ ในกรณีที่กระแสนำเข้ามีความยาวน้อยกว่าค่าคงตัวที่กำหนดหมายถึง $X = Y$ ทำให้ $last(n+1)(X \triangleright x) = last(n+1)(Y \triangleright x)$ เพราะ $X \triangleright x = Y \triangleright x$ หรือ ในกรณีที่กระแสนำเข้ามีความยาวน้อยกว่าค่าคงตัวที่กำหนดหมายถึงสมาชิก $n+1$ ตัวหลังสุดของกระแสนำเข้า X และกระแสนำเข้า Y เหมือนกันตำแหน่งต่อตำแหน่ง ทำให้ $last(n+1)(X \triangleright x) = last(n+1)(Y \triangleright x)$ เพราะว่า สมาชิก $n+1$ ตัวหลังสุดของกระแสนำเข้า $X \triangleright x$ และกระแสนำเข้า $Y \triangleright x$ เหมือนกันตำแหน่งต่อตำแหน่ง นั่นคือ ฟังก์ชัน $last(n+1)$ มีสมบัติการเปลี่ยนสถานะปิด ดังนั้น ฟังก์ชัน $last(n+1)$ เป็นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบกลุ่มนี้

4.2.3 นิยามของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต

นิยาม 4.2 ส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต

ส่วนประกอบที่จำลองด้วยฟังก์ชันการประมวลผลกระแสที่มีฟังก์ชัน $last(n+1)$ เป็นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ เรียกว่า ส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต

4.2.4 ตัวอย่างการแปลงส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขตเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

ตัวอย่างส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขตคือ ส่วนประกอบเรจิสเตอร์การเลื่อนดังแสดงในตัวอย่างที่ 2.2 ต่อไปนี้ผู้วิจัยนำเสนอระบบเปลี่ยนสถานะนำเข้า/นำออกที่ได้รับจากฟังก์ชันภาวะนามธรรมประวัติ $last(n+1)$

4.2.4.1 การแปลงส่วนประกอบเรจิสเตอร์การเลื่อนเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

ต่อไปนี้เป็นการแปลงส่วนประกอบเรจิสเตอร์การเลื่อนเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก โดยแปลงส่วนประกอบเรจิสเตอร์การเลื่อนผ่านทางเครื่องมัวร์แบบบัญญัติ จากนั้นจึงแปลงเครื่องมัวร์แบบบัญญัติที่ได้รับเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

4.2.4.1.1 การแปลงส่วนประกอบเรจิสเตอร์การเลื่อนเป็นเครื่องมัวร์แบบบัญญัติ

ส่วนประกอบเรจิสเตอร์การเลื่อนสามารถแปลงเป็นเครื่องมัวร์แบบบัญญัติได้ดังนี้

$$M_{shift(n)} = (\mathcal{A}^*, \mathcal{A}, \mathcal{A}^*, \delta,\ \varphi_{shift(n)}, \langle\rangle)$$

โดยที่

ฟังก์ชันนำออกเดี่ยว $\varphi_{shift(n)}$ คือ

$$\varphi_{shift(n)}(X) = \langle\rangle \quad (4.9)$$

$$\varphi_{shift(n)}(Y) = \langle Y_{[|Y|-(n+1)]} \rangle \quad (4.10)$$

เมื่อ $|X| \le n$ และ $|Y| > n$

(4.9) หมายถึงไม่มีนำออกเมื่อกระแสนำเข้ามีขนาดน้อยกว่าหรือเท่ากับ n

(4.10) หมายถึงนำออกคือสมาชิกตัวที่ $n+1$ นับมาจากหลังสุดของกระแสนำเข้าเมื่อกระแสนำเข้ามีขนาดมากกว่า n

4.2.4.1.2 การแปลงเครื่องมัวร์แบบบัญญัติเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

เครื่องมัวร์แบบบัญญัติ $M_{shift(n)}$ สามารถแปลงเป็นระบบเปลี่ยนสถานะนำเข้า/ นำออก $T_{shift(n),\alpha}$ ได้ดังนี้

$$T_{shift(n),\alpha} = (\mathcal{A}, \mathcal{A}^*, Q, S, R, s_0)$$

โดยที่

$$Q = \mathcal{A}^{\le n+1} \quad (4.11)$$

$$S, = (\mathcal{A} \cup \{\phi\}) \times \mathcal{A}^{\le n+1} \times \mathcal{A} \quad (4.12)$$

$$R, = \{(\alpha(X), \alpha(X \rhd x)) \mid X \in \mathcal{A}^*, x \in \mathcal{A}\} \quad (4.13)$$

$$s_0 = (\phi, last(n+1), \varphi_{shift(n)}(\langle\rangle)) = (\phi, \langle\rangle, \langle\rangle) \quad (4.14)$$

เมื่อ

$$\alpha(\langle\rangle) = (\phi, last(n+1), \varphi_{shift(n)}(\langle\rangle))$$

$$\alpha(X \rhd x) = (x, last(n+1)(X \rhd x), \varphi_{shift(n)}(X \rhd x))$$

(4.11) หมายถึง เซตของสถานะภายในคือ กระแส $\mathcal{A}^{\le n+1}$ ซึ่งได้รับจากการลดจำนวนสถานะที่ได้รับจากกระแสนำเข้าด้วยฟังก์ชันภาวะนามธรรมประวัติ $last(n+1)$

(4.12) หมายถึง ปริภูมิสถานะ S ที่เกิดจากการรวมเซตข้อมูลนำเข้า สถานะภายใน และเซตข้อมูลนำออกด้วยฟังก์ชันการแปลง α สามารถสร้างสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกภายใต้ความสัมพันธ์การจำลองสองทางกับสถานะของเครื่องมัวร์แบบบัญญัติได้ดังนี้

สร้างสถานะจากสถานะเริ่มต้นที่เป็นกระแสว่าง

$$\langle\rangle \leftrightarrow_H \{(\phi, \langle\rangle, \langle\rangle)\}$$

สร้างสถานะเมื่อ $|X \rhd x| < n+1$

$$X \rhd x \leftrightarrow_H \{(x, X \rhd x, \langle\rangle) \mid x \in \mathcal{A}, x \in \mathcal{A}^*\}$$

สร้างสถานะเมื่อ $|y \lhd X \rhd x| = n+1$

$$y \lhd X \rhd x \leftrightarrow_H \{(x, y \lhd X \rhd x, \langle y \rangle) \mid x, y \in \mathcal{A}, x \in \mathcal{A}^*\}$$

(4.13) หมายถึง ความสัมพันธ์การเปลี่ยนสถานะ R แสดงความสัมพันธ์ของสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออก

การเปลี่ยนสถานะจากสถานะเริ่มต้น

$$(\phi, \langle\rangle, \langle\rangle) \rightarrow (x, \langle x \rangle, \langle\rangle)$$

การเปลี่ยนสถานะ เมื่อ $|X \rhd x| < n$

$$(x, X \rhd x, \langle\rangle) \rightarrow (y, X \rhd x \rhd y, \langle\rangle)$$

การเปลี่ยนสถานะ เมื่อ $|z \lhd X \rhd x| = n$

$$(x, z \lhd X \rhd x, \langle\rangle) \rightarrow (y, X \rhd x \rhd y, \langle z \rangle)$$

การเปลี่ยนสถานะ เมื่อ $|z \lhd w \lhd X \rhd x| = n+1$

$$(x, z \lhd w \lhd X \rhd x, \langle z \rangle) \rightarrow (y, w \lhd X \rhd x \rhd y, \langle w \rangle)$$

เมื่อ $w, x, y, z \in \mathcal{A}$ และ $x \in \mathcal{A}^*$

(4.14) หมายถึง กำหนดสถานะเริ่มต้นของระบบเปลี่ยนสถานะนำเข้า/นำออก s_0 คือ $(\phi, \langle\rangle, \langle\rangle)$

จากระบบเปลี่ยนสถานะนำเข้า/นำออก $T_{shift(n),\alpha}$ สามารถเขียนให้อยู่ในรูปตารางการเปลี่ยนสถานะ ดังตารางที่ 4.2

ตารางที่ 4.2 ตารางการเปลี่ยนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกของส่วนประกอบเรจิสเตอร์การเลื่อน

State			*State′*		
input	*internal state*	*output*	*input′*	*internal state′*	*output′*
ϕ	$\langle\rangle$	$\langle\rangle$	x_1	$\langle x_1 \rangle$	$\langle\rangle$
x_m	$\langle x_1, ..., x_m \rangle$	$\langle\rangle$	x_{m+1}	$\langle x_1, ..., x_{m+1} \rangle$	$\langle\rangle$
x_n	$\langle x_1, ..., x_n \rangle$	$\langle\rangle$	x_{n+1}	$\langle x_1, ..., x_{n+1} \rangle$	$\langle x_1 \rangle$
x_{n+1}	$\langle x_1, ..., x_{n+1} \rangle$	$\langle x_1 \rangle$	x_{n+2}	$\langle x_2, ..., x_{n+2} \rangle$	$\langle x_2 \rangle$
$1 \le m < n$					

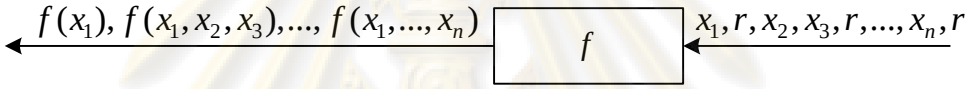
ฟังก์ชันภาวะนามธรรมประวัติ $last(n+1)$ ที่นำเสนอนี้มีความเหมาะสมสำหรับการทวนสอบสมบัตินำเข้า/นำออกด้วยการตรวจสอบแบบจำลอง เนื่องจากสามารถลดปริภูมิสถานะอนันต์ของเครื่องมัวร์แบบบัญญัติเป็นปริภูมิสถานะจำกัดของระบบเปลี่ยนสถานะนำเข้า/นำออก

4.3 ฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบแบบมีคำร้องขอ

ตอนย่อยนี้นำเสนอลักษณะของกลุ่มส่วนประกอบแบบมีคำร้องขอ ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบแบบมีคำร้องขอ นิยามของกลุ่มส่วนประกอบแบบมีคำร้องขอโดยอาศัยฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ที่นำเสนอ และตัวอย่างการแปลงฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบแบบมีคำร้องขอเป็นระบบเปลี่ยนสถานะนำเข้า/นำออกโดยอาศัยฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ที่นำเสนอ

4.3.1 ลักษณะของส่วนประกอบแบบมีคำร้องขอ

จากลักษณะของส่วนประกอบแบบมีคำร้องขอที่แสดงใน 2.2.1 สามารถแสดงได้ดังรูปที่ 4.3



รูปที่ 4.3 ส่วนประกอบแบบมีคำร้องขอ

4.3.2 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบแบบมีคำร้องขอ

ภาคขยายนำออกขึ้นอยู่กับสมาชิกของกระแสนำเข้าที่เป็นข้อมูลชนิดอัพเดทที่มีผลต่อนำออกในอนาคต และชนิดคำร้องขอที่ไม่มีผลต่อนำออกในอนาคต

ฟังก์ชันภาวะนามธรรมสำหรับเครื่องมัวร์ของส่วนประกอบกลุ่มนี้คือ ฟังก์ชัน $filterU$ โดยที่ฟังก์ชัน $filterU$ มีพฤติกรรมดังนิยามใน 2.2.1 ให้ผลลัพธ์คือ กระแสนำเข้าที่ตัดสมาชิกที่เป็นข้อมูลชนิดคำร้องขอออก คงเหลือไว้เฉพาะสมาชิกที่เป็นข้อมูลชนิดอัพเดทเท่านั้น ดังนั้น $filterU(X) = filterU(Y) \Rightarrow \varphi_f(X) = \varphi_f(Y)$ เพราะ $filterU(X) = filterU(Y)$ หมายถึง ข้อมูลชนิดอัพเดทของกระแสนำเข้า X และ ข้อมูลชนิดอัพเดทของกระแสนำเข้า Y มีสมาชิกเหมือนกันตำแหน่งต่อตำแหน่งเมื่อตัดสมาชิกที่เป็นข้อมูลชนิดคำร้องขอออก ทำให้ $\varphi_f(X) = \varphi_f(Y)$ นั่นคือ ฟังก์ชัน $filterU$ มีสมบัตินำออกเข้ากันได้ และ $filterU(X) = filterU(Y) \Rightarrow filterU(X \triangleright x) = filterU(Y \triangleright x)$ เพราะ $filterU(X) = filterU(Y)$ หมายถึง ข้อมูลชนิดอัพเดทของกระแสนำเข้า X และ ข้อมูลชนิดอัพเดทของกระแสนำเข้า Y มีสมาชิกเหมือนกันตำแหน่งต่อตำแหน่งเมื่อตัดสมาชิกที่เป็นข้อมูลชนิดคำร้องขอออก เมื่อเติมสมาชิกเข้าไปด้านหลังของกระแสย่อมทำให้ $filterU(X \triangleright x) = filterU(Y \triangleright x)$ ด้วย นั่นคือ

ฟังก์ชัน *filterU* มีสมบัติการเปลี่ยนสถานะปิด ดังนั้นฟังก์ชัน *filterU* เป็นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบกลุ่มนี้

4.3.3 **นิยามของส่วนประกอบแบบมีคำร้องขอ**

นิยาม 4.3 ส่วนประกอบแบบมีคำร้องขอ

ส่วนประกอบที่จำลองด้วยฟังก์ชันการประมวลผลกระแสที่มีฟังก์ชัน *filterU* เป็นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ เรียกว่า ส่วนประกอบแบบมีคำร้องขอ

4.3.4 **ตัวอย่างการแปลงส่วนประกอบแบบมีคำร้องขอเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก**

ตัวอย่างส่วนประกอบแบบมีคำร้องขอคือ ส่วนประกอบหน่วยความจำ ดังแสดงในตัวอย่าง 2.3 ต่อไปนี้ผู้วิจัยนำเสนอระบบเปลี่ยนสถานะนำเข้า/นำออกที่ได้รับจากฟังก์ชันภาวะนามธรรมประวัติ *filterU*

4.3.4.1 **การแปลงส่วนประกอบหน่วยความจำเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก**

ต่อไปนี้เป็นการแปลงส่วนประกอบหน่วยความจำเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก โดยแปลงส่วนประกอบหน่วยความจำ ผ่านทางเครื่องมัวร์แบบบัญญัติ จากนั้นจึงแปลงเครื่องมัวร์แบบบัญญัติที่ได้รับเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

4.3.4.1.1 **การแปลงส่วนประกอบหน่วยความจำเป็นเครื่องมัวร์แบบบัญญัติ**

ส่วนประกอบหน่วยความจำสามารถแปลงเป็นเครื่องมัวร์แบบบัญญัติได้ดังนี้

$$M_{memory} = (\mathcal{D}^*, (w(\mathcal{D}) \cup \{r\}), \mathcal{D}^*, \delta,\ \varphi_{memory}, \langle\rangle)$$

โดยที่

ฟังก์ชันนำออกเดี่ยว φ_{memory} คือ

$$\varphi_{memory}(\langle\rangle) = \langle\rangle \tag{4.15}$$

$$\varphi_{memory}(X \lhd x) = \langle\rangle \tag{4.16}$$

$$\varphi_{memory}(X \lhd r) = X \tag{4.17}$$

(4.15) หมายถึงไม่มีนำออกเมื่อไม่มีนำเข้า

(4.16) หมายถึงไม่มีนำออกเมื่อได้รับนำเข้าที่ไม่ใช่คำร้องขอ

(4.17) หมายถึงนำออกกระแสที่เก็บไว้เมื่อได้รับคำร้องขอ

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

4.3.4.1.2 การแปลงเครื่องมัวร์แบบบัญญัติเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

เครื่องมัวร์แบบบัญญัติ M_{memory} สามารถแปลงเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก $T_{memory,\alpha}$ ได้ดังนี้

$$T_{memory,\alpha} = ((w(\mathcal{D}) \cup \{r\}), \mathcal{D}^*, Q, S, R, s_0)$$

โดยที่

$$Q = \mathcal{D}^* \tag{4.18}$$

$$S = (\mathcal{D} \cup \{\phi\}) \times \mathcal{D}^* \times \mathcal{D}^* \tag{4.19}$$

$$R = \{(\alpha(X), \alpha(X \rhd x)) \mid X \in \mathcal{D}^*, x \in \mathcal{D}\} \tag{4.20}$$

$$s_0 = (\phi, \mathit{filterU}(\langle\rangle), \varphi_{memory}(\langle\rangle)) = (\phi, \langle\rangle, \langle\rangle) \tag{4.21}$$

เมื่อ

$$\alpha(\langle\rangle) = (\phi, \mathit{filterU}(\langle\rangle), \varphi_{memory}(\langle\rangle))$$

$$\alpha(X \rhd x) = (x, \mathit{filterU}(X \rhd x), \varphi_{memory}(X \rhd x))$$

(4.18) หมายถึง เซตของสถานะภายในคือ เซต $\mathcal{D}^*$ ซึ่งได้รับจากการลดจำนวนสถานะที่ได้รับจากกระแสนำเข้าด้วยฟังก์ชันภาวะนามธรรมประวัติ $\mathit{filterU}$

(4.19) หมายถึง ปริภูมิสถานะ S ที่เกิดจากการรวมเซตข้อมูลนำเข้า สถานะภายใน และเซตข้อมูลนำออกด้วยฟังก์ชันการแปลง α สามารถสร้างสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกภายใต้ความสัมพันธ์การจำลองสองทางกับสถานะของเครื่องมัวร์แบบบัญญัติได้ดังนี้

สร้างสถานะจากสถานะเริ่มต้นที่เป็นกระแสว่าง

$$\langle\rangle \leftrightarrow_H \{(\phi, \langle\rangle, \langle\rangle)\}$$

สร้างสถานะจากสถานะภายในที่นำเข้าปัจจุบันคือ คำร้องขอ

$$\langle X \lhd r\rangle \leftrightarrow_H \{(r, X, X) \mid X \in \mathcal{D}^*\}$$

สร้างสถานะจากสถานะภายในที่นำเข้าปัจจุบันคือ ข้อมูลอัพเดท

$$\langle X \lhd x\rangle \leftrightarrow_H \{(r, X \lhd x, \langle\rangle) \mid x \in \mathcal{D}, X \in \mathcal{D}^*\}$$

(4.20) หมายถึง ความสัมพันธ์การเปลี่ยนสถานะ R แสดงความสัมพันธ์ของสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออก

การเปลี่ยนสถานะจากสถานะเริ่มต้น เมื่อได้รับนำเข้าคือ คำร้องขอ

$$(\phi, \langle\rangle, \langle\rangle) \rightarrow (r, \langle\rangle, \langle\rangle)$$

การเปลี่ยนสถานะจากสถานะเริ่มต้น เมื่อได้รับนำเข้าคือ คำร้องขอ

$$(\phi, \langle\rangle, \langle\rangle) \rightarrow (x, \langle x\rangle, \langle\rangle)$$

การเปลี่ยนสถานะจากสถานะ เมื่อนำเข้าก่อนหน้าคือ คำร้องขอ และนำเข้าปัจจุบันคือ คำร้องขอ

$$(r, X, X) \rightarrow (r, X, X)$$

การเปลี่ยนสถานะจากสถานะ เมื่อนำเข้าก่อนหน้าคือ คำร้องขอ และนำเข้าปัจจุบันคือ ข้อมูลอัพเดต

$$(r, X, X) \rightarrow (x, X \rhd x, \langle\rangle)$$

การเปลี่ยนสถานะจากสถานะ เมื่อนำเข้าก่อนหน้าคือ ข้อมูลอัพเดตและนำเข้าปัจจุบันคือ คำร้องขอ

$$(x, X \rhd x, \langle\rangle) \rightarrow (r, X \rhd x, X \rhd x)$$

การเปลี่ยนสถานะจากสถานะ เมื่อนำเข้าก่อนหน้าคือ คำร้องขอ และนำเข้าปัจจุบันคือ ข้อมูลอัพเดต

$$(x, X \rhd x, \langle\rangle) \rightarrow (y, X \rhd x \rhd y, \langle\rangle)$$

เมื่อ $x, y \in \mathcal{D}$ และ $X \in \mathcal{D}^*$

(4.21) หมายถึง กำหนดสถานะเริ่มต้นของระบบเปลี่ยนสถานะนำเข้า/นำออก s_0 คือ $(\phi, \langle\rangle, \langle\rangle)$

จากระบบเปลี่ยนสถานะนำเข้า/นำออก $T_{memory,\alpha}$ สามารถเขียนให้อยู่ในรูปตารางการเปลี่ยนสถานะ ดังตารางที่ 4.3

ตารางที่ 4.3 ตารางการเปลี่ยนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกของส่วนประกอบหน่วยความจำ

State			*State′*		
input	*internal state*	*output*	*input′*	*internal state′*	*output′*
ϕ	$\langle\rangle$	$\langle\rangle$	r	$\langle\rangle$	$\langle\rangle$
ϕ	$\langle\rangle$	$\langle\rangle$	x	$\langle x\rangle$	$\langle\rangle$
r	X	X	r	X	X
r	X	X	x	$X \rhd x$	$\langle\rangle$
x	$X \rhd x$	$\langle\rangle$	r	$X \rhd x$	$X \rhd x$
x	$X \rhd x$	$\langle\rangle$	y	$X \rhd x \rhd y$	$\langle\rangle$
$x, y \in U, X \in U^*$					

ฟังก์ชันภาวะนามธรรมประวัติ *filterU* สามารถทวนสอบสมบัตินำเข้า/นำออกด้วยการตรวจสอบแบบจำลองที่จำกัดความยาวของกระแสนำเข้าของส่วนประกอบเท่านั้น แต่ไม่เหมาะสมสำหรับการทวนสอบสมบัตินำเข้า/นำออกของส่วนประกอบทั่วไป เนื่องจากไม่สามารถรับรองได้ว่า

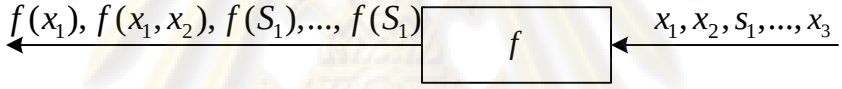
ระบบเปลี่ยนสถานะนำเข้า/นำออกที่ได้รับมีปริภูมิสถานะจำกัดซึ่งถ้าเป็นระบบที่มีปริภูมิสถานะอนันต์จะทำให้เกิดปัญหาสถานะเพิ่มขึ้นอย่างรวดเร็วในการทวนสอบสมบัตินำเข้า/นำออกด้วยการตรวจสอบแบบจำลอง

4.4 ฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบคงที่

ตอนย่อยนี้นำเสนอลักษณะของกลุ่มส่วนประกอบคงที่ ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบคงที่ นิยามของกลุ่มส่วนประกอบคงที่โดยอาศัยฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ที่นำเสนอ และตัวอย่างการแปลงฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบคงที่เป็นระบบเปลี่ยนสถานะนำเข้า/นำออกโดยอาศัยฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ที่นำเสนอ

4.4.1 ลักษณะของส่วนประกอบคงที่

จากลักษณะของส่วนประกอบคงที่ที่แสดงใน 2.2.1 สามารถแสดงได้ดังรูปที่ 4.4



รูปที่ 4.4 ส่วนประกอบคงที่

เมื่อ $\langle x_1 \rangle, \langle x_1, x_2 \rangle \in R$ และ $\langle x_1, x_2, s_1 \rangle, \langle x_1, x_2, s_1, x_3 \rangle \in S_1$

4.4.2 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบคงที่

ภาคขยายนำออกแบ่งเป็นภาคขยายนำออกของกระแสนำเข้าปกติคือ นำออกที่เกิดจากสมาชิกตัวสุดท้ายของกระแส และภาคขยายของการขยายกระแสนำเข้าคงที่คือ นำออกที่กำหนดสำหรับกระแสนำเข้าคงที่แต่ละชนิด

ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์แบบบัญญัติคือ $break : \mathcal{A}^* \rightarrow R \cup \{s_i \mid i \in I\}$ โดยที่ฟังก์ชัน $break$ มีพฤติกรรมดังนิยามใน 2.2.1 ให้ผลลัพธ์คือ แบ่งกระแสนำเข้าเป็นกระแสนำเข้าปกติและกระแสนำเข้าคงที่ ดังนั้น $break(X) = break(Y) \Rightarrow \varphi_f(X) = \varphi_f(Y)$ เพราะ $break(X) = break(Y)$ หมายถึง $X = Y$ หรือ $X, Y \in S_i (i \in I)$ ทำให้ $\varphi_f(X) = \varphi_f(Y)$ นั่นคือ ฟังก์ชัน $break$ มีสมบัตินำออกเข้ากันได้ และ $break(X) = break(Y) \Rightarrow break(X \triangleright x) = break(Y \triangleright x)$ เพราะ $break(X) = break(Y)$ หมายถึง $X = Y$ หรือ $X, Y \in S_i (i \in I)$ เมื่อเติมสมาชิกเข้าไปด้านหลังของกระแสย่อมทำให้ $break(X \triangleright x) = break(Y \triangleright x)$ ด้วย นั่นคือ ฟังก์ชัน $break$ มีสมบัติการเปลี่ยนสถานะปิด ดังนั้นฟังก์ชัน $break$ เป็นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบกลุ่มนี้

4.4.3 **นิยามของส่วนประกอบที่คงที่**

นิยาม 4.4 ส่วนประกอบที่คงที่

ส่วนประกอบที่จำลองด้วยฟังก์ชันการประมวลผลกระแสที่มีฟังก์ชัน *break* เป็นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ เรียกว่า ส่วนประกอบคงที่

4.4.4 **ตัวอย่างการแปลงส่วนประกอบคงที่เป็นระบบเปลี่ยนสถานะนำเข้า/นำออก**

ตัวอย่างส่วนประกอบคงที่คือ ส่วนประกอบตัวกรองกำหนดเอง ดังแสดงในตัวอย่าง 2.4 ต่อไปนี้ผู้วิจัยนำเสนอระบบเปลี่ยนสถานะนำเข้า/นำออกที่ได้รับจากฟังก์ชันภาวะนามธรรมประวัติ *break*

4.4.4.1 **การแปลงส่วนประกอบตัวกรองกำหนดเองเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก**

ต่อไปนี้เป็นการแปลงส่วนประกอบตัวกรองกำหนดเองเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก โดยแปลงส่วนประกอบตัวกรองกำหนดเองผ่านทางเครื่องมัวร์แบบบัญญัติ จากนั้นจึงแปลงเครื่องมัวร์แบบบัญญัติที่ได้รับเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

4.4.4.1.1 **การแปลงส่วนประกอบตัวกรองกำหนดเองเป็นเครื่องมัวร์แบบบัญญัติ**

ส่วนประกอบตัวกรองกำหนดเองสามารถแปลงเป็นเครื่องมัวร์แบบบัญญัติได้ดังนี้

$$M_{filterinit} = (\mathcal{A}^*, \mathcal{A}, \mathcal{A}^*, \delta, \varphi_{filterinit}, \langle\rangle)$$

โดยที่

ฟังก์ชันนำออกเดี่ยว $\varphi_{filterinit}(X)$ คือ

$$\varphi_{filterinit}(\langle\rangle) = \langle\rangle \tag{4.22}$$

$$\varphi_{filterinit}(\langle x\rangle) = \langle\rangle \tag{4.23}$$

$$\varphi_{filterinit}(x \lhd X \lhd x) = \langle x\rangle \tag{4.24}$$

(4.22) หมายถึง ไม่มีนำออกเมื่อไม่มีนำเข้า

(4.23) หมายถึง ไม่มีนำออกเมื่อมีนำเข้าเป็นกระแสเดี่ยว

(4.24) หมายถึง นำออกคือ สมาชิกของกระแสนำเข้าที่เหมือนกับสมาชิกเริ่มต้น

4.4.4.1.2 **การแปลงเครื่องมัวร์แบบบัญญัติเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก**

เครื่องมัวร์แบบบัญญัติ $M_{filterinit}$ สามารถแปลงเป็นระบบเปลี่ยนสถานะนำเข้า/ นำออก $T_{filterinit,\alpha}$ ได้ดังนี้

$$T_{filterinit,\alpha} = (\mathcal{A}, \mathcal{A}^*, Q, S, R, s_0)$$

โดยที่

$$Q = \mathcal{A}^* \tag{4.25}$$

$$S = (\mathcal{A} \cup \{\phi\}) \times \mathcal{A}^* \times \mathcal{A}^* \tag{4.26}$$

$$R = \{(\alpha(X), \alpha(X \rhd x)) \mid X \in \mathcal{A}^*, x \in \mathcal{A}\} \tag{4.27}$$

$$s_0 = (\phi, break(\langle\rangle), \varphi_{filterinit}(\langle\rangle)) \tag{4.28}$$

เมื่อ

$$\alpha(\langle\rangle) = (\phi, break(\langle\rangle), \varphi_{filterinit}(\langle\rangle))$$

$$\alpha(X \rhd x) = (x, break(X \rhd x), \varphi_{filterinit}(X \rhd x))$$

(4.25) หมายถึง เซตของสถานะภายในคือ กระแส $\mathcal{A}^*$ ซึ่งได้รับจากการลดจำนวนสถานะที่ได้รับจากกระแสนำเข้าด้วยฟังก์ชันภาวะนามธรรมประวัติ $break$

(4.26) หมายถึง ปริภูมิสถานะ S ที่เกิดจากการรวมเซตข้อมูลนำเข้า สถานะภายใน และเซตข้อมูลนำออก ด้วยฟังก์ชันการแปลง α สามารถสร้างสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกภายใต้ความสัมพันธ์การจำลองสองทางกับสถานะของเครื่องมัวร์แบบบัญญัติได้ดังนี้

สร้างสถานะจากสถานะเริ่มต้นที่เป็นกระแสว่าง

$$\langle\rangle \leftrightarrow_H \{(\phi, \langle\rangle, \langle\rangle)\}$$

สร้างสถานะจากกระแสที่สมาชิกตัวถัดไปไม่ตรงกับตัวแรก

$$\langle x \lhd X \lhd y\rangle \leftrightarrow_H \{(y, \langle x\rangle, \langle\rangle) \mid x, y \in \mathcal{A}, X \in \mathcal{A}^*\}$$

สร้างสถานะจากกระแสที่สมาชิกตัวถัดไปไม่ตรงกับตัวแรก

$$\langle x \lhd X \lhd x\rangle \leftrightarrow_H \{(x, \langle x\rangle, \langle x\rangle) \mid x \in \mathcal{A}, X \in \mathcal{A}^*\}$$

(4.27) หมายถึง ความสัมพันธ์การเปลี่ยนสถานะ R แสดงความสัมพันธ์ของสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออก

การเปลี่ยนสถานะจากสถานะเริ่มต้น

$$(\phi, \langle\rangle, \langle\rangle) \rightarrow (r, \langle\rangle, \langle\rangle)$$

การเปลี่ยนสถานะจากสถานะ เมื่อนำเข้าก่อนหน้าไม่ตรงกับค่าเริ่มต้น และนำเข้าปัจจุบันไม่ตรงกับค่าเริ่มต้น

$$(y, \langle x\rangle, \langle\rangle) \rightarrow (y, \langle x\rangle, \langle\rangle)$$

การเปลี่ยนสถานะจากสถานะ เมื่อนำเข้าก่อนหน้าไม่ตรงกับค่าเริ่มต้น และนำเข้าปัจจุบันตรงกับค่าเริ่มต้น

$$(y, \langle x\rangle, \langle\rangle) \rightarrow (x, \langle x\rangle, \langle x\rangle)$$

การเปลี่ยนสถานะจากสถานะ เมื่อนำเข้าก่อนหน้าตรงกับค่าเริ่มต้น และนำเข้าปัจจุบันไม่ตรงกับค่าเริ่มต้น

$$(x, \langle x\rangle, \langle x\rangle) \rightarrow (y, \langle x\rangle, \langle\rangle)$$

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

การเปลี่ยนสถานะจากสถานะ เมื่อนำเข้าก่อนหน้าตรงกับค่าเริ่มต้น และนำเข้าปัจจุบันตรงกับค่าเริ่มต้น

$$(x,\langle x\rangle,\langle x\rangle) \rightarrow (x,\langle x\rangle,\langle x\rangle)$$

เมื่อ $x, y \in \mathcal{A}$

(4.28) หมายถึง กำหนดสถานะเริ่มต้นของระบบเปลี่ยนสถานะนำเข้า/นำออก s_0 คือ $(\phi,\langle\rangle,\langle\rangle)$

จากระบบเปลี่ยนสถานะนำเข้า/นำออก $T_{filterinit,\alpha}$ สามารถเขียนให้อยู่ในรูปตารางการเปลี่ยนสถานะ ดังตารางที่ 4.4

ตารางที่ 4.4 ตารางการเปลี่ยนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกของส่วนประกอบตัวกรองกำหนดเอง

State			*State′*		
input	*internal state*	*output*	*input′*	*internal state′*	*output′*
ϕ	$\langle\rangle$	$\langle\rangle$	x	$\langle x\rangle$	$\langle\rangle$
y	$\langle x\rangle$	$\langle\rangle$	y	$\langle x\rangle$	$\langle\rangle$
y	$\langle x\rangle$	$\langle\rangle$	x	$\langle x\rangle$	$\langle x\rangle$
x	$\langle x\rangle$	$\langle x\rangle$	y	$\langle x\rangle$	$\langle\rangle$
x	$\langle x\rangle$	$\langle x\rangle$	x	$\langle x\rangle$	$\langle x\rangle$
$x, y \in \mathcal{A}, x \neq y$					

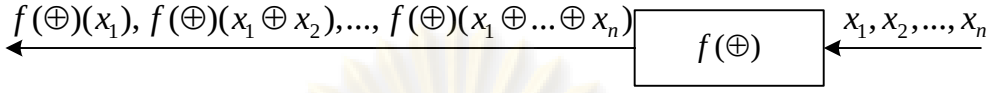
ฟังก์ชันภาวะนามธรรมประวัติ *break* ที่นำเสนอไม่เหมาะสมสำหรับการทวนสอบสมบัตินำเข้า/นำออกด้วยการตรวจสอบแบบจำลอง เนื่องจากไม่สามารถรับรองได้ว่าระบบเปลี่ยนสถานะนำเข้า/นำออกที่ได้รับมีปริภูมิสถานะจำกัดซึ่งถ้าเป็นระบบที่มีปริภูมิสถานะอนันต์จะทำให้เกิดปัญหาสถานะเพิ่มขึ้นอย่างรวดเร็วในการทวนสอบสมบัตินำเข้า/นำออกด้วยการตรวจสอบแบบจำลอง

4.5 ฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบสแกน

ตอนย่อยนี้นำเสนอลักษณะของกลุ่มส่วนประกอบสแกน ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบสแกน นิยามของกลุ่มส่วนประกอบสแกนโดยอาศัยฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ที่นำเสนอ และตัวอย่างการแปลงฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบสแกนเป็นระบบเปลี่ยนสถานะนำเข้า/นำออกโดยอาศัยฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ที่นำเสนอ

4.5.1 **ลักษณะของส่วนประกอบสแกน**

จากลักษณะของส่วนประกอบสแกนที่แสดงใน 2.2.1 สามารถแสดงได้ดังรูปที่ 4.5



รูปที่ 4.5 ส่วนประกอบสแกน

4.5.2 **ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบสแกน**

ภาคขยายนำออกของส่วนประกอบกลุ่มนี้คือ $\varphi_{scan(s,\oplus)}(X) = s$ นั่นคือนำออกขึ้นอยู่กับสถานะที่สะสมกระแสนำเข้า

ฟังก์ชันภาวะนามธรรมสำหรับเครื่องมัวร์ของส่วนประกอบกลุ่มนี้คือ ฟังก์ชัน $reduce(s,\oplus)$ โดยที่ฟังก์ชัน $reduce(s,\oplus)$ มีพฤติกรรมดังนิยามใน 2.2.1 ให้ผลลัพธ์คือ ผลลัพธ์ของการสะสมกระแสนำเข้า ดังนั้น $reduce\ (X) = reduce(Y) \Rightarrow \varphi_f(X) = \varphi_f(Y)$ เพราะ $reduce(X) = reduce(Y)$ หมายถึง การสะสมกระแสนำเข้า X เท่ากับ การสะสมกระแสนำเข้า Y ทำให้ $\varphi_f(X) = \varphi_f(Y)$ นั่นคือ ฟังก์ชัน $reduce(s,\oplus)$ $reduce$ มีสมบัตินำออกเข้ากันได้ และ $reduce(X) = reduce(Y) \Rightarrow reduce\ (X \triangleright x) = reduce(Y \triangleright x)$ เนื่องจาก เพราะ $reduce\ (X) = reduce(Y)$ หมายถึง การสะสมกระแสนำเข้า X เท่ากับ การสะสมกระแสนำเข้า Y ทำให้ $reduce(X \triangleright x) = reduce(Y \triangleright x)$ นั่นคือ ฟังก์ชัน $reduce(s,\oplus)$ มีสมบัติการเปลี่ยนสถานะปิด ดังนั้นฟังก์ชัน $reduce(s,\oplus)$ เป็นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบกลุ่มนี้

4.5.3 **นิยามของส่วนประกอบสแกน**

นิยาม 4.5 ส่วนประกอบสแกน

ส่วนประกอบที่จำลองด้วยฟังก์ชันการประมวลผลกระแสที่มีฟังก์ชัน $reduce(s,\oplus)$ เป็นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ เรียกว่า ส่วนประกอบสแกน

4.5.4 **ตัวอย่างการแปลงส่วนประกอบสแกนเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก**

ตัวอย่างส่วนประกอบสแกนคือ ส่วนประกอบผลบวก ดังแสดงในตัวอย่าง 2.5 ต่อไปนี้ผู้วิจัยนำเสนอระบบเปลี่ยนสถานะนำเข้า/นำออกที่ได้รับจากฟังก์ชันภาวะนามธรรมประวัติ $reduce(s,\oplus)$

4.5.4.1 การแปลงส่วนประกอบผลบวกเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

ต่อไปนี้เป็นการแปลงส่วนประกอบผลบวกเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก โดยแปลงส่วนประกอบผลบวกผ่านทางเครื่องมัวร์แบบบัญญัติ จากนั้นจึงแปลงเครื่องมัวร์แบบบัญญัติที่ได้รับเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

4.5.4.1.1 การแปลงส่วนประกอบผลบวกเป็นเครื่องมัวร์แบบบัญญัติ

ส่วนประกอบผลบวกสามารถแปลงเป็นเครื่องมัวร์แบบบัญญัติได้ดังนี้

$$M_{summation} = (\mathbb{N}^*, \mathbb{N}, \mathbb{N}^*, \delta, \varphi_{summation}, \langle\rangle)$$

โดยที่

ฟังก์ชันนำออกเดี่ยว $\varphi_{summation}(X) = \langle reduce'(s,+)(X)\rangle$ คือ ผลบวกสะสมของกระแสนำเข้า เมื่อ $reduce'(s,+)$ นิยามดังนี้

$$reduce' : \mathbb{N} \times [\mathbb{N} \times \mathbb{N} \to \mathbb{N}] \to [\mathbb{N}^* \to \mathbb{N}]$$

$$reduce'(s,+)(\langle\rangle) = s \tag{4.29}$$

$$reduce'(s,+)(x \lhd X) = reduce'(s+x,+)(X) \tag{4.30}$$

เมื่อ $s, x \in \mathbb{N}$ $X \in \mathbb{N}^*$ และ กำหนดสถานะเริ่มต้น $s = 0$

(4.29) หมายถึง นำเข้าว่างก่อกำเนิดนำออกที่เป็นค่าที่เก็บไว้ในสถานะภายใน s

(4.30) หมายถึง บวกสมาชิกแต่ละตัวของกระแสนำเข้าสะสมไว้ในตัวแปรภายใน s

4.5.4.1.2 การแปลงเครื่องมัวร์แบบบัญญัติเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

เครื่องมัวร์แบบบัญญัติ $M_{summation}$ สามารถแปลงเป็นระบบเปลี่ยนสถานะนำเข้า/ นำออก $T_{summation,\alpha}$ ได้ดังนี้

$$T_{summation,\alpha} = (\mathbb{N}, \mathbb{N}^*, Q, S, R, s_0)$$

โดยที่

$$Q = \mathbb{N} \tag{4.31}$$

$$S = (\mathbb{N} \cup \{\phi\}) \times \mathbb{N} \times \mathbb{N}^* \tag{4.32}$$

$$R = \{(\alpha(X), \alpha(X \rhd x)) \mid X \in \mathbb{N}^*, x \in \mathbb{N}\} \tag{4.33}$$

$$s_0 = (\phi, reduce'(s,+)(\langle\rangle), \varphi_{summation}(\langle\rangle)) = (\phi, s, \langle s\rangle) \tag{4.34}$$

เมื่อ

$$\alpha(\langle\rangle) = (\phi, reduce'(s,+)(\langle\rangle), \varphi_{summation}(\langle\rangle))$$

$$\alpha(X \rhd x) = (x, reduce'(s,+)(X \rhd x), \varphi_{summation}(X \rhd x))$$

(4.31) หมายถึง เซตของสถานะภายในคือ กระแส $\mathbb{N}$ ซึ่งได้รับจากการลดจำนวนสถานะที่ได้รับจากกระแสนำเข้าด้วยฟังก์ชันภาวะนามธรรมประวัติ $reduce'(s,+)$

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

(4.32) หมายถึง ปริภูมิสถานะ S ที่เกิดจากการรวมเซตข้อมูลนำเข้า สถานะภายใน และเซตข้อมูลนำออก ด้วยฟังก์ชันการแปลง α สามารถสร้างสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกภายใต้ความสัมพันธ์การจำลองสองทางกับสถานะของเครื่องมัวร์แบบบัญญัติได้ดังนี้

สร้างสถานะจากสถานะเริ่มต้นที่เป็นกระแสว่าง

$$\langle\rangle \leftrightarrow_H \{(\phi, s, \langle s\rangle)\}$$

สร้างสถานะจากสถานะภายในไม่เป็นกระแสว่าง

$$X \triangleright x \leftrightarrow_H \{(x, reduce'(s,+)(X \triangleright x), \langle reduce'(s,+)(X \triangleright x)\rangle) \mid s, x \in \mathbb{N}, X \in \mathbb{N}^*\}$$

(4.33) หมายถึง ความสัมพันธ์การเปลี่ยนสถานะ R แสดงความสัมพันธ์ของสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออก

การเปลี่ยนสถานะจากสถานะเริ่มต้น เมื่อได้รับนำเข้า $x \in \mathbb{N}$

$$(\phi, s, \langle s\rangle) \rightarrow (x, s+x, \langle s+x\rangle)$$

การเปลี่ยนสถานะจากสถานะที่ไม่ใช่สถานะเริ่มต้น และได้รับนำเข้า $y \in \mathbb{N}$

$$(x, q, \langle q\rangle)) \rightarrow (y, q+y, \langle q+y\rangle)$$

เมื่อ $q = reduce'(s,+)(X \triangleright x),\ s, x, y \in \mathbb{N}$ และ $X \in \mathbb{N}^*$

(4.34) หมายถึง กำหนดสถานะเริ่มต้นของระบบเปลี่ยนสถานะนำเข้า/นำออก s_0 คือ $(\phi, s, \langle s\rangle)$

จากระบบเปลี่ยนสถานะนำเข้า/นำออก $T_{summation,\alpha}$ สามารถเขียนให้อยู่ในรูปตารางการเปลี่ยนสถานะ ดังตารางที่ 4.5

ฟังก์ชันภาวะนามธรรมประวัติ $reduce(s, \oplus)$ สามารถทวนสอบสมบัตินำเข้า/นำออกด้วยการตรวจสอบแบบจำลองที่จำกัดขนาดของข้อมูลนำเข้าของส่วนประกอบเท่านั้น แต่ไม่เหมาะสมสำหรับการทวนสอบสมบัตินำเข้า/นำออกของส่วนประกอบทั่วไป เนื่องจากไม่สามารถรับรองได้ว่าระบบเปลี่ยนสถานะนำเข้า/นำออกที่ได้รับมีขอบเขตบนของการสะสมกระแสนำเข้า

ตารางที่ 4.5 ตารางการเปลี่ยนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกของส่วนประกอบผลรวม

State			*State′*		
input	*internal state*	*output*	*input′*	*internal state′*	*output′*
ϕ	s	$\langle s\rangle$	x	$s+x$	$\langle s+x\rangle$
x	q	$\langle q\rangle$	y	$q+y$	$\langle q+y\rangle$
$q = reduce'(s,+)(X \triangleright x), s, x, y \in \mathbb{N}, X \in \mathbb{N}^*$					

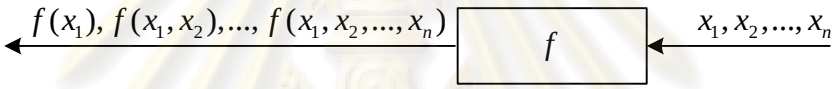
ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

4.6 ฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบที่ต้องการประวัติ

ตอนย่อยนี้นำเสนอลักษณะของกลุ่มส่วนประกอบที่ต้องการประวัติ ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบที่ต้องการประวัติ นิยามของกลุ่มส่วนประกอบที่ต้องการประวัติโดยอาศัยฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ที่นำเสนอ และตัวอย่างการแปลงฟังก์ชันการประมวลผลกระแสของกลุ่มส่วนประกอบที่ต้องการประวัติเป็นระบบเปลี่ยนสถานะนำเข้า/นำออกโดยอาศัยฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ที่นำเสนอ

4.6.1 ลักษณะของส่วนประกอบที่ต้องการประวัติ

จากลักษณะของส่วนประกอบที่ต้องการประวัติที่แสดงใน 2.2.1 สามารถแสดงได้ดังรูปที่ 4.6



รูปที่ 4.6 ส่วนประกอบที่ต้องการประวัติ

4.6.2 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบที่ต้องการประวัติ

ภาคขยายนำออกของฟังก์ชัน $f : \mathcal{A}^* \to \mathcal{B}^*$ คือ

$$\varphi_f(X \triangleright x) = f(X \triangleright x) \ominus f(X)$$

นั่นคือภาคขยายนำออกขึ้นอยู่กับประวัตินำเข้าทั้งหมด

ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบกลุ่มนี้คือ ฟังก์ชัน $identity : \mathcal{A}^* \to \mathcal{A}^*$ โดยที่ $identity(X) = X$ เมื่อ $X \in \mathcal{A}^*$

ฟังก์ชัน $identity$ ให้ผลลัพธ์คือ กระแสนำเข้าเดิม ดังนั้น $identity(X) = identity(Y)$ $\Rightarrow \varphi_f(X) = \varphi_f(Y)$ เพราะ $identity(X) = identity(Y)$ หมายถึง $X = Y$ ทำให้ $\varphi_f(X) = \varphi_f(Y)$ นั่นคือ ฟังก์ชัน $identity$ มีสมบัตินำออกเข้ากันได้ และ $identity(X) = identity(Y)$ $\Rightarrow identity(X \triangleright x) = identity(Y \triangleright x)$ เนื่องจาก เพราะ $identity(X) = identity(Y)$ หมายถึง $X = Y$ ทำให้ $identity(X \triangleright x) = identity(Y \triangleright x)$ นั่นคือ ฟังก์ชัน $identity$ มีสมบัติการเปลี่ยนสถานะปิด ดังนั้นฟังก์ชัน $identity$ เป็นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบกลุ่มนี้

4.6.3 นิยามของส่วนประกอบที่ต้องการประวัติ

นิยาม 4.8 ส่วนประกอบที่ต้องการประวัติ

ส่วนประกอบที่จำลองด้วยฟังก์ชันการประมวลผลกระแสที่มีฟังก์ชัน *identity* เป็นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ เรียกว่า ส่วนประกอบที่ต้องการประวัติ

4.6.4 ตัวอย่างการแปลงส่วนประกอบที่ต้องการประวัติเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

ตัวอย่างส่วนประกอบที่ต้องการประวัติคือ ส่วนประกอบตำแหน่ง ดังแสดงในตัวอย่าง 2.6 ต่อไปนี้ผู้วิจัยนำเสนอระบบเปลี่ยนสถานะนำเข้า/นำออกที่ได้รับจากฟังก์ชันภาวะนามธรรมประวัติ *identity*

4.6.4.1 การแปลงส่วนประกอบตำแหน่งเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

ต่อไปนี้เป็นการแปลงส่วนประกอบตำแหน่งเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก โดยแปลงส่วนประกอบตำแหน่งผ่านทางเครื่องมัวร์แบบบัญญัติ จากนั้นจึงแปลงเครื่องมัวร์แบบบัญญัติที่ได้รับเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

4.6.4.1.1 การแปลงส่วนประกอบตำแหน่งเป็นเครื่องมัวร์แบบบัญญัติ

ส่วนประกอบตำแหน่งสามารถแปลงเป็นเครื่องมัวร์แบบบัญญัติได้ดังนี้

$$M_{positions} = (\mathbb{N}^*, \mathbb{N}, \mathbb{N}^*, \delta, \varphi_{position}, \langle\rangle)$$

โดยที่

ฟังก์ชันนำออกเดี่ยว $\varphi_{positions}(X)$ คือ

$$\varphi_{positions}(\langle\rangle) = \langle\rangle \tag{4.35}$$

$$\varphi_{positions}(X \lhd x) = lookup(X, x) \tag{4.36}$$

(4.35) หมายถึงไม่มีนำออกเมื่อไม่มีนำเข้า

(4.36) หมายถึงนำออกคือ สมาชิกของกระแสก่อนหน้าที่มีตำแหน่งตรงกับค่าของสมาชิกปัจจุบัน

4.6.4.1.2 การแปลงเครื่องมัวร์แบบบัญญัติเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

เครื่องมัวร์แบบบัญญัติ $M_{positions}$ สามารถแปลงเป็นระบบเปลี่ยนสถานะนำเข้า/ นำออก $T_{positions,\alpha}$ ได้ดังนี้

$$T_{positions,\alpha} = (\mathbb{N}, \mathbb{N}^*, Q, S, R, s_0)$$

โดยที่

$$Q = \mathbb{N}^* \tag{4.37}$$

$$S = (\mathbb{N} \cup \{\phi\}) \times \mathbb{N}^* \times \mathbb{N}^* \tag{4.38}$$

$$R = \{(\alpha(X), \alpha(X \rhd x)) \mid X \in \mathbb{N}^*, x \in \mathbb{N}\} \tag{4.39}$$

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

$$s_0 = (\phi, identity(\langle\rangle), \varphi_{positions}(\langle\rangle)) = (\phi, \langle\rangle, \langle\rangle) \quad (4.40)$$

เมื่อ $$\alpha(\langle\rangle) = (\phi, identity(\langle\rangle), \varphi_{positions}(\langle\rangle))$$

$$\alpha(X \triangleright x) = (x, identity(X \triangleright x), \varphi_{positions}(X \triangleright x))$$

(4.37) หมายถึง เซตของสถานะภายในคือ กระแส $\mathbb{N}^*$ ซึ่งได้รับจากฟังก์ชันภาวะนามธรรมประวัติ *identity*

(4.38) หมายถึง ปริภูมิสถานะ S ที่เกิดจากการรวมเซตข้อมูลนำเข้า สถานะภายใน และเซตข้อมูลนำออก ด้วยฟังก์ชันการแปลง α สามารถสร้างสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกภายใต้ความสัมพันธ์การจำลองสองทางกับสถานะของเครื่องมัวร์แบบบัญญัติได้ดังนี้

สร้างสถานะจากสถานะเริ่มต้นที่เป็นกระแสว่าง

$$\langle\rangle \leftrightarrow_H \{(\phi, \langle\rangle, \langle\rangle)\}$$

สร้างสถานะจากสถานะภายในที่ไม่เป็นกระแสว่าง

$$X \triangleright x \leftrightarrow_H \{(x, X \triangleright x, \langle lookup(X, x)\rangle) \mid x \in \mathbb{N}, X \in \mathbb{N}^*\}$$

(4.39) หมายถึง ความสัมพันธ์การเปลี่ยนสถานะ R แสดงความสัมพันธ์ของสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออก

การเปลี่ยนสถานะจากสถานะเริ่มต้น

$$(\phi, \langle\rangle, \langle\rangle) \rightarrow (x, \langle x\rangle, \langle\rangle)$$

การเปลี่ยนสถานะจากสถานะที่ไม่ใช่สถานะเริ่มต้น

$$(x, X \triangleleft x, lookup(X, x)) \rightarrow (y, X \triangleleft x \triangleleft y, lookup(X \triangleleft x, y))$$

เมื่อ $x, y \in \mathcal{A}$ และ $X \in \mathcal{A}^*$

(4.40) หมายถึง กำหนดสถานะเริ่มต้นของระบบเปลี่ยนสถานะนำเข้า/นำออก s_0 คือ $(\phi, \langle\rangle, \langle\rangle)$

จากระบบเปลี่ยนสถานะนำเข้า/นำออก $T_{positions,\alpha}$ สามารถเขียนให้อยู่ในรูปตารางการเปลี่ยนสถานะ ดังตารางที่ 4.6

ตารางที่ 4.6 ตารางการเปลี่ยนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกของส่วนประกอบตำแหน่ง

State			*State′*		
input	*internal state*	*output*	*input′*	*internal state′*	*output′*
ϕ	$\langle\rangle$	$\langle\rangle$	x	$\langle x\rangle$	$\langle\rangle$
x	$X \triangleleft x$	$lookup(X, x)$	y	$X \triangleleft x \triangleleft y$	$lookup(X \triangleleft x, y)$
$x, y \in \mathbb{N}, X \in \mathbb{N}^*$					

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ฟังก์ชันภาวะนามธรรมประวัติ *identity* สามารถทวนสอบสมบัตินำเข้า/นำออกด้วยการตรวจสอบแบบจำลองที่จำกัดความยาวของกระแสนำเข้าของส่วนประกอบเท่านั้น แต่ไม่เหมาะสมสำหรับการทวนสอบสมบัตินำเข้า/นำออกของส่วนประกอบทั่วไป เนื่องจากไม่สามารถรับรองได้ว่าระบบเปลี่ยนสถานะนำเข้า/นำออกที่ได้รับมีปริภูมิสถานะจำกัดซึ่งถ้าเป็นระบบที่มีปริภูมิสถานะอนันต์จะทำให้เกิดปัญหาสถานะเพิ่มขึ้นอย่างรวดเร็วในการทวนสอบสมบัตินำเข้า/นำออกด้วยการตรวจสอบแบบจำลอง

4.7 สรุป

ในบทนี้นำเสนอฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ 6 ฟังก์ชัน โดยทั้ง 6 ฟังก์ชันสามารถแบ่งเป็น 3 กลุ่มที่แตกต่างกันตามความสามารถในการลดจำนวนสถานะในระบบเปลี่ยนสถานะนำเข้า/นำออกเพื่อที่จะทวนสอบสมบัตินำเข้า/นำออกด้วยการตรวจสอบแบบจำลอง

กลุ่มที่หนึ่งประกอบด้วย 2 ฟังก์ชันคือ ฟังก์ชัน *current* และ ฟังก์ชัน $last(n+1)$ ซึ่งสามารถลดปริภูมิสถานะอนันต์เป็นปริภูมิสถานะจำกัดที่มีความเหมาะสมที่จะทวนสอบสมบัตินำเข้า/นำออกด้วยการตรวจสอบแบบจำลอง

กลุ่มที่สองประกอบด้วย 3 ฟังก์ชันคือ ฟังก์ชัน *filterU* ฟังก์ชัน *break* และ ฟังก์ชัน $reduce(s,\oplus)$ ซึ่งสามารถจำนวนของสถานะในระบบเปลี่ยนสถานนะนำเข้า/นำออก แต่ไม่สามารถรับรองว่าสามารถลดปริภูมิสถานะอนันต์เป็นปริภูมิสถานะจำกัดได้ จึงไม่มีความเหมาะสมที่จะทวนสอบสมบัตินำเข้า/นำออกด้วยการตรวจสอบแบบจำลอง

กลุ่มที่สามประกอบด้วย 1 ฟังก์ชันคือ ฟังก์ชัน *identity* ซึ่งไม่สามารถลดจำนวนสถานะระบบเปลี่ยนสถานะนำเข้า/นำออก จึงไม่มีความเหมาะสมที่จะทวนสอบสมบัตินำเข้า/นำออกด้วยการตรวจสอบแบบจำลอง

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 5
ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบ

บทนี้นำเสนอวิธีการกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบเพื่อใช้ในการแปลงแบบจำลองประกอบเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก เพื่อที่จะทวนสอบสมบัตินำเข้า/นำออกที่คาดหมายของแบบจำลองประกอบด้วยการตรวจสอบแบบจำลอง โดยเนื้อหามีดังนี้ ตอนที่ 5.1 แสดงวิธีการกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบ และตอนที่ 5.2 แสดงการทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบด้วยการตรวจสอบแบบจำลอง โดยที่ ตอนที่ 5.1 แบ่งเป็น 3 ตอนย่อยคือ ตอนที่ 5.1.1 แสดงการกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบ ตอนที่ 5.1.2 แสดงตัวอย่างการกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบ และการแปลงแบบจำลองประกอบเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก และตอนที่ 5.1.3 วิเคราะห์ความสัมพันธ์ของฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบกับฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองบนพื้นฐานของกระแสเฉพาะรายที่นำมาประกอบ และที่ ตอนที่ 5.2 แบ่งเป็น 3 ตอนย่อยคือ ตอนที่ 5.2.1 แสดงวิธีการในการกำหนดสมบัตินำเข้า/นำออกของแบบจำลองประกอบ ตอนที่ 5.2.2 แสดงการทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบด้วยการตรวจสอบแบบจำลอง และตอนที่ 5.2.3. แสดงตัวอย่างการกำหนดสมบัตินำเข้า/นำออกของแบบจำลองประกอบ และวิธีการทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบด้วยการตรวจสอบแบบจำลอง

5.1 การกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบ

ในตอนนี้นำเสนอวิธีการกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบ เพื่อที่จะใช้ลดจำนวนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกที่ได้รับจากการแปลงแบบจำลองประกอบ จากการพิจารณาวิธีการลดจำนวนสถานะของฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ สามารถจำแนกได้เป็น 4 ลักษณะด้วยกันคือ

1. การลดความยาวของกระแสนำเข้า
2. การจำแนกข้อมูลนำเข้า
3. การจำแนกกระแสนำเข้า
4. การกำหนดตัวแปรภายในเพื่อใช้เก็บข้อมูลที่ต้องการ

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ทั้ง 6 ฟังก์ชันที่นำเสนอในบทที่ 4 สามารถจำแนกตามลักษณะการลดจำนวนสถานะของฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ทั้ง 4 ลักษณะคือ

1. ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ที่เกี่ยวข้องกับการลดความยาวของกระแสนำเข้า ได้แก่ ฟังก์ชัน *current* ที่เก็บเฉพาะสมาชิกตัวหลังสุดของกระแสนำเข้าฟังก์ชัน $last(n+1)$ ที่เก็บสมาชิกของกระแสนำเข้า $n+1$หลังสุด และฟังก์ชัน *identity* ที่คงสภาพกระแสนำเข้าเดิม

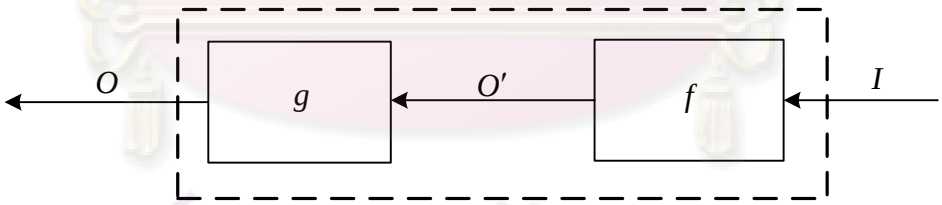
2. ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ที่เกี่ยวข้องกับการจำแนกข้อมูลนำเข้า ได้แก่ ฟังก์ชัน $filterU$ ที่จำแนกชนิดข้อมูลนำเข้าเป็นชนิดอัพเดท และคำร้องขอ

3. ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ที่เกี่ยวข้องกับการจำแนกกระแสนำเข้า ได้แก่ ฟังก์ชัน *break* ที่จำแนกกระแสนำเข้าเป็นกระแสนำเข้าปกติ และกระแสนำเข้าคงที่

4. ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ที่เกี่ยวข้องกับการกำหนดตัวแปรภายในเพื่อใช้เก็บข้อมูลที่ต้องการ ได้แก่ ฟังก์ชัน $reduce(s,\oplus)$ ที่ดำเนินการสมาชิกของกระแสนำเข้าด้วยตัวดำเนินการทวิภาคแล้วเก็บสะสมไว้ที่ตัวแปรภายใน

5.1.1 การกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบ

หลักในการกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบโดยพิจารณาพฤติกรรมนำออกของแบบจำลองประกอบสามารถแสดงได้ดังนี้



รูปที่ 5.1 แบบจำลองประกอบ

5.1.1.1 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติ

จากรูปที่ 5.1 กำหนดให้ f และ g เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติ และ $I = x_1, x_2, ..., x_n$ ดังนั้น $O' = f(x_1), f(x_2), ..., f(x_n)$ และ $O = g(f(x_1)), g(f(x_2)), ..., g(f(x_n))$ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติและ

ฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับสมาชิกของกระแสนำเข้าตัวปัจจุบัน ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน *current*

5.1.1.2 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต

จากรูปที่ 5.1 กำหนดให้ f เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติ $g(n'')$ เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต (กำหนด $n''=2$) และ $I=x_1,x_2,...,x_n$ ดังนั้น $O'=f(x_1),f(x_2),...,f(x_n)$ และ $O=g(2)(f(x_1)),g(2)(f(x_1),f(x_2)),...,g(2)(f(x_{n-1}),f(x_n))$ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขตด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับสมาชิกของกระแสนำเข้า 2 ตัวหลังสุดในกรณีที่ความยาวของกระแสนำเข้ามากกว่า 2 หรือสมาชิกของกระแสนำเข้าทั้งหมดในกรณีที่ความยาวของกระแสนำเข้าน้อยกว่าหรือเท่ากับ 2 ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือฟังก์ชัน $last(n''+1)$

5.1.1.3 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติและฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอ

จากรูปที่ 5.1 กำหนดให้ f เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติ g เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอ และ $I=x_1,x_2,...,x_n$ ดังนั้น $O'=f(x_1),f(x_2),...,f(x_n)$ และ $O=g(f(x_1),f(x_2),...,f(x_{n-1}))$ เมื่อ $f(x_n)$ คือ คำสั่งร้องขอ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติและฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับชนิดของข้อมูลนำเข้า ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน *filterU*

5.1.1.4 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติและฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่

จากรูปที่ 5.1 กำหนดให้ f เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติ g เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่ และ $I = x_1, x_2, ..., x_n$ ดังนั้น $O' = f(x_1), f(x_2), ..., f(x_n)$ และ $O = g(f(x_1)), g(f(x_1), f(x_2)), ..., g(f(x_1), f(x_2))$ เมื่อ $f(x_1), f(x_2)$ คือ กระแสคงที่ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติและฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่ ด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับชนิดของกระแสนำเข้า ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน *break*

5.1.1.5 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติและฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกน

จากรูปที่ 5.1 กำหนดให้ f เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติ $g(\otimes)$ เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกน และ $I = x_1, x_2, ..., x_n$ ดังนั้น $O' = f(x_1), f(x_2), ..., f(x_n)$ และ $O = g(\otimes)(f(x_1)), g(\otimes)(f(x_1) \otimes f(x_2)), ..., g(\otimes)(f(x_1) \otimes f(x_2) \otimes, ..., \otimes f(x_n))$ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติและฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกนด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับผลลัพธ์ของการดำเนินการสมาชิกของกระแสนำเข้าด้วยตัวดำเนินการทวิภาคที่กำหนด ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน $reduce(c, \otimes)$

5.1.1.6 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติ

จากรูปที่ 5.1 กำหนดให้ f เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติ g เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติและ $I = x_1, x_2, ..., x_n$ ดังนั้น $O' = f(x_1), f(x_2), ..., f(x_n)$ และ $O = g(f(x_1)), g(f(x_1), f(x_2)), ..., g(f(x_1), f(x_2), ..., f(x_n))$ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจาก

ประวัติและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับกระแสนำเข้าทั้งหมด ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน *identity*

5.1.1.7 **ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขตและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติ**

จากรูปที่ 5.1 กำหนดให้ $f(n')$ เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต (กำหนด $n'=2$) g เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติ และ $I=x_1,x_2,...,x_n$ ดังนั้น $O'=f(x_1),f(x_1,x_2),...,f(x_{n-1},x_n)$ และ $O=g(f(x_1)),g(f(x_1,x_2)),...,g(f(x_{n-1},x_n))$ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต และฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับสมาชิกของกระแสนำเข้า 2 ตัวหลังสุดในกรณีที่ความยาวของกระแสนำเข้ามากกว่า 2 หรือสมาชิกของกระแสนำเข้าทั้งหมดในกรณีที่ความยาวของกระแสนำเข้าน้อยกว่าหรือเท่ากับ 2 ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน $last(n'+1)$

5.1.1.8 **ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขตและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต**

จากรูปที่ 5.1 กำหนดให้ $f(n')$ เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต (กำหนด $n'=2$) $g(n'')$ เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต (กำหนด $n''=2$) และ $I=x_1,x_2,...,x_n$ ดังนั้น $O'=f(2)(x_1),f(2)(x_1,x_2),...,f(2)(x_{n-1},x_n)$ และ $O=g(2)(f(2)(x_1)),g(2)(f(2)(x_1),f(2)(x_1,x_2)),...,g(2)(f(2)(x_{n-4},x_{n-3}),(f(2)(x_{n-3},x_{n-2}),f(2)(x_{n-2},x_{n-1}),f(2)(x_{n-1},x_n))$ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขตและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขตด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับสมาชิกของกระแสนำเข้า 4 ตัวหลังสุดใน

กรณีที่ความยาวของกระแสนำเข้ามากกว่า 4 หรือสมาชิกของกระแสนำเข้าทั้งหมดในกรณีที่ความยาวของกระแสนำเข้าน้อยกว่าหรือเท่ากับ 4 ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน $last((n'+n'')+1)$

5.1.1.9 **ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขตและฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอ**

จากรูปที่ 5.1 กำหนดให้ $f(n')$ เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต (กำหนด $n'=2$) g เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอ และ $I=x_1,x_2,...,x_n$ ดังนั้น $O'=f(x_1),f(x_1,x_2),...,f(x_{n-1},x_n)$ และ $O=g(f(x_1),f(x_1,x_2),...,f(x_{n-2},x_{n-1}))$ เมื่อ $f(x_{n-1},x_n)$ คือ คำสั่งร้องขอ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขตและฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับชนิดของข้อมูลนำเข้า ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน *filterU*

5.1.1.10 **ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขตและฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่**

จากรูปที่ 5.1 กำหนดให้ $f(n')$ เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต (กำหนด $n'=2$) g เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่ และ $I=x_1,x_2,...,x_n$ ดังนั้น $O'=f(x_1),f(x_1,x_2),...,f(x_{n-1},x_n)$ และ $O=g(f(x_1)),g(f(x_1),f(x_1,x_2)),...,g(f(x_1),f(x_1,x_2))$ เมื่อ $f(x_1),f(x_1,x_2)$ คือ กระแสคงที่ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขตและฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่ ด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับชนิดของกระแสนำเข้า ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน *break*

5.1.1.11 **ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขตและฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกน**

จากรูปที่ 5.1 กำหนดให้ $f(n')$ เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต (กำหนด $n'=2$) $g(\otimes)$ เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกน และ $I = x_1, x_2, ..., x_n$ ดังนั้น $O' = f(x_1), f(x_1, x_2), ..., f(x_{n-1}, x_n)$ และ $O = g(\otimes)(f(x_1)), g(\otimes)(f(x_1) \otimes f(x_1, x_2)), ..., g(\otimes)(f(x_1) \otimes f(x_1, x_2) \otimes, ..., \otimes f(x_{n-1}, x_n))$ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขตและฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกนด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับผลลัพธ์ของการดำเนินการสมาชิกของกระแสนำเข้าด้วยตัวดำเนินการทวิภาคที่กำหนด ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน $reduce(c, \otimes)$

5.1.1.12 **ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขตและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติ**

จากรูปที่ 5.1 กำหนดให้ $f(n')$ เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต (กำหนด $n'=2$) g เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติและ $I = x_1, x_2, ..., x_n$ ดังนั้น $O' = f(x_1), f(x_1, x_2), ..., f(x_{n-1}, x_n)$ และ $O = g(f(x_1)), g(f(x_1), f(x_1, x_2)), ..., g(f(x_1), f(x_1, x_2), ..., f(x_{n-1}, x_n))$ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขตและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับกระแสนำเข้าทั้งหมด ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน *identity*

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

5.1.1.13 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติ

จากรูปที่ 5.1 กำหนดให้ f เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอ g เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติ และ $I = x_1, r, x_2, x_3, r, ..., x_n, r$ ดังนั้น $,..., x_n)$ และ $O = g(f(x_1)), g(f(x_1, x_2, x_3)), ..., g(f(x_1, x_2, x_3, ..., x_n))$ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับชนิดของข้อมูลนำเข้า ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน *filterU*

5.1.1.14 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต

จากรูปที่ 5.1 กำหนดให้ f เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอ $g(n'')$ เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต (กำหนด $n'' = 2$) และ $I = x_1, r, x_2, x_3, r, ..., x_n, r$ ดังนั้น $O' = f(x_1), f(x_1, x_2, x_3), ..., f(x_1, x_2, x_3, ..., x_n)$ และ $O = g(2)(f(x_1)), g(2)(f(x_1), f(x_1, x_2, x_3)), ..., g(2)(f(x_1, x_2, x_3, ..., x_{n-1}), f(x_1, x_2, x_3, ..., x_n))$ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขตด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับชนิดของข้อมูลนำเข้า ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน *filterU*

5.1.1.15 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอและฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอ

จากรูปที่ 5.1 กำหนดให้ f เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอ g เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอ และ $I = x_1, r, x_2, x_3, r, ..., x_n, r$ ดังนั้น $O' = f(x_1), f(x_1, x_2, x_3), ..., f(x_1, x_2, x_3, ...$

$,x_n)$ และ $O = g(f(x_1), f(x_1,x_2,x_3),..., f(x_1,x_2,x_3,...,x_{n-1}))$ เมื่อ $f(x_1,x_2,x_3,...,x_n)$ คือ คำสั่งร้องขอ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอและฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับชนิดของข้อมูลนำเข้า ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน $filterU$

5.1.1.16 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอและฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่

จากรูปที่ 5.1 กำหนดให้ f เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอ g เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่ และ $I = x_1, r, x_2, x_3, r, ..., x_n, r$ ดังนั้น $O' = f(x_1), f(x_1,x_2,x_3),..., f(x_1,x_2,x_3,...,x_n)$ และ $O = g(f(x_1)), g(f(x_1), f(x_1,x_2,x_3)),..., g(f(x_1), f(x_1,x_2,x_3),..., f(x_1,x_2,x_3,...,x_{n-1}))$ เมื่อ $f(x_1,x_2,x_3,...,x_n)$ คือ กระแสคงที่ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอและฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่ด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับชนิดของข้อมูลนำเข้า ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน $filterU$

5.1.1.17 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอและฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกน

จากรูปที่ 5.1 กำหนดให้ f เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอ $g(\otimes)$ เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกน และ $I = x_1, r, x_2, x_3, r, ..., x_n, r$ ดังนั้น $O' = f(x_1), f(x_1,x_2,x_3),..., f(x_1,x_2,x_3,...,x_n)$ และ $O = g(\otimes)(f(x_1)), g(\otimes)(f(x_1) \otimes f(x_1,x_2,x_3)),..., g(\otimes)(f(x_1) \otimes f(x_1,x_2,x_3) \otimes ,..., \otimes f(x_1,x_2,x_3,...,x_n))$ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอและฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกนด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับชนิดของข้อมูลนำเข้า และผลลัพธ์ของการดำเนินการสมาชิกของกระแสนำเข้าด้วยตัวดำเนินการทวิภาคที่กำหนด ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน $filterU; reduce(c, \otimes)$

5.1.1.18 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติ

จากรูปที่ 5.1 กำหนดให้ f เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอ g เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติและ $I = x_1, r, x_2, x_3, r, ..., x_n, r$ ดังนั้น $O' = f(x_1), f(x_1, x_2, x_3), ..., f(x_1, ..., x_n)$ และ $O = g(f(x_1)), g(f(x_1), f(x_1, x_2, x_3)), ..., g(f(x_1), f(x_1, x_2, x_3), ..., f(x_1, ..., x_n))$ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับชนิดของข้อมูลนำเข้า ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน *filterU*

5.1.1.19 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่และฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติ

จากรูปที่ 5.1 กำหนดให้ f เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่ g เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติ และ $I = x_1, x_2, s_1, ..., x_n$ ดังนั้น $O' = f(x_1), f(x_1, x_2), f(x_1, x_2, s_1), ..., f(x_1, x_2, s_1)$ และ $O = g(f(x_1)), g(f(x_1, x_2)), g(f(x_1, x_2, s_1)), ..., g(f(x_1, x_2, s_1)))$ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่และฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับชนิดของกระแสนำเข้า ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน *break*

5.1.1.20 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่และฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต

จากรูปที่ 5.1 กำหนดให้ f เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่ $g(n'')$ เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต (กำหนด $n'' = 2$) และ $I = x_1, x_2, s_1, ..., x_n$ ดังนั้น $O' = f(x_1), f(x_1, x_2), f(x_1, x_2, s_1), ..., f(x_1, x_2, s_1)$ และ $O = g(2)(f(x_1)), g(2)(f(x_1), f(x_1, x_2)), g(2)(f(x_1, x_2), f(x_1, x_2, s_1)), ..., g(2)(f(x_1, x_2), f(x_1, x_2, s_1))$ นั่นคือพฤติกรรมนำออกของ

แบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่และฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขตด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับชนิดของกระแสนำเข้า ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน *break*

5.1.1.21 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่และฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอ

จากรูปที่ 5.1 กำหนดให้ f เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่ g เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอ และ $I = x_1, x_2, s_1, ..., x_n$ ดังนั้น $O' = f(x_1), f(x_1, x_2), f(x_1, x_2, s_1), ..., f(x_1, x_2, s_1)$ และ $O = g(f(x_1), f(x_1, x_2)), ..., g(f(x_1), f(x_1, x_2))$ เมื่อ $f(x_1, x_2, s_1)$ คือ คำสั่งร้องขอ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่และฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับชนิดของกระแสนำเข้า ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน *break*

5.1.1.22 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่และฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่

จากรูปที่ 5.1 กำหนดให้ f เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่ g เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่ และ $I = x_1, x_2, s_1, ..., x_n$ ดังนั้น $O' = f(x_1), f(x_1, x_2), f(x_1, x_2, s_1), ..., f(x_1, x_2, s_1)$ และ $O = g(f(x_1)), g(f(x_1), f(x_1, x_2)), g(f(x_1), f(x_1, x_2), f(x_1, x_2, s_1)), ..., g(f(x_1), f(x_1, x_2), f(x_1, x_2, s_1))$ เมื่อ $f(x_1), f(x_1, x_2), f(x_1, x_2, s_1)$ คือ กระแสคงที่ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่และฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่ ด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับชนิดของกระแสนำเข้า ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน *break*

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

5.1.1.23 **ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่และฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกน**

จากรูปที่ 5.1 กำหนดให้ f เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่ $g(\otimes)$ เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกน และ $I = x_1, x_2, s_1, ..., x_n$ ดังนั้น $O' = f(x_1), f(x_1, x_2), f(x_1, x_2, s_1), ..., f(x_1, x_2, s_1)$ และ $O = g(\otimes)(f(x_1)), g(\otimes)(f(x_1) \otimes f(x_1, x_2)), g(\otimes)(f(x_1) \otimes f(x_1, x_2) \otimes f(x_1, x_2, s_1)), ..., g(\otimes)(f(x_1) \otimes f(x_1, x_2) \otimes f(x_1, x_2, s_1))$ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่และฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกนด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับชนิดของกระแสนำเข้าและผลลัพธ์ของการดำเนินการสมาชิกของกระแสนำเข้าด้วยตัวดำเนินการทวิภาคที่กำหนด ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือฟังก์ชัน $break; reduce(c, \otimes)$

5.1.1.24 **ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่และฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติ**

จากรูปที่ 5.1 กำหนดให้ f เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่ g เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติและ $I = x_1, x_2, s_1, ..., x_n$ ดังนั้น $O' = f(x_1), f(x_1, x_2), f(x_1, x_2, s_1), ..., f(x_1, x_2, s_1)$ และ $O = g(f(x_1)), g(f(x_1), f(x_1, x_2)), g(f(x_1), f(x_1, x_2), f(x_1, x_2, s_1)), ..., g(f(x_1), f(x_1, x_2), f(x_1, x_2, s_1), ..., f(x_1, x_2, s_1))$ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่และฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับชนิดของกระแสนำเข้า ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือฟังก์ชัน *break*

5.1.1.25 **ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกนและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติ**

จากรูปที่ 5.1 กำหนดให้ $f(\oplus)$ เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกน g เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติ และ $I = x_1, x_2, ..., x_n$ ดังนั้น $O' = f(\oplus)(x_1), f(\oplus)(x_1 \oplus x_2), ..., f(\oplus)(x_1 \oplus ... \oplus x_n)$ และ

$O = g(f(\oplus)(x_1)), g(f(\oplus)(x_1 \oplus x_2)), ..., g(f(\oplus)(x_1 \oplus ... \oplus x_n))$ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกนและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับผลลัพธ์ของการดำเนินการสมาชิกของกระแสนำเข้าด้วยตัวดำเนินการทวิภาคที่กำหนด ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน $reduce(s, \oplus)$

5.1.1.26 **ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกนและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต**

จากรูปที่ 5.1 กำหนดให้ $f(\oplus)$ เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกน $g(n'')$ เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต (กำหนด $n'' = 2$) และ $I = x_1, x_2, ..., x_n$ ดังนั้น $O' = f(\oplus)(x_1), f(\oplus)(x_1 \oplus x_2), ..., f(\oplus)(x_1 \oplus ... \oplus x_n)$ และ $O = g(f(\oplus)(x_1)), g(f(\oplus)(x_1), f(\oplus)(x_1 \oplus x_2)), ..., g(f(\oplus)(x_1 \oplus ... \oplus x_{n-1}), f(\oplus)(x_1 \oplus ... \oplus x_n))$ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกนและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขตด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่ ผลลัพธ์ของการดำเนินการสมาชิกของกระแสนำเข้าด้วยตัวดำเนินการทวิภาคที่กำหนด ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน $reduce(s, \oplus)$

5.1.1.27 **ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกนและฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอ**

จากรูปที่ 5.1 กำหนดให้ $f(\oplus)$ เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกน g เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอ และ $I = x_1, x_2, ..., x_n$ ดังนั้น $O' = f(\oplus)(x_1), f(\oplus)(x_1 \oplus x_2), ..., f(\oplus)(x_1 \oplus ... \oplus x_n)$ และ $O = g(f(\oplus)(x_1), f(\oplus)(x_1 \oplus x_2), ..., f(\oplus)(x_1 \oplus ... \oplus x_{n-1}))$ เมื่อ $f(\oplus)(x_1 \oplus ... \oplus x_n)$ คือ คำสั่งร้องขอ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกนและฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับชนิดของข้อมูลนำเข้า ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน *filterU*

5.1.1.28 **ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกนและฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่**

จากรูปที่ 5.1 กำหนดให้ $f(\oplus)$ เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกน g เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่ และ $I = x_1, x_2, ..., x_n$ ดังนั้น $O' = f(\oplus)(x_1), f(\oplus)(x_1 \oplus x_2), ..., f(\oplus)(x_1 \oplus ... \oplus x_n)$ และ $O = g(f(\oplus)(x_1)), g(f(\oplus)(x_1), f(\oplus)(x_1 \oplus x_2)), ..., g(f(\oplus)(x_1), f(\oplus)(x_1 \oplus x_2))$ เมื่อ $f(\oplus)(x_1), f(\oplus)(x_1 \oplus x_2)$ คือ กระแสคงที่ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกนและฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่ ด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับชนิดของกระแสนำเข้า ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือฟังก์ชัน *break*

5.1.1.29 **ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกนและฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกน**

จากรูปที่ 5.1 กำหนดให้ $f(\oplus)$ เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกน $g(\otimes)$ เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกน และ $I = x_1, x_2, ..., x_n$ ดังนั้น $O' = f(\oplus)(x_1), f(\oplus)(x_1 \oplus x_2), ..., f(\oplus)(x_1 \oplus ... \oplus x_n)$ และ $O = g(\otimes)(f(\oplus)(x_1)), g(\otimes)(f(\oplus)(x_1) \otimes f(\oplus)(x_1 \oplus x_2)), ..., g(\otimes)(f(\oplus)(x_1) \otimes f(\oplus)(x_1 \oplus x_2) \otimes, ..., \otimes f(\oplus)(x_1 \oplus ... \oplus x_n))$ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกนและฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกนด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับผลลัพธ์ของการดำเนินการสมาชิกของกระแสนำเข้าด้วยตัวดำเนินการทวิภาคที่กำหนด ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน $reduce(s, \oplus)(c, \otimes)$

5.1.1.30 **ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกนและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติ**

จากรูปที่ 5.1 กำหนดให้ $f(\oplus)$ เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกน g เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติและ $I = x_1, x_2, ..., x_n$ ดังนั้น $O' = f(\oplus)(x_1), f(\oplus)(x_1 \oplus x_2), ..., f(\oplus)(x_1 \oplus ... \oplus x_n)$ และ

$O = g(f(\oplus)(x_1)), g(f(\oplus)(x_1), f(\oplus)(x_1 \oplus x_2)), ..., g(f(\oplus)(x_1), f(\oplus)(x_1 \oplus x_2), ...,$ $f(\oplus)(x_1 \oplus ... \oplus x_n)$ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกนและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับกระแสนำเข้าทั้งหมด ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน *identity*

5.1.1.31 **ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติ**

จากรูปที่ 5.1 กำหนดให้ f เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติ g เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติ และ $I = x_1, x_2, ..., x_n$ ดังนั้น $O' = f(x_1), f(x_1, x_2), ..., f(x_1, x_2, ..., x_n)$ และ $O = g(f(x_1)), g(f(x_1, x_2)), ..., g(f(x_1, x_2, ..., x_n))$ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับกระแสนำเข้าทั้งหมด ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน *identity*

5.1.1.32 **ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต**

จากรูปที่ 5.1 กำหนดให้ f เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติ $g(n'')$ เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต (กำหนด $n'' = 2$) และ $I = x_1, x_2, ..., x_n$ ดังนั้น $O' = f(x_1), f(x_1, x_2), ..., f(x_1, x_2, ..., x_n)$ และ $O = g(f(x_1)), g(f(x_1), f(x_1, x_2)), ...,$ $g(f(x_1, x_2, ..., x_{n-1}), f(x_1, x_2, ..., x_n))$ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขตด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับกระแสนำเข้าทั้งหมด ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน *identity*

5.1.1.33 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติและฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอ

จากรูปที่ 5.1 กำหนดให้ f เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติ g เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอ และ $I = x_1, x_2, ..., x_n$ ดังนั้น $O' = f(x_1), f(x_1, x_2), ..., f(x_1, x_2, ..., x_n)$ และ $O = g(f(x_1), f(x_1, x_2), ..., f(x_1, x_2, ..., x_{n-1}))$ เมื่อ $f(x_1, x_2, ..., x_{n-1})$ คือ คำสั่งร้องขอ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติและฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับชนิดของข้อมูลนำเข้า ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน *filterU*

5.1.1.34 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติและฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่

จากรูปที่ 5.1 กำหนดให้ f เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติ g เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่ และ $I = x_1, x_2, ..., x_n$ ดังนั้น $O' = f(x_1), f(x_1, x_2), ..., f(x_1, x_2, ..., x_n)$ และ $O = g(f(x_1)), g(f(x_1), f(x_1, x_2)), ..., g(f(x_1), f(x_1, x_2))$ เมื่อ $f(x_1), f(x_1, x_2)$ คือ กระแสคงที่ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติและฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่ ด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับชนิดของกระแสนำเข้า ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน *break*

5.1.1.35 ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติและฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกน

จากรูปที่ 5.1 กำหนดให้ f เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติ $g(\otimes)$ เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกน และ $I = x_1, x_2, ..., x_n$ ดังนั้น $O' = f(x_1), f(x_1, x_2), ..., f(x_1, x_2, ..., x_n)$ และ $O = g(\otimes)$ $(f(x_1)), g(\otimes)(f(x_1) \otimes f(x_1, x_2)), ..., g(\otimes)(f(x_1) \otimes f(x_1, x_2) \otimes, ..., \otimes f(x_1, x_2, ..., x_n))$ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติและฟังก์ชันการประมวลผลกระแสของส่วนประกอบ

สแกนด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับกระแสนำเข้าทั้งหมด ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน *identity*

5.1.1.36 **ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติ**

จากรูปที่ 5.2 กำหนดให้ f เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติ g เป็นฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติและ $I = x_1, x_2, ..., x_n$ ดังนั้น $O' = f(x_1), f(x_1, x_2), ..., f(x_1, x_2, ..., x_n)$ และ $O = g(f(x_1)), g(f(x_1), f(x_1, x_2)), ..., g(f(x_1), f(x_1, x_2), ..., f(x_1, x_2, ..., x_n))$ นั่นคือพฤติกรรมนำออกของแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่ต้องการประวัติด้วยตัวดำเนินการประกอบแบบลำดับขึ้นอยู่กับกระแสนำเข้าทั้งหมด ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน *identity*

ดังนั้นสามารถกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบได้จากการพิจารณาแบบจำลองบนพื้นฐานของกระแสเฉพาะรายที่นำมาประกอบ

5.1.2 **ตัวอย่างการกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบ และการแปลงแบบจำลองประกอบเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก**

ต่อไปนี้เป็นการแสดงตัวอย่างการกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบ เพื่อใช้ในการแปลงแบบจำลองประกอบเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

5.1.2.1 **ตัวอย่างส่วนประกอบค่าเฉลี่ย** (Average component)

ตัวอย่างที่ 5.1 ส่วนประกอบค่าเฉลี่ย

ส่วนประกอบค่าเฉลี่ย เป็นส่วนประกอบที่คำนวณค่าเฉลี่ยจากสมาชิกของกระแสนำเข้า โดยนำส่วนประกอบผลรวม $sum(s,+)$ และส่วนประกอบผลหาร $div(c,+)$ มาประกอบกันด้วยตัวดำเนินการประกอบแบบลำดับ ดังแสดงในรูปที่ 5.1 นิยามดังนี้

$$ave : \mathbb{N} \times [\mathbb{N} \times \mathbb{N} \to \mathbb{N}] \to \mathbb{N} \times [\mathbb{N} \times \mathbb{N} \to \mathbb{N}] \to [\mathbb{N}^* \to \mathbb{R}^*]$$

โดยที่

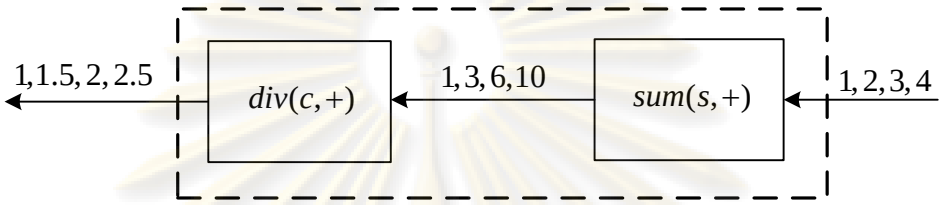
$$ave(s,+)(c,+)(\langle\rangle) = \langle\rangle \quad (5.1)$$

$$ave(s,+)(c,+)(x \lhd X) = ((s+x)/(c+1)) \lhd ave(s+x,+)(c+1,+)(X) \quad (5.2)$$

(5.1) หมายถึง กระแสนำเข้าว่างก่อกำเนิดกระแสนำออกว่าง

(5.2) หมายถึง ดำเนินการบวกสมาชิกของกระแสนำเข้าแล้วเก็บสะสมไว้ในตัวแปรภายในและดำเนินการนับจำนวนสมาชิกของกระแสนำเข้าแล้วเก็บสะสมไว้ในตัวแปรภายในก่อกำเนิดกระแสนำออกด้วยผลบวกสะสมที่เก็บไว้หารด้วยจำนวนสมาชิก

เมื่อ $c, s, x \in \mathbb{N}, X \in \mathbb{N}^*$ และกำหนดค่าเริ่มต้น $s = 0$ และ $c = 0$



รูปที่ 5.2 ส่วนประกอบค่าเฉลี่ย

ส่วนประกอบผลรวม เป็นส่วนประกอบที่ดำเนินการบวกสมาชิกของกระแสนำเข้าแล้วเก็บสะสมไว้ในตัวแปรภายในและก่อกำเนิดกระแสนำออกด้วยผลบวกสะสมที่เก็บไว้ ดังแสดงในรูปที่ 5.3 นิยามดังนี้

$$sum : \mathbb{N} \times [\mathbb{N} \times \mathbb{N} \to \mathbb{N}] \to [\mathbb{N}^* \to \mathbb{N}^*]$$

โดยที่

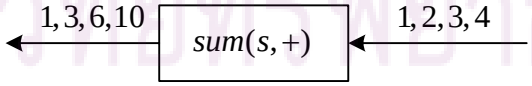
$$sum(s,+)(\langle\rangle) = \langle\rangle \tag{5.3}$$

$$sum(s,+)(x \lhd X) = (s+x) \lhd sum(s+x,+)(X) \tag{5.4}$$

เมื่อ $s, x \in \mathbb{N}, X \in \mathbb{N}^*$ และกำหนดค่าเริ่มต้น $s = 0$

(5.3) หมายถึง กระแสนำเข้าว่างก่อกำเนิดกระแสนำออกว่าง

(5.4) หมายถึง ดำเนินการบวกสมาชิกของกระแสนำเข้าแล้วเก็บสะสมไว้ในตัวแปรภายในและก่อกำเนิดกระแสนำออกด้วยผลบวกสะสมที่เก็บไว้



รูปที่ 5.3 ส่วนประกอบผลรวม

ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบผลรวม คือ ฟังก์ชัน $reduce'(s,+)$ ที่บวกสะสมสมาชิกของกระแสนำเข้า

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ส่วนประกอบผลหาร เป็นส่วนประกอบที่นับจำนวนสมาชิกของกระแสนำเข้าแล้วนำสมาชิกแต่ละตัวหารด้วยจำนวนสมาชิกที่นับได้จากนั้นส่งออกเป็นกระแสนำออก ดังแสดงในรูปที่ 5.4 นิยามดังนี้

$$div : \mathbb{N} \times [\mathbb{N} \times \mathbb{N} \to \mathbb{N}] \to [\mathbb{N}^* \to \mathbb{R}^*]$$

โดยที่

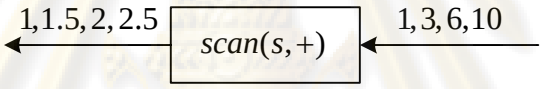
$$div(c,+)(\langle\rangle) = \langle\rangle \tag{5.5}$$

$$div(c,+)(x \lhd X) = x/(c+1) \lhd div(c+1,+)(X) \tag{5.6}$$

เมื่อ $c, x \in \mathbb{N}, X \in \mathbb{N}^*$ และกำหนดค่าเริ่มต้น $c = 0$

(5.5) หมายถึง กระแสนำเข้าว่างก่อกำเนิดกระแสนำออกว่าง

(5.6) หมายถึง นับจำนวนสมาชิกของกระแสนำเข้าแล้วนำสมาชิกแต่ละตัวหารด้วยจำนวนสมาชิกที่นับได้จากนั้นส่งออกเป็นกระแสนำออก



รูปที่ 5.4 ส่วนประกอบผลหาร

ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบผลหาร คือ ฟังก์ชัน $reduce''(c,+)$ ที่นับจำนวนสมาชิกของกระแสนำเข้า

5.1.2.1.1 **การกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบค่าเฉลี่ย**

ส่วนประกอบค่าเฉลี่ยเป็นแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกนและฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกนด้วยตัวดำเนินการประกอบแบบลำดับ ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน $reduce(s,+)(c,+)$ นิยามดังนี้

$$reduce''' : \mathbb{N} \times [\mathbb{N} \times \mathbb{N} \to \mathbb{N}] \to \mathbb{N} \times [\mathbb{N} \times \mathbb{N} \to \mathbb{N}] \to [\mathbb{N}^* \to \mathbb{R}]$$

โดยที่

$$reduce'''(s,+)(c,+)(\langle\rangle) = \begin{cases} \langle\rangle & if\ (c = 0) \\ s/c & if\ (c \neq 0) \end{cases}$$

$$reduce'''(s,+)(c,+)(x \lhd X) = reduce'''(s+x,+)(c+1,+)(X)$$

5.1.2.1.2 **การแปลงส่วนประกอบผลบวกเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก**

ต่อไปนี้เป็นการแปลงส่วนประกอบส่วนประกอบค่าเฉลี่ยเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก โดยแปลงส่วนประกอบส่วนประกอบค่าเฉลี่ยผ่านทางเครื่องมัวร์แบบบัญญัติ จากนั้นจึงแปลงเครื่องมัวร์แบบบัญญัติที่ได้รับเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

5.1.2.1.2.1 **การแปลงส่วนประกอบผลบวกเป็นเครื่องมัวร์แบบบัญญัติ**

ส่วนประกอบผลบวกสามารถแปลงเป็นเครื่องมัวร์แบบบัญญัติได้ดังนี้

$$M_{ave} = (\mathbb{N}^*, \mathbb{N}, \mathbb{R}^*, \delta, \varphi_{ave}, \langle\rangle)$$

โดยที่

ฟังก์ชันนำออกเดี่ยว φ_{ave} คือ

$$\varphi_{ave}(\langle\rangle) = \langle\rangle \tag{5.7}$$

$$\varphi_{ave}(X) = \langle reduce'(s,+)(X) / reduce''(c,+)(X)\rangle \tag{5.8}$$

เมื่อ $c, s \in \mathbb{N}$ และ $X \in \mathbb{N}^*$

(5.7) หมายถึง ไม่มีนำออกเมื่อไม่มีนำเข้า

(5.8) หมายถึง นำออกคือกระแสของผลบวกสะสมหารด้วยจำนวนข้อมูล

5.1.2.1.2.2 **การแปลงเครื่องมัวร์แบบบัญญัติเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก**

เครื่องมัวร์แบบบัญญัติ M_{ave} สามารถแปลงเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก $T_{ave,\alpha}$ ได้ดังนี้

$$T_{ave,\alpha} = (\mathbb{N}, \mathbb{N}^*, Q, S, R, s_0)$$

โดยที่

$$Q = \mathbb{R} \tag{5.9}$$

$$S = (\mathbb{N} \cup \{\phi\}) \times \mathbb{N} \times \mathbb{N}^* \tag{5.10}$$

$$R = \{(\alpha(X), \alpha(X \triangleright x)) \mid X \in \mathbb{N}^*, x \in \mathbb{N}\} \tag{5.11}$$

$$s_0 = (\phi, \langle\rangle, \langle\rangle) \tag{5.12}$$

เมื่อ

$$\alpha(\langle\rangle) = (\phi, reduce'''(s,+)(c,+)(\langle\rangle), \varphi_{ave}(\langle\rangle))$$

$$\alpha(X \triangleright x) = (x, reduce'''(s,+)(c,+)(X \triangleright x), \varphi_{ave}(X \triangleright x))$$

(5.9) หมายถึง เซตของสถานะภายในคือ เซต $\mathbb{R}$ ซึ่งได้รับจากการลดจำนวนสถานะที่ได้รับจากกระแสนำเข้าด้วยฟังก์ชันภาวะนามธรรมประวัติ $reduce'(s,+) / \ reduce''(c,+)$

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

(5.10) หมายถึง ปริภูมิสถานะ S ที่เกิดจากการรวมเซตข้อมูลนำเข้า สถานะภายใน และเซตข้อมูลนำออก ด้วยฟังก์ชันการแปลง α สามารถสร้างสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกภายใต้ความสัมพันธ์การจำลองสองทางกับสถานะของเครื่องมัวร์แบบบัญญัติได้ดังนี้

สร้างสถานะจากสถานะเริ่มต้นที่เป็นกระแสว่าง

$$\langle\rangle \leftrightarrow_H \{(\phi, \langle\rangle, \langle\rangle)\}$$

สร้างสถานะจากสถานะภายในไม่เป็นกระแสว่าง

$$X \triangleright x \leftrightarrow_H \{(x, reduce'(s,+)(X \triangleright x) / reduce''(c,+)(X \triangleright x),$$

$$\langle reduce'(s,+)(X \triangleright x) / reduce''(c,+)(X \triangleright x)\rangle) \mid c, s, x \in \mathbb{N}, X \in \mathbb{N}^*\}$$

(5.11) หมายถึง ความสัมพันธ์การเปลี่ยนสถานะ R แสดงความสัมพันธ์ของสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออก

การเปลี่ยนสถานะจากสถานะเริ่มต้น เมื่อได้รับนำเข้า $x \in \mathbb{N}$

$$(\phi, \langle\rangle, \langle\rangle) \rightarrow (x, (x,1), \langle x/1\rangle)$$

การเปลี่ยนสถานะจากสถานะที่ไม่ใช่สถานะเริ่มต้น และได้รับนำเข้า $y \in \mathbb{N}$

$$(x, (q'+x)/(q''+1), \langle (q'+x)/(q''+1)\rangle) \rightarrow (y, (q'+x+y)/(q''+2), \langle (q'+x+y)/(q''+2)\rangle)$$

เมื่อ $x, y \in \mathbb{N}, q' = reduce'(s,+)(X)$ และ $q'' = reduce''(c,+)(X)$

(5.12) หมายถึง กำหนดสถานะเริ่มต้นของระบบเปลี่ยนสถานะนำเข้า/นำออก s_0 คือ $(\phi, \langle\rangle, \langle\rangle)$

จากระบบเปลี่ยนสถานะนำเข้า/นำออก $T_{ave,\alpha}$ สามารถเขียนให้อยู่ในรูปตารางการเปลี่ยนสถานะ ดังตารางที่ 5.1

ตารางที่ 5.1 ตารางการเปลี่ยนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกของส่วนประกอบส่วนประกอบค่าเฉลี่ย

State			*State′*		
input	*internal state*	*output*	*input′*	*internal state′*	*output′*
ϕ	$\langle\rangle$	$\langle\rangle$	x	$(x,1)$	$\langle x/1\rangle$
x	$(q'+x)/(q''+1)$	$\langle (q'+x)/(q''+1)\rangle$	y	$(q'+x+y)/(q''+2)$	$\langle (q'+x+y)/(q''+2)\rangle$
$x, y \in \mathbb{N}, q' = reduce'(s,+)(X), q'' = reduce''(c,+)(X)$					

5.1.2.2 **ตัวอย่างส่วนประกอบเรจิสเตอร์การเลื่อนคู่** (Couple shift register component)

ตัวอย่างที่ 5.2 ส่วนประกอบเรจิสเตอร์การเลื่อนคู่

ส่วนประกอบเรจิสเตอร์การเลื่อนคู่ เป็นส่วนประกอบที่หน่วงจำนวนสมาชิกของกระแสนำเข้าเท่ากับจำนวนที่กำหนด โดยนำส่วนประกอบเรจิสเตอร์การเลื่อน $shift(n)$ (ดังแสดงในตัวอย่างที่ 2.2) และสองส่วนประกอบมาต่อกันด้วยตัวดำเนินการประกอบแบบลำดับ เมื่อกำหนดค่าคงที่ให้ส่วนประกอบเรจิสเตอร์การเลื่อนแต่ละตัวด้วย 2 สามารถแสดงพฤติกรรมของส่วนประกอบหน่วง ดังแสดงในรูปที่ 5.5 นิยามดังนี้

$$shift' : \mathbb{N}/0 \rightarrow [\mathcal{A}^* \rightarrow \mathcal{A}^*]$$

โดยที่

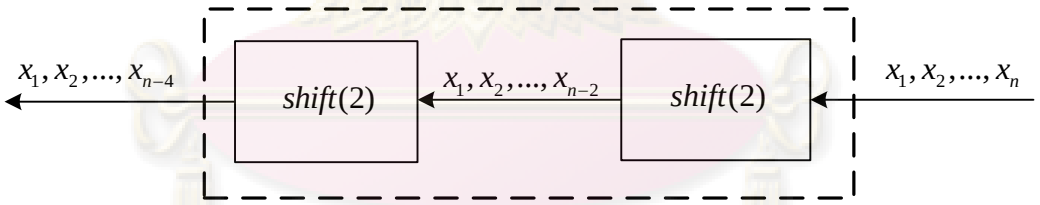
$$shift'(n+n)(\langle\rangle) = \langle\rangle \tag{5.13}$$

$$shift'(n+n)(x \lhd X) = \begin{cases} x \lhd shift'(n+n)(X) \; if \; (|\, x \lhd X \,| > (n+n)) \\ shift'(n+n)(X) \; if \; (|\, x \lhd X \,| \leq (n+n)) \end{cases} \tag{5.14}$$

เมื่อ $x \in \mathcal{A}, X \in \mathcal{A}^*$ และ $n \in (\mathbb{N} \setminus 0)$

(5.13) หมายถึง กระแสนำเข้าว่างก่อกำเนิดกระแสนำออกว่าง

(5.14) หมายถึง ปล่อยนำออกเป็นกระแสของสมาชิกของกระแสนำเข้าเมื่อกระแสนำเข้ามีสมาชิกเกินจำนวนที่กำหนด



รูปที่ 5.5 ส่วนประกอบเรจิสเตอร์การเลื่อนคู่

ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบเรจิสเตอร์การเลื่อนทั้งสองส่วนประกอบคือ ฟังก์ชัน $last(n+1)$ ที่เก็บเฉพาะสมาชิก $n+1$ ตัวหลังสุดไว้

5.1.2.2.1 **การกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบเรจิสเตอร์การเลื่อนคู่**

ส่วนประกอบเรจิสเตอร์การเลื่อนคู่เป็นแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขตและฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมี

ขอบเขตด้วยตัวดำเนินการประกอบแบบลำดับ ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน $last(n'+1)$ ที่เก็บเฉพาะสมาชิก $n'+1$ ตัวหลังสุดไว้ เมื่อ $n' = n+n$

5.1.2.2.2 **การแปลงส่วนประกอบเรจิสเตอร์การเลื่อนคู่เป็นระบบเปลี่ยนสถานะนำเข้า/นำออก**

ต่อไปนี้เป็นการแปลงส่วนประกอบเรจิสเตอร์การเลื่อนคู่เป็นระบบเปลี่ยนสถานะนำเข้า/นำออก โดยแปลงส่วนประกอบเรจิสเตอร์การเลื่อนคู่ผ่านทางเครื่องมัวร์แบบบัญญัติ จากนั้นจึงแปลงเครื่องมัวร์แบบบัญญัติที่ได้รับเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

1) **การแปลงส่วนประกอบเรจิสเตอร์การเลื่อนคู่เป็นเครื่องมัวร์แบบบัญญัติ**

ส่วนประกอบเรจิสเตอร์การเลื่อนสามารถแปลงเป็นเครื่องมัวร์แบบบัญญัติได้ดังนี้

$$M_{shift'(n')} = (\mathcal{A}^*, \mathcal{A}, \mathcal{A}^*, \delta,\ \varphi_{shift'(n')}, \langle\rangle)$$

โดยที่

ฟังก์ชันนำออกเดี่ยว $\varphi_{shift'(n')}$ คือ

$$\varphi_{shift'(n')}(X) = \langle\rangle \tag{5.15}$$

$$\varphi_{shift'(n')}(Y) = \langle Y_{[|Y|-(n'+1)]} \rangle \tag{5.16}$$

เมื่อ $|X| \le n'$ และ $|Y| > n'$

(5.15) หมายถึงไม่มีนำออกเมื่อกระแสนำเข้ามีขนาดน้อยกว่าหรือเท่ากับ n

(5.16) หมายถึงนำออกคือสมาชิกตัวที่ $n'+1$ นับมาจากหลังสุดของกระแสนำเข้าเมื่อกระแสนำเข้ามีขนาดมากกว่า n

2) **การแปลงเครื่องมัวร์แบบบัญญัติเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก**

เครื่องมัวร์แบบบัญญัติ $M_{shift'(n')}$ สามารถแปลงเป็นระบบเปลี่ยนสถานะนำเข้า/ นำออก $T_{shift'(n'),\alpha}$ ได้ดังนี้

$$T_{shift'(n'),\alpha} = (\mathcal{A}, \mathcal{A}^*, Q, S, R, s_0)$$

โดยที่

$$Q = \mathcal{A}^{\le n'+1} \tag{5.17}$$

$$S = (\mathcal{A} \cup \{\phi\}) \times \mathcal{A}^{\le n'+1} \times \mathcal{A} \tag{5.18}$$

$$R = \{(\alpha(X), \alpha(X \rhd x)) \mid X \in \mathcal{A}^*, x \in \mathcal{A}\} \tag{5.19}$$

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

$$s_0 = (\phi, last(n'+1), \varphi_{shift'(n')}(\langle\rangle)) = (\phi, \langle\rangle, \langle\rangle) \quad (5.20)$$

เมื่อ $$\alpha(\langle\rangle) = (\phi, last(n'+1), \varphi_{shift'(n')}(\langle\rangle))$$

$$\alpha(X \rhd x) = (x, last(n'+1)(X \rhd x), \varphi_{shift'(n')}(X \rhd x))$$

(5.17) หมายถึง เซตของสถานะภายในคือ กระแส $\mathcal{A}^{\leq n'+1}$ ซึ่งได้รับจากการลดจำนวนสถานะที่ได้รับจากกระแสนำเข้าด้วยฟังก์ชันภาวะนามธรรมประวัติ $last(n'+1)$

(5.18) หมายถึง ปริภูมิสถานะ S ที่เกิดจากการรวมเซตข้อมูลนำเข้า สถานะภายใน และเซตข้อมูลนำออกด้วยฟังก์ชันการแปลง α สามารถสร้างสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกภายใต้ความสัมพันธ์การจำลองสองทางกับสถานะของเครื่องมัวร์แบบบัญญัติได้ดังนี้

สร้างสถานะจากสถานะเริ่มต้นที่เป็นกระแสว่าง

$$\langle\rangle \leftrightarrow_H \{(\phi, \langle\rangle, \langle\rangle)\}$$

สร้างสถานะเมื่อ $|X \rhd x| < n'+1$

$$X \rhd x \leftrightarrow_H \{(x, X \rhd x, \langle\rangle) \mid x \in \mathcal{A}, X \in \mathcal{A}^*\}$$

สร้างสถานะเมื่อ $|y \lhd X \rhd x| = n'+1$

$$y \lhd X \rhd x \leftrightarrow_H \{(x, y \lhd X \rhd x, \langle y\rangle) \mid x, y \in \mathcal{A}, X \in \mathcal{A}^*\}$$

(5.19) หมายถึง ความสัมพันธ์การเปลี่ยนสถานะ R แสดงความสัมพันธ์ของสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออก

การเปลี่ยนสถานะจากสถานะเริ่มต้น

$$(\phi, \langle\rangle, \langle\rangle) \rightarrow (x, \langle x\rangle, \langle\rangle)$$

การเปลี่ยนสถานะ เมื่อ $|X \rhd x| < n'$

$$(x, X \rhd x, \langle\rangle) \rightarrow (y, X \rhd x \rhd y, \langle\rangle)$$

การเปลี่ยนสถานะ เมื่อ $|z \lhd X \rhd x| = n'$

$$(x, z \lhd X \rhd x, \langle\rangle) \rightarrow (y, X \rhd x \rhd y, \langle z\rangle)$$

การเปลี่ยนสถานะ เมื่อ $|z \lhd w \lhd X \rhd x| = n'+1$

$$(x, z \lhd w \lhd X \rhd x, \langle z\rangle) \rightarrow (y, w \lhd X \rhd x \rhd y, \langle w\rangle)$$

เมื่อ $w, x, y, z \in \mathcal{A}$ และ $X \in \mathcal{A}^*$

(5.20) หมายถึง กำหนดสถานะเริ่มต้นของระบบเปลี่ยนสถานะนำเข้า/นำออก s_0 คือ $(\phi, \langle\rangle, \langle\rangle)$

จากระบบเปลี่ยนสถานะนำเข้า/นำออก $T_{shift'(n'),\alpha}$ สามารถเขียนให้อยู่ในรูปตารางการเปลี่ยนสถานะ ดังตารางที่ 5.2

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ตารางที่ 5.2 ตารางการเปลี่ยนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกของส่วนประกอบเรจิสเตอร์การเลื่อนคู่

State			*State′*		
input	*internal state*	*output*	*input′*	*internal state′*	*output′*
ϕ	$\langle\rangle$	$\langle\rangle$	x_1	$\langle x_1\rangle$	$\langle\rangle$
x_m	$\langle x_1,...,x_m\rangle$	$\langle\rangle$	x_{m+1}	$\langle x_1,...,x_{m+1}\rangle$	$\langle\rangle$
$x_{n'}$	$\langle x_1,...,x_{n'}\rangle$	$\langle\rangle$	$x_{n'+1}$	$\langle x_1,...,x_{n'+1}\rangle$	$\langle x_1\rangle$
$x_{n'+1}$	$\langle x_1,...,x_{n'+1}\rangle$	$\langle x_1\rangle$	$x_{n'+2}$	$\langle x_1,...,x_{n'+2}\rangle$	$\langle x_2\rangle$
$1 \le m < n'$					

5.1.2.3 **ตัวอย่างส่วนประกอบตัวหน่วงการสำเนา** (Duplicate delay component)

ตัวอย่างที่ 5.3 ส่วนประกอบตัวหน่วงการสำเนา

ส่วนประกอบตัวหน่วงการสำเนา เป็นส่วนประกอบที่หน่วงจำนวนสมาชิกของกระแสนำเข้าเท่ากับจำนวนที่กำหนดแล้วสำเนาสมาชิกแต่ละตัว โดยส่วนประกอบตัวหน่วงการสำเนาเป็นแบบจำลองประกอบที่ประกอบด้วยส่วนประกอบเรจิสเตอร์การเลื่อน $shift(n)$ (ดังแสดงในตัวอย่างที่ 2.2) และส่วนประกอบสำเนา *duplicate* (ดังแสดงในตัวอย่างที่ 4.1) ด้วยตัวดำเนินการประกอบแบบลำดับ เมื่อกำหนดค่าคงที่ด้วย 3 สามารถแสดงพฤติกรรมของส่วนประกอบหน่วงการสำเนา ดังแสดงในรูปที่ 5.6 นิยามดังนี้

$$dupDelay : \mathbb{N}/0 \to [\mathcal{A}^* \to \mathcal{A}^*]$$

โดยที่

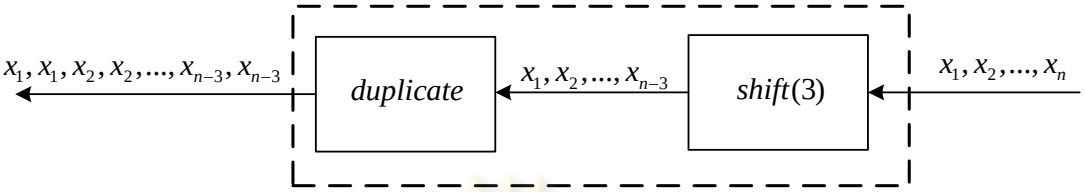
$$dupDelay(n)(\langle\rangle) = \langle\rangle \tag{5.21}$$

$$dupDelay(n)(x \lhd X) = \begin{cases} \langle x, x\rangle \,\&\, dupDelay(n)(X) \; if \; (|\, x \lhd X \,| > n) \\ dupDelay(n)(X) \; if \; (|\, x \lhd X \,| \le n) \end{cases} \tag{5.22}$$

เมื่อ $x \in \mathcal{A}, X \in \mathcal{A}^*$ และ $n \in (\mathbb{N} \setminus 0)$

(5.21) หมายถึง กระแสนำเข้าว่างก่อกำเนิดกระแสนำออกว่าง

(5.22) หมายถึง ปล่อยนำออกเป็นกระแสของสำเนาของสมาชิกของกระแสนำเข้าเมื่อกระแสนำเข้ามีสมาชิกเกินจำนวนที่กำหนด



รูปที่ 5.6 ส่วนประกอบหน่วงการสำเนา

ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบเรจิสเตอร์การเลื่อนคือ ฟังก์ชัน $last(n+1)$ ที่เก็บเฉพาะสมาชิก $n+1$ ตัวหลังสุดไว้

ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบสำเนาคือ ฟังก์ชันตัวปัจจุบัน $current$ ที่เก็บสมาชิกตัวปัจจุบันไว้

5.1.2.3.1 **การกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบตัวหน่วงการสำเนา**

ส่วนประกอบตัวหน่วงการสำเนาเป็นแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต และฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่อิสระจากประวัติด้วยตัวดำเนินการประกอบแบบลำดับ ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน $last(n+1)$ ที่ที่เก็บเฉพาะสมาชิก $n+1$ ตัวหลังสุดไว้

5.1.2.3.2 **การแปลงส่วนประกอบตัวหน่วงการสำเนาเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก**

ต่อไปนี้เป็นการแปลงส่วนประกอบตัวหน่วงการสำเนาเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก โดยแปลงส่วนประกอบตัวหน่วงการสำเนาผ่านทางเครื่องมัวร์แบบบัญญัติ จากนั้นจึงแปลงเครื่องมัวร์แบบบัญญัติที่ได้รับเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

1) **การแปลงส่วนประกอบตัวหน่วงการสำเนาเป็นเครื่องมัวร์แบบบัญญัติ**

ส่วนประกอบตัวหน่วงการสำเนาสามารถแปลงเป็นเครื่องมัวร์แบบบัญญัติได้ดังนี้

$$M_{dupDelay(n)} = (\mathcal{A}^*, \mathcal{A}, \mathcal{A}^*, \delta,\ \varphi_{dupDelayt(n)}, \langle\rangle)$$

โดยที่

ฟังก์ชันนำออกเดี่ยว $\varphi_{dupDelay}$ คือ

$$\varphi_{dupDelay(n)}(X) = \langle\rangle \tag{5.23}$$

$$\varphi_{dupDelay(n)}(Y) = \langle Y_{[|Y|-(n+1)]}, Y_{[|Y|-(n+1)]} \rangle \tag{5.24}$$

เมื่อ $|X| \le n$ และ $|Y| > n$

(5.23) หมายถึงไม่มีนำออกเมื่อกระแสนำเข้ามีขนาดน้อยกว่าหรือเท่ากับ n

(5.24) หมายถึงนำออกคือสมาชิกตัวที่ $n+1$ นับมาจากหลังสุดของกระแสนำเข้าเมื่อกระแสนำเข้ามีขนาดมากกว่า n

2) การแปลงเครื่องมัวร์แบบบัญญัติเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

เครื่องมัวร์แบบบัญญัติ $M_{dupDelay(n)}$ สามารถแปลงเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก $T_{dupDelay(n),\alpha}$ ได้ดังนี้

$$T_{dupDelay(n),\alpha} = (\mathcal{A}, \mathcal{A}^*, Q, S, R, s_0)$$

โดยที่

$$Q = \mathcal{A}^{\le n+1} \quad (5.25)$$

$$S = (\mathcal{A} \cup \{\phi\}) \times \mathcal{A}^{\le n+1} \times \mathcal{A} \quad (5.26)$$

$$R = \{(\alpha(X), \alpha(X \rhd x)) \mid X \in \mathcal{A}^*, x \in \mathcal{A}\} \quad (5.27)$$

$$s_0 = (\phi, last(n+1), \varphi_{dupDelay(n)}(\langle\rangle)) = (\phi, \langle\rangle, \langle\rangle) \quad (5.28)$$

เมื่อ $\alpha(\langle\rangle) = (\phi, last(n+1), \varphi_{dupDelay(n)}(\langle\rangle))$

$$\alpha(X \rhd x) = (x, last(n+1)(X \rhd x), \varphi_{dupDelay(n)}(X \rhd x))$$

(5.25) หมายถึง เซตของสถานะภายในคือ กระแส $\mathcal{A}^{\le n+1}$ ซึ่งได้รับจากการลดจำนวนสถานะที่ได้รับจากกระแสนำเข้าด้วยฟังก์ชันภาวะนามธรรมประวัติ $last(n+1)$

(5.26) หมายถึง ปริภูมิสถานะ S ที่เกิดจากการรวมเซตข้อมูลนำเข้า สถานะภายใน และเซตข้อมูลนำออกด้วยฟังก์ชันการแปลง α สามารถสร้างสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกภายใต้ความสัมพันธ์การจำลองสองทางกับสถานะของเครื่องมัวร์แบบบัญญัติได้ดังนี้

สร้างสถานะจากสถานะเริ่มต้นที่เป็นกระแสว่าง

$$\langle\rangle \leftrightarrow_H \{(\phi, \langle\rangle, \langle\rangle)\}$$

สร้างสถานะเมื่อ $|X \rhd x| < n+1$

$$X \rhd x \leftrightarrow_H \{(x, X \rhd x, \langle\rangle) \mid x \in \mathcal{A}, X \in \mathcal{A}^*\}$$

สร้างสถานะเมื่อ $|y \lhd X \rhd x| = n+1$

$$y \lhd X \rhd x \leftrightarrow_H \{(x, y \lhd X \rhd x, \langle y, y\rangle) \mid x, y \in \mathcal{A}, X \in \mathcal{A}^*\}$$

(5.27) หมายถึง ความสัมพันธ์การเปลี่ยนสถานะ R แสดงความสัมพันธ์ของสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออก

การเปลี่ยนสถานะจากสถานะเริ่มต้น

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

$$(\phi, \langle\rangle, \langle\rangle) \rightarrow (x, \langle x\rangle, \langle\rangle)$$
การเปลี่ยนสถานะ เมื่อ $|X \triangleright x| < n$

$$(x, X \triangleright x, \langle\rangle) \rightarrow (y, X \triangleright x \triangleright y, \langle\rangle)$$
การเปลี่ยนสถานะ เมื่อ $|z \triangleleft X \triangleright x| = n$

$$(x, z \triangleleft X \triangleright x, \langle\rangle) \rightarrow (y, X \triangleright x \triangleright y, \langle z, z\rangle)$$
การเปลี่ยนสถานะ เมื่อ $|z \triangleleft w \triangleleft X \triangleright x| = n+1$

$$(x, z \triangleleft w \triangleleft X \triangleright x, \langle z, z\rangle) \rightarrow (y, w \triangleleft X \triangleright x \triangleright y, \langle w, w\rangle)$$

เมื่อ $w, x, y, z \in \mathcal{A}$ และ $X \in \mathcal{A}^*$

(5.28) หมายถึง กำหนดสถานะเริ่มต้นของระบบเปลี่ยนสถานะนำเข้า/นำออก s_0 คือ $(\phi, \langle\rangle, \langle\rangle)$

จากระบบเปลี่ยนสถานะนำเข้า/นำออก $T_{dupDelay(n),\alpha}$ สามารถเขียนให้อยู่ในรูปตารางการเปลี่ยนสถานะ ดังตารางที่ 5.3

ตารางที่ 5.3 ตารางการเปลี่ยนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกของส่วนประกอบตัวหน่วงการสำเนา

State			*State′*		
input	*internal state*	*output*	*input′*	*internal state′*	*output′*
ϕ	$\langle\rangle$	$\langle\rangle$	x_1	$\langle x_1\rangle$	$\langle\rangle$
x_m	$\langle x_1, ..., x_m\rangle$	$\langle\rangle$	x_{m+1}	$\langle x_1, ..., x_{m+1}\rangle$	$\langle\rangle$
x_n	$\langle x_1, ..., x_n\rangle$	$\langle\rangle$	x_{n+1}	$\langle x_1, ..., x_{n+1}\rangle$	$\langle x_1, x_1\rangle$
x_{n+1}	$\langle x_1, ..., x_{n+1}\rangle$	$\langle x_1, x_1\rangle$	x_{n+2}	$\langle x_2, ..., x_{n+2}\rangle$	$\langle x_2, x_2\rangle$
$1 \le m < n$					

5.1.2.4 **ตัวอย่างส่วนประกอบตัวนับตัวเลข** (Number counter component)

ตัวอย่างที่ 5.4 ส่วนประกอบตัวนับตัวเลข

ส่วนประกอบตัวนับตัวเลข เป็นส่วนประกอบที่นับจำนวนสมาชิกของกระแสนำเข้าที่เป็นตัวเลขแล้วส่งผลลัพธ์เป็นนำออก โดยส่วนประกอบตัวนับตัวเลขเป็นแบบจำลองประกอบที่ประกอบด้วยส่วนประกอบตัวกรองตัวเลข *filtN* และส่วนประกอบตัวนับ *count* ด้วยตัวดำเนินการประกอบแบบลำดับ ดังแสดงในรูปที่ 5.7 นิยามดังนี้

$$countN : \mathbb{N} \times [\mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}] \rightarrow [\mathcal{A}^* \rightarrow \mathbb{N}^*]$$

โดยที่

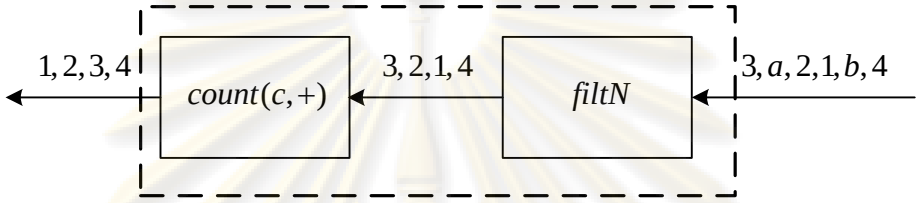
$$countN(c,+)(\langle\rangle) = \langle\rangle \tag{5.29}$$

$$countN(c,+)(x \lhd X) = \begin{cases} (c+1) \lhd countN(c+1,+)(X) & if\ (x \in \mathbb{N}) \\ countN(c,+)(X) & if\ (x \notin \mathbb{N}) \end{cases} \tag{5.30}$$

เมื่อ $c \in \mathbb{N}, x \in \mathcal{A}, X \in \mathcal{A}^*$ และกำหนดค่าเริ่มต้น $c = 0$

(5.29) หมายถึง กระแสนำเข้าว่างก่อกำเนิดกระแสนำออกว่าง

(5.30) หมายถึง นับจำนวนสมาชิกของกระแสนำเข้าที่เป็นตัวเลขแล้วส่งผลลัพธ์เป็นกระแสนำออก



รูปที่ 5.7 ส่วนประกอบตัวนับตัวเลข

ส่วนประกอบตัวกรองตัวเลข (Number filter component) เป็นส่วนประกอบที่กรองเฉพาะสมาชิกของกระแสนำเข้าที่เป็นตัวเลขส่งไปเป็นสมาชิกของกระแสนำออก ดังแสดงในรูปที่ 5.8 นิยามดังนี้

$$filtN : \mathcal{A}^* \to \mathbb{N}^*$$

โดยที่

$$filtN(\langle\rangle) = \langle\rangle \tag{5.31}$$

$$filtN(x \lhd X) = \begin{cases} x \lhd filtN(X) & if\ x \in \mathbb{N} \\ filtN(X) & if\ x \notin \mathbb{N} \end{cases} \tag{5.32}$$

เมื่อ $x \in \mathcal{A}, X \in \mathcal{A}^*$ และ $\mathbb{N} \subseteq \mathcal{A}$

(5.31) หมายถึง กระแสนำเข้าว่างก่อกำเนิดกระแสนำออกว่าง

(5.32) หมายถึง กรองเฉพาะสมาชิกของกระแสนำเข้าที่เป็นตัวเลขส่งไปเป็นสมาชิกของกระแสนำออก

รูปที่ 5.8 ส่วนประกอบตัวกรองตัวเลข

ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบตัวกรองตัวเลขคือ ฟังก์ชันตัวปัจจุบัน $filterU$ ที่เก็บเฉพาะสมาชิกตัวปัจจุบันไว้

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ส่วนประกอบตัวนับ (Counter component) เป็นส่วนประกอบที่นับจำนวนสมาชิกของกระแสนำเข้าแล้วเป็นนำออก ดังแสดงในรูปที่ 5.9 นิยามดังนี้

$$count : \mathbb{N} \times [\mathbb{N} \times \mathbb{N} \to \mathbb{N}] \to [\mathbb{N}^* \to \mathbb{N}^*]$$

โดยที่

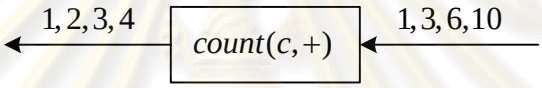
$$count(c,+)(\langle\rangle) = \langle\rangle \tag{5.33}$$

$$count(c,+)(x \lhd X) = (c+1) \lhd div(c+1,+)(X) \tag{5.34}$$

เมื่อ $c, x \in \mathbb{N}, X \in \mathbb{N}^*$ และกำหนดค่าเริ่มต้น $c = 0$

(5.33) หมายถึง นำเข้าว่างก่อกำเนิดกระแสนำออกเป็นค่าที่เก็บไว้ในตัวแปรภายใน

(5.34) หมายถึง นับจำนวนสมาชิกของกระแสนำเข้าแล้วส่งเป็นกระแสนำออก



รูปที่ 5.9 ส่วนประกอบตัวนับ

ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบตัวนับ คือ ฟังก์ชันลด $reduce(s,+)$ ที่ดำเนินการบวกสะสมกระแสนำเข้าไว้ในตัวแปรภายใน

5.1.2.4.1 **การกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบตัวนับตัวเลข**

ส่วนประกอบตัวนับตัวเลขเป็นแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบแบบมีคำร้องขอและฟังก์ชันการประมวลผลกระแสของส่วนประกอบสแกนด้วยตัวดำเนินการประกอบแบบลำดับ ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน $h_{countN}(c,+) = filterU; reduce(s,+)$ นิยามดังนี้

$$h_{countN} : \mathbb{N} \times [\mathbb{N} \times \mathbb{N} \to \mathbb{N}] \to [\mathcal{A}^* \to \mathbb{N}]$$

โดยที่

$$h_{countN}(c,+)(\langle\rangle) = c \tag{5.35}$$

$$h_{countN}(c,+)(x \lhd X) = \begin{cases} h_{countN}(c+1,+)(X) & if\ x \in \mathbb{N} \\ h_{countN}(c,+)(X) & if\ x \notin \mathbb{N} \end{cases} \tag{5.36}$$

เมื่อ $c, \mathbb{N}, x \in \mathcal{A}, X \in \mathcal{A}^*$ และกำหนดค่าเริ่มต้น $c = 0$

(5.35) หมายถึง นำเข้าว่างก่อกำเนิดนำออกที่เป็นค่าที่เก็บไว้ในสถานะภายใน

(5.36) หมายถึง นับจำนวนสมาชิกของกระแสนำเข้าที่เป็นตัวเลข

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

5.1.2.4.2 **การแปลงส่วนประกอบตัวนับตัวเลขเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก**

ต่อไปนี้เป็นการแปลงส่วนประกอบตัวนับตัวเลขเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก โดยแปลงส่วนประกอบตัวนับตัวเลขผ่านทางเครื่องมัวร์แบบบัญญัติ จากนั้นจึงแปลงเครื่องมัวร์แบบบัญญัติที่ได้รับเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

1) **การแปลงส่วนประกอบตัวนับตัวเลขเป็นเครื่องมัวร์แบบบัญญัติ**

ส่วนประกอบตัวนับตัวเลขสามารถแปลงเป็นเครื่องมัวร์แบบบัญญัติได้ดังนี้

$$M_{countN} = (\mathcal{A}^*, \mathcal{A}, \mathbb{N}^*, \delta, \varphi_{countN}, \langle\rangle)$$

โดยที่

ฟังก์ชันนำออกเดี่ยว $\varphi_{countN}(X) = \langle reduce'(s,+)(X)\rangle$ คือผลบวกสะสมของกระแสนำเข้า

2) **การแปลงเครื่องมัวร์แบบบัญญัติเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก**

เครื่องมัวร์แบบบัญญัติ M_{countN} สามารถแปลงเป็นระบบเปลี่ยนสถานะนำเข้า/ นำออก $T_{countN,\alpha}$ ได้ดังนี้

$$T_{summation,\alpha} = (\mathbb{N}, \mathbb{N}^*, Q, S, R, s_0)$$

โดยที่

$$Q = \mathbb{N} \tag{5.37}$$

$$S = (\mathbb{N} \cup \{\phi\}) \times \mathbb{N} \times \mathbb{N}^* \tag{5.38}$$

$$R = \{(\alpha(X), \alpha(X \triangleright x)) \mid X \in \mathbb{N}^*, x \in \mathbb{N}\} \tag{5.39}$$

$$s_0 = (\phi, h_{countN}(c,+)(\langle\rangle), \varphi_{countN}(\langle\rangle)) = (\phi, s, \langle s\rangle) \tag{5.40}$$

เมื่อ

$$\alpha(\langle\rangle) = (\phi, h_{countN}(c,+)(\langle\rangle), \varphi_{countN}(\langle\rangle))$$

$$\alpha(X \triangleright x) = (x, h_{countN}(c,+)(X \triangleright x), \varphi_{countN}(X \triangleright x))$$

(5.37) หมายถึง เซตของสถานะภายในคือ เซต $\mathbb{N}$ ซึ่งได้รับจากการลดจำนวนสถานะที่ได้รับจากกระแสนำเข้าด้วยฟังก์ชันภาวะนามธรรมประวัติ $h_{countN}(c,+)$

(5.38) หมายถึง ปริภูมิสถานะ S ที่เกิดจากการรวมเซตข้อมูลนำเข้า สถานะภายใน และเซตข้อมูลนำออก ด้วยฟังก์ชันการแปลง α สามารถสร้างสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกภายใต้ความสัมพันธ์การจำลองสองทางกับสถานะของเครื่องมัวร์แบบบัญญัติได้ดังนี้

สร้างสถานะจากสถานะเริ่มต้นที่เป็นกระแสว่าง

$$\langle\rangle \leftrightarrow_H \{(\phi, s, \langle\rangle)\}$$

สร้างสถานะจากสถานะภายในไม่เป็นกระแสว่าง

$$X \triangleright x \leftrightarrow_H \{(x, h_{countN}(c,+)(X \triangleright x), \langle h_{countN}(c,+)(X \triangleright x)\rangle) \mid s, x \in \mathbb{N}, X \in \mathbb{N}^*\}$$

(5.39) หมายถึง ความสัมพันธ์การเปลี่ยนสถานะ R แสดงความสัมพันธ์ของสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออก

การเปลี่ยนสถานะจากสถานะเริ่มต้น เมื่อได้รับนำเข้า $b \in \mathcal{A} \backslash \mathbb{N}$

$$(\phi, s, \langle\rangle) \rightarrow (b, s, \langle\rangle)$$

การเปลี่ยนสถานะจากสถานะเริ่มต้น เมื่อได้รับนำเข้า $x \in \mathbb{N}$

$$(\phi, s, \langle s\rangle) \rightarrow (x, s+x, \langle s+x\rangle)$$

การเปลี่ยนสถานะจากสถานะที่ได้รับกระแสนำเข้าก่อนหน้าคือ $X \triangleleft a$ และได้รับนำเข้าปัจจุบันคือ $b \in \mathcal{A} \backslash \mathbb{N}$

$$(a, q, \langle\rangle) \rightarrow (b, q, \langle\rangle)$$

เมื่อ $a, b \in \mathcal{A} \setminus \mathbb{N}, X \in \mathcal{A}^*$ และ $q = h_{countN}(c,+)(X)$

การเปลี่ยนสถานะจากสถานะที่ได้รับกระแสนำเข้าก่อนหน้าคือ $X \triangleleft a$ และได้รับนำเข้าปัจจุบันคือ $y \in \mathbb{N}$

$$(a, q, \langle\rangle) \rightarrow (y, q+y, \langle q+y\rangle)$$

เมื่อ $a \in \mathcal{A} \setminus \mathbb{N}, y \in \mathbb{N}, X \in \mathcal{A}^*$ และ $q = h_{countN}(c,+)(X)$

การเปลี่ยนสถานะจากสถานะที่ได้รับกระแสนำเข้าก่อนหน้าคือ $X \triangleleft x$ และได้รับนำเข้าปัจจุบันคือ $b \in \mathcal{A} \backslash \mathbb{N}$

$$(x, q+x, \langle q+x\rangle) \rightarrow (b, q+x, \langle\rangle)$$

เมื่อ $b \in \mathcal{A} \setminus \mathbb{N}, x \in \mathbb{N}, X \in \mathcal{A}^*$ และ $q = h_{countN}(c,+)(X)$

การเปลี่ยนสถานะจากสถานะที่ได้รับกระแสนำเข้าก่อนหน้าคือ $X \triangleleft x$ และได้รับนำเข้าปัจจุบันคือ $y \in \mathbb{N}$

$$(x, q+x, \langle q+x\rangle) \rightarrow (y, q+x+y, \langle q+x+y\rangle)$$

เมื่อ $x, y \in \mathbb{N}, X \in \mathcal{A}^*$ และ $q = h_{countN}(c,+)(X)$

(5.40) หมายถึง กำหนดสถานะเริ่มต้นของระบบเปลี่ยนสถานะนำเข้า/นำออก s_0 คือ $(\phi, s, \langle\rangle)$

จากระบบเปลี่ยนสถานะนำเข้า/นำออก $T_{countN,\alpha}$ สามารถเขียนให้อยู่ในรูปตารางการเปลี่ยนสถานะ ดังตารางที่ 5.4

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ตารางที่ 5.4 ตารางการเปลี่ยนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกของส่วนประกอบตัวนับตัวเลข

State			*State′*		
input	*internal state*	*output*	*input′*	*internal state′*	*output′*
ϕ	s	$\langle\rangle$	b	s	$\langle\rangle$
ϕ	s	$\langle\rangle$	y	$s+y$	$\langle s+y\rangle$
a	q	$\langle\rangle$	a	q	$\langle\rangle$
a	q	$\langle\rangle$	y	$q+y$	$\langle q+y\rangle$
x	$q+x$	$\langle q+x\rangle$	a	$q+x$	$\langle\rangle$
x	$q+x$	$\langle q+x\rangle$	y	$q+x+y$	$\langle q+x+y\rangle$
$a,b\in\mathcal{A}\setminus\mathbb{N}, x,y\in\mathbb{N}, X\in\mathcal{A}^*, q=h_{countN}(c,+)(X)$					

5.1.2.5 **ตัวอย่างส่วนประกอบตัวหน่วงผลบวก** (Summation delay component)

ตัวอย่างที่ 5.5 ส่วนประกอบตัวหน่วงผลบวก

ส่วนประกอบตัวหน่วงผลบวก เป็นส่วนประกอบที่คำนวณผลบวกจากสมาชิกของกระแสนำเข้าที่มีความยาวเกินที่กำหนด โดยนำส่วนประกอบเรจิสเตอร์การเลื่อน $shift(n)$ (ดังตัวอย่างที่ 2.2) และส่วนประกอบผลรวม $sum(s,+)$ (ดังแสดงในตัวอย่างที่ 5.1) มาประกอบกันด้วยตัวดำเนินการประกอบแบบลำดับ เมื่อกำหนดค่าคงที่ด้วย 3 สามารถแสดงดังแสดงในรูปที่ 5.10 นิยามดังนี้

$$sumDelay:\mathbb{N}\setminus 0\to\mathbb{N}\times[\mathbb{N}\times\mathbb{N}\to\mathbb{N}]\to[\mathbb{N}^*\to\mathbb{N}^*]$$

โดยที่

$$sumDelay(n)(s,+)(\langle\rangle)=\langle\rangle \tag{5.41}$$

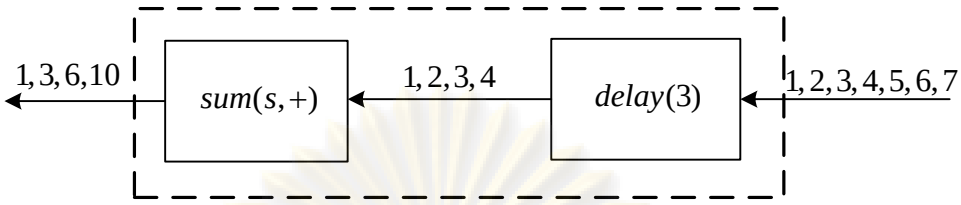
$$sumDelay(n)(s,+)(x\lhd X)=\begin{cases}(s+x)\lhd sumDelay(n)(s+x,+)(X) & if\ (|\ x\lhd X\ |>n)\\ sumDelay(n)(s,+)(X) & if\ (|\ x\lhd X\ |\le n)\end{cases} \tag{5.42}$$

เมื่อ $s,x\in\mathbb{N}, X\in\mathbb{N}^*$ และ $n\in(\mathbb{N}\setminus 0)$

(5.41) หมายถึง กระแสนำเข้าว่างก่อกำเนิดกระแสนำออกว่าง

(5.42) หมายถึง ดำเนินการบวกสมาชิกของกระแสนำเข้าแล้วเก็บสะสมไว้ในตัวแปรภายในและปล่อยเป็นกระแสนำออกเมื่อกระแสนำเข้ามีสมาชิกเกินจำนวนที่กำหนด

เมื่อ $c, s, x \in \mathbb{N}, X \in \mathbb{N}^*$ และกำหนดค่าเริ่มต้น $s = 0$



รูปที่ 5.10 ส่วนประกอบหน่วงผลบวก

ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบเรจิสเตอร์การเลื่อนคือ ฟังก์ชัน $last(n+1)$ ที่เก็บเฉพาะสมาชิก $n+1$ ตัวหลังสุดไว้

ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบผลรวมคือ ฟังก์ชัน $reduce'(s,+)$ ที่บวกสะสมสมาชิกของกระแสนำเข้า

5.1.2.5.1 **การกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบตัวหน่วงผลบวก**

ส่วนประกอบตัวหน่วงผลบวกเป็นแบบจำลองประกอบที่ได้รับจากการประกอบฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่มีหน่วยความจำพจน์สิ้นแบบมีขอบเขตและฟังก์ชันการประมวลผลกระแสของส่วนประกอบคงที่ ด้วยตัวดำเนินการประกอบแบบลำดับ ดังนั้นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบนี้คือ ฟังก์ชัน $h_{sumDelay}(n+1)(s,+) = last(n+1); reduce(s,+)$ นิยามดังนี้

$$h_{sumDelay} : \mathbb{N} \setminus 0 \to \mathbb{N} \times [\mathbb{N} \times \mathbb{N} \to \mathbb{N}] \to [\mathbb{N}^* \to \mathbb{N}]$$

โดยที่

$$h_{sumDelay}(n+1)(s,+)(\langle\rangle) = s \tag{5.43}$$

$$h_{sumDelay}(n+1)(s,+)(x \lhd X) = \begin{cases} h_{sumDelay}(n+1)(s+x,+)(X) & if\ (|\ x \lhd X\ | > n) \\ h_{sumDelay}(n+1)(s,+)(X) & if\ (|\ x \lhd X\ | \leq n) \end{cases} \tag{5.44}$$

เมื่อ $s, x \in \mathbb{N}, n \in \mathbb{N} \setminus 0, X \in \mathbb{N}^*$ และกำหนดค่าเริ่มต้น $s = 0$

(5.43) หมายถึง นำเข้าว่างก่อกำเนิดนำออกที่เป็นค่าที่เก็บไว้ในตัวแปรภายใน s

(5.44) หมายถึง ดำเนินการบวกสมาชิกของกระแสนำเข้าแล้วเก็บสะสมไว้ในตัวแปรภายใน s เมื่อจำนวนสมาชิกของกระแสนำเข้ามากกว่าที่กำหนด

5.1.2.5.2 การแปลงส่วนประกอบตัวหน่วงผลบวกเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

ต่อไปนี้เป็นการแปลงส่วนประกอบตัวหน่วงผลบวกเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก โดยแปลงส่วนประกอบตัวหน่วงผลบวกผ่านทางเครื่องมัวร์แบบบัญญัติ จากนั้นจึงแปลงเครื่องมัวร์แบบบัญญัติที่ได้รับเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

1) การแปลงส่วนประกอบตัวหน่วงผลบวกเป็นเครื่องมัวร์แบบบัญญัติ

ส่วนประกอบตัวหน่วงผลบวกสามารถแปลงเป็นเครื่องมัวร์แบบบัญญัติได้ดังนี้

$$M_{sumDelay(n)(s,+)} = (\mathbb{N}^*, \mathbb{N}, \mathbb{N}^*, \delta,\ \varphi_{sumDelay(n)(s,+)}, \langle\rangle)$$

โดยที่

ฟังก์ชันนำออกเดี่ยว $\varphi_{sumDelay(n)(s,+)}$ คือ

$$\varphi_{sumDelay(n)(s,+)}(\langle\rangle) = \langle\rangle \quad (5.45)$$

$$\varphi_{sumDelay(n)(s,+)}(X) = \langle h_{sumDelay}(n+1)(s,+)(X)\rangle \quad (5.46)$$

เมื่อ $|X| \le n$ และ $|Y| > n$

(5.45) หมายถึงไม่มีนำออกเมื่อกระแสนำเข้ามีขนาดน้อยกว่าหรือเท่ากับ n

(5.46) หมายถึงนำออกคือสมาชิกตัวที่ $n+1$ นับมาจากหลังสุดของกระแสนำเข้าเมื่อกระแสนำเข้ามีขนาดมากกว่า n

2) การแปลงเครื่องมัวร์แบบบัญญัติเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก

เครื่องมัวร์แบบบัญญัติ $M_{sumDelay(n)(s,+)}$ สามารถแปลงเป็นระบบเปลี่ยนสถานะนำเข้า/นำออก $T_{sumDelay(n)(s,+)}$ ได้ดังนี้

$$T_{sumDelay(n)(s,+)} = (\mathbb{N}, \mathbb{N}^*, Q, S, R, s_0)$$

โดยที่

$$Q = \mathbb{N} \quad (5.47)$$

$$S = (\mathbb{N} \cup \{\phi\}) \times \mathbb{N} \times \mathbb{N}^* \quad (5.48)$$

$$R = \{(\alpha(X), \alpha(X \triangleright x)) \mid X \in \mathbb{N}^*, x \in \mathbb{N}\} \quad (5.49)$$

$$s_0 = (\phi, h_{sumDelay}(n+1)(s,+), \varphi_{sumDelay(n)(s,+)}(\langle\rangle)) = (\phi, \langle\rangle, \langle\rangle) \quad (5.50)$$

เมื่อ $\alpha(\langle\rangle) = (\phi, h_{sumDelay}(n+1)(s,+), \varphi_{sumDelay(n)(s,+)}(\langle\rangle))$

$$\alpha(X \triangleright x) = (x, h_{sumDelay}(n+1)(s,+)(X \triangleright x), \varphi_{sumDelay(n)(s,+)}(X \triangleright x))$$

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

(5.47) หมายถึง เซตของสถานะภายในคือ เซต $\mathbb{N}$ ซึ่งได้รับจากการลดจำนวนสถานะที่ได้รับจากกระแสนำเข้าด้วยฟังก์ชันภาวะนามธรรมประวัติ $h_{sumDelay}(n+1)(s,+)$

(5.48) หมายถึง ปริภูมิสถานะ S ที่เกิดจากการรวมเซตข้อมูลนำเข้า สถานะภายใน และเซตข้อมูลนำออกด้วยฟังก์ชันการแปลง α สามารถสร้างสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกภายใต้ความสัมพันธ์การจำลองสองทางกับสถานะของเครื่องมัวร์แบบบัญญัติได้ดังนี้

สร้างสถานะจากสถานะเริ่มต้นที่เป็นกระแสว่าง

$$\langle\rangle \leftrightarrow_H \{(\phi, \langle\rangle, \langle\rangle)\}$$

สร้างสถานะจากสถานะที่ไม่ใช่สถานะเริ่มต้น

$$X \triangleright x \leftrightarrow_H \{(x, h_{sumDelay}(n+1)(s,+)(X \triangleright x), \langle h_{sumDelay}(n+1)(s,+)(X \triangleright x)\rangle)$$

$$| n, s, x \in \mathbb{N}, X \in \mathbb{N}^*\}$$

(5.49) หมายถึง ความสัมพันธ์การเปลี่ยนสถานะ R แสดงความสัมพันธ์ของสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออก

การเปลี่ยนสถานะจากสถานะเริ่มต้น

$$(\phi, \langle\rangle, \langle\rangle) \rightarrow (x, \langle x\rangle, \langle\rangle)$$

การเปลี่ยนจากสถานะที่ไม่ใช่สถานะเริ่มต้น

$$(x, h_{sumDelay}(n+1)(s,+)(X \triangleright x), \langle h_{sumDelay}(n+1)(s,+)(X \triangleright x)\rangle) \rightarrow$$

$$(y, h_{sumDelay}(n+1)(s,+)(X \triangleright x \triangleright y), \langle h_{sumDelay}(n+1)(s,+)(X \triangleright x \triangleright y)\rangle)$$

เมื่อ $n, s, x \in \mathbb{N}$ และ $X \in \mathbb{N}^*$

(5.50) หมายถึง กำหนดสถานะเริ่มต้นของระบบเปลี่ยนสถานะนำเข้า/นำออก s_0 คือ $(\phi, \langle\rangle, \langle\rangle)$

จากระบบเปลี่ยนสถานะนำเข้า/นำออก $T_{sumDelay(n)(s,+),\alpha}$ สามารถเขียนให้อยู่ในรูปตารางการเปลี่ยนสถานะ ดังตารางที่ 5.5

ตารางที่ 5.5 ตารางการเปลี่ยนสถานะของระบบเปลี่ยนสถานะนำเข้า/นำออกของส่วนประกอบตัวหน่วงผลบวก

State			*State′*		
input	*internal state*	*output*	*input′*	*internal state′*	*output′*
ϕ	s	$\langle\rangle$	x	s	$\langle\rangle$
x	$h_{sumDelay}$ $(n+1)(s,+)$ $(X \triangleright x)$	$\langle h_{sumDelay}$ $(n+1)(s,+)$ $(X \triangleright x)\rangle$	y	$h_{sumDelay}$ $(n+1)(s,+)$ $(X \triangleright x \triangleright y)$	$\langle h_{sumDelay}$ $(n+1)(s,+)$ $(X \triangleright x \triangleright y)\rangle$
$n, s, x, y \in \mathbb{N}, X \in \mathbb{N}^*$					

5.1.3 ความสัมพันธ์ของฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบกับแบบจำลองบนพื้นฐานของกระแสเฉพาะรายที่นำมาประกอบ

ส่วนประกอบค่าเฉลี่ย $ave(s,+)(c,+)$ เป็นแบบจำลองประกอบที่ได้รับจากการประกอบส่วนประกอบผลรวม $sum(s,+)$ และส่วนประกอบผลหาร $div(c,+)$ ด้วยตัวดำเนินการประกอบแบบลำดับ ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบ $ave(s,+)(c,+)$ คือ ฟังก์ชัน $reduce'''(s,+)(c,+)$ ซึ่งได้รับจากการประยุกต์ใช้ฟังก์ชัน $reduce'(s,+)$ ซึ่งเป็นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบผลรวม $sum(s,+)$ และฟังก์ชัน $reduce''(c,+)$ ซึ่งเป็นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบผลหาร $div(c,+)$ ร่วมกัน

ส่วนประกอบเรจิสเตอร์การเลื่อนคู่ $shift'(n')$ เป็นแบบจำลองประกอบที่ได้รับจากการประกอบส่วนประกอบเรจิสเตอร์การเลื่อน $shift(n)$ สองส่วนประกอบด้วยตัวดำเนินการประกอบบบลำดับ ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบเรจิสเตอร์การเลื่อนคู่ $shift'(n')$ คือ ฟังก์ชัน $last(n'+1)$ ซึ่งเป็นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบเรจิสเตอร์การเลื่อน $last(n+1)$

ส่วนประกอบตัวหน่วงการสำเนา $dupDelay(n)$ เป็นแบบจำลองประกอบที่ได้รับจากการประกอบส่วนประกอบเรจิสเตอร์การเลื่อน $shift(n)$ และส่วนประกอบสำเนา $duplicate$ ด้วยตัวดำเนินการแบบลำดับ ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบตัวหน่วงการสำเนา $dupDelay(n)$ คือ ฟังก์ชัน $last(n+1)$ ซึ่งเป็นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบเรจิสเตอร์การเลื่อน

ส่วนประกอบตัวนับตัวเลข $countN$ เป็นแบบจำลองประกอบที่ได้รับจากการประกอบส่วนประกอบตัวกรอง $filtN$ และส่วนประกอบตัวนับ $count$ ด้วยตัวดำเนินการประกอบแบบลำดับ ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบตัวนับตัวเลข $countN$ คือ ฟังก์ชัน $h_{countN}(c,+)$ ซึ่งเป็นฟังก์ชันที่ได้รับจากการประกอบฟังก์ชัน $filterU$ ซึ่งเป็นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบตัวกรอง และฟังก์ชัน $reduce(s,+)$ ซึ่งเป็นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบตัวนับ $count$ ด้วยตัวดำเนินการประกอบแบบลำดับ

ส่วนประกอบตัวหน่วงผลบวก $sumDelay(n)(s,+)$ เป็นแบบจำลองประกอบที่ได้รับจากการประกอบส่วนประกอบเรจิสเตอร์การเลื่อน $shift(n)$ และส่วนประกอบผลรวม $sum(s,+)$ ด้วยตัวดำเนินการประกอบแบบลำดับ ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบตัวหน่วงผลบวก $sumDelay(n)(s,+)$ คือ ฟังก์ชัน $h_{sumDelay(nH)(s,+)}$ ที่ซึ่ง

ได้รับจากการประกอบฟังก์ชัน $last(n+1)$ ซึ่งเป็นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบเรจิสเตอร์การเลื่อน และฟังก์ชัน $reduce(s,+)$ ซึ่งเป็นฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของส่วนประกอบผลรวม $(s,+)$ ด้วยตัวดำเนินการประกอบแบบลำดับ

จากผลลัพธ์ที่ได้แสดงให้เห็นว่าฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบมีความสัมพันธ์กับฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองบนพื้นฐานของกระแสที่นำมาประกอบ แต่ความสัมพันธ์นี้ไม่ได้มีรูปแบบที่แน่นอน ดังนั้นจึงไม่สามารถหาฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบจากฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองบนพื้นฐานของกระแสเฉพาะรายที่นำมาประกอบโดยตรง แต่สามารถนำมาพิจารณาประกอบในการตัดสินใจเลือกได้

5.2 การทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบด้วยการตรวจสอบแบบจำลอง

จากการกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่เหมาะสมดังที่ได้กล่าวไปแล้วในตอนที่ 5.1 ทำให้สามารถสร้างระบบเปลี่ยนสถานะนำเข้า/นำออกจากแบบจำลองประกอบ ซึ่งระบบเปลี่ยนสถานะนำเข้า/นำออกนี้เป็นแบบจำลองบนพื้นฐานของสถานะที่เป็นองค์ประกอบหนึ่งในการทวนสอบสมบัติของแบบจำลองด้วยการตรวจสอบแบบจำลอง แต่ยังขาดองค์ประกอบที่สำคัญอีกอย่างหนึ่ง นั่นก็คือการกำหนดสมบัตินำเข้า/นำออกที่คาดหมายของแบบจำลองประกอบ และวิธีการทวนสอบสมบัตินำเข้า/นำออกด้วยการตรวจสอบแบบจำลอง ต่อไปนี้เป็นการนำเสนอแนวทางในการกำหนดการกำหนดสมบัตินำเข้า/นำออกที่คาดหมายของแบบจำลองประกอบ และนำเสนอวิธีการทวนสอบสมบัตินำเข้า/นำออกด้วยการตรวจสอบแบบจำลอง

5.2.1 วิธีการในการกำหนดสมบัตินำเข้า/นำออกของแบบจำลองประกอบ

เนื่องจากแบบจำลองประกอบได้รับพฤติกรรมนำเข้า/นำออกจากการประกอบแบบจำลองบนพื้นฐานของกระแสเฉพาะรายด้วยตัวดำเนินการประกอบแบบลำดับ ดังนั้นพฤติกรรมนำเข้าเกิดจากส่วนประกอบแรก และพฤติกรรมนำออกเกิดจากการประกอบกันของพฤติกรรมนำออกของส่วนประกอบแรกและพฤติกรรมนำออกของส่วนประกอบที่สอง นั่นคือสามารถนำเอาสมบัตินำเข้า/นำออกที่คาดหมายของส่วนประกอบแรกมากำหนดนำออกใหม่ด้วยการประยุกต์ฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่สองกับนำออกของฟังก์ชันการประมวลผลของส่วนประกอบแรก โดยที่นำเข้าคงเดิม และสามารถนำเอาสมบัตินำเข้า/นำออกที่คาดหมายของส่วนประกอบที่สองมากำหนดนำเข้าใหม่ด้วยการประยุกต์ฟังก์ชันผกผันของฟังก์ชัน

การประมวลผลกระแส (ถ้ามี) ของส่วนประกอบแรกกับนำเข้าของส่วนประกอบที่สองโดยที่นำออกคงเดิม

ต่อไปนี้เป็นการแสดงวิธีการกำหนดสมบัตินำเข้า/นำออกที่คาดหมายของแบบจำลองประกอบจากสมบัตินำเข้า/นำออกของแบบจำลองบนพื้นฐานของกระแสเฉพาะราย

กำหนดให้ฟังก์ชันการประมวลผลกระแสของส่วนประกอบแรกคือ $f:\mathcal{A}^* \to \mathcal{B}^*$ และฟังก์ชันการประมวลผลกระแสของส่วนประกอบที่สองคือ $g:\mathcal{B}^* \to \mathcal{C}^*$

นำเข้า/นำออกของสมบัตินำเข้า/นำออกที่คาดหมายของส่วนประกอบแรกคือ $x/\langle f(x)\rangle$

และนำเข้า/นำออกของสมบัตินำเข้า/นำออกที่คาดหมายของส่วนประกอบที่สองคือ $y/\langle g(y)\rangle$

จากฟังก์ชันการประมวลผลกระแสที่กำหนด สามารถสร้างแบบจำลองประกอบจากการประกอบฟังก์ชันการประมวลผลกระแส f และ g ด้วยตัวดำเนินการประกอบแบบลำดับได้ดังนี้คือ

$$(f;g):\mathcal{A}^* \to \mathcal{C}^*$$

โดยที่

$$(f;g)(x) = g(f(x))$$

สมบัตินำเข้า/นำออกที่คาดหมายของแบบจำลองประกอบที่สามารถสร้างมาจากสมบัตินำเข้า/นำออกที่คาดหมายเดิมของส่วนประกอบแรกโดยการเปลี่ยนนำออกคือ $x/\langle g(f(x))\rangle$ นั่นคือ นำฟังก์ชัน g มาประยุกต์กับนำออกเดิมของส่วนประกอบแรกโดยที่นำเข้าคงเดิม

สมบัตินำเข้า/นำออกที่คาดหมายของแบบจำลองประกอบที่สามารถสร้างมาจากสมบัติที่คาดหมายเดิมของส่วนประกอบที่สองโดยการเปลี่ยนนำเข้าคือ $f^{-1}(y)/\langle g(y)\rangle$ ถ้า f^{-1}มี นั่นคือ นำฟังก์ชัน f^{-1} มาประยุกต์กับนำเข้าเดิมของส่วนประกอบที่สองโดยที่นำออกคงเดิม

นอกจากสมบัตินำเข้า/นำออกที่คาดหมายของแบบจำลองประกอบจะสามารถสร้างมาจากสมบัตินำเข้า/นำออกที่คาดหมายเดิมของแบบจำลองประกอบบนพื้นฐานของกระแสเฉพาะรายที่นำมาประกอบด้วยตัวดำเนินการแบบลำดับแล้วยังสามารถกำหนดพฤติกรรมนำเข้า/นำออกที่คาดหมายเพิ่มเติมได้ตามความเหมาะสม

5.2.2 การทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบด้วยการตรวจสอบแบบจำลอง

ในการทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบด้วยการตรวจสอบแบบจำลองสามารถทำได้ โดยนำระบบเปลี่ยนสถานะนำเข้า/นำออกที่ได้รับมาเขียนเป็นแบบจำลองและนำสมบัตินำเข้า/นำออกที่คาดหมายของแบบจำลองประกอบมาเขียนเป็นข้อกำหนดด้วยภาษาเอสเอ็มวี จากนั้นนำไปทวนสอบด้วยตัวตรวจสอบเอ็มยูเอสเอ็มวี

5.2.3. ตัวอย่างการกำหนดสมบัตินำเข้า/นำออกของแบบจำลองประกอบ และวิธีการทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบด้วยการตรวจสอบแบบจำลอง

จากตัวอย่างระบบเปลี่ยนสถานะนำเข้า/นำออกที่ได้รับจากแบบจำลองประกอบที่นำเสนอในตอนที่ 5.1 พบว่าระบบเปลี่ยนสถานะนำเข้า/นำออกของส่วนประกอบเรจิสเตอร์การเลื่อนคู่ และส่วนประกอบตัวหน่วงการสำเนาเท่านั้นที่มีความเหมาะสมที่จะทวนสอบสมบัตินำเข้า/นำออกด้วยการตรวจสอบแบบจำลอง ต่อไปนี้เป็นการแสดงการทวนสอบสมบัตินำเข้า/นำออกของส่วนประกอบเรจิสเตอร์การเลื่อนคู่ด้วยการตรวจสอบแบบจำลอง โดยกำหนดสมบัตินำเข้า/นำออกที่คาดหมายให้อยู่ในรูปตรรกแอลทีแอล แล้วสร้างแนวทางในการกำหนดสมบัตินำเข้า/นำออกที่คาดหมายของแบบจำลองประกอบจากสมบัตินำเข้า/นำออกที่คาดหมายของแบบจำลองบนพื้นฐานของกระแสเฉพาะรายเดิม และทวนสอบสมบัตินำเข้า/นำออกของส่วนประกอบเรจิสเตอร์การเลื่อนคู่ ด้วยการตรวจสอบแบบจำลองตามวิธีที่กำหนด

ส่วนประกอบตัวหน่วงการสำเนา $dupDelay(n)$ ซึ่งเกิดจากการประกอบส่วนประกอบเรจิสเตอร์การเลื่อน $shift(n)$ และส่วนประกอบสำเนา $duplicate$ ด้วยตัวดำเนินการประกอบแบบลำดับ สามารถแสดงการกำหนดสมบัตินำเข้า/นำออก และทวนสอบสมบัตินำเข้า/นำออกด้วยการตรวจสอบแบบจำลองได้ดังนี้

5.2.3.1 การกำหนดสมบัตินำเข้า/นำออก และการทวนสอบสมบัตินำเข้า/นำออกของส่วนประกอบเรจิสเตอร์การเลื่อนด้วยการตรวจสอบแบบจำลอง

สมบัตินำเข้า/นำออกที่คาดหมายของส่วนประกอบเรจิสเตอร์การเลื่อน $shipt(n)$

$$\mathrm{G}((output = \langle\rangle)\,\mathrm{U}(inputLength > n)) \tag{5.51}$$

$$\mathrm{G}((input = x) \rightarrow \mathrm{X}^{n}(output = \langle g(x)\rangle)) \tag{5.52}$$

$$\begin{aligned}&\mathrm{G}(((input = x_1) \wedge \mathrm{X}(input = x_2)) \rightarrow \\ &\mathrm{X}^{n}((output = \langle g(x_1)\rangle) \wedge \mathrm{X}(output = \langle g(x_2)\rangle)))\end{aligned} \tag{5.53}$$

เมื่อ $inputLength$ แทนความยาวของกระแสที่กำหนด

(5.51) หมายถึง นำออกจากเรจิสเตอร์การเลื่อนคือ กระแสว่างจนกระทั่งจำนวนสมาชิกของกระแสนำเข้ามากกกว่าจำนวนที่กำหนด

(5.52) หมายถึง ถ้ากระแสนำเข้าปัจจุบันคือ x แล้วนำออกคือ $\langle x \rangle$ เมื่อนำเข้าเพิ่ม n ตัว

(5.53) หมายถึง ถ้านำเข้าปัจจุบันคือ x_1 และนำเข้าถัดไปคือ x_2 แล้วนำออกคือ $\langle x_1 \rangle$ เมื่อรับนำเข้าเพิ่ม n ตัว และ $\langle x_2 \rangle$ เมื่อรับนำเข้าตัวถัดไป

เมื่อแทนค่าคงที่ของส่วนประกอบเรจิสเตอร์การเลื่อนด้วย 4 และเซตข้อมูลนำเข้าทวิภาคแล้วนำระบบเปลี่ยนสถานะนำเข้า/นำออกไปทำให้เกิดผลด้วยภาษาเอสเอ็มวี (รายละเอียดอยู่ในภาคผนวก ข) แล้วนำไปทวนสอบด้วยตัวตรวจสอบแบบจำลองเอ็นยูเอสเอ็มวี ผลปรากฏว่าสมบัตินำเข้า/นำออกที่คาดหมายสอดคล้องกับแบบจำลองเรจิสเตอร์การเลื่อน

5.2.3.2 การกำหนดสมบัตินำเข้า/นำออก และการทวนสอบสมบัตินำเข้า/นำออกของส่วนประกอบสำเนาด้วยการตรวจสอบแบบจำลอง

สมบัตินำเข้า/นำออกที่คาดหมายของส่วนประกอบสำเนา $duplicate$

$$\mathrm{G}((input = \langle\rangle) \rightarrow (output = \langle\rangle)) \tag{5.54}$$

$$\mathrm{G}((input = x) \rightarrow (output = \langle x, x \rangle)) \tag{5.55}$$

(5.54) หมายถึง ถ้านำเข้าปัจจุบันคือกระแสว่าง แล้วนำออกปัจจุบันคือกระแสว่าง

(5.55) หมายถึง ถ้านำเข้าปัจจุบันคือ x, x แล้วออกปัจจุบันคือ $\langle x, x \rangle$

เมื่อแทนเซตข้อมูลทวิภาคแล้วนำระบบเปลี่ยนสถานะนำเข้า/นำออกของส่วนประกอบสำเนา $duplicate$ ไปทำให้เกิดผลด้วยภาษาเอสเอ็มวี (รายละเอียดอยู่ในภาคผนวก ข) แล้วนำไปทวนสอบด้วยตัวตรวจสอบเอ็นยูเอสเอ็มวี ผลปรากฏว่าสมบัตินำเข้า/นำออกที่คาดหมายสอดคล้องกับแบบจำลอง

5.2.3.3 การกำหนดสมบัตินำเข้า/นำออก และการทวนสอบสมบัตินำเข้า/นำออกของส่วนประกอบตัวหน่วงการสำเนาด้วยการตรวจสอบแบบจำลอง

สมบัตินำเข้า/นำออกที่คาดหมายของส่วนประกอบตัวหน่วงสำเนา $dupDelay$ สามารถสร้างจากสมบัตินำเข้า/นำออกที่คาดหมายของส่วนประกอบเรจิสเตอร์การเลื่อน $shift(n)$ และส่วนประกอบสำเนา $duplicate$ ดังนี้

$$\mathrm{G}((output = \langle\rangle)\,\mathrm{U}(inputLength > n)) \tag{5.56}$$

$$\mathrm{G}((input = x) \rightarrow \mathrm{X}^n(output = \langle x, x \rangle)) \tag{5.57}$$

$$\begin{aligned} &\mathrm{G}(((input = x_1) \wedge \mathrm{X}(input = x_2)) \rightarrow \\ &\mathrm{X}^n((output = \langle x_1, x_1 \rangle) \wedge \mathrm{X}(output = \langle x_2, x_2 \rangle))) \end{aligned} \tag{5.58}$$

(5.56) หมายถึง นำออกของส่วนประกอบตัวหน่วงการสำเนาคือ กระแสว่างจนกระทั่งสมาชิกของกระแสนำเข้ามากกว่าจำนวนที่กำหนด เกิดจากการปรับนำออกด้วย $duplicate(\langle\rangle) = \langle\rangle$

(5.57) หมายถึง ถ้านำเข้าปัจจุบันคือ x แล้วนำออคือ $\langle x, x\rangle$ เมื่อรับนำเข้าเพิ่ม n ตัว เกิดจากการปรับ $duplicate(\langle x\rangle) = \langle x, x\rangle$

(5.58) หมายถึง ถ้านำเข้าปัจจุบันคือ x_1 และนำเข้าถัดไปคือ x_2 แล้วนำออกเมื่อรับนำเข้าเพิ่ม n ตัวคือ $\langle x_1, x_1\rangle$ และนำออกถัดไปคือ $\langle x_2, x_2\rangle$

เนื่องจากฟังก์ชันการประมวลผลกระแส $shift(n)$ ไม่เป็นฟังก์ชันหนึ่งต่อหนึ่ง ดังนั้นจึงไม่มีฟังก์ชันผกผัน $shift^{-1}(n)$ จึงไม่สามารถกำหนดสมบัตินำเข้า/นำออกที่คาดหมายของส่วนประกอบตัวหน่วงการสำเนา $dupDelay(n)$ จากสมบัตินำเข้า/นำออกที่คาดหมายของส่วนประกอบสำเนา $duplicate$ ได้

เมื่อแทนค่าคงที่ของส่วนประกอบตัวหน่วงการสำเนาด้วย 4 และเซตข้อมูลนำเข้าทวิภาคแล้วนำระบบเปลี่ยนสถานะนำเข้า/นำออกไปทำให้เกิดผลด้วยภาษาเอสเอ็มวี (รายละเอียดอยู่ในภาคผนวก ข) แล้วนำไปทวนสอบด้วยตัวตรวจสอบแบบจำลองเอ็นยูเอสเอ็มวี ผลปรากฏว่าสมบัตินำเข้า/นำออกที่คาดหมายสอดคล้องกับแบบจำลองตัวหน่วงการสำเนา

5.3 สรุป

ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบสามารถพิจารณาจากลักษณะพฤติกรรมของแบบจำลองประกอบซึ่งประกอบด้วย ขนาดความยาวของกระแสนำเข้า ชนิดข้อมูลนำเข้า ชนิดกระแสนำเข้า และตัวแปรภายใน ซึ่งจากลักษณะดังกล่าวทำให้สามารถกำหนดฟังก์ชันภาวะนามธรรมประวัติที่เหมาะสมให้กับแบบจำลองประกอบได้จากฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ที่นำเสนอในบทที่ 4 ได้แก่ ฟังก์ชัน $current$ ที่เก็บเฉพาะสมาชิกตัวหลังสุดของกระแสนำเข้าไว้ ฟังก์ชัน $last(n+1)$ ที่เก็บกระแสที่เป็นสมาชิก $n+1$ ตัวหลังสุดของกระแสนำเข้าไว้ ฟังก์ชัน $filterU$ ที่จำแนกข้อมูลนำเข้าเป็นเซตข้อมูลอัพเดท และเซตข้อมูลคำร้องขอ ฟังก์ชัน $break$ ที่จำแนกกระแสนำเข้าเป็นกระแสปกติ และกระแสคงที่ ฟังก์ชัน $reduce(s,+)$ ที่ดำเนินการสมาชิกของกระแสนำเข้าแล้วเก็บไว้ที่ตัวแปรภายใน ฟังก์ชัน $identity$ ที่ยังคงสภาพกระแสนำเข้าเดิมไว้

ความสัมพันธ์ของฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบกับฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองบนพื้นฐานของกระแสเฉพาะรายที่นำมาประกอบด้วยตัวดำเนินการประกอบแบบลำดับคือ สามารถนำเอาฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองบนพื้นฐานของกระแสเฉพาะรายมาพิจารณาเลือกฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบได้ แต่ไม่ได้มี

ความสัมพันธ์ที่แน่นอนที่สามารถกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบจากฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองบนพื้นฐานของกระแสเฉพาะรายโดยตรง

แนวทางในการกำหนดสมบัตินำเข้า/นำออกที่คาดหมายของแบบจำลองประกอบจากสมบัตินำเข้า/นำออกที่คาดหมายของแบบจำลองบนพื้นฐานของกระแสเฉพาะราย สามารถทำได้โดยการนำเอาสมบัตินำเข้า/นำออกที่คาดหมายของแบบจำลองบนพื้นฐานของกระแสเฉพาะรายมาปรับนำเข้าหรือนำออกขึ้นอยู่กับลำดับของการประกอบ โดยนำสมบัตินำเข้า/นำออกที่คาดหมายของส่วนประกอบแรกมาเปลี่ยนนำออกตามนำออกของส่วนประกอบหลังโดยที่นำเข้าคงเดิม และนำสมบัตินำเข้า/นำออกที่คาดหมายของส่วนประกอบหลังมาเปลี่ยนนำเข้าตามนำเข้าของส่วนประกอบแรก (ถ้าฟังก์ชันการประมวลผลกระแสของส่วนประกอบแรกมีฟังก์ชันผกผัน) โดยที่นำออกคงเดิม ทั้งนี้อาจกำหนดสมบัตินำเข้า/นำออกที่คาดหมายของแบบจำลองประกอบเพิ่มเติมได้ตามความเหมาะสม

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 6
บทสรุป และข้อเสนอแนะ

6.1 บทสรุป

วิทยานิพนธ์นี้นำเสนอวิธีการทวนสอบสมบัตินำเข้า/นำออกที่คาดหมายของแบบจำลองประกอบด้วยวิธีการตรวจสอบแบบจำลอง โดยการประยุกต์ระเบียบวิธีการทวนสอบสมบัตินำเข้า/นำออกที่คาดหมายของแบบจำลองบนพื้นฐานของกระแสเฉพาะรายด้วยการตรวจสอบแบบจำลอง ภายใต้แนวความคิดที่ว่าแบบจำลองประกอบทุกรูปแบบสามารถเขียนแทนด้วยการประกอบส่วนประกอบย่อยด้วยตัวดำเนินการประกอบแบบลำดับ ตัวดำเนินการประกอบแบบขนาน ผลป้อนกลับ และส่วนประกอบการจัดเส้นทางซึ่งได้แก่ ส่วนประกอบการไขว้ ส่วนประกอบเอกลักษณ์ ส่วนประกอบสำเนา ส่วนประกอบต้นทาง และส่วนประกอบปลายทาง ด้วยวิธีการนี้สามารถแทนแบบจำลองประกอบด้วยฟังก์ชันการประมวลผลกระแสที่ได้รับจากการประกอบแบบจำลองบนพื้นฐานของกระแสด้วยตัวดำเนินการประกอบได้ โดยฟังก์ชันการประมวลผลกระแสที่ได้รับสามารถแปลงเป็นระบบเปลี่ยนสถานะนำเข้า/นำออกที่สามารถทวนสอบสมบัตินำเข้า/นำออกที่คาดหมายได้ด้วยการตรวจสอบแบบจำลอง วิทยานิพนธ์นี้สนใจแบบจำลองประกอบที่ได้รับจากการประกอบแบบจำลองบนพื้นฐานของกระแสเฉพาะรายด้วยตัวดำเนินการประกอบแบบลำดับ

ผลการทดลองพบว่าฟังก์ชันการประมวลผลกระแสของแบบจำลองประกอบสามารถแปลงเป็นระบบเปลี่ยนสถานะนำเข้า/นำออกที่มีความเหมาะสมที่จะทวนสอบสมบัตินำเข้า/นำออกด้วยการตรวจสอบแบบจำลอง เฉพาะระบบเปลี่ยนสถานะนำเข้า/นำออกที่มีจำนวนสถานะจำกัดเท่านั้นที่สามารถทวนสอบสมบัตินำเข้า/นำออกที่คาดหมายด้วยการตรวจสอบแบบจำลอง การลดจำนวนสถานะในระบบเปลี่ยนสถานะนำเข้า/นำออกขึ้นอยู่กับการเลือกฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ที่เหมาะสม ดังนั้นถ้ากำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ที่สามารถลดจำนวนสถานะในระบบให้มีจำนวนจำกัดได้ก็จะสามารถทวนสอบสมบัตินำเข้า/นำออกที่คาดหมายด้วยการตรวจสอบแบบจำลองได้

ฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ได้ถูกนำเสนอตามกลุ่มฟังก์ชันการประมวลผลกระแสที่สำคัญ 6 กลุ่ม ได้แก่ ฟังก์ชัน *current* ที่เก็บเฉพาะสมาชิกตัวหลังสุดของกระแสนำเข้าไว้ ฟังก์ชัน $last(n+1)$ ที่เก็บกระแสที่เป็นสมาชิก $n+1$ ตัวหลังสุดของกระแสนำเข้าไว้ ฟังก์ชัน *fillU* ที่จำแนกข้อมูลนำเข้าเป็นเซตข้อมูลอัพเดท และเซตข้อมูลคำร้องขอ ฟังก์ชัน *break* ที่จำแนกกระแสนำเข้าเป็นกระแสปกติ และกระแสคงที่ ฟังก์ชัน $reduce(s,+)$ ที่ดำเนินการ

สมาชิกของกระแสนำเข้าแล้วเก็บไว้ที่ตัวแปรภายใน ฟังก์ชัน *identity* ที่ยังคงสภาพกระแสนำเข้าเดิมไว้ ด้วยฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ที่นำเสนอ ทำให้สามารถนำไปใช้กับแบบจำลองประกอบโดยพิจารณาลักษณะพฤติกรรมของแบบจำลองประกอบซึ่งประกอบด้วย ขนาดความยาวของกระแสนำเข้า ชนิดข้อมูลนำเข้า ชนิดกระแสนำเข้า และตัวแปรภายใน ซึ่งจากลักษณะดังกล่าวทำให้สามารถกำหนดฟังก์ชันภาวะนามธรรมประวัติที่เหมาะสมให้กับแบบจำลองประกอบได้

งานวิจัยนี้ได้ศึกษาความสัมพันธ์ของฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบกับฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองบนพื้นฐานของกระแสเฉพาะรายที่นำมาประกอบด้วยตัวดำเนินการประกอบแบบลำดับ จากการทดลองพบว่าฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์มีความสัมพันธ์กับฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองบนพื้นฐานของกระแสเฉพาะราย โดยสามารถนำเอาฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองบนพื้นฐานของกระแสเฉพาะรายมาพิจารณาเลือกฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบได้

วิธีการกำหนดสมบัตินำเข้า/นำออกที่คาดหมายของแบบจำลองประกอบจากสมบัตินำเข้า/นำออกที่คาดหมายของแบบจำลองบนพื้นฐานของกระแสเฉพาะราย สามารถทำได้โดยการนำเอาสมบัตินำเข้า/นำออกที่คาดหมายของแบบจำลองบนพื้นฐานของกระแสเฉพาะรายมาปรับนำเข้าหรือนำออกขึ้นอยู่กับลำดับของการประกอบ โดยนำสมบัตินำเข้า/นำออกที่คาดหมายของส่วนประกอบแรกมาเปลี่ยนนำออกตามนำออกของส่วนประกอบหลังโดยที่นำเข้าคงเดิม และนำสมบัตินำเข้า/นำออกที่คาดหมายของส่วนประกอบหลังมาเปลี่ยนนำเข้าตามนำเข้าของส่วนประกอบแรกโดยที่นำออกคงเดิม ทั้งนี้อาจกำหนดสมบัตินำเข้า/นำออกของแบบจำลองประกอบเพิ่มเติมได้

6.2 **ข้อจำกัด**

1. แบบจำลองบนพื้นฐานของกระแสเฉพาะรายที่นำมาประกอบต้องเป็นฟังก์ชันการประมวลผลกระแสในกลุ่มส่วนประกอบที่อิสระจากประวัติ กลุ่มส่วนประกอบที่มีหน่วยความจำพจน์สั้นแบบมีขอบเขต กลุ่มส่วนประกอบคงที่ กลุ่มส่วนประกอบแบบมีคำร้องขอ กลุ่มส่วนประกอบสแกน และกลุ่มส่วนประกอบที่ต้องการประวัติเท่านั้น เนื่องจากมีการนำเสนอฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของฟังก์ชันการประมวลผลกระแสในแต่ละกลุ่ม

2. ตัวอย่างแบบจำลองประกอบที่ใช้ในวิทยานิพนธ์เป็นแบบจำลองประกอบที่มีหนึ่งช่องทางนำเข้า และหนึ่งช่องทางนำออกเท่านั้น เนื่องจากฟังก์ชันการประมวลผลกระแสที่มีหนึ่งช่องทางนำเข้าและหนึ่งช่องทางนำออกเท่านั้นที่มีระเบียบวิธีสำหรับการแปลงเป็นเครื่องมัวร์

3. วิธีการกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบสามารถกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบแบบจำลองบนพื้นฐานของกระแสเฉพาะรายเท่านั้น ไม่สามารถกำหนดฟังก์ชันภาวะนามธรรมประวัติสำหรับเครื่องมัวร์ของแบบจำลองประกอบที่ได้รับจากการประกอบแบบจำลองประกอบบางส่วน

6.3 ข้อเสนอแนะ

1. แบบจำลองประกอบที่ใช้ทดลองในงานวิจัยนี้ เป็นแบบจำลองประกอบที่ได้รับจากการประกอบแบบจำลองบนพื้นฐานของกระแสเฉพาะรายด้วยตัวดำเนินการประกอบแบบลำดับเท่านั้น ไม่รวมถึงตัวดำเนินการประกอบแบบขนาน และผลป้อนกลับ ดังนั้นสามารถขยายการทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองประกอบด้วยการตรวจสอบแบบจำลองที่ได้รับจากการประกอบแบบจำลองบนพื้นฐานของกระแสเฉพาะรายด้วยตัวดำเนินการประกอบขนาน และผลป้อนกลับ เพื่อให้ครอบคลุมแบบจำลองประกอบมากขึ้น

2. ฟังก์ชันการประมวลผลกระแสที่สามารถแปลงเป็นระบบเปลี่ยนสถานะนำเข้า/นำออกเป็นฟังก์ชันการประมวลผลกระแสที่มีหนึ่งช่องทางนำเข้า และหนึ่งช่องทางนำออกเท่านั้น ดังนั้นหากสามารถสร้างระเบียบวิธีที่แปลงฟังก์ชันการประมวลผลกระแสที่มีหลายช่องทางนำเข้าและหลายช่องทางนำออกจะทำให้สามารถทวนสอบสมบัตินำเข้า/นำออกของแบบจำลองบนพื้นฐานของกระแสมากขึ้น

3. ตัวอย่างแบบจำลองประกอบที่ใช้ทดลองเป็นแบบจำลองที่สร้างขึ้นมาเพื่อประโยชน์ในการทดลองเนื่องจากข้อจำกัดทางด้านตัวดำเนินการประกอบ ดังนั้นหากสามารถยกตัวอย่างแบบจำลองประกอบที่ใช้งานได้จริงจะทำให้น่าสนใจมากยิ่งขึ้น

6.4 ผลงานตีพิมพ์

ระหว่างดำเนินงานวิทยานิพนธ์ได้ตีพิมพ์ผลงานวิจัยในการประชุมวิชาการระดับนานาชาติทางด้านวิทยาการคอมพิวเตอร์และวิศวกรรมซอฟต์แวร์ ครั้งที่ 5 “The 5th International Joint Conference on Computer Science and Software Engineering (JCSSE2008)” ซึ่งได้จัดขึ้นที่

จังหวัดกาญจนบุรี ประเทศไทย ระหว่างวันที่ 7-9 พฤษภาคม พ.ศ.2551 รายละเอียดอยู่ในภาคผนวก ค

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

รายการอ้างอิง

[1] Dosch, W., W. Ruanthong, and A. Stümpel. From stream transformers to Moore machines with input and output. In Y. T. Song, C. Lu, R. Lee editors, *Proceedings of the Seventh International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD'06)*, pages 65-72. IEEE, 2006.

[2] Dosch, W. and A. Stumpel. Deriving state-based implementations of interactive components with history abstractions. In I. Virbitskaite and A. Voronkov, editors, *Perspectives of System Informatics (PSI 2006)*, number 4378 in Lecture Notes in Computer Science, page 180-194. Springer, 2007.

[3] Dosch, W. and A. Stumpel. Transforming stream processing functions into state transition machines. In W. Docsh, R. Y. Lee, and C. Wu, editors, *SERA*, volume 3647 of *Lecture Notes in Computer Science*, pages 1-18. Springer, 2004.

[4] Stumpel, A. *Stream Based Design of Distributed Systems though Refinement*. PhD thesis, Faculty of Technology and Sciences, University of Lubeck, 2003.

[5] Clarke, E. M., O. Grumberg, and D. A. Peled. *Model Checking*. The MIT Press, 2001.

[6] McMillan, K. L. *Symbolic Model Checking*. Kluwer Academic Publishers, Norwell, MA, USA, 1993.

[7] Dosch, W., P. Muenchaisri, W. Ruanthong, and A. Stümpel. Model checking for input/output properties of a black-box model. In S. Sahni editor*, Proceedings of the Third International Conference on Advances in Computer Science and Technology (ACST 2007)*, pages 120-127. Acta Press, 2007.

[8] Dosch, W. and M. Meriste. High-Level Design of a Pull Protocol. In G. Hu editor, *Proceedings of the 20th International Conference on Computers and Their Applications,* pages 66-73. International Society for Computers and Their Applications 2005, 2005.

[9] Broy, M., F. Dederichs and C. Dendorfer. The design of distributed system – An introduction to Focus. Technical Report TUM-I9202, Technische Universitat Munchen, 1992.

[10] Broy, M. Compositional refinement of interactive systems. *DIGITAL Systems Research Center SRC 89*, 1992.

[11] Cimatti, A., E. M. Clark, E. Giunchiglia, F. Guinchiglia, M. Pistoe, M. Roveri, R. Sebastiani, and A. Tacchella. Nusmv 2: An opensource tool for symbolic model checking. In *CAV '02: Proceedings of the 14th International Conference on Computer Aided Verification*, pages 359-364, London, UK, 2002. Springer-Verleg.

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

บรรณานุกรม

ภาษาอังกฤษ

Broy, M. From state to histories: Relating state and history views onto systems. In T. Hoare, M. Broy, R. Steinbruggen, editors, *Engineering Theories of Software Construction*, volume 180 of *Series III: Computer and System Sciences*, pages 149-186. IOS Press, 2001.

Broy , M. and K. Stølen. *Specification and Development of Interactive Systems: Focus on Streams, Interfaces,and Refinement*. Monographs in Computer Science. Springer, 2001.

Broy , M. and G. Stefanescu. The algebra of steam processing functions. Technical Report TUM-I9620, Technische Universitat Munchen, May 1996.

Cadava, R., A. Cimatti, G. Keighren, E. Olivetti, M. Pistore and M. Roveri. *NuSMV 2.2 Tutorial.* [Online]. Available from: http://nusmv.fbk.eu/NuSMV/tutorial/v24/tutorial.pdf/: [2008, August 18].

Cadava, R., A. Cimatti, G. Keighren, E. Olivetti, M. Pistore, M. Roveri and A. Tchaltsev. *NuSMV 2.4 User Manual.* [Online]. Available from: http://nusmv.fbk.eu/NuSMV/userman/v24/nusmv.pdf/: [2008, August 18].

Kahn, G. The semantics of a simple language for parallel programming. In J. Rosenfeld, editor, *Information Processing 74*, pages 471–475. North–Holland, 1974.

Kahn, G. and D. B. MacQueen. Coroutines and networks of parallel processes. In B. Gilchrist, editor, *Information Processing 77*, pages 993–998. North–Holland, 1977.

Roever , W. P., H. Langmaack, and A. Pnueli. The need for compositional proof systems: A survey. *COMPOS'97, LNCS 1356*. Springer, 1998.

Stackebrandt, A. *Component-Based Development of ePayment Systems*. Diploma thesis, Faculty of Technology and Sciences, University of Lubeck, 2008.

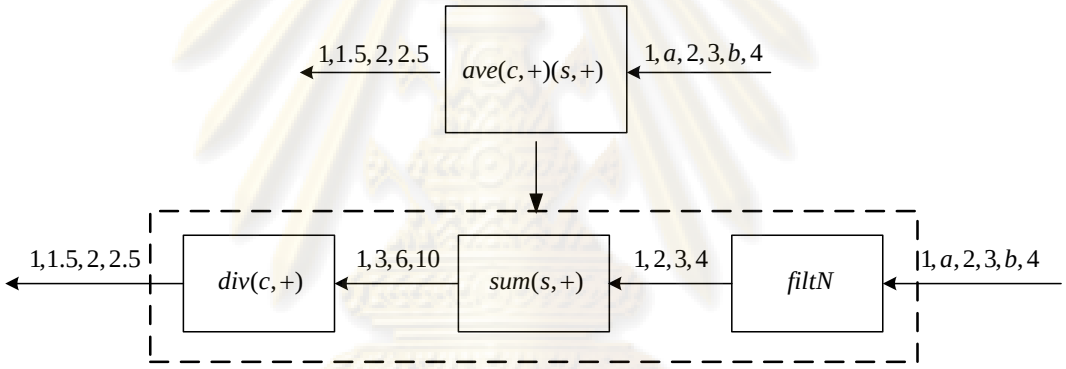
Stephens, R. A survey of stream processing. *Acta Informatica*, 34(7):491-541, 1997.

ภาคผนวก

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ก
การทดสอบสมบัติการเปลี่ยนกลุ่มของตัวดำเนินการประกอบแบบลำดับ

ส่วนนี้นำเสนอการทดสอบสมบัติการเปลี่ยนกลุ่มของตัวดำเนินการประกอบแบบลำดับ โดยตัวอย่างที่ใช้ทำการทดสอบนี้คือ ส่วนประกอบตัวกรองค่าเฉลี่ย (Average filter component) เป็นส่วนประกอบที่คำนวณค่าเฉลี่ยจากสมาชิกของกระแสนำเข้าที่เป็นจำนวนเต็มเท่านั้น โดยส่วนประกอบค่าเฉลี่ยเป็นแบบจำลองประกอบที่ประกอบส่วนประกอบย่อย 3 ส่วนประกอบด้วยตัวดำเนินการประกอบแบบลำดับ คือ ส่วนประกอบตัวกรองจำนวนเต็ม $filtN$ ดังแสดงในตัวอย่างที่ 5.4 ส่วนประกอบผลรวม $sum(s,+)$ ดังแสดงในตัวอย่างที่ 5.1 และส่วนประกอบผลหาร $div(c,+)$ ดังแสดงในตัวอย่างที่ 5.1 ตามลำดับ ดังในรูปที่ ก.1



รูปที่ ก.1 ส่วนประกอบค่าเฉลี่ย

ต้องการทดสอบสมบัติการเปลี่ยนกลุ่มของตัวดำเนินการประกอบแบบลำดับ ดังนั้นจะต้องแสดงว่า $(filtN;sum(s,+));div(c,+) = filtN;(sum(s,+);div(c,+))$

กำหนดให้

$$f = (filtN;sum(s,+))$$

ดังนั้น

$$f:\mathbb{N}\times[\mathbb{N}\times\mathbb{N}\to\mathbb{N}]\to[\mathcal{A}^*\to\mathbb{N}^*]$$

โดยที่

$$f(s,+)(\langle\rangle) = \langle\rangle$$

$$f(s,+)(x \lhd X) = \begin{cases} (s+x) \lhd f(s+x,+)(X) & if\ x\in\mathbb{N} \\ f(s,+)(X) & if\ x\notin\mathbb{N} \end{cases}$$

เมื่อ $x\in\mathcal{A}, X\in\mathcal{A}^*, s\in\mathbb{N}, \mathbb{N}\subseteq\mathcal{A}$ และกำหนดค่าเริ่มต้น $s=0$

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

กำหนดให้

$$g = f; div(c,+)$$

ดังนั้น

$$g : \mathbb{N}\times[\mathbb{N}\times\mathbb{N}\to\mathbb{N}]\to\mathbb{N}\times[\mathbb{N}\times\mathbb{N}\to\mathbb{N}]\to[\mathcal{A}^*\to\mathbb{R}^*]$$

โดยที่

$$g(s,+)(c,+)(\langle\rangle) = \langle\rangle$$

$$g(s,+)(c,+)(x \lhd X) = \begin{cases} ((s+x)/(c+1)) \lhd g(s+x,+)(c+1,+)(X) & if\ x \in \mathbb{N} \\ g(s,+)(c,+)(X) & if\ x \notin \mathbb{N} \end{cases}$$

เมื่อ $x \in \mathcal{A}, X \in \mathcal{A}^*, c, s \in \mathbb{N}, \mathbb{N} \subseteq \mathcal{A}$ และกำหนดค่าเริ่มต้น $s = 0$ และ $c = 0$

กำหนดให้

$$f' = sum(s,+); div(c,+)$$

ดังนั้น

$$f' : \mathbb{N}\times[\mathbb{N}\times\mathbb{N}\to\mathbb{N}]\to\mathbb{N}\times[\mathbb{N}\times\mathbb{N}\to\mathbb{N}]\to[\mathbb{N}^*\to\mathbb{R}^*]$$

โดยที่

$$f'(s,+)(c,+)(\langle\rangle) = \langle\rangle$$

$$f'(s,+)(c,+)(x \lhd X) = ((s+x)/(c+1)) \lhd f'(s+x,+)(c+1,+)(X)$$

เมื่อ $s, c, x \in \mathbb{N}, X \in \mathbb{N}^*$ และกำหนดค่าเริ่มต้น $s = 0$ และ $c = 0$

กำหนดให้

$$g' = filtN; f'$$

ดังนั้น

$$g' : \mathbb{N}\times[\mathbb{N}\times\mathbb{N}\to\mathbb{N}]\to\mathbb{N}\times[\mathbb{N}\times\mathbb{N}\to\mathbb{N}]\to[\mathcal{A}^*\to\mathbb{R}^*]$$

โดยที่

$$g'(s,+)(c,+)(\langle\rangle) = \langle\rangle$$

$$g'(s,+)(c,+)(x \lhd X) = \begin{cases} ((s+x)/(c+1)) \lhd g'(s+x,+)(c+1,+)(X) & if\ x \in \mathbb{N} \\ g'(s,+)(c,+)(X) & if\ x \notin \mathbb{N} \end{cases}$$

เมื่อ $x \in \mathcal{A}, X \in \mathcal{A}^*, c, s \in \mathbb{N}, \mathbb{N} \subseteq \mathcal{A}$ และกำหนดค่าเริ่มต้น $s = 0$ และ $c = 0$

จาก $g = g'$

นั่นคือ $(filtN; sum(s,+)); div(c,+) = filtN; (sum(s,+); div(c,+))$

ดังนั้น ตัวดำเนินการประกอบแบบลำดับมีสมบัติการเปลี่ยนกลุ่มจริง

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ข
รหัสต้นฉบับภาษาเอสเอ็มวี

ส่วนนี้นำเสนอรหัสต้นฉบับ (Source code) ภาษาเอสเอ็มวีที่ใช้ในการทวนสอบสมบัตินำเข้า/นำออกของส่วนประกอบในวิทยานิพนธ์นี้ ได้แก่ส่วนประกอบเรจิสเตอร์การเลื่อน ส่วนประกอบสำเนา และส่วนประกอบตัวหน่วงการสำเนา

ข.1 รหัสต้นฉบับภาษาเอสเอ็มวีที่ใช้ในการทวนสอบสมบัตินำเข้า/นำออกของส่วนประกอบเรจิสเตอร์การเลื่อน

```
MODULE main
-- shift register size = 4


VAR
outp:{empty, 0,1};
state: array 1.. 5 of {empty, 0,1};
length: 0 .. 5;
inpCmd: {empty, 0,1};


ASSIGN
init(inpCmd):=empty;
next(inpCmd):={0,1};

init(length):=0;
next(length):=
case
(length<5):length+1;
1: length;
esac;

init(state[1]):=empty;
init(state[2]):=empty;
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```
init(state[3]):=empty;
init(state[4]):=empty;
init(state[5]):=empty;
next(state[1]) :=
case
(state[1]=empty): next(inpCmd);
(state[1]!=empty)&(length=5):state[2];
1: state[1];
esac;
next(state[2]) :=
case
(state[1]!=empty)&(state[2]=empty):next(inpCmd);
(state[2]!=empty)&(length=5):state[3];
1: state[2];
esac;
next(state[3]) :=
case
(state[1]!=empty)&(state[2]!=empty)&(state[3]=empty): next(inpCmd);
(state[3]!=empty)&(length=5):state[4];
1: state[3];
esac;
next(state[4]) :=
case
(state[1]!=empty)&(state[2]!=empty)&(state[3]!=empty)&(state[4]=empty):next(inpCmd);
(state[4]!=empty)&(length=5):state[5];
1: state[4];
esac;
next(state[5]) :=
case
(state[1]!=empty)&(state[2]!=empty)&(state[3]!=empty)&(state[4]!=empty)&(state[5]=empty
):next(inpCmd);
(state[4]!=empty)&(length=5):next(inpCmd);
1: state[5];
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```
esac;

outp:=
case
(length<5): empty;
(length=5)&(state[1]=1): 1;
(length=5)&(state[1]=0): 0;
1: state[1];
esac;

LTLSPEC
G ((outp = empty) U (length > 4))

LTLSPEC
G ((inpCmd = 1) -> X(X(X(X(outp=1)))))

LTLSPEC
G (((inpCmd = 1)& X(inpCmd = 0)) ->
X(X(X(X((outp=1) & X(outp=0))))))
```

ข.2 รหัสต้นฉบับภาษาเอสเอ็มวีที่ใช้ในการทวนสอบสมบัตินำเข้า/นำออกของส่วนประกอบสำเนา

```
MODULE main
-- duplicate


VAR
outp:{empty, 00,11};
state:{empty, 0,1};
inpCmd: {empty, 0,1};


ASSIGN
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```
init(inpCmd):= empty;
next(inpCmd):= {0,1};

init(state):= empty;
next(state):= next(inpCmd);

outp:=
case
(state= empty): empty;
(state= 0): 00;
1: 11;
esac;



LTLSPEC
G ((inpCmd = empty) -> (outp =  empty))

LTLSPEC
G ((inpCmd = 0) -> (outp =  00))
LTLSPEC
G ((inpCmd = 1) -> (outp =  11))
```

ข.3 รหัสต้นฉบับภาษาเอสเอ็มวีที่ใช้ในการทวนสอบสมบัตินำเข้า/นำออกของส่วนประกอบตัวหน่วงการสำเนา

```
MODULE main
-- duplicate ; shift register size = 4


VAR
outp:{empty, 00,11};
state: array 1.. 5 of {empty, 0,1};
length: 0 .. 5;
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```
inpCmd: {empty, 0,1};


ASSIGN
init(inpCmd):=empty;
next(inpCmd):={0,1};

init(length):=0;
next(length):=
case
(length<5):length+1;
1: length;
esac;

init(state[1]):=empty;
init(state[2]):=empty;
init(state[3]):=empty;
init(state[4]):=empty;
init(state[5]):=empty;
next(state[1]) :=
case
(state[1]=empty): next(inpCmd);
(state[1]!=empty)&(length=5):state[2];
1: state[1];
esac;
next(state[2]) :=
case
(state[1]!=empty)&(state[2]=empty):next(inpCmd);
(state[2]!=empty)&(length=5):state[3];
1: state[2];
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```
esac;
next(state[3]) :=
case
(state[1]!=empty)&(state[2]!=empty)&(state[3]=empty): next(inpCmd);
(state[3]!=empty)&(length=5):state[4];
1: state[3];
esac;
next(state[4]) :=
case
(state[1]!=empty)&(state[2]!=empty)&(state[3]!=empty)&(state[4]=empty):next(inpCmd)
;
(state[4]!=empty)&(length=5):state[5];
1: state[4];
esac;
next(state[5]) :=
case
(state[1]!=empty)&(state[2]!=empty)&(state[3]!=empty)&(state[4]!=empty)&(state[5]=e
mpty):next(inpCmd);
(state[4]!=empty)&(length=5):next(inpCmd);
1: state[5];
esac;

outp:=
case
(length<5): empty;
(length=5)&(state[1]=1): 11;
(length=5)&(state[1]=0): 00;
1: state[1];
esac;
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

LTLSPEC

G ((outp = empty) U (length > 4))

LTLSPEC

G ((inpCmd = 1) -> X(X(X(X(outp=11)))))

LTLSPEC

G (((inpCmd = 1)& X(inpCmd = 0)) ->

X(X(X(X((outp=11) & X(outp=00))))))

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ค
ผลงานตีพิมพ์

History Abstractions for Verifying Input/Output Properties with Model Checking

Nontasak Janchum and Pornsiri Muenchaisri
Departmentment of Computer Engineering, Chulalongkorn University
Bangkok, Thailand
Email: nontasak.j@student.chula.ac.th, pornsiri.mu@chula.ac.th

Abstract

History abstractions play a crucial role in both the stream-based development approach and the input/output properties verification with model checking. A finite input/output transition system is accepted as a basis of model checking. History abstractions are used to reduce an infinite state space to a finite state space of state transition machines and input/output transition systems which are derived from stream processing functions. A stream processing function describes input/output behaviors of a component. History abstractions for Moore machines are used in both refinement and verification. In this paper, we present six history abstractions for Moore machines. We can transform all stream processing functions into Moore machines with this history abstractions. And we investigate each history abstraction whether it is suitable for input/output transition systems.

Key Words: history abstraction, Moore machine, stream processing function, input/output transition system

1. Introduction

The stream-based development approach [3, 12], which is a top-down development approach, starts by modeling input/output behavior with a stream processing function. In subsequent development step the stream processing function is refined to the state transition machine which is suitable for implement-tation. State transition machines are distinguished between Moore machines [6, 7] and Mealy machines [12]. In this paper we emphasize Moore machines.

A stream processing function [9, 10], which describes input/output behavior of a component, is called black-box model of component. It processes input streams arriving at its input ports and emits output streams from its output ports by presenting a mapping from input streams to output streams.

Functional verification of black-box model with model checking [6] is automatic input/output properties verification method. This method has two transforming steps. In the first step, stream processing functions are transformed into Moore machines. In the second step, Moore machines are transformed into input/output transition systems which are suitable for input/output properties verification with model checking.

Transforming the stream processing function into the canonical Moore machine is conceptually simple. But, transforming the canonical Moore machine into the input/output transition system requires some intuition for finding a suitable history abstraction.

History abstractions are used to reduce a number of states in Moore machines and input/output transition systems. From the same stream processing function, the input/output transition system can use the same history abstraction as the Moore machine. Thus, if history abstractions for Moore machines exist, history abstractions for input/output transition systems also exist.

Some history abstractions for Moore machines differ from history abstractions for Mealy machines because outputs of Moore machines are related with states but outputs of Mealy machines are related with transitions.

In this paper, we present six history abstractions for Moore machines. We can transform all stream processing functions into Moore machines with this history abstractions. And we investigate each history abstraction whether it is suitable for input/output transition systems.

The paper is organized as follows: In section 2 the basic notions about stream and stream processing

function are surveyed. The Moore machine and input/output transition system are summarized in section 3. In section 4 the motivating example is demonstrated. In section 5 history abstractions for Moore machine are presented and the possibility of using them to verify input/output properties with model checking is investigated. The conclusion is presented in section 6.

2. Streams and Stream Processing Functions.

In this section, we briefly summarize the basic notion about streams and stream processing functions.

2.1 Streams

A stream models the communication history of a channel which is determined by the sequence of data.

Given a non-empty set $\mathcal{A}$ of data, the set $\mathcal{A}^*$ of finite *streams* $X = \langle x_1, x_2, ..., x_m \rangle$ with elements $x_i \in \mathcal{A}$ $(i \in [1, m])$ of length $m > 0$. Throughout the paper, streams are denoted by capital letters and their elements by small letters.

2.2 Operations on streams

The *concatenation* of two streams $X = \langle x_1, x_2, ..., x_m \rangle$ and $Y = \langle y_1, y_y, ..., y_n \rangle$ forms the stream $X \,\&\, Y = \langle x_1, x_2, ..., x_m, y_1, y_2, ..., y_n \rangle$ of length $m + n$.

The operation $\triangleright$ appends an element to the rear of a stream: $X \triangleright x = X \,\&\, \langle x \rangle$. Symmetrically, the operation $\triangleleft$ attaches an element to the front of a stream: $x \triangleleft X = \langle x \rangle \,\&\, X$.

The subtraction $Z \odot X$ of prefix X from an extension Z yields the stream Y iff $X \,\&\, Y = Z$.

The selection $.[.] : \mathcal{A}^* \times \mathbb{N} \to \mathcal{A}$ accesses an element of the stream: $\langle x_1, x_2, ..., x_k \rangle[i] = x_i \,(1 \le i \le k)$.

2.3 Stream processing functions

A deterministic component is modeled as a stream processing function mapping input histories to output histories as show in Figure 1.

A *stream processing function* $f : \mathcal{A}^* \to \mathcal{B}^*$ maps an input stream to an output stream. The input type $\mathcal{A}$ and the output type $\mathcal{B}$ determine the syntactic interface of the component.

We require that a stream processing function is *monotonic* with respect to prefix order: $f(X) \sqsubseteq f(X \,\&\, Y)$. This property ensures that a prolongation of the input history leads to an extension of the output history.

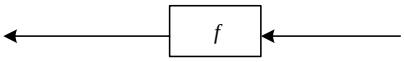


Figure 1. Component as a stream processing function

3. Model checking for input/output properties

To verify input/output properties with model checking [6] , we need two transformation steps. In the first step, we transform the specified stream processing function into a canonical Moore machine taking input histories as states. In the second step, we transform the obtained canonical Moore machine into an input/output transition system using history abstraction to reduce a number of states.

Given a stream processing function $f : \mathcal{A}^* \to \mathcal{B}^*$, we can transform f into a input/output transition system as follows.

3.1 Structure of the canonical Moore machine

A canonical Moore machine with input and output $M_f = (\mathcal{A}^*, \mathcal{A}, \mathcal{B}, \delta, \varphi_f, \langle\rangle)$ consists of the set $\mathcal{A}^*$ of all input histories forms the *state space*, a non-empty set $\mathcal{A}$ of *input data*, a non-empty set $\mathcal{B}$ of *output data*, a single step *state transition function* $\delta : \mathcal{A}^* \times \mathcal{A} \to \mathcal{A}^*$ with $\delta(X, x) = X \triangleright x$, a single state *output function* $\varphi_f : \mathcal{A}^* \to \mathcal{B}^*$ with $\varphi_f(X \triangleright x) = f(X \triangleright x) \odot f(X)$, and the empty history $\langle\rangle$ as *initial state*.

3.2 History abstraction

A history abstraction extracts from an input history certain information that influences the component's behavior.

A function $h : \mathcal{A}^* \to Q$ is called a *history abstraction* for f, if it is *output compatible*: $h(X) = h(Y) \Rightarrow \varphi_f(X) = \varphi_f(Y)$ and *transition closed*: $h(X) = h(Y) \Rightarrow h(X \triangleright x) = h(Y \triangleright x)$.

3.3 Structure of the input/output transition system

An *input/output transition system* $T_{f,\alpha} = (\mathcal{A}, \mathcal{B}, Q, S, R, s_0)$ consists of a non-empty set $\mathcal{A}$ of *input data*, a non-empty set $\mathcal{B}$ of *output data*, a set of *internal states* Q, the set of *states* $S \subseteq (I \cup \{\phi\}) \times Q \times O^*$ where $\phi \notin I$ denotes *empty input*, a *transition relation* $R = \{(\alpha(X), \alpha(X \triangleright x)) \mid X \in \mathcal{A}^*, x \in \mathcal{A}\}$ and *initial state* $s_0 = (\phi, h(\langle\rangle), \varphi_f(\langle\rangle))$. Where the *transforming function* $\alpha : \mathcal{A}^* \to S$ maps the states of the canonical Moore machine into the states of the input/output transition system:

$$\alpha(\langle\rangle) = (\phi, h(\langle\rangle), \varphi_f(\langle\rangle)) \quad (9)$$

$$\alpha(X \triangleright x) = (x, h(X \triangleright x), \varphi_f(X \triangleright x)) \quad (10)$$

4. Motivating example

Before starting with introduction of history abstractions, we demonstrate that history abstractions play an important role in input/output verification with model checking. We demonstrate how to transform the stream processing function into the input/output transition system.

4.1 Stuttering removal component

Sometimes an element may be duplicated during a transmission through stuttering. The component, which eliminates all consecutive duplicates of elements in its input stream, is called a stuttering removal component. The stuttering removal component is defined as follow: $unst : \mathcal{A}^* \to \mathcal{A}^*$

$$unst(\langle\rangle) = \langle\rangle \quad (1)$$

$$unst(\langle x\rangle) = \langle x\rangle \quad (2)$$

$$unst(x \lhd x \lhd X) = unst(x \lhd X) \quad (3)$$

$$unst(x \lhd y \lhd X) = x \lhd unst(y \lhd X) \quad (4)$$

where $x \neq y$.

An empty input stream generates no output (1). A single input stream sends element to output stream (2). A duplicate element of input stream is eliminated (3). An element which does not duplicate is sent to output stream (4).

4.2 Canonical Moore machine

We construct the canonical Moore machine for stuttering removal component as:

$M_{unst} = (\mathcal{A}^*, \mathcal{A}, \mathcal{B}, \delta, \varphi_f, \langle\rangle)$ with

$$\varphi_f(\langle\rangle) = \langle\rangle \quad (5)$$

$$\varphi_f(\langle x\rangle) = \langle x\rangle \quad (6)$$

$$\varphi_f(x \lhd x \lhd X) = \varphi_f(x \lhd X) \quad (7)$$

$$\varphi_f(x \lhd y \lhd X) = \langle x\rangle \lhd \varphi_f(y \lhd X) \quad (8)$$

where $x \neq y$.

An empty input stream generates no output (5). A single input stream sends an element to output stream (6). A duplicate element of input stream is eliminated (7). An element which does not duplicate is sent to output stream (8).

4.3 History abstraction

When we consider output of the canonical Moore machine M_{unst}, we see that output depend on the last two elements of input stream only. Thus, the history abstraction of canonical Moore machine M_{unst} is $h : \mathcal{A}^* \to \mathcal{A}^{\leq 2}$. Since internal states $\mathcal{A}^{\leq 2}$ are streams whose size is less than or equal to 2. Therefore, a number of states is finite if the set of data $\mathcal{A}$ has finite elements.

4.4 Input/output transition system

The input/output transition system $T_{unst,\alpha} = (\mathcal{A}, \mathcal{B}, Q, S, R, s_0)$, which is derived from the canonical Moore machine M_{unst} can demonstrate state transition with state transition table, see Table 1.

Table 1. State transition table of stuttering removal

inp	*State*	*out*	*inp'*	*state'*	*out'*
ϕ	$\langle\rangle$	$\langle\rangle$	x	$\langle x\rangle$	$\langle\rangle$
x	$\langle x\rangle$	$\langle\rangle$	x	$\langle x, x\rangle$	$\langle\rangle$
x	$\langle x\rangle$	$\langle x\rangle$	y	$\langle x, y\rangle$	$\langle x\rangle$
x	$\langle x, x\rangle$	$\langle\rangle$	x	$\langle x, x\rangle$	$\langle\rangle$
x	$\langle x, x\rangle$	$\langle\rangle$	y	$\langle x, y\rangle$	$\langle x\rangle$
y	$\langle x, y\rangle$	$\langle x\rangle$	x	$\langle y, x\rangle$	$\langle y\rangle$
y	$\langle x, y\rangle$	$\langle x\rangle$	y	$\langle y, y\rangle$	$\langle\rangle$

5. History abstraction for Moore machine

Stumpel presented six history abstractions for refining stream processing functions to Mealy machines [12], but they do not suit for refining stream processing functions to Moore machines. Verifying input/output properties with model checking need the same history abstractions as Moore machines. We present history abstractions for Moore machines as each of the six sets of stream processing functions which Stumpel presented. Finally, we use them to verify input/output properties with model checking.

5.1 History independent components

In this section, the stream processing functions of which output depend on a current element of stream, not depend on previous input are presented such as duplicate component and iterator component.

The duplicate component duplicates every element of a finite input stream. The duplicate component is described using the function $dupl : \mathcal{A}^* \to \mathcal{A}^*$ with

$$dupl(\langle\rangle) = \langle\rangle \quad (9)$$

$$dupl(x \lhd X) = x \lhd x \lhd dupl(X) \quad (10)$$

An empty input stream generates no output (9). All elements of input stream are duplicated (10).

The iterator component applies a basic function to the single element of its input stream. The iterator component is described using the function map: $[\mathcal{A} \to \mathcal{B}^*] \to [\mathcal{A}^* \to \mathcal{B}^*]$ with

$$map(g)(\langle\rangle) = \langle\rangle \quad (11)$$

$$map(g)(x \lhd X) = g(x) \,\&\, map(g)(X) \quad (12)$$

An empty input stream generates no output (11). Each of the elements is applied to function g (12).

The history abstraction for Moore machines is $current : \mathcal{A}^* \to \mathcal{A}^{\trianglelefteq 1}$. The output only depends on the current input. Since it is independent of the previous input history, the history abstraction need not preserve any information of the previous input. The function *current* is transition closed, because $current(X \rhd x) = current(Y \rhd y)$ implies $x = y$. Thus $current(X \rhd x \rhd z) = current(Y \rhd y \rhd z)$.

Definition 5.1 (History independent components)
The stream processing functions for which *current* function is a history abstraction describe *history independent* components.

This history abstraction suits for verifying input/output properties with model checking because this history abstraction can reduce infinite states space of canonical Moore machines to finite states space of input/output transition systems.

5.2 Stationing components

In this section, the stream processing functions of which input streams are distinguished between regular input streams and irregular input streams are presented such as self-initializing filter component.

Definition 5.2 (Stationing input) [12]
Let the set $\mathcal{A}^* = R \cup \bigcup_{i \in I} S_i$ of input streams be partitioned into non-empty, mutually disjoint, upward closed sets $S_i (i \in I)$ of *stationing* input streams, and the non-empty set R of *regular* input streams which is the prefix closed.

A self-initializing filter component filters the remainder of its input stream for its first element provided it exists. The self-initializing filter component is described using the function $finit : \mathcal{A}^* \to \mathcal{A}^*$ with

$$finit(\langle\rangle) = \langle\rangle \quad (13)$$

$$finit(\langle x \rangle) = \langle\rangle \quad (14)$$

$$finit(x \lhd y \lhd X) = \begin{cases} y \lhd finit(x \lhd X) & \text{if } x = y \\ finit(x \lhd X) & \text{if } x \neq y \end{cases} \quad (15)$$

An empty input stream (13) and a single input (14) stream generates no output . The function filters the remainder of its input stream for its first element provided it exists (15).

The history abstraction for Moore machines is filter functional $break : \mathcal{A}^* \to R \cup \{s_i \mid i \in I\}$ with $s_i \neq s_j$ for $i \neq j$ where R is set of *regular streams* and I is set of *irregular streams*. The function *break* is indeed transition closed, because $break(X) = break(Y)$ implies $X = Y$ or $X, Y \in S_i$ for an $i \in I$. In both cases $break(X \rhd x) = break(Y \rhd x)$.

Definition 5.3 (Stationing components)
The stream processing functions of which *break* function is a history abstraction describe *stationing* component.

This history abstraction does not suit for verifying input/output properties with model checking because of the state explosion problem.

5.3 Components with a bounded short-term memory

In this section, the stream processing functions of which output depends on last n elements of input stream are presented such as shift register and stuttering component.

The shift register delays the input stream for a fixed number of steps. As long as the length of the input stream does not exceed the size of the shift register, no output will appear. Otherwise, at least the first element is shifted out of the shift register. The shift register is described using the function $shift$: $\mathbb{N} \setminus \{0\} \to [\mathcal{A}^* \to \mathcal{A}^*]$ with

$$shift(n)(\langle\rangle) = \langle\rangle \quad (16)$$

$$shift(n)(x \lhd X) = \begin{cases} \langle\rangle & \text{if } |x \lhd X| \leq n \\ x \lhd shift(n)(X) & \text{if } |x \lhd X| > n \end{cases} \quad (17)$$

An "empty" input stream (16) and a "short" input stream (17 up) generate no output. From a "long" input stream (17 down) all elements out of the rear segment of length n are passed to output stream.

The history abstraction for Moore machines is $last(n+1) : \mathcal{A}^* \to \mathcal{A}^{\trianglelefteq n+1}$. The function *last(n+1)* is transition closed, because $last(n+1)(X) = last(n+1)(Y)$ implies $x = y$ element by element for last n+1 elements. Thus $last(n+1)(X \rhd x) = last(n+1)(Y \rhd x)$.

Definition 5.4 (Components with a bounded short-term memory)
The stream processing functions for which *last(n+1)* is a history abstraction *(n>1)* describe *Components with a bounded short-term memory* with capacity *n*.

This history abstraction suits for verifying input/output properties with model checking because this history abstraction can reduce infinite states

space of canonical Moore machines to finite states space of input/output transition systems.

5.4 Components with requests

In this section, the stream processing functions of which input data are distinguished between *request* input data and *update* input data are presented. The data of the first type having an effect on further output of the component are called *updates* and the data of the latter type without any effect on the future output of the component are called *requests*. The example of component with requests is a memory cell. The memory cell stores one datum in reaction to a write command. The stored datum can be retrieved with a read command. The memory cell is described using the function $cell : (w(\mathcal{D}) \cup \{r\})^* \to \mathcal{D}^*$ with

$$cell(\langle\rangle) = \langle\rangle \quad (18)$$

$$cell(\langle w(d)\rangle) = \langle\rangle \quad (19)$$

$$cell(w(d) \lhd r \lhd X) = d \lhd cell(w(d) \lhd X) \quad (20)$$

$$cell(w(d) \lhd w(e) \lhd X) = cell(w(e) \lhd X) \quad (21)$$

$$cell(r \lhd X) = cell(X). \quad (22)$$

An empty input stream generates no output (18). A single input stream which is a write command generates no output (19). The stored datum can be retrieved with a read command (20). The memory cell stores last one datum (21).
Where $\mathcal{D}$ is a set of data, w is a *write* command , r is a *read* command, $d, e \in \mathcal{D}$ and $X \in \mathcal{D}^*$.

The history abstraction for Moore machines is filter functional $filter : (U \cup R)^* \to U^*$ where U is set of *updates* and R is set of *requests*. The function *filter* is transition closed, because all elements of input streams are eliminated by *filter* function without any effect on the further output of the components.

Definition 5.5 (Components with requests)
The stream processing functions for which $filter(U)$ is a history abstraction describe *components with requests.* The elements in the set U are called *updates*, and the others are called *requests*.

This history abstraction does not suit for verifying input/output properties with model checking because of the state explosion problem.

5.5 Scan components

In this section the stream processing functions which accumulate their input with an operation are presented. The scan component, which has one input and one output port, produces the stream of the prefixes of the input stream reduced under a binary operation. The scan component is described using the function $scan : \mathcal{B} \times [\mathcal{B} \times \mathcal{A} \to \mathcal{B}] \to [\mathcal{A}^* \to \mathcal{B}^*]$ with

$$scan(s, \oplus)(\langle\rangle) = \langle s\rangle \quad (23)$$

$$scan(s, \oplus)(x \lhd X) = s \oplus x \lhd scan(s \oplus x, \oplus)(X) \quad (24)$$

An empty input stream sends an initial state to output stream (23). The scan component produces the stream of the prefixes of the input stream reduced under a binary operation (24).

The history abstraction for Moore machines is a reduce functional $red : \mathcal{B} \times [\mathcal{B} \times \mathcal{A} \to \mathcal{B}] \to [\mathcal{A}^* \to \mathcal{B}]$ with

$$red(s, \oplus)(\langle\rangle) = s \quad (25)$$

$$red(s, \oplus)(x \lhd X) = red(s \oplus x, \oplus)(X) \quad (26)$$

An empty input stream sends an initial state to state (25). The reduce function produces the state of the prefixes of the input stream under a binary operation (26).

For every $\oplus$ and s, the function $red(s, \oplus)$ is transition closed, because it validates $red(s, \oplus)(X \rhd x) = red(s, \oplus)(X) \oplus x$.

Definition 5.6 (Scan components)
The stream processing functions for which $red(s, \oplus)$ is a history abstraction describe *scan* components.

This history abstraction does not suit for verifying input/output properties with model checking because of the state explosion problem.

5.6 History requiring components

In this section, the stream processing functions of which all input streams do not have output equivalent are presented such as positions component.

The stream processing function $pos : (\mathbb{N} \setminus 0)^* \to [(\mathbb{N} \setminus 0)^* \to (\mathbb{N} \setminus 0)^*]$ stores data as stream of position number and receives input stream containing position number. Its output stream depends on its data storage and input stream:

$$pos : (X)(\langle\rangle) = \langle\rangle \quad (27)$$

$$pos : (\langle\rangle)(N) = \langle\rangle \quad (28)$$

$$pos(X)(n \lhd N) = \begin{cases} X[n] \,\&\, pos(X)(N) & \text{if } n \le |X| \\ pos(X)(N) & \text{if } n > |X| \end{cases} \quad (29)$$

An empty input stream generates no output (27). No data storage generates no output (28). The function sends the data element at position *n* of stream if it exists (29).

The history abstraction for Moore machines is a identity functional $id : \mathcal{A}^* \to \mathcal{A}^*$ with $id(X) = (X)$. The function *id* is transition closed, because $id(X) = id(Y)$ implies $x_i = y_i$ for all $1 \le i \le |X|$. Thus $id(X \rhd x) = id(Y \rhd x)$.

Definition 5.7 (History requiring components)
The stream processing functions for which *id* function is a history abstraction describe *history requiring* components.

This history abstraction does not suit for verifying input/output properties with model checking because of the state explosion problem.

6. Conclusion

We present six history abstractions for Moore machines. They can be divided into three different groups in accordance with their capability to reduce state space. The first group, which consists of two history abstractions such as *history independent components* and *components with a bounded short-term memory*, can reduce an infinite state space to a finite state space. The second group, which consists of three history abstractions such as *stationing components*, *components with requests* and *scan components*, can reduce a number of states, but it cannot guarantee to reduce an infinite state space to a finite state space. The last group, which consists of one history abstraction such as *history requiring components*, cannot reduce a number of states. Therefore, the first group is suitable but the other groups are not suitable for input/output transition systems which are state-based model of model checking.

We are currently working on finding single history abstraction is suitable for single component. In long term, we envisage combination of these history abstractions and composition of components.

7. References

[1] M. Broy. From state to histories: Relating state and history views onto systems. In T. Hoare, M. Broy, R. Steinbruggen, editors, *Engineering Theories of Software Construction*, volume 180 of *Series III: Computer and System Sciences*, pages 149-186. IOS Press, 2001.

[2] M. Broy, F. Dederichs and C. Dendorfer. The design of distributed system – An introduction to Focus. Technical Report TUM-I9202, Technische Universitat Munchen, 1992.

[3] M. Broy and K. Stølen. *Specification and Development of Interactive Systems: Focus on Streams, Interfaces,and Refinement*. Monographs in Computer Science. Springer, 2001.

[4] A. Cimatti, E. M. Clark, E. Giunchiglia, F. Guinchiglia, M. Pistoe, M. Roveri, R. Sebastiani, and A. Tacchella. Nusmv 2: An opensource tool for symbolic model checking. In *CAV '02: Proceedings of the 14th International Conference on Computer Aided Verification*, pages 359-364, London, UK, 2002. Springer-Verleg.

[5] E. M. Clarke, O. Grumberg, and D. A. Peled. *Model Checking*. The MIT Press, 2001

[6] W. Dosch, P. Muenchaisri, W. Ruanthong, A. Stümpel. Model checking for input/output properties of a black-box model. In S. Sahni editor, *Proceedings of the Third International Conference on Advances in Computer Science and Technology (ACST 2007)*, pages 120-127. Acta Press, 2007

[7] W. Dosch, W. Ruanthong, A. Stümpel. From stream transformers to Moore state transition machines with input and output. In Y. T. Song, C. Lu, R. Lee editors**,** *Proceedings of the Seventh International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD'06)*, pages 65-72. IEEE, 2006.

[8] G. Kahn. The semantics of a simple language for parallel programming. In J. Rosenfeld, editor, *Information Processing 74*, pages 471–475. North–Holland, 1974.

[9] G. Kahn and D. B. MacQueen. Coroutines and networks of parallel processes. In B. Gilchrist, editor, *Information Processing 77*, pages 993–998. North–Holland, 1977.

[10] K. L. McMillan. *Symbolic Model Checking*. Kluwer Academic Publishers, Norwell, MA, USA, 1993

[11] R. Stephens. A Survey of stream processing. *Acta Informatica*, 34(7):491-541, 1997.

[12] A. Stumpel. *Stream Based Design of Distributed Systems though Refinement*. PhD thesis, Faculty of Technology and Sciences, University of Lubeck, 2003

ประวัติผู้เขียนวิทยานิพนธ์

นายนนทศักดิ์ จันทร์ชุม เกิดวันที่ 10 มีนาคม 2525 สำเร็จการศึกษาปริญญาวิทยาศาสตรบัณฑิต(คณิตศาสตร์) เกียรตินิยมอันดับหนึ่ง ภาควิชาคณิตศาสตร์ คณะวิทยาศาสตร์ มหาวิทยาลัยสงขลานครินทร์ เมื่อปีการศึกษา 2547 และสำเร็จการศึกษาประกาศนียบัตรบัณฑิต (วิชาชีพครู) สาขาวิชาการมัธยมศึกษา คณะศึกษาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ เมื่อปีการศึกษา 2548 เข้าศึกษาต่อหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ปีการศึกษา 2549

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย