การออกแบบและพัฒนาเครื่องมือจำลองอัลกอริทึมการคำนวณแบบเชื่อมตรง

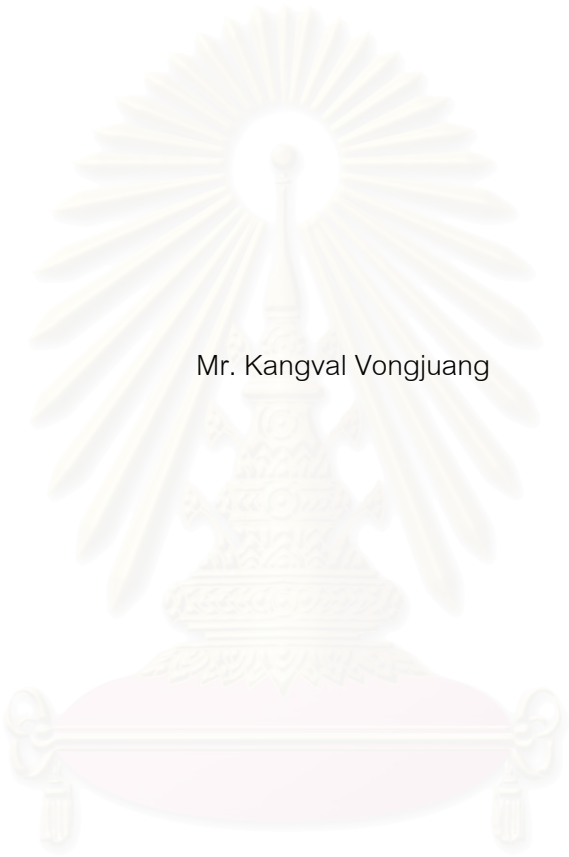
นายกังวาล วงษ์จวง

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2549
ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

# DESIGN AND DEVELOPMENT OF A SIMULATION TOOL FOR AN ON-LINE COMPUTATIONAL ALGORITHM

Mr. Kangval Vongjuang



สถาบันวิทยบริการ
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 2006

Copyright of Chulalongkorn University

| | |
|---|---|
| หัวข้อวิทยานิพนธ์ | การออกแบบและพัฒนาเครื่องมือจำลองอัลกอริทึมการคำนวณแบบเชื่อมตรง |
| โดย | นาย กังวาล วงษ์จวง |
| สาขาวิชา | วิทยาศาสตร์คอมพิวเตอร์ |
| อาจารย์ที่ปรึกษา | อาจารย์ ดร. อรรถสิทธิ์ สุรฤกษ์ |

---

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์

(ศาสตราจารย์ ดร. ดิเรก ลาวัณย์ศิริ)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ

(รองศาสตราจารย์ ดร. วันชัย ริ้วไพบูลย์)

..... อาจารย์ที่ปรึกษา

(อาจารย์ ดร. อรรถสิทธิ์ สุรฤกษ์)

..... กรรมการ

(อาจารย์ ดร. เศรษฐา ปานงาม)

..... กรรมการ

(ผู้ช่วยศาสตราจารย์ ดร. อานนท์ รุ่งสว่าง)

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

กังวาล วงษ์จวง : การออกแบบและพัฒนาเครื่องมือจำลองอัลกอริทึมการคำนวณแบบเชื่อมตรง. (DESIGN AND DEVELOPMENT OF A SIMULATION TOOL FOR AN ON-LINE COMPUTATIONAL ALGORITHM) อ. ที่ปรึกษา : อ. ดร. อรรถสิทธิ์ สุรฤกษ์, 126 หน้า.

วิทยานิพนธ์นี้มีวัตถุประสงค์เพื่อออกแบบและพัฒนาเครื่องมือสำหรับจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรงในรูปของวงจรย่อยต่าง ๆ ประกอบเข้าด้วยกัน ซึ่งกำหนดการทำงานได้ด้วยตารางค่าความจริง เครื่องมือที่ได้เป็นประโยชน์ในการศึกษาและตรวจสอบความถูกต้องในการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรง วิทยานิพนธ์ฉบับนี้เสนอเครื่องมือที่ทำงานบนสภาพแวดล้อมแบบกราฟิกเพื่อให้ผู้ออกแบบสามารถจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรงได้สะดวกโดยผู้ออกแบบสามารถเลือกรูปแบบการแสดงผลการจำลองการทำงานได้เองผ่านทางส่วนต่อประสานกับผู้ใช้และเครื่องมือนี้พัฒนาด้วยภาษาจาวาที่ใช้งานได้ในรูปแบบของแอพเพล็ททำงานเป็นลักษณะของเว็บแอพลิเคชั่นและจัดเก็บอุปกรณ์และวงจรที่ออกแบบลงไฟล์ในรูปแบบของแฟ้มเอกสารเอ็กซ์เอ็มแอล

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาควิชา....วิศวกรรมคอมพิวเตอร์..... ลายมือชื่อนิสิต.....

สาขาวิชา....วิทยาศาสตร์คอมพิวเตอร์..... ลายมือชื่ออาจารย์ที่ปรึกษา.....

ปีการศึกษา....2549.....

## 4671401321 : MAJOR COMPUTER SCIENCE

KEY WORD: ON-LINE ARITHMETIC COMPUTATION / REDUNDANT NUMBER SYSTEM / XML / EVENT-DRIVEN SIMULATION.

KANGVAL VONGJUANG : DESIGN AND DEVELOPMENT OF A SIMULATION TOOL FOR AN ON-LINE COMPUTATIONAL ALGORITHM. THESIS ADVISOR : ATHASIT SURARERKS, Ph.D., 126 pp.

The purposes of this thesis are to design and to development of a simulation tool for on-line computational algorithms which can be represented by a combination of sub-circuits together with their truth tables. This tool is useful for any people who like to study on-line computation algorithms especially in algorithm verification. The introduced tools work in graphical user interface (GUI) mode that is suitable for designer. The program is designed and developed using java programming concept in applet web application. Designed components and circuits can be saved in extensible markup language (XML) format.

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

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

Field of study..Computer Science ..... Advisor's signature.....Athasit Surarerks

Academic year.....2006.....

# กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ ด้วยความอนุเคราะห์และความช่วยเหลืออย่างดียิ่งจาก อาจารย์ ดร. อรรถสิทธิ์ สุรฤกษ์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่ได้สละเวลาให้คำแนะนำและข้อคิดเห็นที่เป็นประโยชน์อย่างยิ่งแก่ผู้จัดทำวิทยานิพนธ์ ตลอดจนเป็นผู้ตรวจทานแก้ไขวิทยานิพนธ์ฉบับนี้ให้มีความถูกต้องและสมบูรณ์มากที่สุด ขอกราบขอบพระคุณเป็นอย่างสูง

ขอกราบขอบพระคุณคณะกรรมการสอบวิทยานิพนธ์ รองศาสตราจารย์ ดร. วันชัย ริ้วไพบูลย์ อาจารย์ ดร. เศรษฐา ปานงาม คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย และผู้ช่วยศาสตราจารย์ ดร. อานนท์ รุ่งสว่าง คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเกษตรศาสตร์ ที่ได้กรุณาให้คำแนะนำแนวทางในการแก้ไขวิทยานิพนธ์ฉบับนี้ให้มีคุณภาพยิ่งขึ้น

ขอกราบขอบพระคุณอาจารย์ทุกท่านในภาควิชาวิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย ที่ได้ให้ความรู้และความเมตตาต่อผู้จัดทำวิทยานิพนธ์ระหว่างการศึกษา

ขอขอบคุณเพื่อนๆ พี่ ๆ น้อง ๆ และเพื่อนร่วมรุ่นทุกท่านเป็นอย่างยิ่ง ที่ได้ให้คำปรึกษาและกำลังใจระหว่างจัดทำวิทยานิพนธ์ฉบับนี้

สุดท้ายนี้ขอกราบขอบพระคุณบิดา มารดา ที่เป็นผู้ให้กำลังใจเสมอมา ช่วยเหลือในทุกๆ ด้านจนสามารถทำวิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงเป็นอย่างดี

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

# สารบัญ

หน้า


บทคัดย่อภาษาไทย ..... ง

บทคัดย่อภาษาอังกฤษ ..... จ

กิตติกรรมประกาศ ..... ฉ

สารบัญ ..... ช

สารบัญตาราง ..... ฌ

สารบัญภาพ ..... ฎ

บทที่

1. บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา ..... 1

1.2 วัตถุประสงค์ของการวิจัย ..... 2

1.3 ขอบเขตของการวิจัย ..... 2

1.4 ขั้นตอนและวิธีการดำเนินการวิจัย ..... 3

1.5 ประโยชน์ที่คาดว่าจะได้รับ ..... 4

2. ทฤษฎีและงานวิจัยที่เกี่ยวข้อง ..... 5

2.1 ทฤษฎีที่เกี่ยวข้อง ..... 5

2.1.1 ระบบจำนวน (Number System) ..... 5

2.1.2 การคำนวณทางคณิตศาสตร์แบบเชื่อมตรง (On-line Arithmetic Computation) ..... 8

2.1.3 เอ็กซ์เอ็มแอล (XML: Extensible Markup Language) ..... 12

2.1.4 การจำลองการทำงานตามสถานการณ์ที่เกิด (Event-Driven Simulation) ..... 14

2.2 เอกสารและงานวิจัยที่เกี่ยวข้อง ..... 15

3 แนวคิดและวิธีวิจัย ..... 17

3.1 แนวคิดในการออกแบบและพัฒนาเครื่องมือ ..... 17

3.2 การวิเคราะห์องค์ประกอบของเครื่องมือ ..... 19

3.3 การประยุกต์เครื่องมือกับอัลกอริทึมการคำนวณแบบเชื่อมตรง ..... 45

3.4 การประยุกต์ใช้งานเครื่องมือบนระบบอินเทอร์เน็ต ..... 46

4 การออกแบบเครื่องมือจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรง ..... 47

4.1 ภาพรวมของการออกแบบเครื่องมือ ..... 47

4.2 แผนภาพอธิบายการออกแบบเครื่องมือ ..... 49

บทที่ หน้า

5 การพัฒนาและการทดสอบ 90

5.1 การพัฒนาเครื่องมือ 90

5.1.1 สภาพแวดล้อมที่ใช้ในการพัฒนาเครื่องมือ 90

5.1.2 เทคนิคที่ใช้ในการพัฒนาเครื่องมือ 91

5.1.3 โครงสร้างของเครื่องมือที่พัฒนา 93

5.2 การทดสอบเครื่องมือ 106

5.2.1 สภาพแวดล้อมที่ใช้ในการทดสอบ 106

5.2.2 กรณีทดสอบ 106

5.2.3 สรุปผลการทดสอบ 114

6 บทสรุปและข้อเสนอแนะ 116

6.1 บทสรุป 116

6.2 ขอบเขตและข้อจำกัดของเครื่องมือ 116

6.3 ข้อเสนอแนะ 117

รายการอ้างอิง 118

ภาคผนวก 119

ภาคผนวก ก คู่มือการใช้งานเครื่องมือจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรง 120

ประวัติผู้เขียนวิทยานิพนธ์ 126

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

# สารบัญตาราง

ตาราง หน้า

ตารางที่ 2.1 การคูณแบบเชื่อมตรงบนฐานสอง 11
ตารางที่ 3.1 ตารางค่าความจริงตามสมการบูลีน AB=Z 21
ตารางที่ 3.2 ตารางค่าความจริงตามสมการบูลีน AB=Y 23
ตารางที่ 3.3 ตารางค่าความจริงตามสมการบูลีน ~(AB)=Y หรือ ~A+~B=Y 24
ตารางที่ 3.4 ตารางค่าความจริงตามสมการบูลีน A+B=Y 24
ตารางที่ 3.5 ตารางค่าความจริงตามสมการบูลีน ~(A+B)=Y หรือ ~A~B=Y 25
ตารางที่ 3.6 ตารางค่าความจริงตามสมการบูลีน A~B+~AB=Y 25
ตารางที่ 3.7 ตารางค่าความจริงตามสมการบูลีน AB+~A~B=Y 26
ตารางที่ 3.8 ตารางค่าความจริงตามสมการบูลีน ~A=Y 26
ตารางที่ 3.9 ตารางค่าความจริงตามสมการบูลีน A=Y 27
ตารางที่ 3.10 ตัวอย่างการป้อนค่าตัวเลขให้กับตัวแปรนำเข้า A และ B 28
ตารางที่ 3.11 ผลลัพธ์การทำงานที่ได้จากการประกอบอุปกรณ์เกตดีเลย์เข้ากับอุปกรณ์เกตแอนด์ 28
ตารางที่ 3.12 ตัวอย่างการป้อนค่าตัวเลขให้กับตัวแปรนำเข้าชนิด NR 33
ตารางที่ 3.13 แสดงการเพิ่มจำนวนแถวในส่วนกำหนดค่าตัวแปรนำเข้าเมื่อเลือกอุปกรณ์ตัวแปรนำเข้าชนิด NR 35
ตารางที่ 3.14 ตัวอย่างการป้อนค่าตัวเลขให้กับตัวแปรนำเข้าชนิด R 36
ตารางที่ 3.15 แสดงผลของการป้อนค่าตัวเลข -1786 ให้กับตัวแปรนำเข้าชนิด R ชื่อ RA+ 36
ตารางที่ 3.16 การทดสอบการแปลงค่าตัวเลข 1786 และ -1786 ตามหลักการแปลงค่าของตัวแปรนำเข้าชนิด R 37
ตารางที่ 3.17 แสดงการเพิ่มจำนวนแถวในส่วนกำหนดค่าตัวแปรนำเข้าเมื่อเลือกอุปกรณ์ตัวแปรนำเข้าชนิด R 38
ตารางที่ 3.18 แสดงผลการกำหนดค่าตัวเลขให้กับตัวแปรนำเข้าชนิด R ชื่อ R0+ 39
ตารางที่ 4.1 รายละเอียดยูสเคสการออกแบบสร้างวงจรใหม่ด้วยสมการบูลีน 51
ตารางที่ 4.2 รายละเอียดยูสเคสการออกแบบสร้างวงจรใหม่ด้วยการวาดเป็นผังวงจร 52
ตารางที่ 4.3 รายละเอียดยูสเคสการเชื่อมต่ออุปกรณ์ต่าง ๆ เข้าด้วยกันบนส่วนวาดวงจร 53
ตารางที่ 4.4 รายละเอียดยูสเคสการเลือกอุปกรณ์ที่มีอยู่จากคลังอุปกรณ์มาใช้งาน 54
ตารางที่ 4.5 รายละเอียดยูสเคสการเรียกไฟล์ข้อมูลของอุปกรณ์หรือวงจรที่เคยออกแบบเก็บไว้มาทำงาน 55

ตาราง หน้า

ตารางที่ 4.6 รายละเอียดยูสเคสการแก้ไขอุปกรณ์หรือวงจรที่เคยออกแบบเก็บไว้ 55
ตารางที่ 4.7 รายละเอียดยูสเคสการเพิ่มอุปกรณ์เข้าไปในวงจรที่เคยออกแบบเก็บไว้ 56
ตารางที่ 4.8 รายละเอียดยูสเคสการลบอุปกรณ์ออกจากวงจรที่เคยออกแบบเก็บไว้ 57
ตารางที่ 4.9 รายละเอียดยูสเคสการจัดวางตำแหน่งของอุปกรณ์ใหม่ 57
ตารางที่ 4.10 รายละเอียดยูสเคสการบันทึกอุปกรณ์หรือวงจรที่ออกแบบลงไฟล์ในรูปแบบแฟ้มเอกสารเอ็กซ์เอ็มแอล 58
ตารางที่ 4.11 รายละเอียดยูสเคสการออกแบบสร้างอุปกรณ์ UDP 59
ตารางที่ 4.12 รายละเอียดยูสเคสการออกแบบสร้างอุปกรณ์ UDE 60
ตารางที่ 4.13 รายละเอียดยูสเคสการเลือกกำหนดค่าตัวเลขให้กับตัวแปรนำเข้าชนิดต่าง ๆ 61
ตารางที่ 4.14 รายละเอียดยูสเคสการเลือกระดับความเร็วในการแสดงผลจำลองการทำงาน 62
ตารางที่ 4.15 รายละเอียดยูสเคสการแสดงผลการจำลองการทำงาน 63
ตารางที่ 4.16 รายละเอียดยูสเคสเลือกจำลองการทำงานให้เกิดขึ้นต่อเนื่องจนแล้วเสร็จ 63
ตารางที่ 4.17 รายละเอียดยูสเคสเลือกจำลองการทำงานให้เกิดขึ้นทีละขั้น 64
ตารางที่ 4.18 รายละเอียดยูสเคสเลือกยกเลิกการจำลองการทำงาน 64
ตารางที่ 5.1 อุปกรณ์เกตพื้นฐาน 96
ตารางที่ 5.2 อุปกรณ์ตัวแปรนำเข้าและตัวแปรผลลัพธ์ 101
ตารางที่ 5.3 แถบเมนูไฟล์ 103
ตารางที่ 5.4 แถบเมนูแก้ไข 104
ตารางที่ 5.5 แถบเมนูตัวเลือก 104
ตารางที่ 5.6 แถบเมนูการจำลอง 105
ตารางที่ 5.7 แถบเมนูช่วยเหลือ 105
ตารางที่ 5.8 แสดงกรณีทดสอบการเรียกใช้งานเครื่องมือในรูปแบบของแอพเพ็ท 106
ตารางที่ 5.9 แสดงกรณีทดสอบการทำงานผ่านระบบอินเทอร์เน็ตและบันทึกอุปกรณ์หรือวงจรที่สร้างลงบนเครื่องคอมพิวเตอร์ลูกข่าย 107
ตารางที่ 5.10 แสดงกรณีทดสอบการจำลองการทำงานที่ผู้ออกแบบสามารถเห็นการเปลี่ยนแปลงข้อมูลนำเข้าและผลลัพธ์ผ่านทางส่วนต่อประสานกับผู้ใช้ 108
ตารางที่ 5.11 แสดงกรณีทดสอบการกำหนดจำนวนตัวแปรนำเข้าและตัวแปรผลลัพธ์สูงสุด 20 จำนวนเพื่อการทดสอบการจำลองการทำงาน 108

ตาราง หน้า

ตารางที่ 5.12 แสดงกรณีทดสอบการออกแบบสร้างวงจรขึ้นใหม่และการนำวงจรที่มีอยู่มาประกอบร่วมเข้าไปกับวงจรที่ออกแบบใหม่ 109
ตารางที่ 5.13 แสดงกรณีทดสอบการบันทึกวงจรที่ออกแบบในรูปแบบแฟ้มเอกสารเอ็กซ์เอ็มแอลและสามารถเรียกกลับมาใช้งานได้ 109
ตารางที่ 5.14 แสดงกรณีทดสอบการออกแบบวงจรโดยการกำหนดด้วยสมการบูลีน 110
ตารางที่ 5.15 แสดงกรณีทดสอบการกำหนดค่าตัวแปรนำเข้าด้วยค่าตัวเลขฐานสิบหรือฐานสองและสามารถเลือกชนิดตัวแปรนำเข้าเป็นชนิดจำนวนซ้ำซ้อนได้ 110
ตารางที่ 5.16 แสดงกรณีทดสอบการทำงานเครื่องมือตามอัลกอริทึมการคำนวณแบบเชื่อมตรง 111
ตารางที่ 5.17 แสดงกรณีทดสอบการทำงานของอุปกรณ์หรือวงจรที่มีอุปกรณ์เกตดีเลย์เป็นส่วนประกอบอยู่ภายในอุปกรณ์หรือวงจรด้วย 111
ตารางที่ 5.18 แสดงกรณีทดสอบการนำอุปกรณ์หรือวงจรที่มีอุปกรณ์เกตดีเลย์ประกอบอยู่ภายในไปใช้ประกอบร่วมกับอุปกรณ์หรือวงจรที่สร้างใหม่ 112
ตารางที่ 5.19 แสดงกรณีทดสอบการทำงานของอุปกรณ์เกตพื้นฐานกับตารางค่าความจริง 112
ตารางที่ 5.20 แสดงกรณีทดสอบการทำงานของอุปกรณ์ UDP ที่สร้างจากการกำหนดค่าความหน่วงและตารางค่าความจริงเอง 113
ตารางที่ 5.21 แสดงกรณีทดสอบการทำงานของอุปกรณ์ UDE ที่สร้างจากสมการทางคณิตศาสตร์ 113
ตารางที่ 5.22 แสดงสรุปผลการทดสอบในแต่ละกรณี 114

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

# สารบัญภาพ

ภาพประกอบ หน้า


รูปที่ 2.1 ค่าตัวเลข 14 บนระบบจำนวนแบบมีเครื่องหมายฐานสอง 7

รูปที่ 2.2 ตัวดำเนินการแบบเชื่อมตรงเมื่อมีค่าความหน่วงเชื่อมตรงเท่ากับ $\delta$ 9

รูปที่ 2.3 การบวกแบบเชื่อมตรงบนฐานห้าของ 708 และ 766 10

รูปที่ 2.4 โครงสร้างของเอกสารเอ็กซ์เอ็มแอล 14

รูปที่ 2.5 ตัวอย่างวงจรแสดงความสัมพันธ์ของข้อมูลนำเข้า 15

รูปที่ 3.1 สัญลักษณ์แทนอุปกรณ์เกตแอนด์ (AND) 23

รูปที่ 3.2 สัญลักษณ์แทนอุปกรณ์เกตแนนด์ (NAND) 23

รูปที่ 3.3 สัญลักษณ์แทนอุปกรณ์เกตออร์ (OR) 24

รูปที่ 3.4 สัญลักษณ์แทนอุปกรณ์เกตนอร์ (NOR) 25

รูปที่ 3.5 สัญลักษณ์แทนอุปกรณ์เกตเอ็กซ์คลูซีฟออร์ (XOR) 25

รูปที่ 3.6 สัญลักษณ์แทนอุปกรณ์เกตเอ็กซ์คลูซีฟนอร์ (XNOR) 26

รูปที่ 3.7 สัญลักษณ์แทนอุปกรณ์เกตอินเวอร์เตอร์ (NOT) 26

รูปที่ 3.8 สัญลักษณ์แทนอุปกรณ์เกตบัฟเฟอร์ (BUFFER) 27

รูปที่ 3.9 สัญลักษณ์แทนอุปกรณ์เกตดีเลย์ (DELAY) 27

รูปที่ 3.10 แสดงการประกอบอุปกรณ์เกตดีเลย์เข้ากับอุปกรณ์เกตแอนด์ 28

รูปที่ 3.11 สัญลักษณ์แทนอุปกรณ์ตัวแปรนำเข้าชนิด NR 35

รูปที่ 3.12 สัญลักษณ์แทนอุปกรณ์ตัวแปรนำเข้าชนิด R 38

รูปที่ 3.13 สัญลักษณ์แทนอุปกรณ์ตัวแปรนำเข้าชนิดค่าคงที่ 40

รูปที่ 3.14 สัญลักษณ์แทนอุปกรณ์ตัวแปรผลลัพธ์ 41

รูปที่ 4.1 แผนภาพของแพ็กเกจแสดงองค์ประกอบของระบบงาน 47

รูปที่ 4.2 แผนภาพยูสเคสรวมของระบบ 49

รูปที่ 4.3 ยูสเคสของการออกแบบสร้างวงจรใหม่ด้วยสมการบูลีน 51

รูปที่ 4.4 ยูสเคสของการออกแบบสร้างวงจรใหม่ด้วยการวาดเป็นผังวงจร 52

รูปที่ 4.5 ยูสเคสของการเรียกไฟล์ข้อมูลของอุปกรณ์หรือวงจรมาทำงาน 54

รูปที่ 4.6 ยูสเคสของการบันทึกอุปกรณ์หรือวงจรที่ออกแบบเก็บลงไฟล์ 58

รูปที่ 4.7 ยูสเคสของการออกแบบสร้างอุปกรณ์ UDP 59

รูปที่ 4.8 ยูสเคสของการออกแบบสร้างอุปกรณ์ UDE 60

รูปที่ 4.9 ยูสเคสของการเลือกกำหนดค่าตัวเลขให้กับตัวแปรนำเข้าชนิดต่าง ๆ 61

รูปที่ 4.10 ยูสเคสของการเลือกกำหนดระดับความเร็วในการจำลองการทำงาน 61

ภาพประกอบ หน้า

รูปที่ 4.11 ยูสเคสของการของการแสดงผลการจำลองการทำงาน 62
รูปที่ 4.12 แผนภาพคลาสของเครื่องมือจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรงสำหรับส่วนการออกแบบ 65
รูปที่ 4.13 แผนภาพคลาสของเครื่องมือจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรงสำหรับส่วนการจำลองการทำงาน 65
รูปที่ 4.14 คลาส ocasim.ui.MainFrame 66
รูปที่ 4.15 คลาส ocasim.core.CircuitView 66
รูปที่ 4.16 คลาส ocasim.core.CircuitViewInterface 67
รูปที่ 4.17 คลาส ocasim.core.EmptyView 67
รูปที่ 4.18 คลาส ocasim.core.PaintableContainer 68
รูปที่ 4.19 คลาส ocasim.core.PaintableContainerControl 69
รูปที่ 4.20 คลาส ocasim.core.CircuitModel 70
รูปที่ 4.21 คลาส ocasim.core.ComponentModel 71
รูปที่ 4.22 คลาส ocasim.core.ComponentView 72
รูปที่ 4.23 คลาส ocasim.core.Paintable 72
รูปที่ 4.24 คลาส ocasim.core.WireModel 73
รูปที่ 4.25 คลาส ocasim.comp.TerminalModel 73
รูปที่ 4.26 คลาส ocasim.core.ICModel 74
รูปที่ 4.27 คลาส ocasim.core.ComponentControl 75
รูปที่ 4.28 คลาส ocasim.ui.BooleanDialog 75
รูปที่ 4.29 คลาส ocasim.ui.UDPDialog 75
รูปที่ 4.30 คลาส ocasim.ui.UDPTableModel 76
รูปที่ 4.31 คลาส ocasim.core.CircuitMemento 77
รูปที่ 4.32 คลาส ocasim.core.ComponentMemento 77
รูปที่ 4.33 คลาส ocasim.core.Command 78
รูปที่ 4.34 คลาส ocasim.core.Actions 79
รูปที่ 4.35 คลาส ocasim.expr.LogicExprBuilder 80
รูปที่ 4.36 คลาส ocasim.expr.LogicExpression 80
รูปที่ 4.37 คลาส ocasim.expr.ExpressionTruthTableBuilder 80

ภาพประกอบ หน้า

รูปที่ 4.38 คลาส ocasim.expr.TruthCaseBuilder 80
รูปที่ 4.39 คลาส ocasim.ui.SimulationPanel 81
รูปที่ 4.40 คลาส ocasim.ui.InputPanel 81
รูปที่ 4.41 คลาส ocasim.ui.InputTableModel 82
รูปที่ 4.42 คลาส ocasim.ui.OutputPanel 82
รูปที่ 4.43 คลาส ocasim.ui.OutputTableModel 83
รูปที่ 4.44 คลาส ocasim.core.SimulationRunner 83
รูปที่ 4.45 แผนภาพซีเควนซ์แสดงขั้นตอนการสร้างวงจรด้วยสมการบูลีน 84
รูปที่ 4.46 แผนภาพซีเควนซ์แสดงขั้นตอนการสร้างวงจรด้วยการวาดเป็นผังวงจรจาก อุปกรณ์ที่มีอยู่บนคลังอุปกรณ์ 85
รูปที่ 4.47 แผนภาพซีเควนซ์แสดงขั้นตอนการสร้างอุปกรณ์ UDP 85
รูปที่ 4.48 แผนภาพซีเควนซ์แสดงขั้นตอนการสร้างอุปกรณ์ UDE 86
รูปที่ 4.49 แผนภาพซีเควนซ์แสดงขั้นตอนการเรียกคืนหรือการบันทึกข้อมูลแฟ้มเอกสาร เอ็กซ์เอ็มแอล 87
รูปที่ 4.50 แผนภาพซีเควนซ์แสดงขั้นตอนการกำหนดค่าให้กับตัวแปรนำเข้า 87
รูปที่ 4.51 แผนภาพซีเควนซ์แสดงขั้นตอนการจำลองการทำงานทั้ง 2 รูปแบบ 88
รูปที่ 5.1 วงจรย่อย ๆ ประกอบรวมเป็นส่วนประกอบของวงจรที่ใหญ่ขึ้น 92
รูปที่ 5.2 หน้าจอการทำงานเป็นแอพเพล็ทบนบราวเซอร์ 93
รูปที่ 5.3 หน้าจอหลักของเครื่องมือ 93
รูปที่ 5.4 ส่วนวาดวงจร 94
รูปที่ 5.5 เมนูสร้างวงจรที่กำหนดด้วยสมการบูลีน 95
รูปที่ 5.6 หน้าจอสำหรับกำหนดสมการบูลีน 95
รูปที่ 5.7 วงจรที่วาดจากการกำหนดสมการบูลีน 95
รูปที่ 5.8 ส่วนคลังอุปกรณ์ 96
รูปที่ 5.9 อุปกรณ์เกตพื้นฐาน 96
รูปที่ 5.10 อุปกรณ์ประกอบ 97
รูปที่ 5.11 แทนอุปกรณ์ประกอบชื่อ AD+RT ด้วยรูปสี่เหลี่ยม 97
รูปที่ 5.12 โครงสร้างภายในของอุปกรณ์ประกอบที่ชื่อ AD+RT 97

ภาพประกอบ หน้า

รูปที่ 5.13 เมนูสร้างอุปกรณ์ที่กำหนดค่าความหน่วงและตารางค่าความจริงเอง 98
รูปที่ 5.14 หน้าจอกำหนดค่าความหน่วงสำหรับตัวอุปกรณ์ UDP 98
รูปที่ 5.15 หน้าจอสำหรับกำหนดค่าในตารางค่าความจริงเอง 99
รูปที่ 5.16 อุปกรณ์ที่สร้างจากการกำหนดค่าความหน่วงและตารางค่าความจริงเอง 99
รูปที่ 5.17 เมนูสร้างอุปกรณ์ที่กำหนดด้วยสมการทางคณิตศาสตร์ 99
รูปที่ 5.18 หน้าจอสำหรับการกำหนดสมการทางคณิตศาสตร์ 100
รูปที่ 5.19 อุปกรณ์ UDE ที่สร้างจากการกำหนดด้วยสมการทางคณิตศาสตร์ 100
รูปที่ 5.20 อุปกรณ์ตัวแปรนำเข้าและตัวแปรผลลัพธ์ 100
รูปที่ 5.21 ส่วนกำหนดค่าตัวแปรนำเข้า 101
รูปที่ 5.22 ส่วนแสดงค่าผลลัพธ์ผ่านทางตัวแปรผลลัพธ์ 102
รูปที่ 5.23 ส่วนควบคุมความเร็วการจำลองการทำงาน 102
รูปที่ 5.24 ส่วนประมวลผลการจำลองการทำงาน 102
รูปที่ 5.25 ส่วนแสดงจำนวนคล็อก 103
รูปที่ 5.26 แถบเมนูและแถบเครื่องมือ 103
รูปที่ 5.27 แถบเมนูไฟล์ 103
รูปที่ 5.28 แถบเมนูแก้ไข 104
รูปที่ 5.29 แถบเมนูตัวเลือก 104
รูปที่ 5.30 แถบเมนูการจำลอง 105
รูปที่ 5.31 แถบเมนูช่วยเหลือ 105
รูปที่ 5.32 แถบสถานะการทำงาน 105
รูปที่ ก1 หน้าจอการทำงานเป็นแอพเพล็ทบนบราวเซอร์ 120
รูปที่ ก2 หน้าจอหลักของเครื่องมือ 121
รูปที่ ก3 การออกแบบการออกแบบสร้างวงจรด้วยอุปกรณ์เกตพื้นฐานต่าง ๆ 123
รูปที่ ก4 การป้อนค่าตัวแปรนำเข้าเพื่อทดสอบจำลองการทำงาน 124
รูปที่ ก5 การนำวงจรที่เคยออกแบบไว้แล้วมาทดสอบการทำงาน 125

# บทที่ 1

# บทนำ

## 1.1 ความเป็นมาและความสำคัญของปัญหา

การออกแบบและพัฒนาอัลกอริทึมสำหรับการคำนวณทางคณิตศาสตร์นั้น โดยทั่วไปมุ่งเน้นไปที่ความเร็วในการคำนวณบ้าง ความถูกต้องในการคำนวณบ้าง ตามแต่วัตถุประสงค์หลักของอัลกอริทึม ซึ่งในปัจจุบันอัลกอริทึมต่าง ๆ มีความซับซ้อนสูงขึ้นเป็นลำดับ ปัญหาของการทดสอบและทำความเข้าใจอัลกอริทึมจึงเป็นเรื่องที่ต้องใช้เวลาในการทำงานสูงขึ้นไปด้วย วิธีหนึ่งที่เพิ่มความสะดวกให้แก่ผู้ออกแบบก็คือ การนำคอมพิวเตอร์มาช่วยจำลองการทำงานของอัลกอริทึมที่ออกแบบซึ่งจะช่วยลดเวลาที่ต้องใช้ลงได้

เครื่องมือการจำลองต่าง ๆ ที่ใช้อยู่ในปัจจุบัน เช่น DLSim [1] หรือ Logisim [2] เป็นโปรแกรมสำเร็จรูปที่ผู้ออกแบบสามารถจำลองการทำงานของอัลกอริทึมต่าง ๆ ได้โดยไม่ต้องสร้างอัลกอริทึมนั้น ๆ ขึ้นจริง แต่อัลกอริทึมเหล่านั้นจะต้องถูกแปลงในอยู่ในมุมมองของวงจรเสียก่อน นั่นคืออัลกอริทึมจะต้องถูกแปลงให้อยู่ในรูปของการประกอบกันของเกตทางตรรกะ (Logic Gate) ซึ่งมิใช่เรื่องง่ายในกรณีที่อัลกอริทึมมีความซับซ้อนสูง และโดยเฉพาะกรณีที่ผู้ออกแบบมิได้มีความถนัดในเรื่องของวงจรประมวลผล อย่างไรก็ดี การจำลองการทำงานของอัลกอริทึมในระดับนี้ก็ยังเป็นสิ่งที่มีความสำคัญ เพราะทำให้ผู้ออกแบบคำนึงถึงปัญหาการติดตั้งด้วยในขณะทำการออกแบบ

ความซับซ้อนของอัลกอริทึมการคำนวณ หรือความซับซ้อนของวงจรการคำนวณล้วนเกิดจากการประกอบกันขึ้นมาของหน่วยย่อยพื้นฐาน หรือ เกต (Gate) โดยเกตคือวงจรอิเล็กทรอนิกส์ที่มีจำนวนของส่วนข้อมูลนำเข้าหนึ่งค่าหรือมากกว่าและมีจำนวนของผลลัพธ์หนึ่งค่า โดยที่ค่าของแต่ละผลลัพธ์จะมีค่าเป็นศูนย์หรือหนึ่ง อย่างใดอย่างหนึ่ง อุปกรณ์ในงานดิจิตอลไม่ได้จำกัดเพียงเกตพื้นฐาน แต่อาจมีวงจรรวมต่าง ๆ จำนวนมากที่มีความซับซ้อนสูงถูกนำมาใช้เป็นส่วนประกอบอื่นอีกด้วย นอกจากนี้ในการตรวจสอบการทำงานของอัลกอริทึมที่ได้ออกแบบมานั้น การจำลองการทำงานของอัลกอริทึมอาจทำให้ประหยัดเวลาในการตรวจสอบการทำงานของอัลกอริทึมด้วย

การคำนวณแบบเชื่อมตรงนั้น มีลักษณะพิเศษ คือข้อมูลนำเข้าจะถูกนำไปประมวลผลแบบลำดับควบคู่ไปกับการคำนวณแบบท่อตรง และให้ผลลัพธ์ในลักษณะตำแหน่งต่อตำแหน่ง (Digit by Digit Manner) โดยทุกข้อมูลนำเข้าจะถูกพิจารณาตามลำดับจากหลักที่มีนัยสำคัญมากสุด (MSDF: Most Significant Digit First) ไปยังหลักที่มีนัยสำคัญต่ำสุด (LSDF: Least Significant Digit First) เช่นเดียวกับข้อมูลนำเข้า ผลลัพธ์ก็ถูกคำนวณและแสดงออกมาในรูปแบบเดียวกันคือตามลำดับ ลักษณะเด่นของการทำงานแบบนี้คือ ข้อมูลนำเข้าหรือตัวดำเนินการแต่ละตัวจะถูกส่งมาในวงจรครั้งละหนึ่งบิตและบิตถัดไปจะทยอยมาตามลำดับเวลาที่ดำเนินไปด้วยกัน โดยการคำนวณไม่ขึ้นกับความซับซ้อน หรือความยาวของตัวถูกดำเนินการที่ใช้ในการคำนวณ และความเร็วในการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรงจะขึ้นอยู่กับค่าความหน่วงเชื่อมตรง (On-Line Delay) ของแต่ละอัลกอริทึมด้วย

วิทยานิพนธ์นี้จึงได้เสนอการออกแบบและพัฒนาเครื่องมือที่ใช้สำหรับจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรงในรูปของวงจรย่อยต่าง ๆ ประกอบเข้าด้วยกัน ซึ่งกำหนดได้ด้วยตารางค่าความจริงของหน่วยนั้น และจัดเก็บอุปกรณ์และวงจรที่ออกแบบในรูปแบบของแฟ้มเอกสารเอ็กซ์เอ็มแอล (XML: Extensible Markup Language) โดยเครื่องมือจะทำงานบนสภาพแวดล้อมแบบกราฟิกเพื่อให้ผู้ออกแบบสามารถจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรงได้สะดวก และเครื่องมือนี้พัฒนาด้วยภาษาจาวาที่ใช้งานได้ในรูปแบบของแอพเพล็ท (Applet) และทำงานเป็นลักษณะของเว็บแอพลิเคชั่น (Web Application)

## 1.2 วัตถุประสงค์ของการวิจัย

เพื่อออกแบบและพัฒนาเครื่องมือสำหรับจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรงในระดับวงจร เพื่อประโยชน์ในการศึกษาและตรวจสอบความถูกต้องในการทำงานของอัลกอริทึม

## 1.3 ขอบเขตของการวิจัย

1. ออกแบบและพัฒนาสร้างเครื่องมือจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรง

2. เครื่องมือพัฒนาเป็นเว็บแอพลิเคชั่น ด้วยภาษาจาวา รุ่น 1.4 ขึ้นไป ในรูปแบบจาวาแอพเพล็ท

3. เครื่องมือสามารถใช้งานผ่านระบบเครือข่ายอินเตอร์เน็ตได้ และเก็บบันทึกวงจรที่ออกแบบลงบนเครื่องคอมพิวเตอร์ลูกข่าย (Client)

4. การจำลองการทำงานแสดงผลในรูปแบบของกราฟิกที่ผู้ออกแบบสามารถเห็นการเปลี่ยนแปลงค่าของตัวแปรนำเข้าและตัวแปรผลลัพธ์ผ่านทางส่วนต่อประสานกับผู้ใช้ (User Interface) ได้โดยตรง

5. จำนวนตัวแปรนำเข้าและตัวแปรผลลัพธ์ที่ใช้ในการทดสอบจำลองการทำงานไม่เกิน 20 จำนวน

6. สามารถออกแบบสร้างวงจรใหม่และจัดเตรียมวงจรบางส่วนสำหรับนำมาประกอบรวมเข้าไปกับวงจรที่ออกแบบใหม่ โดยการออกแบบสร้างวงจรใหม่สามารถกำหนดด้วยสมการบูลีน (Boolean Equation) ได้

7. เก็บบันทึกวงจรที่ออกแบบลงไฟล์ในรูปแบบของแฟ้มเอกสารเอ็กซ์เอ็มแอลและสามารถเรียกกลับมาใช้ภายหลังได้

8. สามารถออกแบบสร้างอุปกรณ์ที่กำหนดด้วยสมการทางคณิตศาสตร์ (UDE: User-defined Expression) ได้ และสามารถออกแบบสร้างอุปกรณ์พื้นฐานที่กำหนดโดยผู้ออกแบบเอง (UDP: User-defined Primitive) ได้ หากมีการแก้ไขตารางค่าความจริง (Truth Table) ของอุปกรณ์ UDP จะต้องบันทึกอุปกรณ์นั้นเป็นชื่อใหม่

9. ส่วนกำหนดค่าตัวแปรนำเข้าสามารถป้อนค่าด้วยตัวเลขฐานสิบหรือฐานสองได้โดยที่สามารถป้อนค่าติดลบได้หากตัวแปรนำเข้าเป็นชนิดระบบจำนวนซ้ำซ้อน (Redundant Number System) และหากมีการเปลี่ยนแปลงค่าตัวแปรนำเข้าที่ค่าตัวเลขฐานใดให้คำนวณค่าที่ถูกต้องของอีกฐานด้วย

## 1.4 ขั้นตอนและวิธีการดำเนินการวิจัย

1. ศึกษาวิธีการเขียนโปรแกรมภาษาจาวาที่ใช้งานในรูปแบบของแอพเพล็ทในการออกแบบและสร้างเครื่องมือที่สามารถแสดงผลกราฟิกได้

2. ศึกษาวิธีการเขียนโปรแกรมภาษาจาวาในการสร้างแฟ้มเอกสารเอ็กซ์เอ็มแอลเพื่อจัดเก็บวงจรที่ออกแบบ

3. ศึกษาอัลกอริทึมการบวกและการลบแบบเชื่อมตรง

4. ออกแบบและพัฒนาสร้างเครื่องมือจำลองการทำงานของอัลกอริทึมด้วยโปรแกรมภาษาจาวาที่ใช้งานในรูปแบบของแอพเพล็ทร่วมกับแฟ้มเอกสารเอ็กซ์เอ็มแอล

5. พัฒนาเครื่องมือให้ทำงานตามอัลกอริทึมการคำนวณแบบเชื่อมตรงได้

6. ทดสอบเครื่องมือกับอัลกอริทึมการคำนวณแบบเชื่อมตรง

7. สรุปผลและจัดทำรูปเล่มวิทยานิพนธ์

## 1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้เครื่องมือจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรง

2. สามารถนำเครื่องมือไปพัฒนาประยุกต์การเรียนรู้เพื่อสร้างเสริมทักษะและประสบการณ์ให้แก่ผู้ที่ต้องการศึกษาการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรง

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

# บทที่ 2

# ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

## 2.1 ทฤษฎีที่เกี่ยวข้อง

ในการออกแบบและพัฒนาเครื่องมือจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรงที่ผู้วิจัยได้นำเสนอนั้นประกอบไปด้วยทฤษฎีที่เกี่ยวข้องดังนี้

### 2.1.1 ระบบจำนวน (Number System)

ระบบจำนวนสามารถเขียนอยู่ในรูปของ ($\beta$ , D) ประกอบด้วยเลขฐาน $\beta$ ที่สามารถเป็นได้ทั้งจำนวนจริงและจำนวนเชิงซ้อน โดยที่ $\beta > 1$ และเซตจำกัดของตัวเลข (Finite Digit Set) D ที่เป็นได้ทั้งจำนวนจริงและจำนวนเชิงซ้อน ในระบบจำนวนตามวิธีคลาสสิค

ถ้า $\beta$ เป็นจำนวนเต็มและมีเซตของตัวเลข D อยู่ในรูปของ $\{c \in I \mid 0 \leq c \leq |\beta|-1\}$ เรียกเซตของตัวเลขนี้ว่า เซตจำกัดของตัวเลขแบบคาโนนิคอล (Canonical Digit Set) แทนด้วย C ซึ่ง C มีค่าอยู่ระหว่าง $\{0, 1, 2, 3, \ldots, \beta-1\}$ สำหรับระบบจำนวนที่ประกอบด้วยเลขฐาน $\beta$

สามารถแทนตัวเลขของ X ใน D ที่มีเลขฐาน $\beta$ อยู่ในรูปแบบดังนี้

$$X = (X_a X_{a-1} X_{a-2} X_{a-3} X_{a-4} \ldots X_0 . X_{-1} X_{-2} X_{-3} \ldots)_\beta \text{ โดยที่ } X_i \in D \text{ และ } i \leq a$$

ค่าเชิงตัวเลข (Numerical Value) ของ X ในฐาน $\beta$ เขียนแทนด้วยสัญลักษณ์ ||X|| นิยามดังนี้

$$||X|| = \sum_{i=a}^{-\infty} X^i \beta^i$$

ในระบบจำนวน ($\beta$ , D) ถ้าสามารถแทนตัวเลขที่มีเลขฐาน $\beta$ ที่ยาวจำกัดได้ทั้ง $X_1$ และ $X_2$ โดยที่ $||X_1|| = ||X_2||$ แล้ว เรียกระบบจำนวนนี้ว่าระบบจำนวนซ้ำซ้อน (RNS: Redundant Number System)

ตัวอย่างเช่น กำหนดให้ $\beta$ = 2 และ D = {-1, 0, 1} สามารถเขียนแทนค่า 5 ได้มากกว่าหนึ่งรูปแบบดังนี้ $(101)_2$ และ $(10\text{-}1\text{-}1)_2$

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

2.1.1.1 ระบบจำนวนแบบมีเครื่องหมาย (Signed-Digit Number Systems)

ในปี 1961, Avizienis [3] ได้เสนอแนวคิดเรื่อง ระบบจำนวนแบบมีเครื่องหมาย ($\beta$ , D) โดยที่ $\beta$ > 2 และเซตของตัวเลข D เขียนแทนด้วย [-a, a] ที่มีลักษณะเป็นเซตของตัวเลขแบบสมมาตรโดยที่ $\lfloor \beta/2 \rfloor + 1 \leq a \leq \beta - 1$ ต่อมาเรียกว่า ระบบจำนวนแบบมีเครื่องหมายแบบปกติ (OSD: Ordinary Signed Digit Number System)

ระบบจำนวนแบบมีเครื่องหมาย กำหนดให้ เซตจำกัดของตัวเลข D = {-1, 0, 1} เมื่อเลขฐาน $\beta$ = 2 แต่สำหรับเลขฐานที่มีค่า $\beta \geq 3$ เซตจำกัดของตัวเลข D จะมีค่าอยู่ในรูปของ $\{e \in Z \mid -d \leq e \leq d\}$ เมื่อ $\beta/2 < d \leq \beta - 1$

นิยาม 1:

ระบบจำนวนแบบมีเครื่องหมาย ($\beta$ , D) ประกอบไปด้วยเลขฐาน $\beta$ โดยที่ $\beta$ เป็นจำนวนเต็มที่มากกว่าหรือเท่ากับ 2 และเซตจำกัดของตัวเลข D = $\{d \in I \mid a \leq d \leq b\}$, a และ b เป็นจำนวนเต็มที่ $a \leq 0 \leq b$

หมายเหตุ 1:

1) จำนวนลบจะไม่สามารถแสดงในระบบจำนวนนี้ได้ ถ้า a = 0

2) จำนวนบวกจะไม่สามารถแสดงในระบบจำนวนนี้ได้ ถ้า b = 0

จำนวนของตัวเลขในเซตจำกัดของตัวเลข D จะเท่ากับ |D| = b–a+1 จำนวนของตัวเลขนี้สามารถอธิบายคุณสมบัติที่ซ้ำซ้อนของระบบที่อาศัยการแทนตัวเลขได้ ระบบที่ว่านี้จะถูกเรียกว่า ระบบซ้ำซ้อน (Redundancy System) ถ้าหากว่าสามารถแทนค่าตัวเลขค่าหนึ่งได้มากกว่าหนึ่งรูปแบบ

หมายเหตุ 2:

1) ถ้า $|D| < \beta$ จำนวนจริงบางจำนวนไม่สามารถแสดงในระบบจำนวนนี้ได้

2) ถ้า $|D| = \beta$ จำนวนเต็มบวกทุกจำนวนสามารถแสดงแบบจำกัด และจำนวนจริงทุกจำนวนสามารถที่จะแสดงในระบบจำนวนนี้ได้

3) ถ้า $|D| > \beta$ ระบบจำนวนนี้จะเป็นระบบซ้ำซ้อน

นิยาม 2:

1) เซตจำกัดของตัวเลข D จะเป็นเซตของตัวเลขที่มีลักษณะซ้ำซ้อนได้น้อยที่สุด ถ้า $|D| = \beta+1$

2) เซตจำกัดของตัวเลข D จะเป็นเซตของตัวเลขที่มีลักษณะซ้ำซ้อนได้มากที่สุด ถ้า $|D| = 2\beta-1$

3) เซตจำกัดของตัวเลข D จะเป็นเซตของตัวเลขที่มีลักษณะสมมาตร ถ้า b = |a|

ตัวอย่างเช่น กำหนดให้เลขฐาน $\beta$ = 2 เซตจำกัดของตัวเลข D = {-1, 0, 1} ค่าตัวเลข 14 สามารถแทนค่าได้มากกว่าหนึ่งรูปแบบ ดังแสดงในรูปที่ 2.1 ค่าตัวเลข 14 สามารถเขียนได้เป็น $(001110)_2$ หรือ $(0100\text{-}10)_2$

| $\beta^5$ | $\beta^4$ | $\beta^3$ | $\beta^2$ | $\beta^1$ | $\beta^0$ |
|---|---|---|---|---|---|
| 32 | 16 | 8 | 4 | 2 | 1 |
| 0 | 0 | 1 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | -1 | 0 |

รูปที่ 2.1 ค่าตัวเลข 14 บนระบบจำนวนแบบมีเครื่องหมายฐานสอง

2.1.1.2 ระบบจำนวนซ้ำซ้อน (RNS: Redundant Number System)

ปัญหาที่มีความสำคัญต่อตัวทำดำเนินการ (Operator) ขั้นพื้นฐานที่ใช้ในระบบดิจิตอลที่เพิ่มมากขึ้นก็คือการกระจายของค่าที่เกิดจากการทด (Carry) ระบบจำนวนซ้ำซ้อนถูกนำมาใช้เพื่อแก้ปัญหาดังกล่าว ซึ่งมีหลักการคือ การแทนค่าของตัวเลขหนึ่งตัวด้วยภาคแสดง (Representation) มากกว่าหนึ่งรูปแบบ เพื่อหลีกเลี่ยงปัญหาการแพร่ตัวทด (Carry Propagation) ซึ่งระบบจำนวนแบบมีเครื่องหมายก็เป็นรูปแบบหนึ่งของระบบจำนวนซ้ำซ้อน [4]

ในกรณีนี้กำหนดให้เลขฐาน $\beta$ มีค่าเท่ากับ 2 และเซตจำกัดของตัวเลขคือ {-1, 0, 1} โดยการแสดงตัวเลขแต่ละตัวจะแสดงในระดับบิต เรียกว่า การแทนค่าแบบจำกัดการทด (BS: Borrow-Save) ซึ่งนิยามดังนี้

กำหนดให้แทนตัวเลขลำดับที่ i ($a_i$) ของจำนวน ๆ หนึ่ง (a) ด้วยจำนวนบิต 2 บิต ได้แก่ $a_i^+$ และ $a_i^-$ โดยที่ $a_i = a_i^+ - a_i^-$ ตัวอย่างเช่น

ตัวเลข 1 เมื่อแสดงตัวเลขในระดับบิต จะได้ (1, 0)

ตัวเลข -1 เมื่อแสดงตัวเลขในระดับบิต จะได้ (0, 1)

ตัวเลข 0 สามารถแสดงตัวเลขในระดับบิตได้สองแบบ คือ (0, 0) หรือ (1, 1)

เป็นต้น

2.1.2 การคำนวณทางคณิตศาสตร์แบบเชื่อมตรง (On-line Arithmetic Computation)

ตัวดำเนินการพื้นฐานสำหรับการคำนวณแบบลำดับส่วนใหญ่เช่น การบวก การลบ และการคูณ โดยปกติแล้วจะทำงานในรูปแบบที่เรียกว่า (LSDF: Least Significant Digit First) ซึ่งเป็นการคำนวณหาค่าจากหลักที่มีเลขนัยสำคัญต่ำสุดก่อน แต่การหารจะทำการในรูปแบบตรงกันข้าม ซึ่งเรียกว่า (MSDF: Most Significant Digit First) ซึ่งทำจากหลักที่มีเลขนัยสำคัญมากสุดก่อน จึงทำให้การคำนวณแบบลำดับไม่สามารถทำไปพร้อมกันทุกตัวดำเนินการได้ต้องรอให้การทำงานของตัวก่อนหน้าเสร็จก่อน ในกรณีที่ต้องการเพิ่มความเร็วในการทำงานโดยใช้แนวคิดของการคำนวณแบบท่อตรง (Pipeline) ก็ไม่สามารถทำได้ เพราะทุกตัวดำเนินการต้องทำงานในทิศทางเดียวกัน สำหรับการทำการในลักษณะท่อตรงนั้น แต่ละตัวดำเนินการสามารถเริ่มทำงานได้โดยไม่ต้องรอให้การทำงานของตัวดำเนินการก่อนหน้าเสร็จก่อน การคำนวณแบบเชื่อมตรงถูกออกแบบมาเพื่อใช้งานร่วมกับแนวคิดการทำท่อตรง แต่เนื่องจากการหารได้มีการพิสูจน์แล้วว่าไม่สามารถทำงานในแบบ LSDF ได้ ดังนั้น การคำนวณแบบเชื่อมตรงจึงเลือกใช้รูปแบบของ MSDF ทั้งหมด [5,6]

ในการคำนวณแบบเชื่อมตรง ตัวถูกดำเนินการ (Operand) และทิศทางของผลลัพธ์ที่ได้จะเป็นไปตามลำดับทางคณิตศาสตร์ผ่านทางหน่วยการคำนวณ ในลักษณะของตำแหน่งต่อตำแหน่ง (Digit By Digit) เริ่มจากตำแหน่งที่มีเลขนัยสำคัญสูงสุดก่อน การคำนวณแบบเชื่อมตรงจะมีคุณลักษณะอย่างหนึ่งที่เรียกว่า ค่าความหน่วงเชื่อมตรง (On-line Delay) แทนด้วยสัญลักษณ์ $\delta$ ซึ่งจะเป็นตัวเลขจำนวนเต็มบวกขนาดเล็กเป็นค่าที่ระบุว่าผลลัพธ์จำนวน n ตำแหน่งแรกนั้นสามารถคำนวณออกมาได้จากตัวถูกดำเนินการ $(n+\delta)$ ตำแหน่งแรก ดังรูปที่ 2.2

การกระทำดังกล่าวนี้ ระบบจำนวนที่ใช้ในการแสดงผลควรจะเป็นระบบซ้ำซ้อน ระบบจำนวนแบบมีเครื่องหมายจึงได้ถูกนำมาใช้กับการคำนวณแบบเชื่อมตรง

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Input x | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ | $x_6$ | $x_7$ |
| Input y | $y_1$ | $y_2$ | $y_3$ | $y_4$ | $y_5$ | $y_6$ | $y_7$ |
| Output z | $\leftarrow$ | $\delta$ | $\rightarrow$ | $z_1$ | $z_2$ | $z_3$ | $z_4$ |

รูปที่ 2.2 ตัวดำเนินการแบบเชื่อมตรงเมื่อมีค่าความหน่วงเชื่อมตรงเท่ากับ $\delta$

2.1.2.1 การบวกแบบเชื่อมตรง [7]

อัลกอริทึมข้างล่างนี้แสดงให้เห็นถึงลักษณะการทำงานของการบวกแบบเชื่อมตรง โดยกำหนดให้ $\beta$ เป็นเลขฐานของระบบจำนวนและ D เป็นเซตจำกัดของตัวเลข ดังนี้

Algorithm

Input : $X := (x_a x_{a-1}\ldots)_\beta$ and $Y := (y_a y_{a-1}\ldots)_\beta$ where $x_i , y_i \in D$

Output : $Z := (z_a z_{a-1}\ldots)_\beta$ where $z_i \in D$

begin

$r_{a+1} := 0;$

$j := a;$

while $j \leq a$ do

$s_j := x_j + y_j;$

if $-2b \leq s_j < -b+1$ then $c_{j+1} := -1;\ r_j := s_j + \beta;$ endif;

if $-b+1 \leq s_j \leq b-1$ then $c_{j+1} := 0;\ r_j := s_j;$ endif;

if $b-1 < s_j \leq 2b$ then $c_{j+1} := 1;\ r_j := s_j - \beta;$ endif;

$z_{j+1} := c_{j+1} + r_{j+1};$

$j := j - 1;$

enddo;

end;

ตัวอย่างเช่น กำหนดให้เลขฐานเท่ากับ $\beta$ = 5 และเซตจำกัดของตัวเลข D = {-3, -2, -1, 0, 1, 2, 3} ในการทำการบวกค่าของ 708 กับ 766 ดังแสดงในรูปที่ 2.3

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

| | | $\beta^5$ | $\beta^4$ | $\beta^3$ | $\beta^2$ | $\beta^1$ | $\beta^0$ | |
|---|---|---|---|---|---|---|---|---|
| 708 | = | | 1 | 1 | -2 | 1 | 3 | + |
| 766 | = | | 1 | 1 | 1 | -2 | 1 | |
| | | | 2 | 2 | -1 | -1 | 4 | |
| | | | 2 | 2 | -1 | -1 | -1 | |
| remainder carry digit | | 0 | 0 | 0 | 0 | 1 | | |
| | | 0 | 2 | 2 | -1 | 0 | -1 | |

รูปที่ 2.3 การบวกแบบเชื่อมตรงบนฐานห้าของ 708 และ 766

จะเห็นว่าคำตอบที่ได้มีค่าเท่ากับ 1474 แสดงได้โดย

$$(0\ 2\ 2\ -1\ 0\ -1)_5 = (2*5^4) + (2*5^3) + (-1*5^2) + (-1)$$
$$= 1250 + 250 - 25 - 1 = 1474$$

จากตัวอย่างดังรูปที่ 2.3 จะเห็นว่าผลลัพธ์ถูกสร้างขึ้นในลักษณะของตำแหน่งต่อตำแหน่ง เริ่มจากตำแหน่งที่มีเลขนัยสำคัญสูงสุดก่อน ด้วยค่าความหน่วงเชื่อมตรงมีค่าเท่ากับหนึ่งและ -1 จากตำแหน่ง $\beta^1$ ไม่สามารถที่จะสร้างผลลัพธ์ได้จนกว่าการกระจายของตัวทดของตำแหน่ง $\beta^0$ จะทราบค่า ในระบบจำนวนนี้ จะเห็นได้ว่าผลกระทบของการกระจายของตัวทดจะถูกจำกัดให้มีแค่เพียงตัวเดียวที่ตำแหน่งที่มีเลขนัยสำคัญต่ำสุด

2.1.2.2 การลบแบบเชื่อมตรง

สำหรับการลบแบบเชื่อมตรงใช้แนวคิดเดียวกันกับการบวกแบบเชื่อมตรง เพียงแต่พิจารณาค่าลบของตัวถูกดำเนินการในระหว่างการทำงานเท่านั้น

2.1.2.3 การคูณแบบเชื่อมตรง

การคูณแบบเชื่อมตรงเป็นการใช้สองเทคนิครวมเข้าด้วยกัน คือ การเพิ่มค่าขึ้นสำหรับการคูณ (Incremental Multiplication) และระบบจำนวนซ้ำซ้อน ซึ่งการคูณแบบเชื่อมตรง ค่าความหน่วงจะขึ้นอยู่กับขอบเขตของค่านำเข้า (Input) ค่าความหน่วงนี้สามารถที่จะไม่นำมาพิจารณาได้ โดยเติมค่าศูนย์เท่ากับจำนวนค่าความหน่วงให้กับแต่ละตัวถูกดำเนินการทาง

ด้านซ้าย นั่นคือทำให้แต่ละตัวถูกดำเนินการมีค่าน้อยกว่า $1/\beta^{\delta}$ ซึ่งจะเรียกว่า ขอบเขตของตัวถูกดำเนินการ (Operand Bound)

**Algorithm $M_R$**

**Input:** $A = (.a_{-1}a_{-2}\ldots)_\beta$ and $B := (.b_{-1}b_{-2}\ldots)_\beta$

**Output:** $X = (.x_{-1}x_{-2}\ldots)_\beta$ where $||X|| = \sum_{j\le -1} x_j\beta^j = ||A|| \,.\, ||B||$

**begin**

$x_{-1} := x_{-2} := \ldots := x_{-\delta} := 0;$

$W_{-\delta} := 0;$ $\quad j := -\delta-1;$

**while** $j \le -\delta-1$ **do**

$W_j := \beta(W_{j+1} - x_{j+1}) + A_jb_j + B_{j+1}a_j;$

**if** $|W_j| \le b$ **then** $x_j := \text{Sign}(W_j)\lfloor |W_j| + 1/2 \rfloor$

**else** $x_j := \text{Sign}(W_j)\lfloor |W_j| \rfloor$ **end if;**

$j := j - 1;$

**enddo;**

**end;**

อัลกอริทึมข้างบนจะแสดงให้เห็นถึงลักษณะการทำงานของการคูณแบบเชื่อมตรง โดยกำหนดให้ $\beta$ เป็นเลขฐานของระบบจำนวนและ D เป็นเซตจำกัดของตัวเลข มีค่าเท่ากับ {-b, -b+1, -b+2, …, 0, 1, …, b

ตัวอย่างเช่น กำหนดให้ A และ B มีเลขฐานเท่ากับ $\beta$ = 2 ซึ่ง A = B = $(.0011011111)_2$ ผลลัพธ์จากการคำนวณตามอัลกอริทึมข้างบนแสดงดังตารางที่ 2.1

ตารางที่ 2.1 การคูณแบบเชื่อมตรงบนฐานสอง

| J | $A_jb_j + B_{j+1}a_j$ | Wj | xj | 2(Wj - xj) |
|---|---|---|---|---|
| -1 | 0.0 | 0.0 | 0 | 0.0 |
| -2 | 0.0 | 0.0 | 0 | 0.0 |
| -3 | 0.001 | 0.001 | 0 | 0.01 |
| -4 | 0.0101 | 0.1001 | 1 | -0.111 |

ตารางที่ 2.1 การคูณแบบเชื่อมตรงบนฐานสอง (ต่อ)

| J | $A_jb_j + B_{j+1}a_j$ | Wj | xj | 2(Wj - xj) |
|---|---|---|---|---|
| -5 | 0.0 | -0.111 | -1 | 0.01 |
| -6 | 0.011001 | 0.101001 | 1 | -0.10111 |
| -7 | 0.0110101 | -0.0100111 | 0 | -0.100111 |
| -8 | 0.01101101 | -0.00101111 | 0 | -0.0101111 |
| -9 | 0.011011101 | 0.000100001 | 0 | 0.00100001 |

ผลลัพธ์ที่ได้จากการคำนวณ คือ $(.0001\text{-}11000\ldots.)_2$

2.1.3 เอ็กซ์เอ็มแอล (XML: Extensible Markup Language) [8] [9] [10]

เอ็กซ์เอ็มแอลเป็นภาษาที่ใช้อธิบายข้อมูล ได้รับการออกแบบให้มีความสามารถที่จะอธิบายความหมายของตัวเองได้หรือนิยามข้อมูลได้ ทำให้ยืดหยุ่น ผู้พัฒนาสามารถขยายข้อมูลเพิ่มเติมได้มากเท่าที่ต้องการทำให้เอ็กซ์เอ็มแอลกลายเป็นภาษามาตรฐานเพื่ออธิบายข้อมูลสำหรับการแลกเปลี่ยนข้อมูลระหว่างแพลตฟอร์มและเทคโนโลยีที่แตกต่างกัน เอกสารเอ็กซ์เอ็มแอลมีกฎเกณฑ์โครงสร้างและการตรวจสอบความถูกต้อง สรุปได้ดังนี้

<u>กฎเกณฑ์เบื้องต้นสำหรับการสร้างเอกสารเอ็กซ์เอ็มแอล</u>

1) อีลีเมนต์ ประกอบด้วยแท็กเปิด แท็กปิดและเนื้อหา เช่น

```
<CIRCUITMEMENTO>
    <COMPONENTS>
        <COMPONENT ID="0" TYPE="NOT" PINCOUNT="1">
            <POSITION X="340" Y="100"/>
        </COMPONENT>
    </COMPONENTS>
</CIRCUITMEMENTO>
```

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

2) อีลีเมนต์ จะต้องซ้อนกันเป็นลำดับ เช่น

```
<CIRCUITMEMENTO>
    <COMPONENTS>
        <COMPONENT ID="0" TYPE="NOT" PINCOUNT="1">
            <POSITION X="360" Y="100"/>
        </COMPONENT>
        <COMPONENT ID="1" TYPE="AND" PINCOUNT="2">
            <POSITION X="260" Y="100"/>
        </COMPONENT>
    </COMPONENTS>
</CIRCUITMEMENTO>
```

3) แท็กของเอ็กซ์เอ็มแอลนั้น ตัวอักษรตัวใหญ่และตัวเล็กมีความแตกต่างกัน (Case Sensitive)

4) อีลีเมนต์ว่าง คือ อีลีเมนต์ที่ไม่มีเนื้อหา สามารถเขียนได้ 2 แบบ คือ

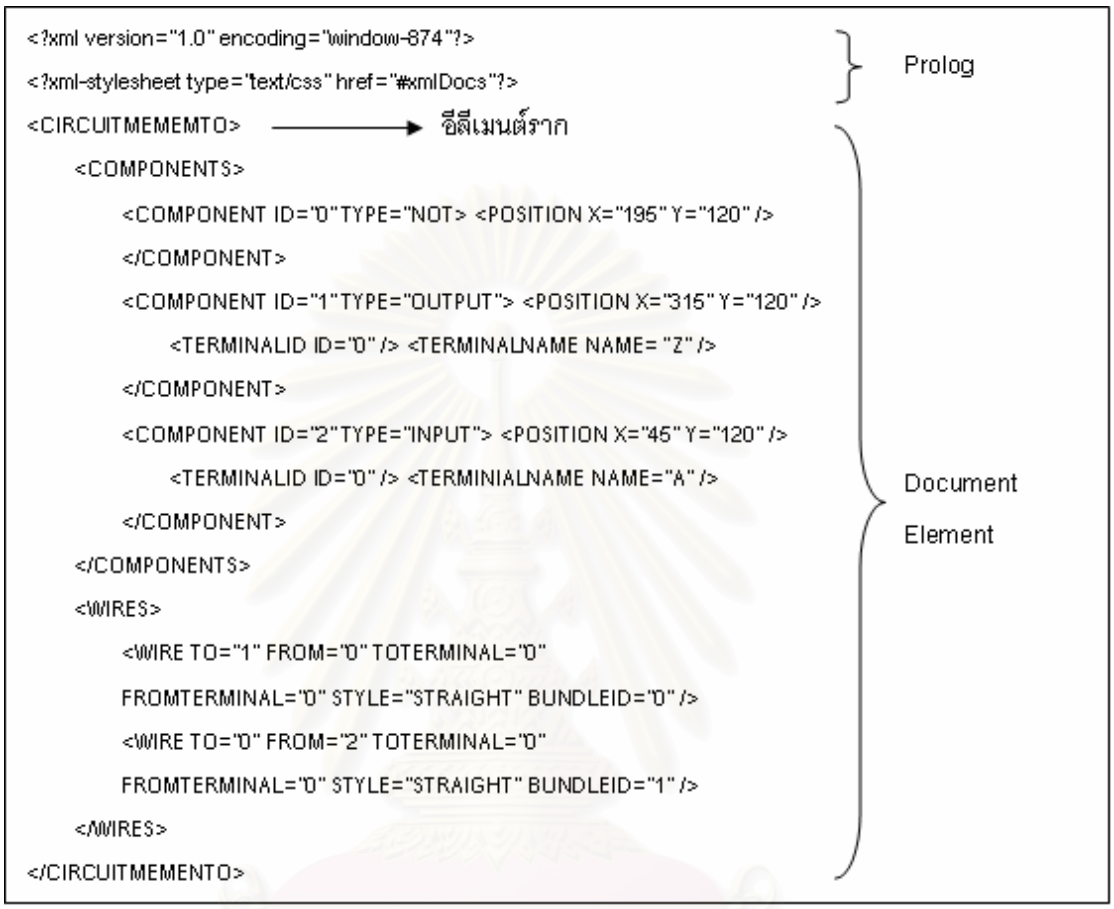
<COMPONENTS></COMPONENTS> หรือ <COMPONENTS/>

5) เอกสารเอ็กซ์เอ็มแอลมีอีลีเมนต์ราก (Root Element) เพียงอีลีเมนต์เดียวส่วนอีลีเมนต์ลูกอื่น ๆ จะซ้อนกันเป็นลำดับชั้นภายในอีลีเมนต์ราก

<u>โครงสร้างของเอกสารเอ็กซ์เอ็มแอล</u>

เอกสารเอ็กซ์เอ็มแอลประกอบด้วย 2 ส่วนหลัก คือ การประกาศส่วนที่เรียกว่า Prolog หรือ XML Declaration หมายถึงส่วนหัวของเอกสารเอ็กซ์เอ็มแอลที่ต้องการใช้งานและ

ส่วนที่ 2 คือส่วนพื้นที่แสดงแหล่งข้อมูลเอกสารเอ็กซ์เอ็มแอล เรียกว่า Document Elements ซึ่งประกอบด้วยอีลีเมนต์ราก และอีลีเมนต์ลูก ดังแสดงในรูปที่ 2.4




รูปที่ 2.4 โครงสร้างของเอกสารเอ็กซ์เอ็มแอล

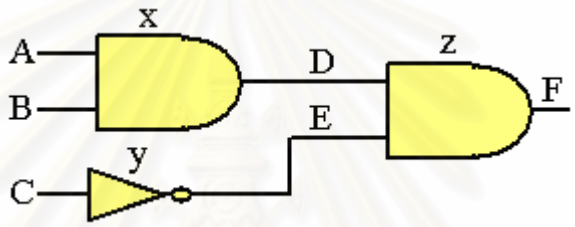
2.1.4 การจำลองการทำงานตามสถานการณ์ที่เกิด (Event-Driven Simulation)

การจำลองการทำงานของวงจรด้วยคอมพิวเตอร์มีหลายวิธี วิธีหนึ่งที่เหมาะสม คือ การจำลองการทำงานตามสถานการณ์ที่เกิด [11] วิธีนี้อาศัยการเปลี่ยนแปลงที่เกิดขึ้นของระดับลอจิกเป็นตัวกำหนดการทำงาน การเปลี่ยนแปลงทางลอจิกส่งผลให้เครื่องมือจำลองการทำงานปฏิบัติหน้าที่ซึ่งอาจเกิดขึ้นทันที หรือรอเวลาระยะหนึ่งก่อนจึงจะเกิดการทำงานขึ้นจริง

วงจรจริง ๆ ประกอบไปด้วยวงจรย่อยหลาย ๆ วงจรนำมาประกอบเข้าด้วยกัน จึงจำเป็นต้องเก็บการเชื่อมโยงกันของวงจรย่อยต่าง ๆ เอาไว้เพื่อที่จะสามารถส่งการเปลี่ยนแปลงของข้อมูลนำเข้าไปให้วงจรอื่นที่เกี่ยวข้องกับข้อมูลนำเข้านั้นได้ โดยทั่วไปแล้วผลลัพธ์ของวงจรหนึ่งก็จะเป็นข้อมูลนำเข้าของวงจรอื่น ๆ และข้อมูลนำเข้าของวงจรนั้น ๆ ก็มักจะต่อกับผลลัพธ์จากวงจร

อื่นเช่นกัน สิ่งจำเป็นคือต้องมีวิธีการที่เหมาะสมสำหรับเก็บการเชื่อมต่อกันของวงจรต่าง ๆ เอาไว้เพื่อที่จะสามารถส่งต่อส่วนข้อมูลนำเข้าระหว่างวงจรได้ถูกต้องและมีประสิทธิภาพ

จากตัวอย่างในรูปที่ 2.5 การเปลี่ยนแปลงของข้อมูลนำเข้าที่จุด A หรือจุด B ซึ่งเป็นส่วนนำเข้าของ x อาจทำให้เกิดการเปลี่ยนแปลงของข้อมูลนำเข้าที่จุด D ซึ่งเป็นผลลัพธ์ของ x และก็เป็นส่วนนำเข้าของ z ด้วย การเปลี่ยนแปลงนี้อาจทำให้เกิดการเปลี่ยนแปลงของข้อมูลนำเข้าที่จุด F ได้ในเวลาต่อมา วิธีการจำลองการทำงานตามสถานการณ์ที่เกิดนี้จะอาศัยการเปลี่ยนแปลงนี้ในการเลือกว่าจะต้องมีการจำลองการทำงานที่จุดใดของวงจร เนื่องจากวิธีนี้จะทำงานกับเฉพาะส่วนที่เกิดการเปลี่ยนแปลงขึ้นของวงจรเท่านั้น วิธีนี้จึงเป็นวิธีที่มีประสิทธิภาพมาก




รูปที่ 2.5 ตัวอย่างวงจรแสดงความสัมพันธ์ของข้อมูลนำเข้า

## 2.2 เอกสารและงานวิจัยที่เกี่ยวข้อง

การศึกษาในปัจจุบัน โปรแกรมสำหรับออกแบบและจำลองการทำงานของวงจรดิจิตอล คือทางเลือกหนึ่งในการนำมาใช้เสริมสร้างประสบการณ์ให้กับนักเรียนและอาจารย์ในการแลกเปลี่ยนประสบการณ์เพื่อพัฒนาการเรียนรู้ให้เข้าใจการวางแผนจำลองการออกแบบของจริง ในปัจจุบันมีโปรแกรมที่สามารถใช้งานผ่านทางอินเตอร์เน็ตได้โดยง่าย ทำให้การเรียนรู้ไม่จำเพาะแต่ภายในห้องเรียนเท่านั้น

### 2.2.1 เครื่องมือ “DLSim” [1]

DLSim เป็นเครื่องมือที่พัฒนาขึ้นเพื่อเป็นเครื่องมือทางการศึกษา (Educational Tool) และยังเป็นโปรแกรมประยุกต์บนเว็บ ซึ่งมีลักษณะของส่วนต่อประสานกับผู้ใช้ (User Interface) เป็นแบบกราฟิกที่ใช้งานง่าย สะดวกแก่การออกแบบและทดสอบจำลองการทำงานวงจรดิจิตอล เป็นเครื่องมือที่ผู้ออกแบบสามารถสร้างวงจรขึ้นมาและสามารถนำวงจรนั้นไปใช้ประกอบร่วมกับวงจรที่สร้างใหม่ได้โดยง่าย นอกจากนี้ DLSim ยังเป็นเครื่องมือที่เก็บบันทึกวงจรที่

ออกแบบลงไฟล์ในรูปแบบแฟ้มเอกสารเอ็กซ์เอ็มแอลอีกด้วย แต่มีข้อกำจัดอยู่ที่ไม่สามารถพิมพ์วงจรที่ออกแบบออกทางเครื่องพิมพ์ได้

เครื่องมือนี้ถูกพัฒนาด้วยโปรแกรมภาษาจาวาที่ใช้งานในรูปแบบของแอพเพล็ทและแอพลิเคชั่น โดยผู้ออกแบบวงจรสามารถใช้งานเครื่องมือนี้ได้ภายในทุกสภาพแวดล้อมที่มี Java Virtual Machine (JVM)

2.2.2 เครื่องมือ “Logisim” [2]

Logisim เป็นเครื่องมือที่พัฒนาขึ้นโดยนาย Carl Burch จากมหาวิทยาลัย St John’s University, Collegeville MN มีวัตถุประสงค์เพื่อให้ผู้ออกแบบวงจรได้ศึกษาถึงการออกแบบและทดสอบจำลองการทำงานของวงจรลอจิกต่าง ๆ (Logic Circuit) ได้โดยง่ายด้วยคอมพิวเตอร์แทนการออกแบบวงจรขึ้นจริง ๆ เป็นเครื่องมือทางการศึกษา (Educational Tool) ที่ใช้กันอย่างแพร่หลายในวงการการศึกษาทั้งในระดับวิทยาลัยและมหาวิทยาลัยทั่วโลก เนื่องจากใช้งานง่าย การทำงานของเครื่องมือจะทำบนสภาพแวดล้อมแบบกราฟิกเพื่อความสะดวกในการจำลองการทำงาน เหมาะสำหรับการเรียนรู้พื้นฐานด้านวงจรลอจิกต่าง ๆ ตลอดจนการออกแบบสร้างวงจรซีพียูอย่างง่าย โดยที่ความสามารถของเครื่องมือนี้รวมไปถึงการเก็บบันทึกวงจรที่ออกแบบลงไฟล์ในรูปแบบของแฟ้มรูปภาพ (GIF File) และการเก็บวงจรที่สร้างนั้นเป็นเสมือนวงจรย่อย (Sub-circuit) อันหนึ่งที่สามารถนำกลับมาใช้ร่วมในการออกแบบสร้างวงจรใหม่ได้อีกด้วย แต่มีข้อจำกัดอยู่ที่วงจรย่อยต่าง ๆ ที่ออกแบบนั้นจะสามารถใช้งานได้เพียงในแฟ้มเอกสารเดียวกันกับวงจรใหม่ที่ออกแบบโดยไม่สามารถเรียกใช้งานข้ามแฟ้มเอกสารได้

เครื่องมือนี้ถูกพัฒนาด้วยโปรแกรมภาษาจาวาที่สามารถใช้งานได้ทั้งบนระบบปฏิบัติการวินโดวส์และยูนิกซ์ เป็นเครื่องมือที่มีการพัฒนาปรับปรุงให้มีความทันสมัยอย่างต่อเนื่อง

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

# บทที่ 3

# แนวคิดและวิธีวิจัย

ในบทนี้ ผู้วิจัยได้เสนอแนวคิดในการออกแบบและพัฒนาเครื่องมือ วิเคราะห์องค์ประกอบของเครื่องมือที่พัฒนา อธิบายแนวทางการประยุกต์เครื่องมือที่พัฒนากับอัลกอริทึมการคำนวณแบบเชื่อมตรง ตลอดจนสรุปลักษณะการประยุกต์ใช้งานเครื่องมือบนระบบอินเตอร์เน็ต โดยมีรายละเอียดดังนี้

## 3.1 แนวคิดในการออกแบบและพัฒนาเครื่องมือ

วิทยานิพนธ์นี้มุ่งเน้นการออกแบบและพัฒนาเครื่องมือที่ใช้สำหรับจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรง โดยพิจารณาว่า เครื่องมือที่มีอยู่ทั่วไปที่ใช้ในการออกแบบสร้างวงจรนั้น ผู้ออกแบบจะต้องให้ความสำคัญในระดับเกตย่อย ๆ และอธิบายโดยใช้ตรรกะของเกต นอกจากนี้การออกแบบในระดับตรรกะนั้น เครื่องมือที่มีอยู่เน้นที่การออกแบบที่เหมือนกับวงจรจริงเพื่อประโยชน์ในการสร้างฮาร์ดแวร์ ทำให้วงจรอาจมีความซับซ้อนจนทำให้การมองภาพในระดับของการคำนวณเข้าใจยากหรืออาจเห็นได้ไม่ชัดเจน แต่ในแง่ของนักออกแบบอัลกอริทึม การสร้างอัลกอริทึมของการคำนวณแบบต่าง ๆ แต่ละส่วนของการคำนวณนักออกแบบมิได้ต้องการทราบถึงการทำงานในระดับตรรกะเพียงแต่ต้องการทราบในภาพรวมของการทำงานในแต่ละส่วนและสนใจการติดต่อกัน (Communication) ของแต่ละส่วน

สำหรับการออกแบบอัลกอริทึมหรืออีกนัยหนึ่งการออกแบบสร้างวงจรสำหรับการคำนวณแบบเชื่อมตรงนั้น เนื่องจากลักษณะการทำงานมีความแตกต่างจากวงจรทั่วไปเพราะ การคำนวณมีลักษณะเป็นแบบวนซ้ำ (Recursion) และใช้หลักการทำงานของการประมวลผลแบบท่อตรง วิทยานิพนธ์นี้จึงต้องการสร้างเครื่องมือสำหรับการทำงานดังกล่าวให้รองรับและเหมาะสมกับการทำงานแบบเชื่อมตรงดังกล่าว นอกจากนี้ การแสดงผลการจำลองการทำงานของอัลกอริทึมยังเป็นสิ่งสำคัญที่ทำให้นักออกแบบสามารถเข้าใจและตรวจสอบความถูกต้องได้ง่ายขึ้น

ผู้วิจัยสนใจปัญหาการทดสอบการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรง จึงเกิดแนวคิดที่จะแก้ปัญหาโดยการออกแบบและพัฒนาเครื่องมือสำหรับจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรงในรูปของวงจรย่อยต่าง ๆ ประกอบเข้าด้วยกัน ซึ่งกำหนดการทำงานได้ด้วยตารางค่าความจริง โดยมุ่งที่จะพัฒนาเพื่อให้ใช้งานได้ง่ายและเป็นประโยชน์แก่ผู้ที่

ต้องการศึกษาการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรง ซึ่งสามารถวิเคราะห์ความต้องการพื้นฐานสำหรับการออกแบบและพัฒนาเครื่องมือ ได้ดังนี้

ความต้องการพื้นฐานสำหรับการออกแบบและพัฒนาเครื่องมือ

1) เครื่องมือที่ได้ทำงานบนสภาพแวดล้อมแบบกราฟิกโหมดเพื่อให้ผู้ออกแบบสามารถจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรงได้สะดวก ซึ่งทำงานเป็นลักษณะของเว็บแอพลิเคชั่นที่ใช้งานได้ในรูปแบบของแอพเพล็ท

2) เครื่องมือที่ได้ประกอบด้วยส่วนการทำงานสำคัญ 2 ส่วน คือ ส่วนการออกแบบสำหรับการออกแบบสร้างอุปกรณ์หรือวงจร และส่วนการจำลองการทำงานสำหรับการทดสอบการจำลองการทำงานของอัลกอริทึม

3) เครื่องมือที่ได้สามารถนำอุปกรณ์หรือวงจรที่เคยสร้างเก็บไว้มาประกอบเป็นชิ้นส่วนหนึ่งของวงจรที่ออกแบบใหม่ได้ โดยอุปกรณ์หรือวงจรเหล่านั้นถูกบันทึกเก็บลงไฟล์ในรูปแบบของแฟ้มเอกสารเอ็กซ์เอ็มแอล

4) เครื่องมือที่ได้สามารถเลือกออกแบบสร้างวงจรได้ 2 วิธี คือ การออกแบบสร้างวงจรด้วยสมการบูลีน (Designing with Boolean Equations) และการออกแบบสร้างวงจรด้วยการวาดเป็นผังวงจร (Schematic-based Design)

5) เครื่องมือที่ได้สามารถเลือกออกแบบสร้างอุปกรณ์ขึ้นมาใหม่ได้ 2 ประเภท คือ การออกแบบสร้างอุปกรณ์พื้นฐานที่กำหนดโดยผู้ออกแบบเอง (UDP: User-defined Primitive) หรือเรียกว่าอุปกรณ์ UDP โดยสามารถกำหนดฟังก์ชั่นการทำงานของอุปกรณ์ด้วยค่าความหน่วงและตารางค่าความจริงที่ผู้ออกแบบเป็นผู้กำหนดค่าผลลัพธ์เอง และการออกแบบสร้างอุปกรณ์ที่กำหนดด้วยสมการทางคณิตศาสตร์ (UDE: User-defined Expression) หรือเรียกว่าอุปกรณ์ UDE ซึ่งกำหนดฟังก์ชั่นการทำงานของอุปกรณ์ด้วยตารางค่าความจริง

6) อุปกรณ์ที่จะนำมาประกอบกันในวงจรได้นั้นมี 2 ประเภทด้วยกัน ประเภทแรก คือ อุปกรณ์พื้นฐาน ได้แก่ อุปกรณ์เกตพื้นฐานต่าง ๆ (Predefined Gate Primitives) ที่เครื่องมือสร้างเตรียมไว้แล้ว ประเภทที่สอง คือ อุปกรณ์ประกอบ ได้แก่ วงจรย่อยต่าง ๆ (Sub-circuits) ที่เกิดจากการนำอุปกรณ์พื้นฐานและ/หรืออุปกรณ์ประกอบมาประกอบเข้าด้วยกันแล้วบันทึกเก็บไว้อุปกรณ์ UDP และอุปกรณ์ UDE ซึ่งเป็นอุปกรณ์ที่ผู้ออกแบบสร้างขึ้นใหม่

7) เครื่องมือที่ได้สามารถเลือกป้อนค่าตัวเลขให้กับตัวแปรนำเข้าชนิดต่าง ๆ เพื่อทดสอบดูผลลัพธ์ของการจำลองการทำงานได้ 2 รูปแบบ คือ ป้อนค่าด้วยตัวเลขฐานสิบหรือตัวเลขฐานสอง โดยเครื่องมือจะทำการแปลงค่าที่ถูกต้องของอีกฐานให้ควบคู่กันไปด้วย

8) เครื่องมือที่ได้สามารถเลือกรูปแบบการแสดงผลลัพธ์ของการจำลองการทำงานได้ 2 รูปแบบ คือ เลือกให้การแสดงผลลัพธ์เกิดขึ้นอย่างต่อเนื่องจนแล้วเสร็จหรือเกิดขึ้นทีละขั้นโดยผู้ออกแบบเป็นผู้ควบคุมการทำงานผ่านทางส่วนต่อประสานกับผู้ใช้

9) เครื่องมือที่ได้จะแสดงผลลัพธ์ของการจำลองการทำงานผ่านทางส่วนต่อประสานกับผู้ใช้ใน 2 รูปแบบ คือ ผลลัพธ์ที่แสดงในรูปของกราฟิกซึ่งถูกแสดงในแต่ละขั้นตอนผ่านทางส่วนวาดวงจร และผลลัพธ์ทั้งหมดในรูปแบบของเลขฐานสองและเลขฐานสิบผ่านทางส่วนแสดงค่าผลลัพธ์

## 3.2 การวิเคราะห์องค์ประกอบของเครื่องมือ

จากแนวคิดในการออกแบบและพัฒนาเครื่องมือ ตลอดจนการวิเคราะห์ความต้องการพื้นฐานสำหรับการออกแบบและพัฒนาเครื่องมือข้างต้น ผู้วิจัยได้นำมาวิเคราะห์หาองค์ประกอบของเครื่องมือที่ต้องการพัฒนาว่าควรมีส่วนประกอบต่าง ๆ เป็นอย่างไรบ้างเพื่อประโยชน์สำหรับขั้นตอนการออกแบบและพัฒนาเครื่องมือต่อไป โดยผู้วิจัยได้ทำการแบ่งองค์ประกอบของเครื่องมือตามลักษณะการใช้งานและการทำงานออกเป็น 4 ส่วน ดังนี้

### 3.2.1 ส่วนการออกแบบ (Design Entry)

ส่วนการออกแบบจะเป็นส่วนที่เกี่ยวข้องกับการออกแบบสร้างอุปกรณ์หรือวงจร ตลอดจนการบันทึกเก็บอุปกรณ์หรือวงจรที่สร้างขึ้นให้อยู่ในรูปแบบของแฟ้มเอกสารเอ็กซ์เอ็มแอล และสามารถเรียกอุปกรณ์หรือวงจรเหล่านั้นกลับมาใช้สำหรับการแก้ไขหรือทดสอบจำลองการทำงานได้อีกด้วย เพื่อให้ส่วนการออกแบบสามารถทำงานดังกล่าวข้างต้นได้ทั้งหมด ผู้วิจัยได้ทำการแบ่งส่วนการออกแบบนี้ ออกเป็นส่วนย่อย 2 ส่วน คือ

#### 3.2.1.1 ส่วนวาดวงจร (Schematic Area)

ส่วนวาดวงจรเป็นส่วนสำคัญสำหรับการออกแบบสร้างอุปกรณ์หรือวงจร โดยผู้ออกแบบสามารถเลือกออกแบบสร้างวงจรได้ 2 วิธี คือ การออกแบบสร้างวงจรด้วยสมการบูลีนโดยใช้วิธีการแบบผลรวมของผลคูณ (SOP: Sum of Product) และการออกแบบสร้างวงจรด้วย

การวาดเป็นผังวงจรโดยเลือกอุปกรณ์ที่มีอยู่จากส่วนคลังอุปกรณ์มาวางและเชื่อมโยงอุปกรณ์เหล่านั้นประกอบเข้าด้วยกันเป็นวงจร โดยพื้นที่การทำงานส่วนวาดวงจรนี้จะถูกแสดงด้วยกริด (Grid) และการจัดวางอุปกรณ์ต่าง ๆ จะกระทำในรูปแบบลากแล้วปล่อย (Drag-and-Drop) นอกจากนี้การออกแบบสร้างอุปกรณ์ UDP และอุปกรณ์ UDE ขึ้นมาใหม่ก็จะถูกแสดงบนพื้นที่ส่วนนี้ด้วยเช่นกัน ผู้ออกแบบสามารถบันทึกอุปกรณ์ UDP และอุปกรณ์ UDE หรือวงจรย่อยต่าง ๆ ที่ออกแบบเก็บไว้ในคลังอุปกรณ์ในส่วนของอุปกรณ์ประกอบได้อีกด้วย เพื่อสามารถนำกลับมาใช้ได้ในภายหลัง รวมถึงการนำอุปกรณ์หรือวงจรที่บันทึกเก็บไว้มาประกอบเป็นชิ้นส่วนหนึ่งของวงจรที่ออกแบบใหม่ได้ โดยอุปกรณ์หรือวงจรเหล่านี้จะถูกบันทึกเก็บไว้ในรูปแบบของแฟ้มเอกสารเอ็กซ์เอ็มแอล สามารถอธิบายลักษณะการออกแบบสร้างวงจร 2 วิธีดังกล่าวข้างต้น ได้ดังนี้

1) การออกแบบด้วยสมการบูลีน (Designing with Boolean Equations)

การออกแบบสร้างวงจรด้วยสมการบูลีนนั้นต้องการสมการสำหรับการกำหนดการใช้งานของแต่ละอุปกรณ์เกตพื้นฐาน โดยสามารถเขียนสมการบูลีนได้โดยวิธีการแบบผลรวมของผลคูณ ซึ่งเป็นการแสดงค่าผลคูณของตัวแปรนำเข้าตั้งแต่ 2 ตัวขึ้นไปด้วยฟังก์ชัน AND แล้วนำผลคูณแต่ละส่วนมารวมกันโดยใช้ฟังก์ชัน OR เช่น AB+~C+DE=Y เป็นต้น ซึ่งสมการบูลีนที่เขียนโดยวิธีการดังกล่าวนี้ ยังคงเป็นสมการพื้นฐานที่คิดจากค่าเอาท์พุทที่มีค่า 1 โดยพิจารณาแยกสำหรับแต่ละค่าของเอาท์พุทที่เป็น 1 ว่าเกิดจากอินพุทค่าพื้นฐานใด (ค่า 0 หรือ 1) แล้วนำค่าที่ได้มา AND กัน เมื่อได้ครบทุกเทอมสำหรับค่าเอาท์พุทที่เป็น 1 ให้นำเทอมทั้งหมดมา OR กันก็จะได้สมการบูลีนตามต้องการ โดยเครื่องมือจะทำการวาดวงจรด้วยรูปที่สอดคล้องกันกับสมการบูลีนที่กำหนดลงบนส่วนวาดวงจร โดยที่ผู้ออกแบบไม่สามารถนำอุปกรณ์ต่าง ๆ จากคลังอุปกรณ์มาประกอบเพิ่มเข้าไปได้และตารางค่าความจริงสำหรับวงจรที่สร้างขึ้นจากสมการบูลีนนี้ก็จะเป็นไปตามจริงของการประกอบเข้าด้วยกันของอุปกรณ์เกตพื้นฐานต่าง ๆ ที่เครื่องมือวาดขึ้นและแก้ไขค่าในตารางค่าความจริงไม่ได้แต่ผู้ออกแบบสามารถดูค่าในตารางค่าความจริงของอุปกรณ์ต่าง ๆ ที่เครื่องมือวาดขึ้นได้ สมการบูลีนจะประกอบไปด้วย ตัวแปรนำเข้าและตัวกระทำทางตรรกะพื้นฐาน ซึ่งสามารถอธิบายได้ดังนี้

<u>ตัวแปรนำเข้า</u> เป็นตัวแปรที่รับค่าเพียงสองค่า หรือเรียกอีกอย่างหนึ่งว่า ตัวแปรสองสถานะ (Bi-state Variable) โดยมีข้อกำหนดคือ สามารถมีสถานะได้เพียงสองสถานะเท่านั้น และจะอยู่ในสถานะใดสถานะหนึ่งเท่านั้นจะอยู่พร้อมกันทั้งสองสถานะในเวลาเดียวกันไม่ได้ สถานะดังกล่าวคือ 1 หรือ 0 นั่นเอง

ตัวกระทำทางตรรกะพื้นฐาน เป็นตัวรับเอาตัวแปรนำเข้ามาดำเนินการเพื่อให้ได้ผลลัพธ์ โดยผลลัพธ์ที่ได้ขึ้นอยู่กับชนิดของตัวกระทำและสถานะของตัวแปรนำเข้าที่ถูกกระทำ

ตัวแปรนำเข้าหนึ่งตัว สามารถทำให้เกิดสถานะที่แตกต่างกันได้ 2 กรณี เช่น ตัวแปรนำเข้า A มีสถานะที่แตกต่างกันได้ 2 กรณี คือ A = 0 หรือ A = 1 เมื่อเพิ่มจำนวนตัวแปรนำเข้าเป็น 2 ตัว เช่น A และ B สถานะที่แตกต่างกันจะเพิ่มเป็น 4 กรณี หรือ $2^2$ กรณี คือ A = 0 , B = 0 หรือ A = 0 , B = 1 หรือ A = 1 , B = 0 และ A = 1 , B = 1 เป็นต้น

ดังนั้นถ้ามีตัวแปรนำเข้าจำนวน n ตัว จะมีสถานะที่แตกต่างกันทั้งหมด $2^n$ กรณี สำหรับตัวกระทำทางตรรกะพื้นฐานที่ผู้วิจัยได้ออกแบบเพื่อให้ใช้ได้สำหรับการกำหนดสมการบูลีนนั้นจะประกอบไปด้วย AND, OR และ NOT เท่านั้น โดยกฎเกณฑ์ในการกำหนดสมการบูลีนสามารถสรุปได้ดังนี้

กฎเกณฑ์ในการกำหนดสมการบูลีนสำหรับการออกแบบสร้างวงจร

1. สามารถกำหนดการกระทำสำหรับตัวกระทำทางตรรกะพื้นฐาน AND ได้ด้วยการเขียนเป็นสมการ ดังนี้ AB หรือ CD เป็นต้น

2. สามารถกำหนดการกระทำสำหรับตัวกระทำทางตรรกะพื้นฐาน OR ได้ด้วยการเขียนเป็นสมการ ดังนี้ A+B หรือ C+D เป็นต้น

3. สามารถกำหนดการกระทำสำหรับตัวกระทำทางตรรกะพื้นฐาน NOT ได้ด้วยการเขียนเป็นสมการ ดังนี้ ~A หรือ ~B เป็นต้น

ตารางที่ 3.1 ตารางค่าความจริงตามสมการบูลีน AB=Z

| Input | | Output |
|---|---|---|
| A | B | Z=AB |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

สำหรับวงจรที่ออกแบบจากสมการบูลีนนั้นจะทำงานตามตารางค่าความจริงที่ได้จากสมการบูลีนที่กำหนดนั่นเอง โดยตารางค่าความจริงเป็นตารางที่แสดงความสัมพันธ์ค่าตรรกะ

ระหว่างตัวแปรนำเข้าและตัวแปรผลลัพธ์ที่เป็นไปได้ทั้งหมดที่เกิดจากสมการบูลีนดังกล่าว ตัวอย่างตารางค่าความจริงที่ได้จากการกำหนดสมการบูลีน AB=Z แสดงดังตารางที่ 3.1

2) การออกแบบด้วยการวาดเป็นผังวงจร (Schematic-based Design)

การออกแบบด้วยการวาดเป็นผังวงจร ช่วยในการขยายความสามารถในการออกแบบด้วยสมการบูลีน เนื่องจากไม่ใช่มีแค่อุปกรณ์เกตที่เป็นอุปกรณ์พื้นฐานเท่านั้นแต่ยังมีอุปกรณ์หรือวงจรย่อยต่าง ๆ ที่อยู่ในส่วนของอุปกรณ์ประกอบด้วย นอกจากนั้นวิธีนี้ยังสามารถออกแบบวงจรเป็นลักษณะของลำดับชั้น (Hierarchy) ได้อีกด้วย ซึ่งทำให้สามารถออกแบบวงจรที่ซับซ้อนได้และสามารถประหยัดเวลาในการออกแบบเนื่องจากสามารถมองเห็นเป็นรูปร่างของวงจรที่ชัดเจนว่าอุปกรณ์ตัวไหนต่อกันอย่างไรและสามารถเรียกใช้งานอุปกรณ์หรือวงจรย่อยต่าง ๆ ได้หลาย ๆ ครั้งอีกด้วย

สำหรับการออกแบบสร้างวงจรด้วยการวาดเป็นผังวงจร ผู้ออกแบบสามารถเลือกอุปกรณ์ต่าง ๆ จากส่วนคลังอุปกรณ์มาวางและเชื่อมโยงอุปกรณ์เหล่านั้นเข้าเป็นวงจรได้

3.2.1.2 ส่วนคลังอุปกรณ์ (Available Component)

ส่วนคลังอุปกรณ์ คือส่วนที่เก็บอุปกรณ์หรือวงจรย่อยต่าง ๆ เอาไว้ โดยที่เครื่องมือสร้างส่วนคลังอุปกรณ์นี้เพื่อให้ผู้ออกแบบสามารถเลือกนำอุปกรณ์หรือวงจรย่อยต่าง ๆ ที่มีอยู่ไปใช้ในการออกแบบสร้างวงจรใหม่ได้สะดวกหรือแม้แต่เลือกอุปกรณ์หรือวงจรย่อยต่าง ๆ เหล่านั้นเพื่อมาทดสอบจำลองการทำงานก็ตาม ซึ่งในส่วนคลังอุปกรณ์นี้จะประกอบไปด้วยอุปกรณ์ 3 ประเภทด้วยกัน คือ

1) อุปกรณ์พื้นฐาน (Primitives Component) ได้แก่ อุปกรณ์เกตพื้นฐานต่าง ๆ (Predefined Gate Primitives) ที่เครื่องมือสร้างเตรียมไว้แล้ว ประกอบด้วย อุปกรณ์เกตแอนด์ (AND), แนนด์ (NAND), ออร์ (OR), นอร์ (NOR), เอ็กซ์คลูซีฟออร์ (XOR), เอ็กซ์คลูซีฟนอร์ (XNOR), อินเวอร์เตอร์ (NOT), บัฟเฟอร์ (BUFFER) และดีเลย์ (DELAY) โดยแต่ละตัวนั้นเป็นตัวกระทำทางตรรกะพื้นฐาน ซึ่งเครื่องมือสร้างเตรียมไว้ให้แล้วผู้ออกแบบไม่จำเป็นต้องสร้างขึ้นเอง เป็นอุปกรณ์ที่ทำงานด้วยตารางค่าความจริงทางตรรกะ โดยที่ค่าในตารางค่าความจริงของอุปกรณ์เกตพื้นฐานเหล่านี้จะไม่สามารถแก้ไขได้ สามารถสรุปรายละเอียดของอุปกรณ์เกตพื้นฐานแต่ละตัวได้ดังนี้

รายละเอียดอุปกรณ์เกตพื้นฐานแต่ละตัว

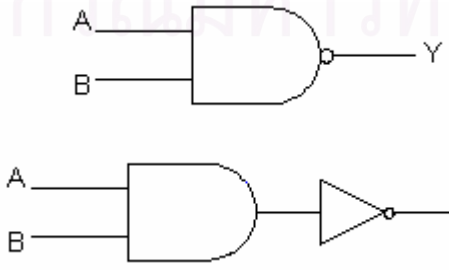
1. อุปกรณ์เกตแอนด์ (AND) เป็นอุปกรณ์เกตที่มีตัวแปรนำเข้าตั้งแต่ 2 ตัวแปรนำเข้าขึ้นไป มี 1 ตัวแปรผลลัพธ์ วงจรจะให้ผลลัพธ์เป็น 1 เมื่อตัวแปรนำเข้าทุกตัวมีค่าเป็น 1 กรณีอื่น ๆ ค่าผลลัพธ์จะมีค่าเป็น 0 สัญลักษณ์แสดงดังรูปที่ 3.1 สมการบูลีนคือ AB=Y และตารางค่าความจริงแสดงดังตารางที่ 3.2

รูปที่ 3.1 สัญลักษณ์แทนอุปกรณ์เกตแอนด์ (AND)

ตารางที่ 3.2 ตารางค่าความจริงตามสมการบูลีน AB=Y

| Input | | Output |
|---|---|---|
| A | B | Y=AB |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

2. อุปกรณ์เกตแนนด์ (NAND) เป็นอุปกรณ์เกตที่มีตัวแปรนำเข้าตั้งแต่ 2 ตัวแปรนำเข้าขึ้นไป มี 1 ตัวแปรผลลัพธ์ มีการทำงานตรงกันข้ามกับอุปกรณ์เกตแอนด์ คือ วงจรจะให้ผลลัพธ์เป็น 0 เมื่อตัวแปรนำเข้าทุกตัวมีค่าเป็น 1 กรณีอื่น ๆ ค่าผลลัพธ์จะมีค่าเป็น 1 สัญลักษณ์แสดงดังรูปที่ 3.2 สมการบูลีนคือ ~(AB)=Y หรือ ~A+~B=Y และตารางค่าความจริงแสดงดังตารางที่ 3.3

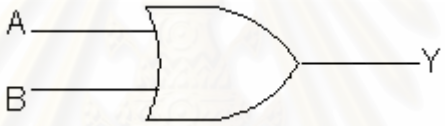



รูปที่ 3.2 สัญลักษณ์แทนอุปกรณ์เกตแนนด์ (NAND)

ตารางที่ 3.3 ตารางค่าความจริงตามสมการบูลีน ~(AB)=Y หรือ ~A+~B=Y

| Input | | Output |
|---|---|---|
| A | B | Y=~(AB) **หรือ** Y= ~A+~B |
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

3. อุปกรณ์เกตออร์ (OR) เป็นอุปกรณ์เกตที่มีตัวแปรนำเข้าตั้งแต่ 2 ตัวแปรนำเข้าขึ้นไป มี 1 ตัวแปรผลลัพธ์ วงจรจะให้ผลลัพธ์เป็น 1 เมื่อตัวแปรนำเข้าตัวใดตัวหนึ่งมีค่าเป็น 1 และเมื่อตัวแปรนำเข้าทุกตัวมีค่าเป็น 0 ค่าผลลัพธ์จะมีค่าเป็น 0 สัญลักษณ์แสดงดังรูปที่ 3.3 สมการบูลีนคือ A+B=Y และตารางค่าความจริงแสดงดังตารางที่ 3.4

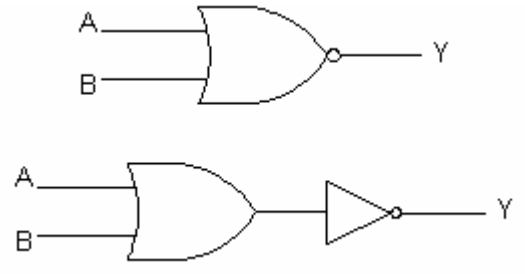



รูปที่ 3.3 สัญลักษณ์แทนอุปกรณ์เกตออร์ (OR)

ตารางที่ 3.4 ตารางค่าความจริงตามสมการบูลีน A+B=Y

| Input | | Output |
|---|---|---|
| A | B | Y=A+B |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

4. อุปกรณ์เกตนอร์ (NOR) เป็นอุปกรณ์เกตที่มีตัวแปรนำเข้าตั้งแต่ 2 ตัวแปรนำเข้าขึ้นไป มี 1 ตัวแปรผลลัพธ์ มีการทำงานตรงกันข้ามกับอุปกรณ์เกตออร์ คือ วงจรจะให้ผลลัพธ์เป็น 0 เมื่อตัวแปรนำเข้าตัวใดตัวหนึ่งมีค่าเป็น 1 และเมื่อตัวแปรนำเข้าทุกตัวมีค่าเป็น 0 ค่าผลลัพธ์จะมีค่าเป็น 1 สัญลักษณ์แสดงดังรูปที่ 3.4 สมการบูลีนคือ ~(A+B)=Y หรือ ~A~B=Y และตารางค่าความจริงแสดงดังตารางที่ 3.5




รูปที่ 3.4 สัญลักษณ์แทนอุปกรณ์เกตนอร์ (NOR)

ตารางที่ 3.5 ตารางค่าความจริงตามสมการบูลีน ~(A+B)=Y หรือ ~A~B=Y

| Input | | Output |
|---|---|---|
| A | B | Y=~(A+B) **หรือ** Y=~A~B |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

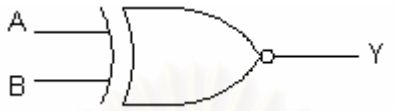
5. อุปกรณ์เกตเอ็กซ์คลูซีฟออร์ (XOR) เป็นอุปกรณ์เกตที่มีตัวแปรนำเข้าตั้งแต่ 2 ตัวแปรขึ้นไป มี 1 ตัวแปรผลลัพธ์ วงจรจะให้ผลลัพธ์เป็น 1 เมื่อตัวแปรนำเข้ามีค่าต่างกันและให้ผลลัพธ์เป็น 0 เมื่อตัวแปรนำเข้ามีค่าเหมือนกัน สัญลักษณ์แสดงดังรูปที่ 3.5 สมการบูลีนคือ A~B+~AB=Y และตารางค่าความจริงแสดงดังตารางที่ 3.6

รูปที่ 3.5 สัญลักษณ์แทนอุปกรณ์เกตเอ็กซ์คลูซีฟออร์ (XOR)

ตารางที่ 3.6 ตารางค่าความจริงตามสมการบูลีน A~B+~AB=Y

| Input | | Output |
|---|---|---|
| A | B | Y=A~B+~AB |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

6. อุปกรณ์เกตเอ็กซ์คลูซีฟนอร์ (XNOR) เป็นอุปกรณ์เกตที่มีตัวแปรนำเข้าตั้งแต่ 2 ตัวแปรนำเข้าขึ้นไป มี 1 ตัวแปรผลลัพธ์ วงจรจะให้ผลลัพธ์เป็น 1 เมื่อตัวแปรนำเข้ามีค่าเหมือนกัน และให้ผลลัพธ์เป็น 0 เมื่อตัวแปรนำเข้ามีค่าต่างกัน สัญลักษณ์แสดงดังรูปที่ 3.6 สมการบูลีนคือ AB+~A~B=Y และตารางค่าความจริงแสดงดังตารางที่ 3.7

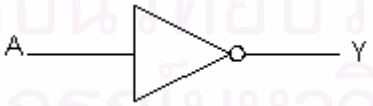



รูปที่ 3.6 สัญลักษณ์แทนอุปกรณ์เกตเอ็กซ์คลูซีฟนอร์ (XNOR)

ตารางที่ 3.7 ตารางค่าความจริงตามสมการบูลีน AB+~A~B=Y

| Input | | Output |
|---|---|---|
| A | B | Y=AB+~A~B |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

7. อุปกรณ์เกตอินเวอร์เตอร์ (NOT) เป็นอุปกรณ์เกตที่มี 1 ตัวแปรนำเข้าและ 1 ตัวแปรผลลัพธ์ วงจรจะให้ค่าผลลัพธ์ที่ตรงกันข้ามกับค่าของตัวแปรนำเข้า เช่น ถ้าตัวแปรนำเข้ามีค่าเท่ากับ 0 ผลลัพธ์ที่ได้จะมีค่าเท่ากับ 1 และถ้าตัวแปรนำเข้ามีค่าเท่ากับ 1 ผลลัพธ์ที่ได้จะมีค่าเท่ากับ 0 สัญลักษณ์แสดงดังรูปที่ 3.7 สมการบูลีนคือ ~A=Y และตารางค่าความจริงแสดงดังตารางที่ 3.8

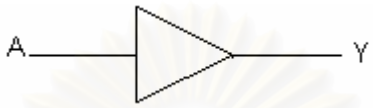



รูปที่ 3.7 สัญลักษณ์แทนอุปกรณ์เกตอินเวอร์เตอร์ (NOT)

ตารางที่ 3.8 ตารางค่าความจริงตามสมการบูลีน ~A=Y

| Input | Output |
|---|---|
| A | Y=~A |
| 0 | 1 |
| 1 | 0 |

8. อุปกรณ์เกตบัฟเฟอร์ (BUFFER) เป็นอุปกรณ์เกตที่มี 1 ตัวแปรนำเข้าและ 1 ตัวแปรผลลัพธ์ วงจรจะให้ค่าผลลัพธ์ที่ตรงกันกับค่าของตัวแปรนำเข้าเสมอ เช่น ตัวแปรนำเข้ามีค่าเท่ากับ 0 ผลลัพธ์ที่ได้ก็จะมีค่าเท่ากับ 0 และถ้าตัวแปรนำเข้ามีค่าเท่ากับ 1 ผลลัพธ์ที่ได้ก็จะมีค่าเท่ากับ 1 เช่นกัน สัญลักษณ์แสดงดังรูปที่ 3.8 สมการบูลีนคือ A=Y และตารางค่าความจริงแสดงดังตารางที่ 3.9

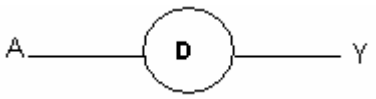



รูปที่ 3.8 สัญลักษณ์แทนอุปกรณ์เกตบัฟเฟอร์ (BUFFER)

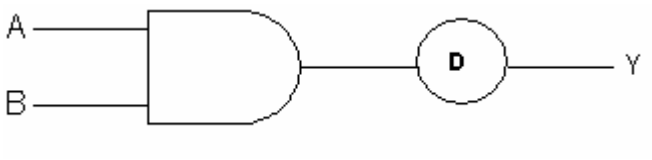
ตารางที่ 3.9 ตารางค่าความจริงตามสมการบูลีน A=Y

| Input | Output |
|---|---|
| A | Y=A |
| 0 | 0 |
| 1 | 1 |

9. อุปกรณ์เกตดีเลย์ (DELAY) เป็นอุปกรณ์เกตที่มี 1 ตัวแปรนำเข้าและ 1 ตัวแปรผลลัพธ์เช่นเดียวกันกับอุปกรณ์เกตอินเวอร์เตอร์และอุปกรณ์เกตบัฟเฟอร์ วงจรจะให้ค่าผลลัพธ์ที่ตรงกันกับค่าของตัวแปรนำเข้าเสมอแต่อุปกรณ์เกตดีเลย์นี้จะมีความพิเศษอยู่อย่างหนึ่งคือ มันจะมีค่าความหน่วงอยู่ในตัวอุปกรณ์เองซึ่งมีค่าเท่ากับ 1 คล็อกเสมอ ดังนั้นเมื่อมีการนำอุปกรณ์เกตดีเลย์ประกอบในการออกแบบสร้างวงจรหากต้องการทดสอบการทำงานของวงจรที่ออกแบบ การแสดงค่าผลลัพธ์ที่ถูกผลิตออกจากตัวอุปกรณ์เกตดีเลย์นั้นจะมีการเก็บค่าค้างไว้ 1 คล็อกเสมอ เมื่อค่าจากตัวแปรนำเข้าถูกส่งเข้ามาใหม่ในรอบถัดไปค่าที่ค้างอยู่ในตัวอุปกรณ์เกตดีเลย์จึงจะถูกส่งไปแสดงผลที่ตัวแปรผลลัพธ์และจะเป็นเช่นนี้ไปทุก ๆ รอบของการอ่านค่าตัวแปรนำเข้าเข้ามาในวงจร เครื่องมือจะสิ้นสุดการจำลองการทำงานก็ต่อเมื่อไม่มีค่าใด ๆ ค้างอยู่ในตัวอุปกรณ์เกตดีเลย์ สัญลักษณ์แสดงดังรูปที่ 3.9 อุปกรณ์เกตดีเลย์นี้ไม่สามารถกำหนดได้ด้วยสมการบูลีน




รูปที่ 3.9 สัญลักษณ์แทนอุปกรณ์เกตดีเลย์ (DELAY)




รูปที่ 3.10 แสดงการประกอบอุปกรณ์เกตดีเลย์เข้ากับอุปกรณ์เกตแอนด์

ตารางที่ 3.10 ตัวอย่างการป้อนค่าตัวเลขให้กับตัวแปรนำเข้า A และ B

| Input Name | Decimal Value | Binary Value |
|---|---|---|
| A | 15 | 0000000000001111 |
| B | 9 | 0000000000001001 |

ตารางที่ 3.11 ผลลัพธ์การทำงานที่ได้จากการประกอบอุปกรณ์เกตดีเลย์เข้ากับอุปกรณ์เกตแอนด์

| รอบที่ | ค่าตัวแปรนำเข้า A, Bที่ถูกอ่านเข้ามาทำงานในแต่ละรอบ | | ค่าที่อุปกรณ์เกตดีเลย์เก็บค้างไว้ในแต่ละรอบ | ค่าตัวแปรผลลัพธ์ Y |
|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 (ไม่มี) |
| 2 | 0 | 0 | 0 | 0 |
| 3 | 0 | 0 | 0 | 0 |
| 4 | 0 | 0 | 0 | 0 |
| 5 | 0 | 0 | 0 | 0 |
| 6 | 0 | 0 | 0 | 1 |
| 7 | 0 | 0 | 0 | 0 |
| 8 | 0 | 0 | 0 | 0 |
| 9 | 0 | 0 | 0 | 1 |
| 10 | 0 | 0 | 0 | 0 |
| 11 | 0 | 0 | 0 | 0 |
| 12 | 0 | 0 | 0 | 0 |
| 13 | 1 | 1 | 0 | 0 |
| 14 | 1 | 0 | 0 | 1 |
| 15 | 1 | 0 | 1 | 0 |
| 16 | 1 | 1 | 0 | 0 |
| 17 | 0 (ไม่มี) | 0 (ไม่มี) | 0 (ไม่มี) | 1 |

สามารถอธิบายตัวอย่างขั้นตอนการทำงานของอุปกรณ์เกตดีเลย์ได้ดังรูปที่ 3.10 และจากตารางที่ 3.10 และตารางที่ 3.11

จากตารางที่ 3.11 ข้างต้นแสดงการทำงานที่ถูกต้องของรูปที่ 3.10 เมื่อตัวแปรนำเข้า A มีค่าเท่ากับ 15 และตัวแปรนำเข้า B มีค่าเท่ากับ 9 โดยจะให้ค่าผลลัพธ์ Y เท่ากับ 9 และจะได้จำนวนรอบทั้งหมดที่ใช้ในการจำลองการทำงานเท่ากับ 17 รอบหรือ 17 คล็อก สำหรับค่าตัวเลขที่วงเล็บไว้ว่า “ไมมี” นั้น หมายความว่า จริง ๆ แล้วไม่มีค่าสัญญาณเข้าแต่ให้มองว่าค่าที่เข้ามาคือ 0 นั่นเอง

สำหรับอุปกรณ์เกตดีเลย์นั้น เป็นอุปกรณ์เกตพื้นฐานเพียงตัวเดียวที่มีค่าความหน่วงอยู่ในตัวอุปกรณ์เองซึ่งมีค่าเท่ากับ 1 คล็อก โดยที่ค่าความหน่วงนี้จะมีผลต่อการแสดงค่าผลลัพธ์ของอุปกรณ์หรือวงจรขณะทำการจำลองการทำงานเสมอ เช่น ถ้าอุปกรณ์หรือวงจรที่สร้างขึ้นมีอุปกรณ์เกตดีเลย์ประกอบอยู่ด้วยกันจำนวน 3 ตัว แสดงว่าอุปกรณ์หรือวงจรนั้นมีค่าความหน่วงเท่ากับ 3 คล็อก นั่นหมายความว่า ค่าผลลัพธ์ที่จะแสดงค่าออกมาที่ตัวแปรผลลัพธ์นั้นจะถูกผลิตออกมาจากอุปกรณ์หรือวงจรนั้นค่าแรกก็ต่อเมื่อมีการทำงานผ่านไปแล้ว 3 คล็อก หรือเรียกว่าเกิดขึ้นในคล็อกที่ 4 นั่นเอง

นอกจากนี้ในส่วนคลังอุปกรณ์เครื่องมือจะเตรียมส่วนของ Pin Count ซึ่งเลือกค่าได้ตั้งแต่ 2 – 5 จำนวน สำหรับให้ผู้ออกแบบเลือกกำหนดจำนวนตัวแปรนำเข้าให้กับอุปกรณ์เกตพื้นฐานจำพวกอุปกรณ์เกต AND, NAND, OR, NOR, XOR และ XNOR ไว้อีกด้วย

2) อุปกรณ์ประกอบ ได้แก่ วงจรย่อยต่าง ๆ (Sub-circuits) ที่เกิดจากการนำอุปกรณ์พื้นฐานและ/หรืออุปกรณ์ประกอบที่มีอยู่มาประกอบเข้าด้วยกันแล้วทำการบันทึกเก็บไว้อุปกรณ์ UDP และอุปกรณ์ UDE ซึ่งเป็นอุปกรณ์ที่ผู้ออกแบบสร้างขึ้นใหม่ โดยอุปกรณ์หรือวงจรที่ผู้ออกแบบได้ทำการสร้างขึ้นแล้วทำการบันทึกเก็บไว้นั้นจะถูกเก็บลงคลังอุปกรณ์ในส่วนของอุปกรณ์ประกอบนั่นเอง การลากอุปกรณ์ประกอบไปใช้งานในวงจรที่ออกแบบใหม่นั้นจะแทนอุปกรณ์ประกอบได้ด้วยรูปสี่เหลี่ยมที่แสดงชื่ออุปกรณ์ จำนวนขาและชื่อของตัวแปรนำเข้าและตัวแปรผลลัพธ์เท่านั้น หากผู้ออกแบบต้องการทราบว่าอุปกรณ์ประกอบตัวนั้นมีโครงสร้างภายในประกอบด้วยอุปกรณ์ใดบ้างก็สามารถทำได้ด้วยการคลิกเมาส์ขวาที่ตัวอุปกรณ์ตัวนั้นแล้วเลือกเมนูวิวอินเทอร์แนล (View Internal) เพื่อให้เครื่องมือเปิดหน้าต่างย่อยที่แสดงโครงสร้างภายในของอุปกรณ์ตัวนั้นขึ้นมาแต่จะแก้ไขโครงสร้างภายในของอุปกรณ์ตัวนั้นไม่ได้

<u>รายละเอียดของอุปกรณ์ UDP และอุปกรณ์ UDE</u>

1. อุปกรณ์พื้นฐานที่กำหนดโดยผู้ออกแบบเอง (UDP: User-defined Primitives) หรือเรียกว่า อุปกรณ์ UDP ถึงแม้ว่าจะมีอุปกรณ์เกตพื้นฐานที่ถูกสร้างไว้แล้ว และสามารถใช้งานได้ทันที แต่อาจไม่เพียงพอกับการออกแบบที่ต้องการอุปกรณ์หลาย ๆ ประเภท นอกเหนือจากอุปกรณ์เกตพื้นฐานต่าง ๆ ดังนั้นเครื่องมือที่พัฒนาขึ้นนี้จึงอนุญาตให้ผู้ออกแบบสามารถกำหนดอุปกรณ์พื้นฐานของตัวเองได้ โดยสามารถกำหนดฟังก์ชั่นการทำงานของอุปกรณ์ด้วยค่าความหน่วงและตารางค่าความจริงที่ผู้ออกแบบเป็นผู้กำหนดค่าผลลัพธ์ได้เองตามต้องการ

สำหรับการใช้งานอุปกรณ์ UDP นั้น สามารถเรียกใช้งานได้เหมือนกันกับวงจรย่อยต่าง ๆ เนื่องจากอุปกรณ์ UDP ที่สร้างขึ้นจะถูกบันทึกเก็บลงในส่วนของอุปกรณ์ประกอบซึ่งเป็นที่เก็บวงจรย่อยต่าง ๆ นั่นเอง แต่อุปกรณ์ UDP มีข้อจำกัดคือ สามารถกำหนดจำนวนตัวแปรนำเข้าได้สูงสุด 5 จำนวน และจำนวนตัวแปรผลลัพธ์ได้เพียง 1 จำนวนเท่านั้น ไม่สามารถมีตัวแปรนำเข้าและตัวแปรผลลัพธ์เป็นแบบเวกเตอร์ได้ และต้องกำหนดการทำงานด้วยค่าความหน่วงและตารางค่าความจริงเสมอ โดยโครงสร้างของอุปกรณ์ UDP นั้นจะประกอบไปด้วยจำนวนตัวแปรนำเข้าที่กำหนดค่าได้ตั้งแต่ 1 – 5 จำนวน และจำนวนตัวแปรผลลัพธ์ 1 จำนวน จำนวนค่าความหน่วง (Delay) กำหนดค่าได้ตั้งแต่ 0 – 10 คล็อก และส่วนของตารางค่าความจริงสำหรับกำหนดฟังก์ชั่นการทำงานของตัวอุปกรณ์ UDP นั้น ๆ โดยเครื่องมือจะแสดงกรณีทั้งหมดที่เป็นไปได้ของตัวแปรนำเข้า ซึ่งการสร้างตารางค่าความจริงนี้จะพิจารณาจากจำนวนตัวแปรนำเข้าที่ผู้ออกแบบได้กำหนดไว้ โดยจะพิจารณาสร้างตารางในรูปของ $2^n$ กรณีเสมอ โดยที่ n คือจำนวนตัวแปรนำเข้าที่กำหนดนั่นเอง ยกตัวอย่างเช่น ผู้ออกแบบกำหนดให้จำนวนตัวแปรนำเข้ามีค่าเท่ากับ 3 จำนวน เครื่องมือจะสร้างตารางขึ้นมามีจำนวนแถวข้อมูลเท่ากับ $2^3 = 8$ แถวข้อมูล หรือ 8 กรณี เป็นต้น ส่วนค่าตัวแปรผลลัพธ์นั้นจะให้ผู้ออกแบบเป็นผู้กำหนดค่าได้เอง ซึ่งอุปกรณ์ UDP เป็นการกำหนดการใช้งานของอุปกรณ์พื้นฐานเท่านั้น จึงมีข้อจำกัดต่าง ๆ เช่น ผู้ออกแบบไม่สามารถสร้างตัวอุปกรณ์ที่มีหลายตัวแปรผลลัพธ์หรือที่มีตัวแปรนำเข้าและตัวแปรผลลัพธ์เป็นแบบเวกเตอร์ได้

นอกจากนี้การใช้งานอุปกรณ์ UDP ยังสามารถเรียกใช้งานได้เหมือนกับอุปกรณ์เกตพื้นฐานต่าง ๆ ที่เครื่องมือสร้างไว้แล้วในส่วนคลังอุปกรณ์และเนื่องจากอุปกรณ์ UDP อยู่ในระดับเดียวกับวงจรย่อยต่าง ๆ จึงไม่สามารถสร้างอุปกรณ์ UDP ได้ในตัววงจรย่อย ดังนั้นผู้ออกแบบจะต้องออกแบบอุปกรณ์ UDP ที่ต้องการใช้ไว้ก่อนตัววงจรย่อยที่ต้องการเรียกอุปกรณ์ UDP มาใช้งาน เมื่อเรียกอุปกรณ์ UDP มาใช้งาน เครื่องมือจะวาดรูปแทนอุปกรณ์ UDP ด้วยรูปสี่เหลี่ยมที่

แสดงเฉพาะจำนวนตัวแปรนำเข้าและตัวแปรผลลัพธ์เท่านั้น อุปกรณ์ UDP เป็นอุปกรณ์พื้นฐานประเภทเดียวในเครื่องมือที่สามารถแก้ไขค่าในตารางค่าความจริงได้แต่ต้องทำการบันทึกอุปกรณ์ UDP นั้นด้วยชื่อใหม่

2. อุปกรณ์ที่กำหนดด้วยสมการทางคณิตศาสตร์ (UDE: User-defined Expression) หรือเรียกว่า อุปกรณ์ UDE โดยอุปกรณ์ประเภทนี้สร้างขึ้นด้วยการกำหนดสมการทางคณิตศาสตร์ที่ประกอบด้วยตัวทำดำเนินการ (Operator) ได้แก่ เครื่องหมายการบวก (+) เครื่องหมายการลบ (-) เครื่องหมายการคูณ (*) เครื่องหมายการหาร (/) เครื่องหมายการยกกำลัง (^) และเครื่องหมายเท่ากับ (=) โดยที่สามารถมีตัวถูกดำเนินการ (Operand) ได้แก่ ตัวแปรนำเข้าและตัวแปรผลลัพธ์ซึ่งอยู่ในแต่ละข้างของเครื่องหมายเท่ากับได้สูงสุดไม่เกิน 5 จำนวน

สำหรับการใช้งานอุปกรณ์ UDE นั้น สามารถเรียกใช้งานได้เหมือนกันกับวงจรย่อยต่าง ๆ และอุปกรณ์ UDP เนื่องจากอุปกรณ์ UDE ที่สร้างขึ้นจะถูกบันทึกเก็บลงในส่วนของอุปกรณ์ประกอบซึ่งเป็นที่เก็บวงจรย่อยต่าง ๆ และอุปกรณ์ UDP นั่นเอง โดยโครงสร้างของอุปกรณ์ UDE นั้นจะประกอบไปด้วยจำนวนตัวแปรนำเข้าและจำนวนตัวแปรผลลัพธ์ที่กำหนดค่าผ่านทางสมการทางคณิตศาสตร์ ซึ่งมีค่าได้สูงสุดไม่เกินอย่างละ 5 จำนวน และกำหนดฟังก์ชั่นการทำงานสำหรับอุปกรณ์ UDE นี้ ด้วยตารางค่าความจริง โดยเครื่องมือจะทำการสร้างตารางค่าความจริงให้อัตโนมัติตามสมการทางคณิตศาสตร์ที่ผู้ออกแบบกำหนดและไม่สามารถแก้ไขค่าในตารางค่าความจริงที่สร้างขึ้นอัตโนมัตินี้ได้

สำหรับการสร้างตารางค่าความจริงให้กับอุปกรณ์ UDE โดยอัตโนมัตินั้น เครื่องมือจะทำการสร้างตารางค่าความจริงด้วยการเริ่มต้นจาก การพิจารณาจำนวนตัวแปรนำเข้าและจำนวนตัวแปรผลลัพธ์ทั้งหมดจากสมการทางคณิตศาสตร์ที่ผู้ออกแบบกำหนด เพื่อทำการสร้างกรณีที่เป็นไปได้ทั้งหมดของตัวแปรนำเข้าและกรณีที่เป็นไปได้ทั้งหมดของตัวแปรผลลัพธ์ในการที่จะนำค่าจากกรณีต่าง ๆ ไปแทนค่าในสมการที่กำหนดเพื่อหากรณีที่ทำให้สมการนั้นเป็นจริง คือ ค่าที่คำนวณได้จากทั้งสองข้างของเครื่องหมายเท่ากับมีค่าเท่ากันนั่นเอง

หากกรณีใด ๆ ก็ตามที่นำไปแทนค่าในสมการแล้วทำให้ค่าที่คำนวณได้จากทั้งสองข้างของเครื่องหมายเท่ากับมีค่าเท่ากัน เครื่องมือจะนำกรณีนั้น ๆ ของตัวแปรนำเข้าและของตัวแปรผลลัพธ์มาบันทึกเก็บลงตารางค่าความจริง ถ้าหากมีกรณีใด ๆ ก็ตามของตัวแปรนำเข้า 1 กรณี ที่เมื่อนำกรณีที่เป็นไปได้ของตัวแปรผลลัพธ์มาแทนในสมการมากกว่า 1 กรณี แล้วให้ค่าที่คำนวณได้จากทั้งสองข้างของเครื่องหมายเท่ากับมีค่าเท่ากันเหมือนกัน เครื่องมือจะพิจารณาจาก

กรณีแรกที่ทำให้ได้ค่าที่คำนวณได้เท่ากันเท่านั้นมาบันทึกเก็บลงตารางค่าความจริง และหากไม่มีกรณีใด ๆ ของตัวแปรผลลัพธ์ที่เมื่อนำไปแทนค่าในสมการแล้วให้ค่าที่คำนวณได้จากทั้งสองข้างของเครื่องหมายเท่ากับมีค่าเท่ากัน เครื่องมือจะไม่อนุญาตให้สร้างอุปกรณ์ UDE จากสมการทางคณิตศาสตร์ที่กำหนด

สำหรับการออกแบบแบบลำดับชั้น (Hierarchical design) ไม่จำเป็นต้องใช้แค่อุปกรณ์พื้นฐานทั่วไปเท่านั้น ผู้ออกแบบยังสามารถเรียกใช้วงจรย่อยที่ออกแบบไว้แล้วมาประกอบกันเป็นวงจรที่มีความซับซ้อนได้ การเรียกใช้งานวงจรย่อยในเครื่องมือนี้มีวิธีการเหมือนกับการเรียกใช้อุปกรณ์พื้นฐานต่าง ๆ โดยไม่จำเป็นต้องใช้การเรียนรู้เพิ่มเติม นอกจากนี้การใช้งานวงจรย่อยในการออกแบบจะมีความยืดหยุ่นในการออกแบบมากกว่าอุปกรณ์พื้นฐานต่าง ๆ เนื่องจากสามารถมีหลายจำนวนตัวแปรนำเข้า หลายจำนวนตัวแปรผลลัพธ์ เป็นต้น อีกอย่างคือ วงจรย่อยสามารถเรียกใช้อุปกรณ์พื้นฐานได้ แต่ในทางกลับกันอุปกรณ์พื้นฐานไม่สามารถเรียกใช้วงจรย่อยได้ ดังนั้นการออกแบบโดยใช้วงจรย่อยที่ประกอบด้วยวงจรย่อยจึงสามารถทำการออกแบบเป็นลำดับชั้นได้โดยง่ายและสะดวกกว่า แต่ข้อจำกัดที่สำคัญอย่างหนึ่งในการใช้งานวงจรย่อยคือผู้ออกแบบไม่สามารถออกแบบวงจรย่อยภายในวงจรย่อยใด ๆ ได้ ดังนั้นผู้ออกแบบจะต้องออกแบบแยกไว้ก่อนตัววงจรที่จะเรียกมาใช้งาน

3) อุปกรณ์ตัวแปรนำเข้าและอุปกรณ์ตัวแปรผลลัพธ์ คือ อุปกรณ์อีกประเภทหนึ่งที่เครื่องมือสร้างเตรียมไว้ให้แล้ว สำหรับให้ผู้ออกแบบได้เลือกนำไปประกอบเข้ากับวงจรที่สร้างขึ้นเพื่อให้วงจรที่สร้างขึ้นนั้นกลายเป็นวงจรย่อยและบันทึกเก็บลงในส่วนของอุปกรณ์ประกอบนั่นเอง หรืออีกวัตถุประสงค์สำหรับให้ผู้ออกแบบได้เลือกนำไปประกอบเข้ากับอุปกรณ์พื้นฐานหรืออุปกรณ์ประกอบหรือวงจรอื่น ๆ ซึ่งผู้ออกแบบเรียกขึ้นมาแสดงบนส่วนวาดวงจรเพื่อการทดสอบจำลองการทำงานของอุปกรณ์หรือวงจรเหล่านั้น ซึ่งสามารถทดสอบการทำงานได้ด้วยการป้อนค่าตัวเลขให้กับตัวแปรนำเข้าชนิดต่าง ๆ ที่ประกอบอยู่ในอุปกรณ์หรือวงจรที่สร้างขึ้นเพื่อดูค่าผลลัพธ์ผ่านทางตัวแปรผลลัพธ์ การป้อนค่าตัวเลขให้กับตัวแปรนำเข้าชนิดต่าง ๆ นั้นจะทำในส่วนกำหนดค่าตัวแปรนำเข้า ซึ่งเครื่องมือสร้างเตรียมไว้ในส่วนการจำลองการทำงาน ผู้ออกแบบสามารถเลือกทำการป้อนค่าตัวเลขที่ช่องใดช่องหนึ่งก็ได้ระหว่างช่องป้อนค่าตัวเลขฐานสิบและช่องป้อนค่าตัวเลขฐานสอง โดยเครื่องมือจะทำการแปลงค่าตัวเลขที่ถูกต้องของอีกช่องให้ควบคู่กันไปด้วย สำหรับจำนวนบิตสูงสุดที่ผู้วิจัยทำการกำหนดไว้สำหรับการป้อนค่าตัวเลขลงในช่องป้อนค่าตัวเลขฐานสองคือ 16 บิต ทำให้ค่าตัวเลขที่สามารถป้อนได้ในช่องป้อนค่าตัวเลขฐานสิบต้องมี

ค่าไม่เกินตัวเลข 65535 เท่านั้น โดยในส่วนของอุปกรณ์ตัวแปรนำเข้านั้นผู้วิจัยได้ทำการออกแบบอุปกรณ์ตัวแปรนำเข้าเตรียมไว้ 3 ชนิด ซึ่งแบ่งตามลักษณะการทำงานที่แตกต่างกัน ได้ดังนี้

ลักษณะของอุปกรณ์ตัวแปรนำเข้าชนิดต่าง ๆ

1. อุปกรณ์ตัวแปรนำเข้าชนิดไม่ใช่ระบบจำนวนซ้ำซ้อน (Non-Redundant Number System) หรือเรียกว่า อุปกรณ์ตัวแปรนำเข้าชนิด NR ซึ่งเป็นอุปกรณ์ตัวแปรนำเข้าชนิดที่ผู้ออกแบบไม่สามารถป้อนค่าตัวเลขด้วยค่าติดลบให้กับตัวแปรนำเข้าได้ไม่ว่าจะเป็นค่าตัวเลขฐานสิบหรือค่าตัวเลขฐานสองก็ตาม ตัวอย่างแสดงดังตารางที่ 3.12

ตารางที่ 3.12 ตัวอย่างการป้อนค่าตัวเลขให้กับตัวแปรนำเข้าชนิด NR

| Input Name | Decimal Value | Binary Value |
|---|---|---|
| A | 1786 | 0000011011111010 |
| B | 354 | 0000000101100010 |

จากตารางข้างต้น กำหนดให้ตัวแปรนำเข้า A และ B เป็นตัวแปรนำเข้าชนิด NR ผู้ออกแบบไม่สามารถป้อนค่าตัวเลขให้กับตัวแปรนำเข้า A และ B ด้วยค่าตัวเลขติดลบได้ทั้งในช่องป้อนค่าตัวเลขฐานสิบและช่องป้อนค่าตัวเลขฐานสอง โดยผู้วิจัยได้ทำการออกแบบให้สามารถเลือกป้อนค่าตัวเลขในช่องใดช่องหนึ่งก็ได้แล้วให้เครื่องมือทำการแปลงค่าที่ถูกต้องของอีกช่องควบคู่กันไปด้วย ผู้ออกแบบไม่สามารถป้อนค่าตัวเลข -1786 ในช่องป้อนค่าตัวเลขฐานสิบได้ และในทำนองเดียวกันในช่องป้อนค่าตัวเลขฐานสองนั้นก็สามารถป้อนค่าตัวเลขได้ด้วยตัวเลข 0 กับ 1 เท่านั้น เนื่องจากระบบตัวเลขฐานสองเป็นระบบตัวเลขที่มีเลขที่ใช้เขียนจำนวน 2 ตัว คือ เลข 0 และเลข 1 สำหรับระบบตัวเลขฐานสิบและตัวเลขฐานสองสามารถอธิบายได้ดังนี้

ระบบตัวเลขฐานสิบและตัวเลขฐานสอง

1. ระบบเลขฐานสิบ คือ

$$10^n \quad 10^{n-1} \quad \ldots \quad 10^2 \quad 10^1 \quad 10^0$$

ตัวอย่างเช่น $1786_{10} = (1x10^3) + (7x10^2) + (8x10^1) + (6x10^0)$

$= 1000 + 700 + 80 + 6$

$= 1786$

2. ระบบเลขฐานสอง คือ

$2^n$ $2^{n-1}$ ... $2^2$ $2^1$ $2^0$

ตัวอย่างเช่น $10110_2 = (1x2^4) + (0x2^3) + (1x2^2) + (1x2^1) + (0x2^0)$

$= 16 + 0 + 4 + 2 + 0$

$= 22$

ตัวอย่างการแปลงค่า $1000111_2$ ไปเป็นเลขฐานสิบ แสดงได้ดังนี้

| | | | | | | |
|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| x | x | x | x | x | x | x |
| $2^6$ | $2^5$ | $2^4$ | $2^3$ | $2^2$ | $2^1$ | $2^0$ |
| 64 | 32 | 16 | 8 | 4 | 2 | 1 |

(1x64) + (0x32) + (0x16) + (0x8) + (1x4) + (1x2) + (1x1)

= 64 + 0 + 0 + 0 + 4 + 2 + 1= 71

ตัวอย่างการแปลงค่า $45_{10}$ ไปเป็นเลขฐานสอง แสดงได้ดังนี้

2)45

2)22 เหลือเศษ 1

2)11 เหลือเศษ 0

2) 5 เหลือเศษ 1

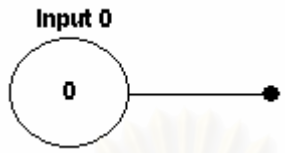
2) 2 เหลือเศษ 1

1 เหลือเศษ 0

$45_{10} = 101101_2$

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สำหรับการเรียกใช้งานอุปกรณ์ตัวแปรนำเข้าชนิด NR นี้ เมื่อผู้ออกแบบทำการลากอุปกรณ์ตัวแปรนำเข้าชนิดนี้ ไปวางลงบนส่วนวาดวงจร เครื่องมือจะวาดรูปแทนอุปกรณ์ตัวแปรนำเข้าชนิดนี้ขึ้นมาหนึ่งรูปตั้งชื่อว่า Input 0 สัญลักษณ์แสดงดังรูปที่ 3.11




รูปที่ 3.11 สัญลักษณ์แทนอุปกรณ์ตัวแปรนำเข้าชนิด NR

ตารางที่ 3.13 แสดงการเพิ่มจำนวนแถวในส่วนกำหนดค่าตัวแปรนำเข้าเมื่อเลือกอุปกรณ์ตัวแปรนำเข้าชนิด NR

| Input Name | Decimal Value | Binary Value |
|---|---|---|
| Input 0 | | |
| Input 1 | | |

ผู้ออกแบบสามารถเปลี่ยนชื่อตัวแปรนำเข้าได้เองตามต้องการ ในขณะเดียวกันกับที่เครื่องมือวาดรูปแทนอุปกรณ์ตัวแปรนำเข้าชนิด NR นี้ บนส่วนวาดวงจร เครื่องมือก็จะทำการเพิ่มจำนวนแถวในส่วนกำหนดค่าตัวแปรนำเข้าจำนวนหนึ่งแถวด้วยเช่นกัน และหากมีการลากอุปกรณ์ตัวแปรนำเข้าชนิด NR นี้ มาวางบนส่วนวาดวงจรเพิ่มขึ้นอีกหนึ่งตัวเครื่องมือก็จะตั้งชื่อว่า Input 1 และทำการเพิ่มจำนวนแถวในส่วนกำหนดค่าตัวแปรนำเข้าเพิ่มขึ้นอีกหนึ่งแถว แสดงได้ดังตารางที่ 3.13 ซึ่งส่วนกำหนดค่าตัวแปรนำเข้าถูกสร้างไว้ในส่วนการจำลองการทำงานของเครื่องมือนั่นเอง

ในการทดสอบจำลองการทำงานผู้ออกแบบสามารถป้อนค่าตัวเลขให้กับตัวแปรนำเข้าชนิด NR นี้ ได้ทั้งช่องป้อนค่าตัวเลขฐานสิบหรือช่องป้อนค่าตัวเลขฐานสอง โดยเครื่องมือจะอ่านค่าจากช่องป้อนค่าตัวเลขฐานสองเท่านั้นไปทำงานในวงจรทีละบิตในแต่ละรอบจากตำแหน่งซ้ายสุดไปยังตำแหน่งขวาสุดตามลำดับจนครบทุกบิต เพื่อแสดงค่าผลลัพธ์ที่ถูกต้อง และผลลัพธ์ก็จะถูกแสดงเป็นลำดับทีละบิตในแต่ละรอบด้วยเช่นกัน

2. อุปกรณ์ตัวแปรนำเข้าชนิดระบบจำนวนซ้ำซ้อน (Redundant Number System) หรือเรียกว่า อุปกรณ์ตัวแปรนำเข้าชนิด R ซึ่งเป็นอุปกรณ์ตัวแปรนำเข้าชนิดที่ผู้ออกแบบสามารถป้อนค่าตัวเลขได้ทั้งค่าบวกและค่าลบให้แก่ตัวแปรนำเข้าชนิดนี้ได้ ทั้งค่าตัวเลขฐานสิบหรือค่าตัวเลขฐานสอง ตัวอย่างแสดงดังตารางที่ 3.14

ตารางที่ 3.14 ตัวอย่างการป้อนค่าตัวเลขให้กับตัวแปรนำเข้าชนิด R

| Input Name | Decimal Value | Binary Value |
|---|---|---|
| RA+ | -1786 | |
| RA- | | |

จากตารางข้างต้น กำหนดให้ตัวแปรนำเข้า RA+ และ RA- เป็นตัวแปรนำเข้าชนิด R ที่ต้องทำงานเป็นคู่กัน (เนื่องจากตัวแปรนำเข้าชนิดนี้จะทำงานได้ต้องใช้ตัวแปรนำเข้าชนิดนี้ 2 ตัว มาทำงานควบคู่กัน ผู้วิจัยได้ทำการออกแบบให้ตัวแปรนำเข้าชนิดนี้สามารถจับคู่กันโดยการนับเรียงตามลำดับของการลากตัวแปรนำเข้าชนิดนี้มาวางลงบนส่วนวาดวงจร เช่น ตัวที่ 1 คู่กับตัวที่ 2 และตัวที่ 3 คู่กับตัวที่ 4 เป็นอย่างนี้ไปเรื่อย ๆ และที่สำคัญคือ ต้องตั้งชื่อตัวแปรนำเข้าชนิดนี้ขึ้นต้นด้วย R เสมอทุกตัว ไม่ว่าจะเป็นคู่ของตัวเองหรือไม่ เพื่อให้มองภาพการจับคู่ได้ชัดเจน ผู้ออกแบบควรทำการตั้งชื่อตัวแปรนำเข้าชนิดนี้ 2 ตัว ที่เป็นคู่กันให้มีชื่อเหมือนกันแต่มีประจุด้านหลังต่างกัน ตามตัวอย่างในตารางที่ 3.14 เป็นต้น)

จากตารางที่ 3.14 นี้ เมื่อผู้ออกแบบทำการป้อนค่าตัวเลข -1786 ลงในช่องป้อนค่าตัวเลขฐานสิบแล้วทำการกดปุ่มเอ็นเทอร์ (Enter) บนคีบอร์ด เครื่องมือจะแสดงผลของการแปลงค่าตัวเลขดังกล่าวสำหรับตัวแปรนำเข้าชนิด R ดังแสดงในตารางที่ 3.15

ตารางที่ 3.15 แสดงผลของการป้อนค่าตัวเลข -1786 ให้กับตัวแปรนำเข้าชนิด R ชื่อ RA+

| Input Name | Decimal Value | Binary Value |
|---|---|---|
| RA+ | 0 | 0000000000000000 |
| RA- | 1786 | 0000011011111010 |

ตามปกติแล้วค่าตัวเลข -1786 นี้จะต้องถูกแปลงเป็นตัวเลขฐานสองและแสดงในช่องป้อนค่าตัวเลขฐานสองไปควบคู่กันด้วยค่าตัวเลข 00000-1-10-1-1-1-1-10-10 แต่เนื่องจากในการจำลองการทำงาน เครื่องมือจะอ่านค่าจากช่องป้อนค่าตัวเลขฐานสองเท่านั้นไปทำงานทีละบิตเพื่อแสดงค่าผลลัพธ์ที่ถูกต้อง และสามารถอ่านค่าได้เฉพาะตัวเลข 0 กับ 1 เท่านั้น ไม่สามารถอ่านค่าตัวเลข -1 ไปทำงานได้ ฉะนั้นสำหรับตัวแปรนำเข้าชนิด R นี้ ที่ช่องป้อนค่าตัวเลขฐานสองสามารถมีค่าตัวเลข -1 ได้ เครื่องมือจะต้องทำการแปลงค่าในช่องป้อนค่าตัวเลขฐานสองสำหรับตัวแปรชนิดนี้อีกครั้งหนึ่งให้แสดงค่าเฉพาะตัวเลข 0 กับ 1 เท่านั้น จึงได้ผลของการแปลงค่าแล้วแสดงดังตารางที่ 3.15 ข้างต้น สำหรับค่าตัวเลข -1786

ในทำนองเดียวกัน หากผู้ออกทำการป้อนค่าในช่องป้อนค่าตัวเลขฐานสองด้วยค่าตัวเลข 00000-1-10-1-1-1-1-10-10 ซึ่งจะต้องถูกแปลงเป็นตัวเลขฐานสิบด้วยค่าตัวเลข -1786 และแสดงในช่องป้อนค่าตัวเลขฐานสิบเช่นกัน แต่เมื่อผู้ออกแบบทำการกดปุ่มเอ็นเทอร์ เครื่องมือก็จะทำการแสดงผลเช่นเดียวกันกับที่แสดงในตารางที่ 3.15 ข้างต้น

ซึ่งหลักการสำหรับการแปลงค่าตัวเลขฐานสองของตัวแปรนำเข้าชนิด R นี้เพื่อกำจัดตัวเลข -1 มีหลักการแปลงค่า ดังนี้

หลักการแปลงค่าตัวแปรนำเข้าชนิด R

1. สำหรับค่าตัวเลขในช่องป้อนค่าตัวเลขฐานสอง ถ้าค่าตัวเลขที่ตำแหน่งบิตใดก็ตามเป็นตัวเลข 0 จะแทนค่าตัวเลข ณ ตำแหน่งบิตนั้นใหม่ด้วยค่าตัวเลข 00 หรือ 11 เสมอ

2. สำหรับค่าตัวเลขในช่องป้อนค่าตัวเลขฐานสอง ถ้าค่าตัวเลขที่ตำแหน่งบิตใดก็ตามเป็นตัวเลข 1 จะแทนค่าตัวเลข ณ ตำแหน่งบิตนั้นใหม่ด้วยตัวเลข 10 เสมอ

3. สำหรับค่าตัวเลขในช่องป้อนค่าตัวเลขฐานสอง ถ้าค่าตัวเลขที่ตำแหน่งบิตใดก็ตามเป็นตัวเลข -1 จะแทนค่าตัวเลข ณ ตำแหน่งบิตนั้นใหม่ด้วยตัวเลข 01 เสมอ

ตัวอย่างการแปลงค่าตามหลักการดังกล่าวข้างต้นสามารถแสดงได้ดังตารางที่ 3.16 ซึ่งเป็นการทดสอบการแปลงค่าของตัวเลข 1786 และค่าตัวเลข -1786

ตารางที่ 3.16 การทดสอบการแปลงค่าตัวเลข 1786 และค่าตัวเลข -1786 ตามหลักการแปลงค่าของตัวแปรนำเข้าชนิด R

| Decimal Value | Binary Value (1) | Binary Value (2) |
|---|---|---|
| 1786 | 0000011011111010 | 00000000001010001010101010001000 |
| -1786 | 00000-1-10-1-1-1-1-10-10 | 00000000000101000101010101000100 |

จากตารางที่ 3.16 ข้างต้น ค่าที่ได้จากการแปลงในช่อง Binary Value (2) สำหรับค่าตัวเลข -1786 นั้น จะถูกแบ่งออกเป็น 2 ส่วน ตามตำแหน่งหลักเลขคี่ (1 , 3 , 5 , 7 , 9 , … , 31) และตามตำแหน่งหลักเลขคู่ (2 , 4 , 6 , 8 , 10 , … , 32) ซึ่งจะได้ค่าตัวเลขแบ่งเป็น 2 ชุด คือ

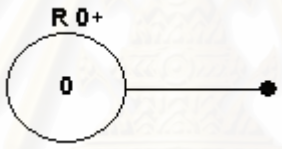
ชุดที่ 1 ได้จากตำแหน่งหลักคี่ ค่าตัวเลขที่ได้คือ 0000000000000000

ชุดที่ 2 ได้จากตำแหน่งหลักคู่ ค่าตัวเลขที่ได้คือ 0000011011111010

ค่าตัวเลขทั้ง 2 ชุดนี้จะถูกนำไปแสดงผลในช่องป้อนค่าตัวเลขฐานสองเพื่อใช้สำหรับการจำลองการทำงานต่อไป โดยค่าตัวเลขชุดแรกจะถูกแสดงผลให้กับตัวแปรนำเข้าชนิด R ชื่อ RA+ และค่าตัวเลขชุดที่ 2 จะถูกแสดงผลให้กับตัวแปรนำเข้าชนิด R ชื่อ RA- ซึ่งเป็นไปตามที่ได้แสดงผลดังตารางที่ 3.15 นั่นเอง

จากคุณลักษณะพิเศษของตัวแปรนำเข้าชนิดนี้ ข้างต้น ผู้วิจัยจึงมีแนวคิดในการออกแบบเครื่องมือให้สามารถใช้งานตัวแปรนำเข้าชนิดนี้ได้โดยทำการจับคู่ให้กับตัวแปรนำเข้าชนิดนี้เป็นคู่ ๆ เพื่อให้สอดคล้องกับลักษณะการทำงานข้างต้นและสามารถเห็นลักษณะการทำงานได้ชัดเจนยิ่งขึ้น

สำหรับการเรียกใช้งานอุปกรณ์ตัวแปรนำเข้าชนิด R นี้ เมื่อผู้ออกแบบทำการลากอุปกรณ์ตัวแปรนำเข้าชนิดนี้ ไปวางลงบนส่วนวาดวงจร เครื่องมือจะวาดรูปแทนอุปกรณ์ตัวแปรนำเข้าชนิดนี้ขึ้นมาหนึ่งรูปตั้งชื่อว่า R0+ สัญลักษณ์แสดงดังรูปที่ 3.11




รูปที่ 3.12 สัญลักษณ์แทนอุปกรณ์ตัวแปรนำเข้าชนิด R

ตารางที่ 3.17 แสดงการเพิ่มจำนวนแถวในส่วนกำหนดค่าตัวแปรนำเข้าเมื่อเลือกอุปกรณ์ตัวแปรนำเข้าชนิด R

| Input Name | Decimal Value | Binary Value |
|---|---|---|
| R 0+ | | |
| R 1+ | | |

ผู้ออกแบบสามารถเปลี่ยนชื่อตัวแปรนำเข้าได้เองตามต้องการ แต่มีหลักการอยู่ว่าจะต้องตั้งชื่อขึ้นต้นด้วยตัวอักษร R เสมอ ไม่เช่นนั้นตัวแปรนำเข้าตัวนั้นจะไม่สามารถทำงานเป็นตัวแปรนำเข้าชนิด R ได้ถูกต้อง ในขณะเดียวกันกับที่เครื่องมือวาดรูปแทนอุปกรณ์ตัวแปรนำเข้าชนิด R นี้ ลงบนส่วนวาดวงจร เครื่องมือก็จะทำการเพิ่มจำนวนแถวในส่วนกำหนดค่าตัวแปรนำเข้าจำนวนหนึ่งแถวด้วยเช่นกัน และหากมีการลากตัวแปรนำเข้าชนิด R นี้มาวางลงบนส่วนวาดวงจรเพิ่มขึ้นอีกหนึ่งตัวเครื่องมือก็จะตั้งชื่อว่า R 1+ และทำการเพิ่มจำนวนแถวในส่วนกำหนดค่าตัวแปรนำเข้าเพิ่มขึ้นอีกหนึ่งแถว แสดงได้ดังตารางที่ 3.17

ดังนั้น เพื่อให้สอดคล้องกับลักษณะพิเศษของตัวแปรนำเข้าชนิด R ที่ได้กล่าวไว้ข้างต้น ผู้ออกแบบต้องเรียกใช้งานอุปกรณ์ตัวแปรนำเข้าชนิด R นี้ เป็นคู่ ๆ จึงจะสามารถใช้งานได้ถูกต้อง ซึ่งเครื่องมือจะทำการจับคู่อุปกรณ์ตัวแปรนำเข้าชนิด R นี้ในลักษณะคือ อุปกรณ์ตัวแปรนำเข้าชนิด R ที่ลากมาวางตัวที่ 1 จะจับคู่กับอุปกรณ์ตัวแปรนำเข้าชนิด R ที่ลากมาวางตัวที่ 2 และอุปกรณ์ตัวแปรนำเข้าชนิด R ที่ลากมาวางตัวที่ 3 จะจับคู่กับอุปกรณ์ตัวแปรนำเข้าชนิด R ที่ลากมาวางตัวที่ 4 จะเป็นลักษณะเช่นนี้ไปเรื่อย ๆ เพื่อให้การจับคู่กันของอุปกรณ์ตัวแปรนำเข้าชนิดนี้มองเห็นได้ชัดเจนขึ้น ผู้ออกแบบควรตั้งชื่อตัวแปรนำเข้าชนิด R นี้ ให้มีชื่อที่เหมือนกันแต่มีประจุตามหลังต่างกัน เช่น ตั้งชื่ออุปกรณ์ตัวแปรนำเข้าชนิด R นี้ตัวที่ 1 ด้วยชื่อ R0+ และตั้งชื่ออุปกรณ์ตัวแปรนำเข้าชนิด R นี้ตัวที่ 2 ด้วยชื่อ R0- ตั้งชื่ออุปกรณ์ตัวแปรนำเข้าชนิด R นี้ ตัวที่ 3 ด้วยชื่อ R1+ และตั้งชื่ออุปกรณ์ตัวแปรนำเข้าชนิด R นี้ ตัวที่ 4 ด้วยชื่อ R1- เป็นต้น

เนื่องจากอุปกรณ์ตัวแปรนำเข้าชนิด R นี้ ต้องเรียกใช้งานเป็นคู่ ๆ จึงจะสามารถทำงานได้ ดังนี้ผู้วิจัยจึงทำการออกแบบให้สามารถป้อนค่าตัวเลขให้กับตัวแปรนำเข้าชนิดR นี้ ได้เฉพาะตัวแรกในแต่ละคู่เท่านั้น โดยสามารถป้อนค่าได้ทั้งช่องป้อนค่าตัวเลขฐานสิบหรือช่องป้อนค่าตัวเลขฐานสอง ตัวอย่างเช่น ผู้ออกแบบทำการลากอุปกรณ์ตัวแปรนำเข้าชนิด R นี้ มาวางลงบนส่วนวาดวงจรจำนวน 2 ตัว โดยอุปกรณ์ตัวที่ 1 ตั้งชื่อว่า R0+ และอุปกรณ์ตัวที่ 2 ตั้งชื่อว่า R0- เมื่อผู้ออกแบบทำการป้อนค่าในช่องป้อนค่าตัวเลขฐานสิบด้วยค่าตัวเลข -5 เมื่อทำการกดปุ่มเอ็นเทอร์บนคีย์บอร์ดแล้ว เครื่องมือจะทำการแสดงค่าสำหรับตัวแปรที่ชื่อ R0+ ในช่องป้อนค่าตัวเลขฐานสิบด้วยค่าตัวเลข 0 และแสดงค่าในช่องป้อนค่าตัวเลขฐานสองด้วยค่าตัวเลข 0000000000000000 และเครื่องมือจะทำการแสดงค่าสำหรับตัวแปรที่ชื่อ R0- ในช่องป้อนค่าตัวเลขฐานสิบด้วยค่าตัวเลข 5 และแสดงค่าในช่องป้อนค่าตัวเลขฐานสองด้วยค่าตัวเลข 0000000000000101 ดังแสดงในตารางที่ 3.18

ตารางที่ 3.18 แสดงผลการกำหนดค่าตัวเลขให้กับตัวแปรนำเข้าชนิด R ชื่อ R0+

| Input Name | Decimal Value | Binary Value |
|---|---|---|
| R0+ | 0 | 0000000000000000 |
| R0- | 5 | 0000000000000101 |

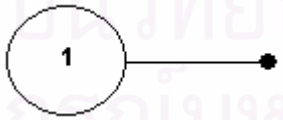
ในทำนองเดียวกัน ผู้ออกแบบสามารถป้อนค่าในช่องป้อนค่าตัวเลขฐานสองของตัวแปรนำเข้าชนิด R นี้ ที่ชื่อ R0+ ได้ด้วยตัวเลข 0000000000000-10-1 เมื่อทำการกดปุ่มเอ็นเทอร์

บนคีย์บอร์ดแล้ว เครื่องมือก็จะแสดงผลการกำหนดค่าให้กับตัวแปรนำเข้าชนิด R นี้ ทั้งตัวแปรนำเข้าชนิด R ที่ชื่อ R0+ และชื่อ R0- ด้วยค่าเช่นเดียวกันกับที่แสดงดังตารางที่ 3.18 ข้างต้น

ในการประกอบอุปกรณ์ตัวแปรนำเข้าชนิด R นี้ เข้ากับอุปกรณ์หรือวงจรย่อยต่าง ๆ ที่มีในวงจรที่ทำการสร้างขึ้นนั้น ผู้ออกแบบสามารถลากอุปกรณ์ตัวแปรนำเข้าชนิด R นี้ ไปเชื่อมต่อเข้ากับอุปกรณ์ต่าง ๆ ได้ตามต้องการ โดยไม่จำเป็นว่าอุปกรณ์ตัวแปรนำเข้าชนิด R นี้ที่เป็นคู่กันต้องต่อเข้ากับอุปกรณ์หรือวงจรย่อยตัวเดียวกัน

ตัวแปรนำเข้าชนิด R นี้ ยังเป็นตัวแปรนำเข้าชนิดที่มีความพิเศษอยู่อย่างหนึ่งคือสามารถแทนค่าได้ด้วยค่าตัวเลขในฐานสองได้มากกว่าหนึ่งรูปแบบอีกด้วย ตัวอย่างเช่น ในช่องป้อนค่าตัวเลขฐานสองของตัวแปรนำเข้าชนิด R นี้ ที่ชื่อ R0+ นั้น ผู้ออกแบบสามารถป้อนค่าตัวเลข 0000000000000-10-1 หรือป้อนค่าตัวเลข 0000000000000-1-11 ก็ได้ ซึ่งก็จะมีค่าเท่ากับ -5 ในตัวเลขฐานสิบเช่นเดียวกัน โดยค่าตัวเลข 0000000000000-10-1 คำนวณเป็นค่าตัวเลขฐานสิบ ได้จาก $(0x2^{15}) + (0x2^{14}) + (0x2^{13}) + (0x2^{12}) + (0x2^{11}) + (0x2^{10}) + (0x2^{9}) + (0x2^{8}) + (0x2^{7}) + (0x2^{6}) + (0x2^{5}) + (0x2^{4}) + (0x2^{3}) + (-1x2^{2}) + (0x2^{1}) + (-1x2^{0}) = -5$ และค่าตัวเลข 0000000000000-1-11 คำนวณเป็นค่าตัวเลขฐานสิบ ได้จาก $(0x2^{15}) + (0x2^{14}) + (0x2^{13}) + (0x2^{12}) + (0x2^{11}) + (0x2^{10}) + (0x2^{9}) + (0x2^{8}) + (0x2^{7}) + (0x2^{6}) + (0x2^{5}) + (0x2^{4}) + (0x2^{3}) + (-1x2^{2}) + (-1x2^{1}) + (1x2^{0}) = -5$ เช่นกัน

3. อุปกรณ์ตัวแปรนำเข้าชนิดค่าคงที่ (Constant) เป็นอุปกรณ์ตัวแปรนำเข้าชนิดที่จะทำการส่งค่าเข้าสู่วงจร เพื่อทดสอบจำลองการทำงานด้วยค่าคงที่ค่าใดค่าหนึ่งเท่านั้น (ค่า 0 หรือค่า 1) ตลอดเวลาของการจำลองการทำงาน สัญลักษณ์แสดงดังรูปที่ 3.13



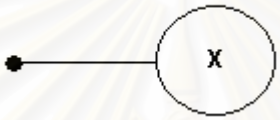

รูปที่ 3.13 สัญลักษณ์แทนอุปกรณ์ตัวแปรนำเข้าชนิดค่าคงที่

การเรียกใช้งานอุปกรณ์ตัวแปรนำเข้าชนิดค่าคงที่ เมื่อผู้ออกแบบลากอุปกรณ์ตัวแปรนำเข้าชนิดค่าคงที่นี้มาวางลงบนส่วนวาดวงจร เครื่องมือจะทำการวาดรูปแทนอุปกรณ์ตัวแปรนำเข้าชนิดค่าคงที่นี้ลงบนส่วนวาดวงจรเพียงอย่างเดียวโดยไม่ทำการเพิ่มจำนวนแถวให้กับส่วนกำหนดค่าตัวแปรนำเข้า เนื่องจากตัวแปรนำเข้าชนิดนี้จะมีค่าที่ใช้ในการจำลองการทำงานด้วยค่า

ตัวเลข 0 หรือค่าตัวเลข 1 เพียงค่าเดียวตลอดเวลาขณะทำการจำลองการทำงาน โดยผู้ออกแบบสามารถเลือกเปลี่ยนค่าของอุปกรณ์ตัวแปรนำเข้าชนิดค่าคงที่นี้สลับกันด้วยค่าตัวเลข 0 หรือค่าตัวเลข 1 ได้ด้วยการคลิกเมาส์ขวาที่ตัวอุปกรณ์แล้วทำการเลือกเมนูเปลี่ยนค่า (Change value)

ลักษณะของอุปกรณ์ตัวแปรผลลัพธ์ (Output)

อุปกรณ์ตัวแปรผลลัพธ์เป็นอุปกรณ์ที่จะแสดงค่าผลลัพธ์ทั้งหมดที่ได้จากการจำลองการทำงาน โดยจะไม่มีการแปลงค่าตัวเลขใด ๆ ทั้งสิ้น ค่าตัวเลขที่แสดงในแต่ละรอบนั้นเป็นค่าตัวเลขที่เป็นผลลัพธ์แล้วของในแต่รอบ สัญลักษณ์แสดงดังรูปที่ 3.14




รูปที่ 3.14 สัญลักษณ์แทนอุปกรณ์ตัวแปรผลลัพธ์

การเรียกใช้งานอุปกรณ์ตัวแปรผลลัพธ์นี้ เมื่อผู้ออกแบบลากอุปกรณ์ตัวแปรผลลัพธ์นี้มาวางลงบนส่วนวาดวงจร เครื่องมือจะทำการวาดรูปแทนอุปกรณ์ตัวแปรผลลัพธ์ลงบนส่วนวาดวงจรและทำการเพิ่มจำนวนแถวในส่วนแสดงค่าผลลัพธ์จำนวนหนึ่งแถวด้วยเช่นกัน โดยตั้งชื่อว่า Output 0 และหากมีการลากอุปกรณ์ตัวแปรผลลัพธ์นี้มาวางลงบนส่วนวาดวงจรเพิ่มขึ้นอีกหนึ่งตัว เครื่องมือก็จะตั้งชื่อว่า Output 1 โดยในส่วนแสดงค่าผลลัพธ์นี้จะไม่สามารถป้อนค่าตัวเลขใด ๆ ลงไปได้ โดยค่าตัวเลขจะถูกแสดงก็ต่อเมื่อทำการทดสอบจำลองการทำงานเท่านั้น และผู้ออกแบบสามารถเปลี่ยนชื่อตัวแปรผลลัพธ์ได้ตามต้องการ

### 3.2.2 ส่วนการจำลองการทำงาน (Simulation)

ส่วนการจำลองการทำงานของเครื่องมือเป็นส่วนที่ทำหน้าที่เกี่ยวข้องกับการกำหนดค่าตัวแปรนำเข้าเพื่อทดสอบจำลองการทำงานและการควบคุมการจำลองการทำงาน โดยผู้ออกแบบเป็นผู้ควบคุมการทำงานผ่านทางส่วนต่อประสานกับผู้ใช้ ผู้วิจัยได้ทำการแบ่งส่วนการจำลองการทำงานนี้ ออกเป็นส่วนย่อย 3 ส่วน คือ

#### 3.2.2.1 ส่วนกำหนดค่าตัวแปรนำเข้าเพื่อแสดงค่าผลลัพธ์ (Input/Output Simulation)

ส่วนกำหนดค่าตัวแปรนำเข้าเพื่อแสดงค่าผลลัพธ์ คือ ส่วนที่สร้างเตรียมไว้สำหรับให้ผู้ออกแบบทำการป้อนค่าตัวเลขให้กับตัวแปรนำเข้าชนิดต่าง ๆ เพื่อการทดสอบการจำลองการ

ทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรงจากอุปกรณ์หรือวงจรที่สร้างขึ้น ซึ่งแบ่งออกได้เป็น 2 ส่วน ดังนี้

1) ส่วนกำหนดค่าตัวแปรนำเข้า (Input) เป็นส่วนที่ออกแบบเตรียมไว้สำหรับให้ผู้ออกแบบได้ป้อนค่าตัวเลขให้กับตัวแปรนำเข้าตามแต่ชนิดของตัวแปรนำเข้าเพื่อทดสอบจำลองการทำงานอุปกรณ์หรือวงจรที่สร้างขึ้น โดยมีช่องสำหรับป้อนค่าตัวเลขอยู่ด้วยกัน 2 ช่อง คือ ช่องป้อนค่าตัวเลขฐานสิบและช่องป้อนค่าตัวเลขฐานสอง ผู้ออกแบบสามารถเลือกทำการป้อนค่าที่ช่องใดช่องหนึ่งก็ได้ โดยเครื่องมือจะทำการแปลงค่าที่ถูกต้องของอีกช่องให้ควบคู่กันไปด้วย แต่ในการอ่านค่าตัวเลขเข้าไปจำลองการทำงานนั้น เครื่องมือจะอ่านค่าตัวเลขจากช่องป้อนค่าตัวเลขฐานสองไปทำงานเพื่อแสดงค่าผลลัพธ์ที่ถูกต้อง เครื่องมือที่พัฒนาขึ้นได้ออกแบบจำนวนบิตของตัวเลขฐานสองไว้สูงสุดที่ 16 บิต ถ้าหากทำการป้อนค่าตัวเลขฐานสิบแล้วแปลงค่าได้ค่าตัวเลขฐานสองที่เกิน 16 บิต จะไม่อนุญาต แต่สำหรับตัวแปรนำเข้าชนิด R นั้น อนุญาตให้ผู้ออกแบบสามารถป้อนค่าได้ทั้งในช่องป้อนค่าตัวเลขฐานสิบและช่องป้อนค่าตัวเลขฐานสองสำหรับตัวแปรนำเข้าชนิด R ตัวแรกในแต่ละคู่ได้เท่านั้น ผู้ออกแบบสามารถลากอุปกรณ์ตัวแปรนำเข้าจากคลังอุปกรณ์มาประกอบในวงจรที่สร้างขึ้นโดยมีทั้งอุปกรณ์ตัวแปรนำเข้าชนิด NR และอุปกรณ์ตัวแปรนำเข้าชนิด R ประกอบอยู่ร่วมกันในวงจรที่ออกแบบได้

2) ส่วนแสดงค่าผลลัพธ์ (Output) เป็นส่วนที่ออกแบบเตรียมไว้สำหรับการแสดงค่าผลลัพธ์ที่เกิดจากการจำลองการทำงานผ่านทางตัวแปรผลลัพธ์ โดยแสดงค่าผลลัพธ์ด้วยค่าตัวเลขฐานสิบและค่าตัวเลขฐานสองไปควบคู่กัน สำหรับผลลัพธ์ของการจำลองการทำงาน การเปลี่ยนแปลงของค่าตัวเลขที่ส่งเข้ามาที่ตัวแปรนำเข้าอาจทำให้เกิดการเปลี่ยนแปลงค่าที่ตัวแปรผลลัพธ์ในช่วงเวลาถัดไปขึ้นอยู่ที่ค่าความหน่วงของอุปกรณ์หรือวงจรนั้น ๆ

3.2.2.2 ส่วนควบคุมการจำลองการทำงาน (Control Simulation)

ส่วนควบคุมการจำลองการทำงานของอัลกอริทึมหรือวงจรที่ออกแบบ เป็นส่วนที่เกี่ยวข้องกับการแสดงผลการทำงานผ่านทางส่วนต่อประสานผู้ใช้นั่นเอง ซึ่งในส่วนนี้เครื่องมือได้เตรียมการควบคุมสำหรับให้ผู้ออกแบบเป็นผู้กำหนดการทำงานเองไว้ 2 ส่วน คือ

1) ส่วนควบคุมความเร็วการจำลองการทำงาน (Simulation Speed) เพื่อให้ผู้ออกแบบได้เลือกระดับความเร็วที่ต้องการในการแสดงผลการจำลองการทำงานในแต่ละครั้ง

2) ส่วนประมวลผลการจำลองการทำงาน เป็นส่วนที่รองรับการแสดงผลลัพธ์ของการจำลองการทำงาน ซึ่งผู้ออกแบบสามารถเลือกประมวลผลได้ 2 รูปแบบ คือ เลือกให้การจำลองการทำงานเกิดขึ้นอย่างต่อเนื่องจนแล้วเสร็จหรือทำไปทีละขั้น โดยผู้ออกแบบเป็นผู้ควบคุมการทำงานผ่านทางส่วนต่อประสานกับผู้ใช้

3.2.2.3 ส่วนแสดงจำนวนคล็อก (Clock Counter) เป็นส่วนที่แสดงจำนวนคล็อกของการทำงานที่เกิดจากการเลือกจำลองการทำงานให้เกิดขึ้นอย่างต่อเนื่องจนแล้วเสร็จหรือเลือกทำไปทีละขั้น เพื่อประโยชน์ในการตรวจสอบความเร็วและความถูกต้องของวงจรที่ออกแบบ

3.2.3 ส่วนแถบเมนูและแถบเครื่องมือ

ส่วนแถบเมนูและแถบเครื่องมือ จะเป็นส่วนที่เก็บชุดคำสั่งการทำงานต่าง ๆ ของเครื่องมือเอาไว้ ผู้ออกแบบสามารถเลือกชุดคำสั่งการทำงานจากแถบเมนูหรือแถบเครื่องมือก็ได้ซึ่งจะให้ผลการทำงานที่เหมือนกัน ซึ่งส่วนของแถบเมนูจะประกอบไปด้วย

1) เมนูไฟล์ (File) จะเก็บชุดคำสั่งเกี่ยวกับการเก็บบันทึกอุปกรณ์หรือวงจรที่ออกแบบและการเรียกอุปกรณ์หรือวงจรเหล่านั้นกลับขึ้นมาทำงาน ซึ่งประกอบด้วยชุดคำสั่ง 5 ประเภทด้วยกัน คือ

1.1) คำสั่งสร้างไฟล์ใหม่ (New file) สำหรับการเริ่มต้นการทำงานใหม่บนส่วนวาดวงจร ทุกครั้งที่ทำการเลือกคำสั่งนี้ เครื่องมือจะแจ้งเตือนให้ผู้ออกแบบเลือกกระทำคำสั่งบันทึก (Save) หรือบันทึกเป็น (Save as) ก่อน เพราะจะทำการเคลียร์ทุกอย่างออกจากส่วนวาดวงจร

1.2) คำสั่งเรียกไฟล์มาทำงาน (Load file) สำหรับการเรียกไฟล์อุปกรณ์หรือวงจรที่เคยบันทึกเก็บไว้ขึ้นมาทำงาน เพื่อทำการแก้ไขหรือทดสอบจำลองการทำงานก็ได้

1.3) คำสั่งบันทึก (Save) สำหรับการเก็บบันทึกอุปกรณ์หรือวงจรที่สร้างขึ้นลงไฟล์ โดยจะบันทึกในรูปแบบแฟ้มเอกสารเอ็กซ์เอ็มแอล

1.4) คำสั่งบันทึกเป็น (Save as) สำหรับการเก็บบันทึกอุปกรณ์หรือวงจรที่ทำการแก้ไขด้วยชื่อใหม่ เพื่อป้องกันการตั้งชื่อซ้ำกับอุปกรณ์เดิมที่มีอยู่ เพราะการบันทึกซ้ำด้วยชื่อเดิมอาจจะส่งผลกับการทำงานของอุปกรณ์หรือวงจรที่เคยสร้างได้แล้วที่มีการนำอุปกรณ์หรือวงจรชื่อนี้ไปเป็นส่วนประกอบด้วย

1.5 คำสั่งออกจากโปรแกรม (Exit) สำหรับการออกจากเครื่องมือ เมื่อต้องการเลิกการใช้งาน

2) เมนูแก้ไข (Edit) จะเก็บชุดคำสั่งเกี่ยวกับการยกเลิกหรือกระทำซ้ำการทำงานตลอดจนการคัดลอกอุปกรณ์เพิ่มลงบนส่วนวาดวงจรหรือการลบอุปกรณ์ออกจากส่วนวาดวงจร ซึ่งประกอบด้วยชุดคำสั่ง 5 ประเภทด้ยวกัน คือ

2.1) คำสั่งยกเลิกการทำงาน (Undo) สำหรับการยกเลิกการทำงานล่าสุดที่ผู้ออกแบบทำบนเครื่องมือ

2.2) คำสั่งการทำซ้ำการทำงาน (Redo) สำหรับการทำซ้ำการทำงานล่าสุดที่ถูกยกเลิกการทำงาน

2.3) คำสั่งคัดลอก (Copy) สำหรับการคัดลอกอุปกรณ์หรือวงจรที่มีอยู่บนส่วนวาดวงจรและเพิ่มอุปกรณ์หรือวงจรตัวนั้นเข้าไป

2.4) คำสั่งลบ (Delete) สำหรับการลบอุปกรณ์หรือวงจรที่มีอยู่ออกจากส่วนวาดวงจร

2.5) คำสั่งเลือกทั้งหมด (Select All) สำหรับการเลือกอุปกรณ์หรือวงจรที่มีอยู่บนส่วนวาดวงจรทั้งหมดทุกตัวอาจเพื่อทำคำสั่ง Delete หรือเพื่อการย้ายตำแหน่งของอุปกรณ์หรือวงจรที่ทำการเลือก

3) เมนูตัวเลือก (Options) จะเก็บชุดคำสั่งเกี่ยวกับการเลือกออกแบบสร้างวงจรด้วยสมการบูลีน การสร้างอุปกรณ์ UDP และการสร้างอุปกรณ์ UDE ประกอบด้วยชุดคำสั่ง 2 ประเภทด้วยกัน คือ

3.1) คำสั่งเลือกออกแบบ (Custom design) สำหรับการแสดงเมนูเลือกสร้างวงจรด้วยสมการบูลีนหรือสร้างอุปกรณ์ UDP หรือสร้างอุปกรณ์ UDE

3.2) คำสั่งตารางค่าความจริงของวงจร (Circuit truth table) สำหรับผู้ออกแบบเลือกดูตารางค่าความจริงของวงจรที่สร้างขึ้น

4) เมนูการจำลอง (Simulate) จะเก็บชุดคำสั่งที่เกี่ยวกับการจำลองการทำงานของอุปกรณ์หรือวงจรที่สร้างขึ้น ประกอบด้วยชุดคำสั่ง 3 ประเภทด้วยกัน คือ

4.1) คำสั่งเริ่มทำงานทั้งหมด (Start) สำหรับเลือกจำลองการทำงานให้เกิดขึ้นอย่างต่อเนื่องจนแล้วเสร็จ

4.2) คำสั่งเริ่มทำงานทีละขั้น (Step) สำหรับเลือกจำลองการทำงานให้เกิดขึ้นทีละขั้น

4.3) คำสั่งหยุดทำงาน (Stop) สำหรับการสั่งหยุดการจำลองการทำงาน

5) เมนูช่วยเหลือ (Help) จะเก็บชุดคำสั่งที่เกี่ยวกับข้อมูลเครื่องมือ ประกอบด้วยชุดคำสั่ง 2 ประเภทด้วยกัน คือ

5.1) คำสั่งช่วยเหลือ (Help contents) สำหรับคำแนะนำการใช้งานเครื่องมือ

5.2) คำสั่งเกี่ยวกับ (About) สำหรับแสดงรายละเอียดเครื่องมือ

สำหรับส่วนแถบเครื่องมือจะประกอบด้วยปุ่มต่าง ๆ แสดงด้วยไอคอนซึ่งเหมือนกันกับในแถบเมนูเพื่อความสะดวกในการเลือกใช้งาน

3.2.4 ส่วนแถบสถานะการทำงาน

ส่วนแถบสถานะการทำงาน จะเป็นส่วนที่แสดงข้อมูลเกี่ยวกับสถานะการทำงานในขณะนั้นของเครื่องมือ

## 3.3 การประยุกต์เครื่องมือกับอัลกอริทึมการคำนวณแบบเชื่อมตรง

เนื่องจากการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรงนั้นมีลักษณะการทำงาน คือข้อมูลนำเข้าจะถูกนำไปประมวลผลแบบลำดับควบคู่ไปกับการคำนวณแบบท่อตรง และให้ผลลัพธ์ในลักษณะตำแหน่งต่อตำแหน่ง โดยทุกข้อมูลนำเข้าจะถูกพิจารณาตามลำดับจากหลักที่มีนัยสำคัญมากสุดไปยังหลักที่มีนัยสำคัญต่ำสุด เช่นเดียวกับข้อมูลนำเข้า ผลลัพธ์ก็ถูกคำนวณและแสดงออกมาในรูปแบบเดียวกันคือตามลำดับ ข้อมูลนำเข้าหรือตัวดำเนินการแต่ละตัวจะถูกส่งมาในวงจรครั้งละหนึ่งบิตและบิตถัดไปจะทยอยมาตามลำดับเวลาที่ดำเนินไปด้วยกัน โดยการคำนวณไม่ขึ้นกับความซับซ้อน หรือความยาวของตัวถูกดำเนินการที่ใช้ในการคำนวณ

จากลักษณะการทำงานดังกล่าว ผู้วิจัยได้นำมาประยุกต์เข้ากับเครื่องมือที่พัฒนาขึ้นเพื่อจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรง โดยการทำงานจะอยู่ภายใต้ส่วนประมวลผลการจำลองการทำงานและแสดงผลการทำงานผ่านทางส่วนต่อประสานกับผู้ใช้ มีหลักการทำงานดังนี้

เมื่อผู้ออกแบบทำการทดสอบจำลองการทำงานจากอุปกรณ์หรือวงจรที่สร้างขึ้นโดยการเลือกกดปุ่มเริ่มทำงานทั้งหมด (Start) หรือปุ่มเริ่มทำงานทีละขั้น (Step) ก็ตาม เครื่องมือจะเริ่มต้นการทำงานรอบแรกหรือคล็อกแรกด้วยการอ่านค่าตัวเลขจากช่องป้อนค่าตัวเลขฐานสองของตัวแปรนำเข้าชนิดต่าง ๆ ทุกตัวจากบิตตำแหน่งซ้ายมือสุดก่อนเพื่อส่งค่าเข้าไปทำงานในวงจรพร้อมกันเพื่อผลิตผลลัพธ์บิตแรกของทุกตัวแปรผลลัพธ์ออกมา จากนั้นจะขยับไปยังบิตถัดไปเรื่อย ๆ (จากซ้ายสุดไปขวาสุด) จนครบทุกตำแหน่ง ผลลัพธ์ก็จะถูกผลิตออกมาตามลำดับของตำแหน่งที่ขยับเช่นกัน นอกจากนี้ถ้าภายในวงจรมีอุปกรณ์เกตดีเลย์หรืออุปกรณ์ UDP ซึ่งเป็นอุปกรณ์ที่มีค่าความหน่วง ประกอบอยู่ภายในวงจรที่ออกแบบด้วยนั้น ถึงแม้ว่าเครื่องมือจะขยับตำแหน่งไปจนถึงตำแหน่งสุดท้ายแล้วก็ตาม ถ้าหากว่าผลลัพธ์ยังไม่ถูกผลิตออกมาจนหมด เครื่องมือก็จะยังทำงานต่อจนกระทั่งผลลัพธ์ถูกผลิตออกมาจนหมด

โดยสรุปแล้วการแสดงผลลัพธ์ของส่วนจำลองการทำงานของเครื่องมือนี้จะแสดงผ่านทางส่วนต่อประสานกับผู้ใช้ใน 2 รูปแบบ คือ ผลลัพธ์ที่แสดงในรูปของกราฟิกซึ่งถูกแสดงในแต่ละขั้นตอนผ่านทางส่วนวาดวงจร และผลลัพธ์ทั้งหมดในรูปแบบของเลขฐานสองและเลขฐานสิบผ่านทางส่วนจำลองการทำงานของเครื่องมือในส่วนแสดงค่าผลลัพธ์

## 3.4 การประยุกต์ใช้งานเครื่องมือบนระบบอินเทอร์เน็ต

สำหรับการประยุกต์ใช้งานเครื่องมือจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรงกับระบบอินเทอร์เน็ตนั้น ผู้วิจัยได้ทำการพัฒนาเครื่องมือให้ใช้งานได้ในรูปแบบของแอพเพล็ท ซึ่งก็คือแอพลิเคชั่นที่ถูกพัฒนาขึ้นมาจากภาษาจาวาที่ทำงานบนพื้นที่ใช้งานบราวเซอร์อีกที โดยที่บราวเซอร์จะต้องติดตั้งเจวีเอ็ม (JVM : Java Virtual Machine) หรือจาวาปลั๊กอิน (Java Plug-in) ไว้รองรับการทำงาน เนื่องจากโปรแกรมภาษาจาวาทุกตัวจะต้องทำงานอยู่ภายใต้เจวีเอ็ม และหลักการทำงานของแอพเพล็ทก็คือ การสร้างคอนเน็กชั่นไปหาแอพลิเคชั่นที่เครื่องคอมพิวเตอร์แม่ข่าย (Server) หลังจากนั้นก็จะทำการโหลดแอพลิเคชั่นไปอยู่บนฝั่งของเครื่องคอมพิวเตอร์ลูกข่าย (Client) และใช้ซีพียู เมมโมรี่บนฝั่งเครื่องคอมพิวเตอร์ลูกข่าย ส่วนว่าจะมีการติดต่อกลับไปยังเครื่องคอมพิวเตอร์แม่ข่ายหรือไม่ขึ้นอยู่กับการทำงานของแอพเพล็ทแต่ไม่จำเป็นต้องมีการติดต่อกลับไป บางครั้งแอพเพล็ทอาจมีการดึงข้อมูลจากเครื่องคอมพิวเตอร์แม่ข่ายมาบ้าง แต่เฉพาะข้อมูลส่วนการประมวลผลก็ยังอยู่ฝั่งเครื่องคอมพิวเตอร์ลูกข่ายอยู่ดี

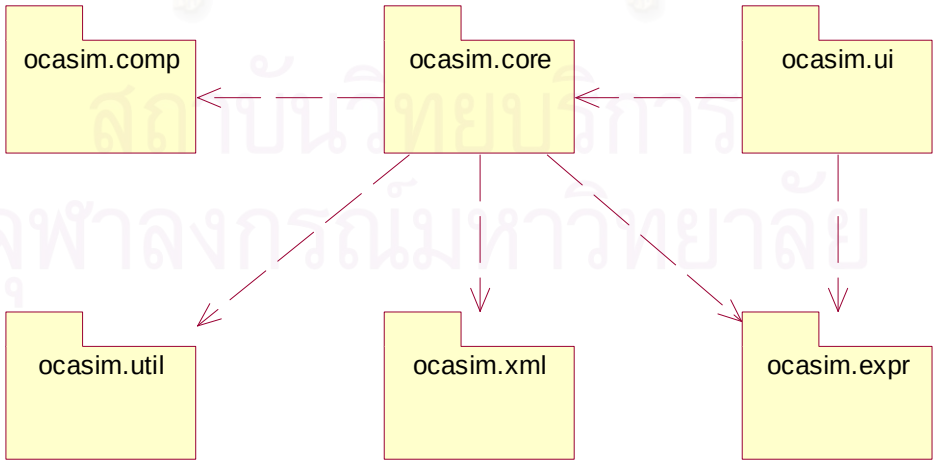
# บทที่ 4

# การออกแบบเครื่องมือจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรง

จากบทที่แล้ว ผู้วิจัยได้แสดงแนวคิดและทำการวิเคราะห์องค์ประกอบของเครื่องมือเพื่อนำมาใช้ในขั้นตอนการออกแบบเครื่องมือ ซึ่งจุดประสงค์หลักก็เพื่อให้ได้เครื่องมือที่เป็นประโยชน์แก่ผู้ที่ต้องการศึกษาการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรง ในบทนี้จะกล่าวถึงการออกแบบเครื่องมือตามแนวคิดและการวิเคราะห์องค์ประกอบของเครื่องมือจากบทที่แล้ว โดยใช้แผนภาพยูสเคส (Use Case Diagram) แผนภาพคลาส (Class Diagram) และแผนภาพซีเควนซ์ (Sequence Diagram) เป็นเครื่องมือในการอธิบายการออกแบบ โดยมีรายละเอียดดังต่อไปนี้

## 4.1 ภาพรวมของการออกแบบเครื่องมือ

จากแนวคิดและการวิเคราะห์จากบทที่แล้วนั้น ผู้วิจัยได้นำมาออกแบบเครื่องมือโดยเลือกใช้วิธีการเชิงวัตถุในการออกแบบ โดยทำการจัดกลุ่มคลาสต่าง ๆ เน้นที่โครงสร้างของคลาสเป็นหลักจัดให้คลาสที่สัมพันธ์กันรวมอยู่ในแพ็กเกจเดียวกันเพื่อง่ายต่อการนำไปใช้งาน เครื่องมือที่ออกแบบจะประกอบไปด้วยแพ็กเกจต่าง ๆ จำนวน 6 แพ็กเกจหลัก ซึ่งสามารถแสดงความสัมพันธ์ระหว่างแพ็กเกจต่าง ๆ ของระบบได้ดังรูปที่ 4.1




รูปที่ 4.1 แผนภาพของแพ็กเกจแสดงองค์ประกอบของระบบงาน

โดยที่แต่ละแพ็กเกจจะเป็นการจัดกลุ่มความสัมพันธ์ให้แก่คลาสต่าง ๆ ให้เป็นหมวดหมู่เพื่องายต่อการนำไปใช้ต่อไป โดยในแต่ละแพ็กเกจจะเก็บรายละเอียดดังนี้

1) แพ็กเกจ ocasim.comp เป็นแพ็กเกจสำหรับเก็บโมเดลคลาสของอุปกรณ์ต่าง ๆ ได้แก่ อุปกรณ์เกตแอนด์ (AND), แนนด์ (NAND), ออร์ (OR), นอร์ (NOR), เอ็กซ์คลูซีฟออร์ (XOR), เอ็กซ์คลูซีฟนอร์ (XNOR), อินเวอร์เตอร์ (NOT), บัฟเฟอร์ (BUFFER) และดีเลย์ (DELAY) นอกจากนี้ก็มีอุปกรณ์ตัวแปรนำเข้าชนิด NR, อุปกรณ์ตัวแปรนำเข้าชนิด R,อุปกรณ์ตัวแปรผลลัพธ์ อุปกรณ์ UDP และอุปกรณ์ UDE ตัวอย่างคลาสในแพ็กเกจนี้ เช่น คลาส AndCompont, คลาส OrComponent, คลาส NotComponent, คลาส UDPComponent, คลาส UDEComponent, คลาส InputTerminal, คลาส InputRTerminal, คลาส OutputTerminal เป็นต้น

2) แพ็กเกจ ocasim.core เป็นแพ็กเกจสำหรับเก็บชุดของคลาสที่ทำหน้าที่สร้างและควบคุมส่วนต่อประสานกับผู้ใช้ต่าง ๆ รวมทั้งควบคุมการกระทำต่าง ๆ กับระบบงาน เช่น การวาดรูปอุปกรณ์ต่าง ๆ การออกแบบสร้างอุปกรณ์หรือวงจร การบันทึกข้อมูลเอ็กซ์เอ็มแอลลงไฟล์ และการจำลองการทำงานวงจรที่ออกแบบ ตัวอย่างคศาสในแพ็กเกจนี้ เช่น คลาส Actions, คลาส Command, คลาส ComponentModel, คลาส CircuitModel, คลาส ICModel, คลาส CircuitMemento, คลาส ComponentMemento , คลาส SimulationRunner เป็นต้น

3) แพ็กเกจ ocasim.ui เป็นแพ็กเกจสำหรับเก็บชุดของคลาสที่ใช้สำหรับการแสดงผลผ่านทางส่วนต่อประสานกับผู้ใช้ ตัวอย่างคลาสในแพ็กเกจนี้ เช่น คลาส MainFrame, คลาส MenuBar, คลาส ToolBar, คลาส ToolPanel, คลาส StatusBar, คลาส InputPanel, คลาส OutputPanel และพวกไดอะล็อกคลาสต่าง ๆ เช่น คลาส TruthTableDialog, คลาส PropertiesDialog, ไดอะล็อกคลาสสำหรับการออกแบบสร้างวงจรด้วยสมการบูลีน การออกแบบสร้างอุปกรณ์ UDP และการออกแบบสร้างอุปกรณ์ UDE เป็นต้น

4) แพ็กเกจ ocasim.util เป็นแพ็กเกจสำหรับเก็บชุดของยูทิลิตี้คลาสของระบบ เช่น การวาดรูปอุปกรณ์ตามตำแหน่งต่าง ๆ การดึงค่าตำแหน่งของแต่ละอุปกรณ์เกตพื้นฐาน การเก็บข้อมูลการทำงานสำหรับการจำลองการทำงานแบบทีละขั้น การทำงานที่เป็นลักษณะกระทำซ้ำ ๆ กัน เช่น การวาดรูปสี่เหลี่ยมแทนวงจรย่อยต่าง ๆ การเคลียร์ค่าบัฟเฟอร์ของอุปกรณ์เกตดีเลย์ ตัวอย่างคลาสในแพ็กเกจนี้ เช่น คลาส Operation, คลาส Queue, คลาส StaticUtil เป็นต้น

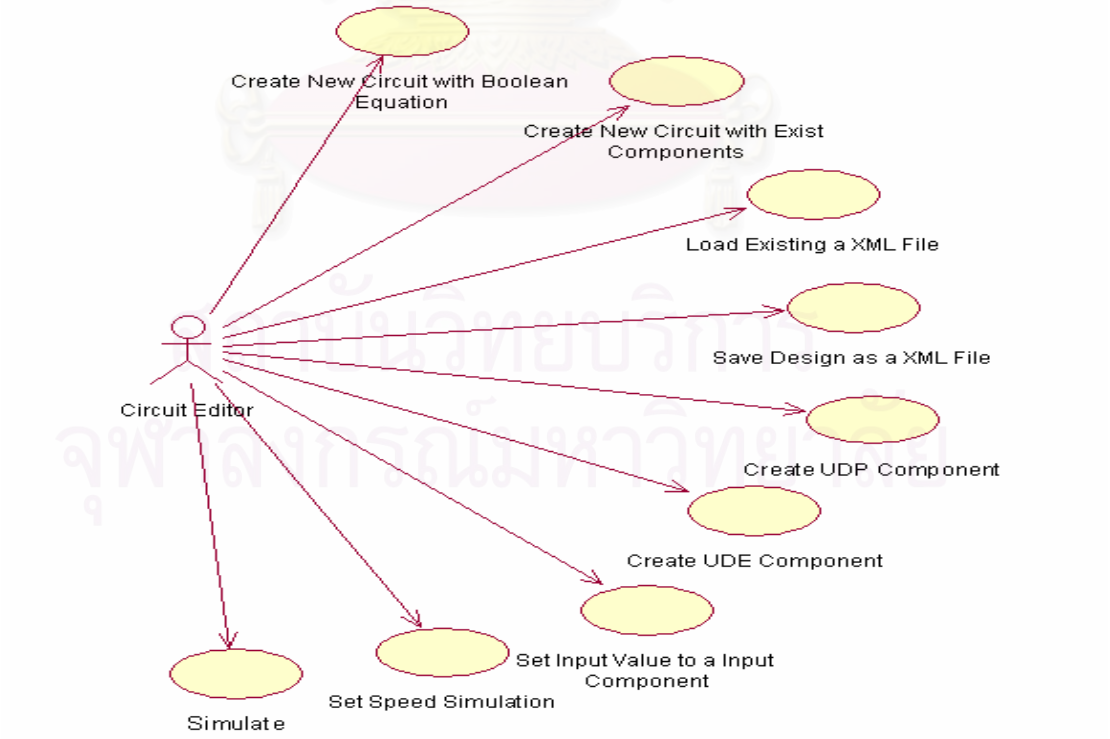
5) แพ็กเกจ ocasim.xml เป็นแพ็กเกจสำหรับเก็บชุดของคลาสที่ทำการอ่านและเขียนข้อมูลในรูปแบบของแฟ้มเอกสารเอ็กซ์เอ็มแอล ตัวอย่างคลาสในแพ็กเกจนี้ เช่น คลาส Document, คลาส Element, คลาส ParseReader, คลาส ParseWriter เป็นต้น

6) แพ็กเกจ ocasim.expr เป็นแพ็กเกจสำหรับเก็บชุดของคลาสต่าง ๆ ที่ใช้ในการออกแบบสร้างอุปกรณ์ที่กำหนดด้วยสมการทางคณิตศาสตร์ ตัวอย่างคลาสในแพ็กเกจนี้ เช่น คลาส LogicExprBuilder, คลาส ExpressionTruthTableBuilder, คลาส LogicExpression, คลาส LogicExprEqual เป็นต้น

## 4.2 แผนภาพอธิบายการออกแบบเครื่องมือ

### 4.2.1 แผนภาพยูสเคสรวมของระบบ

แผนภาพยูสเคสเป็นแผนภาพที่ใช้แสดงหน้าที่ต่าง ๆ ของระบบ ซึ่งสามารถแสดงให้ทราบว่าผู้ใช้งานระบบสามารถใช้งานส่วนใดของระบบได้บ้าง แผนภาพยูสเคสในรูปที่ 4.2 แสดงหน้าที่การทำงานของเครื่องมือสำหรับการออกแบบสร้างอุปกรณ์หรือวงจรและการทดสอบจำลองการทำงานตามอัลกอริทึมการคำนวณแบบเชื่อมตรง




รูปที่ 4.2 แผนภาพยูสเคสรวมของระบบ

จากแผนภาพยูสเคสรวมของระบบในรูปที่ 4.2 แสดงหน้าที่ทั้งหมดของเครื่องมือ โดยมีรายละเอียดดังนี้

1) Create New Circuit with Boolean Equation

เครื่องมือจะต้องสามารถออกแบบสร้างวงจรใหม่ด้วยสมการบูลีนได้

2) Create New Circuit with Exist Components

เครื่องมือจะต้องเตรียมอุปกรณ์พื้นฐานต่าง ๆ สำหรับการเลือกมาออกแบบสร้างวงจรใหม่ด้วยการวาดเป็นผังวงจรได้

3) Load Existing a XML File

เครื่องมือจะต้องสามารถเรียกไฟล์ข้อมูลของอุปกรณ์หรือวงจรต่าง ๆ ที่เคยบันทึกเก็บไว้ในรูปแบบของแฟ้มเอกสารเอ็กซ์เอ็มแอลขึ้นมาเพื่อทำการแก้ไขหรือทดสอบจำลองการทำงานได้

4) Save Design as a XML File

เครื่องมือจะต้องสามารถเก็บบันทึกอุปกรณ์หรือวงจรที่ออกแบบลงไฟล์ในรูปแบบของแฟ้มเอกสารเอ็กซ์เอ็มแอลได้

5) Create UDP Component

เครื่องมือจะต้องสามารถออกแบบสร้างอุปกรณ์ UDP ซึ่งเป็นอุปกรณ์ที่ผู้ออกแบบสามารถกำหนดค่าความหน่วงและค่าผลลัพธ์ในตารางค่าความจริงได้เอง ขึ้นมาใหม่ได้

6) Create UDE Component

เครื่องมือจะต้องสามารถออกแบบสร้างอุปกรณ์ UDE ซึ่งเป็นอุปกรณ์ที่สร้างจากสมการทางคณิตศาสตร์ ขึ้นมาใหม่ได้

7) Set Input Value to a Input Component

เครื่องมือจะต้องสามารถเลือกกำหนดค่าตัวเลขให้กับตัวแปรนำเข้าชนิดต่าง ๆ ได้ เพื่อการทดสอบจำลองการทำงาน

8) Set Speed Simulation

เครื่องมือจะต้องสามารถเลือกกำหนดระดับความเร็วในการจำลองการทำงานได้

9) Simulate

เครื่องมือจะต้องสามารถเลือกจำลองการทำงานได้ 2 รูปแบบ คือ เลือกให้การแสดงผลลัพธ์เกิดขึ้นอย่างต่อเนื่องจนแล้วเสร็จหรือเกิดขึ้นทีละขั้น โดยผู้ออกแบบเป็นผู้ควบคุมการทำงานผ่านทางส่วนต่อประสานกับผู้ใช้

4.2.1.1 แผนภาพยูสเคสของการออกแบบสร้างวงจรใหม่ด้วยสมการบูลีน (Create New Circuit with Boolean Equation) แสดงดังรูปที่ 4.3

รูปที่ 4.3 ยูสเคสของการออกแบบสร้างวงจรใหม่ด้วยสมการบูลีน

สำหรับรายละเอียดของยูสเคสแสดงดังตารางที่ 4.1

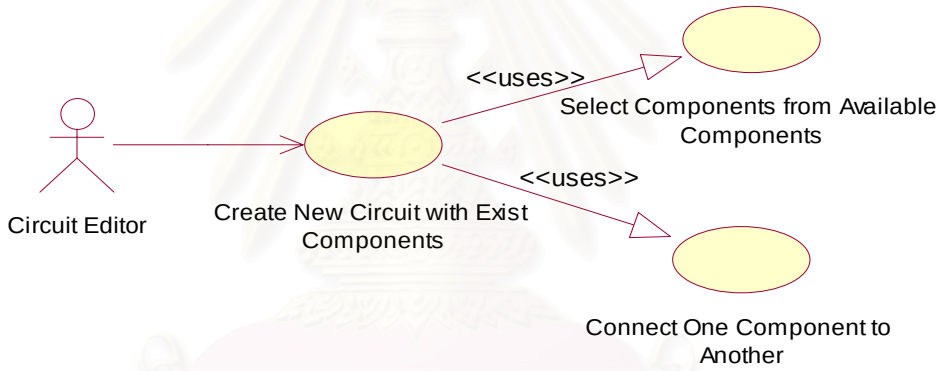
ตารางที่ 4.1 รายละเอียดยูสเคสการออกแบบสร้างวงจรใหม่ด้วยสมการบูลีน

| | |
|---|---|
| **ยูสเคส** | Create New Circuit with Boolean Equation |
| **แอคเตอร์** | ผู้ออกแบบ (Circuit Editor) |
| **เป้าหมาย** | ผู้ออกแบบสามารถออกแบบสร้างวงจรขึ้นใหม่ โดยทำการออกแบบด้วยการกำหนดสมการบูลีน |
| **ยูสเคสที่สัมพันธ์** | - |
| **เงื่อนไขก่อนหน้า** | ผู้ออกแบบเลือกว่าจะออกแบบสร้างวงจรด้วยการกำหนดสมการบูลีนหรือการวาดเป็นผังวงจร |
| **ขั้นตอน** | 1. ผู้ออกแบบเลือกฟังก์ชันการ New file ด้วยการเลือกจากแถบเมนูหรือแถบเครื่องมือ<br>2. เครื่องมือถามยืนยันการเลือกฟังก์ชันการ New file เพื่อทำการเคลียร์อุปกรณ์ต่าง ๆ ออกจากส่วนวาดวงจร |

ตารางที่ 4.1 รายละเอียดยูสเคสการออกแบบสร้างวงจรใหม่ด้วยสมการบูลีน (ต่อ)

| | |
|---|---|
| **ขั้นตอน** | 3. ผู้ออกแบบเลือกเมนูตัวเลือกเพื่อเข้าสู่เมนูย่อยการกำหนดสมการบูลีน<br>4. เครื่องมือทำการเปิดหน้าจอสำหรับกำหนดสมการบูลีน<br>5. ผู้ออกแบบกำหนดสมการบูลีนที่ต้องการลงบนหน้าจอ<br>6. เครื่องมือวาดรูปแทนสมการบูลีนลงบนส่วนวาดวงจร |
| **เงื่อนไขภายหลัง** | เครื่องมือบันทึกวงจรที่ออกแบบด้วยสมการบูลีนเก็บลงไฟล์ในรูปแบบแฟ้มเอกสารเอ็กซ์เอ็มแอล |

4.2.1.2 แผนภาพยูสเคสของการออกแบบสร้างวงจรใหม่ด้วยการวาดเป็นผังวงจร (Create New Circuit with Exist Component) แสดงดังรูปที่ 4.4




รูปที่ 4.4 ยูสเคสของการออกแบบสร้างวงจรใหม่ด้วยการวาดเป็นผังวงจร

สำหรับรายละเอียดของแต่ละยูสเคสแสดงดังตารางที่ 4.2 ถึงตารางที่ 4.4

ตารางที่ 4.2 รายละเอียดยูสเคสการออกแบบสร้างวงจรใหม่ด้วยการวาดเป็นผังวงจร

| | |
|---|---|
| **ยูสเคส** | Create New Circuit with Exist Components |
| **แอคเตอร์** | ผู้ออกแบบ (Circuit Editor) |
| **เป้าหมาย** | ผู้ออกแบบสามารถออกแบบสร้างวงจรใหม่ โดยทำการออกแบบด้วยการนำอุปกรณ์ต่าง ๆ ที่มีอยู่มาวาดประกอบเข้าด้วยกันเป็นผังวงจร |
| **ยูสเคสที่สัมพันธ์** | Uses:<br>■ Select Component from Available Component<br>■ Connect One Component to Another |

ตารางที่ 4.2 รายละเอียดยูสเคสการออกแบบสร้างวงจรใหม่ด้วยการวาดเป็นผังวงจร (ต่อ)

| | |
|---|---|
| **เงื่อนไขก่อนหน้า** | ผู้ออกแบบเลือกว่าจะออกแบบวงจรด้วยการกำหนดสมการบูลีนหรือการวาดเป็นผังวงจร |
| **ขั้นตอน** | 1. ผู้ออกแบบเลือกฟังก์ชันการ New file ด้วยการเลือกจากแถบเมนูหรือแถบเครื่องมือ<br>2. เครื่องมือถามยืนยันการเลือกฟังก์ชันการ New file เพื่อทำการเคลียร์อุปกรณ์ต่าง ๆ ออกจากส่วนวาดวงจร<br>3. ผู้ออกแบบเลือกอุปกรณ์ที่มีอยู่บนคลังอุปกรณ์ทั้งอุปกรณ์พื้นฐานและอุปกรณ์ประกอบมาวางลงบนส่วนวาดวงจร<br>4. ผู้ออกแบบทำการลากเส้นเชื่อมต่ออุปกรณ์ต่าง ๆ เหล่านั้นเข้าด้วยกัน |
| **เงื่อนไขภายหลัง** | เครื่องมือบันทึกวงจรที่ออกแบบด้วยการนำอุปกรณ์ต่าง ๆ ที่มีมาประกอบเข้าด้วยกันวาดเป็นผังวงจรเก็บลงไฟล์ในรูปแบบแฟ้มเอกสารเอ็กซ์เอ็มแอล |

ตารางที่ 4.3 รายละเอียดยูสเคสการเชื่อมต่ออุปกรณ์ต่าง ๆ เข้าด้วยกันบนส่วนวาดวงจร

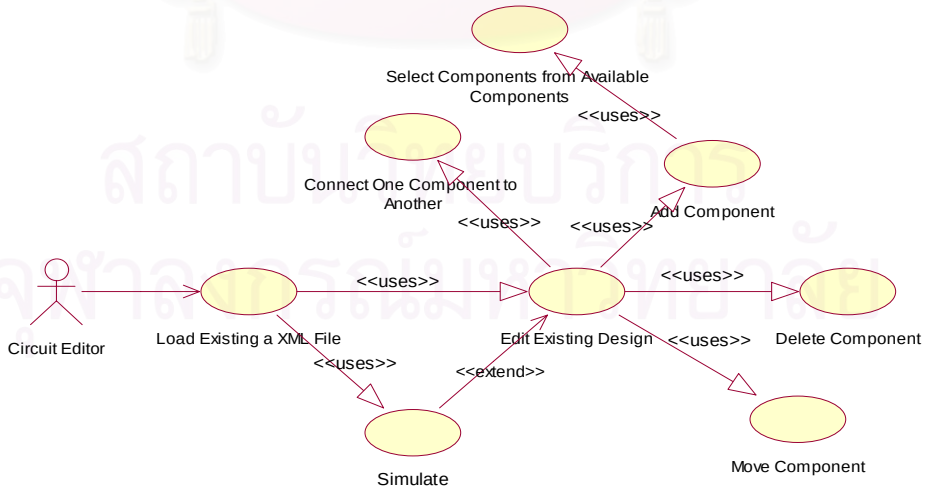
| | |
|---|---|
| **ยูสเคส** | Connect One Component to Another |
| **แอคเตอร์** | ผู้ออกแบบ (Circuit Editor) |
| **เป้าหมาย** | ผู้ออกแบบสามารถลากเส้นเชื่อมต่ออุปกรณ์ต่างที่มีบนส่วนวาดวงจรเข้าด้วยกันได้ตามต้องการ |
| **ยูสเคสที่สัมพันธ์** | Uses:<br>■ Create New Circuit with Exist Component |
| **เงื่อนไขก่อนหน้า** | ผู้ทดสอบผ่านการเลือกอุปกรณ์ต่าง ๆ จากคลังอุปกรณ์มาวางลงบนส่วนวาดวงจร |
| **ขั้นตอน** | ผู้ออกแบบทำการเชื่อมต่ออุปกรณ์ต่าง ๆ ที่มีอยู่บนส่วนวาดวงจรเข้าด้วยกันด้วยการลากเส้นเชื่อมจากขาของอุปกรณ์ตัวหนึ่งไปยังขาของอุปกรณ์อีกตัวหนึ่ง |
| **เงื่อนไขภายหลัง** | อุปกรณ์ที่ถูกนำมาใช้ประกอบเข้าด้วยกันในการออกแบบสร้างวงจรถูกลากเส้นเชื่อมต่อกันผ่านทางขาของตัวอุปกรณ์ |

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ตารางที่ 4.4 รายละเอียดยูสเคสการเลือกอุปกรณ์ที่มีอยู่จากคลังอุปกรณ์มาใช้งาน

| | |
|---|---|
| **ยูสเคส** | Select Components from Available Components |
| **แอคเตอร์** | ผู้ออกแบบ (Circuit Editor) |
| **เป้าหมาย** | เพื่อทำการเลือกอุปกรณ์ต่าง ๆ ที่ถูกสร้างเก็บไว้บนส่วนคลังอุปกรณ์มาใช้สำหรับการออกแบบสร้างวงจรด้วยการวาดเป็นผังวงจรหรือใช้สำหรับเพิ่มอุปกรณ์เข้าไปในวงจรที่เรียกคืนกลับขึ้นมาแก้ไข |
| **ยูสเคสที่สัมพันธ์** | Uses:<br>■ Create New Circuit with Exist Component |
| **เงื่อนไขก่อนหน้า** | เครื่องมือสร้างอุปกรณ์พื้นฐานต่าง ๆ เตรียมไว้แล้วบนคลังอุปกรณ์ |
| **ขั้นตอน** | เครื่องมือสร้างอุปกรณ์พื้นฐานต่าง ๆ เตรียมไว้ให้แล้วบนส่วนคลังอุปกรณ์ ประกอบด้วย อุปกรณ์เกตพื้นฐานต่าง ๆ อุปกรณ์ตัวแปรนำเข้าชนิดต่าง ๆ อุปกรณ์ตัวแปรผลลัพธ์ |
| **เงื่อนไขภายหลัง** | ผู้ออกแบบสามารถเลือกอุปกรณ์ที่มีอยู่บนคลังอุปกรณ์ทั้งอุปกรณ์พื้นฐานและอุปกรณ์ประกอบไปใช้ในการออกแบบสร้างวงจรด้วยการประกอบกันเข้าเป็นผังวงจรหรือเพิ่มเติมเข้าไปในวงจรที่ทำการแก้ไข |

4.2.1.3 แผนภาพยูสเคสของการเรียกไฟล์ข้อมูลของอุปกรณ์หรือวงจรที่เคยบันทึกเก็บไว้มาทำงาน (Load Existing a XML File) แสดงดังรูปที่ 4.5




รูปที่ 4.5 ยูสเคสของการเรียกไฟล์ข้อมูลของอุปกรณ์หรือวงจรมาทำงาน

สำหรับรายละเอียดของแต่ละยูสเคสแสดงดังตารางที่ 4.5 ถึงตารางที่ 4.9

ตารางที่ 4.5 รายละเอียดยูสเคสการเรียกไฟล์ข้อมูลของอุปกรณ์หรือวงจรที่เคยออกแบบเก็บไว้มาทำงาน

| | |
|---|---|
| **ยูสเคส** | Load Existing a XML File |
| **แอคเตอร์** | ผู้ออกแบบ (Circuit Editor) |
| **เป้าหมาย** | เพื่อทำการเรียกไฟล์ข้อมูลอุปกรณ์หรือวงจรที่เคยออกแบบเก็บไว้แล้วกลับขึ้นมาทำงานเพื่อทำการแก้ไขหรือทดสอบจำลองการทำงาน |
| **ยูสเคสที่สัมพันธ์** | Uses:<br>■ Edit Existing Design<br>■ Simulate |
| **เงื่อนไขก่อนหน้า** | - |
| **ขั้นตอน** | 1. ผู้ออกแบบเลือกฟังก์ชันการเรียกไฟล์ข้อมูล Load file ด้วยการเลือกจากแถบเมนูหรือแถบเครื่องมือ<br>2. เครื่องมือเปิดหน้าจอระบุอยู่ที่โฟลเดอร์ชื่อว่า circuits เพื่อให้ผู้ออกแบบได้เลือกชื่อไฟล์ข้อมูลอุปกรณ์หรือวงจรที่ต้องการเรียกมาแสดงบนส่วนวาดวงจร<br>3. ผู้ออกแบบทำการเลือกชื่อไฟล์อุปกรณ์หรือวงจรที่ต้องการ |
| **เงื่อนไขภายหลัง** | เครื่องมือแสดงรูปอุปกรณ์หรือวงจรที่ต้องการเรียกมาทำงานบนพื้นที่ส่วนวาดวงจรจากนั้นผู้ออกแบบสามารถเลือกทำการแก้ไขหรือทดสอบจำลองการทำงานอุปกรณ์หรือวงจรดังกล่าวก็ได้ |

ตารางที่ 4.6 รายละเอียดยูสเคสการแก้ไขอุปกรณ์หรือวงจรที่เคยออกแบบเก็บไว้

| | |
|---|---|
| **ยูสเคส** | Edit Existing Design |
| **แอคเตอร์** | ผู้ออกแบบ (Circuit Editor) |
| **เป้าหมาย** | ผู้ออกแบบสามารถทำการเพิ่มอุปกรณ์ ลบอุปกรณ์ ลากอุปกรณ์หรือเชื่อมต่ออุปกรณ์ต่าง ๆ บนส่วนวาดวงจรได้ |
| **ยูสเคสที่สัมพันธ์** | Uses:<br>■ Add Component<br>■ Delete Component<br>■ Move Component |

ตารางที่ 4.6 รายละเอียดยูสเคสการแก้ไขอุปกรณ์หรือวงจรที่เคยออกแบบเก็บไว้ (ต่อ)

| | |
|---|---|
| **ยูสเคสที่สัมพันธ์** | Uses:<br>■ Connect One Component to Another<br>Extends:<br>■ Simulate |
| **เงื่อนไขก่อนหน้า** | ผู้ทดสอบผ่านการเรียกไฟล์ข้อมูลอุปกรณ์หรือวงจรที่เคยออกแบบเก็บไว้มาทำงานเพื่อทำการแก้ไข |
| **ขั้นตอน** | 1. ผู้ออกแบบทำการเลือกเพิ่มอุปกรณ์เข้าไปในวงจรที่เรียกมาแก้ไข หรือเลือกทำการลบอุปกรณ์ออกจากวงจรที่เรียกมาแก้ไข หรือเลือกทำการย้ายตำแหน่งการจัดวางของอุปกรณ์ต่าง ๆ ตลอดจนการเปลี่ยแปลงการเชื่อมต่อของอุปกรณ์ต่าง ๆ จากที่เคยเป็น<br>2. หากการเรียกไฟล์ข้อมูลอุปกรณ์หรือวงจรที่เคยออกแบบเก็บไว้แล้วกลับขึ้นมาเพื่อทดสอบจำลองการทำงาน เครื่องมือจะไม่อนุญาตให้ทำการแก้ไขอุปกรณ์หรือวงจรได้ |
| **เงื่อนไขภายหลัง** | อุปกรณ์หรือวงจรที่ถูกเรียกมาแสดงบนส่วนวาดวงจรถูกแก้ไขด้วยการเพิ่มอุปกรณ์เข้าไปหรือลบอุปกรณ์ที่มีอยู่ออกบางตัวหรือย้ายตำแหน่งการจัดวางอุปกรณ์ต่าง ๆ ตลอดจนการเปลี่ยนแปลงการเชื่อมต่อกันของอุปกรณ์ |

ตารางที่ 4.7 รายละเอียดยูสเคสการเพิ่มอุปกรณ์เข้าไปในวงจรที่เคยออกแบบเก็บไว้

| | |
|---|---|
| **ยูสเคส** | Add Component |
| **แอคเตอร์** | ผู้ออกแบบ (Circuit Editor) |
| **เป้าหมาย** | ผู้ออกแบบสามารถเลือกอุปกรณ์ต่าง ๆ จากคลังอุปกรณ์มาเพิ่มเข้าไปในวงจรที่ถูกเรียกมาเพื่อทำการแก้ไขได้ |
| **ยูสเคสที่สัมพันธ์** | Uses:<br>■ Select Component from Available Component<br>■ Edit Existing Design |
| **เงื่อนไขก่อนหน้า** | ผู้ทดสอบผ่านการเรียกไฟล์ข้อมูลอุปกรณ์หรือวงจรที่เคยออกแบบเก็บไว้มาทำงานเพื่อทำการแก้ไข |

ตารางที่ 4.7 รายละเอียดยูสเคสการเพิ่มอุปกรณ์เข้าไปในวงจรที่เคยออกแบบเก็บไว้ (ต่อ)

| | |
|---|---|
| **ขั้นตอน** | ผู้ออกแบบทำการเลือกอุปกรณ์ต่าง ๆ ที่มีอยู่บนคลังอุปกรณ์เพิ่มเข้าไปในอุปกรณ์หรือวงจรที่ถูกเรียกมาเพื่อทำการแก้ไข |
| **เงื่อนไขภายหลัง** | อุปกรณ์หรือวงจรย่อยต่าง ๆ ถูกเพิ่มเข้าไปในวงจรที่ถูกเรียกมาเพื่อทำการแก้ไข |

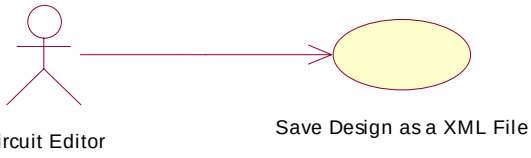
ตารางที่ 4.8 รายละเอียดยูสเคสการลบอุปกรณ์ออกจากวงจรที่เคยออกแบบเก็บไว้

| | |
|---|---|
| **ยูสเคส** | Delete Component |
| **แอคเตอร์** | ผู้ออกแบบ (Circuit Editor) |
| **เป้าหมาย** | ผู้ออกแบบสามารถเลือกลบอุปกรณ์ต่าง ๆ ออกจากวงจรที่ถูกเรียกมาเพื่อทำการแก้ไขได้ |
| **ยูสเคสที่สัมพันธ์** | Uses:<br>■ Edit Existing Design |
| **เงื่อนไขก่อนหน้า** | ผู้ทดสอบผ่านการเรียกไฟล์ข้อมู,อุปกรณ์หรือวงจรที่เคยออกแบบเก็บไว้มาทำงานเพื่อทำการแก้ไข |
| **ขั้นตอน** | ผู้ออกแบบทำการเลือกลบอุปกรณ์ต่าง ๆ ออกจากวงจรที่ถูกเรียกมาเพื่อทำการแก้ไข |
| **เงื่อนไขภายหลัง** | วงจรที่ถูกเรียกมาถูกแก้ไขโดยมีการลบอุปกรณ์บางตัวออก |

ตารางที่ 4.9 รายละเอียดยูสเคสการจัดวางตำแหน่งของอุปกรณ์ใหม่

| | |
|---|---|
| **ยูสเคส** | Move Component |
| **แอคเตอร์** | ผู้ออกแบบ (Circuit Editor) |
| **เป้าหมาย** | ผู้ออกแบบสามารถจัดวางตำแหน่งของอุปกรณ์ได้ใหม่ตามต้องการ |
| **ยูสเคสที่สัมพันธ์** | Uses:<br>■ Edit Existing Design |
| **เงื่อนไขก่อนหน้า** | ผู้ทดสอบผ่านการเรียกไฟล์ข้อมูลอุปกรณ์หรือวงจรที่เคยออกแบบเก็บไว้มาทำงานเพื่อทำการแก้ไข |
| **ขั้นตอน** | ผู้ออกแบบทำการจัดวางตำแหน่งของอุปกรณ์ใหม่ |
| **เงื่อนไขภายหลัง** | วงจรที่ถูกเรียกมาถูกแก้ไขโดยมีการจัดวางตำแหน่งของอุปกรณ์ใหม่ในรูปแบบลากแล้วปล่อย |

4.2.1.4 แผนภาพยูสเคสของการบันทึกอุปกรณ์หรือวงจรที่ออกแบบลงไฟล์ในรูปแบบของแฟ้มเอกสารเอ็กซ์เอ็มแอล (Save Design as a XML File) แสดงดังรูปที่ 4.6




รูปที่ 4.6 ยูสเคสของการบันทึกอุปกรณ์หรือวงจรที่ออกแบบเก็บลงไฟล์

สำหรับรายละเอียดของยูสเคสแสดงดังตารางที่ 4.10

ตารางที่ 4.10 รายละเอียดยูสเคสการบันทึกอุปกรณ์หรือวงจรที่ออกแบบลงไฟล์ในรูปแบบแฟ้มเอกสารเอ็กซ์เอ็มแอล

| | |
|---|---|
| **ยูสเคส** | Save Design as a XML File |
| **แอคเตอร์** | ผู้ออกแบบ (Circuit Editor) |
| **เป้าหมาย** | เพื่อทำการบันทึกอุปกรณ์หรือวงจรที่ออกแบบเก็บลงไฟล์ในรูปแบบของแฟ้มเอกสารเอ็กซ์เอ็มแอล |
| **ยูสเคสที่สัมพันธ์** | - |
| **เงื่อนไขก่อนหน้า** | ผู้ออกแบบผ่านการเลือกออกแบบสร้างอุปกรณ์ UDP หรืออุปกรณ์ UDE หรือสร้างวงจรด้วยการกำหนดสมการบูลีนหรือวาดเป็นผังวงจร |
| **ขั้นตอน** | 1. ผู้ออกแบบเลือกฟังก์ชันการบันทึกข้อมูล Save ด้วยการเลือกจากแถบเมนูหรือแถบเครื่องมือ<br>2. เครื่องมือเปิดหน้าจอสำหรับให้ผู้ออกแบบตั้งชื่ออุปกรณ์หรือวงจรที่ออกแบบเพื่อเก็บลงในโฟลเดอร์ชื่อว่า circuits |
| **เงื่อนไขภายหลัง** | เครื่องมือทำการบันทึกอุปกรณ์หรือวงจรที่ออกแบบเก็บลงไฟล์ในรูปแบบแฟ้มเอกสารเอ็กซ์เอ็มแอล |

4.2.1.5 แผนภาพยูสเคสของการออกแบบสร้างอุปกรณ์ UDP (Create UDP Component) แสดงดังรูปที่ 4.7

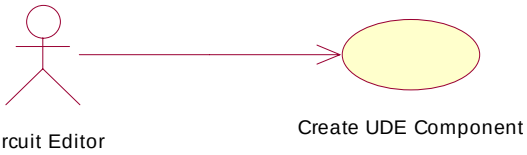
รูปที่ 4.7 ยูสเคสของการออกแบบสร้างอุปกรณ์ UDP

สำหรับรายละเอียดของยูสเคสแสดงดังตารางที่ 4.11

ตารางที่ 4.11 รายละเอียดยูสเคสการออกแบบสร้างอุปกรณ์ UDP

| | |
|---|---|
| **ยูสเคส** | Create UDP Component |
| **แอคเตอร์** | ผู้ออกแบบ (Circuit Editor) |
| **เป้าหมาย** | ผู้ออกแบบสามารถออกแบบสร้างอุปกรณ์ UDP ขึ้นใหม่ได้เอง |
| **ยูสเคสที่สัมพันธ์** | - |
| **เงื่อนไขก่อนหน้า** | ผู้ออกแบบเลือกออกแบบสร้างอุปกรณ์ UDP |
| **ขั้นตอน** | 1. ผู้ออกแบบเลือกฟังก์ชันการ New file ด้วยการเลือกจากแถบเมนูหรือแถบเครื่องมือ<br>2. เครื่องมือถามยืนยันการเลือกฟังก์ชันการ New file เพื่อทำการเคลียร์อุปกรณ์ต่าง ๆ ออกจากส่วนวาดวงจร<br>3. ผู้ออกแบบเลือกเมนูตัวเลือกเพื่อเข้าสู่เมนูย่อยการออกแบบอุปกรณ์ UDP<br>4. เครื่องมือทำการเปิดหน้าจอสำหรับการกำหนดจำนวนตัวแปรนำเข้าและจำนวนค่าความหน่วงให้กับตัวอุปกรณ์เอง<br>5. ผู้ออกแบบทำการป้อนค่าที่ต้องการลงบนหน้าจอที่เครื่องมือเปิดขึ้นมาแล้วกดปุ่ม Next เพื่อทำขั้นตอนต่อไป<br>6. เครื่องมือทำการสร้างหน้าจอสำหรับกำหนดค่าผลลัพธ์ที่ต้องการลงในตารางค่าความจริง<br>7. ผู้ออกแบบทำการป้อนค่าผลลัพธ์ลงในตารางค่าความจริงแล้วกดปุ่มสิ้นสุด (Finish) เพื่อยืนยันการทำงาน<br>8. เครื่องมือวาดรูปสี่เหลี่ยมแทนอุปกรณ์ UDP บนส่วนวาดวงจร |
| **เงื่อนไขภายหลัง** | เครื่องมือบันทึกอุปกรณ์ UDP ที่ออกแบบเก็บลงไฟล์ในส่วนเก็บอุปกรณ์ประกอบในรูปแบบแฟ้มเอกสารเอ็กซ์เอ็มแอล |

4.2.1.6 แผนภาพยูสเคสของการออกแบบสร้างอุปกรณ์ UDE (Create UDE Component) แสดงดังรูปที่ 4.8



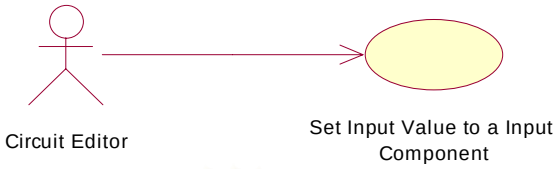

รูปที่ 4.8 ยูสเคสของการออกแบบสร้างอุปกรณ์ UDE

สำหรับรายละเอียดของยูสเคสแสดงดังตารางที่ 4.12

ตารางที่ 4.12 รายละเอียดยูสเคสการออกแบบสร้างอุปกรณ์ UDE

| | |
|---|---|
| **ยูสเคส** | Create UDE Component |
| **แอคเตอร์** | ผู้ออกแบบ (Circuit Editor) |
| **เป้าหมาย** | ผู้ออกแบบสามารถออกแบบสร้างอุปกรณ์ UDE ขึ้นใหม่ได้เอง |
| **ยูสเคสที่สัมพันธ์** | - |
| **เงื่อนไขก่อนหน้า** | ผู้ออกแบบเลือกออกแบบสร้างอุปกรณ์ UDE |
| **ขั้นตอน** | 1. ผู้ออกแบบเลือกฟังก์ชันการ New file ด้วยการเลือกจากแถบเมนูหรือแถบเครื่องมือ<br>2. เครื่องมือถามยืนยันการเลือกฟังก์ชันการ New file เพื่อทำการเคลียร์อุปกรณ์ต่าง ๆ ออกจากส่วนวาดวงจร<br>3. ผู้ออกแบบเลือกเมนูตัวเลือกเพื่อเข้าสู่เมนูย่อยการออกแบบอุปกรณ์ UDE<br>4. เครื่องมือทำการเปิดหน้าจอสำหรับการกำหนดสมการทางคณิตศาสตร์<br>5. เครื่องมือทำการสร้างตารางค่าความจริงให้อัตโนมัติ หากสมการทางคณิตศาสตร์นั้นถูกต้อง<br>6. เครื่องมือวาดรูปสี่เหลี่ยมแทนอุปกรณ์ UDE บนส่วนวาดวงจร |
| **เงื่อนไขภายหลัง** | เครื่องมือบันทึกอุปกรณ์ UDE ที่ออกแบบเก็บลงไฟล์ในส่วนเก็บอุปกรณ์ประกอบในรูปแบบแฟ้มเอกสารเอ็กซ์เอ็มแอล |

4.2.1.7 แผนภาพยูสเคสของการเลือกกำหนดค่าตัวเลขให้กับตัวแปรนำเข้าชนิดต่าง ๆ (Set Input Value to a Input Component) แสดงดังรูปที่ 4.9



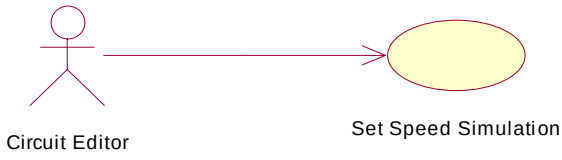

รูปที่ 4.9 ยูสเคสของการเลือกกำหนดค่าตัวเลขให้กับตัวแปรนำเข้าชนิดต่าง ๆ

สำหรับรายละเอียดของยูสเคสแสดงดังตารางที่ 4.13

ตารางที่ 4.13 รายละเอียดยูสเคสการเลือกกำหนดค่าตัวเลขให้กับตัวแปรนำเข้าชนิดต่าง ๆ

| | |
|---|---|
| **ยูสเคส** | Set Input Value to a Input Component |
| **แอคเตอร์** | ผู้ออกแบบ (Circuit Editor) |
| **เป้าหมาย** | ผู้ออกแบบสามารถทำการป้อนค่าตัวแปรนำเข้าทุกชนิดเพื่อทดสอบจำลองการทำงาน |
| **ยูสเคสที่สัมพันธ์** | - |
| **เงื่อนไขก่อนหน้า** | ผู้ทดสอบผ่านการเรียกไฟล์ข้อมูลอุปกรณ์หรือวงจรที่เคยออกแบบเก็บไว้มาทำงานเพื่อทำการทดสอบจำลองการทำงาน |
| **ขั้นตอน** | ผู้ออกแบบป้อนค่าตัวแปรนำเข้าทั้งหมดในช่องป้อนค่าตัวเลขฐานสิบหรือช่องป้อนค่าตัวเลขฐานสองเพื่อนำไปใช้สำหรับจำลองการทำงาน |
| **เงื่อนไขภายหลัง** | ค่าตัวเลขที่ผู้ออกแบบทำการป้อนค่านั้นในช่องป้อนค่าตัวเลขฐานสองถูกนำไปใช้ในการทดสอบจำลองการทำงาน |

4.2.1.8 แผนภาพยูสเคสของการเลือกกำหนดระดับความเร็วในการจำลองการทำงาน (Set Speed Simulation) แสดงดังรูปที่ 4.10



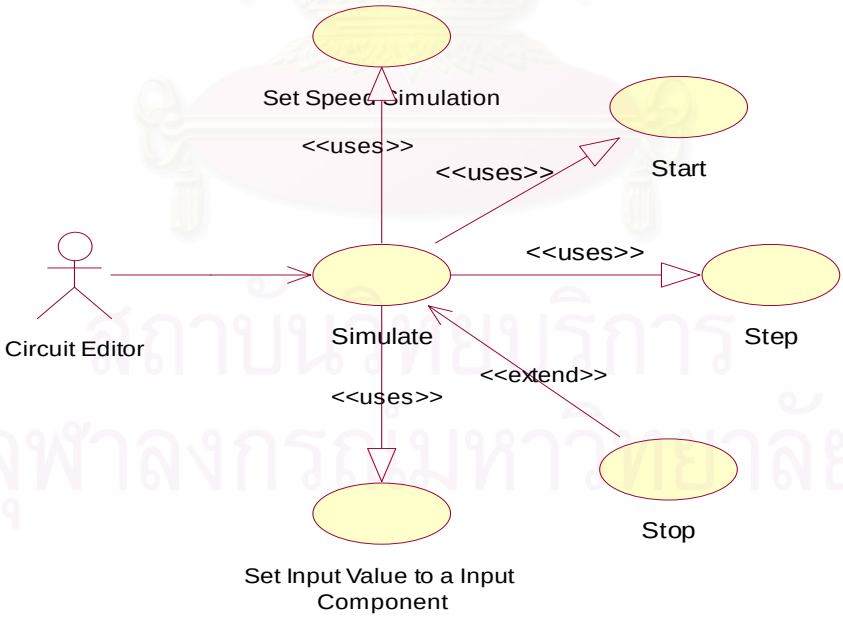

รูปที่ 4.10 ยูสเคสของการเลือกกำหนดระดับความเร็วในการจำลองการทำงาน

สำหรับรายละเอียดของยูสเคสแสดงดังตารางที่ 4.14

ตารางที่ 4.14 รายละเอียดยูสเคสการเลือกระดับความเร็วในการแสดงผลจำลองการทำงาน

| | |
|---|---|
| **ยูสเคส** | Set Speed Simulation |
| **แอคเตอร์** | ผู้ออกแบบ (Circuit Editor) |
| **เป้าหมาย** | ผู้ออกแบบสามารถทำการเลือกระดับความเร็วที่ต้องการในการแสดงผลการจำลองการทำงานในแต่ละครั้งได้ |
| **ยูสเคสที่สัมพันธ์** | - |
| **เงื่อนไขก่อนหน้า** | ผู้ทดสอบผ่านการเรียกไฟล์ข้อมูลอุปกรณ์หรือวงจรที่เคยออกแบบเก็บไว้มาทำงานเพื่อทำการทดสอบจำลองการทำงาน |
| **ขั้นตอน** | ผู้ออกแบบเลือกกำหนดระดับความเร็วที่ต้องการจากแถบเครื่องมือด้านบนซึ่งเป็นแถบเครื่องมือของส่วนจำลองการทำงาน |
| **เงื่อนไขภายหลัง** | เครื่องมือจำลองการทำงานตามระดับความเร็วที่ผู้ออกแบบกำหนดเอง |

4.2.1.9 แผนภาพยูสเคสของการแสดงผลการจำลองการทำงาน (Simulate) แสดงดังรูปที่ 4.11




รูปที่ 4.11 ยูสเคสของการแสดงผลการจำลองการทำงาน

สำหรับรายละเอียดของแต่ละยูสเคสแสดงดังตารางที่ 4.15 ถึงตารางที่ 4.18

ตารางที่ 4.15 รายละเอียดยูสเคสการแสดงผลการจำลองการทำงาน

| | |
|---|---|
| **ยูสเคส** | Simulate |
| **แอคเตอร์** | ผู้ออกแบบ (Circuit Editor) |
| **เป้าหมาย** | ผู้ออกแบบสามารถทำการทดสอบจำลองการทำงานอุปกรณ์หรือวงจรที่ออกแบบไว้ได้ โดยสามารถเลือกแสดงผลได้ 2 รูปแบบ |
| **ยูสเคสที่สัมพันธ์** | Uses:<br>■ Set Speed Simulation<br>■ Set Input Value to a Input Component<br>■ Start<br>■ Step<br>Extend:<br>■ Stop |
| **เงื่อนไขก่อนหน้า** | ผู้ทดสอบผ่านการกำหนดค่าตัวเลขให้กับตัวแปรนำเข้าทุกตัว |
| **ขั้นตอน** | 1. ผู้ออกแบบทำการป้อนค่าตัวเลขในช่องป้อนค่าตัวเลขฐานสิบหรือช่องป้อนค่าตัวเลขฐานสอง<br>2. ผู้ออกแบบเลือกทดสอบจำลองการทำงานด้วยการเลือกกดปุ่มสตาร์ทหรือปุ่มเสต็พบนแถบเครื่องมือ |
| **เงื่อนไขภายหลัง** | เครื่องมือแสดงผลการจำลองการทำงานตามที่ผู้ออกแบบเลือก |

ตารางที่ 4.16 รายละเอียดยูสเคสเลือกจำลองการทำงานให้เกิดขึ้นต่อเนื่องจนแล้วเสร็จ

| | |
|---|---|
| **ยูสเคส** | Start |
| **แอคเตอร์** | ผู้ออกแบบ (Circuit Editor) |
| **เป้าหมาย** | ผู้ออกแบบสามารถเลือกให้การจำลองการทำงานเกิดขึ้นอย่างต่อเนื่องจนแล้วเสร็จได้ |
| **ยูสเคสที่สัมพันธ์** | Uses:<br>■ Simulate |
| **เงื่อนไขก่อนหน้า** | ผู้ทดสอบผ่านการกำหนดค่าตัวเลขให้กับตัวแปรนำเข้าทุกตัว |
| **ขั้นตอน** | ผู้ออกแบบเลือกกดปุ่มสตาร์ทบนแถบเครื่องมือหรือเลือกจากแถบเมนูก็ได้เพื่อทดสอบจำลองการทำงานอย่างต่อเนื่องจนแล้วเสร็จ |
| **เงื่อนไขภายหลัง** | เครื่องมือจำลองการทำงานได้อย่างต่อเนื่องจนแล้วเสร็จ |

ตารางที่ 4.17 รายละเอียดยูสเคสเลือกจำลองการทำงานให้เกิดขึ้นทีละขั้น

| | |
|---|---|
| **ยูสเคส** | Step |
| **แอคเตอร์** | ผู้ออกแบบ (Circuit Editor) |
| **เป้าหมาย** | ผู้ออกแบบสามารถเลือกให้การจำลองการทำงานเกิดขึ้นทีละขั้นได้ |
| **ยูสเคสที่สัมพันธ์** | Uses:<br>■ Simulate |
| **เงื่อนไขก่อนหน้า** | ผู้ทดสอบผ่านการกำหนดค่าตัวเลขให้กับตัวแปรนำเข้าทุกตัว |
| **ขั้นตอน** | ผู้ออกแบบเลือกกดปุ่มสเต็พบนแถบเครื่องมือหรือเลือกจากแถบเมนูก็ได้เพื่อทดสอบจำลองการทำงานเกิดขึ้นทีละขั้น |
| **เงื่อนไขภายหลัง** | เครื่องมือจำลองการทำงานให้เกิดขึ้นทีละขั้นได้ |

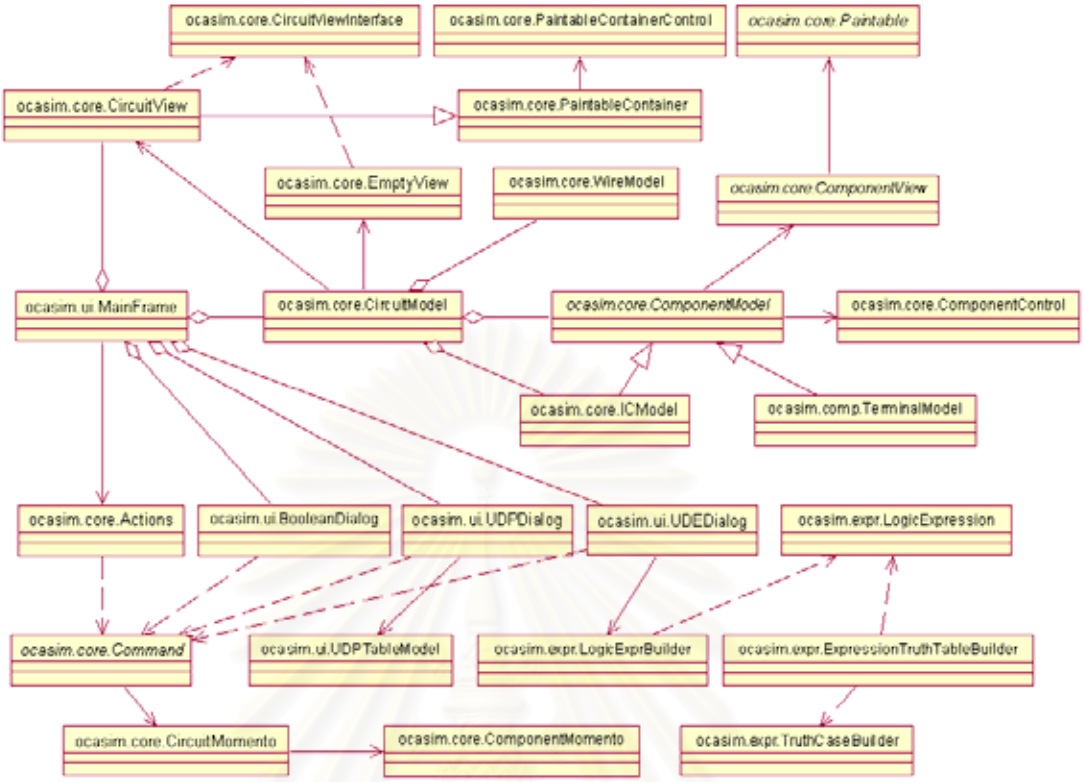
ตารางที่ 4.18 รายละเอียดยูสเคสเลือกยกเลิกการจำลองการทำงาน

| | |
|---|---|
| **ยูสเคส** | Stop |
| **แอคเตอร์** | ผู้ออกแบบ (Circuit Editor) |
| **เป้าหมาย** | ผู้ออกแบบสามารถยกเลิกการจำลองการทำงานได้ขณะจำลองการทำงาน |
| **ยูสเคสที่สัมพันธ์** | Extend:<br>■ Simulate |
| **เงื่อนไขก่อนหน้า** | ผู้ทดสอบผ่านการกำหนดค่าตัวเลขให้กับตัวแปรนำเข้าทุกตัว |
| **ขั้นตอน** | ผู้ออกแบบเลือกกดปุ่มสต็อพบนแถบเครื่องมือหรือเลือกจากแถบเมนูเพื่อยกเลิกการจำลองการทำงาน |
| **เงื่อนไขภายหลัง** | เครื่องมือทำการยกเลิกการจำลองการทำงาน |

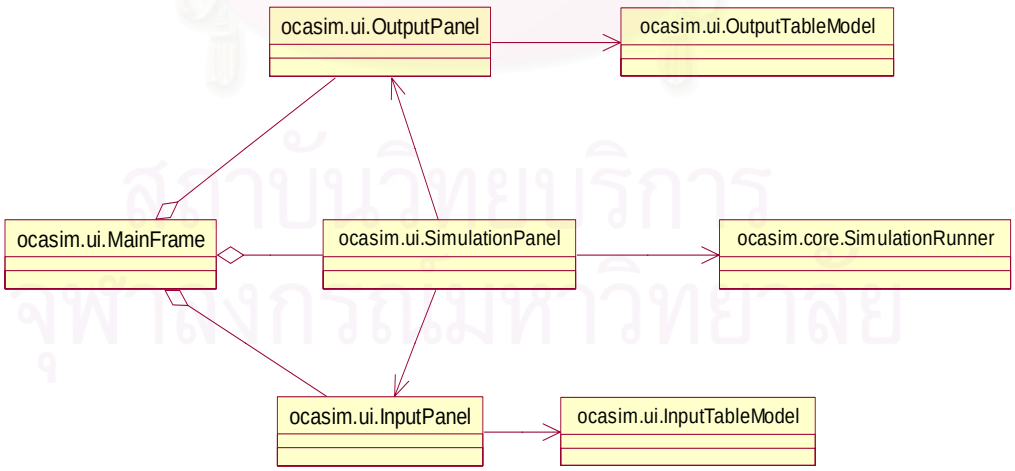
ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

4.2.2 แผนภาพคลาส

แผนภาพคลาสคือ แผนภาพที่ใช้แสดงโครงสร้างความสัมพันธ์ของแต่ละคลาสในระบบ แผนภาพคลาสในหัวข้อนี้เป็นแผนภาพคลาสที่ได้จากการออกแบบเครื่องมือสำหรับส่วนการออกแบบและส่วนการจำลองการทำงาน โดยมีรายละเอียดแสดงดังรูปที่ 4.12 และรูปที่ 4.13 ตามลำดับ



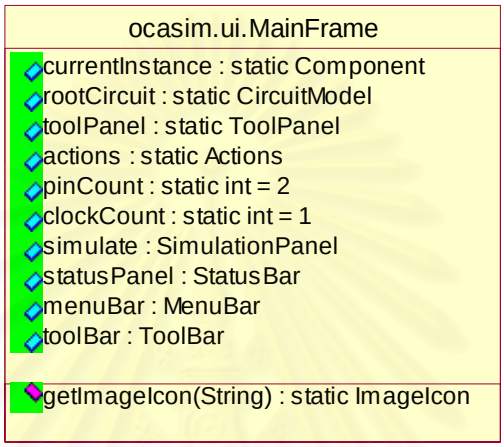

รูปที่ 4.12 แผนภาพคลาสของเครื่องมือจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรงสำหรับส่วนการออกแบบ




รูปที่ 4.13 แผนภาพคลาสของเครื่องมือจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรงสำหรับส่วนการจำลองการทำงาน

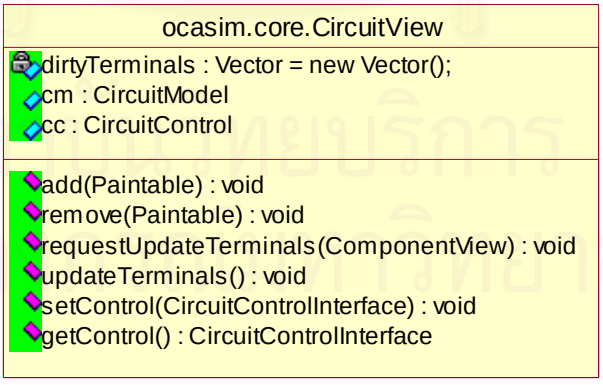
รายละเอียดของคลาสต่าง ๆ อธิบายได้ดังนี้

1) คลาส ocasim.ui.MainFrame คือ คลาสหลักที่ออกแบบสำหรับแสดงส่วนต่อประสานกับผู้ใช้ โดยจะแสดงส่วนประกอบต่าง ๆ ของเครื่องมือเพื่อให้ผู้ออกแบบสามารถเลือกใช้งานได้ตามสะดวก รายละเอียดของคลาสแสดงดังรูปที่ 4.14



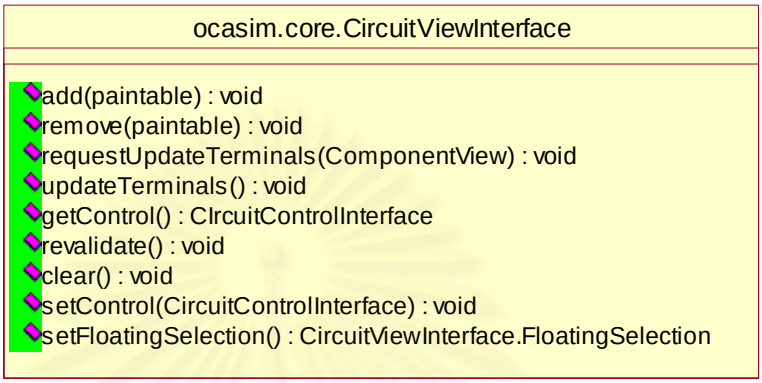

รูปที่ 4.14 คลาส ocasim.ui.MainFrame

2) คลาส ocasim.core.CircuitView คือ คลาสที่ออกแบบเพื่อทำหน้าที่สำหรับการแสดงรูปของวงจรที่สร้างขึ้นโดยผู้ออกแบบสามารถทำการเพิ่ม ลบหรือแก้ไขอุปกรณ์เกตพื้นฐานต่าง ๆ ที่ประกอบกันในวงจรที่ออกแบบได้ รายละเอียดของคลาสแสดงดังรูปที่ 4.15



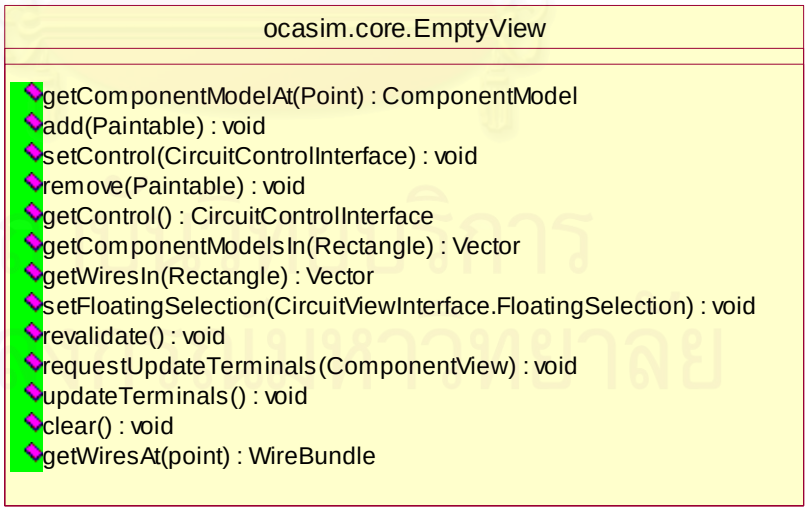

รูปที่ 4.15 คลาส ocasim.core.CircuitView

3) คลาส ocasim.core.CircuitViewInterface คือ คลาสที่ออกแบบให้เป็นอินเทอร์เฟสคลาสเพื่อทำหน้าที่กำหนดเมทธอดต่าง ๆ ที่สามารถทำงานได้กับคลาส ocasim.core.CircuitView รายละเอียดของคลาสแสดงดังรูปที่ 4.16



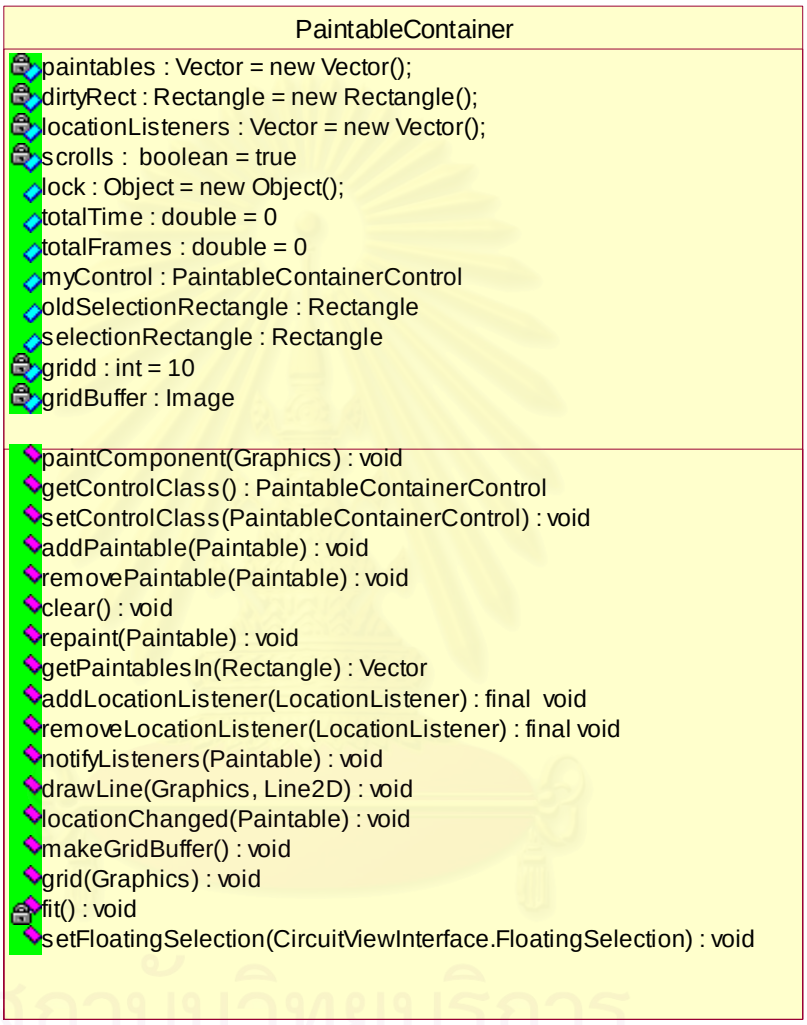

รูปที่ 4.16 คลาส ocasim.core.CircuitViewInterface

4) คลาส ocasim.core.EmptyView คือ คลาสที่ออกแบบให้ทำหน้าที่สำหรับการกำหนดค่าเริ่มต้นของส่วนวาดวงจร ซึ่งจะกำหนดให้เป็นค่าเริ่มต้น เป็นการเคลียร์หน้าจอ เช่น Popup หน้าจอของการกดเลือกเมนู View internal ที่ตัวอุปกรณ์ เมื่อทำการปิด Popup หน้าจอจะทำการเคลียร์โดยใช้คลาสนี้ รายละเอียดของคลาสแสดงดังรูปที่ 4.17



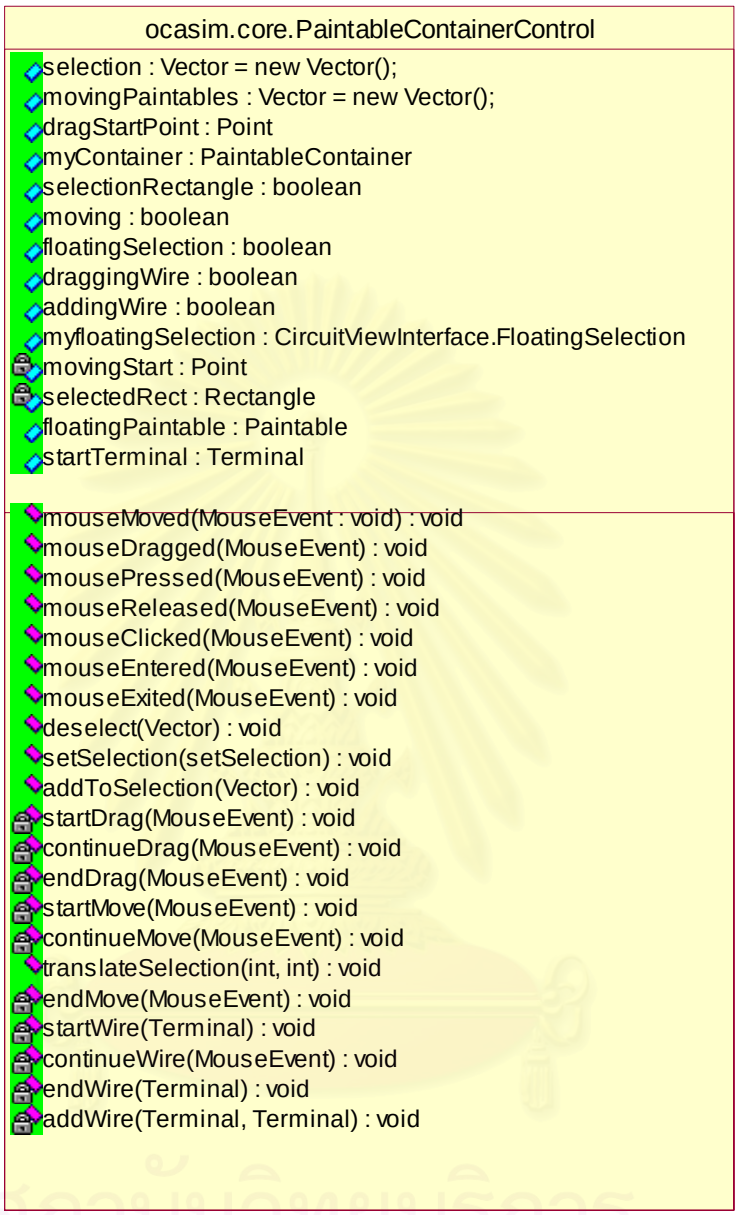

รูปที่ 4.17 คลาส ocasim.core.EmptyView

5) คลาส ocasim.core.PaintableContainer คือ คลาสที่ออกแบบให้ทำหน้าเป็นเหมือนกระดาษสำหรับวาดรูป จะมีเมทธอดที่เกี่ยวข้องกับการวาดรูปหรือการลบรูปที่ทำการวาดลงไป รายละเอียดของคลาสแสดงดังรูปที่ 4.18



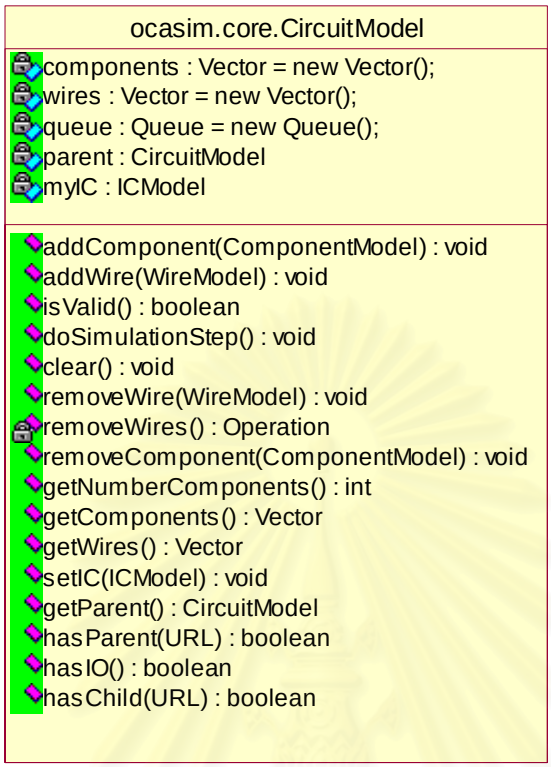

รูปที่ 4.18 คลาส ocasim.core.PaintableContainer

6) คลาส ocasim.core.PaintableContainerControl คือ คลาสที่ออกแบบให้ทำหน้าที่ควบคุมและจัดการเกี่ยวกับ Mouse Event ที่กระทำกับอุปกรณ์ต่าง ๆ เช่น การกด (Click) หรือการลากแล้วปล่อย (Drag and Drop) เป็นต้น รายละเอียดของคลาสแสดงดังรูปที่ 4.19




รูปที่ 4.19 คลาส ocasim.core.PaintableContainerControl

7) คลาส ocasim.core.CircuitModel คือ คลาสที่ออกแบบสำหรับการจัดการกับวงจรที่ออกแบบสร้างขึ้น เช่น การเพิ่มอุปกรณ์ต่าง ๆ ลงในวงจรที่สร้าง การลากเส้นเชื่อมต่ออุปกรณ์ต่าง ๆ เข้าด้วยกัน ตลอดจนการลบอุปกรณ์ออกจากส่วนวาดวงจรหรือลบเส้นเชื่อมต่อระหว่างอุปกรณ์ รายละเอียดของคลาสแสดงดังรูปที่ 4.20

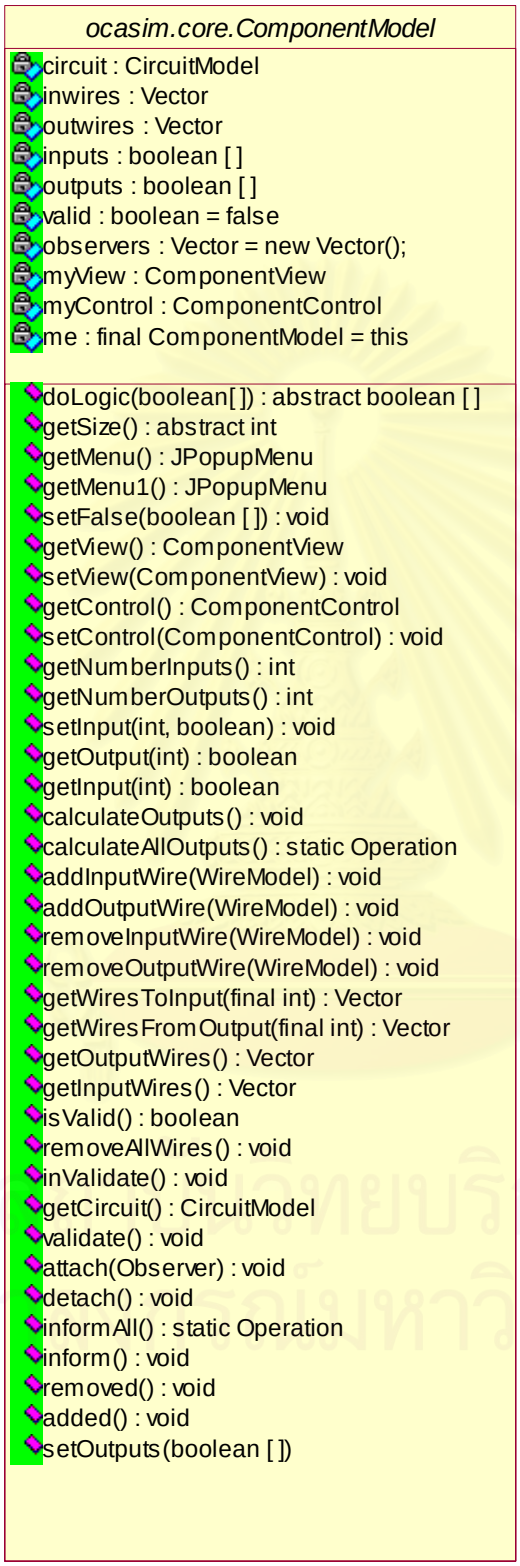



รูปที่ 4.20 คลาส ocasim.core.CircuitModel

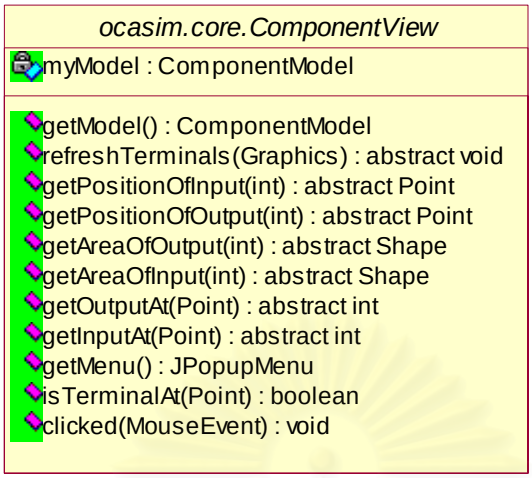
8) คลาส ocasim.core.ComponentModel คือ คลาสที่ออกแบบเป็นคลาสนามธรรม (Abstract Class) สำหรับเป็นคลาสต้นแบบของอุปกรณ์ต่าง ๆ ทั้งอุปกรณ์พื้นฐานและอุปกรณ์ประกอบ โดยทำการออกแบบให้มีเมนูผูกติดอยู่กับตัวอุปกรณ์ต่าง ๆ เพื่อสามารถกระทำกับตัวอุปกรณ์ต่าง ๆ เหล่านั้นได้ รายละเอียดของคลาสแสดงดังรูปที่ 4.21

9) คลาส ocasim.core.ComponentView คือ คลาสที่ออกแบบเป็นคลาสนามธรรมสำหรับการเป็นคลาสต้นแบบของการแสดงผลอุปกรณ์ต่าง ๆ โดยมีฟังก์ชันการทำงานที่เกี่ยวกับการ Refresh ส่วนที่ทำการวาดอุปกรณ์ การที่มีเมนูผูกติดอยู่กับตัวอุปกรณ์ รายละเอียดของคลาสแสดงดังรูปที่ 4.22

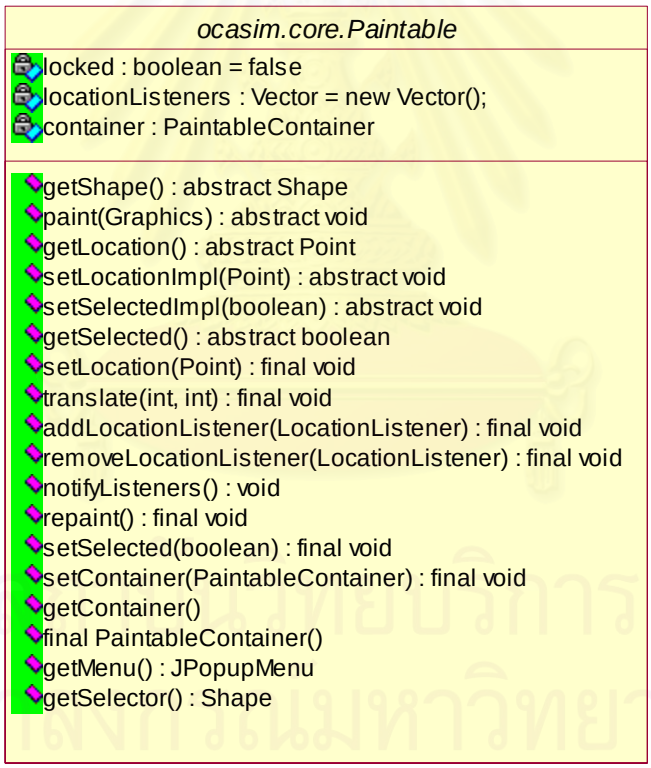
10) คลาส ocasim.core.Paintable คือ คลาสที่ออกแบบเป็นคลาสนามธรรมสำหรับการวาดรูปต่าง ๆ ของวงจร หรือเป็น Template class นั่นเอง รายละเอียดของคลาสแสดงดังรูปที่ 4.23




รูปที่ 4.21 คลาส ocasim.core.ComponentModel

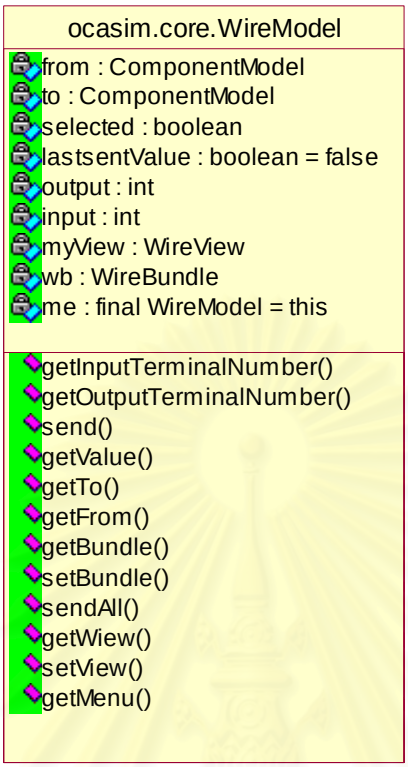



รูปที่ 4.22 คลาส ocasim.core.ComponentView



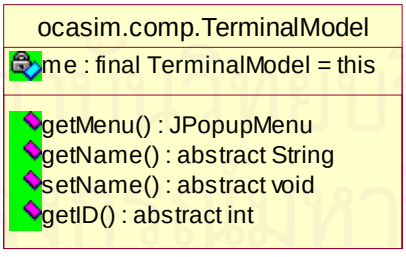

รูปที่ 4.23 คลาส ocasim.core.Paintable

11) คลาส ocasim.core.WireModel คือ คลาสที่ออกแบบเพื่อทำหน้าที่จัดการเกี่ยวกับการลากเส้นเพื่อเชื่อมต่อระหว่างอุปกรณ์รวมถึงการส่งผ่านค่าสัญญาณระหว่างอุปกรณ์ด้วย รายละเอียดของคลาส แสดงดังรูปที่ 4.24



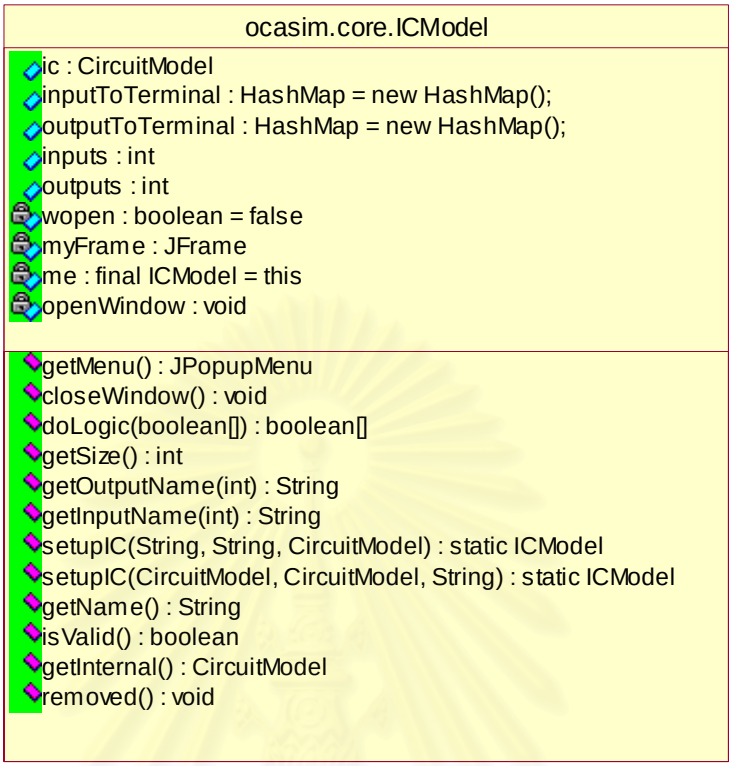

รูปที่ 4.24 คลาส ocasim.core.WireModel

12) คลาส ocasim.comp.TerminalModel คือ คลาสที่ออกแบบเป็นคลาสนามธรรมเพื่อเป็นคลาสต้นแบบสำหรับการสร้างคลาสตัวแปรนำเข้าชนิดต่าง ๆ และคลาสผลลัพธ์ รายละเอียดของคลาสแสดงดังรูปที่ 4.25




รูปที่ 4.25 คลาส ocasim.comp.TerminalModel

13) คลาส ocasim.core.ICModel คือ คลาสที่ออกแบบเพื่อทำหน้าที่เก็บข้อมูลของตัวแปรนำเข้าและตัวแปรผลลัพธ์ทั้งหมดในวงจรที่ออกแบบสร้างขึ้น รายละเอียดของคลาส แสดงดังรูปที่ 4.26

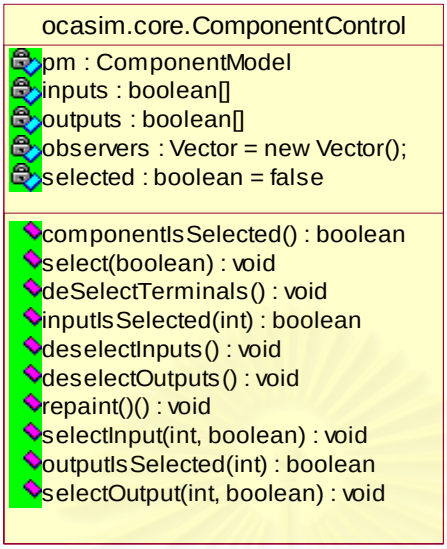



รูปที่ 4.26 คลาส ocasim.core.ICModel

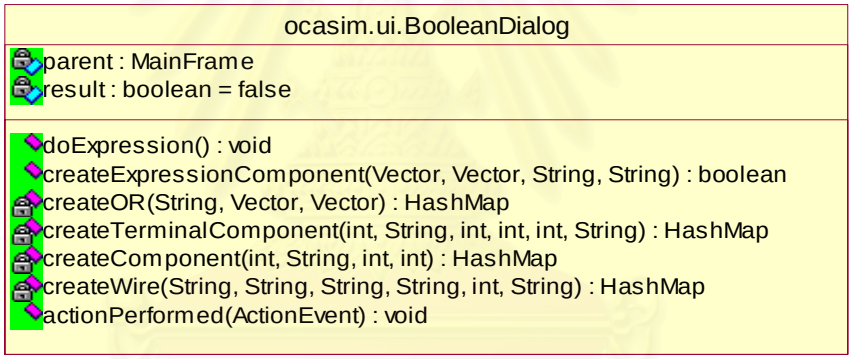
14) คลาส ocasim.core.ComponentControl คือ คลาสที่ออกแบบเพื่อทำหน้าที่สำหรับควบคุมการทำงานกับอุปกรณ์ต่าง เช่น การเลือกอุปกรณ์เกตแอนด์มาวางบนพื้นที่วาดวงจร คลาสนี้จะจำได้ว่าเลือกอุปกรณ์อะไรมาเพื่อที่จะแสดงผลและทำงานกับอุปกรณ์ในวงจรนั้น ๆ ได้อย่างถูกต้อง รายละเอียดของคลาสแสดงดังรูปที่ 4.27

15) คลาส ocasim.ui.BooleanDialog คือ ไดอะล็อกคลาสที่ออกแบบสำหรับสร้างหน้าจอขึ้นมาบนส่วนต่อประสานกับผู้ใช้เพื่อให้ผู้ออกแบบทำการกำหนดสมการบูลีนลงบนหน้าจอไดอะล็อกนี้ รายละเอียดของคลาสแสดงดังรูปที่ 4.28

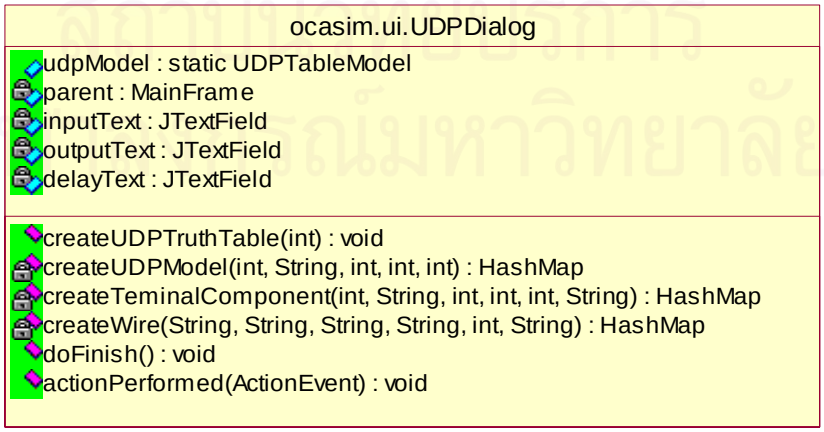
16) คลาส ocasim.ui.UDPDialog คือ ไดอะล็อกคลาสที่ออกแบบสำหรับสร้างหน้าจอขึ้นมาบนส่วนต่อประสานกับผู้ใช้เพื่อให้ผู้ออกแบบทำการกำหนดค่าจำนวนตัวแปรนำเข้าและค่าความหน่วงให้กับตัวอุปกรณ์ UDP ที่ต้องการสร้างขึ้น โดยกำหนดค่าลงบนหน้าจอไดอะล็อกนี้ รายละเอียดของคลาสแสดงดังรูปที่ 4.29




รูปที่ 4.27 คลาส ocasim.core.ComponentControl

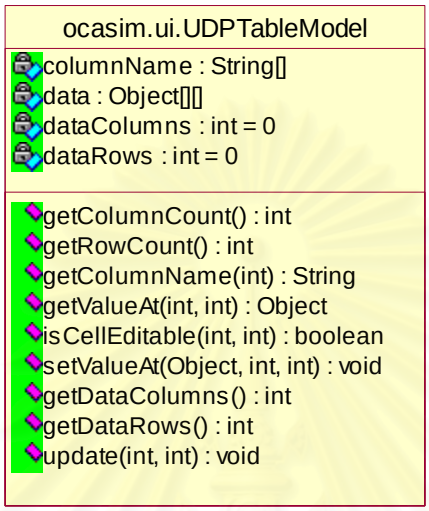



รูปที่ 4.28 คลาส ocasim.ui.BooleanDialog




รูปที่ 4.29 คลาส ocasim.ui.UDPDialog

17) คลาส ocasim.ui.UDPTableModel คือ คลาสที่ออกแบบสำหรับการสร้างตารางค่าความจริงตามจำนวนตัวแปรนำเข้าที่ได้จากการกำหนดค่าที่หน้าจอของคลาส ocasim.ui.UDPDialog รายละเอียดของคลาสแสดงดังรูปที่ 4.30

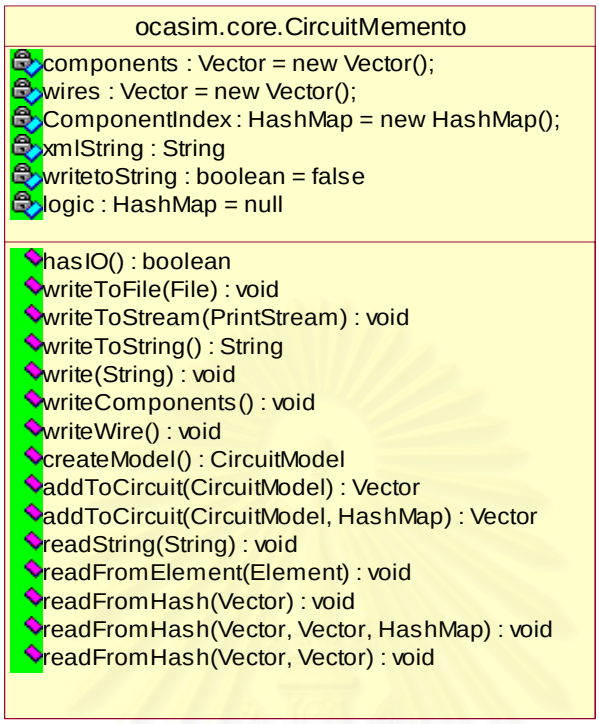



รูปที่ 4.30 คลาส ocasim.ui.UDPTableModel

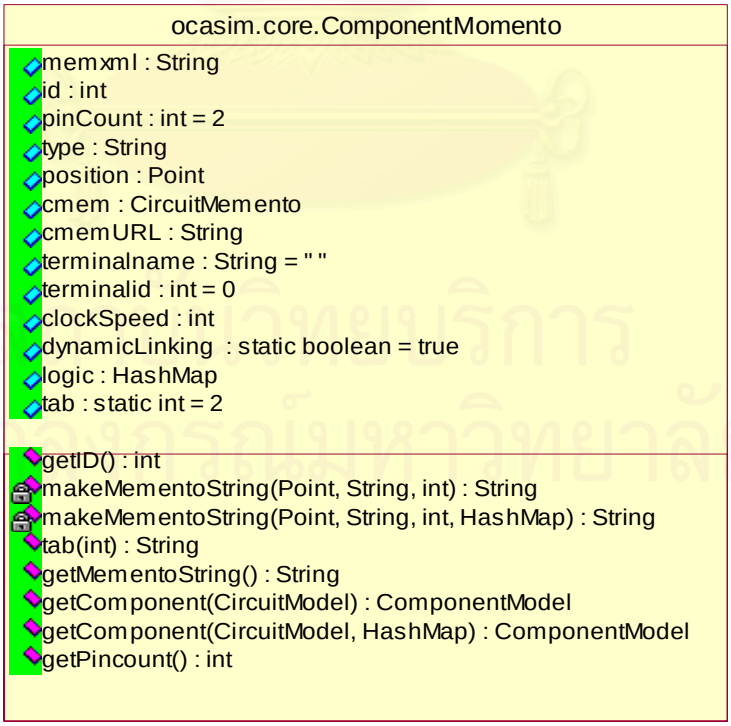
18) คลาส ocasim.core.CircuitMemento คือ คลาสที่ออกแบบให้เป็นคลาสหลักสำหรับการจัดการกับอุปกรณ์หรือวงจรที่สร้างขึ้นมาเพื่อทำหน้าที่บันทึกข้อมูลแฟ้มเอกสารเอ็กซ์เอ็มแอลลงไฟล์และการเรียกคืนข้อมูลไฟล์เอ็กซ์เอ็มแอลกลับขึ้นมาทำงานอีก รายละเอียดของคลาสแสดงดังรูปที่ 4.31

19) คลาส ocasim.core.ComponentMemento คือ คลาสที่ออกแบบเพื่อทำหน้าที่ในการแปลงข้อมูลจากแฟ้มเอ็กซ์เอ็มแอลกลับไปเป็นอ็อบเจ็คที่นำมาสร้างวงจรและการแปลงจากอ็อบเจ็คไปเป็นแฟ้มข้อมูลเอ็กซ์เอ็มแอลเพื่อนำไปบันทึกเป็นไฟล์อีกที โดยคลาสนี้จะถูกคลาส ocasim.core.CircuitMemento เรียกใช้งานอีกที รายละเอียดของคลาสแสดงดังรูปที่ 4.32

20) คลาส ocasim.core.Command คือ คลาสที่ออกแบบสำหรับเก็บการทำงานต่าง ๆ ตามแต่ละชุดคำสั่งที่ได้มีการร้องขอมาจากคลาส Actions เช่น คำสั่งการออกแบบสร้างวงจรใหม่ คำสั่งการบันทึกวงจรที่ออกแบบลงไฟล์ในรูปแบบแฟ้มเอกสารเอ็กซ์เอ็มแอลและทำสั่งการเรียกคืนอุปกรณ์หรือวงจรกลับขึ้นมาเพื่อทำการแก้ไขหรือทดสอบการทำงาน รายละเอียดของคลาสแสดงดังรูปที่ 4.33




รูปที่ 4.31 คลาส ocasim.core.CircuitMemento

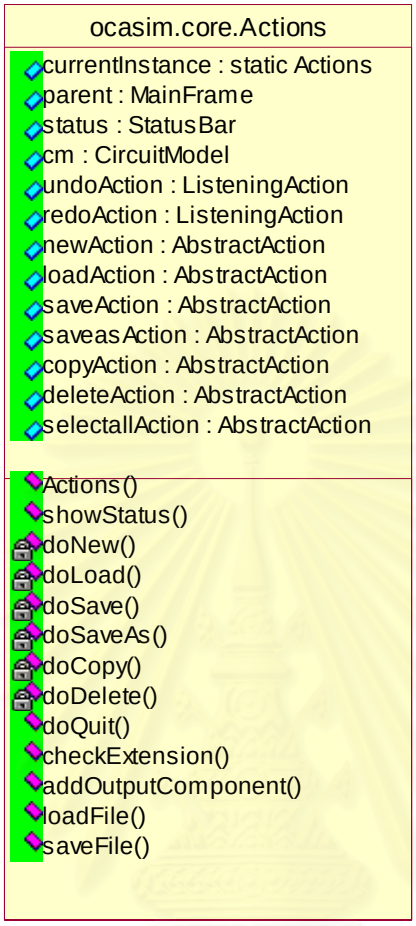



รูปที่ 4.32 คลาส ocasim.core.ComponentMemento

รูปที่ 4.33 คลาส ocasim.core.Command

21) คลาส ocasim.core.Actions คือ คลาสที่ออกแบบสำหรับเก็บชุดคำสั่งต่าง ๆ ของเครื่องมือ แต่การทำงานตามชุดคำสั่งที่เลือกนั้นจะเก็บอยู่ที่คลาส ocasim.core.Command อีกที รายละเอียดของคลาสแสดงดังรูปที่ 4.34

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



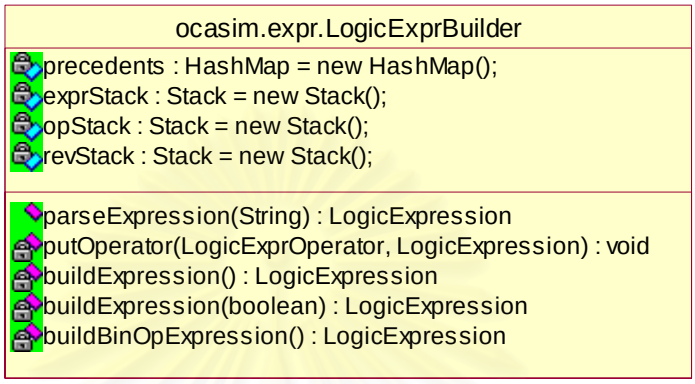

รูปที่ 4.34 คลาส ocasim.core.Actions

22) คลาส ocasim.expr.LogicExprBuilder คือ คลาสที่ออกแบบสำหรับแปลงสมการทางคณิตศาสตร์ที่ผู้ออกแบบกำหนดให้กลายเป็น Expression Tree รายละเอียดของคลาสแสดงดังรูปที่ 4.35

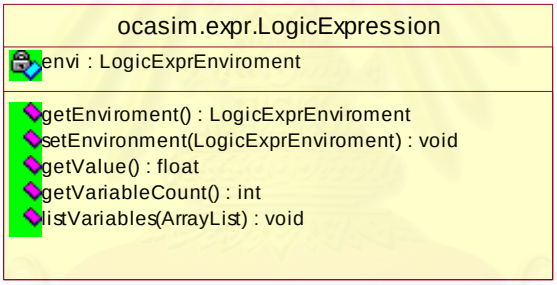
23) คลาส ocasim.expr.LogicExpression คือ คลาสที่ออกแบบสำหรับเป็นคลาสแม่ของคลาสตัวทำดำเนินการต่าง ๆ และตัวถูกดำเนินการต่าง ๆ รายละเอียดแสดงดังรูปที่ 4.36

24) คลาส ocasim.expr.ExpressionTruthTableBuilder คือ คลาสที่ออกแบบสำหรับการสร้าง Truth Table จาก Expression Tree ที่ได้ รายละเอียดแสดงดังรูปที่ 4.37

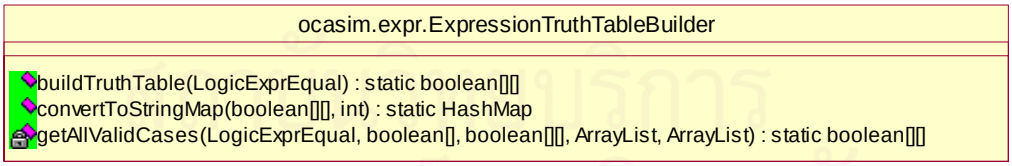
25) คลาส ocasim.expr.TruthCaseBuilder คือ คลาสที่ออกแบบสำหรับการสร้างกรณีที่เป็นไปได้ทั้งหมดจากจำนวนตัวแปรนำเข้าและจำนวนตัวแปรผลลัพธ์ รายละเอียดแสดงดังรูปที่ 4.38



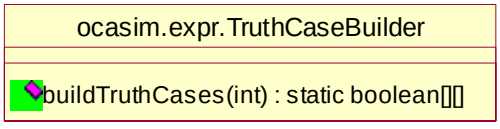

รูปที่ 4.35 คลาส ocasim.expr.LogicExprBuilder




รูปที่ 4.36 คลาส ocasim.expr.LogicExpression

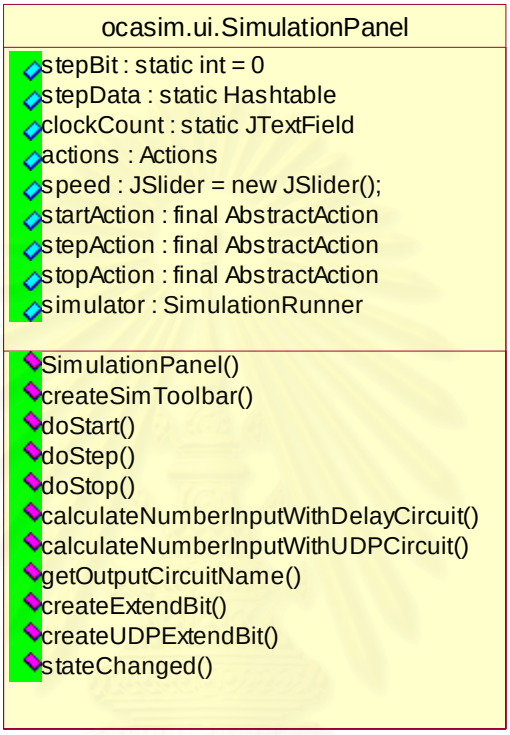



รูปที่ 4.37 คลาส ocasim.expr.ExpressionTruthTableBuilder



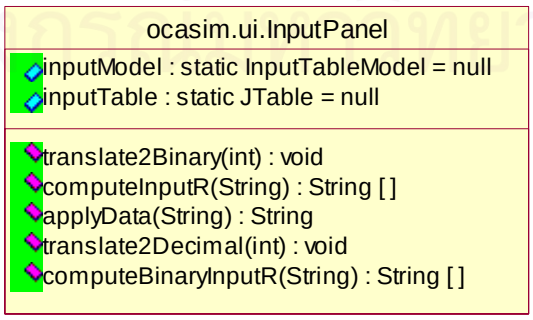

รูปที่ 4.38 คลาส ocasim.expr.TruthCaseBuilder

26) คลาส ocasim.ui.SimulationPanel คือ คลาสหลักที่ออกแบบสำหรับการสร้างส่วนคอนโทรลการจำลองการทำงานและทำงานร่วมกับส่วนกำหนดค่าตัวแปรนำเข้าเพื่อแสดงค่าผลลัพธ์ รายละเอียดของคลาสแสดงดังรูปที่ 4.39



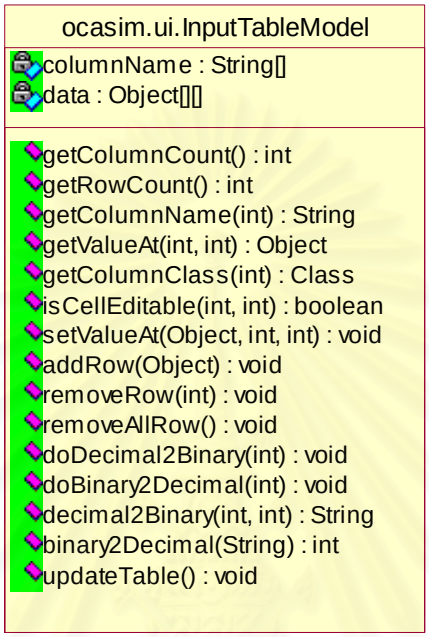

รูปที่ 4.39 คลาส ocasim.ui.SimulationPanel

27) คลาส ocasim.ui.InputPanel คือ คลาสที่ออกแบบสำหรับให้ผู้ออกแบบได้ทำการป้อนค่าตัวเลขให้กับตัวแปรนำเข้าทุกตัวโดยออกแบบให้สามารถป้อนค่าตัวเลขตัวเลขได้ทั้งค่าตัวเลขฐานสิบและฐานสอง รายละเอียดของคลาสแสดงดังรูปที่ 4.40



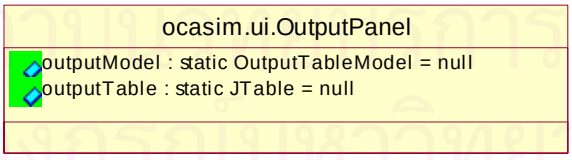

รูปที่ 4.40 คลาส ocasim.ui.InputPanel

28) คลาส ocasim.ui.InputTableModel คือ คลาสที่ออกแบบสำหรับการแสดงส่วนของการกำหนดค่าตัวแปรนำเข้าผ่านทางคลาส ocasim.ui.InputPanel รายละเอียดของคลาสแสดงดังรูปที่ 4.41



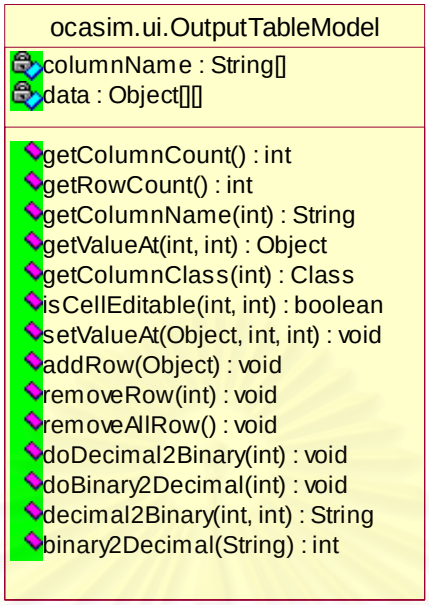

รูปที่ 4.41 คลาส ocasim.ui.InputTableModel

29) คลาส ocasim.ui.OutputPanel คือ คลาสที่ออกแบบสำหรับการแสดงผลค่าผลลัพธ์ที่ได้จากการจำลองการทำงานผ่านทางตัวแปรผลลัพธ์ โดยออกแบบให้แสดงผลได้ 2 รูปแบบคือ ตัวเลขฐานสิบและตัวเลขฐานสอง รายละเอียดของคลาสแสดงดังรูปที่ 4.42



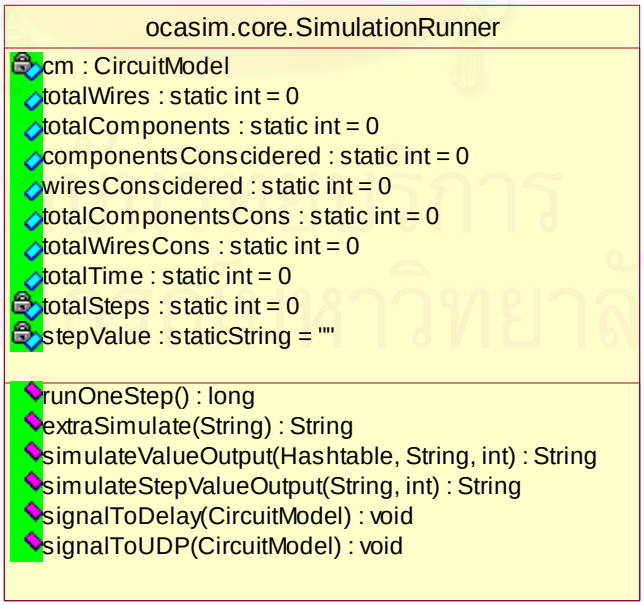

รูปที่ 4.42 คลาส ocasim.ui.OutputPanel

30) คลาสคลาส ocasim.ui.OutputTableModel คือ คลาสที่ออกแบบสำหรับการแสดงส่วนของกาแสดงค่าผลลัพธ์ผ่านทางตัวแปรผลลัพธ์โดยทำงานร่วมกับคลาส ocasim.ui.InputPanel รายละเอียดของคลาสแสดงดังรูปที่ 4.43




รูปที่ 4.43 คลาส ocasim.ui.OutputTableModel

31) คลาส ocasim.core.SimulationRunner คือ คลาสที่ออกแบบสำหรับการจำลองการทำงานของวงจรที่ออกแบบโดยสามารถเลือกจำลองการทำงานได้ 2 รูปแบบ คือ จำลองการทำงานทั้งวงจรภายในครั้งเดียวและจำลองการทำงานทีละขั้น (ครั้งละ 1 คล็อก) รายละเอียดของคลาสแสดงดังรูปที่ 4.44

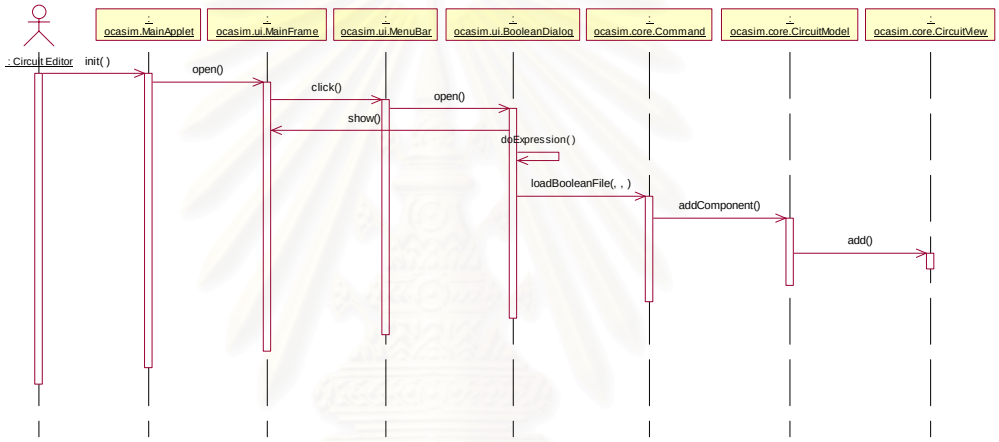



รูปที่ 4.44 คลาส ocasim.core.SimulationRunner

4.2.3 แผนภาพซีเควนซ์

แผนภาพซีเควนซ์คือ แผนภาพที่ใช้สำหรับแสดงการโต้ตอบของวัตถุในระบบ โดยแสดงให้เห็นถึงลำดับการส่งข้อความร้องขอของวัตถุในระบบ แผนภาพซีเควนซ์ที่แสดงต่อไปนี้เป็นแผนภาพที่อธิบายการโต้ตอบของวัตถุในระบบโดยจำแนกตามหน้าที่การทำงานที่แสดงไว้แล้วในแผนภาพยูสเคสดังนี้

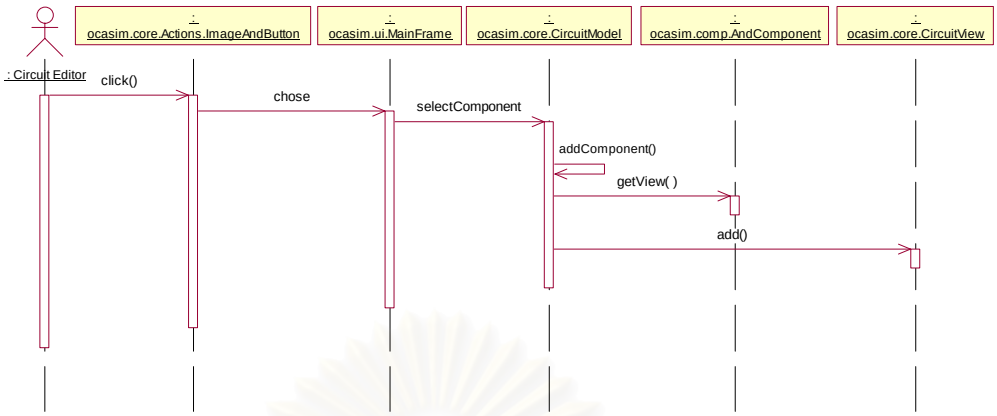
1) จากรายละเอียดยูสเคส Create New Circuit with Boolean Equation สามารถอธิบายขั้นตอนการทำงานด้วยแผนภาพซีเควนซ์ แสดงดังรูปที่ 4.45




รูปที่ 4.45 แผนภาพซีเควนซ์แสดงขั้นตอนการสร้างวงจรด้วยสมการบูลีน

จากแผนภาพซีเควนซ์นี้ การทำงานจะเริ่มต้นจาก ผู้ออกแบบทำการเปิดหน้าจอแอพเพล็ทของเครื่องมือขึ้นมาทำงานจากนั้นทำการกดปุ่ม Start บนหน้าจอแอพเพล็ทเพื่อให้เครื่องมือเปิดหน้าจอส่วนต่อประสานกับผู้ใช้ขึ้นมาทำงาน หากผู้ออกแบบต้องการออกแบบสร้างวงจรด้วยการกำหนดสมการบูลีนให้ทำการเลือกการทำงานนี้ผ่านทางเมนูที่ชื่อว่า Boolean Equation จากนั้นเครื่องมือจะเปิดหน้าจอสำหรับกำหนดค่าสมการบูลีนที่ต้องการขึ้นมาบนหน้าจอส่วนต่อประสานกับผู้ใช้อีกทีหนึ่ง เมื่อผู้ออกแบบทำการกำหนดสมการบูลีนที่ต้องการเสร็จแล้ว เครื่องมือจะทำสั่งสำหรับการวาดรูปแทนสมการบูลีนที่กำหนดลงบนพื้นที่ส่วนวาดวงจร

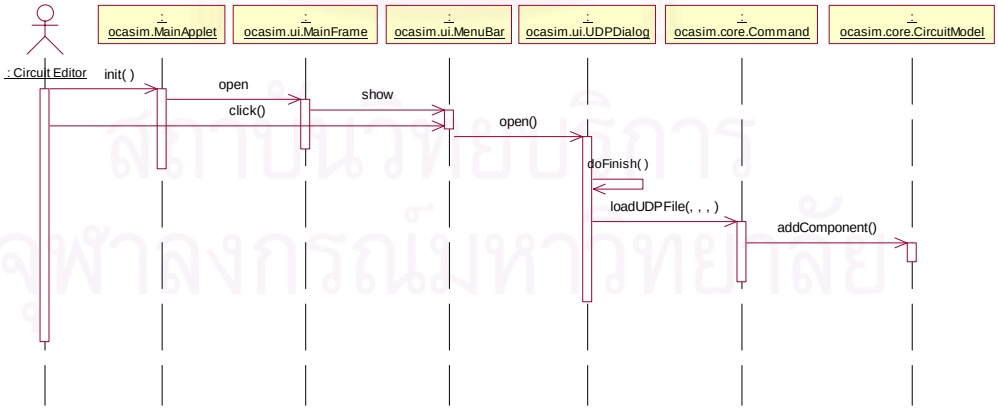
2) จากรายละเอียดยูสเคส Create New Circuit with Exist Component สามารถอธิบายขั้นตอนการทำงานด้วยแผนภาพซีเควนซ์ แสดงดังรูปที่ 4.46




รูปที่ 4.46 แผนภาพซีเควนซ์แสดงขั้นตอนการสร้างวงจรด้วยการวาดเป็นผังวงจรจากอุปกรณ์ที่มีอยู่บนคลังอุปกรณ์

จากแผนภาพซีเควนซ์นี้ เป็นการอธิบายการสร้างวงจรด้วยการวาดเป็นผังวงจรโดยการนำอุปกรณ์เกตแอนด์ซึ่งมีอยู่แล้วบนส่วนคลังอุปกรณ์มาวางลงบนพื้นที่ส่วนวาดวงจร นอกจากนี้ ผู้ออกแบบยังสามารถลากอุปกรณ์เกตพื้นฐานอื่น ๆ อุปกรณ์ประกอบหรืออุปกรณ์ตัวแปรนำเข้าและตัวแปรผลลัพธ์มาวางเพิ่มลงไปได้ ซึ่งก็จะมีกระบวนการเหมือนกันเพียงแต่เปลี่ยนตัวอุปกรณ์เท่านั้น

3) จากรายละเอียดยูสเคส Create UDP Component สามารถอธิบายขั้นตอนการทำงานด้วยแผนภาพซีเควนซ์ แสดงดังรูปที่ 4.47

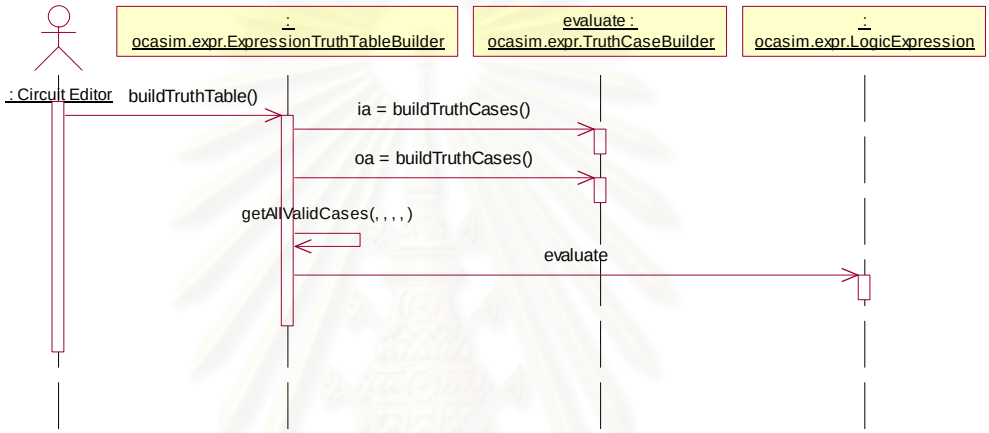



รูปที่ 4.47 แผนภาพซีเควนซ์แสดงขั้นตอนการสร้างอุปกรณ์ UDP

จากแผนภาพซีเควนซ์นี้ การทำงานจะคล้ายกันกับการออกแบบสร้างวงจรด้วยสมการบูลีนเพียงแต่ผู้ออกแบบต้องเลือกการทำงานนี้จากเมนู User-defined primitive เพื่อทำ

การกำหนดจำนวนตัวแปรนำเข้าและค่าความหน่วงที่ต้องการจากนั้นเครื่องมือจะเปิดหน้าจอขึ้นมาให้ทำการกำหนดค่าผลลัพธ์ได้เองในตารางค่าความจริงจากนั้นเครื่องมือจะวาดรูปแทนอุปกรณ์ UDP นี้ ขึ้นมาด้วยรูปสี่เหลี่ยมและมีจำนวนขาเชื่อมต่อเข้ากับตัวแปรนำเข้าและตัวแปรผลลัพธ์ตามที่ได้กำหนดค่าไว้

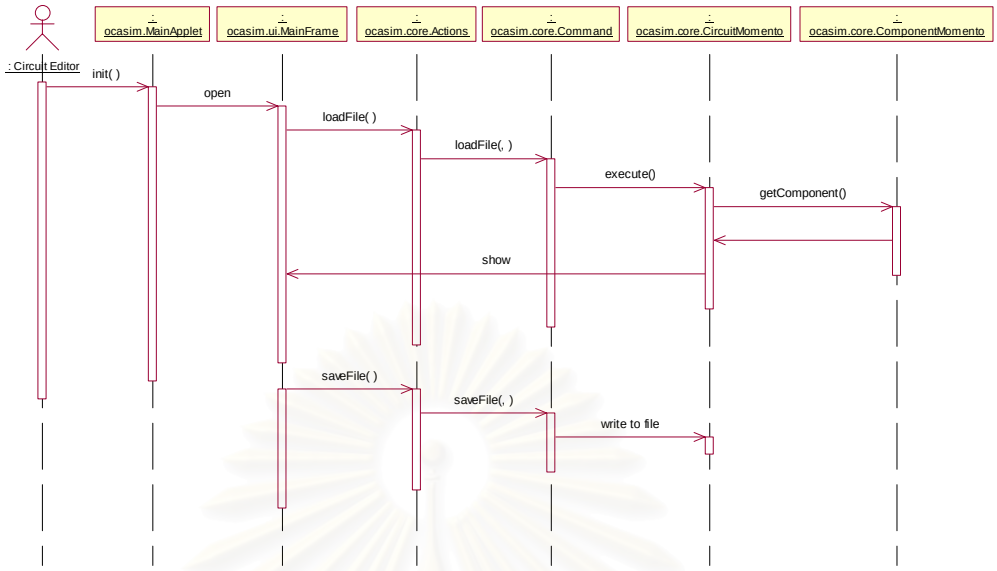
4) รายละเอียดยูสเคส Create UDE Component สามารถอธิบายขั้นตอนการทำงานด้วยแผนภาพซีเควนซ์ แสดงดังรูปที่ 4.48




รูปที่ 4.48 แผนภาพซีเควนซ์แสดงขั้นตอนการสร้างอุปกรณ์ UDE

จากแผนภาพซีเควนซ์นี้ การทำงานจะคล้ายกันกับการออกแบบสร้างวงจรด้วยสมการบูลีนเพียงแต่ผู้ออกแบบต้องเลือกการทำงานนี้จากเมนู User-defined primitive เพื่อทำการกำหนดสมการทางคณิตศาสตร์ จากนั้นเครื่องมือจะทำการตรวจสอบสมการที่ผู้ออกแบบกำหนดว่าให้ค่าที่ถูกต้องหรือไม่เพื่อทำการสร้างตารางค่าความจริงให้อัตโนมัติ หากไม่ถูกต้องก็จะทำการแจ้งเตือน

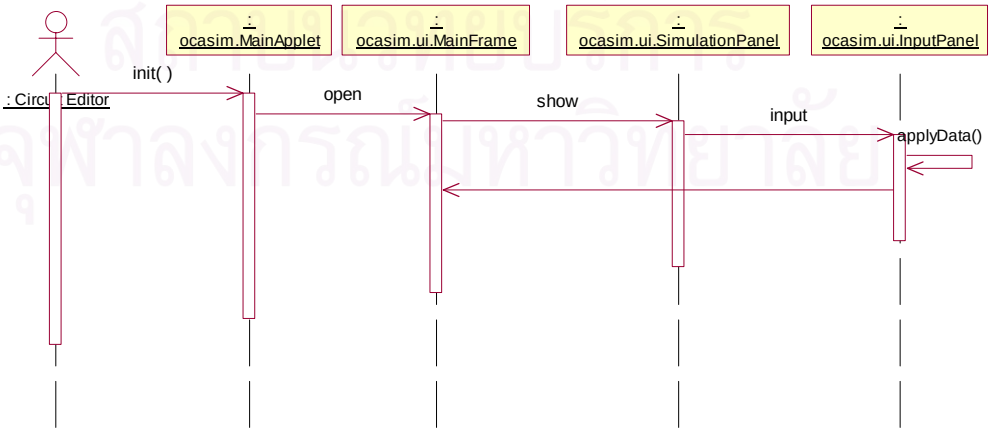
5) จากรายละเอียดยูสเคส Load Exist a XML File และยูสเคส Save Design as a XML File สามารถอธิบายขั้นตอนการทำงานด้วยแผนภาพซีเควนซ์ แสดงดังรูปที่ 4.49




รูปที่ 4.49 แผนภาพซีเควนซ์แสดงขั้นตอนการเรียกคืนหรือการบันทึกข้อมูลแฟ้มเอกสารเอ็กซ์เอ็มแอล

จากแผนภาพซีเควนซ์นี้ แสดงขั้นตอนการทำงานสำหรับการเรียกไฟล์ข้อมูลแฟ้มเอกสารเอ็กซ์เอ็มแอลกลับขึ้นมาแสดงผลบนหน้าจอในรูปของอ็อบเจ็คที่วาดเป็นวงจรและแสดงขั้นตอนสำหรับการทำการบันทึกข้อมูลวงจรที่ออกแบบสร้างขึ้นเก็บลงไฟล์ในรูปแบบของแฟ้มเอกสารเอ็กซ์เอ็มแอล

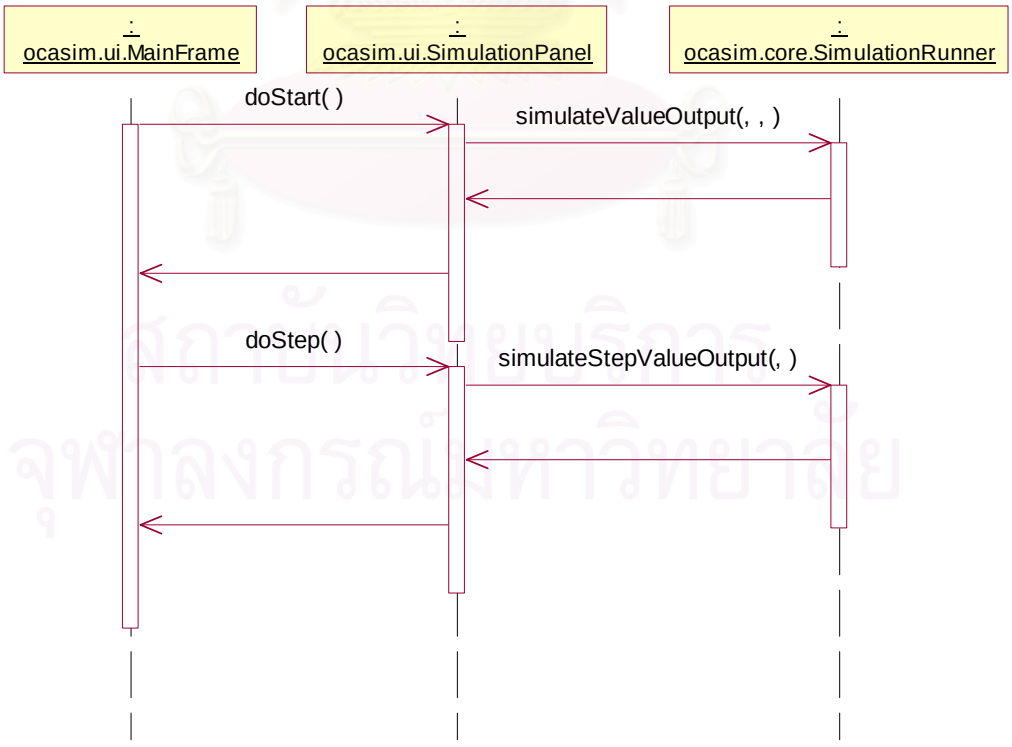
6) จากรายละเอียดยูสเคส Set Input Value to a Input Component สามารถอธิบายขั้นตอนการทำงานด้วยแผนภาพซีเควนซ์ แสดงดังรูปที่ 4.50




รูปที่ 4.50 แผนภาพซีเควนซ์แสดงขั้นตอนการกำหนดค่าให้กับตัวแปรนำเข้า

จากแผนภาพซีเควนซ์นี้ การทำงานจะเริ่มต้นจาก ผู้ออกแบบทำการเปิดหน้าจอแอพเพล็ทของเครื่องมือขึ้นมาทำงานจากนั้นทำการกดปุ่ม Start บนหน้าจอแอพเพล็ทเพื่อให้เครื่องมือเปิดหน้าจอส่วนต่อประสานกับผู้ใช้ขึ้นมาทำงาน จากนั้นผู้ออกแบบจะต้องทำการเรียกคืนวงจรที่เคยออกแบบไว้แล้วขึ้นมาเพื่อทดสอบจำลองการทำงานได้ โดยเมื่อเครื่องมือทำการเรียกคืนวงจรที่ต้องการมาแสดงบนพื้นที่ส่วนวาดวงจรแล้ว ผู้ออกแบบก็สามารถลากอุปกรณ์ตัวแปรนำเข้าและตัวแปรผลลัพธ์ที่ต้องการได้จากคลังอุปกรณ์เพื่อมาประกอบเข้ากับวงจรที่ทำการเรียกคืนขึ้นมา โดยเมื่อทำการวางอุปกรณ์ตัวแปรนำเข้าและตัวแปรผลลัพธ์ลงบนส่วนวาดวงจร เครื่องมือจะทำการเพิ่มจำนวนแถวในส่วนกำหนดค่าตัวแปรนำเข้าให้ตามจำนวนตัวแปรนำเข้าที่ลากมาวาง จากนั้นผู้ออกแบบก็สามารถป้อนค่าให้กับตัวแปรนำเข้าชนิดต่าง ๆ ทั้งหมดที่มีได้ โดยเครื่องมือจะทำการตรวจสอบรูปแบบของการป้อนค่าตัวเลขด้วยว่าถูกต้องหรือไม่ และจะทำการแปลงค่าตัวเลขที่ป้อนให้ควบคู่ไปด้วยกันทั้งช่องป้อนค่าตัวเลขฐานสิบและช่องป้อนค่าตัวเลขฐานสอง

7) จากรายละเอียดยูสเคส Simulate ซึ่งเกี่ยวข้องกับการเลือกยูสเคส Start หรือ Step มาทำงานร่วมด้วย ดังนั้นสามารถอธิบายขั้นตอนการทำงานด้วยแผนภาพซีเควนซ์ แสดงดังรูปที่ 4.51




รูปที่ 4.51 แผนภาพซีเควนซ์แสดงขั้นตอนการจำลองการทำงานทั้ง 2 รูปแบบ

จากแผนภาพซีเควนซ์นี้ การทำงานจะเริ่มต้นจาก ผู้ออกแบบทำการเลือกรูปแบบการจำลองการทำงานกับวงจรที่เรียกคืนมาบนพื้นที่ส่วนวาดวงจร โดยผู้ออกแบบสามารถเลือกรูปแบบการจำลองการทำงานได้ 2 รูปแบบ คือ เลือกให้การจำลองการทำงานเกิดขึ้นอย่างต่อเนื่องจนแล้วเสร็จหรือเลือกให้เกิดการแสดงผลการจำลองการทำงานเกิดขึ้นทีละขั้น



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

# บทที่ 5

# การพัฒนาและการทดสอบ

ในบทนี้จะกล่าวถึง การพัฒนาเครื่องมือและการทดสอบการทำงานของเครื่องมือสำหรับจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรง ซึ่งมีรายละเอียดดังนี้

## 5.1 การพัฒนาเครื่องมือ

ผู้วิจัยได้ทำการพัฒนาเครื่องมือสำหรับจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรงในรูปของวงจรย่อยต่าง ๆ ประกอบเข้าด้วยกัน ซึ่งกำหนดการทำงานด้วยตารางค่าความจริงของหน่วยนั้น เครื่องมือนี้ได้รับการพัฒนาโดยใช้ภาษาจาวาซึ่งเป็นการเขียนโปรแกรมเชิงวัตถุให้ใช้งานได้ในรูปแบบของแอพเพล็ทและจัดเก็บวงจรที่ออกแบบในรูปแบบของแฟ้มเอกสารเอ็กซ์เอ็มแอล สำหรับการพัฒนาเครื่องมือ รายละเอียดดังนี้

### 5.1.1 สภาพแวดล้อมที่ใช้ในการพัฒนาเครื่องมือ

สภาพแวดล้อมของฮาร์ดแวร์และซอฟท์แวร์ ที่ใช้ในการพัฒนาเป็นดังนี้

1) ฮาร์ดแวร์ (Hardware)

   1.1) เครื่องคอมพิวเตอร์ส่วนบุคคลหน่วยประมวลผลกลางอินเทลเพนเทียมเอ็ม 1.5 กิกะเฮิร์ทซ์

   1.2) หน่วยความจำสำรอง 512 เมกะไบต์

   1.3) ฮาร์ดดิสก์ 40 กิกะไบต์

2) ซอฟท์แวร์ (Software)

   2.1) ระบบปฏิบัติการไมโครซอฟท์วินโดวส์เอ็กซ์พี โพรเฟชชันแนล เวอร์ชัน 2002 (Microsoft Windows XP Professional 2002)

   2.2) เครื่องมือที่ใช้พัฒนา บอร์แลนด์ เจบิวเดอร์ เอ็กซ์ เอ็นเตอร์ไพล์ (Borland JBuilder X Enterprise)

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

2.3) ติดตั้งเจดีเคเวอร์ชัน 1.4 (JDK Version 1.4)

2.4) เว็บบราวเซอร์ อินเทอร์เน็ตเอ็กซ์พลอเรอร์เวอร์ชัน 6.0 (IE 6.0)

5.1.2 เทคนิคที่ใช้ในการพัฒนาเครื่องมือ

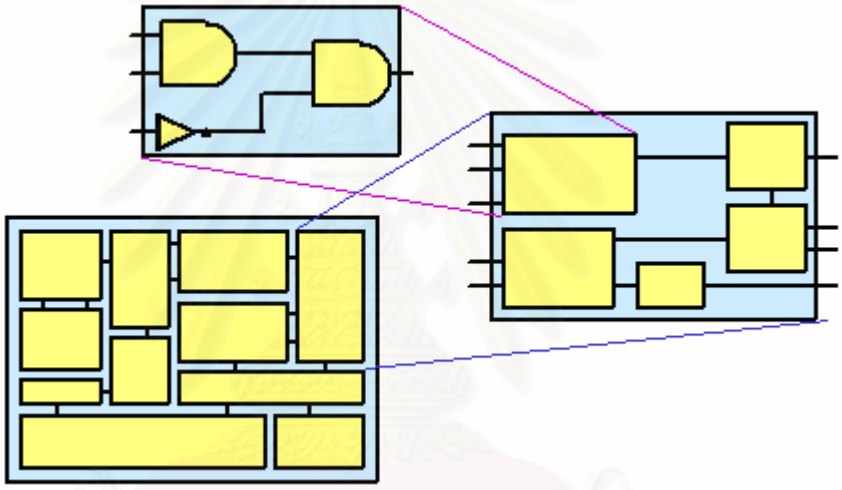
สำหรับเทคนิคที่ใช้ในการพัฒนาเครื่องมือนั้น ผู้วิจัยใช้วิธีการเชิงเหตุการณ์ในการออกแบบ โดยเลือกใช้การเขียนโปรแกรมเชิงวัตถุ (OOP: Object Oriented Programming) นำมาออกแบบพัฒนาสร้างเครื่องมือสำหรับจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรง เนื่องจากการออกแบบอัลกอริทึมหรือการออกแบบสร้างวงจรบนเครื่องมือที่พัฒนาขึ้นนี้เป็นการนำอุปกรณ์หลาย ๆ ตัวมาต่อเชื่อมประกอบเข้าด้วยกัน จึงจำเป็นต้องเก็บการเชื่อมโยงกันของอุปกรณ์ต่าง ๆ เอาไว้เพื่อที่จะสามารถส่งการเปลี่ยนแปลงของสัญญาณไปให้กับอุปกรณ์ที่เกี่ยงข้องกับสัญญาณนั้นได้ โดยทั่วไปแล้วเอาท์พุตของอุปกรณ์หนึ่งก็จะเป็นอินพุตของอุปกรณ์อื่น ๆ และอินพุตของอุปกรณ์นั้น ๆ ก็มักจะต่อเข้ากับเอาท์พุตของอุปกรณ์อื่นเช่นกัน สิ่งจำเป็นคือ ต้องมีวิธีการที่เหมาะสมสำหรับเก็บการเชื่อมต่อกันของอุปกรณ์ต่าง ๆ เอาไว้เพื่อที่จะสามารถส่งต่อสัญญาณระหว่างอุปกรณ์ได้อย่างถูกต้องและมีประสิทธิภาพ

แนวคิดการเขียนโปรแกรมเชิงวัตถุจะมองโปรแกรมเป็นรูปแบบของวัตถุที่มีพฤติกรรมเฉพาะของตัวเองและมีความสัมพันธ์กับวัตถุอื่น การมองปัญหาในลักษณะนี้นำมาใช้กับปัญหาการจำลองการทำงานของวงจรที่ออกแบบได้เป็นอย่างดี กล่าวคือ เมื่อทำการมองอุปกรณ์แต่ละตัวเป็นอ็อบเจ็ค อุปกรณ์แต่ละชนิดก็จะมีคุณสมบัติเป็นของตัวเอง การนำอุปกรณ์มาเชื่อมต่อประกอบเข้าด้วยกันก็คือ การนำอ็อบเจ็คมาประกอบเข้าด้วยกัน

อุปกรณ์แต่ละชนิดมีพฤติกรรมที่ตอบสนองต่ออินพุตแตกต่างกัน การออกแบบโดยใช้ OOP ทำให้ไม่จำเป็นต้องสนใจถึงพฤติกรรมที่เฉพาะตัวของอุปกรณ์แต่ละชนิด และไม่ต้องสนใจรายละเอียดของการสร้างเครื่องมือส่วนนั้นว่าจะเป็นแบบใด หรือจะต้องใช้วิธีการใดเพื่อให้อุปกรณ์มีคุณสมบัติตามที่ต้องการ แต่สิ่งแรกที่จะต้องวิเคราะห์คือการที่อุปกรณ์เหล่านั้นประกอบสิ่งใดที่เหมือนกันบ้าง เช่น เราไม่สนใจว่าอุปกรณ์เกต AND และอุปกรณ์เกต NAND เมื่อป้อนค่าอินพุตที่เหมือนกันจะให้เอาท์พุตต่างกันหรือไม่อย่างไร หรือ เราจะไม่สนใจว่าเราจะสร้างคลาสของเกต NAND ด้วยการเก็บตารางค่าความจริง หรือจะใช้การเขียนโปรแกรมย่อยเฉพาะสำหรับเกตนั้น แต่เราจะสนใจที่ทั้งอุปกรณ์เกต AND และอุปกรณ์เกต NAND หรืออุปกรณ์ใดๆ จะมีช่องสำหรับต่อกับอุปกรณ์อื่นๆ เหมือนๆ กัน ซึ่งช่องนั้นอาจใช้เพื่อรับสัญญาณเข้า หรือส่งสัญญาณออก หรือทั้ง

สองอย่าง ถ้าอุปกรณ์นั้นมีช่องที่ใช้รับสัญญาณเข้า ถ้าสัญญาณที่เข้าเกิดเปลี่ยนแปลง ก็อาจทำให้สัญญาณออกเปลี่ยนแปลงได้ การเปลี่ยนแปลงจะเกิดขึ้นตามแต่เวลาหน่วงของอุปกรณ์ สิ่งที่อุปกรณ์ต่างๆ มีเหมือนๆ กันเหล่านี้จะถูกนำมาเป็นซูเปอร์คลาสของอุปกรณ์ในเครื่องมือนี้ทั้งหมด

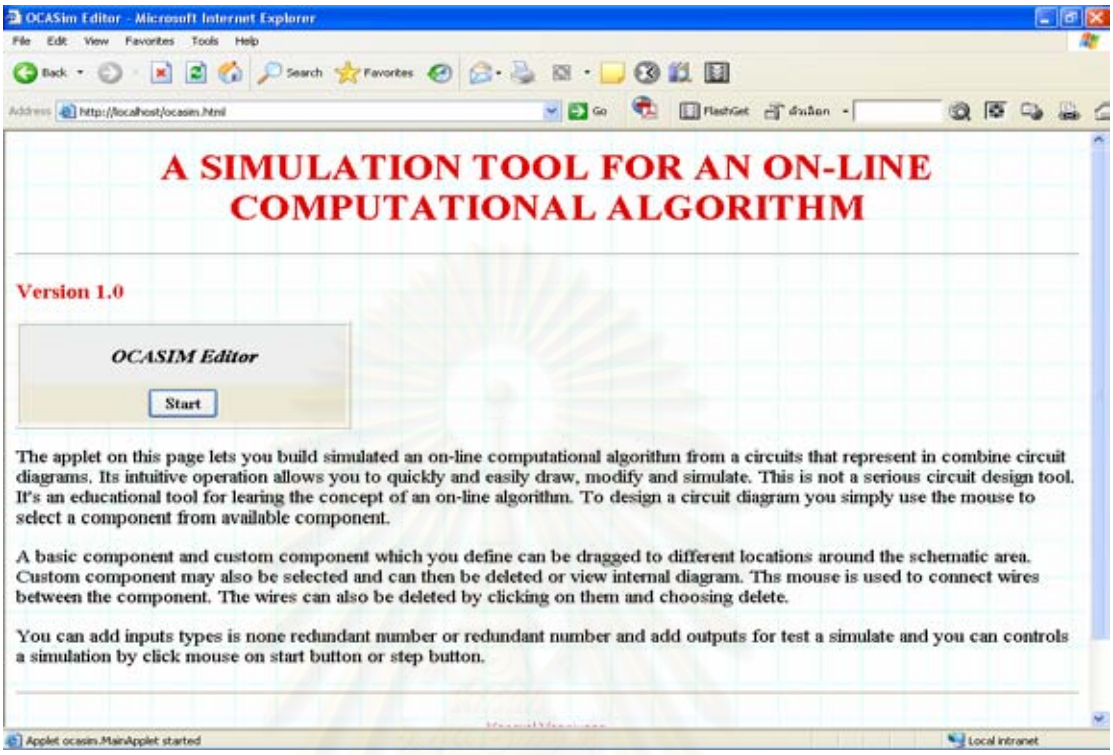
สำหรับวงจรที่ซับซ้อนจะประกอบไปด้วยวงจรที่เล็กกว่าหลาย ๆ วงจรประกอบเข้าด้วยกันดังเช่นในรูปที่ 5.1 ผู้วิจัยได้ออกแบบคลาส ocasim.core.ICModel ซึ่งสืบทอดมาจากคลาส ocasim.core.ComponentModel เป็นตัวเก็บวงจรย่อยเหล่านั้นไว้ภายใน โดยภายนอกจะมองเห็นอ็อบเจ็คของคลาสนี้เป็นเสมือนอ็อบเจ็คของอุปกรณ์อื่น ๆ ซึ่งอ็อบเจ็คของคลาสนี้สามารถประกอบกับอ็อบเจ็คอื่น เพื่อสร้างวงจรที่ซับซ้อนขึ้นไปอีกได้



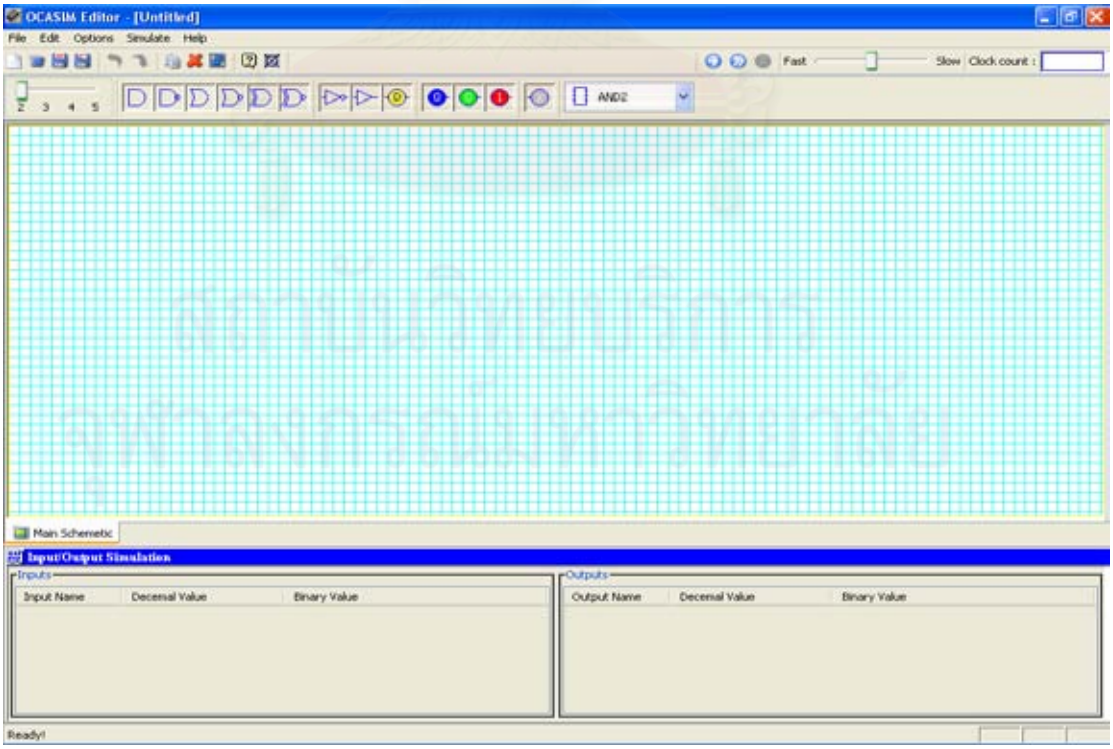
รูปที่ 5.1 วงจรย่อย ๆ ประกอบรวมเป็นส่วนประกอบของวงจรที่ใหญ่ขึ้น

หากพิจารณาการจำลองการทำงานของอุปกรณ์หนึ่งชิ้นซึ่งโดยทั่วไปจะมีทั้งอินพุตและเอาท์พุตที่รับสัญญาณเข้าและส่งออกจากอุปกรณ์นั้น ๆ เมื่อเกิดการเปลี่ยนแปลงของสัญญาณหรือข้อมูลนำเข้าที่อินพุตอาจทำให้เกิดการเปลี่ยนแปลงที่เอาท์พุตในช่วงเวลาถัดไปเท่ากับค่าความหน่วงของอุปกรณ์ตัวนั้น ฉะนั้นก่อนที่จะจำลองเหตุการณ์นี้ได้จะต้องทราบคุณสมบัติของอุปกรณ์นั้นว่าทำงานอย่างไร และการเปลี่ยนแปลงสัญญาณจะใช้ช่วงเวลาหนึ่ง (ค่าความหน่วง) ก่อนที่สัญญาณจะส่งไปที่เอาท์พุตหรือไม่ ตัวอย่างเช่น อุปกรณ์เกต NOT มีคุณสมบัติของตัวมันเองว่าสัญญาณเอาท์พุตจะตรงกันข้ามกับสัญญาณที่อินพุต เป็นต้น

### 5.1.3 โครงสร้างของเครื่องมือที่พัฒนา




รูปที่ 5.2 หน้าจอการทำงานเป็นแอพเพล็ทบนบราวเซอร์

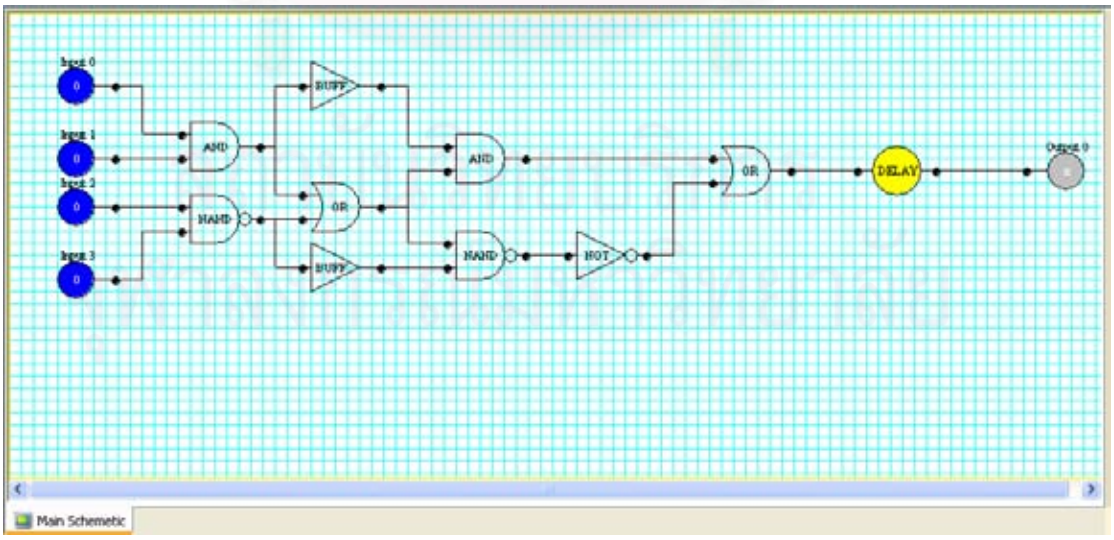



รูปที่ 5.3 หน้าจอหลักของเครื่องมือ

จากรูปที่ 5.2 แสดงหน้าจอการทำงานของเครื่องมือบนบราวเซอร์ในรูปแบบของแอพเพ็ท เมื่อผู้ออกแบบกดปุ่มสตาร์ทบนหน้าจอนี้แอพเพล็ทจะทำการเปิดหน้าจอหลักของเครื่องมือขึ้นมา แสดงดังรูปที่ 5.3 ซึ่งประกอบไปด้วยส่วนการทำงานสี่ส่วนด้วยกัน รายละเอียดดังนี้

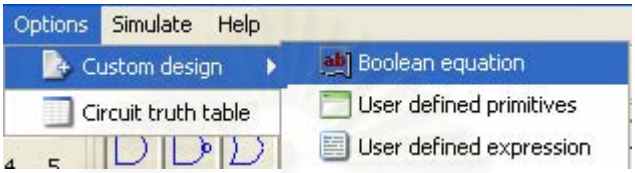
5.1.3.1 ส่วนการออกแบบ คือ ส่วนที่ทำหน้าที่เกี่ยวกับการออกแบบสร้างอุปกรณ์หรือวงจร เป็นส่วนที่ผู้ออกแบบสามารถออกแบบสร้างวงจรได้ 2 วิธี คือ การออกแบบวงจรด้วยสมการบูลีนหรือการออกแบบวงจรด้วยการเลือกอุปกรณ์ที่มีอยู่จากคลังอุปกรณ์มาประกอบเข้าด้วยกันเป็นผังวงจร หรือออกแบบสร้างอุปกรณ์ UDP ขึ้นมา โดยกำหนดการทำงานของอุปกรณ์นั้นด้วยค่าความหน่วงและตารางค่าความจริงที่ผู้ออกแบบเป็นผู้กำหนดค่าได้เอง หรือการออกแบบสร้างอุปกรณ์ UDE ซึ่งได้จากการกำหนดสมการทางคณิตศาสตร์ ส่วนการออกแบบประกอบด้วยส่วนต่าง ๆ ดังนี้

1) ส่วนวาดวงจร คือ ส่วนที่มีไว้สำหรับการออกแบบสร้างอุปกรณ์หรือวงจร โดยผู้ออกแบบสามารถออกแบบสร้างวงจรได้ 2 วิธีคือ การออกแบบสร้างวงจรด้วยการวาดเป็นผังวงจร โดยเลือกอุปกรณ์ที่มีอยู่จากส่วนคลังอุปกรณ์มาวางและเชื่อมโยงอุปกรณ์เหล่านั้นประกอบเข้าด้วยกันเป็นวงจร หรือการออกแบบสร้างวงจรด้วยสมการบูลีน โดยพื้นที่การทำงานส่วนนี้จะถูกแสดงด้วยกริด (Grid) และการจัดวางอุปกรณ์ต่าง ๆ จะกระทำในรูปแบบลากแล้วปล่อย นอกจากนี้การสร้างอุปกรณ์ UDP และอุปกรณ์ UDE ขึ้นมา ก็จะถูกแสดงบนพื้นที่ส่วนนี้ด้วยเช่นกัน แสดงดังรูปที่ 5.4

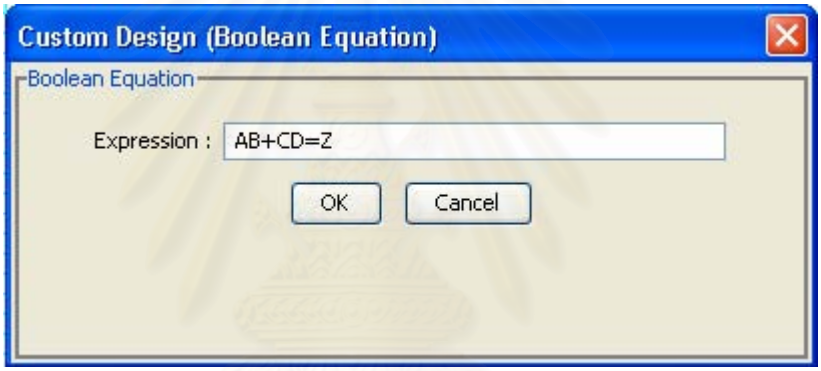



รูปที่ 5.4 ส่วนวาดวงจร

สำหรับการออกแบบสร้างวงจรด้วยสมการบูลีน เครื่องมือจะทำการวาดวงจรขึ้นบนส่วนวาดวงจรด้วยรูปที่สอดคล้องกันกับสมการที่กำหนด แต่ไม่สามารถกำหนดค่าความหน่วงให้กับวงจรได้และตารางค่าความจริงสำหรับวงจรก็จะเป็นไปตามจริงของการประกอบเข้าด้วยกันของอุปกรณ์ต่าง ๆ ที่เครื่องมือวาดขึ้น การสร้างวงจรประเภทนี้เริ่มต้นจากการเลือกเมนูแสดงดังรูปที่ 5.5 จากนั้นเครื่องมือจะสร้างหน้าต่างแสดงดังรูป 5.6 สำหรับให้กำหนดสมการบูลีน

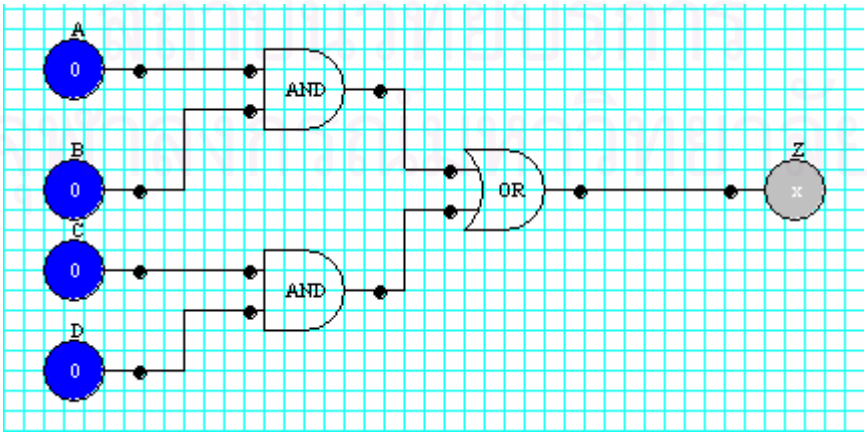



รูปที่ 5.5 เมนูสร้างวงจรที่กำหนดด้วยสมการบูลีน




รูปที่ 5.6 หน้าจอสำหรับกำหนดสมการบูลีน

เมื่อทำการกำหนดสมการบูลีนเสร็จแล้วเครื่องมือจะวาดวงจรขึ้นมาด้วยรูปที่สอดคล้องกันกับสมการบูลีนที่กำหนดบนส่วนสร้างอุปกรณ์หรือวาดวงจร แสดงดังรูปที่ 5.7




รูปที่ 5.7 วงจรที่วาดจากการกำหนดสมการบูลีน

2) ส่วนคลังอุปกรณ์ คือ ส่วนที่มีไว้สำหรับให้ผู้ออกแบบสามารถเลือกนำอุปกรณ์ที่มีอยู่ไปใช้ในการออกแบบสร้างวงจรได้สะดวกหรือแม้แต่เลือกอุปกรณ์เหล่านั้นเพื่อมาทดสอบจำลองการทำงานก็ตาม แสดงดังรูปที่ 5.8 ซึ่งในส่วนคลังอุปกรณ์นี้จะประกอบไปด้วยอุปกรณ์ 3 ประเภทด้วยกัน ดังนี้

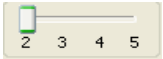
รูปที่ 5.8 ส่วนคลังอุปกรณ์

2.1) อุปกรณ์พื้นฐาน จะประกอบไปด้วย อุปกรณ์เกตพื้นฐาน แสดงดังรูปที่ 5.9 คือ อุปกรณ์เกตพื้นฐานต่าง ๆ ซึ่งทำงานตามตารางค่าความจริงทางตรรกะ โดยที่ค่าในตารางค่าความจริงของอุปกรณ์เกตพื้นฐานเหล่านี้จะไม่สามารถแก้ไขได้ และอุปกรณ์ UDP

รูปที่ 5.9 อุปกรณ์เกตพื้นฐาน

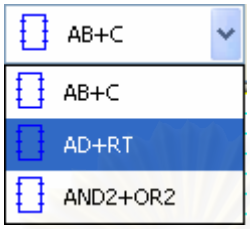
ตารางที่ 5.1 อุปกรณ์เกตพื้นฐาน

| สัญลักษณ์ | ชื่อ | คำอธิบาย |
|---|---|---|
| | AND | อุปกรณ์เกต AND ทำงานตามตารางค่าความจริงทางตรรกะ |
| | NAND | อุปกรณ์เกต NAND ทำงานตามตารางค่าความจริงทางตรรกะ |
| | OR | อุปกรณ์เกต OR ทำงานตามตารางค่าความจริงทางตรรกะ |
| | NOR | อุปกรณ์เกต NOR ทำงานตามตารางค่าความจริงทางตรรกะ |
| | XOR | อุปกรณ์เกต XOR ทำงานตามตารางค่าความจริงทางตรรกะ |
| | XNOR | อุปกรณ์เกต XNOR ทำงานตามตารางค่าความจริงทางตรรกะ |
| | NOT | อุปกรณ์เกต NOT ทำงานตามตารางค่าความจริงทางตรรกะ |
| | BUFFER | อุปกรณ์เกต BUFFER ทำงานตามตารางค่าความจริงทางตรรกะ |
| | DELAY | อุปกรณ์เกต DELAY มีค่าความหน่วงในตัวเองเท่ากับหนึ่งคล็อก |

ผู้ออกแบบสามารถเลือกกำหนดจำนวนตัวแปรนำเข้าให้กับอุปกรณ์เกตพื้นฐาน AND, NAND, OR, NOR, XOR, และ XNOR ได้ตั้งแต่ 2 -5 จำนวน ด้วยการเลือกจาก 

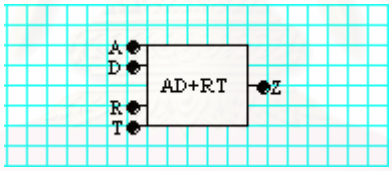
ซึ่งเครื่องมือสร้างเตรียมไว้แล้ว

2.2) อุปกรณ์ประกอบ คือ วงจรย่อยต่าง ๆ ที่เกิดจากการนำอุปกรณ์พื้นฐานและ/หรืออุปกรณ์ประกอบที่มีอยู่มาประกอบเข้าด้วยกันแล้วทำการบันทึกเก็บไว้ อุปกรณ์ UDP และอุปกรณ์ UDE ส่วนเก็บอุปกรณ์ประกอบแสดงดังรูปที่ 5.10

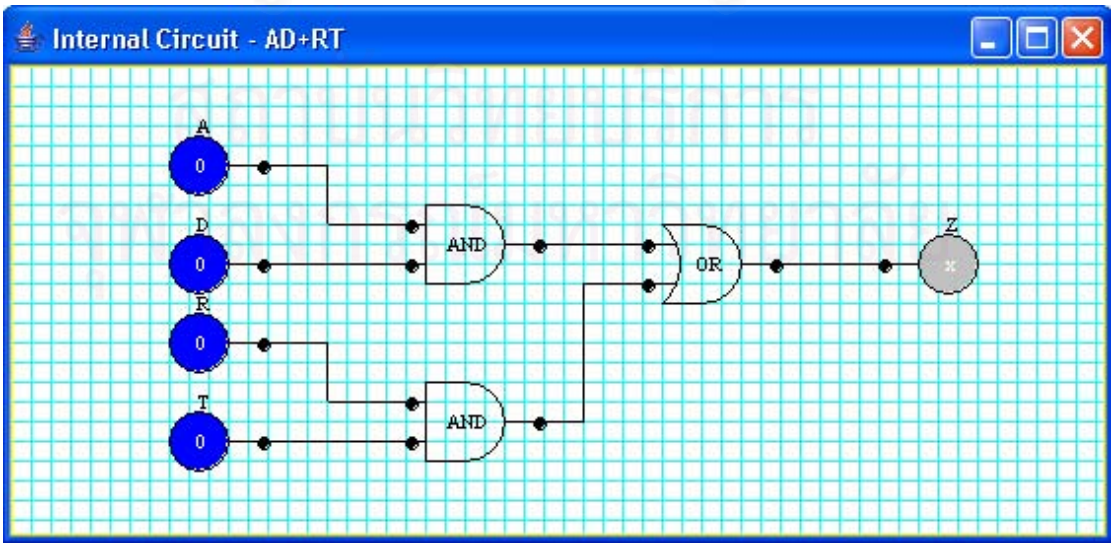



รูปที่ 5.10 อุปกรณ์ประกอบ

อุปกรณ์ประกอบนี้ ถ้าลากไปวางบนส่วนวาดวงจรจะแทนอุปกรณ์ประกอบได้ด้วยรูปสี่เหลี่ยมที่แสดงชื่ออุปกรณ์ จำนวนขาและชื่อของตัวแปรนำเข้าและตัวแปรผลลัพธ์เท่านั้น แสดงดังรูปที่ 5.11 หากต้องการทราบว่าอุปกรณ์ประกอบตัวนั้นมีโครงสร้างภายในประกอบด้วยอุปกรณ์ใดบ้างสามารถคลิกเมาส์ขวาที่ตัวอุปกรณ์แล้วเลือกเมนู View Internal เพื่อให้เครื่องมือเปิดหน้าต่างย่อยที่แสดงโครงสร้างภายในของอุปกรณ์ตัวนั้นขึ้นมา แสดงดังรูปที่ 5.12

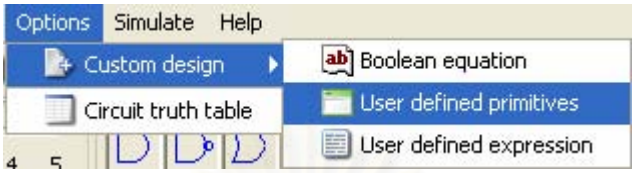



รูปที่ 5.11 แทนอุปกรณ์ประกอบชื่อ AD+RT ด้วยรูปสี่เหลี่ยม



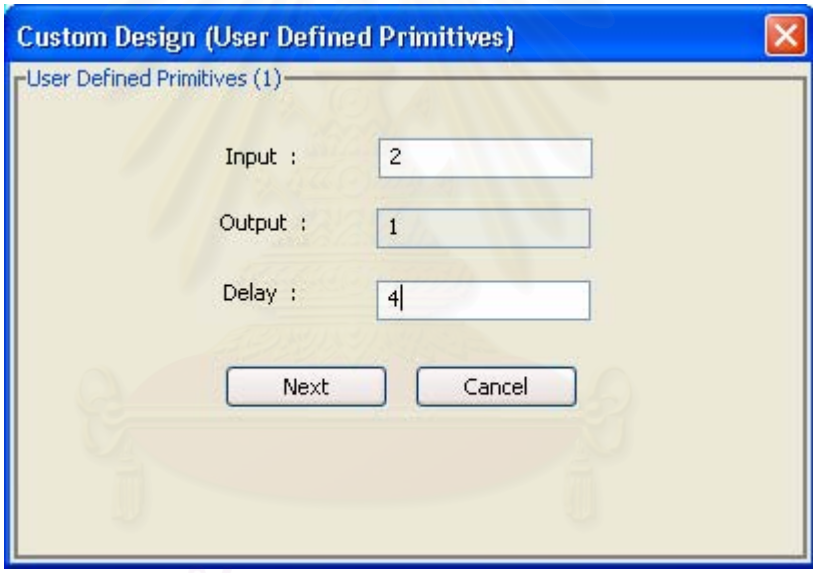

รูปที่ 5.12 โครงสร้างภายในของอุปกรณ์ประกอบที่ชื่อ AD+RT

สำหรับการออกแบบอุปกรณ์ UDP ผู้ออกแบบสามารถกำหนดฟังก์ชั่นการทำงานของอุปกรณ์ด้วยค่าความหน่วงและตารางค่าความจริงที่ผู้ออกแบบเป็นผู้กำหนดค่าผลลัพธ์เองตามต้องการ การสร้างอุปกรณ์ประเภทนี้เริ่มต้นจากการเลือกเมนูแสดงดังรูปที่ 5.13




รูปที่ 5.13 เมนูสร้างอุปกรณ์ที่กำหนดค่าความหน่วงและตารางค่าความจริงเอง

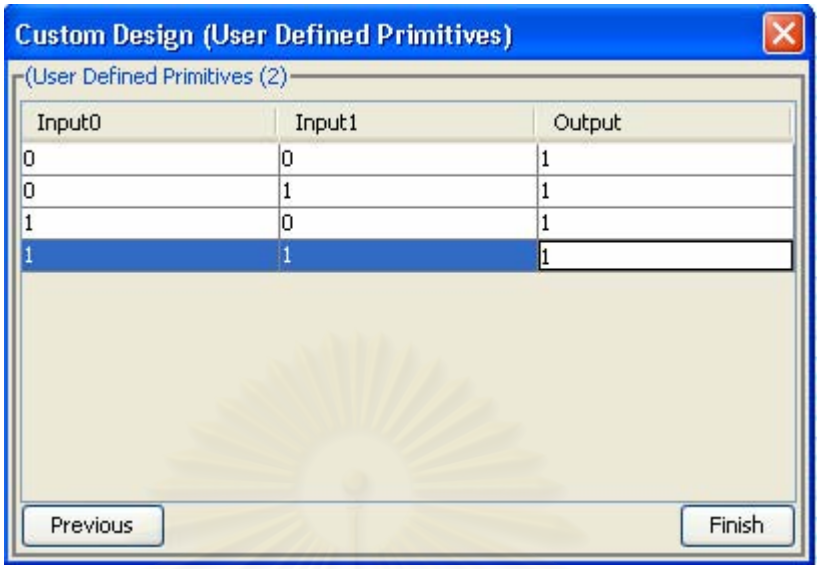
จากนั้นเครื่องมือจะสร้างหน้าต่างแสดงดังรูปที่ 5.14 เพื่อให้ผู้ออกแบบกำหนดจำนวนตัวแปรนำเข้าและกำหนดค่าความหน่วงสำหรับตัวมันเอง



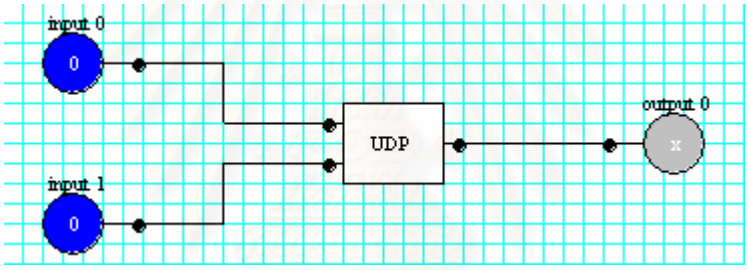

รูปที่ 5.14 หน้าจอกำหนดค่าความหน่วงสำหรับตัวอุปกรณ์ UDP

เมื่อกำหนดค่าเรียบร้อยแล้ว ผู้ออกแบบต้องทำการกดปุ่มเน็ก (Next) บนหน้าต่างนี้เพื่อเข้าสู่หน้าต่างถัดไปสำหรับการกำหนดค่าตารางค่าความจริงเอง ซึ่งสามารถกำหนดค่าได้เฉพาะตัวแปรผลลัพธ์เท่านั้น แสดงดังรูปที่ 5.15

เมื่อผู้ออกแบบทำการกดปุ่มฟินนิส (Finish) เครื่องมือจะวาดอุปกรณ์ขึ้นมาบนส่วนวาดวงจร ซึ่งแทนได้ด้วยรูปสี่เหลี่ยมมีจำนวนตัวแปรนำเข้าและตัวแปรผลลัพธ์ตามที่ได้กำหนดค่าไว้ แสดงดังรูปที่ 5.16




รูปที่ 5.15 หน้าจอสำหรับกำหนดค่าในตารางค่าความจริงเอง

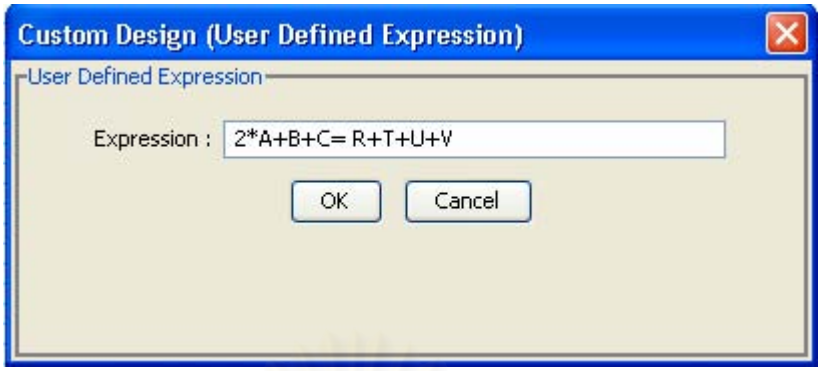



รูปที่ 5.16 อุปกรณ์ที่สร้างจากการกำหนดค่าความหน่วงและตารางค่าความจริงเอง

สำหรับการออกแบบอุปกรณ์ UDE ผู้ออกแบบสามารถกำหนดสมการทางคณิตศาสตร์ได้ตามต้องการ สร้างอุปกรณ์ประเภทนี้เริ่มต้นจากการเลือกเมนูแสดงดังรูปที่ 5.17

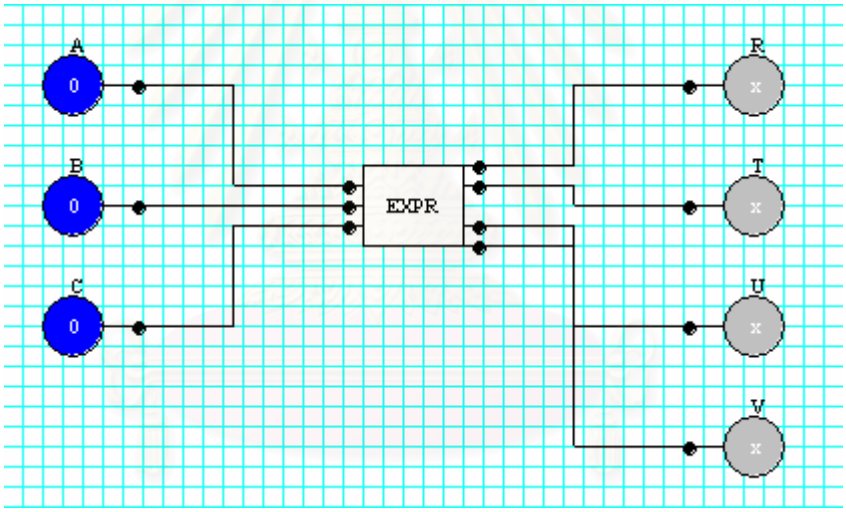
รูปที่ 5.17 เมนูสร้างอุปกรณ์ที่กำหนดด้วยสมการทางคณิตศาสตร์

จากนั้นเครื่องมือจะสร้างหน้าต่างแสดงดังรูปที่ 5.18 เพื่อให้ผู้ออกแบบกำหนดสมการทางคณิตศาสตร์ได้ตามต้องการ โดยจะต้องมีจำนวนตัวแปรนำเข้าและตัวแปรผลลัพธ์อยู่ในสมการได้สูงสุดไม่เกินอย่างละ 5 จำนวน



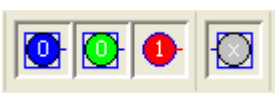

รูปที่ 5.18 หน้าจอสำหรับการกำหนดสมการทางคณิตศาสตร์

หากสมการที่กำหนดถูกต้อง เครื่องมือจะวาดอุปกรณ์ขึ้นมาบนส่วนวาดวงจร ซึ่งแทนได้ด้วยรูปสี่เหลี่ยมมีจำนวนตัวแปรนำเข้าและตัวแปรผลลัพธ์ตามสมการที่ได้กำหนดค่าไว้ แสดงดังรูปที่ 5.19




รูปที่ 5.19 อุปกรณ์ UDE ที่สร้างจากการกำหนดด้วยสมการทางคณิตศาสตร์

2.3) อุปกรณ์ตัวแปรนำเข้าและตัวแปรผลลัพธ์ แสดงดังรูปที่ 5.20 คือ อุปกรณ์ที่เครื่องมือสร้างเตรียมไว้ให้แล้วสำหรับให้ผู้ออกแบบได้เลือกนำไปประกอบเข้ากับอุปกรณ์พื้นฐานหรืออุปกรณ์ประกอบเพื่อการทดสอบจำลองการทำงานของวงจรที่สร้าง ประกอบไปด้วยอุปกรณ์ต่าง ๆ ดังนี้




รูปที่ 5.20 อุปกรณ์ตัวแปรนำเข้าและตัวแปรผลลัพธ์

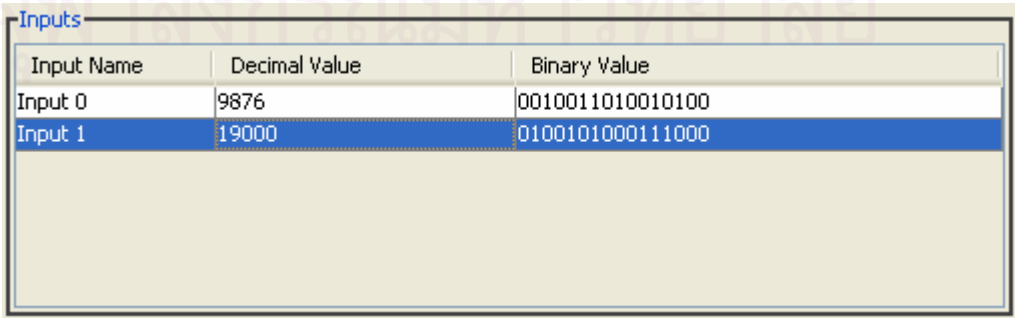
ตารางที่ 5.2 อุปกรณ์ตัวแปรนำเข้าและตัวแปรผลลัพธ์

| สัญลักษณ์ | ชื่อ | คำอธิบาย |
|---|---|---|
| | INPUT NR | ตัวแปรนำเข้าที่ไม่สามารถป้อนค่าติดลบเพื่อจำลองการทำงาน |
| | INPUT R | ตัวแปรนำเข้าที่ป้อนค่าได้ทั้งค่าบวกและลบเพื่อจำลองการทำงาน |
| | CONSTANT | ตัวแปรนำเข้าที่ส่งสัญญาณด้วยค่าคงที่เพื่อจำลองการทำงาน |
| | OUTPUT | ตัวแปรผลลัพธ์แสดงค่าผลลัพธ์ที่เกิดจากการจำลองการทำงาน |

5.1.3.2 ส่วนจำลองการทำงาน คือ ส่วนที่ทำหน้าที่เกี่ยวกับการกำหนดค่าตัวแปรนำเข้าเพื่อแสดงค่าผลลัพธ์การทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรงจากอุปกรณ์หรือวงจรที่สร้างขึ้นและการคอนโทรลการจำลองการทำงาน โดยผู้ออกแบบเป็นผู้ควบคุมการทำงานผ่านทางส่วนต่อประสานกับผู้ใช้ ส่วนจำลองการทำงานของเครื่องมือประกอบไปด้วยส่วนต่าง ๆ ดังนี้

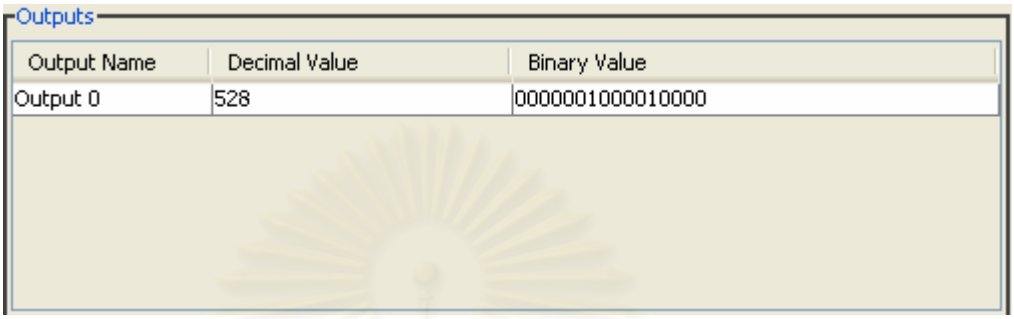
1) ส่วนกำหนดค่าตัวแปรนำเข้าเพื่อแสดงค่าผลลัพธ์ คือ ส่วนที่สร้างไว้สำหรับให้ผู้ออกแบบทำการป้อนค่าให้กับตัวแปรนำเข้าชนิดต่าง ๆ ทั้งหมดที่มีเพื่อการทดสอบการทำงานตามอัลกอริทึมการคำนวณแบบเชื่อมตรงจากอุปกรณ์หรือวงจรที่สร้างขึ้น ประกอบด้วยสองส่วนดังนี้

1.1) ส่วนกำหนดค่าตัวแปรนำเข้า คือ ส่วนที่มีไว้เพื่อให้ผู้ออกแบบได้ป้อนค่าให้กับตัวแปรนำเข้าชนิดต่าง ๆ ทั้งหมดตามแต่ชนิดของตัวแปรนำเข้าเพื่อทดสอบจำลองการทำงานอุปกรณ์หรือวงจรที่สร้างขึ้น โดยมีช่องสำหรับป้อนค่าตัวเลขอยู่ด้วยกันสองช่อง คือ ช่องป้อนค่าตัวเลขฐานสิบและช่องป้อนค่าตัวเลขฐานสอง ถ้าป้อนค่าที่ช่องใดช่องหนึ่งก็ตามเครื่องมือจะทำการแปลงค่าที่ถูกต้องของอีกช่องให้ควบคู่กันไปด้วย แต่ในการอ่านค่าเข้าไปทดสอบจำลองการทำงานเครื่องมือจะอ่านค่าจากช่องป้อนค่าตัวเลขฐานสองไปทำงานเพื่อแสดงค่าผลลัพธ์ที่ถูกต้อง แสดงดังรูปที่ 5.21




รูปที่ 5.21 ส่วนกำหนดค่าตัวแปรนำเข้า

1.2) ส่วนแสดงค่าผลลัพธ์ คือ ส่วนที่ทำการแสดงค่าผลลัพธ์ที่เกิดจากการจำลองการทำงานผ่านทางตัวแปรผลลัพธ์ทั้งหมดที่มี โดยแสดงค่าด้วยตัวเลขฐานสิบและตัวเลขฐานสองไปควบคู่กัน แสดงดังรูปที่ 5.22




รูปที่ 5.22 ส่วนแสดงค่าผลลัพธ์ผ่านทางตัวแปรผลลัพธ์

2) ส่วนควบคุมการจำลองการทำงาน คือ ส่วนที่เกี่ยวข้องกับการแสดงผลการทำงานผ่านทางส่วนต่อประสานผู้ใช้ ในเครื่องมือนี้ได้เตรียมคอลโทรลสำหรับให้ผู้ออกแบบเป็นผู้กำหนดการทำงานเองไว้ 2 ส่วน ดังนี้

2.1) ส่วนควบคุมความเร็วการจำลองการทำงาน คือ ส่วนที่ผู้ออกแบบสามารถเลือกระดับความเร็วในการแสดงผลการจำลองการทำงานของอัลกอริทึมหรือวงจรที่ออกแบบแต่ละครั้งได้ตามต้องการ แสดงดังรูปที่ 5.23

รูปที่ 5.23 ส่วนควบคุมความเร็วการจำลองการทำงาน

2.2) ส่วนประมวลผลการจำลองการทำงาน คือ ส่วนที่รองรับการแสดงผลลัพธ์ของการจำลองการทำงาน ผู้ออกแบบสามารถเลือกประมวลผลได้สองรูปแบบ คือ กดปุ่มสตาร์ทเพื่อเลือกให้การจำลองการทำงานเกิดขึ้นอย่างต่อเนื่องจนแล้วเสร็จหรือกดปุ่มสเต็พเพื่อให้การจำลองการทำงานทำไปทีละขั้น โดยผู้ออกแบบสามารถเลือกการทำงานในส่วนประมวลผลการจำลองการทำงานนี้ได้จากส่วนแถบเมนูการจำลองได้เช่นเดียวกัน แสดงดังรูปที่ 5.24

รูปที่ 5.24 ส่วนประมวลผลการจำลองการทำงาน

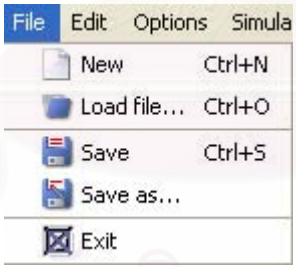
3) ส่วนแสดงจำนวนคล็อก คือ ส่วนที่แสดงจำนวนคล็อกของการทำงานที่เกิดจากการเลือกจำลองการทำงานให้เกิดขึ้นอย่างต่อเนื่องจนแล้วเสร็จหรือเลือกทำไปทีละขั้น เพื่อประโยชน์ในการตรวจสอบความเร็วและความถูกต้องของวงจรที่ออกแบบด้วย แสดงดังรูปที่ 5.25

รูปที่ 5.25 ส่วนแสดงจำนวนคล็อก

5.1.3.3 ส่วนแถบเมนูและแถบเครื่องมือ คือ ส่วนที่เก็บชุดคำสั่งที่เกี่ยวกับไฟล์ การแก้ไขตัวเลือก และการจำลองการทำงาน แสดงดังรูปที่ 5.26 ผู้ออกแบบสามารถเลือกใช้งานคำสั่งได้จากแถบเมนูหรือแถบเครื่องมือ ซึ่งจะให้ผลการทำงานที่เหมือนกัน โดยคำสั่งแบ่งตามแถบเมนูดังนี้

รูปที่ 5.26 แถบเมนูและแถบเครื่องมือ

1) แถบเมนูไฟล์ (File) แสดงดังรูปที่ 5.27 ประกอบไปด้วยคำสั่งต่าง ๆ แสดงดังตารางที่ 5.3

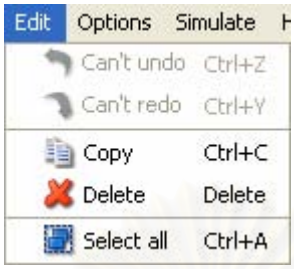



รูปที่ 5.27 แถบเมนูไฟล์

ตารางที่ 5.3 แถบเมนูไฟล์

| สัญลักษณ์ | ชื่อ | คำอธิบาย |
|---|---|---|
| | New | สร้างไฟล์ใหม่ จะเคลียร์ทุกอย่างบนส่วนวาดอุปกรณ์หรือวงจร |
| | Load file | เปิดไฟล์เก่าขึ้นมาแก้ไขหรือทดสอบจำลองการทำงาน |
| | Save | บันทึกอุปกรณ์หรือวงจรที่ออกแบบลงไฟล์ ในรูปแบบเอ็กซ์เอ็มแอล |
| | Save as | บันทึกอุปกรณ์หรือวงจรที่ออกแบบลงไฟล์ด้วยชื่อใหม่ |
| | Exit | ออกจากโปรแกรม |

2) แถบเมนูแก้ไข (Edit) แสดงดังรูปที่ 5.28 ประกอบด้วยคำสั่งต่าง ๆ แสดงดังตารางที่ 5.4

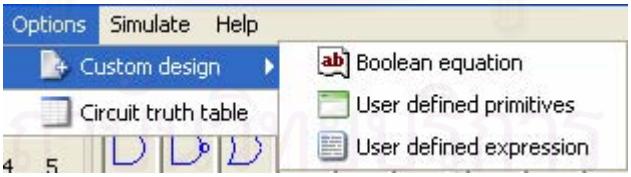



รูปที่ 5.28 แถบเมนูแก้ไข

ตารางที่ 5.4 แถบเมนูแก้ไข

| สัญลักษณ์ | ชื่อ | คำอธิบาย |
|---|---|---|
| | Undo | ยกเลิกการทำงานล่าสุดที่ผู้ออกแบบกระทำ |
| | Redo | กระทำซ้ำการทำงานล่าสุดที่ถูกยกเลิก |
| | Copy | คัดลอกอุปกรณ์หรือวงจรที่ผู้ออกแบบเลือกเพิ่มเข้าไปบนหน้าจอ |
| | Delete | ลบอุปกรณ์หรือวงจรที่ผู้ออกแบบเลือกออกจากหน้าจอ |
| | Select all | เลือกอุปกรณ์หรือวงจรทั้งหมดที่อยู่บนหน้าจอ |

3) แถบเมนูตัวเลือก (Options) แสดงดังรูปที่ 5.29 ประกอบไปด้วยคำสั่งต่าง ๆ แสดงดังตารางที่ 5.5

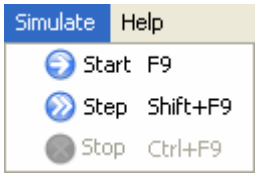



รูปที่ 5.29 แถบเมนูตัวเลือก

ตารางที่ 5.5 แถบเมนูตัวเลือก

| สัญลักษณ์ | ชื่อ | คำอธิบาย |
|---|---|---|
| | Circuit truth table | แสดงตารางค่าความจริงของวงจรที่ออกแบบ |
| | Boolean equation | ออกแบบสร้างวงจรด้วยการกำหนดสมการบูลีน |
| | User defined primitives | สร้างอุปกรณ์ UDP ที่กำหนดตารางค่าความจริงเอง |
| | User defined expression | สร้างอุปกรณ์ UDE จากสมการทางคณิตศาสตร์ |

4) แถบเมนูการจำลอง (Simulate) แสดงดังรูปที่ 5.30 ประกอบไปด้วยคำสั่งต่าง ๆ แสดงดังตารางที่ 5.6

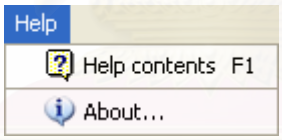



รูปที่ 5.30 แถบเมนูการจำลอง

ตารางที่ 5.6 แถบเมนูการจำลอง

| สัญลักษณ์ | ชื่อ | คำอธิบาย |
|---|---|---|
| | Start | จำลองการทำงานต่อเนื่องจนแล้วเสร็จ |
| | Step | จำลองการทำงานทีละขั้น |
| | Stop | ยุติการจำลองการทำงานของวงจร |

5) แถบเมนูช่วยเหลือ (Help) แสดงดังรูปที่ 5.31 ประกอบไปด้วยคำสั่งต่าง ๆ แสดงดังตารางที่ 5.7

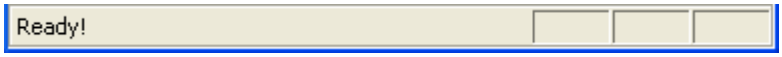



รูปที่ 5.31 แถบเมนูช่วยเหลือ

ตารางที่ 5.7 แถบเมนูช่วยเหลือ

| สัญลักษณ์ | ชื่อ | คำอธิบาย |
|---|---|---|
| | Help contents | แสดงข้อมูลวิธีการใช้งานเครื่องมือ |
| | About | แสดงข้อมูลเกี่ยวกับเครื่องมือ |

5.1.3.4 ส่วนแถบสถานะการทำงานของเครื่องมือ คือ ส่วนที่แสดงข้อมูลเกี่ยวกับสถานะการทำงานในขณะนั้นของเครื่องมือ แสดงดังรูปที่ 5.32




รูปที่ 5.32 แถบสถานะการทำงาน

## 5.2 การทดสอบเครื่องมือ

ผู้วิจัยได้ทำการทดสอบการทำงานของเครื่องมือสำหรับจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรง โดยกำหนดเงื่อนไขการทดสอบ กล่าวคือ

### 5.2.1 สภาพแวดล้อมที่ใช้ในการทดสอบ

ผู้วิจัยทำการจำลองเครื่องคอมพิวเตอร์ส่วนบุคคลให้เป็นเว็บเซิร์ฟเวอร์และเรียกใช้งานเครื่องมือในรูปของแอพเพล็ทบนเครื่องเดียวกัน โดยมีองค์ประกอบดังนี้

1) ระบบปฏิบัติการไมโครซอฟท์วินโดวส์เอ็กซ์พี โพรเฟซชันแนล เวอร์ชัน 2002 (Microsoft Windows XP Professional 2002)

2) ติดตั้งทอมแคทเวอร์ชัน 4.1.27 (TOMCAT Version 4.1.27) ทำหน้าที่เป็นเว็บเซิร์ฟเวอร์

3) ติดตั้งเจทูอาร์อีเวอร์ชัน 1.4 (J2RE Version 1.4)

4) เรียกใช้งานเครื่องมือผ่านบราวเซอร์อินเทอร์เน็ตเอ็กซ์พลอเรอร์เวอร์ชัน 6.0 (IE 6.0)

### 5.2.2 กรณีทดสอบ

ผู้วิจัยได้ทำการแบ่งกรณีทดสอบออกเป็น 2 ส่วน คือ ส่วนขอบเขตของการวิจัยครั้งนี้ และส่วนการวิเคราะห์องค์ประกอบของเครื่องมือจากบทที่ 3 โดยสามารถสรุปกรณีทดสอบในแต่ละส่วน รายละเอียดดังนี้

1) ทดสอบกับส่วนขอบเขตของการวิจัย

ตารางที่ 5.8 แสดงกรณีทดสอบการเรียกใช้งานเครื่องมือในรูปแบบของแอพเพล็ท

| หมายเลขกรณีทดสอบ | TC1 |
|---|---|
| ชื่อ | ทดสอบการเรียกใช้งานในรูปแบบของแอพเพล็ท |
| วัตถุประสงค์ | เพื่อทดสอบการทำงานของเครื่องมือบนระบบอินเทอร์เน็ตบนพื้นที่ใช้งานบราวเซอร์ โดยเรียกใช้งานในรูปแบบของแอพเพล็ท |
| คำอธิบาย | แสดงการทำงานบนบราวเซอร์ในรูปแบบของแอพเพล็ท |

ตารางที่ 5.8 แสดงกรณีทดสอบการเรียกใช้งานเครื่องมือในรูปแบบของแอพเพล็ท (ต่อ)

| หมายเลขกรณีทดสอบ | TC1 | |
|---|---|---|
| เงื่อนไขในการทดสอบ | - | |
| ผลลัพธ์ที่คาดว่าจะได้รับ | แสดงผลการทำงานในรูปแบบของแอพเพล็ทได้ | |
| ผลลัพธ์ที่เกิดขึ้นจริง | แสดงผลการทำงานในรูปแบบของแอพเพล็ทได้ | |
| ผลการทดสอบ | ☒ ผ่าน | ☐ ไม่ผ่าน |
| หมายเหตุ | ทดสอบการเรียกใช้งานบนเครื่องเดียวกับเว็บเซิร์ฟเวอร์ | |

ตารางที่ 5.9 แสดงกรณีทดสอบการทำงานผ่านระบบอินเทอร์เน็ตและบันทึกอุปกรณ์หรือวงจรที่สร้างลงบนเครื่องคอมพิวเตอร์ลูกข่าย

| หมายเลขกรณีทดสอบ | TC2 | |
|---|---|---|
| ชื่อ | ทดสอบการบันทึกอุปกรณ์หรือวงจรที่ออกแบบบนเครื่องคอมพิวเตอร์ลูกข่าย | |
| วัตถุประสงค์ | เพื่อทดสอบการทำงานของเครื่องบนระบบอินเทอร์เน็ตที่สามารถบันทึกการออกแบบสร้างอุปกรณ์หรือวงจรลงบนเครื่องคอมพิวเตอร์ลูกข่ายได้ | |
| คำอธิบาย | 1. แสดงการใช้งานเครื่องมือบนระบบอินเทอร์เน็ตผ่านบราวเซอร์<br>2. แสดงการเก็บบันทึกอุปกรณ์และวงจรที่สร้างขึ้นบนระบบอินเทอร์เน็ตลงบนเครื่องคอมพิวเตอร์ลูกข่าย | |
| เงื่อนไขในการทดสอบ | - | |
| ผลลัพธ์ที่คาดว่าจะได้รับ | แสดงการทำงานของเครื่องมือได้ถูกต้องตามกรณีทดสอบ | |
| ผลลัพธ์ที่เกิดขึ้นจริง | แสดงการทำงานของเครื่องมือได้ถูกต้องตามกรณีทดสอบ | |
| ผลการทดสอบ | ☒ ผ่าน | ☐ ไม่ผ่าน |
| หมายเหตุ | | |

ตารางที่ 5.10 แสดงกรณีทดสอบการจำลองการทำงานที่ผู้ออกแบบสามารถเห็นการเปลี่ยนแปลงข้อมูลนำเข้าและผลลัพธ์ผ่านทางส่วนต่อประสานกับผู้ใช้

| หมายเลขกรณีทดสอบ | TC3 | |
|---|---|---|
| ชื่อ | ทดสอบการเปลี่ยนแปลงการแสดงผลข้อมูลบนหน้าจอเครื่องมือ | |
| วัตถุประสงค์ | เพื่อทดสอบการแสดงการเปลี่ยนแปลงของข้อมูลตัวแปรนำเข้าและตัวแปรผลลัพธ์ขณะที่ผู้ออกแบบทดสอบจำลองการทำงาน | |
| คำอธิบาย | แสดงการเปลี่ยนแปลงของข้อมูลเมื่อทดสอบจำลองการทำงาน | |
| เงื่อนไขในการทดสอบ | ผู้ออกแบบเป็นผู้กำหนดข้อมูลตัวแปรนำเข้าเพื่อจำลองการทำงาน | |
| ผลลัพธ์ที่คาดว่าจะได้รับ | แสดงการเปลี่ยนแปลงของข้อมูลเมื่อทดสอบจำลองการทำงาน | |
| ผลลัพธ์ที่เกิดขึ้นจริง | แสดงการเปลี่ยนแปลงของข้อมูลเมื่อทดสอบจำลองการทำงาน | |
| ผลการทดสอบ | ☒ ผ่าน | ☐ ไม่ผ่าน |
| หมายเหตุ | | |

ตารางที่ 5.11 แสดงกรณีทดสอบการกำหนดจำนวนตัวแปรนำเข้าและตัวแปรผลลัพธ์สูงสุด 20 จำนวนเพื่อการทดสอบการจำลองการทำงาน

| หมายเลขกรณีทดสอบ | TC4 | |
|---|---|---|
| ชื่อ | ทดสอบการจำลองการทำงานกับจำนวนข้อมูล 20 จำนวน | |
| วัตถุประสงค์ | เพื่อทดสอบการจำลองการทำงานของเครื่องมือเมื่อกำหนดจำนวนข้อมูลตัวแปรนำเข้าและตัวแปรผลลัพธ์สูงสุด 20จำนวน | |
| คำอธิบาย | แสดงการจำลองการทำงานของเครื่องมือกับข้อมูล 20จำนวน | |
| เงื่อนไขในการทดสอบ | - | |
| ผลลัพธ์ที่คาดว่าจะได้รับ | เครื่องมือสามารถแสดงผลการจำลองการทำงานได้ถูกต้องเมื่อกำหนดจำนวนตัวแปรนำเข้าและตัวแปรผลลัพธ์สูงสุด 20จำนวน | |
| ผลลัพธ์ที่เกิดขึ้นจริง | เครื่องมือสามารถแสดงผลการจำลองการทำงานได้ถูกต้องเมื่อกำหนดจำนวนตัวแปรนำเข้าและตัวแปรผลลัพธ์สูงสุด 20จำนวน | |
| ผลการทดสอบ | ☒ ผ่าน | ☐ ไม่ผ่าน |
| หมายเหตุ | | |

ตารางที่ 5.12 แสดงกรณีทดสอบการออกแบบสร้างวงจรขึ้นใหม่และการนำวงจรที่มีอยู่มาประกอบร่วมเข้าไปกับวงจรที่ออกแบบใหม่

| หมายเลขกรณีทดสอบ | TC5 | |
|---|---|---|
| ชื่อ | ทดสอบการออกแบบสร้างวงจรใหม่และนำวงจรที่มีมาประกอบร่วม | |
| วัตถุประสงค์ | เพื่อทดสอบการออกแบบสร้างวงจรขึ้นใหม่โดยสามารถนำวงจรที่มีอยู่ก่อนแล้วมาประกอบร่วมเข้าไปได้ | |
| คำอธิบาย | 1. แสดงการออกแบบสร้างวงจรขึ้นมาใหม่ได้ในเครื่องมือ”<br>2. แสดงการนำวงจรที่มีอยู่แล้วมาประกอบร่วมเข้าไปกับวงจรใหม่ | |
| เงื่อนไขในการทดสอบ | - | |
| ผลลัพธ์ที่คาดว่าจะได้รับ | เครื่องมือสามารถสร้างวงจรได้ใหม่และนำวงจรเดิมมาประกอบร่วมได้ | |
| ผลลัพธ์ที่เกิดขึ้นจริง | เครื่องมือสามารถสร้างวงจรได้ใหม่และนำวงจรเดิมมาประกอบร่วมได้ | |
| ผลการทดสอบ | ☒ ผ่าน | ☐ ไม่ผ่าน |
| หมายเหตุ | | |

ตารางที่ 5.13 แสดงกรณีทดสอบการบันทึกวงจรที่ออกแบบในรูปแบบแฟ้มเอกสารเอ็กซ์เอ็มแอลและสามารถเรียกกลับมาใช้งานได้

| หมายเลขกรณีทดสอบ | TC6 | |
|---|---|---|
| ชื่อ | ทดสอบการบันทึกวงจรในรูปแบบเอ็กซ์เอ็มแอลและเรียกมาใช้งานได้ | |
| วัตถุประสงค์ | เพื่อทดสอบการบันทึกวงจรที่ออกแบบในรูปแบบแฟ้มเอกสารเอ็กซ์เอ็มแอลโดยสามารถเรียกวงจรกลับมาใช้ประกอบร่วมในวงจรใหม่ได้ | |
| คำอธิบาย | 3. แสดงการบันทึกวงจรที่ออกแบบในรูปแบบเอ็กซ์เอ็มแอล<br>4. แสดงการนำวงจรในรูปแบบเอ็กซ์เอ็มแอลนั้นมาใช้ร่วมในวงจรใหม่ | |
| เงื่อนไขในการทดสอบ | - | |
| ผลลัพธ์ที่คาดว่าจะได้รับ | แสดงการบันทึกวงจรในรูปแบบเอ็กซ์เอ็มแอลและการนำกลับมาใช้ | |
| ผลลัพธ์ที่เกิดขึ้นจริง | แสดงการบันทึกวงจรในรูปแบบเอ็กซ์เอ็มแอลและการนำกลับมาใช้ | |
| ผลการทดสอบ | ☒ ผ่าน | ☐ ไม่ผ่าน |
| หมายเหตุ | | |

ตารางที่ 5.14 แสดงกรณีทดสอบการออกแบบวงจรโดยการกำหนดด้วยสมการบูลีน

| หมายเลขกรณีทดสอบ | TC7 | |
|---|---|---|
| ชื่อ | ทดสอบการออกแบบวงจรจากสมการบูลีน | |
| วัตถุประสงค์ | เพื่อทดสอบการออกแบบวงจรที่ผู้ออกแบบสามารถกำหนดได้ด้วยสมการบูลีน | |
| คำอธิบาย | 1. แสดงการกำหนดค่าสมการบูลีน<br>2. แสดงวงจรที่สร้างจากสมการบูลีนที่กำหนด | |
| เงื่อนไขในการทดสอบ | - | |
| ผลลัพธ์ที่คาดว่าจะได้รับ | แสดงการออกแบบวงจรที่ได้จากการกำหนดสมการบูลีน | |
| ผลลัพธ์ที่เกิดขึ้นจริง | แสดงการออกแบบวงจรที่ได้จากการกำหนดสมการบูลีน | |
| ผลการทดสอบ | ☒ ผ่าน | ☐ ไม่ผ่าน |
| หมายเหตุ | | |

ตารางที่ 5.15 แสดงกรณีทดสอบการกำหนดค่าตัวแปรนำเข้าด้วยค่าตัวเลขฐานสิบหรือฐานสองและสามารถเลือกชนิดตัวแปรนำเข้าเป็นชนิดจำนวนซ้ำซ้อนได้

| หมายเลขกรณีทดสอบ | TC8 | |
|---|---|---|
| ชื่อ | ทดสอบการกำหนดค่าตัวแปรนำเข้าด้วยตัวเลขฐานสิบหรือตัวเลขฐานสองและกำหนดเป็นชนิดจำนวนซ้ำซ้อน | |
| วัตถุประสงค์ | เพื่อทดสอบการกำหนดค่าให้กับตัวแปรนำเข้าได้สองรูปแบบคือ ตัวเลขฐานสิบหรือตัวเลขฐานสอง และสามารถเลือกชนิดตัวแปรนำเข้าเป็นชนิดจำนวนซ้ำซ้อนได้ | |
| คำอธิบาย | 1. แสดงการกำหนดค่าตัวแปรนำเข้าได้ทั้งเลขฐานสิบและฐานสอง<br>2. แสดงการกำหนดชนิดตัวแปรนำเข้า**เป็นชนิดจำนวนซ้ำซ้อน** | |
| เงื่อนไขในการทดสอบ | - | |
| ผลลัพธ์ที่คาดว่าจะได้รับ | แสดงการทำงานของเครื่องมือได้ถูกต้องตามกรณีทดสอบ | |
| ผลลัพธ์ที่เกิดขึ้นจริง | แสดงการทำงานของเครื่องมือได้ถูกต้องตามกรณีทดสอบ | |
| ผลการทดสอบ | ☒ ผ่าน | ☐ ไม่ผ่าน |
| หมายเหตุ | | |

ตารางที่ 5.16 แสดงกรณีทดสอบการทำงานเครื่องมือตามอัลกอริทึมการคำนวณแบบเชื่อมตรง

| หมายเลขกรณีทดสอบ | TC9 | |
|---|---|---|
| ชื่อ | ทดสอบการทำงานตามอัลกอริทึมการคำนวณแบบเชื่อมตรง | |
| วัตถุประสงค์ | เพื่อทดสอบการทำงานของเครื่องมือที่พัฒนาขึ้นกับการจำลองการทำงานตามอัลกอริทึมการคำนวณแบบเชื่อมตรง | |
| คำอธิบาย | แสดงการจำลองการทำงานของเครื่องมือ | |
| เงื่อนไขในการทดสอบ | - | |
| ผลลัพธ์ที่คาดว่าจะได้รับ | แสดงการจำลองการทำงานตามอัลกอริทึมการคำนวณแบบเชื่อมตรง | |
| ผลลัพธ์ที่เกิดขึ้นจริง | แสดงการจำลองการทำงานตามอัลกอริทึมการคำนวณแบบเชื่อมตรง | |
| ผลการทดสอบ | ☒ ผ่าน | ☐ ไม่ผ่าน |
| หมายเหตุ | | |

2) ทดสอบกันส่วนการวิเคราะห์องค์ประกอบของเครื่องมือ

ตารางที่ 5.17 แสดงกรณีทดสอบการทำงานของอุปกรณ์หรือวงจรที่มีอุปกรณ์เกตดีเลย์เป็นส่วนประกอบอยู่ภายในอุปกรณ์หรือวงจรด้วย

| หมายเลขกรณีทดสอบ | TC10 | |
|---|---|---|
| ชื่อ | ทดสอบการทำงานของอุปกรณ์หรือวงจรที่มีอุปกรณ์เกตดีเลย์เป็นส่วนประกอบ | |
| วัตถุประสงค์ | เพื่อตรวจสอบค่าผลลัพธ์และจำนวนคล็อกของอุปกรณ์หรือวงจรที่มีอุปกรณ์เกตดีเลย์เป็นส่วนประกอบอยู่ภายในอุปกรณ์หรือวงจร | |
| คำอธิบาย | แสดงการทำงานอุปกรณ์หรือวงจรที่มีอุปกรณ์เกตดีเลย์เป็นส่วนประกอบอยู่ภายในอุปกรณ์หรือวงจร | |
| เงื่อนไขในการทดสอบ | - | |
| ผลลัพธ์ที่คาดว่าจะได้รับ | แสดงค่าผลลัพธ์และจำนวนคล็อกได้อย่างถูกต้อง | |
| ผลลัพธ์ที่เกิดขึ้นจริง | แสดงค่าผลลัพธ์และจำนวนคล็อกได้อย่างถูกต้อง | |
| ผลการทดสอบ | ☒ ผ่าน | ☐ ไม่ผ่าน |
| หมายเหตุ | | |

ตารางที่ 5.18 แสดงกรณีทดสอบการนำอุปกรณ์หรือวงจรที่มีอุปกรณ์เกตดีเลย์ประกอบอยู่ภายในไปใช้ประกอบร่วมกับอุปกรณ์หรือวงจรที่สร้างใหม่

| หมายเลขกรณีทดสอบ | TC11 | |
|---|---|---|
| ชื่อ | ทดสอบการนำอุปกรณ์หรือวงจรที่มีอุปกรณ์เกตดีเลย์ประกอบอยู่ภายในไปใช้ในการออกแบบสร้างอุปกรณ์หรือวงจรใหม่ | |
| วัตถุประสงค์ | เพื่อตรวจสอบค่าผลลัพธ์และจำนวนคล็อกของอุปกรณ์หรือวงจรที่มีอุปกรณ์เกตดีเลย์ประกอบอยู่ภายในเมื่อถูกนำไปใช้ประกอบร่วมเข้าไปในอุปกรณ์หรือวงจรที่สร้างใหม่ | |
| คำอธิบาย | แสดงการนำอุปกรณ์หรือวงจรที่มีอุปกรณ์เกตดีเลย์ประกอบอยู่ภายในไปใช้ประกอบในอุปกรณ์หรือวงจรที่สร้างใหม่ | |
| เงื่อนไขในการทดสอบ | - | |
| ผลลัพธ์ที่คาดว่าจะได้รับ | แสดงค่าผลลัพธ์และจำนวนคล็อกได้อย่างถูกต้อง | |
| ผลลัพธ์ที่เกิดขึ้นจริง | แสดงค่าผลลัพธ์และจำนวนคล็อกได้อย่างถูกต้อง | |
| ผลการทดสอบ | ☒ ผ่าน | ☐ ไม่ผ่าน |
| หมายเหตุ | | |

ตารางที่ 5.19 แสดงกรณีทดสอบการทำงานของอุปกรณ์เกตพื้นฐานกับตารางค่าความจริง

| หมายเลขกรณีทดสอบ | TC12 | |
|---|---|---|
| ชื่อ | ทดสอบการทำงานของอุปกรณ์เกตพื้นฐานกับตารางค่าความจริง | |
| วัตถุประสงค์ | เพื่อตรวจสอบค่าผลลัพธ์ที่ได้กับตารางค่าความจริงของอุปกรณ์นั้น ๆ | |
| คำอธิบาย | แสดงการจำลองการทำงานวงจรที่ประกอบด้วยอุปกรณ์เกตพื้นฐาน | |
| เงื่อนไขในการทดสอบ | - | |
| ผลลัพธ์ที่คาดว่าจะได้รับ | แสดงค่าผลลัพธ์ได้ถูกต้องตามค่าในตารางค่าความจริงของอุปกรณ์ | |
| ผลลัพธ์ที่เกิดขึ้นจริง | แสดงค่าผลลัพธ์ได้ถูกต้องตามค่าในตารางค่าความจริงของอุปกรณ์ | |
| ผลการทดสอบ | ☒ ผ่าน | ☐ ไม่ผ่าน |
| หมายเหตุ | | |

ตารางที่ 5.20 แสดงกรณีทดสอบการทำงานของอุปกรณ์ UDP ที่กำหนดค่าความหน่วงและตารางค่าความจริงเอง

| หมายเลขกรณีทดสอบ | TC13 | |
|---|---|---|
| ชื่อ | ทดสอบการทำงานของอุปกรณ์ UDP ที่กำหนดค่าความหน่วงและตารางค่าความจริงเอง | |
| วัตถุประสงค์ | เพื่อทดสอบการทำงานของอุปกรณ์ UDP ที่ผู้ออกแบบสามารถกำหนดค่าความหน่วงและค่าในตารางค่าความจริงให้กับตัวอุปกรณ์ได้เอง | |
| คำอธิบาย | 1. แสดงการทำงานของอุปกรณ์ UDP ที่ผู้ออกแบบสามารถกำหนดค่าความหน่วงและค่าผลลัพธ์ในตารางค่าความจริงให้กับตัวอุปกรณ์ได้เอง<br>2. แสดงการแก้ไขค่าในตารางค่าความจริงได้แต่ต้องตั้งชื่ออุปกรณ์ UDP นี้ ด้วยชื่อใหม่ | |
| เงื่อนไขในการทดสอบ | - | |
| ผลลัพธ์ที่คาดว่าจะได้รับ | แสดงค่าผลลัพธ์และจำนวนคล็อกได้อย่างถูกต้องและสามารถแก้ไขค่าในตารางค่าความจริงได้ | |
| ผลลัพธ์ที่เกิดขึ้นจริง | แสดงค่าผลลัพธ์และจำนวนคล็อกได้อย่างถูกต้องและสามารถแก้ไขค่าในตารางค่าความจริงได้ | |
| ผลการทดสอบ | ☒ ผ่าน | ☐ ไม่ผ่าน |
| หมายเหตุ | | |

ตารางที่ 5.21 แสดงกรณีทดสอบการทำงานของอุปกรณ์ UDE ที่สร้างจากสมการทางคณิตศาสตร์

| หมายเลขกรณีทดสอบ | TC14 |
|---|---|
| ชื่อ | ทดสอบการทำงานของอุปกรณ์ UDE ที่สร้างจากสมการทางคณิตศาสตร์ |
| วัตถุประสงค์ | เพื่อทดสอบการทำงานของอุปกรณ์ UDE ที่ผู้ออกแบบสามารถสร้างขึ้นจากการกำหนดสมการทางคณิตศาสตร์ |

ตารางที่ 5.21 แสดงกรณีทดสอบการทำงานของอุปกรณ์ UDE ที่สร้างจากสมการทางคณิตศาสตร์ (ต่อ)

| หมายเลขกรณีทดสอบ | TC14 | |
|---|---|---|
| คำอธิบาย | แสดงการทำงานของอุปกรณ์ UDE ที่ผู้ออกแบบสามารถกำหนดด้วยสมการทางคณิตศาสตร์ | |
| เงื่อนไขในการทดสอบ | - | |
| ผลลัพธ์ที่คาดว่าจะได้รับ | แสดงค่าผลลัพธ์ได้อย่างถูกต้องตามกรณีทดสอบ | |
| ผลลัพธ์ที่เกิดขึ้นจริง | แสดงค่าผลลัพธ์ได้อย่างถูกต้องตามกรณีทดสอบ | |
| ผลการทดสอบ | ☒ ผ่าน | ☐ ไม่ผ่าน |
| หมายเหตุ | | |

5.2.3 สรุปผลการทดสอบ

สามารถสรุปผลการทดสอบจากข้างต้น14 กรณีตามหมายเลขกรณีทดสอบได้ดังนี้

ตารางที่ 5.22 แสดงสรุปผลการทดสอบในแต่ละกรณี

| ลำดับ | หมายเลขกรณีทดสอบ | ชื่อ | ผ่าน | ไม่ผ่าน |
|---|---|---|---|---|
| 1 | TC1 | ทดสอบการเรียกใช้งานในรูปแบบของแอพเพล็ท | ☒ | ☐ |
| 2 | TC2 | ทดสอบการบันทึกอุปกรณ์หรือวงจรที่ออกแบบบนเครื่องคอมพิวเตอร์ลูกข่าย | ☒ | ☐ |
| 3 | TC3 | ทดสอบการเปลี่ยนแปลงการแสดงผลข้อมูลบนหน้าจอเครื่องมือ | ☒ | ☐ |
| 4 | TC4 | ทดสอบการจำลองการทำงานกับจำนวนข้อมูล 20 จำนวน | ☒ | ☐ |
| 5 | TC5 | ทดสอบการออกแบบสร้างวงจรใหม่และนำวงจรที่มีมาประกอบร่วม | ☒ | ☐ |
| 6 | TC6 | ทดสอบการบันทึกวงจรในรูปแบบเอ็กซ์เอ็มแอลและเรียกมาใช้งานได้ | ☒ | ☐ |

ตารางที่ 5.22 แสดงสรุปผลการทดสอบของแต่ละกรณี (ต่อ)

| ลำดับ | หมายเลขกรณีทดสอบ | ชื่อ | ผ่าน | ไม่ผ่าน |
|---|---|---|---|---|
| 7 | TC7 | ทดสอบการออกแบบวงจรจากสมการบูลีน | ☒ | ☐ |
| 8 | TC8 | ทดสอบการกำหนดค่าตัวแปรนำเข้าด้วยตัวเลขฐานสิบหรือตัวเลขฐานสองและกำหนดเป็นชนิดจำนวนซ้ำซ้อน | ☒ | ☐ |
| 9 | TC9 | ทดสอบการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรง | ☒ | ☐ |
| 10 | TC10 | ทดสอบการทำงานของอุปกรณ์หรือวงจรที่มีอุปกรณ์เกตดีเลย์เป็นส่วนประกอบ | ☒ | ☐ |
| 11 | TC11 | ทดสอบการนำอุปกรณ์หรือวงจรที่มีอุปกรณ์เกตดีเลย์ประกอบอยู่ภายในไปใช้ในการออกแบบสร้างอุปกรณ์หรือวงจรใหม่ | ☒ | ☐ |
| 12 | TC12 | ทดสอบการทำงานของอุปกรณ์เกตพื้นฐานกับตารางค่าความจริง | ☒ | ☐ |
| 13 | TC13 | ทดสอบการทำงานของอุปกรณ์ UDP ที่กำหนดค่าความหน่วงและตารางค่าความจริงเอง | ☒ | ☐ |
| 14 | TC14 | ทดสอบการทำงานของอุปกรณ์ UDE ที่สร้างจากสมการทางคณิตศาสตร์ | ☒ | ☐ |

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

# บทที่ 6

# บทสรุปและข้อเสนอแนะ

## 6.1 บทสรุป

วิทยานิพนธ์นี้เป็นการออกแบบและพัฒนาเครื่องมือสำหรับจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรงในรูปของวงจรย่อยต่าง ๆ ประกอบเข้าด้วยกันและกำหนดการทำงานได้ด้วยตารางค่าความจริง เพื่อประโยชน์ในการศึกษาและตรวจสอบความถูกต้องในการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรง โดยพัฒนาเครื่องมือด้วยภาษาจาวาให้ใช้งานได้ในรูปแบบของแอพเพล็ทและจัดเก็บอุปกรณ์หรือวงจรที่ออกแบบในรูปแบบของแฟ้มเอกสารเอ็กซ์เอ็มแอล

ผู้วิจัยได้พัฒนาเครื่องมือให้ทำงานบนสภาพแวดล้อมแบบกราฟิกเพื่อให้ผู้ออกแบบสามารถจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรงได้สะดวก ดังนั้นการที่เครื่องมือสามารถแสดงผลการจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรงผ่านทางส่วนต่อประสานผู้ใช้ได้โดยตรงและผู้ออกแบบยังสามารถเลือกรูปแบบการแสดงผลการจำลองการทำงานได้เองผ่านทางส่วนต่อประสานผู้ใช้ย่อมเป็นสิ่งสำคัญที่ทำให้นักออกแบบสามารถเข้าใจและตรวจสอบความถูกต้องได้ง่ายยิ่งขึ้น

สำหรับการวิเคราะห์และการออกแบบเครื่องมือในวิทยานิพนธ์ฉบับนี้ ผู้วิจัยได้ใช้ยูเอ็มแอล ซึ่งเป็นโมเดลที่จะทำให้สามารถระบุถึงโครงสร้างและพฤติกรรมของระบบงานที่พัฒนาได้ มาเป็นเครื่องมือในการวิเคราะห์และออกแบบเครื่องมือ โดยได้ทำการแบ่งชุดของคลาสออกเป็น 6 แพ็กเกจด้วยกัน ผู้วิจัยได้พัฒนาเครื่องมือตามที่ได้ออกแบบและทำการทดสอบความสามารถของเครื่องมือในการออกแบบสร้างอุปกรณ์หรือวงจรเพื่อทดสอบจำลองการทำงาน ผลการทดสอบพบว่า เครื่องมือนี้สามารถทำงานได้ถูกต้องตามที่ได้ออกแบบไว้

## 6.2 ขอบเขตและข้อจำกัดของเครื่องมือ

1) อุปกรณ์เกตพื้นฐานจำพวกอุปกรณ์เกต AND, NAND, OR, NOR, XOR และ XNOR ที่เครื่องมือสร้างเตรียมไว้นั้นรองรับจำนวนตัวแปรนำเข้าได้สูงสุด 5 จำนวนเท่านั้น

2) การสร้างอุปกรณ์ UDP สามารถกำหนดค่าจำนวนตัวแปรนำเข้าได้สูงสุด 5 จำนวน ตัวแปรผลลัพธ์ได้สูงสุด 1 จำนวน และกำหนดค่าความหน่วงให้กับตัวอุปกรณ์เองได้สูงสุด 10 คล็อกเท่านั้น

3) การทดสอบจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรงนั้น ผู้ออกแบบสามารถทดสอบได้โดยการป้อนค่าตัวเลขให้กับตัวแปรนำเข้าชนิดต่าง ๆ ได้ 2 รูปแบบเท่านั้น คือ ค่าตัวเลขฐานสิบหรือค่าตัวเลขฐานสอง

4) การเปลี่ยนแปลงของค่าที่ตัวแปรนำเข้าอาจทำให้เกิดการเปลี่ยนแปลงค่าที่ตัวแปรผลลัพธ์ในช่วงเวลาถัดไปขึ้นอยู่ที่ค่าความหน่วงของอุปกรณ์หรือวงจรนั้น

5) การจำลองการทำงานในขณะนี้จะคิดเพียงเวลาหน่วงที่เกิดจากตัวอุปกรณ์เท่านั้น ไม่คิดถึงเวลาหน่วงในสาย ไม่คิดเวลาหน่วงที่ตัวแปรนำเข้าของอุปกรณ์

6) เครื่องมือพัฒนาด้วยภาษาจาวาสามารถทำงานได้ในระบบที่มี Java Virtual Machine หรือเว็บบราวเซอร์ที่สนับสนุน Java Applet (ต้องเป็น Java รุ่น 1.4 ขึ้นไปเท่านั้น) ที่สามารถแสดงผลแบบกราฟิกได้

## 6.3 ข้อเสนอแนะ

ในการวิจัยครั้งนี้ผู้วิจัยได้ทำตามขอบเขตที่กำหนดไว้หากต้องการนำงานวิจัยนี้ไปพัฒนาต่อผู้วิจัยมีข้อเสนอแนะในการปรับปรุง ดังต่อไปนี้

1) อุปกรณ์พื้นฐานที่มีให้เลือกใช้ในเครื่องมือยังมีอยู่จำกัด ควรพัฒนาอุปกรณ์ที่มีความซับซ้อนกว่าอุปกรณ์เกตพื้นฐานต่าง ๆ เช่น RAM, ROM หรือ D FLIP-FLOP เพิ่มเติมเข้าไปให้มากขึ้น

2) ควรเพิ่มเติมอุปกรณ์ประเภทวงจรการคำนวณที่มีความซับซ้อนมากยิ่งขึ้น เช่น วงจรการบวก วงจรการลบ หรือวงจรการคูณ เพื่อให้ผู้ออกแบบสามารถเลือกมาทดสอบจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรงหรือนำไปใช้ในการออกแบบสร้างวงจรใหม่ได้

# รายการอ้างอิง


[1] *DLSim 2.2.* (Online). Available from: http://sourceforge.net/projects/dlsim/: 2002.

[2] *LogiSim 1.09c.* (Online). Available from: http://www.cburch.com/logisim/: 2001.

[3] A. Avizienis. Signed-digit number representation for fast parallel arithmetic. *IRE Transaction on electronic computer,* vol. 10, 389-400, 1961.

[4] กษมา เรืองรอง, กันธิดา พิชิตพรรณ. การคูณทางคณิตศาสตร์แบบออนไลน์. โครงงานวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, 2545.

[5] M.D. Ercegovac. On-line Arithmatic: an Overview. SPIE 495. *Real Time Signal Processing VII,* 86-93, 1984.

[6] K.S. Trivedi, M.D. Ercegovac. On-line algorithms for division and multiplication. *IEEE Transactions on Computers,* vol. 26, 681-687, 1997.

[7] J.M. Muller. Some characterizations of functions computable in on-line arithmetic. *IEEE Transactions on Computers,* vol. 43, 752-755, 1994.

[8] T. Bray, J. Paoli, S. McQueen. Extensible Markup Language (XML) 1.0 Specification. *World Wide Web Consortium* Recommendation, (Online). Available from: http://www.w3.org/TR/REC-xml, 1998.

[9] ขยัน จันทรสถาพร. เรียนลัด XML ฉบับรู้เต็มร้อย! กรุงเทพ: สยามศิลปะการพิมพ์, มกราคม 2544.

[10] ศุภชัย สมพานิช. เข้าใจและใช้ภาษา XML ฉบับโปรแกรมเมอร์. นนทบุรี: อินโฟเพรส, 2544.

[11] A. Miczo. Digital Logic Testing and Simulation. *John Wiley & Sons,* Newyork, Singapore, 1987.


สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

# ภาคผนวก

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

# ภาคผนวก ก

# คู่มือการใช้งานเครื่องมือจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรง

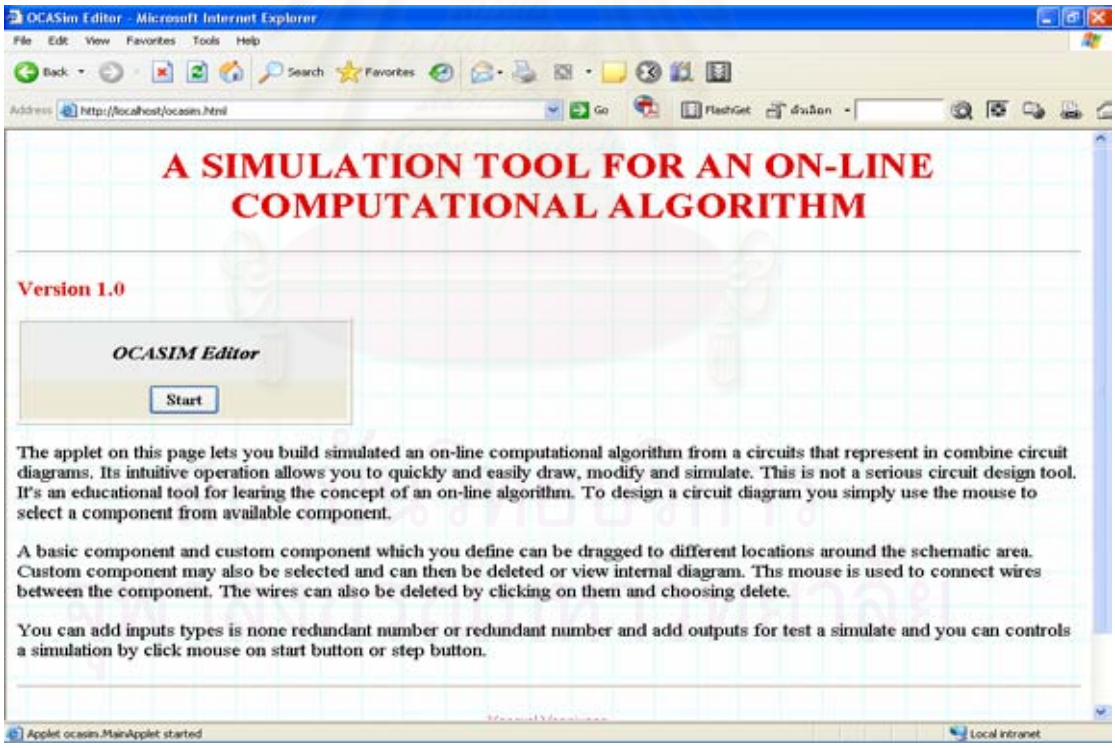
## การใช้งานเครื่องมือจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรง

ผู้ใช้งานสามารถเรียกใช้งานเครื่องมือจำลองการทำงานของอัลกอริทึมการคำนวณแบบเชื่อมตรง โดยทำการพิมพ์ยูอาร์แอล (URL) ดังนี้

ยูอาร์แอล

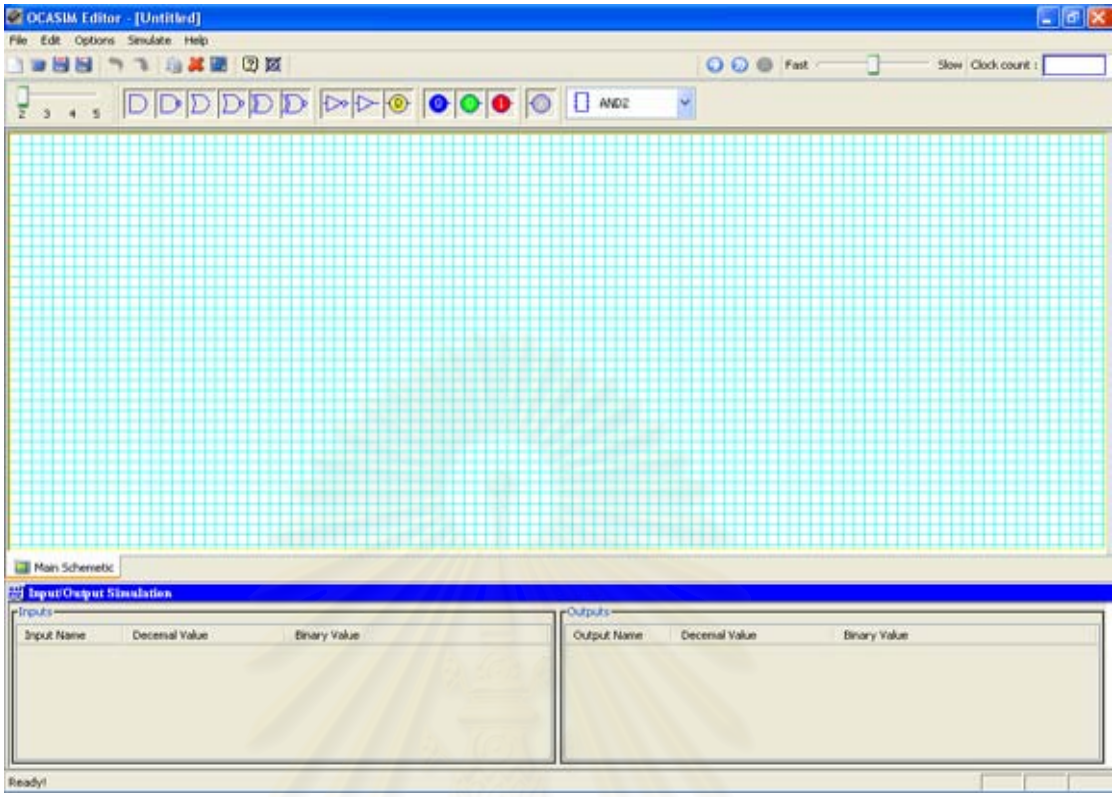
http://localhost/ocasim.html

หลังจากพิมพ์ยูอาร์แอลแล้วจะปรากฏหน้าจอดังรูปที่ ก1




รูปที่ ก1 หน้าจอการทำงานเป็นแอพเพล็ทบนบราวเซอร์

กดปุ่มสตาร์ทบนหน้าจอแอพเพล็ทเพื่อเปิดหน้าจอหลักของเครื่องมือมาทำงานดังรูปที่ ก2




รูปที่ ก2 หน้าจอหลักของเครื่องมือ

<u>การติดตั้งใช้งาน</u>

เนื่องจากเครื่องมือทำงานเป็นลักษณะของเว็บแอพลิเคชั่นที่ใช้งานได้ในรูปแบบของแอพเพล็ท ดังนั้นการใช้งานเครื่องมือจำเป็นต้องมี Java Virtual Machine หรือเว็บบราวเซอร์ที่สนับสนุน Java Applet (ต้องเป็น Java รุ่น 1.4 ขึ้นไปเท่านั้น) ที่สามารถแสดงผลแบบกราฟิกได้

<u>การใช้งานเครื่องมือ</u>

เมื่อเริ่มต้นทำงาน เครื่องมือจะมีลักษณะของส่วนต่อประสานกับผู้ใช้ดังรูปที่ ก2 ซึ่งประกอบไปด้วยส่วนบนสุดของเครื่องมือจะเป็น TitleBar ถัดลงมาจะเป็นแถบเมนู (MenuBar) และแถบเครื่องมือ (ToolBar) ตามลำดับ คำสั่งต่าง ๆ ในแถบเมนูและแถบเครื่องมือจะทำงานเหมือนกัน ผู้ใช้สามารถเลือกใช้ได้ตามความสะดวก ทางด้านขวามือของแถบเครื่องมือจะเป็นแถบเครื่องมือของส่วนจำลองการทำงานของเครื่องมือ ซึ่งผู้ใช้สามารถเลือกกำหนดความเร็วในการจำลองการทำงานและสามารถเลือกทดสอบจำลองการทำงานได้ 2 รูปแบบ คือ กดปุ่มสตาร์ทเพื่อให้การจำลองการทำงานเกิดขึ้นต่อเนื่องจนแล้วเสร็จหรือเลือกให้การจำลองการทำงานเกิดขึ้นทีละขั้นโดยผู้ใช้เป็นผู้ควบคุมการทำงานผ่านส่วนต่อประสาน นอกจากนี้ยังมีส่วนแสดง

จำนวนคล็อกบอกให้ผู้ใช้ทราบเพื่อตรวจสอบความถูกต้องในการจำลองการทำงานด้วย ถัดจากแถบเครื่องมือลงมาจะเป็นทูลบ็อกแทนส่วนคลังอุปกรณ์ซึ่งในส่วนนี้จะประกอบด้วยอุปกรณ์ 3 ประเภทด้วยกัน คือ อุปกรณ์พื้นฐาน อุปกรณ์ประกอบและอุปกรณ์ตัวแปรนำเข้าและตัวแปรผลลัพธ์ ซึ่งผู้ใช้สามารถเลือกอุปกรณ์ต่าง ๆ เหล่านั้นเพื่อมาออกแบบสร้างวงจรขึ้นได้

สำหรับพื้นที่ตรงกลางหน้าต่างจะเป็นส่วนวาดวงจร (Schematic Area) ซึ่งเป็นส่วนที่ผู้ใช้สามารถลากอุปกรณ์จากส่วนคลังอุปกรณ์มาวางและเชื่อมโยงอุปกรณ์เหล่านั้นเข้าเป็นวงจรได้ และยังเป็นส่วนที่แสดงผลลัพธ์ขณะจำลองการทำงานอีกด้วย ในส่วนล่างของหน้าต่างจะเป็นส่วนสำหรับกำหนดค่าตัวแปรนำเข้าเพื่อแสดงค่าของผลลัพธ์ เมื่อผู้ใช้ทำการจำลองการทำงานวงจรที่ออกแบบ ซึ่งผู้ใช้สามารถป้อนค่าเพื่อจำลองการทำงานได้ 2 รูปแบบ คือ ป้อนค่าด้วยตัวเลขฐานสิบหรือตัวเลขฐานสอง โดยเครื่องมือจะทำการแปลงค่าของอีกฐานให้ควบคู่ไป ส่วนล่างสุด คือ แถบบอกสถานะการทำงานของเครื่องมือในขณะนั้น

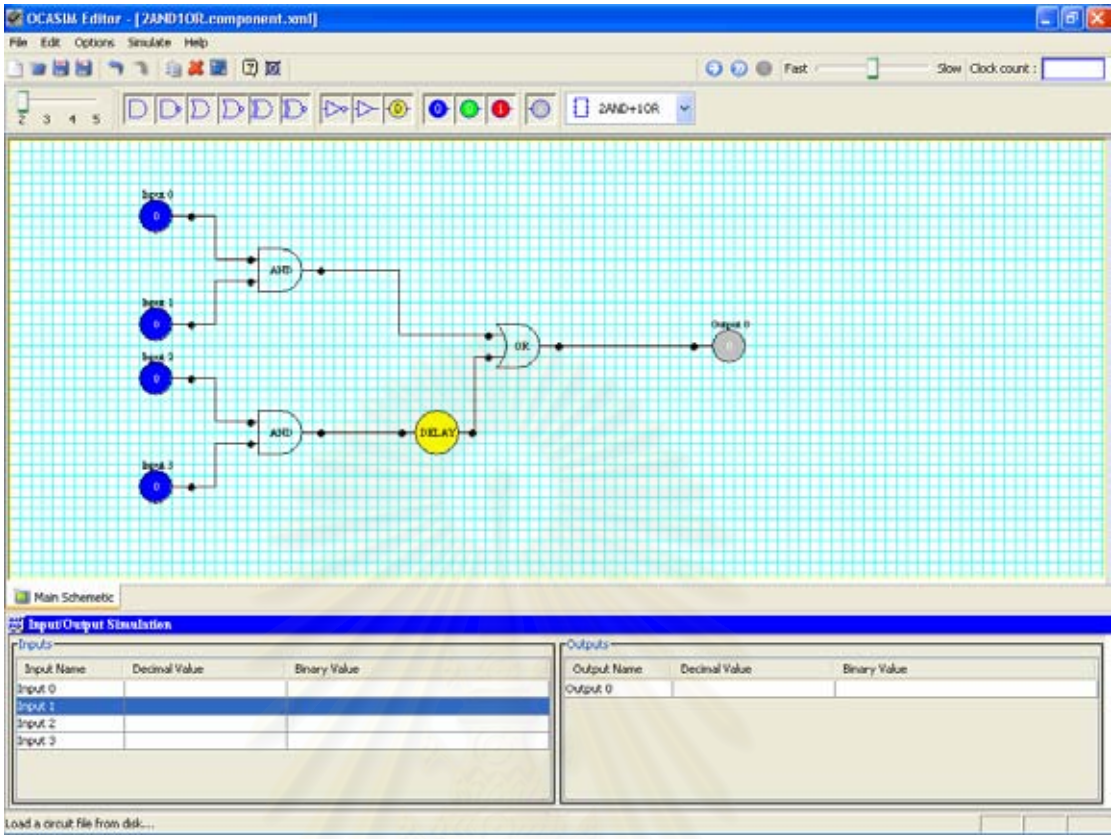
สำหรับการบันทึกเก็บอุปกรณ์หรือวงจรที่ออกแบบนั้นจะเก็บลงส่วนคลังอุปกรณ์ในส่วนของอุปกรณ์ประกอบซึ่งผู้ใช้สามารถนำไปใช้ประกอบในการสร้างวงจรใหม่ได้ด้วย

<u>ตัวอย่างการใช้งาน</u>

การทดลองออกแบบสร้างวงจรที่มีทั้งอุปกรณ์พื้นฐานและอุปกรณ์ประกอบอยู่รวมกันแล้วทดสอบจำลองการทำงาน สามารถทำได้ตามขั้นตอนดังนี้

1) เลือกเมนู File/New เพื่อเริ่มต้นการสร้างอุปกรณหรือวงจร โดยเครื่องมือจะทำการเตือนหากบนส่วนสำหรับสร้างอุปกรณ์หรือวาดวงจรมีการทำงานอยู่ก่อนหน้าแล้วยังไม่ได้ทำการบันทึก หากผู้ใช้ตอบ Yes เครื่องมือจะแสดงหน้าต่างขึ้นมาให้ผู้ใช้ทำการบันทึกเก็บอุปกรณ์หรือวงจรนั้นลงคลังอุปกรณ์ ถ้าตอบ No เครื่องมือจะทำการเคลียร์ทุกอย่างบนหน้าจอเพื่อเริ่มต้นการสร้างอุปกรณ์หรือวงจรใหม่

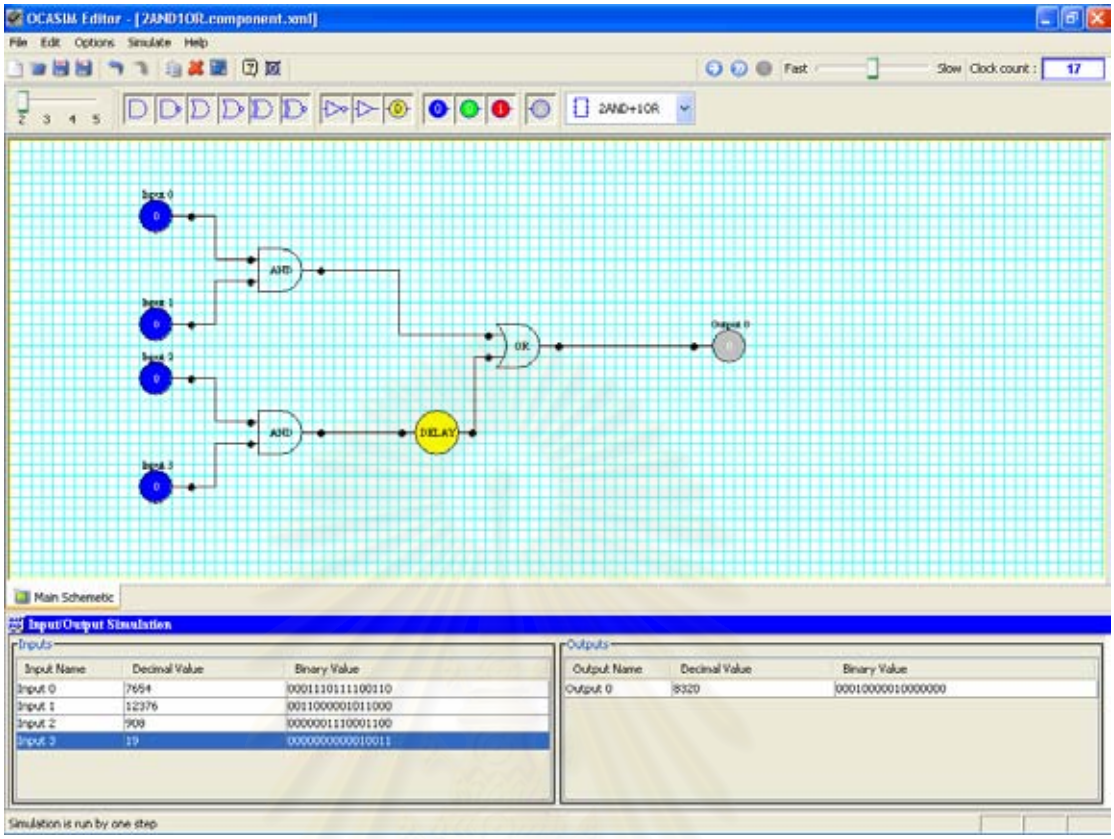
2) จากนั้นผู้ใช้สามารถกดเลือกอุปกรณ์จากคลังอุปกรณ์มาประกอบเข้าด้วยกันเพื่อสร้างเป็นวงจรได้ ดังรูปที่ ก3




รูปที่ ก3 การออกแบบสร้างวงจรด้วยอุปกรณ์เกตพื้นฐานต่าง ๆ

จากรูปอุปกรณ์ที่นำมาประกอบในครั้งนี้เป็นเฉพาะอุปกรณ์เกตพื้นฐานเท่านั้น

3) ป้อนค่าให้กับส่วนตัวแปรนำเข้าทุกตัวเพื่อทดสอบจำลองการทำงานของวงจรที่ออกแบบ จากรูปที่ ก4 เมื่อทำการป้อนค่าตัวแปรนำเข้าทุกตัวแล้วทดสอบจำลองการทำงานด้วยการกดปุ่มสตาร์ทแล้วสังเกตค่าตัวแปรผลลัพธ์ที่เกิดขึ้นจะพบว่าสามารถคำนวณการทำงานได้อย่างถูกต้องโดยเฉพาะอย่างยิ่งวงจรนี้มีอุปกรณ์เกตดีเลย์ประกอบอยู่ด้วย ฉะนั้นเมื่อทำการจำลองการทำงานจำนวนคล็อกที่แสดงออกมาต้องเท่ากับ 17 เพราะค่าอุปกรณ์เกตดีเลย์มีค่าความหน่วงเท่ากับ 1 คล็อก ซึ่งคำนวณได้ถูกต้อง

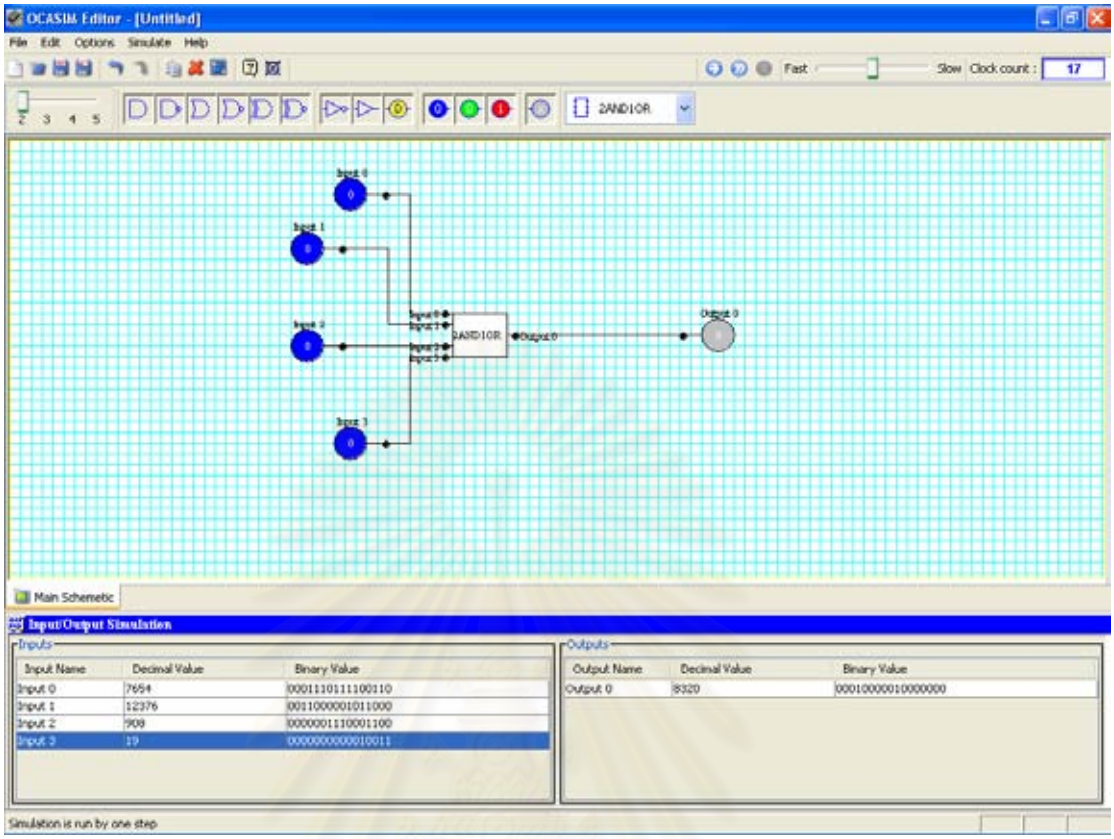



รูปที่ ก4 การป้อนค่าตัวแปรนำเข้าเพื่อทดสอบจำลองการทำงาน

4) เลือกเมนู File/Save เพื่อทำการบันทึกวงจรตัวนี้ลงคลังอุปกรณ์ โดยตั้งชื่อวงจรนี้ว่า 2AND1OR

5) เลือกเมนู File/Load เพื่อทำการโหลดวงจรชื่อ 2AND1OR เพื่อมาทดสอบการทำงานว่าให้ผลลัพธ์ที่ถูกต้องเหมือนเดิมหรือไม่ ปรากฏว่าให้ผลลัพธ์ที่ถูกต้องเหมือนเดิม

6) เลือกเมนู File/New เพื่อเคลียร์ทุกอย่างบนหน้าจอ จากนั้นทดสอบวงจร 2AND1OR หากนำไปประกอบในวงจรใหม่ โดยการลากวงจรย่อย 2AND1OR ที่เก็บอยู่ในส่วนของอุปกรณ์ประกอบมาทดสอบการทำงานก็ให้ค่าผลลัพธ์ที่ถูกต้องเหมือนเดิมดังรูปที่ ก5




รูปที่ ก5 การนำวงจรที่เคยออกแบบไว้แล้วมาทดสอบการทำงาน

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

# ประวัติผู้เขียนวิทยานิพนธ์

นายกังวาล วงษ์จวง เกิดเมื่อวันที่ 30 พฤษภาคม พ.ศ. 2517 ที่จังหวัดปทุมธานี สำเร็จการศึกษาปริญญาวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ จากมหาวิทยาลัยหอการค้าไทย เมื่อปีการศึกษา 2539 และเข้าศึกษาต่อในหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย เมื่อปีการศึกษา 2546

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย