

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

5

โปรแกรมจำลองการทำงานวงจรตรรก

LOGIC SIMULATION PROGRAM



รพ.  
๗๖๓๔๒๒  
๒๕๔๙

เลขหมู่.....  
เลขทะเบียน..... 72225  
วัน,เดือน,ปี..... 12 ต.ย. 2550

b. 117 ๖๕๓๒x  
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมการวัดคุม

ภาควิชาวิศวกรรมการวัดคุม คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# LOGIC SIMULATION PROGRAM



**A THESIS SUBMITTED IN PARTIAL FULFILLMENT  
OF THE REQUIREMENT FOR THE DEGREE OF  
BACHELOR OF ENGINEERING IN INSTRUMENTATION ENGINEERING  
DEPARTMENT OF INSTRUMENTATION ENGINEERING  
FACULTY OF ENGINEERING  
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้  
2006

ภาควิชาวิศวกรรมการวัดคุม  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ใบรับรองปริญญาโท

หัวข้อปริญญาโท โปรแกรมจำลองการทำงานวงจรตรรก  
LOGIC SIMULATION PROGRAM  
นักศึกษาผู้จัดทำ นายณัฐภพ สุรังคพิพรรธน์ รหัสนักศึกษา 46010749  
นางสาวศิรินารถ เลิศสงคราม รหัสนักศึกษา 46010781  
ปริญญา วิศวกรรมศาสตรบัณฑิต  
สาขาวิชา วิศวกรรมการวัดคุม  
ปีการศึกษา 2549

อาจารย์ผู้ควบคุมปริญญาโท	ลายมือชื่อ
รศ.สุพรรณ กุฑพานิชย์	

ภาควิชารับรองแล้ว

  
(รศ.ประภาส อุดคคกิมพันธ์)  
หัวหน้าภาควิชาวิศวกรรมการวัดคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	โปรแกรมจำลองการทำงานวงจรตรรก		
	LOGIC SIMULATION PROGRAM		
นักศึกษาผู้จัดทำ	นายณัฐกพ	สุรงค์พิพรรธน์	รหัสนักศึกษา 46010749
	นางสาวศิรินารด	เลิศสงคราม	รหัสนักศึกษา 46010781
อาจารย์ที่ปรึกษา	รศ.ศุภรณ กุลพานิชย์		
ปีการศึกษา	2549		

### บทคัดย่อ

ในการออกแบบและการทดสอบการทำงานวงจรตรรกนั้น จำเป็นต้องนำอุปกรณ์ต่างๆมาเชื่อมต่อกันและใช้อุปกรณ์เครื่องมือวัดซึ่งเป็นวิธีที่ยุ่งยาก สิ้นเปลืองค่าใช้จ่ายและทำให้เสียเวลามาก ในการทดลองตลอดจนการแก้ไขวงจรทางฮาร์ดแวร์เพื่อให้ได้ผลลัพธ์ที่ต้องการออกมา ดังนั้นปริญญานิพนธ์ฉบับนี้จึงขอเสนอการนำโปรแกรมทางคอมพิวเตอร์ คือโปรแกรม Microsoft Visual C++ มาช่วยในการออกแบบวงจร โดยการออกแบบสร้างโปรแกรมจำลองการทำงานของวงจรตรรก ซึ่งจะทำให้การออกแบบและทดสอบการทำงานของวงจรตรรกใด ๆ ทำได้โดยไม่ต้องใช้อุปกรณ์และเครื่องมือวัดอื่น ๆ เลย โดยผู้ใช้จะต้องเป็นผู้สร้างรูปวงจรและกำหนดฟังก์ชันการทำงานตามที่ต้องการ จากนั้นโปรแกรมจะจำลองการทำงานของรูปวงรดังกล่าว และแสดงผลในรูปของตารางค่าความจริงหรือแผนภูมิเวลา บนจอภาพของคอมพิวเตอร์ นอกจากนี้ผู้ใช้อังสามารถทำการแก้ไขวงจรตรรกและจำลองการทำงานของวงจรใหม่เพื่อให้ได้ผลลัพธ์เป็นไปตามต้องการโดยง่าย

Thesis Title	Logic Simulation Program
Authors	Mr. Nattapob Surangkapiat Miss. Sirinard Lerdsongkram
Thesis Advisor	Assoc. Prof. Suphan Gulpanich
Year	2006

## ABSTRACT

The design and test the operation of logic circuit require some certain equipment and instruments, and of course, it take difficult, use more expense, and longer time for experiment and modify the hardwire circuit until the required result is obtained. This thesis presents one application of computer program, Microsoft Visual C++, for logic circuit aided-design. The logic circuit simulation program is designed and created up so that any logic circuit can be designed and easy to test the result without any equipment or instruments needed. User will only define the schematic diagram and function of required logic circuit, then the program will simulate the operation of the overall circuit and display the result in term of timing diagram or true table through the monitor. Moreover, user can modify the logic circuit using edit mode and re-simulated the operation of modified circuit until the desired result is obtained easily.

## กิตติกรรมประกาศ

ปริญญาานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดีเพราะได้รับความเมตตาจาก รองศาสตราจารย์  
ศุพรรณ กุลพาณิชย์ และ ดร. อรัญญา วลัยรัชต์ ที่ได้ให้คำแนะนำแก่ผู้วิจัยตลอดมา

ขอขอบคุณพระคุณ อาจารย์ภาควิชาวิศวกรรมการวัดคุมทุกท่าน ที่ได้ให้คำแนะนำอันเป็น  
ประโยชน์ต่อการทำปริญญาานิพนธ์ฉบับนี้

และที่ลืมเสียมิได้คือ ขอกราบขอบพระคุณคุณพ่อ คุณแม่ อันเป็นที่รักยิ่ง ที่สนับสนุนและ  
เป็นแรงบันดาลใจในการทำปริญญาานิพนธ์ฉบับนี้

คุณค่าและประโยชน์อันพึงมีจากปริญญาานิพนธ์ฉบับนี้ ผู้วิจัยขอบแต่ผู้มีพระคุณทุกท่าน

คณะผู้จัดทำ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VIII
สารบัญภาพ.....	IX
บทที่ 1 บทนำ.....	1
1.1 ความสำคัญของปริญญานิพนธ์.....	1
1.2 วัตถุประสงค์ของปริญญานิพนธ์.....	1
1.3 ขอบเขตของปริญญานิพนธ์.....	1
1.4 ขั้นตอนการศึกษา.....	2
บทที่ 2 ทฤษฎี.....	3
2.1 หลักการเบื้องต้นของระบบตัวเลข.....	3
2.1.1 บทนำ.....	3
2.1.2 ระบบตัวเลข (Number system).....	3
2.1.3 ระบบเลขฐานสิบ (Decimal number system).....	3
2.1.4 ระบบเลขฐานสอง (Binary number system).....	4
2.1.5 ระบบเลขฐานแปด (Octal number system).....	5
2.1.6 ระบบเลขฐานสิบหก (Hexadecimal number system).....	5
2.1.7 การแปลงเลขฐานสองเป็นเลขฐานสิบ.....	6
2.1.8 การแปลงเลขฐานสิบเป็นเลขฐานสอง.....	7
2.2 หลักการเบื้องต้นของวงจรรตรรก.....	8
2.3 หลักการเบื้องต้นของพีชคณิตบูลีน (Boolean Algebra).....	8
2.3.1 หลักการของเกตแอนด์ (AND gate).....	9
2.3.2 หลักการของเกตออร์ (OR gate).....	10
2.3.3 หลักการของเกตนอท (NOT gate).....	11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
2.3.4 หลักการของเกตแนนด์ (NAND gate).....	12
2.3.5 หลักการของเกตนอร์ (NOR gate).....	14
2.3.6 หลักการของเกตเอ็กซ์คลูซีฟออร์ (Exclusive OR gate).....	15
2.3.7 หลักการของเกตเอ็กซ์คลูซีฟนอร์ (Exclusive NOR gate).....	16
2.3.8 แผนภูมิเวลา (Timing Diagram).....	17
2.3.9 สรุป.....	18
2.4 หลักการเบื้องต้นของภาษาซี (C Language).....	19
2.4.1 กล่าวนำ.....	19
2.4.2 การสั่งงานคอมพิวเตอร์ด้วยภาษาโปรแกรม.....	19
2.4.2.1 ภาษาระดับต่ำ (Low level language).....	20
2.4.2.2 ภาษาระดับสูง (High level language).....	20
2.4.3 จุดเด่นของภาษา C.....	20
2.4.4 สรุป.....	21
<b>บทที่ 3 การสร้างและการออกแบบ.....</b>	<b>22</b>
3.1 ทฤษฎีพื้นฐาน.....	22
3.1.1 ส่วนที่ใช้ติดต่อกับผู้ใช้.....	22
3.1.1.1 สัญลักษณ์และรูปแบบการใช้โปรแกรม.....	22
3.1.2 ส่วนประมวลผล.....	24
3.2 การออกแบบ.....	26
3.2.1 โปรแกรมหลัก.....	28
3.2.2 โปรแกรมสร้างตารางค่าความจริงจากการรับค่าจำนวนอินพุต.....	30
3.2.3 โปรแกรมรับสมการมีวงเล็บ.....	32
3.2.4 โปรแกรมรับสมการไม่มีวงเล็บ.....	34
3.2.5 โปรแกรมวาดรูปวงจรถูก.....	36
3.2.6 โปรแกรมแปลงรูปวงจรถูกเป็นสมการ.....	39
3.2.7 โปรแกรมสร้างสมการ.....	41
3.2.8 โปรแกรมสร้างแผนภูมิเวลา.....	44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
<b>บทที่ 4 การทดลองและผลการทดลอง.....</b>	<b>47</b>
4.1 การทดลอง.....	47
4.2 ผลการทดลอง.....	47
4.2.1 หน้าจอใช้งาน.....	48
4.2.2 เมนู View.....	49
4.2.3 การเคลื่อนย้ายเมนูรูปเกิด และเมนูอินพุต.....	50
4.2.4 เมนู Tool.....	51
4.2.5 การป้อนสมการครerk.....	52
4.2.6 การวาดวงจรรerk.....	53
4.2.7 ตัวอย่างการจำลองการทำงานวงจรรerk.....	54
4.2.7.1 เมื่อมีการป้อนอินพุตที่มีเกตแอนด์.....	54
4.2.7.2 เมื่อมีการป้อนอินพุตที่มีเกตออร์.....	57
4.2.7.3 เมื่อมีการป้อนอินพุตที่มีเกตนอด.....	60
4.2.7.4 เมื่อมีการป้อนอินพุตที่มีเกตแนนด์.....	63
4.2.7.5 เมื่อมีการป้อนอินพุตที่มีเกตนอร์.....	66
4.2.7.6 เมื่อมีการป้อนอินพุตที่มีเกตเอกซ์คลูซีฟออร์.....	69
4.2.7.7 เมื่อมีการป้อนอินพุตที่มีเกตเอกซ์คลูซีฟนอร์.....	72
4.2.7.8 เมื่อมีการนำเกตมาต่อเป็นวงจร.....	75
4.2.7.8.1 ตัวอย่างที่ 1 IP1and(IP2orIP3).....	75
4.2.7.8.2 ตัวอย่างที่ 2 (IP1andIP2)or(IP3andIP4).....	78
4.2.7.8.3 ตัวอย่างที่ 3 (IP1andIP2)or(IP3andIP4).....	81
4.3 สรุปผลการทดลอง.....	84
<b>บทที่ 5 บทสรุปและข้อเสนอแนะ.....</b>	<b>85</b>
5.1 บทสรุป.....	85
5.2 ข้อเสนอแนะ.....	85
<b>บรรณานุกรม.....</b>	<b>87</b>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
ภาคผนวก.....	88
วิธีการใช้โปรแกรม.....	89



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญตาราง

ตารางที่	หน้า
2.1 การเปรียบเทียบเลขฐานต่างๆกับเลขฐานสิบ.....	5
2.2 ตารางค่าความจริงของเกตแอนด์.....	10
2.3 ตารางค่าความจริงของเกตออร์.....	11
2.4 ตารางค่าความจริงของเกตนอท.....	12
2.5 ตารางค่าความจริงของเกตแนนด์.....	13
2.6 ตารางค่าความจริงของเกตนอร์.....	14
2.7 ตารางค่าความจริงของเกตเอ็กซ์คลูซีฟออร์.....	15
2.8 ตารางค่าความจริงของเกตเอ็กซ์คลูซีฟนอร์.....	17
2.9 ตารางสัญลักษณ์ของเกตและสมการพีชคณิต.....	18
3.1 สัญลักษณ์ที่ใช้ในโครงการ.....	22
3.2 ตารางค่าความจริงของจำนวนตัวแปร 3 ตัว.....	24
3.3 ตารางค่าความจริงของการประมวลผลของสมการ IP1 and IP2.....	25
3.4 ตารางค่าความจริงของการประมวลผลของสมการ (IP1andIP2) or IP3.....	26
4.1 ตารางค่าความจริงของการแอนด์.....	56
4.2 ตารางค่าความจริงของการออร์.....	59
4.3 ตารางค่าความจริงของการนอท.....	62
4.4 ตารางค่าความจริงของการแนนด์.....	65
4.5 ตารางค่าความจริงของการนอร์.....	68
4.6 ตารางค่าความจริงของการเอกซ์คลูซีฟออร์.....	71
4.7 ตารางค่าความจริงของการเอกซ์คลูซีฟนอร์.....	74
4.8 ตารางค่าความจริงของสมการ IP1and(IP2orIP3).....	77
4.9 ตารางค่าความจริงของสมการ(IP1andIP2)or(IP3andIP4).....	80
4.10 ตารางค่าความจริงของสมการ $\sim((IP1andIP2)or(IP3andIP4))$ .....	83

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญภาพ

ภาพที่	หน้า
2.1 (ก) Positive Logic.....	8
2.1 (ข) Negative Logic.....	8
2.2 ไฟในห้องจะติดได้ ก็ต่อเมื่อต่อสะพานไฟและมีหลอดไฟอยู่ในกล่องเท่านั้น.....	9
2.3 สัญญาณให้ตื่นนอน.....	9
2.4 วงจรเกตแอนด์ที่ใช้คอนแทค.....	9
2.5 วงจรเกตแอนด์ที่ใช้สวิทช์.....	9
2.6 สัญลักษณ์ของเกตแอนด์ที่มีอินพุต 2 ตัว.....	10
2.7 วงจรเกตออร์ที่ใช้คอนแทค.....	11
2.8 วงจรเกตออร์ที่ใช้สวิทช์.....	11
2.9 สัญลักษณ์ของเกตออร์ที่มีอินพุต 2 ตัว.....	11
2.10 สัญลักษณ์ของเกตนอท.....	12
2.11 วงจรเกตแนนด์ 2 อินพุตที่ใช้คอนแทคและรีเลย์.....	13
2.12 สัญลักษณ์ของเกตแนนด์ที่มีอินพุต 2 ตัว.....	13
2.13 วงจรเกตนอร์ 2 อินพุตที่ใช้คอนแทคและรีเลย์.....	14
2.14 สัญลักษณ์ของเกตนอร์ที่มีอินพุต 2 ตัว.....	14
2.15 (ก) การสร้างเกตเอ็กซ์คลูซีฟออร์.....	15
2.15 (ข) การสร้างเกตเอ็กซ์คลูซีฟออร์.....	15
2.16 สัญลักษณ์ของเกตเอ็กซ์คลูซีฟออร์ที่มีอินพุต 2 ตัว.....	15
2.17 (ก) การสร้างเกตเอ็กซ์คลูซีฟออร์โดยนำเอาค่าคอมพลิเมนต์มาใช้.....	16
2.17 (ข) การสร้างเกตเอ็กซ์คลูซีฟออร์โดยนำเอาค่าคอมพลิเมนต์มาใช้.....	16
2.18 (ก) การสร้างเกตเอ็กซ์คลูซีฟนอร์.....	16
2.18 (ข) การสร้างเกตเอ็กซ์คลูซีฟนอร์.....	17
2.19 สัญลักษณ์ของเกตเอ็กซ์คลูซีฟนอร์ที่มีอินพุต 2 ตัว.....	17
2.20 ตัวอย่างของแผนภูมิเวลาจากสมการ $Y = A \cdot B$ .....	18
2.21 ตัวอย่างของแผนภูมิเวลาจากสมการ $Y = A + B$ .....	18
3.1 IP1 and IP2.....	23
3.2 IP1 and IP2 or IP3.....	23
3.3 (IP1 and IP2) or (IP3 or IP4).....	24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญภาพ (ต่อ)

ภาพที่	หน้า
3.4 แผนผังการทำงานของโปรแกรมหลัก.....	28
3.5 แผนผังการทำงานของโปรแกรมสร้างตารางค่าความจริงจากการรับค่าจำนวนอินพุต.....	30
3.6 แผนผังการทำงานของโปรแกรมรับสมการมีวงเล็บ.....	32
3.7 แผนผังการทำงานของโปรแกรมรับสมการไม่มีวงเล็บ.....	34
3.8 แผนผังการทำงานของโปรแกรมวาดรูปวงจรถูก.....	36
3.9 เกต AND ซึ่งออบเจ็กต์มีสมการ IP1 and IP2.....	37
3.10 พื้นที่ของออบเจ็กต์.....	37
3.11 แผนผังการทำงานของโปรแกรมแปลงรูปวงจรถูกเป็นสมการ.....	39
3.12 แผนผังการทำงานของโปรแกรมสร้างสมการ.....	41
3.13 IP1 and IP2.....	42
3.14 IP1andIP2orIP3.....	42
3.15 (IP1andIP2)or(IP3andIP4).....	43
3.16 แผนผังการทำงานของโปรแกรมสร้างแผนภูมิเวลา.....	44
3.17 ผลของเอาท์พุต line00.....	45
3.18 ผลของเอาท์พุต line10.....	45
3.19 ผลของเอาท์พุต line20.....	46
3.20 ผลของเอาท์พุต line30.....	46
3.21 ผลของเอาท์พุต line40.....	46
4.1 หน้าจอใช้งาน.....	49
4.2 เมนู View.....	50
4.3 การเคลื่อนย้ายเมนู และเมนูอินพุต.....	51
4.4 เมนูใช้งาน.....	52
4.5 ตัวอย่างการป้อนสมการตรรก.....	53
4.6 ตัวอย่างการวาดรูปวงจรถูก.....	54
4.7 ตัวอย่างการจำลองการทำงานของวงจรตรรกของการแอนด์โดยการป้อนสมการ.....	54
4.8 ตัวอย่างการจำลองการทำงานของวงจรตรรกของการแอนด์โดยการป้อนรูปวงจรถูก.....	54
4.9 เอาท์พุตของผลการทดลองในรูปแบบของแผนภูมิเวลา.....	55
4.10 เอาท์พุตของผลการทดลองในรูปแบบของตารางค่าความจริง.....	55

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญญภาพ (ต่อ)

ภาพที่	หน้า
4.11 สัญลักษณ์ของเกตแอนด์ที่ใช้ในการวาดรูปวงจร.....	56
4.12 ตัวอย่างการจำลองการทำงานวงจรตรรกของการออร์โดยการป้อนสมการ.....	57
4.13 ตัวอย่างการจำลองการทำงานวงจรตรรกของการออร์โดยการป้อนรูปวงจร.....	58
4.14 เอาท์พุทของผลการทดลองในรูปแบบของแผนภูมิเวลา.....	58
4.15 เอาท์พุทของผลการทดลองในรูปแบบของตารางค่าความจริง.....	58
4.16 สัญลักษณ์ของเกตออร์ที่ใช้ในการวาดรูปวงจร.....	59
4.17 ตัวอย่างการจำลองการทำงานวงจรตรรกของการนอตโดยการป้อนสมการ.....	60
4.18 ตัวอย่างการจำลองการทำงานวงจรตรรกของการนอตโดยการป้อนรูปวงจร.....	60
4.19 เอาท์พุทของผลการทดลองในรูปแบบของแผนภูมิเวลา.....	61
4.20 เอาท์พุทของผลการทดลองในรูปแบบของตารางค่าความจริง.....	61
4.21 สัญลักษณ์ของเกตนอตที่ใช้ในการวาดรูปวงจร.....	62
4.22 ตัวอย่างการจำลองการทำงานวงจรตรรกของการแนนด์โดยการป้อนสมการ.....	63
4.23 ตัวอย่างการจำลองการทำงานวงจรตรรกของการแนนด์โดยการป้อนรูปวงจร.....	63
4.24 เอาท์พุทของผลการทดลองในรูปแบบของแผนภูมิเวลา.....	64
4.25 เอาท์พุทของผลการทดลองในรูปแบบของตารางค่าความจริง.....	64
4.26 สัญลักษณ์ของเกตแนนด์ที่ใช้ในการวาดรูปวงจร.....	65
4.27 ตัวอย่างการจำลองการทำงานวงจรตรรกของการนอร์โดยการป้อนสมการ.....	66
4.28 ตัวอย่างการจำลองการทำงานวงจรตรรกของการนอร์โดยการป้อนรูปวงจร.....	66
4.29 เอาท์พุทของผลการทดลองในรูปแบบของแผนภูมิเวลา.....	67
4.30 เอาท์พุทของผลการทดลองในรูปแบบของตารางค่าความจริง.....	67
4.31 สัญลักษณ์ของเกตนอร์ที่ใช้ในการวาดรูปวงจร.....	68
4.32 ตัวอย่างการจำลองการทำงานวงจรตรรกของการเอ็กซ์คลูซีฟออร์โดยการป้อนสมการ.....	69
4.33 ตัวอย่างการจำลองการทำงานวงจรตรรกของการเอ็กซ์คลูซีฟออร์โดยการป้อนรูปวงจร.....	69
4.34 เอาท์พุทของผลการทดลองในรูปแบบของแผนภูมิเวลา.....	70
4.35 เอาท์พุทของผลการทดลองในรูปแบบของตารางค่าความจริง.....	70
4.36 สัญลักษณ์ของเกตเอ็กซ์คลูซีฟออร์ที่ใช้ในการวาดรูปวงจร.....	71
4.37 ตัวอย่างการจำลองการทำงานวงจรตรรกของการเอ็กซ์คลูซีฟออร์โดยการป้อนสมการ.....	72
4.38 ตัวอย่างการจำลองการทำงานวงจรตรรกของการเอ็กซ์คลูซีฟออร์โดยการป้อนรูปวงจร.....	72

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญญภาพ (ต่อ)

ภาพที่	หน้า
4.39 เอาท์พุทของผลการทดลองในรูปแบบของแผนภูมิเวลา.....	73
4.40 เอาท์พุทของผลการทดลองในรูปแบบของตารางค่าความจริง.....	74
4.41 สัญลักษณ์ของเกตเอ็กซ์คลูซีฟนอร์ที่ใช้ในการวาดรูปวงจร.....	74
4.42 ตัวอย่างการจำลองการทำงานวงจรรรอกโดยการป้อนสมการ.....	75
4.43 ตัวอย่างการจำลองการทำงานวงจรรรอกโดยการป้อนรูปวงจร.....	75
4.44 เอาท์พุทของผลการทดลองในรูปแบบของแผนภูมิเวลา.....	76
4.45 เอาท์พุทของผลการทดลองในรูปแบบของตารางค่าความจริง.....	76
4.46 ตัวอย่างการจำลองการทำงานวงจรรรอกโดยการป้อนสมการ.....	78
4.47 ตัวอย่างการจำลองการทำงานวงจรรรอกโดยการป้อนรูปวงจร.....	78
4.48 เอาท์พุทของผลการทดลองในรูปแบบของแผนภูมิเวลา.....	79
4.49 เอาท์พุทของผลการทดลองในรูปแบบของตารางค่าความจริง.....	79
4.50 ตัวอย่างการจำลองการทำงานวงจรรรอกโดยการป้อนสมการ.....	81
4.51 ตัวอย่างการจำลองการทำงานวงจรรรอกโดยการป้อนรูปวงจร.....	82
4.52 เอาท์พุทของผลการทดลองในรูปแบบของแผนภูมิเวลา.....	83
4.53 เอาท์พุทของผลการทดลองในรูปแบบของตารางค่าความจริง.....	83

# บทที่ 1

## บทนำ

### 1.1 ความสำคัญของปริศยานิพนธ์

การพัฒนาวงจรตรรกในอดีตที่ผ่านมา เราสามารถตรวจสอบการใช้งานของวงจรได้จากการทดลองกับอุปกรณ์จริงเท่านั้น จึงทำให้การพัฒนาเป็นไปได้อย่างล่าช้าเนื่องจากระหว่างที่ทำการทดสอบหรือทดลองอยู่นั้นอาจจะเกิดความผิดพลาดจากอุปกรณ์การทดลอง เครื่องมือวัด หรือผู้ออกแบบได้ทุกเมื่อ ดังนั้นกว่าจะได้วงจรที่ใช้งานได้จริงจะต้องอาศัยเวลาในการตรวจสอบอยู่นาน ดังนั้นเพื่อเป็นการหลีกเลี่ยงปัญหาเหล่านี้ ปริศยานิพนธ์ฉบับนี้ขอเสนอโปรแกรมจำลองการทำงานวงจรตรรกช่วยในการออกแบบวงจรตรรก โดยผู้ใช้สามารถป้อนสมการตรรกหรือผังภาพวงจรและกำหนดฟังก์ชันการทำงานที่อยู่ใน โปรแกรมให้สอดคล้องกับผังภาพวงจรมานั้นๆลงในโปรแกรมเพื่อประมวลผล จากนั้นนำข้อมูลที่ได้นำมาแสดงผลในรูปแบบของตารางค่าความจริง (True Table) หรือแผนภูมิเวลา (Timing Diagram)

### 1.2 วัตถุประสงค์ของปริศยานิพนธ์

1. สามารถประยุกต์ใช้โปรแกรม Microsoft Visual C++ เพื่อจำลองการทำงานแบบวงจรตรรกได้
2. เพื่อเป็นเครื่องมือในการตรวจสอบว่าเอาต์พุตที่ผู้ใช้ต้องการนั้น ได้ออกแบบไว้แล้ว ถูกต้องหรือไม่ ในกรณีที่มีวงจรยุ่งยากและซับซ้อน เพื่อประหยัดในเรื่องของเวลาและค่าใช้จ่าย
3. เพื่อเป็นแนวทางที่จะสามารถลดจำนวนตัวเกตในวงจรที่มีขนาดใหญ่ โดยการเปรียบเทียบระหว่างเอาต์พุตของสมการตรรกกับเอาต์พุตของรูปวงจรตรรกว่าเหมือนกันหรือไม่

### 1.3 ขอบเขตของปริศยานิพนธ์

1. สามารถประยุกต์ใช้โปรแกรม Microsoft visual C++ เพื่อนำมาเขียนวงจรตรรกได้
2. ขอบเขตของการประมวลผล
  - 2.1 สามารถประมวลผลสมการตรรก ได้
  - 2.2 สามารถประมวลผลวงจรถรก ได้
3. ขอบเขตของตัวฟังก์ชันที่ใช้ได้
  - 3.1 And
  - 3.2 Or
  - 3.3 Not

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 Nand

3.5 Nor

3.6 Exclusive Or

3.7 Exclusive Nor

#### 4. ขอบเขตการแสดงผลเอาต์พุต

4.1 สามารถแสดงผลในรูปตารางค่าความจริงได้

4.2 สามารถแสดงผลในรูปแผนภูมิเวลาได้

### 1.4 ขั้นตอนการศึกษา

ในส่วนของขั้นตอนการศึกษานั้นแบ่งออกเป็น 3 ส่วนหลัก ๆ คือ

ส่วนแรก ทำการวางแผนงานโดยการกำหนดออกมาเป็นรูปแบบของตารางเวลาและหน้าที่แล้วทำการค้นหาข้อมูลต่าง ๆ ที่เกี่ยวข้องกับการทำงานของวงจรรตรรก เป็นต้น

ส่วนที่สอง ทำการเตรียมหาเอกสาร หนังสือ ที่เกี่ยวข้องกับการใช้โปรแกรมภาษา C/C++ และ Microsoft visual C++ เพื่อนำมาเป็นเครื่องมือในการเขียนโปรแกรมจำลองการทำงานของวงจรรตรรก

ส่วนที่สาม เมื่อได้เก็บรวบรวมข้อมูลต่าง ๆ ที่ต้องการได้แล้ว ก็เริ่มทำเขียนโปรแกรมเกี่ยวกับการจำลองการทำงานของวงจรรตรรก

## บทที่ 2

# ทฤษฎีที่ใช้ในการออกแบบ

### 2.1 หลักการเบื้องต้นของระบบตัวเลข

#### 2.1.1 กล่าวนำ

ระบบตัวเลขที่เรารู้จักและใช้กันมากที่สุด คือระบบเลขฐานสิบ (Decimal number System) ซึ่งประกอบด้วยตัวเลข 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 รวม 10 ตัว ซึ่งเราใช้เป็นเครื่องมือหลักในการนับจำนวนทั้งหลายตามความต้องการ เพราะมีสัญลักษณ์ที่ใช้แทนค่าตัวเลขต่างๆมี 10 แบบไม่ซ้ำกัน แต่ละแบบมีเพียง 1 ตำแหน่ง ดังนั้นค่าของตัวเลขที่มีค่ามากกว่า 9 ขึ้นไป ก็จะเป็นจำนวนตัวเลขที่เกิดจากการนำตัวเลข 10 ตัวดังกล่าวมาเรียงประกอบกันขึ้น เช่น ห้าร้อยสิบสอง โดยสามารถเขียนเป็นตัวเลขฐานสิบคือ 512 เป็นต้น แต่ระบบตัวเลขที่ใช้ในคอมพิวเตอร์คือ ระบบเลขฐานสอง (Binary number System) เนื่องจาก หลักการทำงานของเครื่องคอมพิวเตอร์ คืออาศัยการไหลหรือหยุดไหลของสัญญาณในช่วงจังหวะเวลาต่างๆกัน เหมือนกับการปิดเปิดสวิทช์นั่นเอง การทำงานของเครื่องคอมพิวเตอร์ในแต่ละส่วนจึงเป็นแบบ 2 จังหวะตลอดเวลา

#### 2.1.2 ระบบตัวเลข (Number system)

ระบบตัวเลขแต่ละระบบจะมีจำนวนตัวเลขที่ใช้เหมือนกับชื่อของระบบตัวเลขนั้น และมีเลขฐาน (Base) ของจำนวนเลขตามชื่อของมันด้วย เช่น

- ระบบเลขฐานสอง (Binary number system) ประกอบด้วยเลข 2 ตัว คือ 0 และ 1
- ระบบเลขฐานแปด (Octal number system) ประกอบด้วยเลข 8 ตัว คือ 0, 1, 2, 3, 4, 5, 6 และ 7
- ระบบเลขฐานสิบ (Decimal number system) ประกอบด้วยเลข 10 ตัว คือ 0, 1, 2, 3, 4, 5, 6, 7, 8 และ 9
- ระบบเลขฐานสิบหก (Hexadecimal number system) ประกอบด้วยเลข 16 ตัว คือ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E และ F (เมื่อ A = 10, B = 11, C = 12, D = 13, E = 14, F = 15)

#### 2.1.3 ระบบเลขฐานสิบ (Decimal number system)

เราไม่ค่อยพิจารณาถึงค่าที่แท้จริงในแต่ละตำแหน่งของเลขฐานสิบ เช่น 512 คือ 5 อยู่ในหลักร้อย 1 อยู่ในหลักสิบ และ 2 อยู่ในหลักหน่วย โดยที่ตัวเลขในระบบฐานสิบนี้ได้ตั้งแต่ 0-9 ส่วนเลข 10 ซึ่งเป็นชื่อเรียกระบบของเลขฐานชนิดนี้ไม่ได้เป็นเลขที่อยู่ในระบบด้วย ดังนั้น 10 ก็คือ 1 อยู่ในหลักสิบ และ 0 อยู่ในหลักหน่วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

512 เราอาจจะแสดงค่าออกมาได้ดังนี้

$$512 = 5 \times 10^2 + 1 \times 10^1 + 2 \times 10^0$$

= 5 อยู่ในหลักร้อย + 1 อยู่ในหลักสิบ + 2 อยู่ในหลักหน่วย

#### 2.1.4 ระบบเลขฐานสอง (Binary number system)

ระบบดิจิทัลอิเล็กทรอนิกส์ใช้ระบบเลขฐานสอง ซึ่งมี 0 และ 1 เท่านั้นโดยใช้ความต่างศักย์สองระดับ คือสูงและต่ำ แทนด้วย 1 และ 0 ตามลำดับ (1 และ 0 โวลต์)

ในปัจจุบันเราใช้ระบบเลขฐานสิบ ซึ่งมีตัวเลขสิบจาก 0 ถึง 9 ในระบบเลขฐานสองใช้ตัวเลขสองคือเลข 0 และ 1 ในการนับระบบเลขฐานสิบ คอลัมน์แรกจะเป็น 0 และเพิ่มขึ้นจนถึง 9 จะทำให้เต็มหลักพอดีแล้วจะเพิ่มหลักต่อไปโดยบวก 1 เข้าไปที่คอลัมน์ 0 เพิ่มขึ้นจากทางซ้ายไปเรื่อยๆ จนกว่าจะเปลี่ยนหลัก เช่น สิบ (10)

ในการนับเลขฐานสอง หลักแรกเป็น 0 และนับขึ้นเป็น 1 ถ้าแถวแรกเต็มแล้วจะรีเซ็ตและบวก 1 เพิ่มเข้าไปในเลขฐานถัดไปทางซ้ายจาก 0 → 1, 10 ซึ่งแถวแรกจะถูกเติมเต็มอีกครั้งหนึ่งหลังจาก 10 มา 11 ทั้งสองคอลัมน์จะเต็ม แล้วรีเซ็ตทั้งคู่และบวก 1 เข้าไปที่ฐานถัดทางซ้าย หลังจาก 11 มา 100 คอลัมน์แรกจะถูกเติมเต็มอีกครั้งหลังจาก 100 มาเป็น 101, 110, 111, 1000, 1001, 1010, 1011, 1100, 1101 และอื่นๆ การนับระบบเลขฐานสองทำดังนี้

0	
1	คอลัมน์แรกจะเต็ม
1 0	รีเซ็ต และบวก 1 เข้าไปที่คอลัมน์สอง
1 1	สองคอลัมน์แรกจะเต็ม
1 0 0	รีเซ็ตและบวก 1 ในคอลัมน์ที่ 3
1 0 1	
1 1 0	
1 1 1	สามคอลัมน์แรกจะเต็ม
1 0 0 0	รีเซ็ตและบวก 1 ในคอลัมน์ที่ 4
1 0 0 1	
1 0 1 0	
1 0 1 1	
1 1 0 0	
1 1 0 1	
1 1 1 0	
1 1 1 1	สี่คอลัมน์แรกจะเต็ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1 0 0 0 0      รีเซตและบวก1 เข้าไปที่คอลัมน์ที่ 5  
 1 0 0 0 1  
 1 0 0 1 0

คำว่าบิต(Bit) มาจาก Binary digital ซึ่งแต่ละตำแหน่งของเลขฐานสอง เรียกว่า “บิต” เช่น 10110 มีค่าเท่ากับ 5 ตำแหน่งแรกทางขวาสุด เรียกว่าบิตที่มีความสำคัญน้อยที่สุด LSB (Least Significant Bit) และซ้ายมือสุดเรียกว่า บิตที่มีความสำคัญมากที่สุด MSB (Least Significant Bit) และซ้ายมือสุดเรียกว่า บิตที่มีความสำคัญมากที่สุด MSB (Most Significant Bit)

(MSB) → 10110 ← (LSB)

เมื่อใช้ 3 บิตเราสามารถนับ 111 หรือ 7 ถึง 000 ซึ่งมีค่าแตกต่างกัน 8 ค่า

เมื่อกำหนด N บิตจะสามารถนับค่าสูงสุดเท่ากับ  $2^N - 1$  และมีตัวเลขที่แตกต่างกันเท่ากับ

$2^N - 1$  = ค่าสูงสุดที่สามารถนับได้

N = จำนวนบิตของเลขฐานสอง

$2^N$  = ค่าตัวเลขที่นับได้

#### 2.1.5. ระบบเลขฐานแปด (Octal number system)

นิยมใช้มากในการโปรแกรมคอมพิวเตอร์สำหรับงานอุตสาหกรรม เช่น ไมโครคอมพิวเตอร์ เลขฐานแปดใช้หลักการเช่นเดียวกับเลขฐานสิบหรือเลขฐานสอง ต่างกันที่ฐานเป็นแปดและมีสัมประสิทธิ์ของฐานจาก 0 ถึง 7

#### 2.1.6 ระบบเลขฐานสิบหก (Hexadecimal number system)

เลขฐานสิบหกมีระบบคล้ายเลขฐานแปด ขณะเดียวกันการประยุกต์ใช้ก็มีมากมาย เช่น ใช้ในการโปรแกรมคอมพิวเตอร์ หรือใช้ในอุปกรณ์ควบคุมประเภทโปรแกรมเมเบิลคอนโทรลรวมทั้งการโปรแกรมคอมพิวเตอร์ชนิดแผ่นพิมพ์ด้วย เป็นต้น

เลขฐานสิบหกนี้มีการเขียนแทนด้วยตัวเลขจาก 0 ถึง 9 จากนั้นเราเขียนแทนด้วยตัวอักษร A ถึง F

ตารางที่ 2.1 การเปรียบเทียบเลขฐานต่าง ๆ กับเลขฐานสิบ

เลขฐานสิบ Decimal	เลขฐานสอง Binary	เลขฐานแปด Octal	เลขฐานสิบหก Hexadecimal
0	0	0	0
1	1	1	1
2	10	2	2

เลขฐานสิบ Decimal	เลขฐานสอง Binary	เลขฐานแปด Octal	เลขฐานสิบหก Hexadecimal
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

### 2.1.7 การแปลงเลขฐานสองเป็นเลขฐานสิบ

ในระบบเลขฐานสิบ ตัวเลขตัวแรกขวามือสุดเรียกว่า LSB (มีค่าเท่ากับ  $10^0 = 1$ ) แต่ละคอลลัมน์ไปทางซ้ายจะเพิ่มขึ้นทีละ 10 (ระบบฐานสิบ) สามารถแสดงในระบบฐานสองได้ดังนี้  $10^0, 10^1, 10^2, 10^3$  เพิ่มขึ้นไปเรื่อยๆ เช่น 3954 (ฐานสิบ) หมายถึง

$$3 \times 10^3 + 9 \times 10^2 + 5 \times 10^1 + 4 \times 10^0 = 3,000 + 900 + 50 + 4$$

ในระบบเลขฐานสองแต่ละคอลลัมน์จะเพิ่มจากขวา เช่น 1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024 เพิ่มขึ้นไปเรื่อยๆ ในลักษณะที่แสดงในระบบเลขฐานสองของได้ดังนี้  $2^0, 2^1, 2^2, 2^3, 2^4, 2^5, 2^6, 2^7, 2^8, 2^9, 2^{10}$  และอื่นๆระบบเลขฐานสองของ 10110 หมายถึง

$$(1 \times 2^4) + (0 \times 2^3) + (1 \times 2^2) + (1 \times 2^1) + (0 \times 2^0) \\ = 16 + 0 + 4 + 2 + 0$$

การเขียนทำได้โดยห้อยเลขฐานไว้ด้านล่าง ดังนี้

$$10110_2 = 22_{10}$$

### 2.1.8 การแปลงเลขฐานสิบเป็นเลขฐานสอง

วิธีที่ 1 พยายามวางตำแหน่งของฐานสองจนกระทั่งค่าเกินเลขฐานสิบที่จะเปลี่ยนเช่น  $23_{10}$

$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$
32	16	8	4	2	1

ค่า 32 มีค่าเกิน 23 บิตนี้จะมีค่าเป็น 0 หลังจากนั้นเอา 23 ลบด้วย 16 บิตนี้จะมีค่าเป็น 1

[ 0	1	x	x	x	x]
32	16	8	4	2	1
0	1	x	x	x	x

$$23 - 16 = 7$$

ค่า 7 เกินจาก  $23 - 16$  แล้วนำไปเทียบกับหลัก 2 ถ้ามีค่ามากกว่าหลักนี้จะมีค่าเป็น "1" ถ้ามีค่าน้อยกว่า หลักนี้จะมีค่าเป็น "0" นำไปลบกันแล้วทำเช่นนี้เรื่อยไปจนครบทุกหลักที่เหลือ

$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$
1	0	1	x	x	(x = 0,1)
32	16	8	4	2	1
$7 - 4 = 3$					

และใช้ 2 ลบออกจาก 3 ที่คอลัมน์ที่ 2 จะเต็ม 1

0	1	0	1	1	x
32	16	8	4	2	1
$3 - 2 = 1$					

และตำแหน่งคอลัมน์ที่ 1 เป็น 1 และนำไปลบออกจะหมดพอดี เต็ม 1 ที่คอลัมน์ 1

1	1	0	1	1	1
32	16	8	4	2	1

และตำแหน่งคอลัมน์ที่ 1 เป็น 1 และนำไปลบออกจะหมดพอดี เต็ม 1 ที่คอลัมน์ 1

$$1 - 1 = 0$$

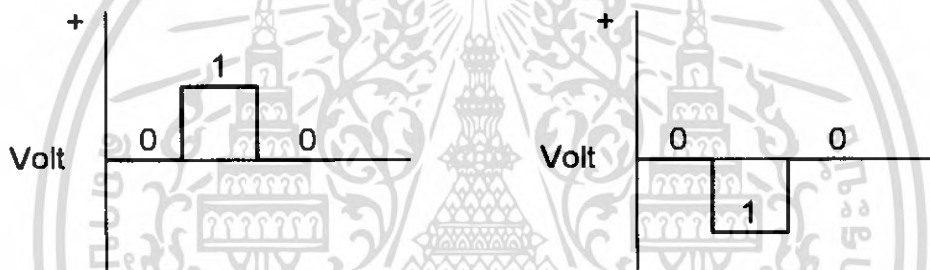
$$23_{10} = 10111_2$$

## 2.2 หลักการเบื้องต้นของวงจรตรรก

วงจรตรรก หมายถึง วงจรไฟฟ้าที่ประกอบด้วยอุปกรณ์อิเล็กทรอนิกส์ หรือระบบรีเลย์ที่มี สัญญาณเพียง 2 ระดับ หรือ 2 สถานะเท่านั้น โดยจะอยู่ในสถานะใดสถานะหนึ่งเท่านั้น และจะอยู่พร้อมกันทั้ง 2 สถานะไม่ได้ ซึ่งอาจใช้แทนความหมายต่างๆ ได้ เช่น การปิด-เปิดวาล์ว, ถูก-ผิด, สูง-ต่ำ, 1-0, ขึ้น-ลง, มี-ไม่มี เป็นต้น

วงจรตรรก มี 2 ชนิด คือ แบบบวก (Positive Logic) กับแบบลบ (Negative Logic) ตรรกแบบบวกใช้สัญญาณไฟฟ้าระดับสูง แทนสถานะตรรก “1” และใช้สัญญาณไฟฟ้าระดับต่ำ แทนสถานะตรรก “0” ส่วนวงจรตรรกแบบลบจะใช้ สัญญาณไฟฟ้าระดับต่ำ แทนสถานะตรรก “1” และใช้สัญญาณไฟฟ้าระดับสูง แทนสถานะตรรก “0”

สถานะทางตรรก คือสถานะ “1” หรือ “0” ใช้แทนการทำงานของอุปกรณ์ที่เปลี่ยนแปลง 2 สถานะ



ภาพที่ 2.1 (ก) Positive Logic

(ข) Negative Logic

## 2.3 หลักการเบื้องต้นของพีชคณิตบูลีน (Boolean Algebra)

พีชคณิตบูลีน เป็นเทคนิคที่ใช้ในการลดรูป Switching Function ซึ่งผู้คิดค้นนี้เป็น นักคณิตศาสตร์ชาวอังกฤษชื่อ George Boole ในพีชคณิตบูลีน เราใช้ตัวอักษร A, B, C, ... แทนตัวแปร ค่า 2 สถานะ คือ 0 หรือ 1 ความสัมพันธ์ระหว่างตัวแปรแต่ละตัว เราใช้เครื่องหมายทางคณิตศาสตร์ แทนความสัมพันธ์ระหว่างค่าตัวแปรนั้นๆ ซึ่งเครื่องหมายทางคณิตศาสตร์ ได้แก่

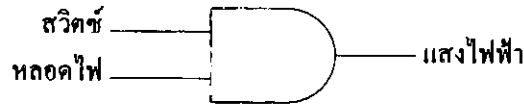
เครื่องหมาย . แทนความหมายของ AND

เครื่องหมาย + แทนความหมายของ OR

เครื่องหมาย - (Bar) แทนความหมายของ NOT

การทำงานของอุปกรณ์ดิจิทัล (Digital Equipment) จะอยู่บนหลักการพื้นฐานของตรรกพื้นฐาน 3 ตัว คือ AND OR และ NOT แต่ละตัวจะมีหลักการทำงานและสัญลักษณ์ของตัวฟังก์ชันเอง โดยจะให้ Y เอาต์พุต (Output) สัญญาณอินพุต (Input) ให้เป็นตัวอักษร ABC ส่วนเลข 1 หมายถึง มีสัญญาณ และเลข 0 หมายถึง ไม่มีสัญญาณ เช่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.2 ไฟในห้องจะติดได้ ก็ต่อเมื่อต่อสะพานไฟและมีหลอดไฟอยู่ในกล่องเท่านั้น

มีสัญญาณเตือน 3 อย่างเพื่อให้ตื่นนอนคือ มีสัญญาณดังขึ้น หรือเพดานห้องยุบลงมา หรือ ลูกออกจากที่นอน



ภาพที่ 2.3 สัญญาณให้ตื่นนอน

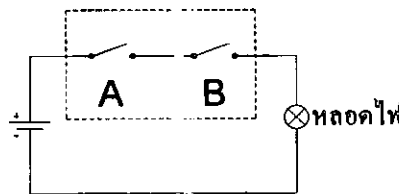
2.3.1 หลักการของเกตแอนด์ (AND gate)

เราสามารถทำความเข้าใจคุณสมบัติของเกตแอนด์ได้ง่ายๆ ถ้าหากเขียนแทนด้วยสวิตช์ (Switch) หรือคอนแทค (Contact) และหลอดไฟดังแสดงในรูปที่ 2.4 ซึ่งประกอบด้วยคอนแทค 2 คอนแทค คือ A และ B ต่อเป็นแบบอนุกรมกัน ส่วนหลอดไฟใช้แทนเอาต์พุต Y โดยกำหนดสถานะของตรรก ดังนี้

คอนแทคปิด (ครบวงจร)	เป็น	1
คอนแทคเปิด (ไม่ครบวงจร)	เป็น	0
หลอดไฟสว่าง	เป็น	1
หลอดไฟดับ	เป็น	0



ภาพที่ 2.4 วงจรเกตแอนด์ที่ใช้คอนแทค



ภาพที่ 2.5 วงจรเกตแอนด์ที่ใช้สวิตช์



ภาพที่ 2.6 สัญลักษณ์ของเกตแอนด์ที่มีอินพุต 2 ตัว

ตารางที่ 2.2 ตารางค่าความจริงของเกตแอนด์

A	B	$Y = A.B$
0	0	0
0	1	0
1	0	0
1	1	1

กฎสำหรับการเขียนตารางค่าความจริงของสถานะอินพุตที่ไม่ซ้ำกันได้จากหลักการของการใช้เลขฐานสองยกกำลังจำนวนอินพุตของวงจร โดยกำหนดให้  $n$  คือจำนวนอินพุตของวงจรไม่ว่าจะมีกี่อินพุต เราก็สามารถใช้กฎนี้ได้เสมอ ซึ่งคูได้จากตารางค่าความจริงของเกตแอนด์แบบ 2 อินพุต จะได้  $2^n = 2^2$  เท่ากับ 4 อินพุต

จากตารางค่าความจริงจะเห็นว่าในกรณี 2 อินพุต จะเขียนสถานะของอินพุตที่ไม่ซ้ำกันได้เพียง 4 แบบ โดยตารางที่เขียนขึ้นมานี้จะเป็นจริงเสมอ เราจึงเรียกว่าตารางค่าความจริง และตารางนี้จะแสดงคุณสมบัติเฉพาะตัวของเกตแอนด์เท่านั้น ซึ่งสามารถเขียนสัญลักษณ์ของเกตแอนด์ได้ดังภาพที่ 2.6

สมการพีชคณิตหรือผลลัพธ์ของวงจรเกตแอนด์สามารถเขียนแทนได้ด้วยสมการ

$$Y = A.B$$

ในทางปฏิบัติ เครื่องหมาย . อาจไม่เขียนก็ได้ เราจึงเขียนสมการได้เป็น

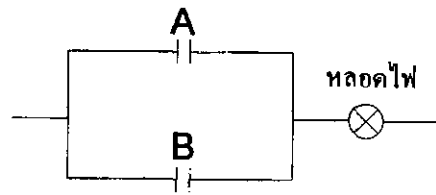
$$Y = AB$$

จากตารางที่ 2.2 สรุปได้ว่าเอาต์พุตของวงจรเกตแอนด์จะเป็น 1 ก็ต่อเมื่อ อินพุต (A และ B) เป็น 1 ทั้งหมด และจะได้เอาต์พุตเป็น 0 เมื่ออินพุตตัวใดตัวหนึ่งหรือทั้งหมดเป็น 0

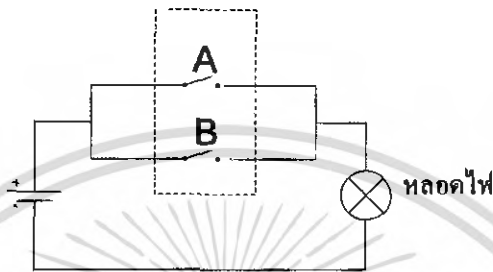
### 2.3.2 หลักการของเกตออร์ (OR gate)

เราสามารถทำความเข้าใจคุณสมบัติของเกตออร์ได้ง่ายๆ ถ้าหากเขียนแทนด้วยสวิทช์ หรือคอนแทคและหลอดไฟดังแสดงในภาพที่ 2.7 ซึ่งประกอบด้วยคอนแทค 2 คอนแทค คือ A และ B ต่อแบบขนานกัน ส่วนหลอดไฟใช้แทนเอาต์พุต Y แล้วกำหนดสถานะทางด้านอินพุตเหมือนกับเกตแอนด์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.7 วงจรเกดออร์ที่ใช้คอนแทค



ภาพที่ 2.8 วงจรเกดออร์ที่ใช้สวิตช์



ภาพที่ 2.9 สัญลักษณ์ของเกดออร์ที่มีอินพุต 2 ตัว

ตารางที่ 2.3 ตารางค่าความจริงของเกดออร์

A	B	$Y = A + B$
0	0	0
0	1	1
1	0	1
1	1	1

ตารางที่ 2.3 จะแสดงคุณสมบัติเฉพาะตัวของเกดออร์ ซึ่งสามารถเขียนสัญลักษณ์ของเกดออร์ได้ดังภาพที่ 2.9 สมการพีชคณิตหรือผลลัพธ์ของวงจรเกดออร์สามารถเขียนแทนได้ด้วยสมการ

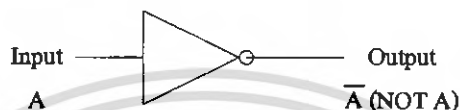
$$Y = A + B$$

จากตารางค่าความจริงของเกดออร์ สรุปได้ว่าเอาต์พุตของวงจรเกดออร์จะเป็น 0 ก็ต่อเมื่ออินพุต (A และ B) เป็น 0 ทั้งหมด และจะได้เอาต์พุตเป็น 1 เมื่ออินพุตตัวใดตัวหนึ่งหรือทั้งหมดเป็น

1

### 2.3.3 หลักการของเกตนอต (NOT gate)

เกตนอต หรือเกตอินเวอร์เตอร์ (Inverter gate) ต่างจากเกตแอนด์และเกตออร์ตรงที่เกตนอตจะมีอินพุตเพียงตัวเดียวเท่านั้น ซึ่งหน้าที่ของนอต หรืออินเวอร์เตอร์คือ การคอมพลิเมนต์ (Complement) หรือกลับสภาวะของตัวแปรค่า หรืออาจจะกล่าวได้ว่าเอาท์พุตของวงจรถูกนอตเป็นคอมพลิเมนต์ของอินพุต เช่น ถ้าอินพุตเท่ากับ 0 แสดงว่าเอาต์พุตเท่ากับ 1 ถ้าอินพุตเท่ากับ 1 แสดงว่าเอาต์พุตเท่ากับ 0



ภาพที่ 2.10 สัญลักษณ์ของเกตนอต

### ตารางที่ 2.4 ตารางค่าความจริงของเกตนอต

A	$Y = \bar{A}$
0	1
1	0

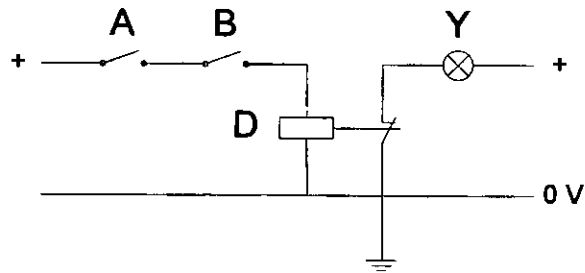
ตารางที่ 2.4 จะแสดงคุณสมบัติเฉพาะตัวของเกตนอต ซึ่งสามารถเขียนสัญลักษณ์ของเกตนอตได้ดังภาพที่ 2.10 สมการพีชคณิตหรือผลลัพธ์ของวงจรถูกนอตสามารถเขียนแทนได้ด้วยสมการ

$$Y = \bar{A}$$

จากตารางค่าความจริงของเกตนอต สรุปได้ว่าเอาท์พุตของวงจรถูกนอตจะเป็น 0 ก็ต่อเมื่ออินพุต (A) เป็น 1 และจะได้เอาท์พุตเป็น 0 เมื่ออินพุตเป็น 1

### 2.3.4 หลักการของเกตแนนด์ (NAND gate)

เราสามารถทำความเข้าใจการทำงาน และคุณสมบัติของเกตแนนด์ได้ง่ายๆ ถ้าหากเขียนวงจรโดยใช้วงจรถอนแทค และรีเลย์ประกอบกัน ด้วยวงจรถอนแทคและเกตอินเวอร์เตอร์ ดังในภาพที่ 2.4 โดยกำหนดให้คอนแทค A และ B เป็นอินพุตของวงจรและหลอดไฟ Y เป็นเอาท์พุตของวงจร การกำหนดสถานะของตรรกอินพุตเหมือนกับเกตแอนด์หรือเกตออร์และเขียนตามตารางค่าความจริงได้ตามกฎคือ  $2^2 = 4$  แบบ



ภาพที่ 2.11 วงจรเกตแนนด์ 2 อินพุตที่ใช้คอนแทคและรีเลย์



ภาพที่ 2.12 สัญลักษณ์ของเกตแนนด์ที่มีอินพุต 2 ตัว

ตารางที่ 2.5 ตารางค่าความจริงของเกตแนนด์

A	B	$Y = \overline{A \cdot B}$
0	0	1
0	1	1
1	0	1
1	1	0

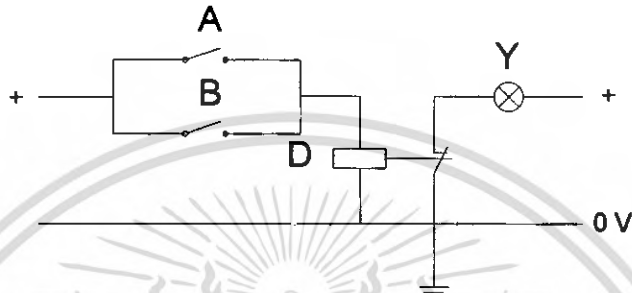
ตารางที่ 2.5 จะแสดงคุณสมบัติเฉพาะตัวของเกตแนนด์ ซึ่งสามารถเขียนสัญลักษณ์ของเกตแนนด์ได้ดังภาพที่ 2.12 สมการพีชคณิตหรือผลลัพธ์ของวงจรถูกแทนด้วยสมการ

$$Y = \overline{A \cdot B}$$

จากตารางค่าความจริงของเกตแนนด์สรุปได้ว่าเอาต์พุตของวงจรถูกแทนด้วย 0 ก็ต่อเมื่อ อินพุต (A และ B) เป็น 1 ทั้งหมด และจะได้เอาต์พุตเป็น 1 เมื่ออินพุตตัวใดตัวหนึ่งหรือทั้งหมดเป็น 0

2.3.5 หลักการของเกตเนอร์ (NOR gate)

เราสามารถทำความเข้าใจการทำงาน และคุณสมบัติของเกตเนอร์ได้ง่ายๆ ถ้าหากเขียนวงจร โดยการใช้วงจรคอนแทค และรีเลย์ประกอบกันด้วยวงจรเกตออร์และเกตอินเวอร์เตอร์ ดังในภาพที่ 2.13 โดยกำหนดให้คอนแทค A และ B เป็นอินพุตของวงจรและหลอดไฟ Y เป็นเอาต์พุตของวงจร การกำหนดสถานะของตรรกอินพุตเหมือนกับเกตแอนด์หรือเกตออร์ และเขียนตามตารางค่าความจริงได้ตามกฎคือ  $2^2 = 4$  แบบ



ภาพที่ 2.13 วงจรเกตเนอร์ 2 อินพุตที่ใช้คอนแทคและรีเลย์



ภาพที่ 2.14 สัญลักษณ์ของเกตเนอร์ที่มีอินพุต 2 ตัว

ตารางที่ 2.6 ตารางค่าความจริงของเกตเนอร์

A	B	$Y = \overline{A + B}$
0	0	1
0	1	0
1	0	0
1	1	0

ตารางที่ 2.6 จะแสดงคุณสมบัติเฉพาะตัวของเกตเนอร์ ซึ่งสามารถเขียนสัญลักษณ์ของเกตเนอร์ได้ดังภาพที่ 2.14 สมการพีชคณิตหรือผลลัพธ์ของวงจรเกตเนอร์สามารถเขียนแทนได้ด้วยสมการ

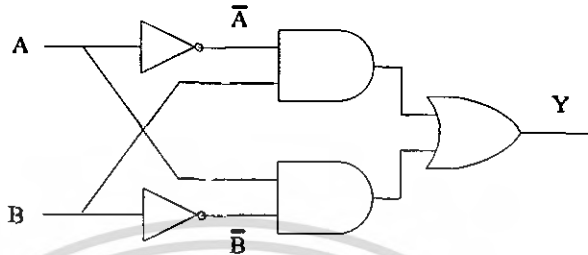
$$Y = \overline{A + B}$$

จากตารางค่าความจริงของเกตเนอร์ สรุปได้ว่าเอาต์พุตของวงจรเกตเนอร์จะเป็น 1 ก็ต่อเมื่ออินพุต (A และ B) เป็น 0 ทั้งหมด และจะได้เอาต์พุตเป็น 0 เมื่ออินพุตตัวใดตัวหนึ่งหรือทั้งหมดเป็น

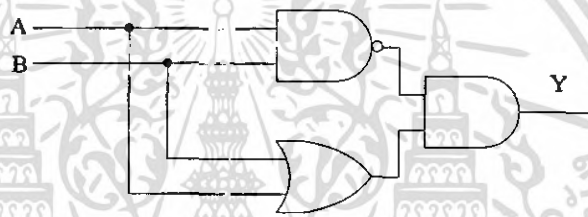
1

2.3.6 หลักการของเกตเอ็กซ์คลูซีฟออร์ (Exclusive OR gate)

คือเกตที่มีอินพุต 2 ตัว และให้อาท์พุต 1 ตัว การทำงานของวงจรจะให้เอาท์พุตเป็น 1 ก็ต่อเมื่ออินพุตมีสภาวะทางตรรกต่างกัน และจะให้เอาท์พุตเป็น 0 ก็ต่อเมื่ออินพุตมีสภาวะทางตรรกเหมือนกัน



ภาพที่ 2.15 (ก) การสร้างเกตเอ็กซ์คลูซีฟออร์



ภาพที่ 2.15 (ข) การสร้างเกตเอ็กซ์คลูซีฟออร์



ภาพที่ 2.16 สัญลักษณ์ของเกตเอ็กซ์คลูซีฟออร์ที่มีอินพุต 2 ตัว

ตารางที่ 2.7 ตารางค่าความจริงของเกตเอ็กซ์คลูซีฟออร์

A	B	$Y = A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

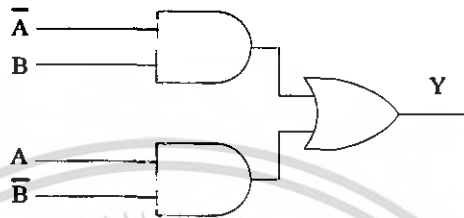
ตารางที่ 2.7 จะแสดงคุณสมบัติเฉพาะตัวของเกตเอ็กซ์คลูซีฟออร์ ซึ่งสามารถเขียนสัญลักษณ์ของเกตเอ็กซ์คลูซีฟออร์ได้ดังภาพที่ 2.16 สมการพีชคณิตหรือผลลัพธ์ของวงจรเกตเอ็กซ์คลูซีฟออร์สามารถเขียนแทนได้ด้วยสมการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

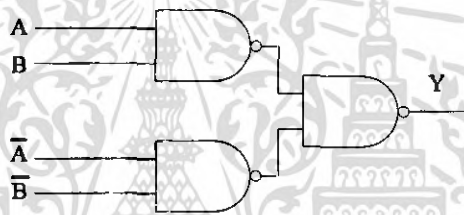
$$\begin{aligned}
 Y &= \bar{A}B + A\bar{B} \\
 &= (A+B)(\bar{A}+\bar{B}) \\
 &= A \oplus B
 \end{aligned}$$

$$Y = A \oplus B$$

แต่คิอิจลอบางระบบนั้นมีค่าคอมพลิเมนต์อยู่แล้ว ดังนั้นเราจึงไม่ต้องสร้างคอมพลิเมนต์นั้นอีก เพียงแค่ต่อสายแล้วนำมาใช้งานได้เลย



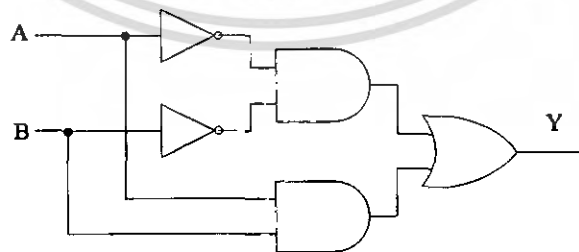
ภาพที่ 2.17 (ก) การสร้างเกตเอ็กซ์คลูซีฟพอร์โดยนำเอาค่าคอมพลิเมนต์มาใช้



ภาพที่ 2.17 (ข) การสร้างเกตเอ็กซ์คลูซีฟพอร์โดยนำเอาค่าคอมพลิเมนต์มาใช้

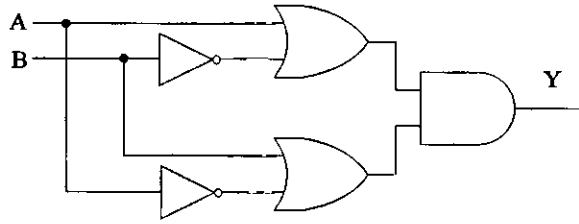
### 2.3.7 หลักการของเกตเอ็กซ์คลูซีฟพอร์ (Exclusive NOR gate)

เกตเอ็กซ์คลูซีฟพอร์ หรือคอมพาราเตอร์ (Comparators) คือเกตที่มีอินพุต 2 ตัว และให้อเอาท์พุต 1 ตัว การทำงานของวงจรจะให้เอาท์พุตเป็น 1 ก็ต่อเมื่ออินพุตมีสภาวะทางตรรกเหมือนกัน และจะให้เอาท์พุตเป็น 0 ก็ต่อเมื่ออินพุตมีสภาวะทางตรรกต่างกัน



ภาพที่ 2.18 (ก) การสร้างเกตเอ็กซ์คลูซีฟพอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.18 (ข) การสร้างเกตเอ็กซ์คลูซีฟนอร์



ภาพที่ 2.19 สัญลักษณ์ของเกตเอ็กซ์คลูซีฟนอร์ที่มีอินพุต 2 ตัว

ตารางที่ 2.8 ตารางค่าความจริงของเกตเอ็กซ์คลูซีฟนอร์

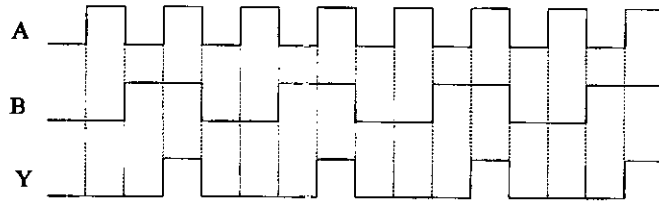
A	B	$Y = A \oplus B$
0	0	1
0	1	0
1	0	0
1	1	1

ตารางนี้จะแสดงคุณสมบัติเฉพาะตัวของเกตเอ็กซ์คลูซีฟนอร์เท่านั้น ซึ่งสามารถเขียนสัญลักษณ์ของเกตเอ็กซ์คลูซีฟนอร์ได้ดังภาพที่ 2.19 สมการพีชคณิตหรือผลลัพธ์ของวงจรถัดเอ็กซ์คลูซีฟนอร์สามารถเขียนแทนได้ด้วยสมการ

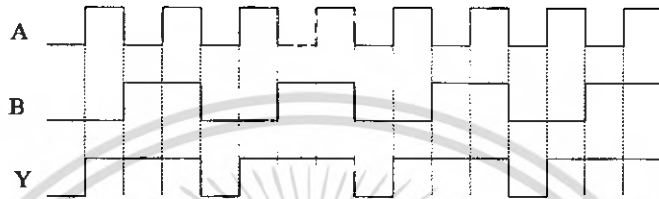
$$\begin{aligned}
 Y &= \bar{A}\bar{B} + AB \\
 &= (A+B)(\bar{A}+B) \\
 &= A \oplus B \\
 &= A \odot B \\
 Y &= A \odot B
 \end{aligned}$$

### 2.3.8 แผนภูมิเวลา (Timing Diagram)

แผนภูมิเวลา คือรูปกราฟแบบหนึ่งที่ใช้แทนความหมายของสัญญาณอินพุตและเอาต์พุตที่มีความสัมพันธ์กันในวงจรถัด หรือสวิตซิงเน็ตเวิร์ค (Switching Network) ซึ่งเราสามารถออกแบบวงจรถัดจากตารางค่าความจริงที่กำหนดให้ และเราก็สามารถออกแบบวงจรถัดจากแผนภูมิเวลาได้เช่นเดียวกัน



ภาพที่ 2.20 ตัวอย่างของแผนภูมิเวลาจากสมการ  $Y = A \cdot B$






ภาพที่ 2.21 ตัวอย่างของแผนภูมิเวลาจากสมการ  $Y = A + B$

2.3.9 สรุป

การศึกษาทางพีชคณิตเกี่ยวกับจำนวนของเลขฐานสองและตรรก เรียกว่า พีชคณิตบูลีน จุดประสงค์เพื่อหาวิธีการเขียนตรรกที่ซับซ้อนให้ง่ายขึ้น ดังตารางที่ 2.9 จะเป็นการสรุปสัญลักษณ์ และพื้นฐานการทำงานของเกตต่างๆ สัญญาณอินพุตจะเขียนด้วยตัวอักษร A, B เป็นต้น ส่วนสัญญาณเอาต์พุตจะแทนด้วย Y

ตารางที่ 2.9 ตารางสัญลักษณ์ของเกตและสมการพีชคณิต

ชื่อของเกตต่างๆ	สัญลักษณ์ของเกตต่างๆ	สมการพีชคณิต	คำอธิบายทางตรรกศาสตร์
AND		$Y = A \cdot B$ $= AB$	ถ้า $A = 1$ และ $B = 1$ จะได้ $Y = 1$
OR		$Y = A + B$	ถ้า $A = 0$ และ $B = 0$ จะได้ $Y = 0$
NOT		$Y = \bar{A}$	ถ้า $A = 0$ จะได้ $Y = 1$ ถ้า $A = 1$ จะได้ $Y = 0$
NAND		$Y = \overline{A \cdot B}$	ถ้า $A = 1$ และ $B = 1$ จะได้ $Y = 0$

ชื่อ ของเกตต่างๆ	สัญลักษณ์ ของเกตต่างๆ	สมการ พีชคณิต	คำอธิบาย ทางตรรกศาสตร์
NOR		$Y = \overline{A + B}$	ถ้า $A = 0$ และ $B = 0$ จะได้ $Y = 1$
Exclusive OR		$Y = A \oplus B$	ถ้า $A = 0$ และ $B = 0$ หรือถ้า $A = 1$ และ $B = 1$ จะได้ $Y = 0$
Exclusive NOR		$Y = A \odot B$	ถ้า $A = 0$ และ $B = 0$ หรือถ้า $A = 1$ และ $B = 1$ จะได้ $Y = 1$

## 2.4 หลักการเบื้องต้นของภาษาซี (C Language)

### 2.4.1 กล่าวนำ

ภาษา C เป็นภาษาที่ถือกำเนิดมายาวนาน โดยแต่เดิมนั้นภาษา C ถูกพัฒนาขึ้นเพื่อให้เป็นภาษาที่ใช้สำหรับการสร้างระบบปฏิบัติการยูนิกซ์ เนื่องจากในขณะนั้นระบบปฏิบัติการยูนิกซ์เขียนด้วยภาษาแอสเซมบลี (Assembly) ซึ่งเป็นภาษาที่ซัดติดกับฮาร์ดแวร์ของเครื่อง ดังนั้นการที่จะย้ายระบบปฏิบัติการไปใช้กับเครื่องอื่นจึงเป็นเรื่องเป็นไปไม่ได้เลย ซึ่งนับเป็นข้อเสียที่ค่อนข้างใหญ่สำหรับภาษาแอสเซมบลี ดังนั้นภาษา C ซึ่งเป็นภาษาที่ไม่ซัดติดกับฮาร์ดแวร์จึงถูกพัฒนาขึ้นมา ในปัจจุบันภาษา C ไม่ได้จำกัดอยู่เพียงแค่การสร้างระบบปฏิบัติการเท่านั้น แต่สามารถนำไปใช้สร้างโปรแกรมเพื่อทำงานในทุกประเภท เช่น งานเกี่ยวกับการคำนวณ ควบคุมการทำงานของอุปกรณ์/ฮาร์ดแวร์ชนิดต่าง ๆ การจัดการฐานข้อมูล หรือสร้างโปรแกรมสำหรับจัดพิมพ์เอกสาร เป็นต้น

### 2.4.2 การสั่งงานคอมพิวเตอร์ด้วยภาษาโปรแกรม

ภาษาที่เครื่องคอมพิวเตอร์เข้าใจ เรียกว่า ภาษาเครื่อง (Machine Language) ซึ่งอยู่ในรูปแบบของรหัสเลขฐานสอง (Binary Code) ซึ่งประกอบด้วยตัวเลขอยู่เพียงแค่ 2 ตัว คือ 0 และ 1 เรียงสลับและต่อกันเป็นความหมายที่เครื่องเข้าใจ

ต่อมาจึงมีการพัฒนาภาษาสำหรับสื่อสารกับคอมพิวเตอร์ที่มนุษย์เข้าใจ โดยใช้ตัวอักษรภาษาอังกฤษมากำหนดเป็นรูปแบบคำสั่งซึ่งส่วนใหญ่เป็นคำที่มีความหมายในภาษาอังกฤษ แต่การที่จะทำให้คอมพิวเตอร์เข้าใจภาษาที่พัฒนาขึ้นมาได้นั้น ต้องมีตัวกลางเพื่อทำหน้าที่แปลภาษานั้นเป็นภาษาเครื่องอีกที ซึ่งสามารถแบ่งระดับตามลักษณะและการทำงานของแต่ละภาษาได้เป็น 2 ระดับ ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**2.4.2.1 ภาษาระดับต่ำ (Low level language)** เป็นภาษาที่ใกล้เคียงกับภาษาเครื่องมากที่สุด สามารถเขียนคำสั่งเพื่อติดต่อกับงานกับอุปกรณ์ฮาร์ดแวร์ได้โดยตรง โปรแกรมที่เขียนด้วยภาษาระดับต่ำจะทำงานได้รวดเร็วมาก แต่ปัญหาของภาษาระดับต่ำก็คือ เขียนยาก รูปแบบคำสั่งจุกจิกและทำความเข้าใจได้ยาก นอกจากนี้อีกเหตุผลที่สำคัญก็คือ ภาษาระดับต่ำยึดติดกับชนิดของเครื่องคอมพิวเตอร์ นั่นหมายถึงถ้าเราเขียนโปรแกรมด้วยภาษาระดับต่ำสำหรับคอมพิวเตอร์แบบหนึ่ง โปรแกรมนั้นจะไม่สามารถนำไปใช้กับคอมพิวเตอร์อีกแบบได้ ตัวอย่างของภาษาระดับต่ำได้แก่ แอสเซมบลี (Assembly)

**2.4.2.2 ภาษาระดับสูง (High level language)** เป็นภาษาที่ใกล้เคียงกับภาษาที่ใช้สื่อสารกันตามปกติ คำสั่งต่าง ๆ มักเป็นคำที่มีความหมายในภาษาอังกฤษ ทำให้จดจำและเขียนได้ง่าย แต่ข้อเสียของภาษาระดับสูงก็คือ มักไม่มีคำสั่งในการติดต่อกับอุปกรณ์ฮาร์ดแวร์โดยตรง และทำงานได้ช้ากว่าภาษาระดับต่ำ ตัวอย่างของภาษาระดับสูง ได้แก่ Pascal, Cobol, Fortran หรือ Basic เป็นต้น

เนื่องจากภาษาระดับต่ำและระดับสูงต่างก็มีข้อดีและข้อเสียที่แตกต่างกัน ดังนั้นภาษา C จึงพัฒนาขึ้นมาโดยปรับปรุงข้อเสียและนำเอาข้อดีของภาษาทั้ง 2 ระดับมาใช้ โดยคำสั่งของภาษา C เป็นคำที่มีความหมายในภาษาอังกฤษ สามารถจดจำและเขียนได้ง่ายเหมือนกับภาษาระดับสูง แต่ภาษา C ทำงานได้อย่างรวดเร็ว และมีคำสั่งที่ให้ผู้เขียนโปรแกรมเรียกใช้เพื่อติดต่อกับงานกับอุปกรณ์ฮาร์ดแวร์โดยตรงเหมือนกับภาษาระดับต่ำ ดังนั้นภาษา C จึงถูกจัดให้เป็นภาษาระดับกลาง

### 2.4.3 จุดเด่นของภาษา C

ในปัจจุบันภาษา C ได้รับการยอมรับและใช้งานกันอย่างกว้างขวาง เนื่องจาก

- ภาษา C เป็นภาษาที่มีการกำหนดมาตรฐานสำหรับเครื่องคอมพิวเตอร์ทุกรุ่น และระบบปฏิบัติการทุกชนิด ทำให้โครงสร้างทางภาษา ฟังก์ชันไลบรารี (Library) ต่าง ๆ สามารถนำไปใช้งานระหว่างเครื่องแต่ละรุ่นและระบบปฏิบัติการแต่ละชนิดได้
- โปรแกรมที่เขียนขึ้นด้วยภาษา C มีขนาดเล็กและทำงานได้เร็ว
- ภาษา C มีโครงสร้างทางภาษาที่ดี และเครื่องหมายสำหรับดำเนินการ ไม่ว่าจะเป็นการคำนวณทางคณิตศาสตร์ ตรรกศาสตร์ หรือการเปรียบเทียบ มีประสิทธิภาพการทำงานสูง
- สามารถเขียนคำสั่งภาษา C เพื่อควบคุมการทำงานของอุปกรณ์ฮาร์ดแวร์บางส่วนได้
- มีฟังก์ชันสำเร็จรูปสำหรับงานประเภทต่าง ๆ ให้เลือกใช้มากมาย ซึ่งช่วยประหยัดเวลาในการเขียนคำสั่ง นอกจากนี้ถ้าฟังก์ชันที่ภาษา C เตรียมไว้ให้ ใช้งานได้ไม่ตรงตามต้องการทั้งหมด เราสามารถเขียนคำสั่งเพิ่มเติมลงไปได้

#### 2.4.4 สรุป

จากจุดเด่นข้างต้นของภาษา C เราจึงนำโปรแกรมภาษา C มาใช้เขียนโปรแกรมจำลองการทำงานของวงจรตรรก แต่เมื่อโปรแกรมมีความซับซ้อนมากขึ้น การเขียนโปรแกรมแบบโครงสร้างนั้นไม่สามารถแก้ปัญหาได้ เราจึงต้องแก้ไขด้วยการเขียนโปรแกรมด้วยการเขียนโปรแกรมแบบเชิงวัตถุ (OOP : Object Oriented Programming) คือวิธีออบเจกต์โอเรียนเต้ดนี้เป็นการรวมระหว่างการเขียนโปรแกรมแบบโครงสร้างกับแนวความคิดที่มองปัญหาเป็นหน่วย หรือที่เรียกว่าออบเจกต์ ซึ่งเป็นคุณลักษณะของโปรแกรมภาษา C++ แต่เพื่อเป็นการสะดวกสำหรับการติดต่อกับผู้ใช้ (User Interface) เราจึงเลือกที่จะใช้โปรแกรม Microsoft Visual C++ นอกจากนี้โปรแกรมนี้อยังมีข้อดีอีกคือ ความสามารถในการพอร์ตเทมิต (Portability) หมายความว่าเราสามารถนำซอร์สโค้ดที่เขียนด้วย Visual C++ ในเวอร์ชันต่ำกว่ามาคอมไพล์ และลิงค์ใหม่ได้โดยใช้ Visual C++ ที่สูงกว่าได้



## บทที่ 3

### การสร้างและการออกแบบ

การสร้างและการออกแบบ ปฏิญาณิพนธ์นี้ใช้หลักการคิดและการออกแบบโดยอ้างอิงจากทฤษฎีพื้นฐานทางตรรกศาสตร์ และวิธีการเขียนโปรแกรมจำลองการทำงานของวงจรตรรกจากทฤษฎีการใช้โปรแกรม Microsoft Visual C++

#### 3.1 ทฤษฎีพื้นฐาน

ทฤษฎีพื้นฐานในการสร้างโครงงานนี้สามารถแบ่งออกได้เป็นสองส่วนคือ ส่วนที่ใช้ติดต่อกับผู้ใช้และส่วนที่นำข้อมูลจากผู้ใช้มาประมวลผล อธิบายได้ดังนี้

##### 3.1.1 ส่วนที่ใช้ติดต่อกับผู้ใช้

ส่วนที่ติดต่อกับผู้ใช้ ซึ่งผู้ใช้จะเลือกรูปแบบการใช้งานของโปรแกรมว่าผู้ใช้ต้องการป้อนอินพุตของการประมวลผลเป็นแบบใดระหว่างการป้อนสมการ หรือการป้อนรูปวงจรรวมเพื่อให้โปรแกรมนำไปประมวลผลต่อไป

สมการและรูปวงจรรวมที่ผู้ใช้ป้อนให้กับโปรแกรมจะอ้างอิงทฤษฎีพื้นฐานทางตรรกศาสตร์ คือ หลักการของเกตแอนด์ เกตออร์ เกตนอด เกตแนนด์ เกตนอร์ เกตเอ็กซ์คลูซีฟออร์ และเกตเอ็กซ์คลูซีฟนอร์ดังที่ได้กล่าวไว้ในบทที่ 2

##### 3.1.1.1 สัญลักษณ์และรูปแบบการใช้โปรแกรม

เนื่องจากโปรแกรมนี้ได้ออกแบบมาให้เหมาะสำหรับผู้ใช้ทั่วไป โดยไม่จำเป็นต้องมีความรู้พื้นฐานทางตรรกศาสตร์ก็สามารถใช้โปรแกรมนี้ได้ ดังนั้นจึงได้ดัดแปลงสัญลักษณ์บางสัญลักษณ์ ซึ่งปฏิญาณิพนธ์ฉบับนี้จะใช้รูปสมการและสัญลักษณ์ดังนี้

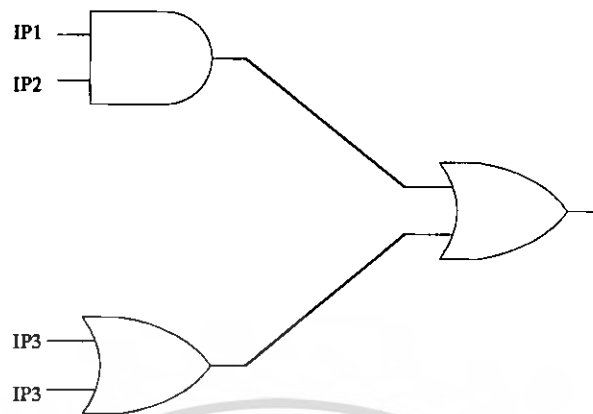
ตารางที่ 3.1 สัญลักษณ์ที่ใช้ในปฏิญาณิพนธ์ฉบับนี้

ลำดับ	สัญลักษณ์ที่ใช้ในโปรแกรม	ความหมาย
1	IP1	ตัวแปรตัวที่1
2	IP2	ตัวแปรตัวที่2
3	IP3	ตัวแปรตัวที่3
4	IP4	ตัวแปรตัวที่4
5	IP5	ตัวแปรตัวที่5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(IP1 and IP2) or (IP3 or IP4)



ภาพที่ 3.3 (IP1 and IP2) or (IP3 or IP4)

### 3.1.2 ส่วนประมวลผล

เมื่อโปรแกรมรับค่าจำนวนตัวแปรจากผู้ใช้แล้ว ส่วนประมวลผลจะนำค่านั้นมาสร้างเป็นตารางค่าความจริงเพื่อใส่ค่าให้ตัวแปรแต่ละตัวยกตัวอย่างเช่น จำนวนตัวแปรเท่ากับ 3 จะได้ตารางค่าความจริงดังตารางที่ 3.2

ตารางที่ 3.2 ตารางค่าความจริงของจำนวนตัวแปร 3 ตัว

IP1	IP2	IP3
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

เมื่อผู้ใช้ป้อนสมการตรรก หรือรูปวงจรถรกเพื่อใช้ในการประมวลผลของโปรแกรมแล้ว ลำดับการทำงานของโปรแกรมต่อไปคือ ถ้าผู้ใช้ป้อนสมการตรรกโปรแกรมจะนำสมการไปเขียนลงในเท็กซ์ไฟล์ และถ้าผู้ใช้ป้อนเป็นวงจรถรกโปรแกรมจะทำการแปลงรูปวงจรถรกให้เป็นสมการตรรกก่อน แล้วจึงนำสมการที่ได้ไปเขียนลงในเท็กซ์ไฟล์ หลังจากนั้น โปรแกรมส่วนประมวลผลจะอ่านข้อมูลจากเท็กซ์ไฟล์เพื่อนำไปประมวลผลต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ยกตัวอย่างเช่น

(IP1 and IP2) or IP3

สมการในเท็กซ์ไฟล์ คือ (IP1 and IP2) or IP3

ขั้นตอนการประมวลผลคือ

1. จากตารางที่ 3.2 โปรแกรมจะทำการประมวลผลในวงเล็บก่อน การเรียงลำดับวงเล็บจะเรียงจากวงเล็บทางด้านในก่อน และเรียงจากทางด้านขวามือไปทางด้านซ้ายมือ

2. หลังจากเลือกวงเล็บได้แล้วโปรแกรมจะใส่ค่าให้กับตัวแปรโดยเรียงทีละแถว เช่น จากตัวอย่างโปรแกรมจะใส่ค่าดังนี้

$$IP1 = 0$$

$$IP2 = 0$$

3. หลังจากใส่ค่าให้ตัวแปรแล้ว โปรแกรมจะอ่านค่าฟังก์ชันจากสัญลักษณ์ “and” โปรแกรมก็จะนำ IP1 มา and กับ IP2 คือ

$$0 \text{ and } 0 = 0$$

4. การประมวลผลจะเรียงลำดับจากข้อ 1. ถึง 3. เช่นนี้ทุกแถวของตารางค่าความจริงจนกระทั่งครบทุกแถวก็จะได้ผลลัพธ์ของวงเล็บแรกดังตารางที่ 3.3

ตารางที่ 3.3 ตารางค่าความจริงของการประมวลผลของสมการ IP1 and IP2

IP1	IP2	IP1andIP2
0	0	0
0	0	0
0	1	0
0	1	0
1	0	0
1	0	0
1	1	1
1	1	1

5. เมื่อได้ผลลัพธ์ของการคำนวณในวงเล็บแรกแล้ว โปรแกรมก็จะนำผลลัพธ์ที่ได้ไปทำการทำการทางตรรกศาสตร์กับตัวแปรลำดับถัดไป ซึ่งจากตัวอย่างคือตัวแปร IP3 โดยใส่ค่าให้กับตัวแปรนี้เช่นเดียวกันกับ IP1 และ IP2 แล้วนำทำการ or กันกับผลลัพธ์ที่ได้จาก IP1 and IP2 ซึ่งจากตารางที่ 3.3 คือ หลักที่สามจะได้ผลดังตารางที่ 3.4

ตารางที่ 3.4 ตารางค่าความจริงของการประมวลผลของสมการ (IP1andIP2) or IP3

IP1 and IP2	IP3	(IP1andIP2) or IP3
0	0	0
0	1	1
0	0	0
0	1	1
0	0	0
0	1	1
1	0	1
1	1	1

ถ้ามีวงเล็บหรือสมการที่ซับซ้อนมากกว่านี้โปรแกรมก็จะประมวลผลโดยการตรวจสอบวงเล็บด้านในสุดก่อน แล้วค่อยประมวลผลสมการที่อยู่ในวงเล็บนั้น จากนั้นโปรแกรมจะวนดูวงเล็บที่อยู่ด้านในสุดถัดไปแล้วค่อยประมวลผล จนกระทั่งโปรแกรมหาวงเล็บไม่เจอแล้ว โปรแกรมก็จะประมวลผลสมการจากด้านซ้ายมือไปทางด้านขวามือ แล้วจะได้ผลลัพธ์สุดท้ายของสมการออกมา จากนั้นจะแสดงผลเอาที่พูดเป็นในหลักสุดท้ายของตารางค่าความจริง เช่นจากสมการตัวอย่าง โปรแกรมก็จะแสดงเอาที่พูดในหลักสุดท้าย ดังแสดงในตารางที่ 3.4

### 3.2 การออกแบบ

โปรแกรมจำลองการทำงานของวงจรตรรก โดยมีหลักการพื้นฐานดังนี้ ก่อนอื่นจะต้องมีการกำหนดข้อมูลจำนวนอินพุต โดยที่ข้อมูลเข้าเหล่านี้จะกำหนดให้เป็นสัญญาณนาฬิกา(CLOCK) ตั้งแต่ สเต็ปการทำงานแรก จนกระทั่งสเต็ปสุดท้ายลงสู่หน่วยความจำหลัก เพื่อเก็บค่าสภาวะของตัวแปรอินพุตเหล่านี้ไว้ จากนั้นจะมีการนำสภาวะของตัวแปรอินพุตผ่านฟังก์ชันการทำงานในหลายๆฟังก์ชันที่เขียนเป็นแบบจำลองเอาไว้ แล้วจึงนำค่าสภาวะที่ได้ไปเก็บไว้ในตัวแปรเอาต์พุตของแต่ละฟังก์ชันอีกครั้ง การทำงานในลักษณะนี้ถือว่าเสร็จสิ้นการประมวลผลหนึ่งสเต็ป สำหรับในสเต็ปอื่น ๆ ก็สามารถทำได้ในลักษณะเดียวกันจนกระทั่งถึงสเต็ปสุดท้ายถึงจะเป็นอันเสร็จสิ้นกระบวนการประมวลผล และในการแสดงผลบนจอภาพนับได้ว่าเป็นกระบวนการสุดท้ายของการจำลองการทำงานของวงจรตรรก

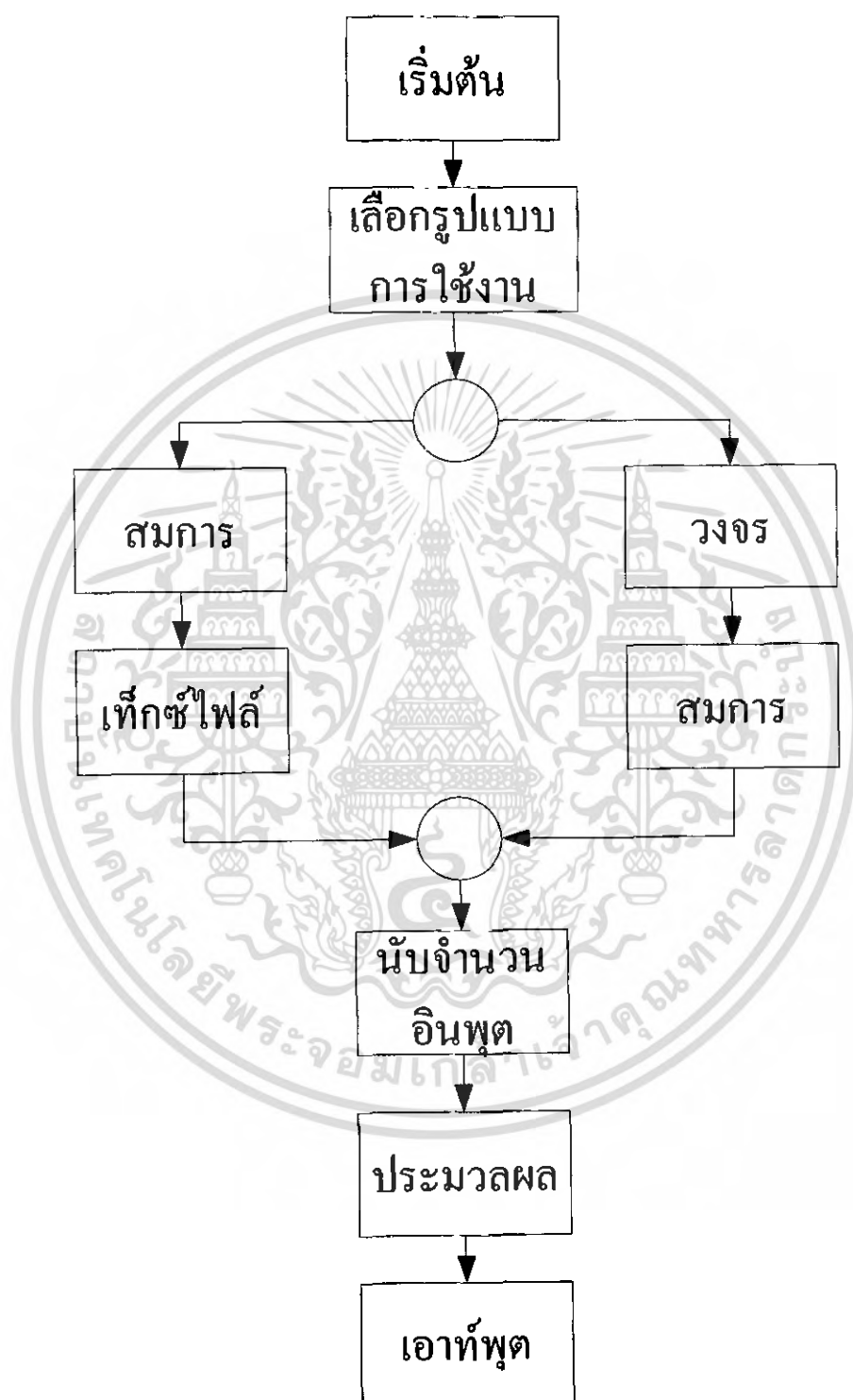
## โปรแกรมสามารถแบ่งหัวข้อตามลักษณะการทำงานได้ดังนี้

- โปรแกรมหลัก
- โปรแกรมสร้างตารางค่าความจริงจากการรับค่าจำนวนอินพุต
- โปรแกรมการรับอินพุตเป็นสมการ
  - รับสมการมีวงเล็บ
  - รับสมการไม่มีวงเล็บ
- โปรแกรมการรับอินพุตเป็นรูปวงจรรรค
  - วาดรูปวงจรร
  - แปลงรูปวงจรรเป็นสมการ
  - สร้างสมการ
- โปรแกรมการเก็บสมการในตัวแปร CString
- โปรแกรมการประมวลผล
- โปรแกรมการแสดงผล
  - สร้างแผนภูมิเวลา
  - สร้างตารางค่าความจริง



### 3.2.1 โปรแกรมหลัก

การทำงานของโปรแกรมหลักสามารถเขียนเป็นแผนผังการทำงานของโปรแกรมได้ดังนี้



ภาพที่ 3.4 แผนผังการทำงานของโปรแกรมหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## อธิบายแผนผังการทำงานของโปรแกรมหลัก

จากภาพที่ 3.4 สามารถแสดงรายละเอียดส่วนต่าง ๆ ของแผนผังการทำงานได้ดังนี้

1. เมื่อโปรแกรมรับจำนวนอินพุต แล้ว โปรแกรมจะนำค่าจำนวนที่ได้นี้ไปสร้างเป็นตารางค่าความจริง

2. ผู้ใช้จะต้องเลือกรูปแบบการใช้งาน

2.1. ป้อนสมการ ผู้ใช้อาจป้อนสมการที่มีวงเล็บ หรือไม่มีวงเล็บก็ได้โดยผู้ใช้อาจป้อนสมการตามที่เรากำหนด คือ

2.1.1 ตัวแปรอินพุตต้องป้อนเป็นคำว่า IP โดยคำว่า "IP" นั้นจะต้องเป็นตัวภาษาอังกฤษตัวพิมพ์ใหญ่แล้วตามด้วยหมายเลขลำดับของตัวแปรอินพุต

2.1.2 ตัวกระทำทางตรรกะ จะต้องป้อนเป็นภาษาอังกฤษตัวพิมพ์เล็กตามชื่อของตัวกระทำทางตรรกะนั้น

2.1.3 สมการจะต้องประกอบด้วย ตัวแปรและตัวกระทำทางตรรกะ

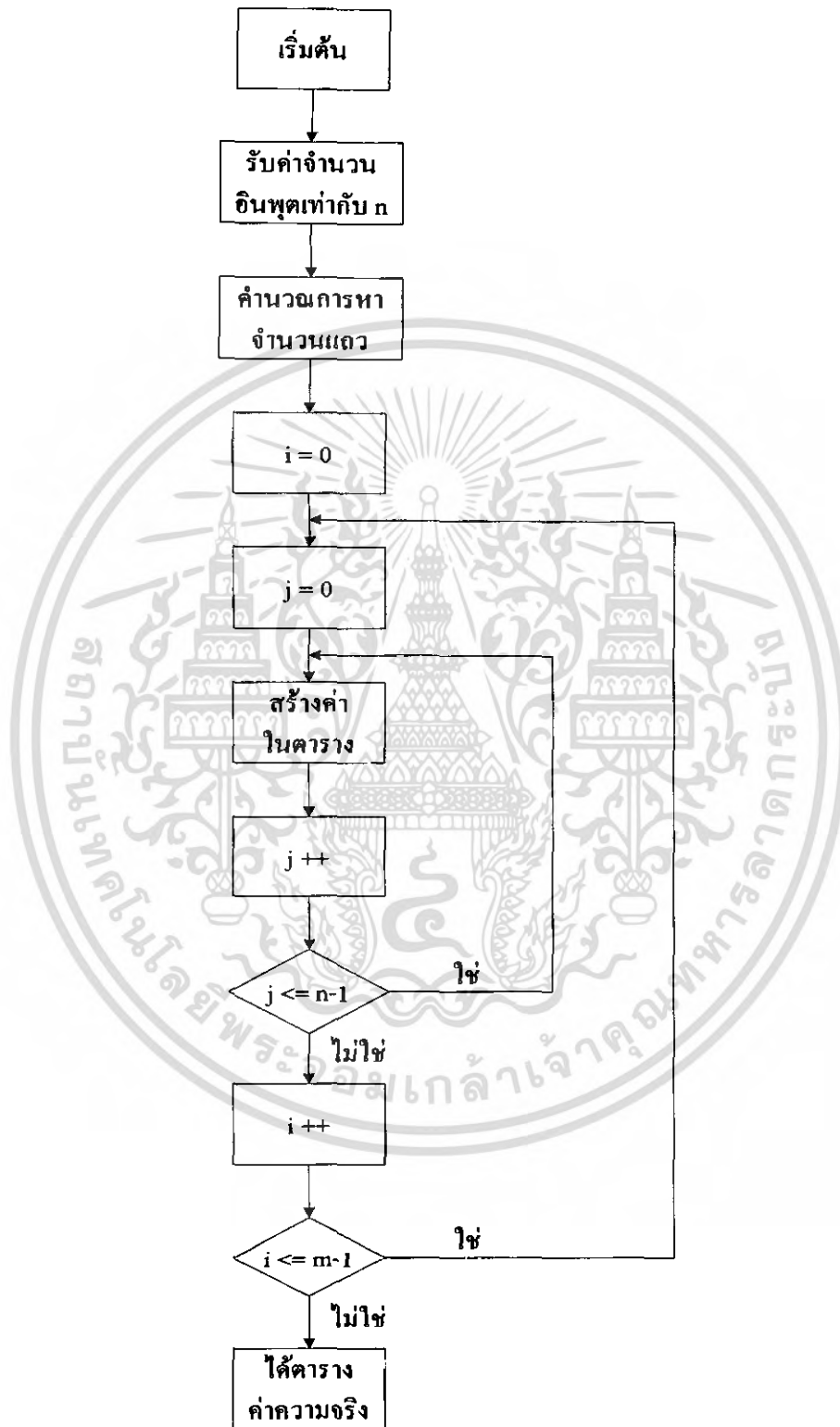
2.2 ป้อนรูปวงจร โดยการสร้างภาพกราฟฟิกของโปรแกรม Microsoft Visual C++ ซึ่งเป็นการสร้างในลักษณะของภาพบิตแมป (Bitmap) ที่เป็นภาพของเกตต่างๆ ที่ใช้ในวงจรถูกโหลด (Load) ภาพที่สร้างเตรียมไว้แล้วมาแสดงตรงหน้าต่างวินโดว์ (Window) ของโปรแกรมตามที่ผู้ใช้เลือกจากเมนู

3. โปรแกรมจะเขียนค่าสมการที่ได้ลงในตัวแปร CString เมื่อมีการกดเมนู RUN ทุกครั้งจะมีการเขียนสมการใหม่ขึ้นทุกครั้ง (เขียนทับสมการเดิม)

4. โปรแกรมจะอ่านข้อมูลจากตัวแปร CString เพื่อนำสมการมาประมวลผล โดยโปรแกรมสามารถประมวลผลได้ทั้งสมการที่มีวงเล็บ และสมการที่ไม่มีวงเล็บ

5. โปรแกรมสามารถแสดงเอาต์พุตในรูปแบบของตารางค่าความจริง และแผนภูมิเวลา

### 3.2.2 โปรแกรมสร้างตารางค่าความจริงจากการรับค่าจำนวนอินพุต



ภาพที่ 3.5 แผนผังการทำงานของโปรแกรมสร้างตารางค่าความจริงจากการรับค่าจำนวนอินพุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อธิบายแผนผังการทำงานของโปรแกรมสร้างตารางค่าความจริงจากการรับค่าจำนวนอินพุต จากภาพที่ 3.5 สามารถแสดงรายละเอียดส่วนต่าง ๆ ของแผนผังการทำงานได้ดังนี้

1. เมื่อได้จำนวนตัวแปรแล้ว โปรแกรมจะทำการคำนวณหาจำนวนแถวจากสมการ

$$m = 2^n$$

เมื่อ

$$m = \text{จำนวนแถว}$$

$$n = \text{จำนวนตัวแปร}$$

2. การคำนวณค่าในตารางค่าความจริง โปรแกรมมีหลักการคำนวณดังนี้

ค่าในตารางค่าความจริง = เศษของการหารจำนวนแถวด้วยสอง

หลักการคิดคือให้

$$y = \text{ลำดับของแถวเริ่มจาก } 0 \ 1 \ 2 \ 3 \ \dots \ m$$

$$i = \text{ลำดับของหลักเริ่มจาก } 0 \ 1 \ 2 \ 3 \ \dots \ n$$

$$x = y / 2$$

ค่าความจริงในแต่ละหลักของแถวที่  $y$  หลักที่  $i = \text{เศษของ } y / 2$  แล้วให้

$$y = x \text{ เพื่อหาเศษของ } y / 2 \text{ จนกระทั่งครบทุกหลักแล้วจึงเริ่มทำในแถวใหม่ ทำ}$$

เช่นนี้จนครบทุกแถวก็จะได้ตารางค่าความจริงของตัวแปรทั้งหมด ( $n$  ตัว)

3. โปรแกรมจะคำนวณตามข้อ 2. จนได้ค่าความจริงครบทุกแถว จะได้ค่าความจริงใน

ตารางตามต้องการ ค่าความจริงที่ได้ คือ กรณีที่เป็นไปได้ทั้งหมดของตัวแปรแต่ละชุด

ตัวอย่างการคำนวณ

$$n = 2$$

$$m = 4$$

ค่าความจริงที่ (0,0)

$$y = 0$$

$$x = 0/2$$

$$x = 0$$

$$(0,0) = \text{เศษของ } y / 2$$

$$= 0$$

### 3.2.3 โปรแกรมรับสมัครมีวงเล็บ



ภาพที่ 3.6 แผนผังการทำงานของโปรแกรมรับสมัครมีวงเล็บ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### อธิบายแผนผังการทำงานของโปรแกรมรับสมการมีวงเล็บ

จากภาพที่ 3.6 สามารถแสดงรายละเอียดส่วนต่าง ๆ ของแผนผังการทำงานเป็นลำดับขั้นตอนได้ดังนี้

#### 1. หาดำแหน่งที่มีวงเล็บเปิดและวงเล็บปิดทั้งหมด เช่น

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	
(	(	I	P	I	a	n	d	I	P	2	)	o	r	(	I	P	3	a	n	d	I	P	4	)	)	)

ตำแหน่งที่มีวงเล็บเปิดและวงเล็บปิดคือ 0, 1, 11, 14, 24, 25

2. เปรียบเทียบตำแหน่งของวงเล็บเปิดและวงเล็บปิดเพื่อหาวงเล็บคู่แรกเพื่อที่จะใช้ทำการคำนวณ ซึ่งมีหลักการคิดคือจะหาดำแหน่งของวงเล็บปิดที่น้อยที่สุด และหาดำแหน่งของวงเล็บเปิดที่มากที่สุดแต่ต้องน้อยกว่าตำแหน่งของวงเล็บปิดก็จะได้สมการคู่แรกที่จะทำการคำนวณ และหาดำแหน่งของวงเล็บเปิดที่มากที่สุด แต่ต้องน้อยกว่าตำแหน่งของวงเล็บปิดก็จะได้สมการคู่แรกที่จะทำการคำนวณ และโปรแกรมก็จะนำสมการที่ได้จากวงเล็บมาคำนวณตามโปรแกรมการคำนวณสมการต่อไป ตำแหน่งที่นำสมการออกมาจะแทนด้วยตัวแปร  $O_1, O_2, O_3, \dots, O_n$  โดยตัวแปร  $O$  จะเก็บเอาที่พูดของสมการที่ถูกแทนที่

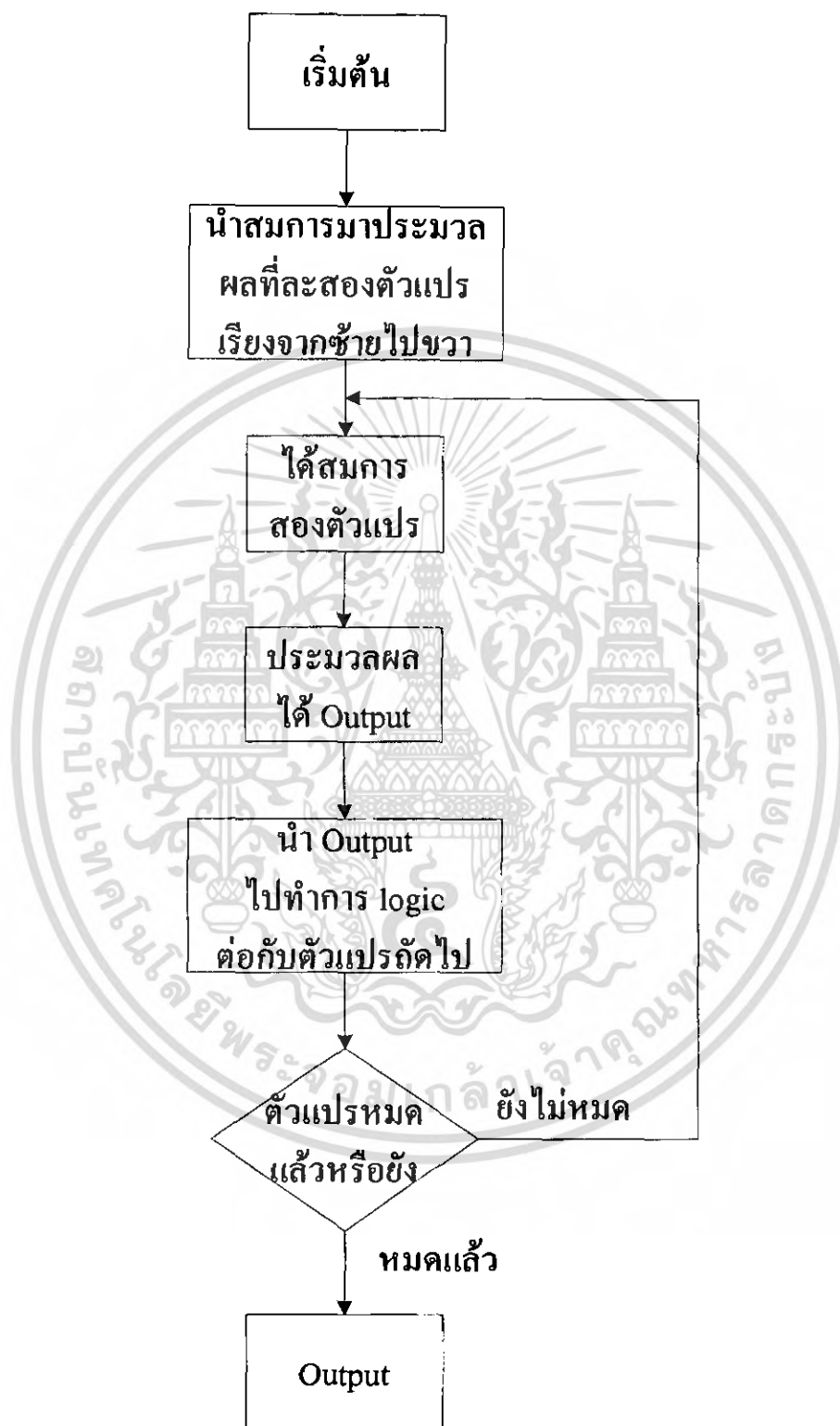
3. ทำการตรวจสอบคำนวณวงเล็บว่าคำนวณทุกวงเล็บแล้วหรือไม่ ถ้ายังไม่หมดโปรแกรมจะทำการคำนวณจนกระทั่งครบทุกวงเล็บ เมื่อครบทุกวงเล็บแล้วก็จะได้อาชีพพูดของสมการออกมา ยกตัวอย่างจากข้อ 1.

วงเล็บที่ทำการคำนวณวงเล็บแรกคือ (IP1andIP2)  
สมการที่เหลือคือ (O1 or (IP3andIP4))  
โดยที่  $O_1 = IP1andIP2$

วงเล็บที่ทำการคำนวณวงเล็บที่สองคือ (IP3andIP4)  
สมการที่เหลือคือ (O1 or O2)  
โดยที่  $O_2 = IP3andIP4$

วงเล็บที่ทำการคำนวณวงเล็บเป็นวงเล็บสุดท้าย (O1 or O2)  
คือ  
เอาที่พูดสุดท้ายคือ  $O_3$

### 3.2.4 โปรแกรมรับสมการไม่มีวงเล็บ



ภาพที่ 3.7 แผนผังการทำงานของโปรแกรมรับสมการไม่มีวงเล็บ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### อธิบายแผนผังการทำงานของโปรแกรมรับสมการไม่มีวงเล็บ

จากภาพที่ 3.7 สามารถแสดงรายละเอียดส่วนต่าง ๆ ของแผนผังการทำงานได้ดังนี้

1. โปรแกรมจะนำสมการมาคำนวณครั้งละสองตัวแปร โดยเรียงจาก ซ้ายไปขวาแล้วจะแทนตำแหน่งที่นำสมการมาจะแทนด้วย ตัวแปร  $O_1, O_2, O_3, \dots, O_n$
2. โปรแกรมจะทำการตรวจสอบว่าคำนวณครบทุกตัวแปรหรือยัง ถ้ายังไม่หมด โปรแกรมก็จะทำการวนลูปคำนวณจนกระทั่งครบทุกตัวแปร จึงจะได้เอาท์พุตสุดท้ายคือ  $O$  ตัวสุดท้าย ยกตัวอย่างเช่น

สมการคือ  $IP1 \text{ and } IP2 \text{ or } IP3 \text{ and } IP4$

สมการที่ทำการคำนวณลำดับแรกคือ  $IP1 \text{ and } IP2$

สมการที่เหลือคือ  $O1 \text{ or } IP3 \text{ and } IP4$

โดยที่  $O1 = IP1 \text{ and } IP2$

สมการที่ทำการคำนวณลำดับที่สองคือ  $O1 \text{ or } IP3$

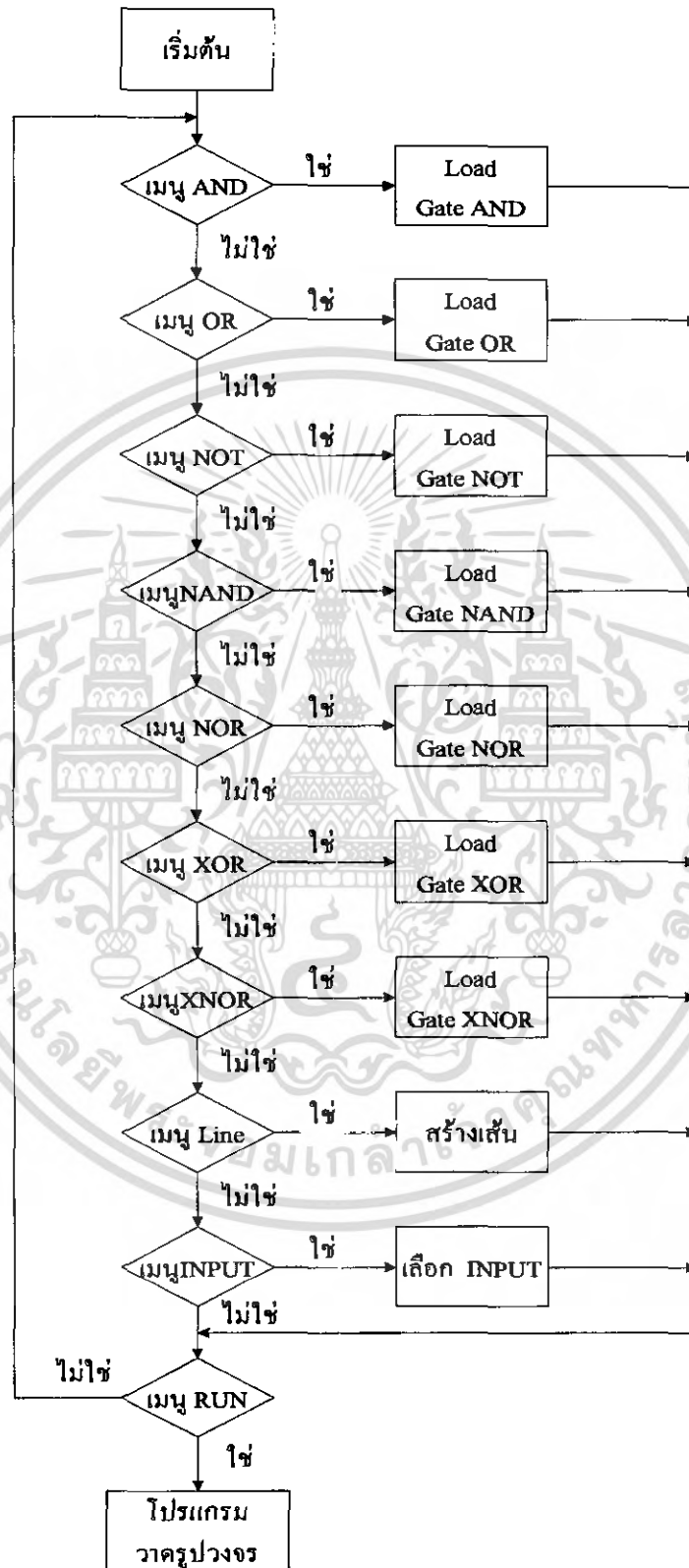
สมการที่เหลือคือ  $O2 \text{ and } IP4$

โดยที่  $O2 = O1 \text{ or } IP3$

สมการที่ทำการคำนวณลำดับสุดท้ายคือ  $O2 \text{ and } IP4$

เอาท์พุตสุดท้ายคือ  $O3$

3.2.5 โปรแกรมวาดรูปวงจร



ภาพที่ 3.8 แผนผังการทำงานของโปรแกรมวาดรูปวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## อธิบายแผนผังการทำงานของโปรแกรมวาดรูปวงจร

จากภาพที่ 3.8 สามารถแสดงรายละเอียดส่วนต่าง ๆ ของแผนผังการทำงานได้ดังนี้

1. เลือกเกตที่จะใช้ในวงจร เมนูที่ใช้เป็นอุปกรณ์ในการวาดวงจรรกสามารถแบ่งออกได้ดังนี้คือ เมนูส่วนที่ใช้ในการแสดงรูปเกิดของฟังก์ชันการทำงานต่างๆ เช่น And, Or, Not เป็นต้น เมื่อมีการกดเลือกฟังก์ชันใดๆ ในเมนูหลักหรือเมนูรูปภาพ แล้วนำเมาท์มาคลิกในบริเวณที่ต้องการวางเกตนั้นๆ ในหน้าจอหลัก โปรแกรมก็จะทำการโหลดรูปเกิดของฟังก์ชันที่เลือกมาแสดงตรงจุดที่ผู้ใช้กด และในขณะที่เดียวกันโปรแกรมก็จะสร้างออบเจกต์ขึ้นเพื่อทำการเก็บข้อมูลต่างๆที่จำเป็นในการสร้างสมการดังนี้

1.1 ชื่อของเกตซึ่งจะเก็บอยู่ในรูปแบบของตัวแปร CString เช่น จากภาพที่ 3.9 ออบเจกต์นี้จะเก็บชื่อของเกต คือ AND ไว้ในตัวแปร CString



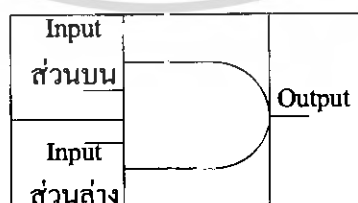
ภาพที่ 3.9 เกต AND ซึ่งออบเจกต์มีสมการ IP1 and IP2

1.2 พื้นที่สี่เหลี่ยมจำนวนสามพื้นที่ ดังนี้

1.2.1 พื้นที่สี่เหลี่ยมของอินพุตส่วนบน จะในรูปแบบของตัวแปร CRect ชื่อ "IP1Area" ซึ่งตัวแปรนี้เป็นส่วนประกอบของออบเจกต์ของ Class CLogic

1.2.2 พื้นที่สี่เหลี่ยมของอินพุตส่วนล่าง จะในรูปแบบของตัวแปร CRect ชื่อ "IP2Area" ซึ่งตัวแปรนี้เป็นส่วนประกอบของออบเจกต์ของ Class CLogic

1.2.3 พื้นที่สี่เหลี่ยมของเอาท์พุต จะในรูปแบบของตัวแปร CRect ชื่อ "OPArea" ซึ่งตัวแปรนี้เป็นส่วนประกอบของออบเจกต์ของ Class CLogic ซึ่งทั้งสามพื้นที่ที่สามารถแสดงได้ดังภาพที่ 3.10



ภาพที่ 3.10 พื้นที่ของออบเจกต์

### 1.3 ออบเจกต์ตัวที่มาก่ออยู่กับบริเวณพื้นที่สี่เหลี่ยมทั้งสามพื้นที่ตามข้อ 1.2

1.3.1 ออบเจกต์ตัวที่มาก่ออยู่กับพื้นที่สี่เหลี่ยมของอินพุตส่วนบน โปรแกรมจะดูจากจุดเริ่มต้นของเส้นเชื่อมระหว่างตัวเกดทั้งสอง ว่าอยู่ในพื้นที่เอ้าท์พุทของเกดตัวใดแล้วเกดตัวนั้นก็จะเป็นออบเจกต์ที่มาก่ออยู่กับพื้นที่สี่เหลี่ยมของอินพุตส่วนบน

1.3.2 ออบเจกต์ตัวที่มาก่ออยู่กับพื้นที่สี่เหลี่ยมของอินพุตส่วนล่าง โปรแกรมจะดูจากจุดเริ่มต้นของเส้นเชื่อมระหว่างตัวเกดทั้งสอง ว่าอยู่ในพื้นที่เอ้าท์พุทของเกดตัวใดแล้วเกดตัวนั้นก็จะเป็นออบเจกต์ที่มาก่ออยู่กับพื้นที่สี่เหลี่ยมของอินพุตส่วนล่าง

1.3.3 ออบเจกต์ตัวที่มาก่ออยู่กับพื้นที่สี่เหลี่ยมของเอ้าท์พุท โปรแกรมจะดูจากจุดปลายของเส้นเชื่อมระหว่างตัวเกดทั้งสอง ว่าอยู่ในพื้นที่อินพุทของเกดตัวใด แล้วเกดตัวนั้นก็จะเป็นออบเจกต์เอ้าท์พุท

## 2. ลากเส้น เมื่อกดเมนู Line โปรแกรมจะใช้คำสั่ง

MoveTo(BeginPoint)

LineTo(EndPoint)

โดย BeginPoint เป็นจุดเริ่มต้นของเส้นซึ่งจุดนี้จะต้องอยู่ในพื้นที่เอ้าท์พุทในข้อ 1.2 ส่วน EndPoint เป็นจุดปลายของเส้นซึ่งจุดนี้จะต้องอยู่ในพื้นที่อินพุทในข้อ 1.2 และทั้ง BeginPoint และ EndPoint เป็นตัวแปรแบบ CPoint

เมื่อมีเส้นมาต่อกันระหว่างเกดแต่ละตัวแล้ว โปรแกรมจะเก็บข้อมูลต่างๆไว้ในออบเจกต์แต่ละตัว โดยออบเจกต์นี้จะเก็บอยู่ในอาร์เรย์

3. เลือกตัวแปร IP1-IP9 เมื่อกดเมนู SetInput โปรแกรมจะมีตัวแปรอินพุทให้เลือกคือ IP1, IP2, IP3, IP4, IP5, IP6, IP7, IP8, IP9

4. เมื่อวาดวงจรเสร็จเรียบร้อยแล้วเมนู Run โปรแกรมจะทำการประมวลผลเพื่อตรวจสอบความถูกต้องดังนี้

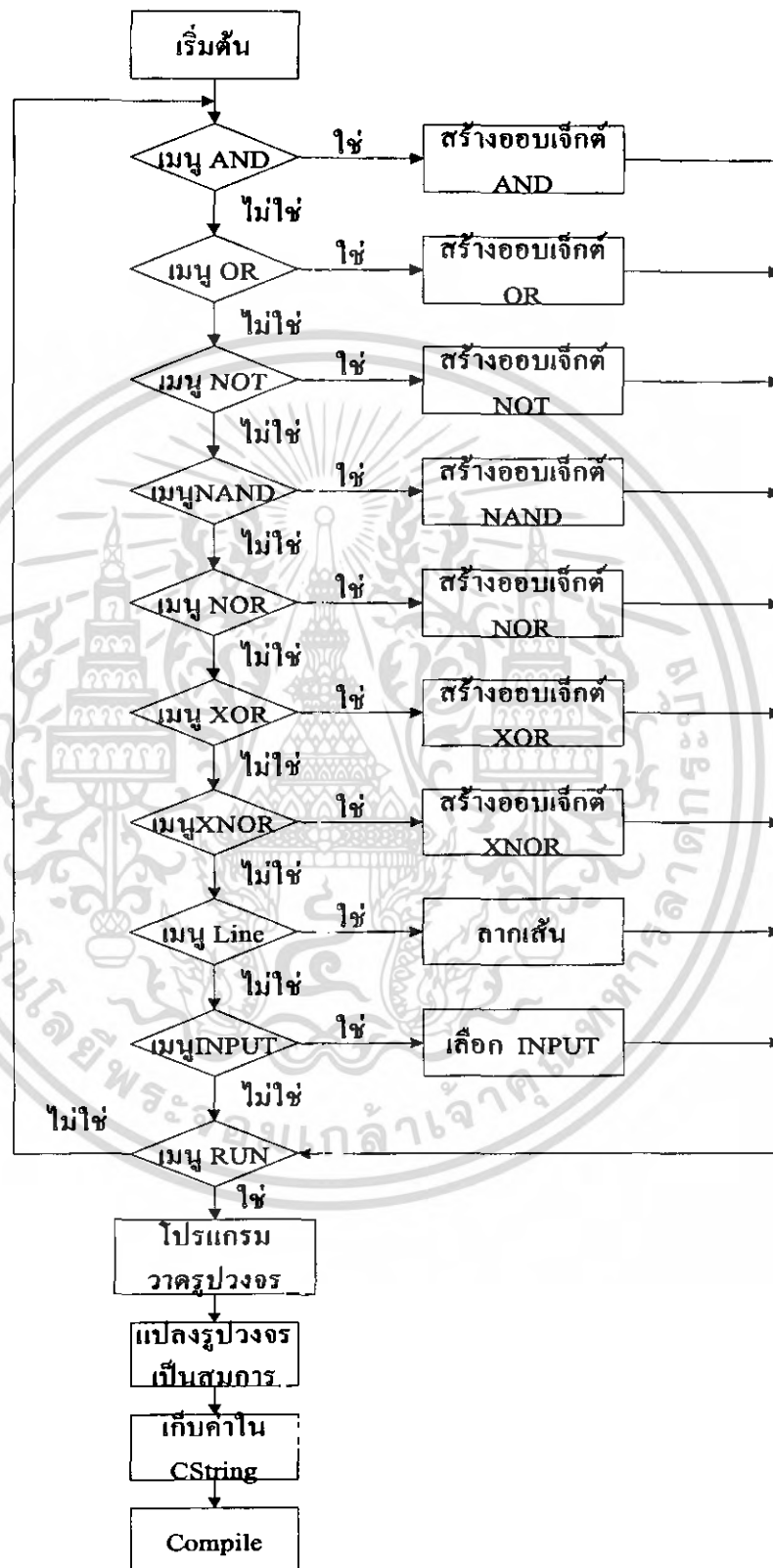
4.1 ขาอินพุทของเกดทุกตัวจะต้องไม่ปล่อยว่างไว้ กล่าวคืออินพุทของเกดจะต้องเป็น IP1-IP9 หรือจะต้องมีเส้นมาจากเกดตัวอื่น และการวาง IP1-IP9 จะต้องอยู่ในพื้นที่ที่กำหนดเท่านั้น

4.2 การลากเส้นที่จุดเริ่มต้นและจุดสุดท้ายจะต้องอยู่ภายในพื้นที่อินพุท หรือเอ้าท์พุทเท่านั้น

4.3 เกดตัวที่เป็นเอ้าท์พุทจะต้องมีเพียงหนึ่งตัวเท่านั้น

4.4 รายละเอียดของการใช้โปรแกรมสามารถดูได้จากภาคผนวก

### 3.2.6 โปรแกรมแปลงรูปวงจรเป็นสมการ



เอกสารนี้เป็นเอกสารที่ภาพที่ 3.11 แผนผังการทำงานของโปรแกรมแปลงรูปวงจรเป็นสมการ โยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### อธิบายแผนผังการทำงานของโปรแกรมแปลงรูปวงจรถ้าเป็นสมการ

จากภาพที่ 3.11 สามารถแสดงรายละเอียดส่วนต่าง ๆ ของแผนผังการทำงาน ได้ดังนี้

1. เมื่อมีการเลือกรูปเกดที่จะใช้ในวงจร โปรแกรมสร้างออบเจกต์ของเกดที่เลือกแล้วเก็บไว้ในหน่วยความจำ
2. เมื่อมีการวาดเส้นจากเกดตัวหนึ่งไปยังเกดอีกตัวหนึ่ง เกดตัวหลังจะมองเกดตัวแรกว่าเป็นออบเจกต์อินพุตของมัน โดยจะเก็บออบเจกต์ของเกดตัวแรกไว้ในส่วนอินพุตของออบเจกต์ตัวหลัง และเกดตัวแรกจะมองเกดตัวหลังว่าเป็นออบเจกต์เอาต์พุตของมัน โดยจะเก็บออบเจกต์ของเกดตัวหลังไว้ในส่วนเอาต์พุตของออบเจกต์ตัวแรก
3. เมื่อมีการกำหนดชื่ออินพุต IP1 - IP9 ให้กับรูปวงจรถ้าผู้ใช้ต้องการแล้ว เกดตัวที่ถูกใส่ชื่ออินพุตแล้วเกดตัวนั้นจะรู้ว่าพื้นที่อินพุตซึ่งเป็นส่วนประกอบหนึ่งในออบเจกต์มีการใส่อินพุตในพื้นที่นั้นแล้ว และจะเก็บชื่ออินพุตนี้ไว้ในส่วนอินพุตของออบเจกต์
4. เมื่อมีการกดปุ่มรัน (Run) โปรแกรมจะมีการประมวลผลเพื่อหาสมการของรูปวงจรถ้าผู้ใช้ป้อน

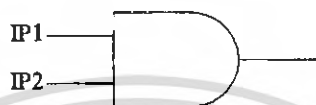




### อธิบายแผนผังการทำงานของโปรแกรมสร้างสมการ

จากภาพที่ 3.12 สามารถแสดงรายละเอียดส่วนต่าง ๆ ของแผนผังการทำงานได้ดังนี้

1. เริ่มแรกต้องหาจำนวนเกตที่มีอยู่ทั้งหมด ซึ่งหาได้จากสมาชิกทั้งหมดที่มีอยู่ในอาร์เรย์ (Array) จากนั้นก็นำจำนวนเกตมาเป็นตัวกำหนดจำนวนครั้งในการวนรูปหาเกตตัวที่ยังมีขว่างอยู่ โดยไม่ได้มีเส้นต่อมาจากเกตตัวอื่น ๆ เมื่อรู้แล้วก็สร้างสมการของเกตตัวนั้น โดยเก็บเป็นตัวแปรหนึ่งในออบเจกต์ตัวนั้น ๆ เช่น

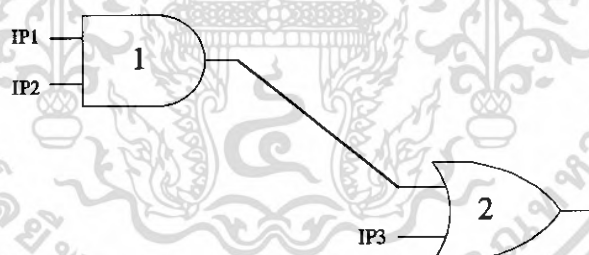


ภาพที่ 3.13 IP1 and IP2

จากภาพที่ 3.13 จะได้สมการ IP1 and IP2

2. เมื่อได้สมการของตัวเกตแล้ว โปรแกรมจะทำการวนรูปเพื่อหาเกตตัวที่ยังไม่มีสมการ ซึ่งจะมีกรณีที่เป็นไปได้ดังนี้

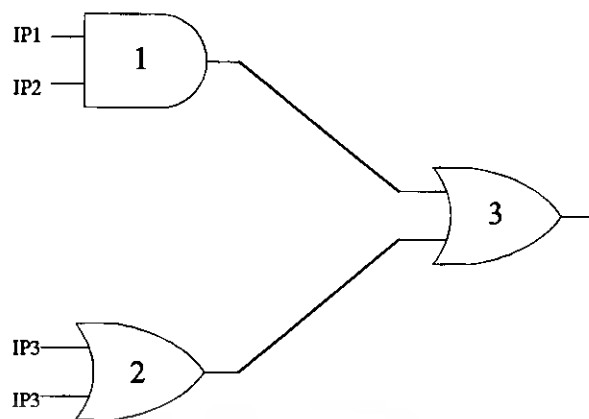
2.1 ขาดขาหนึ่งของเกตตัวที่ยังไม่มีสมการ (ตัวที่2) มีเส้นต่อมาจากเกตตัวที่มีสมการแล้ว และอีกขาถูกใส่ค่าอินพุต IP1-IP9 เช่น



ภาพที่ 3.14 IP1andIP2orIP3

จากภาพที่ 3.14 ตัวแปรชื่อ Equation ของ Object เกตตัวที่ 1 จะเก็บสมการ IP1 and IP2 เพราะฉะนั้นสมการของภาพที่ 3.14 จะเก็บอยู่ในตัวแปรชื่อ Equation ของ Object เกตตัวที่ 2

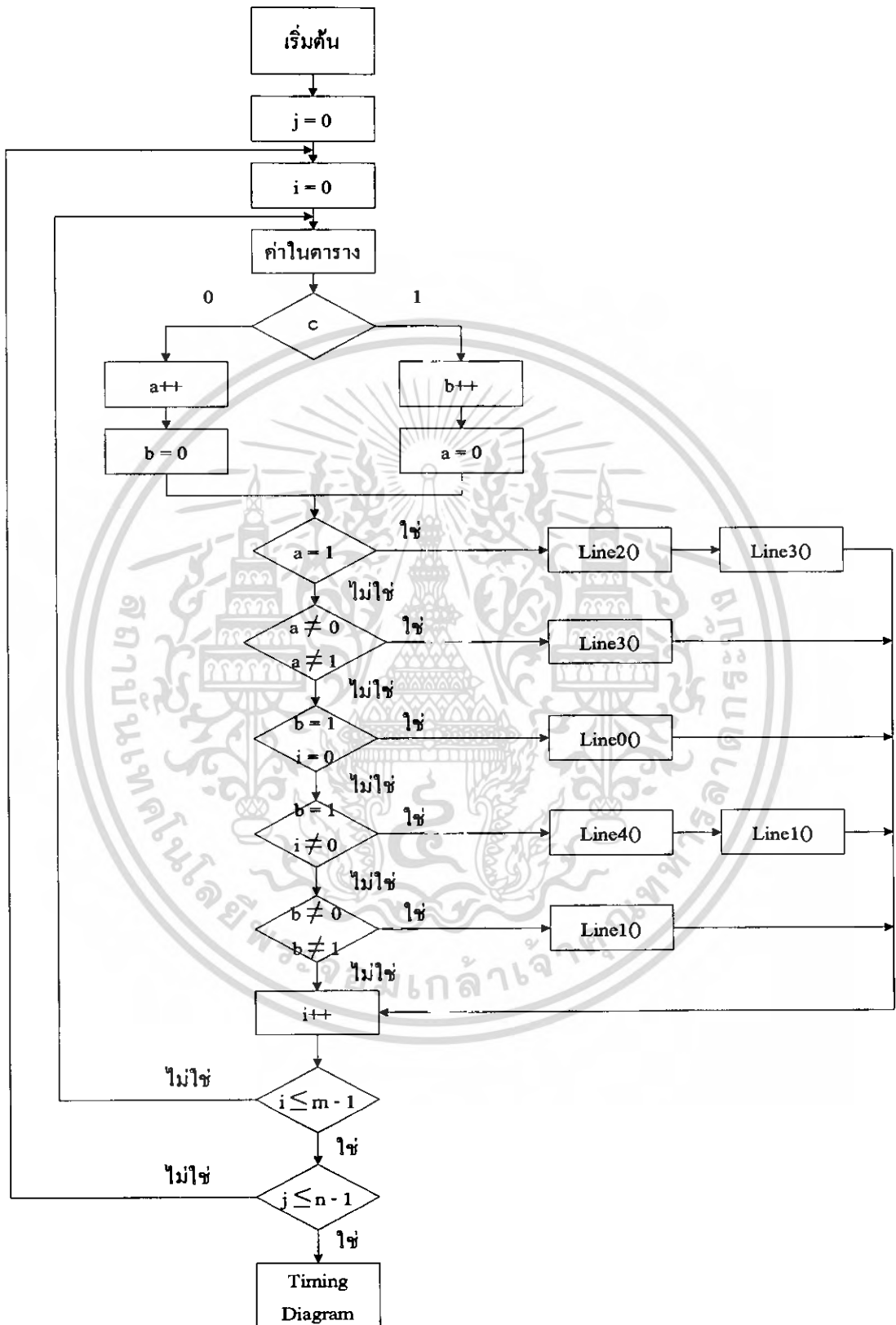
2.2 ทั้งสองขาของเกตตัวที่ยังไม่มีสมการ มีเส้นต่อมาจากขาของเกตที่มีสมการแล้ว เช่น



ภาพที่ 3.15  $(IP1 \text{ and } IP2) \text{ or } (IP3 \text{ and } IP4)$

จากภาพที่ 3.15 ตัวแปรชื่อ Equation ของ Object เกิดตัวที่ 1 จะเก็บสมการ IP1 and IP2 ส่วน Equation ของ Object เกิดตัวที่ 2 จะเก็บสมการ IP3 and IP4 เพราะฉะนั้นสมการของภาพที่ 3.15 จะเก็บอยู่ในตัวแปรชื่อ Equation ของ Object เกิดตัวที่ 3 คือ  $(IP1 \text{ and } IP2) \text{ or } (IP3 \text{ and } IP4)$

3.2.8 โปรแกรมสร้างแผนภูมิเวลา

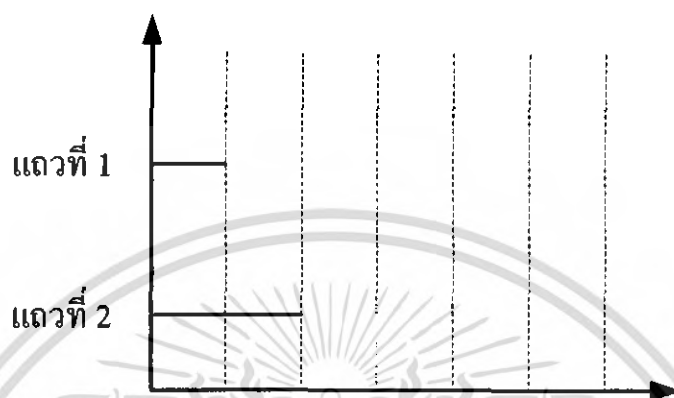


ภาพที่ 3.16 แผนผังการทำงานของโปรแกรมสร้างแผนภูมิเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### อธิบายแผนผังการทำงานของโปรแกรมสร้างแผนภูมิเวลา

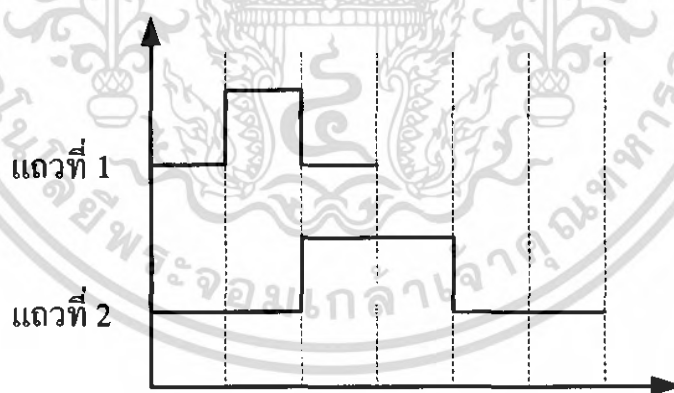
จากภาพที่ 3.16 สามารถแสดงรายละเอียดส่วนต่าง ๆ ของแผนผังการทำงานได้ดังนี้  
`line0()` จะทำหน้าที่ วาดเส้นเริ่มต้นของแผนภูมิเวลาแต่ละเส้น โดยวาดเส้นแนวนอนในสภาวะ 0 เช่น



ภาพที่ 3.17 ผลของเอ้าท์พุท `line0()`

โดยเส้นสีแดงคือเส้นที่แต่ละฟังก์ชันวาดขึ้นมา

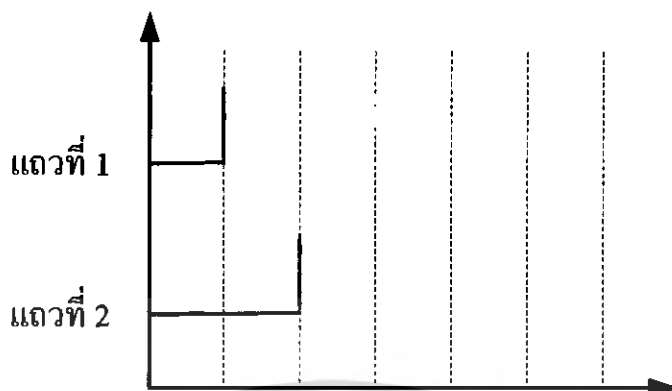
`line1()` จะทำหน้าที่วาดเส้นแนวนอนในสภาวะ 0 ที่ไม่ใช่เส้นเริ่มต้น เช่น



ภาพที่ 3.18 ผลของเอ้าท์พุท `line1()`

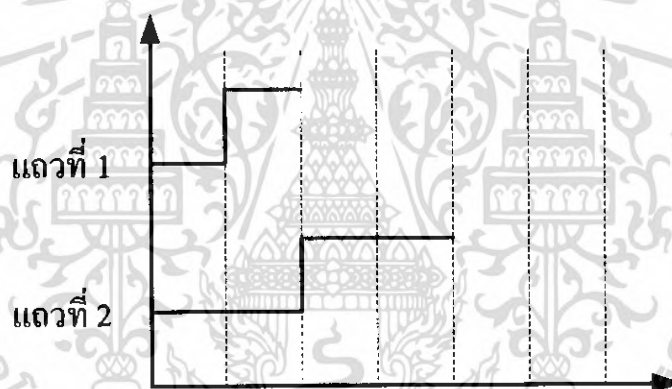
โดยการเรียกฟังก์ชัน `line1()` จะต้องส่งค่าความกว้างให้กับฟังก์ชันด้วย

line2() จะทำหน้าที่วาดเส้นในแนวดิ่งโดยวาดเส้นจากสภาวะ 0 ไปสู่สภาวะ 1 เช่น



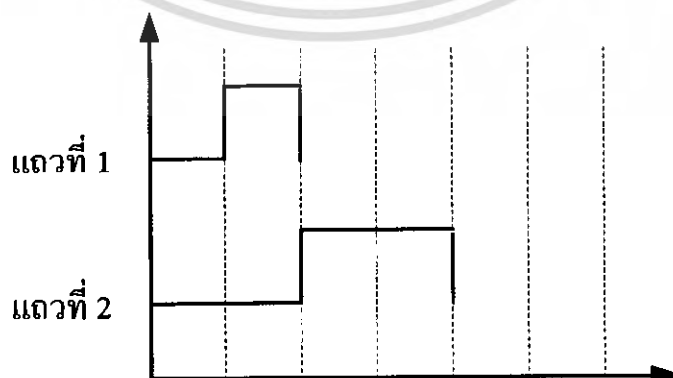
ภาพที่ 3.19 ผลของเอาต์พุต line2()

line3() จะทำหน้าที่วาดเส้นแนวนอนในในสภาวะ 1 เช่น



ภาพที่ 3.20 ผลของเอาต์พุต line3()

line4() จะทำหน้าที่วาดเส้นในแนวดิ่งโดยวาดเส้นจากสภาวะ 1 ไปสู่สภาวะ 0 เช่น



ภาพที่ 3.21 ผลของเอาต์พุต line4()

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

### การทดลองและผลการทดลอง

#### 4.1 การทดลอง

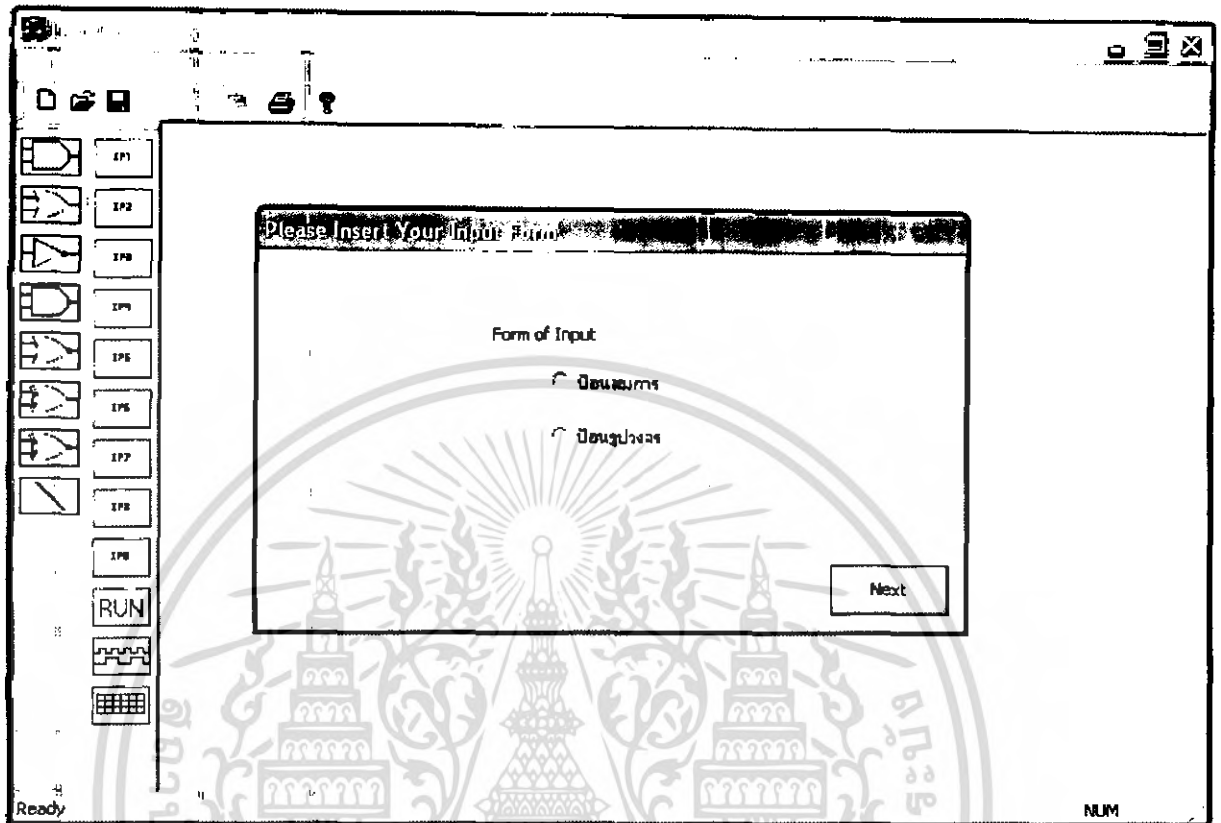
เมื่อเขียนโปรแกรมจำลองการทำงานวงจรตรรกบนโปรแกรม Microsoft Visual C++ เสร็จแล้วเราก็จะทำการทดสอบโปรแกรมที่เขียนขึ้นมาว่า ค่าเอาต์พุตจากรูปวงจรตรรกนั้นถูกต้องตามทฤษฎีที่กล่าวมาแล้วในบทที่ 2 หรือไม่ และยังดูว่าโปรแกรมมีข้อผิดพลาดตรงไหนบ้าง เพื่อทำการแก้ไขโปรแกรมให้สมบูรณ์มากที่สุด

#### 4.2 ผลการทดลอง

โดยในการทดสอบโปรแกรม เพื่อจะดูว่าโปรแกรมที่เขียนขึ้นมาสามารถประมวลผลได้ตามออกแบบไว้หรือไม่ โดยสามารถดูได้จากส่วนที่จะกล่าวต่อไปนี้ คือ

- หน้าจอใช้งาน
- เมนู View
- การเคลื่อนย้ายเมนูรูปเกต และเมนูอินพุต
- เมนู Tool
- การป้อนสมการตรรก
- การวาดวงจรถรก
- ตัวอย่างการจำลองการทำงานวงจรถรก
  - เมื่อมีการป้อนอินพุตที่มีเกตแอนด์
  - เมื่อมีการป้อนอินพุตที่มีเกตออร์
  - เมื่อมีการป้อนอินพุตที่มีเกตนอด
  - เมื่อมีการป้อนอินพุตที่มีเกตแนนด์
  - เมื่อมีการป้อนอินพุตที่มีเกตนอร์
  - เมื่อมีการป้อนอินพุตที่มีเกตเอกซ์คลูซีฟออร์
  - เมื่อมีการป้อนอินพุตที่มีเกตเอกซ์คลูซีฟนอร์
  - เมื่อมีการนำเกตมาต่อเป็นวงจร

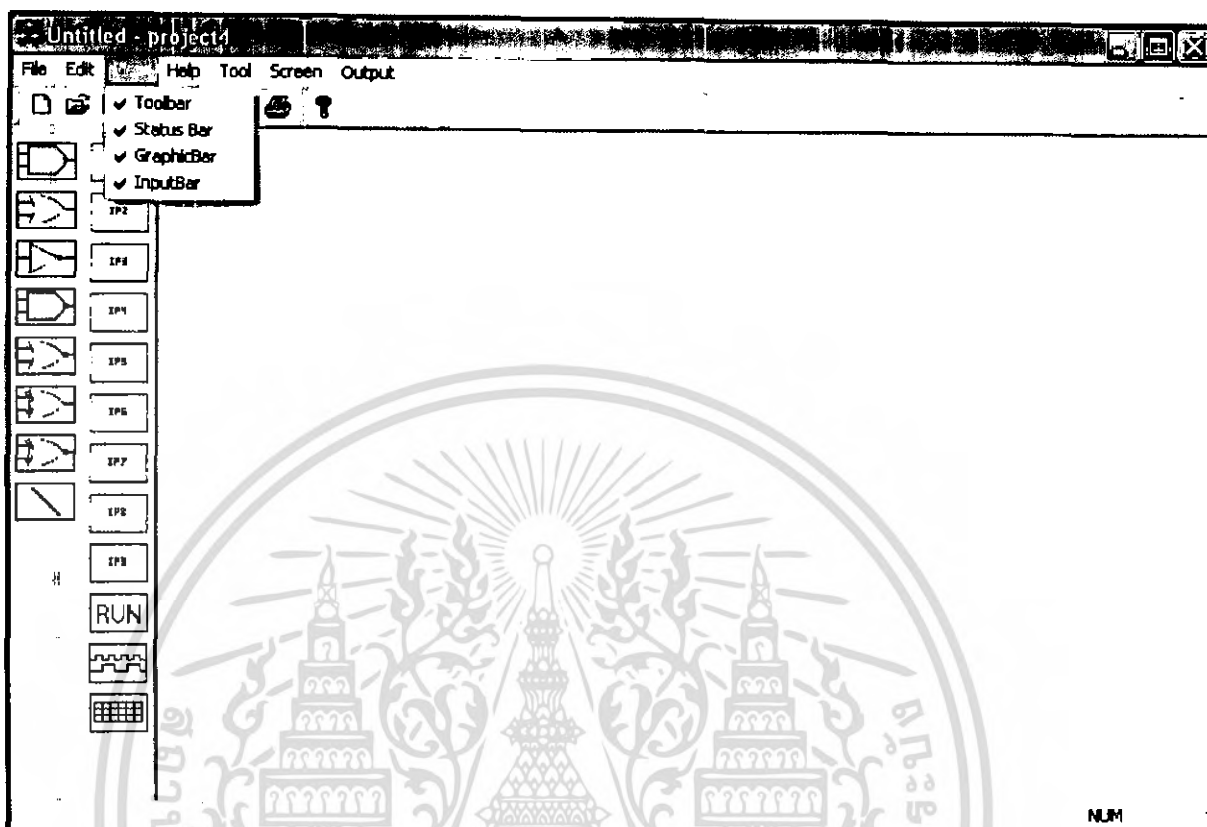
#### 4.2.1 หน้าจอใช้งาน



ภาพที่ 4.1 หน้าจอใช้งาน

จากภาพที่ 4.1 เป็นหน้าจอที่ใช้สำหรับวาดภาพวงจรตรรกโดยที่สามารถเลือกเกิดที่ต้องการได้จาก เมนู ToolBar คือเมนูรูปเกดที่อยู่ด้านข้างทางซ้ายมือ หรือเมนู Tool คือเมนูที่อยู่ด้านบน ซึ่งหน้าจอที่ใช้ในการวาดรูปวงจรนี้สร้างจากฟังก์ชันการทำงานของโปรแกรม Microsoft Visual C++ เมื่อต้องการลบภาพก็สามารถทำได้โดยเลือกที่เมนู Screen ซึ่งจะมีคำสั่ง Clear อยู่เมื่อเลือกที่เมนูนี้ จะทำให้หน้าจอถูกลบทั้งหมดให้หน้าจอหลักเป็นสีขาวเหมือนตอนเริ่มใช้งาน

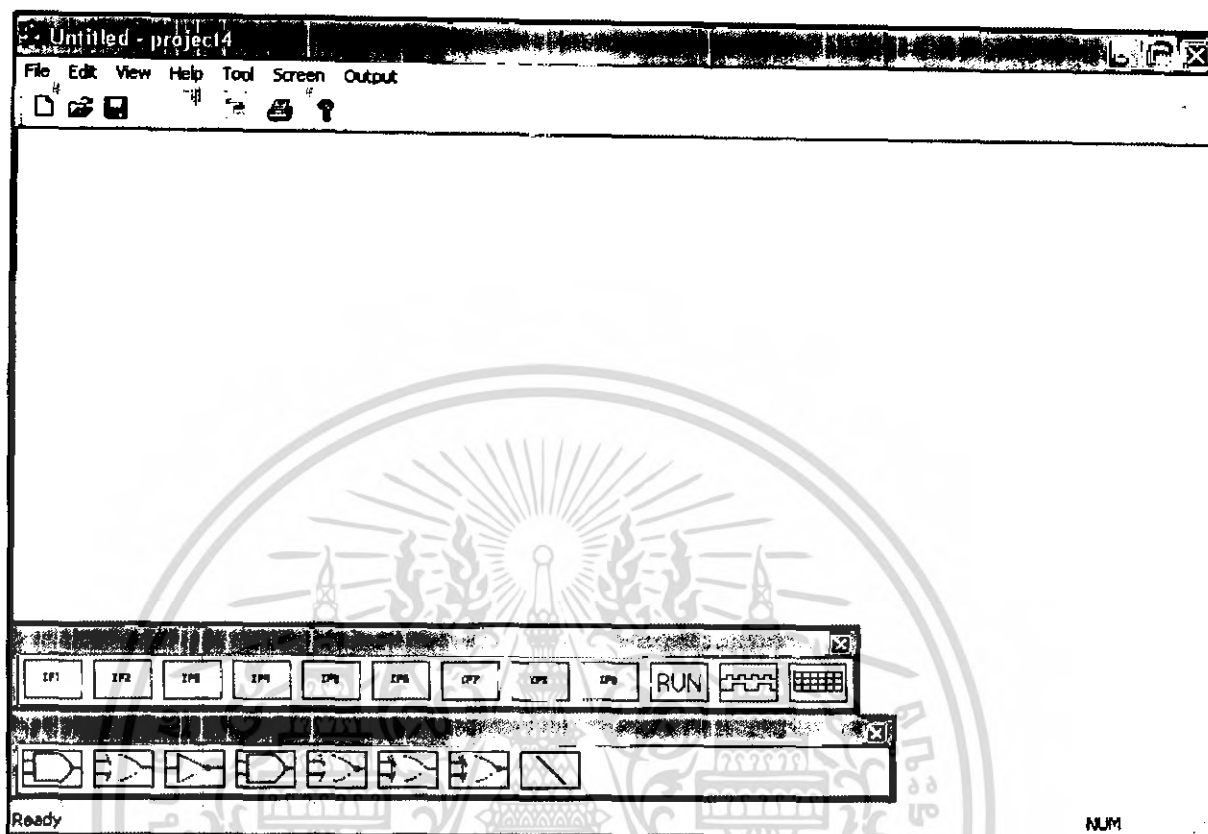
## 4.2.2 เมนู View



ภาพที่ 4.2 เมนู View

จากภาพที่ 4.2 เมื่อผู้ใช้เลือกการป้อนอินพุตแบบป้อนรูปวงจรถัดไป แล้วเลือกเมนู View จะมีเมนูแสดงออกมาให้สามารถเลือกเปิดหรือปิดได้ โดยเลือกที่ชื่อของเมนูนั้นๆ สามารถดูได้จากเครื่องหมายถูกที่อยู่หน้าชื่อเมนูนั้นๆ ถ้ามีเครื่องหมายถูกอยู่แสดงว่าเมนูนั้นถูกแสดงไว้ที่หน้าจอใช้งานเรียบร้อยแล้ว

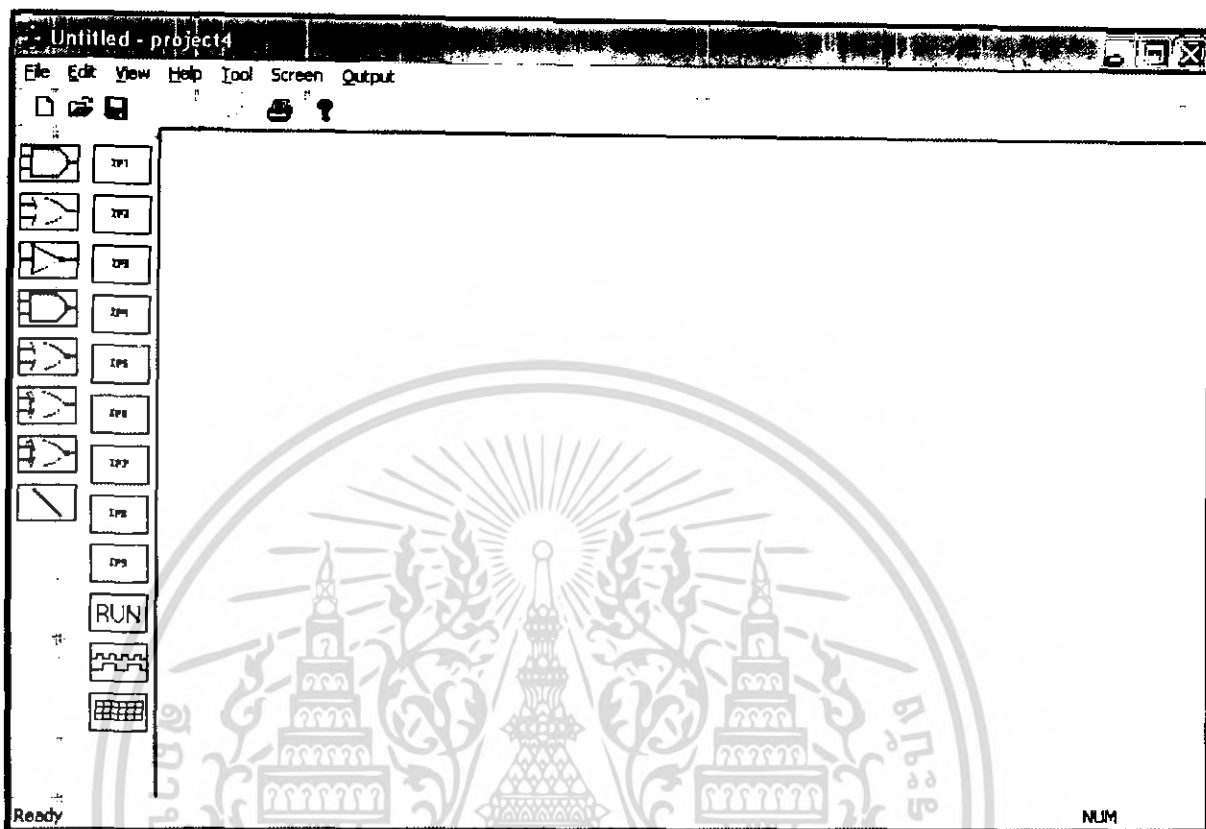
### 4.2.3 การเคลื่อนย้ายเมนูรูปเกต และเมนูอินพุต



ภาพที่ 4.3 การเคลื่อนย้ายเมนู และเมนูอินพุต

จากภาพที่ 4.3 เป็นการแสดงให้เห็นว่าเมนูรูปเกต และเมนูอินพุตสามารถเคลื่อนย้ายได้บนหน้าจอหลัก โดยยังมีคุณสมบัติในการทำงานเหมือนเดิมไม่ว่าจะนำทั้งสองเมนูนี้ไปวางที่ไหนก็ตาม

#### 4.2.4 เมนู Tool

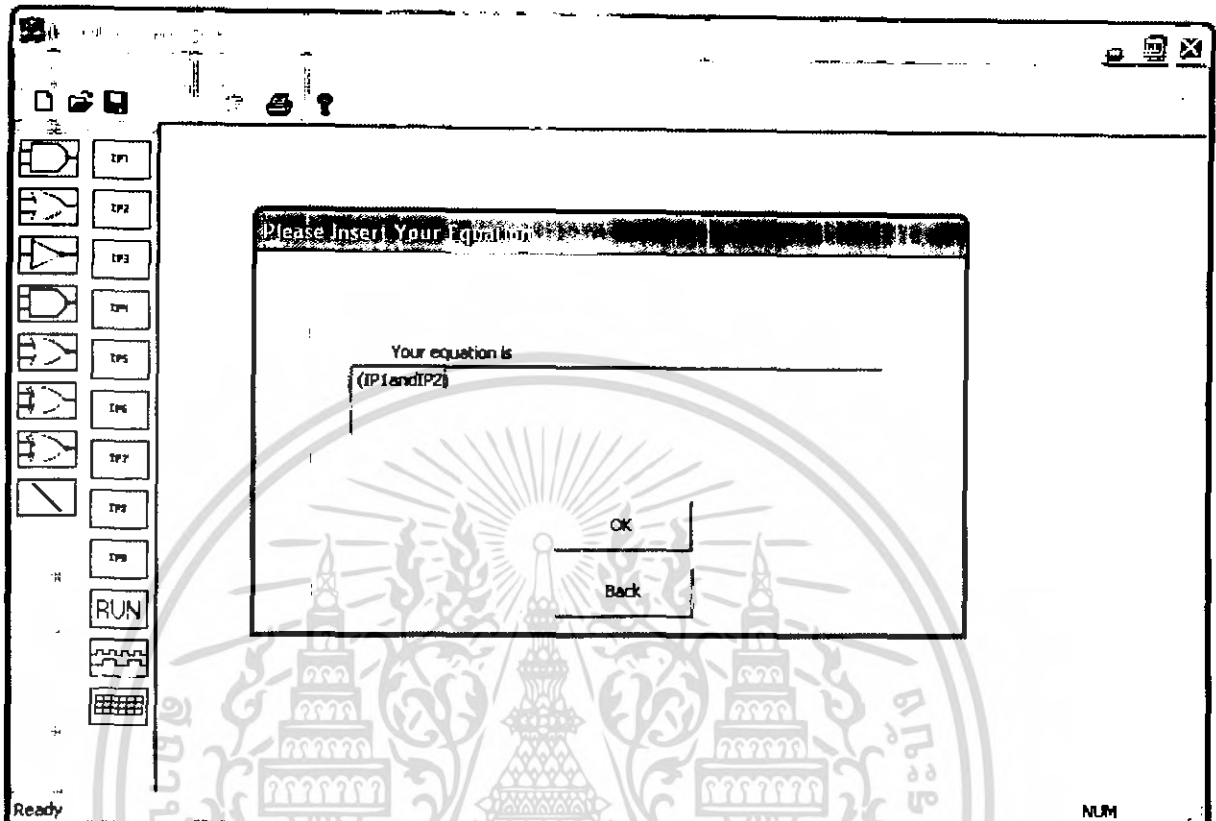


ภาพที่ 4.4 เมนูใช้งาน

จากภาพที่ 4.4 เป็นหน้าจอที่แสดงเมนู Tool คือเมื่อมีการกดเลือกเมนูนี้ แล้วจะทำให้ผู้ใช้สามารถที่เลือกฟังก์ชันเกดที่ต้องการได้ และจะให้ผลเช่นเดียวกันกับการเลือกเครื่องมือจากเมนูรูปเกดที่อยู่ทางด้านซ้ายมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.2.5 การป้อนสมการตรรก



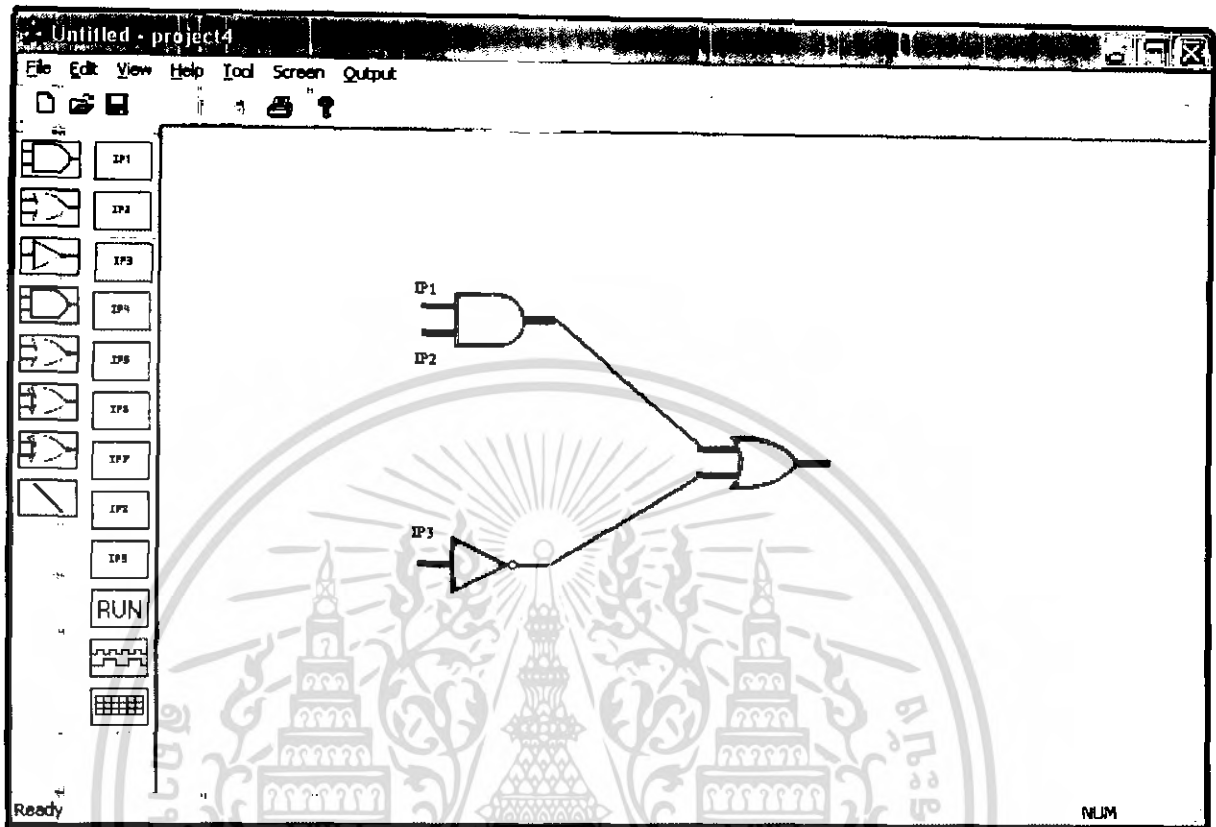
ภาพที่ 4.5 ตัวอย่างการป้อนสมการตรรก

วิธีการป้อนสมการที่ถูกต้อง

1. ผู้ใช้จะต้องป้อนสมการที่ต้องการประมวลผลลงในเท็กซ์บ็อกซ์
2. ผู้ใช้ต้องกดปุ่ม OK
3. ผู้ใช้ต้องกดปุ่ม Run ในเมนู Output หรือในเมนูภาพเพื่อประมวลผล
4. ถ้าผู้ใช้ต้องการเอาท์พุตเป็นตารางค่าความจริงผู้ใช้ต้องกดปุ่มเมนู TrueTable ในเมนู Output หรือในเมนูภาพ
5. ถ้าผู้ใช้ต้องการเอาท์พุตเป็นแผนภูมิเวลาผู้ใช้ต้องกดปุ่มเมนู TimingDiagram ในเมนู Output หรือในเมนูภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.2.6 การวาดรูปวงจรตรรก



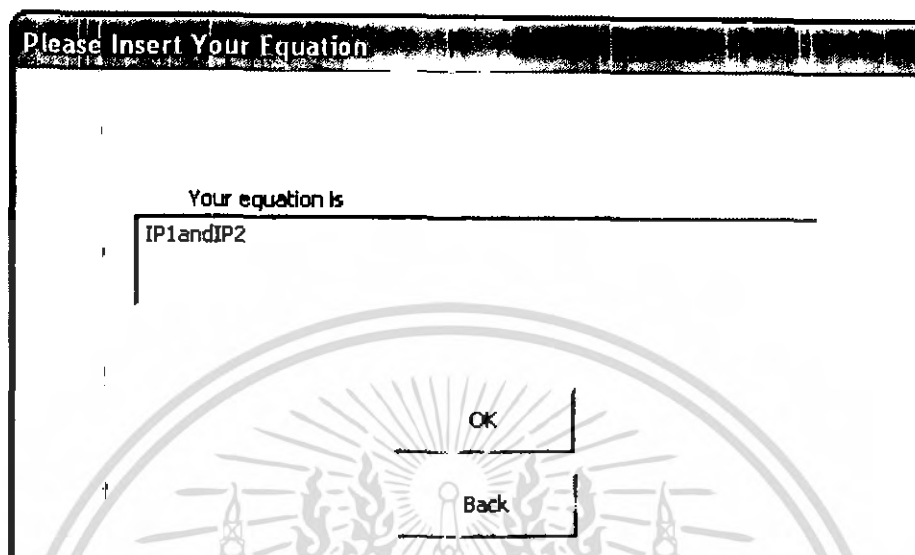
ภาพที่ 4.6 ตัวอย่างการวาดรูปวงจรตรรก

#### วิธีการวาดวงจรที่ถูกต้อง

1. ผู้ใช้ต้องกดปุ่มเมนู Tool หรือเมนูภาพเกต (ToolBar) เพื่อเลือกฟังก์ชันการทำงานของเกตที่ต้องการ แล้วเลือกจุดที่จะวางรูปบนหน้าจอใช้งาน
2. ผู้ใช้จะต้องลากเส้นจากเอาต์พุตของเกตตัวหนึ่ง ไปยังอินพุตของเกตอีกตัวหนึ่ง การลากเส้นนั้นจะต้องกดปุ่มเมนู Tool ก่อนแล้วเลือกฟังก์ชัน line หรือเมนูภาพเส้น จากนั้นจะลากเส้นได้ โดยการคลิกเมาท์ 1 ครั้งที่จุดเริ่มต้น และดับเบิลคลิกเมาท์ที่จุดสุดท้ายของเส้นที่ต้องการ
3. ผู้ใช้จะต้องป้อนอินพุต IP1 - IP9 ใส่ที่ขาอินพุตของเกตที่ต้องการตามรูปแบบวงจรที่ต้องการ

#### 4.2.7 ตัวอย่างการจำลองการทำงานวงจรตรรก

##### 4.2.7.1 เมื่อมีการป้อนอินพุตที่มีเกตแอนด์

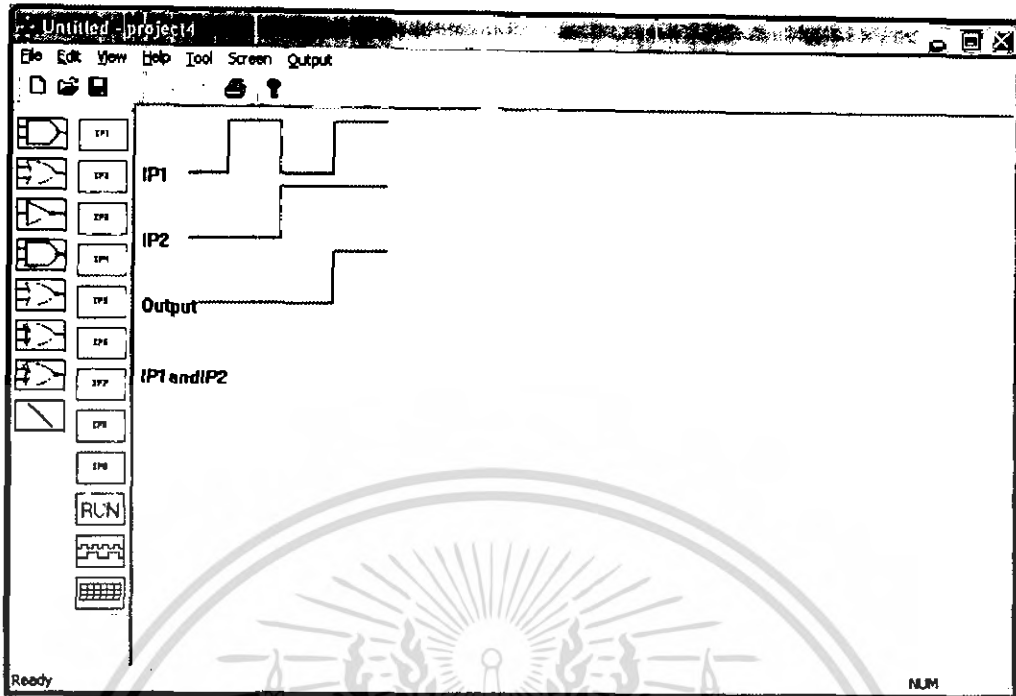


ภาพที่ 4.7 ตัวอย่างการจำลองการทำงานวงจรตรรกของการแอนด์โดยการป้อนสมการ



ภาพที่ 4.8 ตัวอย่างการจำลองการทำงานวงจรตรรกของการแอนด์โดยการป้อนรูปวงจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.9 เอาท์พุทของผลการทดลองในรูปแบบของแผนภูมิเวลา



ภาพที่ 4.10 เอาท์พุทของผลการทดลองในรูปแบบของตารางค่าความจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการทดลองพบว่าเมื่อมีการป้อนอินพุตในรูปแบบของสมการ หรือรูปวงจรรวมของเกตแอนด์แล้ว ค่าเอาต์พุตที่ได้ทั้งตารางค่าความจริงและแผนภูมิเวลานั้นมีค่าตามตารางที่ 4.1 เมื่อเปรียบเทียบกับตารางค่าความจริงของเกตแอนด์ (ตารางที่ 2.2) ในบทที่ 2 แล้วได้ค่าตรงกัน แสดงว่าค่าที่ได้จากการทดลองนั้นถูกต้องตามทฤษฎีทางตรรกศาสตร์

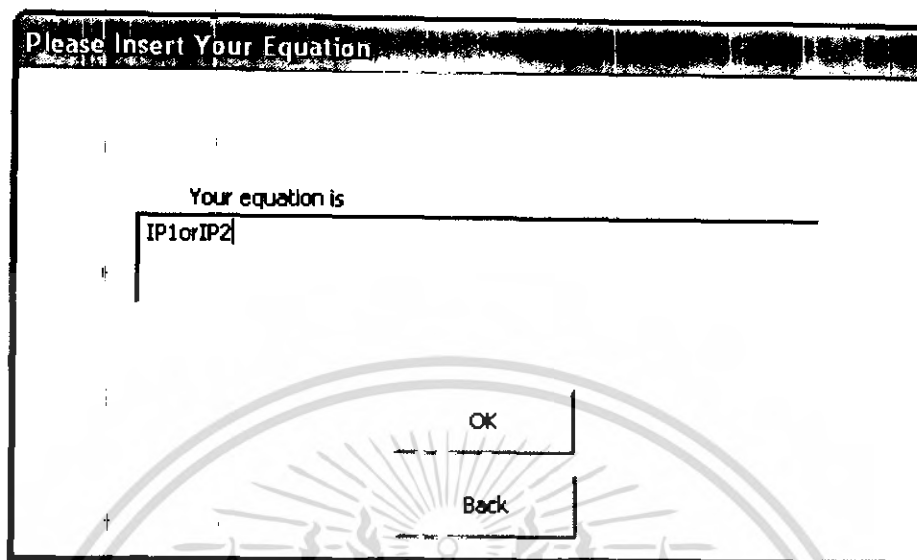
ตารางที่ 4.1 ตารางค่าความจริงของการแอนด์

IP1	IP2	$Y = IP1 \cdot IP2$
0	0	0
0	1	0
1	0	0
1	1	1



ภาพที่ 4.11 สัญลักษณ์ของเกตแอนด์ที่ใช้ในการวาดรูปวงจรรวม

### 4.2.7.2 เมื่อมีการป้อนอินพุตที่มีเกตออร์

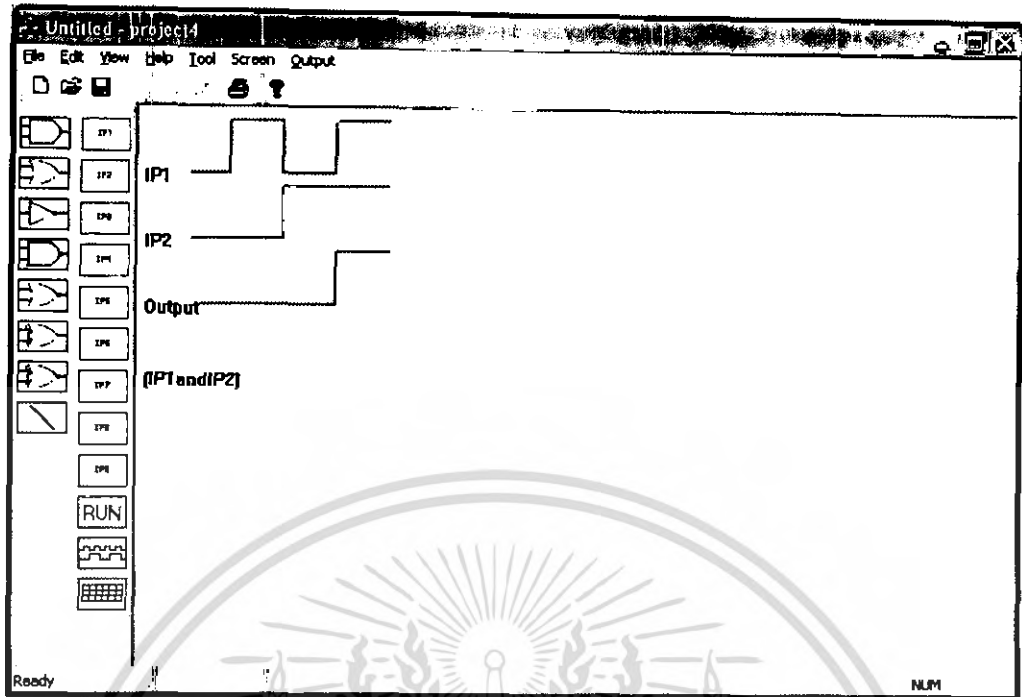


ภาพที่ 4.12 ตัวอย่างการจำลองการทำงานวงจรตรรกะของการออร์โดยการป้อนสมการ



ภาพที่ 4.13 ตัวอย่างการจำลองการทำงานวงจรตรรกะของการออร์โดยการป้อนสมการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.14 เอาท์พุทของผลการทดลองในรูปแบบของแผนภูมิเวลา



ภาพที่ 4.15 เอาท์พุทของผลการทดลองในรูปแบบของตารางค่าความจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการทดลองพบว่าเมื่อมีการป้อนอินพุตในรูปแบบของสมการ หรือรูปวงจรรวมของเกตออร์แล้ว ค่าเอาต์พุตที่ได้ทั้งตารางค่าความจริงและแผนภูมิเวลานั้นมีค่าตามตารางที่ 4.2 เมื่อเปรียบเทียบกับตารางค่าความจริงของเกตออร์ (ตารางที่ 2.3) ในบทที่ 2 แล้วได้ค่าตรงกัน แสดงว่าค่าที่ได้จากการทดลองนั้นถูกต้องตามทฤษฎีทางตรรกศาสตร์

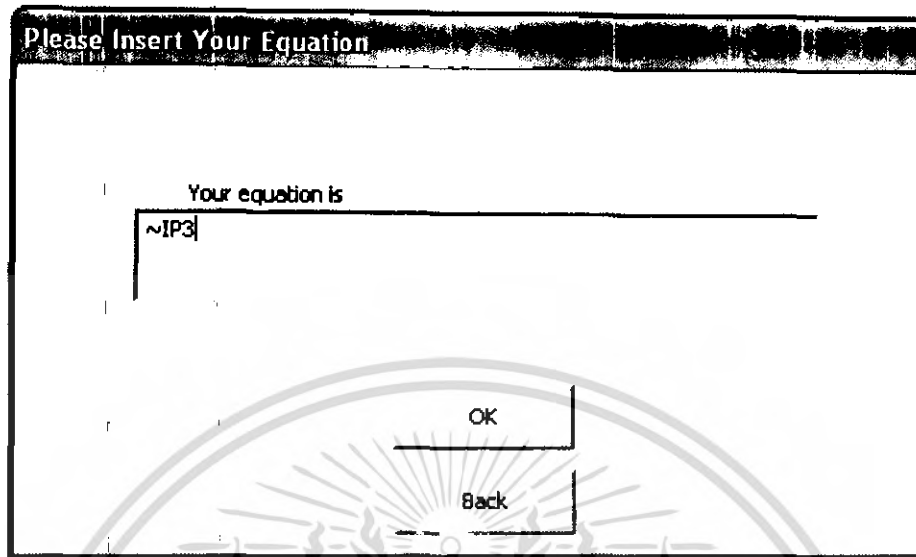
ตารางที่ 4.2 ตารางค่าความจริงของการออร์

IP1	IP2	Y = IP1 + IP2
0	0	0
0	1	1
1	0	1
1	1	1



ภาพที่ 4.16 สัญลักษณ์ของเกตออร์ที่ใช้ในการวาดรูปวงจรรวม

### 4.2.7.3 เมื่อมีการป้อนอินพุตที่มีเกตนอต

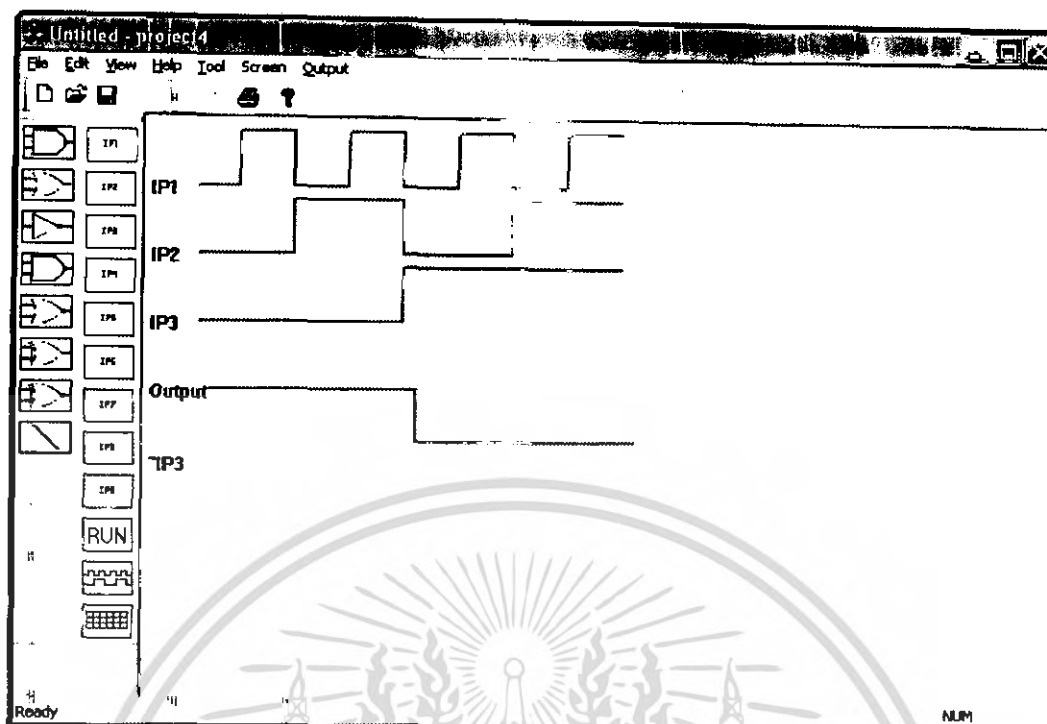


ภาพที่ 4.17 ตัวอย่างการจำลองการทำงานวงจรตรรกะของการนอตโดยการป้อนสมการ



ภาพที่ 4.18 ตัวอย่างการจำลองการทำงานวงจรตรรกะของการนอตโดยการป้อนรูปวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.19 เอาท์พุทของผลการทดลองในรูปแบบของแผนภูมิเวลา



ภาพที่ 4.20 เอาท์พุทของผลการทดลองในรูปแบบของตารางค่าความจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการทดลองพบว่าเมื่อมีการป้อนอินพุตในรูปแบบของสมการ หรือรูปวงจรรรคของ เกตนอตแล้ว ค่าเอาต์พุตที่ได้ทั้งตารางค่าความจริงและแผนภูมิเวลานั้นมีค่าตามตารางที่ 4.3 เมื่อ เปรียบเทียบกับตารางค่าความจริงของเกตนอต (ตารางที่ 2.4) ในบทที่ 2 แล้วได้ค่าตรงกัน แสดงว่า ค่าที่ได้จากการทดลองนั้นถูกต้องตามทฤษฎีทางตรรกศาสตร์

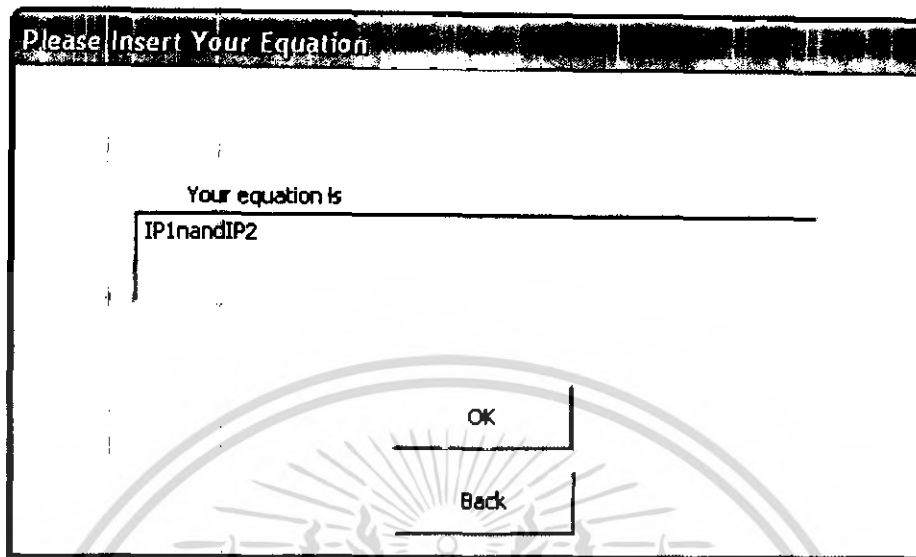
ตารางที่ 4.3 ตารางค่าความจริงของการนอต

IP3	$Y = \overline{IP3}$
0	1
1	0



ภาพที่ 4.21 สัญลักษณ์ของเกตนอตที่ใช้ในการวาดรูปวงจรร

#### 4.2.7.4 เมื่อมีการป้อนอินพุตที่มีเกตแนนด์

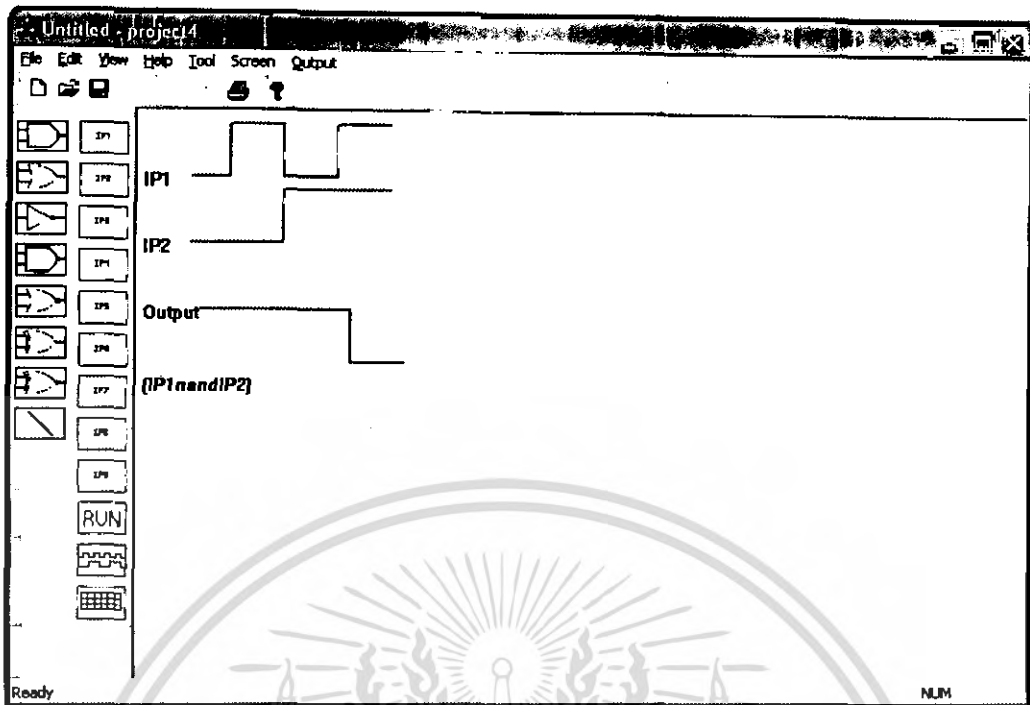


ภาพที่ 4.22 ตัวอย่างการจำลองการทำงานวงจรรกของการแนนด์โดยการป้อนสมการ



ภาพที่ 4.23 ตัวอย่างการจำลองการทำงานวงจรรกของการแนนด์โดยการป้อนรูปวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.24 เอาต์พุตของผลการทดลองในรูปแบบของแผนภูมิเวลา



ภาพที่ 4.25 เอาต์พุตของผลการทดลองในรูปแบบของตารางค่าความจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการทดลองพบว่าเมื่อมีการป้อนอินพุตในรูปแบบของสมการ หรือรูปวงจรรกของเกตแนนด์แล้ว ค่าเอาต์พุตที่ได้ทั้งตารางค่าความจริงและแผนภูมิเวลานั้นมีค่าตามตารางที่ 4.4 เมื่อเปรียบเทียบกับตารางค่าความจริงของเกตแนนด์ (ตารางที่ 2.5) ในบทที่ 2 แล้วได้ค่าตรงกัน แสดงว่าค่าที่ได้จากการทดลองนั้นถูกต้องตามทฤษฎีทางตรรกศาสตร์

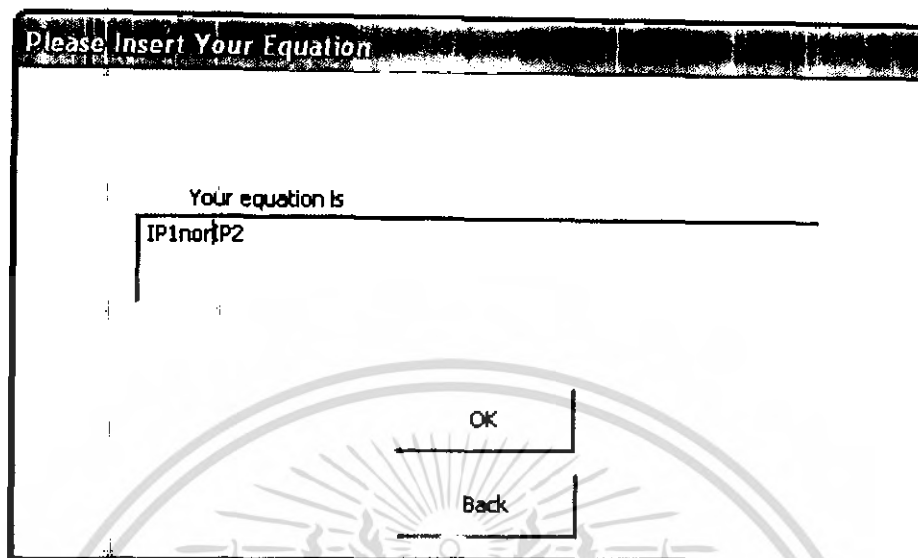
ตารางที่ 4.4 ตารางค่าความจริงของเกตแนนด์

IP1	IP2	$Y = \overline{IP1 \cdot IP2}$
0	0	1
0	1	1
1	0	1
1	1	0



ภาพที่ 4.26 สัญลักษณ์ของเกตแนนด์ที่ใช้ในการวาดรูปวงจร

#### 4.2.7.5 เมื่อมีการป้อนอินพุตที่มีเกตนอร์

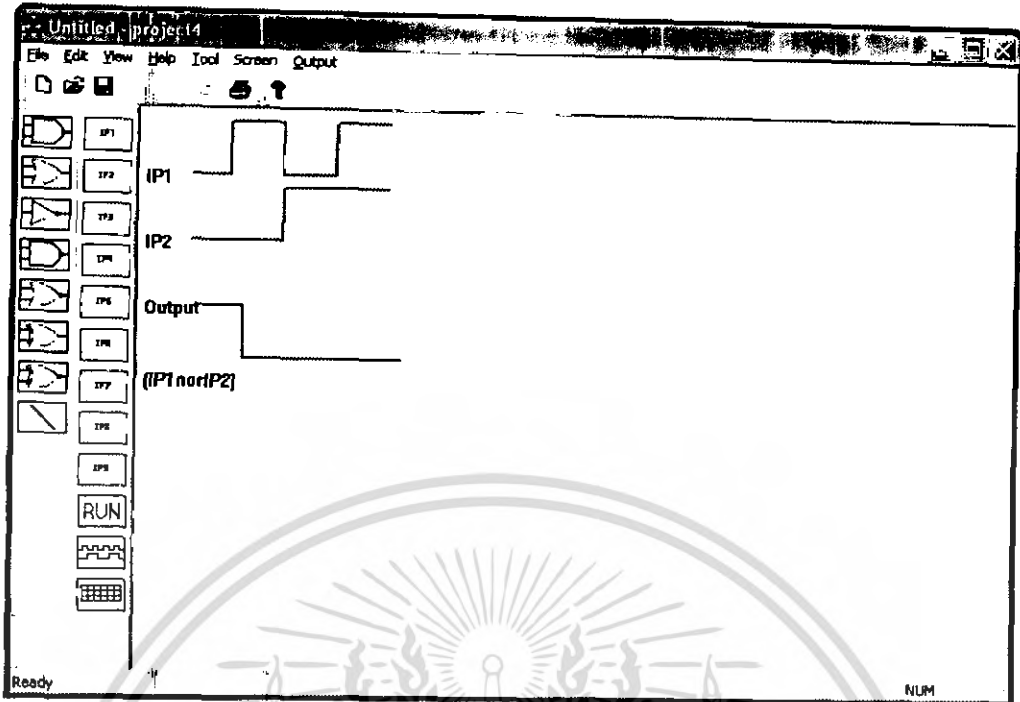


ภาพที่ 4.27 ตัวอย่างการจำลองการทำงานวงจรตรรกะของกรนอร์โดยการป้อนสมการ



ภาพที่ 4.28 ตัวอย่างการจำลองการทำงานวงจรตรรกะของเกตนอร์โดยการป้อนสมการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.29 เอาท์พุทของผลการทดลองในรูปแบบของแผนภูมิเวลา



ภาพที่ 4.30 เอาท์พุทของผลการทดลองในรูปแบบของตารางค่าความจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการทดลองพบว่าเมื่อมีการป้อนอินพุตในรูปแบบของสมการ หรือรูปวงจรรวมของ เกตเนอร์แล้ว ค่าเอาท์พุตที่ได้ทั้งตารางค่าความจริงและแผนภูมิเวลานั้นมีค่าตามตารางที่ 4.5 เมื่อ เปรียบเทียบกับตารางค่าความจริงของเกตเนอร์ (ตารางที่ 2.6) ในบทที่ 2 แล้วได้ค่าตรงกัน แสดงว่า ค่าที่ได้จากการทดลองนั้นถูกต้องตามทฤษฎีทางตรรกศาสตร์

ตารางที่ 4.5 ตารางค่าความจริงของการนอร์

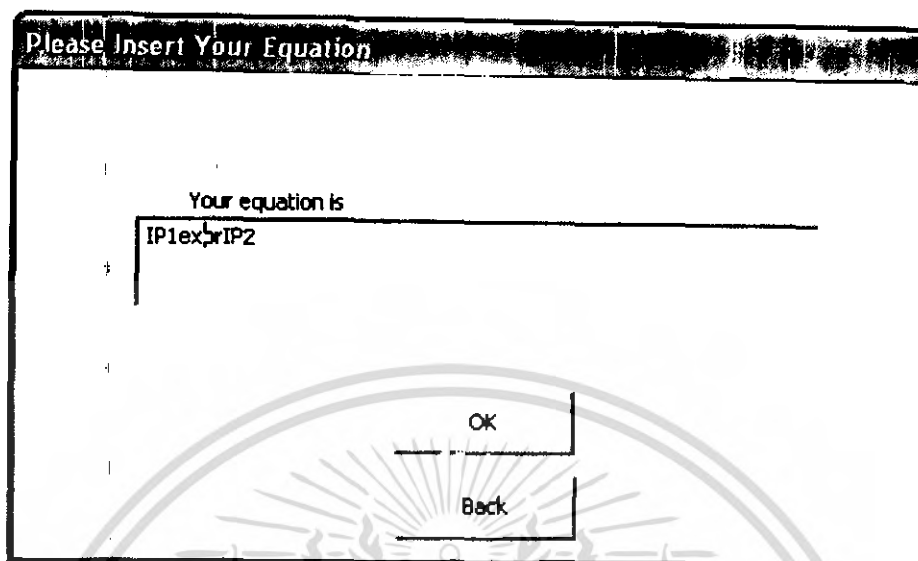
IP1	IP2	$Y = \overline{IP1 + IP2}$
0	0	1
0	1	0
1	0	0
1	1	0



ภาพที่ 4.29 สัญลักษณ์ของเกตเนอร์ที่ใช้ในการวาดรูปวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.7.6 เมื่อมีการป้อนอินพุตที่มีเกตเอกซ์คลูซีฟออร์

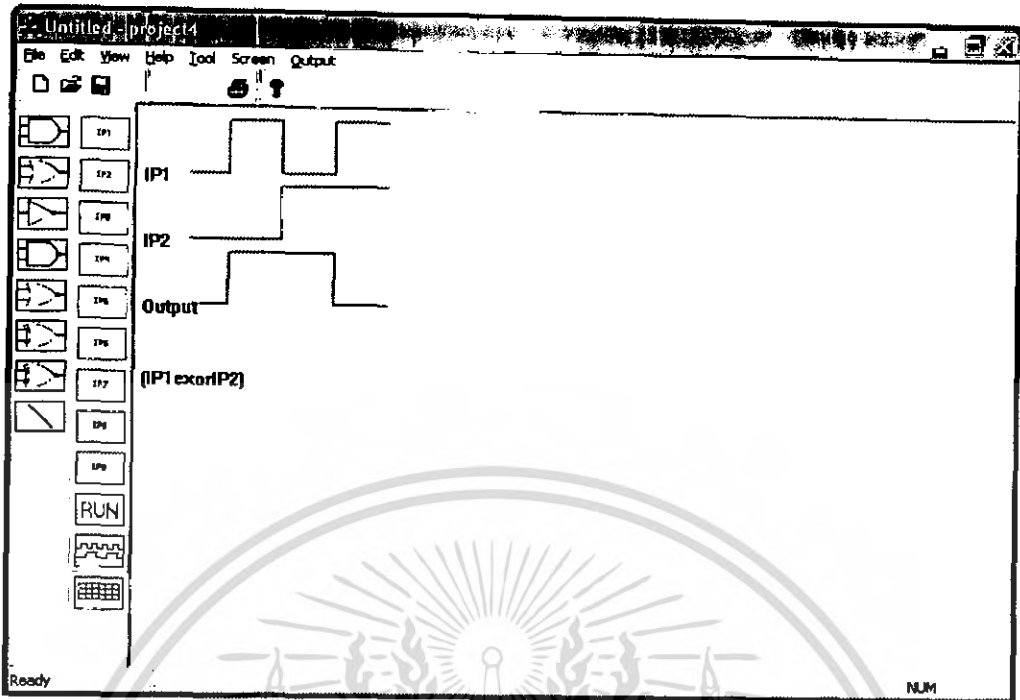


ภาพที่ 4.32 ตัวอย่างการจำลองการทำงานวงจรของการเอกซ์คลูซีฟออร์ โดยการป้อนสมการ



ภาพที่ 4.33 ตัวอย่างการจำลองการทำงานวงจรของการเอกซ์คลูซีฟออร์ โดยการป้อนสมการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.34 เอาท์พุทของผลการทดลองในรูปแบบของแผนภูมิเวลา



ภาพที่ 4.35 เอาท์พุทของผลการทดลองในรูปแบบของตารางค่าความจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการทดลองพบว่าเมื่อมีการป้อนอินพุตในรูปแบบของสมการ หรือรูปวงจรรรคของเกตเอกซ์คลูซีฟออร์แล้ว ค่าเอาท์พุตที่ได้ทั้งตารางค่าความจริงและแผนภูมิเวลานั้นมีค่าตามตารางที่ 4.6 เมื่อเปรียบเทียบกับตารางค่าความจริงของเกตเอกซ์คลูซีฟออร์ (ตารางที่ 2.7) ในบทที่ 2 แล้วได้ค่าตรงกัน แสดงว่าค่าที่ได้จากการทดลองนั้นถูกต้องตามทฤษฎีทางตรรกศาสตร์

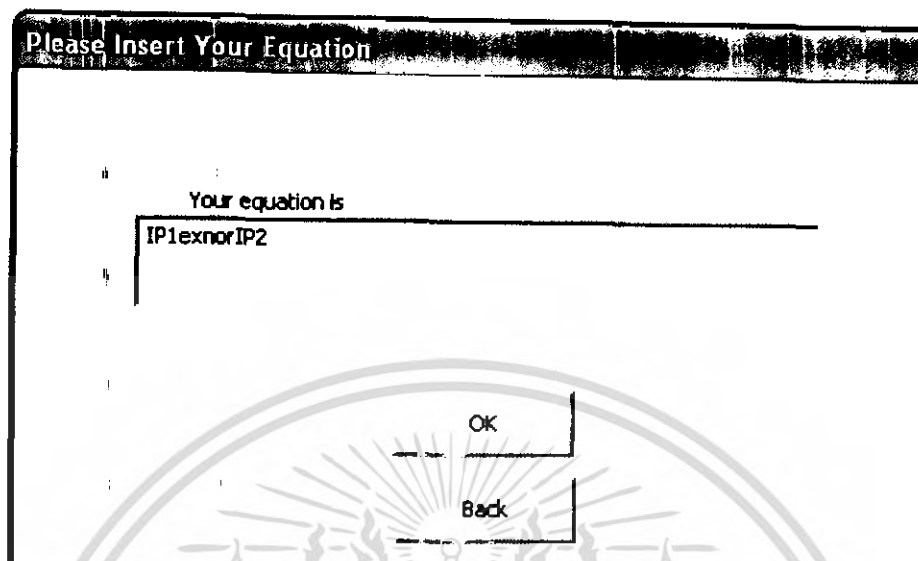
ตารางที่ 4.6 ตารางค่าความจริงของการเอกซ์คลูซีฟออร์

IP1	IP2	$Y = IP1 \oplus IP2$
0	0	0
0	1	1
1	0	1
1	1	0



ภาพที่ 4.36 สัญลักษณ์ของเกตเอกซ์คลูซีฟออร์ที่ใช้ในการวาดรูปวงจรร

#### 4.2.7.7 เมื่อมีการป้อนอินพุตที่มีเกตเอกซ์คลูซีฟเนอร์

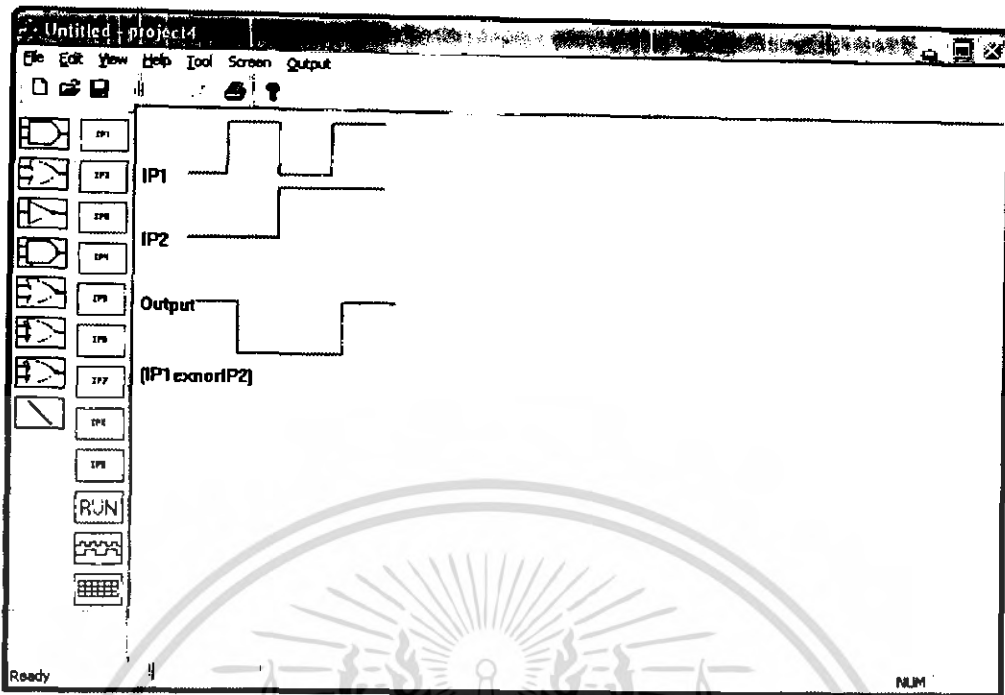


ภาพที่ 4.37 ตัวอย่างการจำลองการทำงานวงจรตรรกะของการเอกซ์คลูซีฟเนอร์โดยการป้อนสมการ



ภาพที่ 4.38 ตัวอย่างการจำลองการทำงานวงจรตรรกะของการเอกซ์คลูซีฟเนอร์โดยการป้อนสมการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.39 เอาท์พุทของผลการทดลองในรูปแบบของแผนภูมิเวลา



ภาพที่ 4.40 เอาท์พุทของผลการทดลองในรูปแบบของตารางค่าความจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการทดลองพบว่าเมื่อมีการป้อนอินพุตในรูปแบบของสมการ หรือรูปวงจรตรรกะของเกตเอกซ์คลูซีฟนอร์แล้ว ค่าเอาท์พุตที่ได้ทั้งตารางค่าความจริงและแผนภูมิเวลานั้นมีค่าตามตารางที่ 4.7 เมื่อเปรียบเทียบกับตารางค่าความจริงของเกตเอกซ์คลูซีฟนอร์ (ตารางที่ 2.8) ในบทที่ 2 แล้วได้ค่าตรงกัน แสดงว่าค่าที่ได้จากการทดลองนั้นถูกต้องตามทฤษฎีทางตรรกศาสตร์

ตารางที่ 4.7 ตารางค่าความจริงของการเอกซ์คลูซีฟนอร์

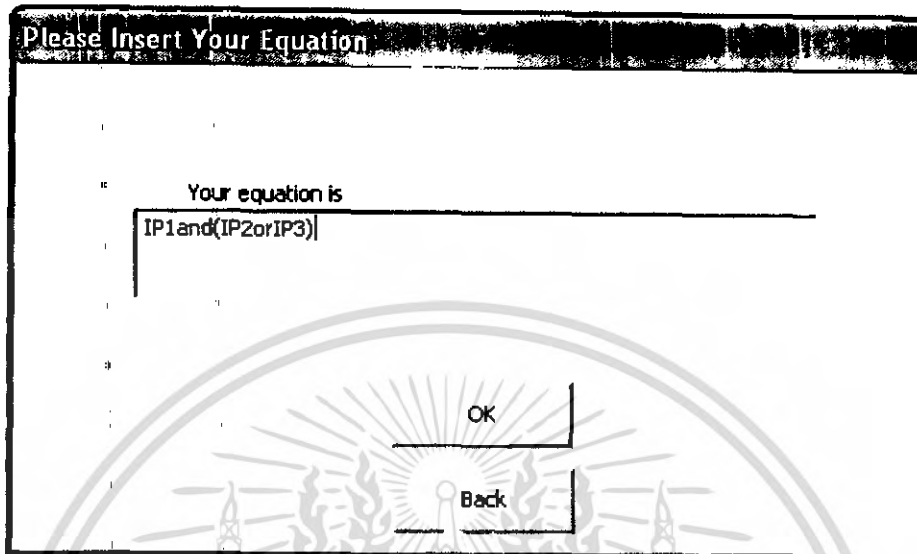
IP1	IP2	$Y = IP1 \oplus IP2$
0	0	1
0	1	0
1	0	0
1	1	1



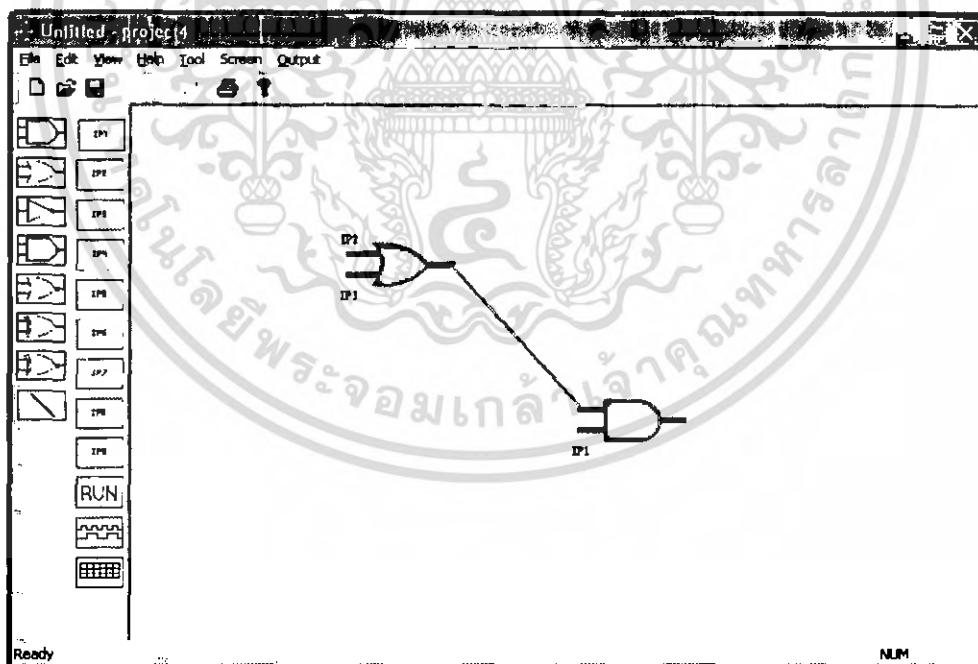
ภาพที่ 4.41 สัญลักษณ์ของเกตเอกซ์คลูซีฟนอร์ที่ใช้ในการวาดรูปวงจร

#### 4.2.7.8 เมื่อมีการนำเกตมาต่อเป็นวงจร

##### 4.2.5.7.1 ตัวอย่างที่ 1 IP1and(IP2orIP3)

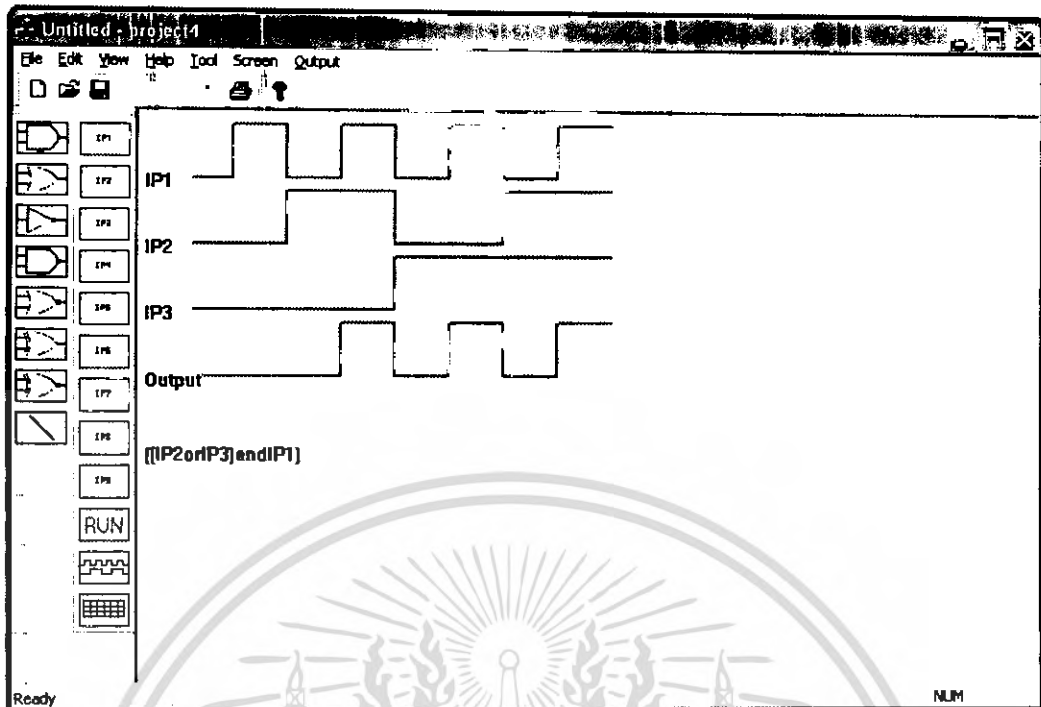


ภาพที่ 4.42 ตัวอย่างการจำลองการทำงานวงจรตรรกะโดยการป้อนสมการ

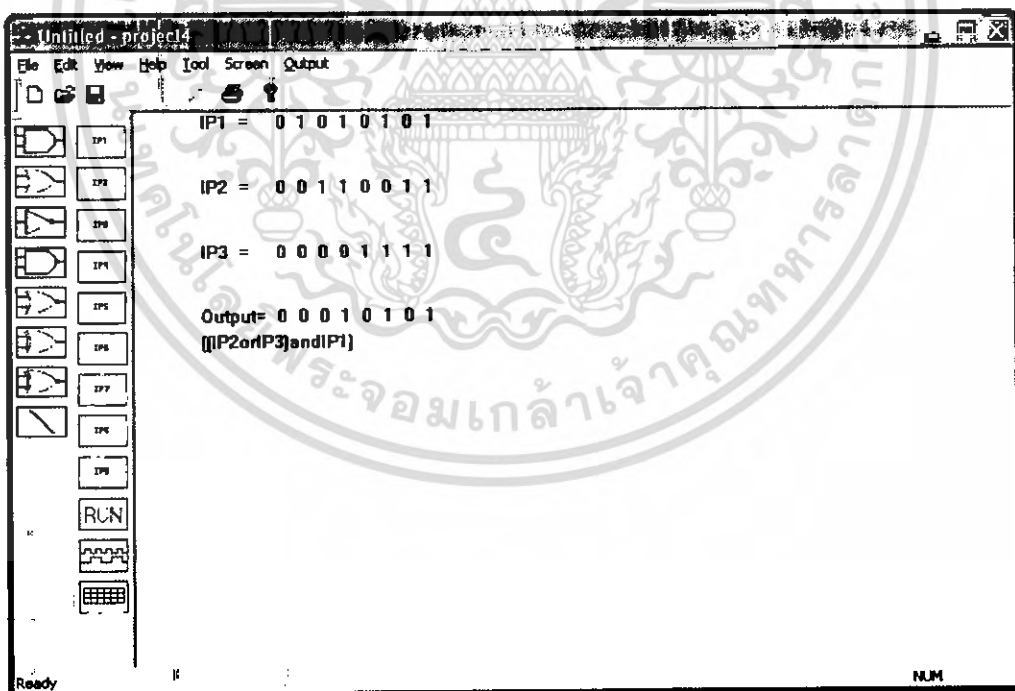


ภาพที่ 4.43 ตัวอย่างการจำลองการทำงานวงจรตรรกะโดยการป้อนรูปวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.44 เอาท์พุทของผลการทดลองในรูปแบบของแผนภูมิเวลา



ภาพที่ 4.45 เอาท์พุทของผลการทดลองในรูปแบบของตารางค่าความจริง

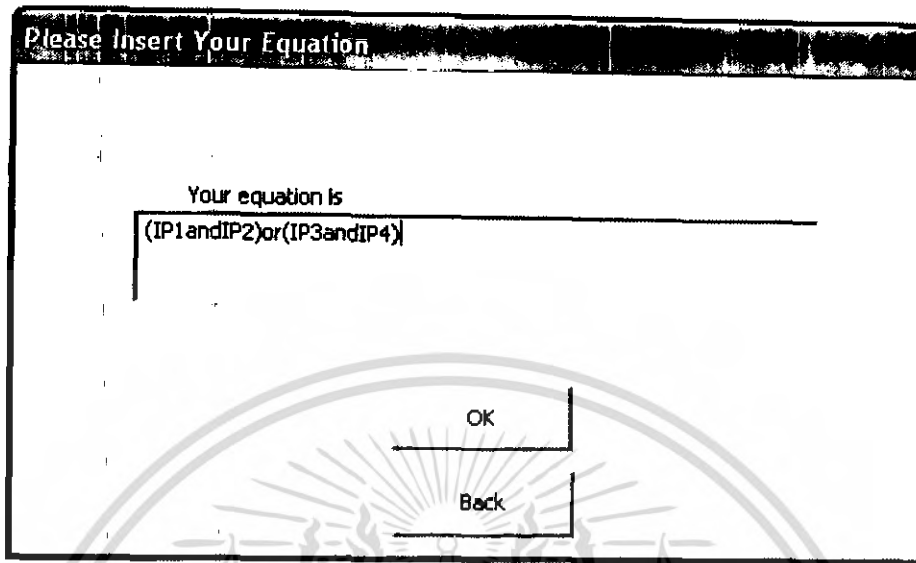
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการทดลองพบว่าเมื่อมีการป้อนอินพุตในรูปแบบของสมการ หรือรูปวงจรรรคของตัวอย่างที่ 1 แล้ว ค่าเอาต์พุตที่ได้ทั้งตารางค่าความจริงและแผนภูมิเวลานั้นมีค่าตามตารางที่ 4.8 เมื่อเปรียบเทียบกับตารางค่าความจริงของเกตแอนด์ (ตารางที่ 2.2) และตารางค่าความจริงของเกตออร์ (ตารางที่ 2.3) ในบทที่ 2 แล้วได้ค่าตรงกัน แสดงว่าค่าที่ได้จากการทดลองนั้นถูกต้องตามทฤษฎีทางตรรกศาสตร์

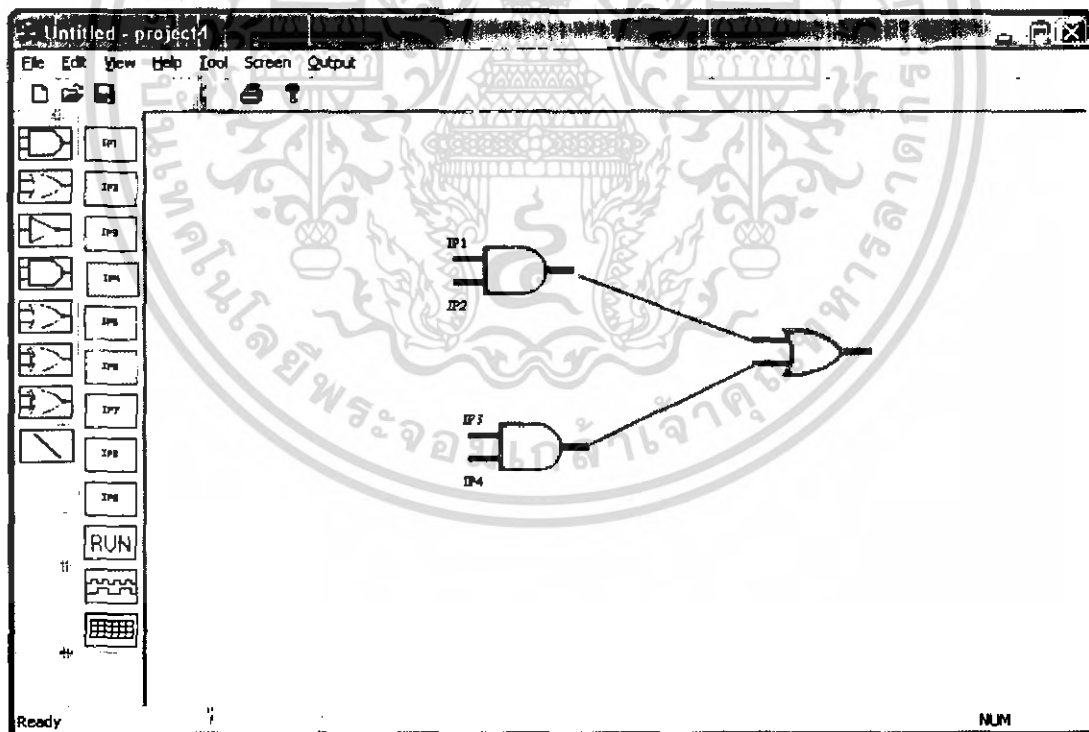
ตารางที่ 4.8 ตารางค่าความจริงของสมการ  $IP1 \text{ and } (IP2 \text{ or } IP3)$

IP1	IP2	IP3	IP2orIP3	IP1and(IP2orIP3)
0	0	0	0	0
1	0	0	0	0
0	1	0	1	0
1	1	0	1	1
0	0	1	1	0
1	0	1	1	1
0	1	1	1	0
1	1	1	1	1

#### 4.2.7.8.2 ตัวอย่างที่ 2 (IP1andIP2)or(IP3andIP4)

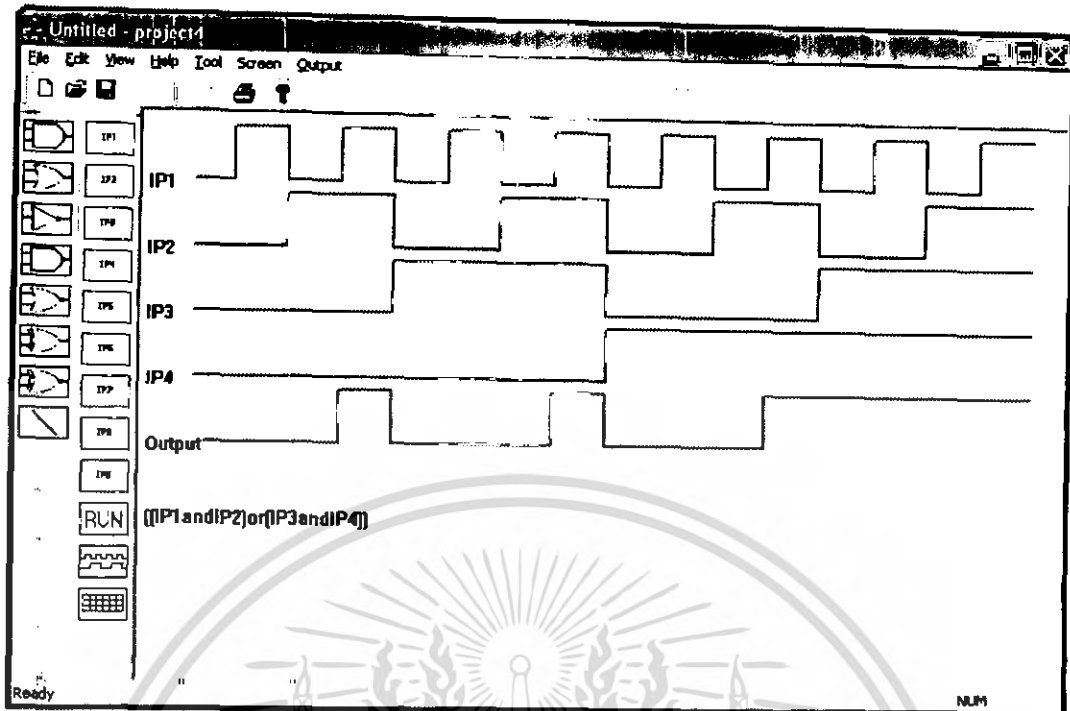


ภาพที่ 4.46 ตัวอย่างการจำลองการทำงานวงจรตรรกโดยการป้อนสมการ

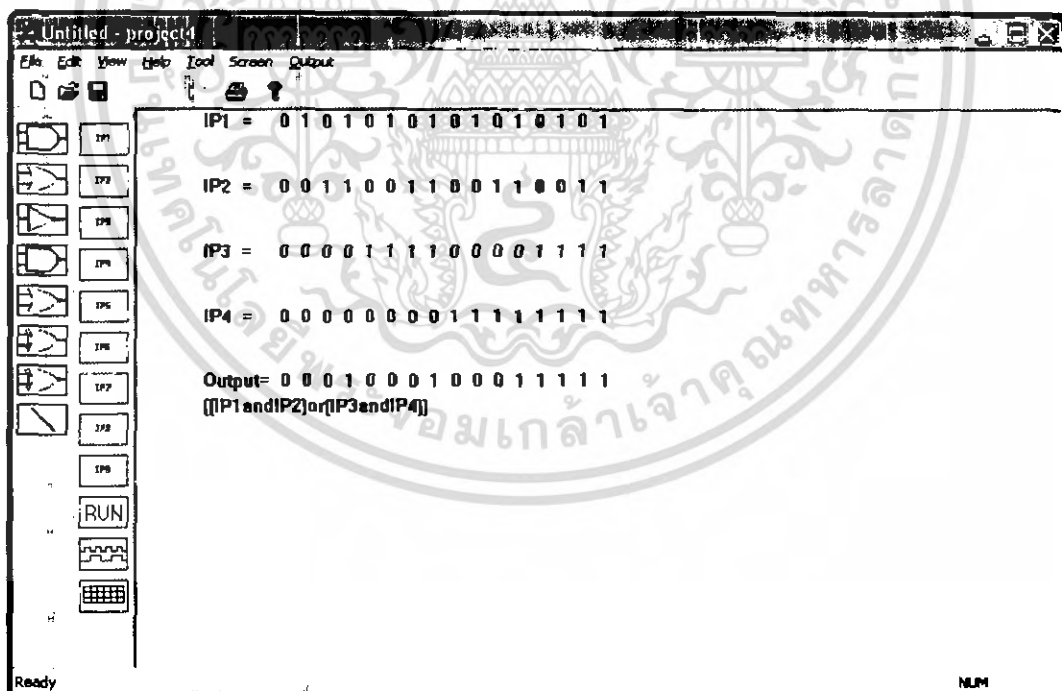


ภาพที่ 4.47 ตัวอย่างการจำลองการทำงานวงจรตรรกโดยการรูปวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.48 เอาต์พุตของผลการทดลองในรูปแบบของแผนภูมิเวลา



ภาพที่ 4.49 เอาต์พุตของผลการทดลองในรูปแบบของตารางค่าความจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

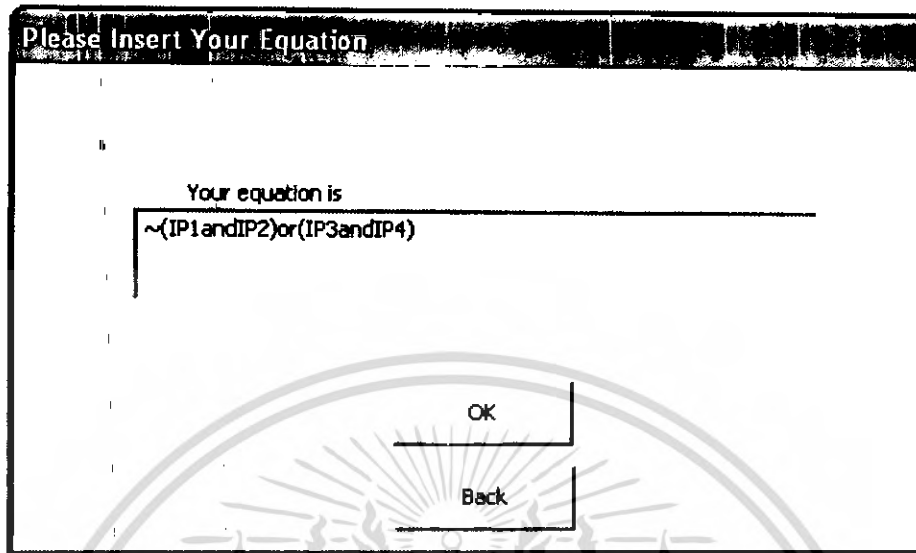
จากการทดลองพบว่าเมื่อมีการป้อนอินพุตในรูปแบบของสมการ หรือรูปวงจรรวมของตัวอย่างที่ 2 แล้ว ค่าเอาต์พุตที่ได้ทั้งตารางค่าความจริงและแผนภูมิเวกานั้นมีค่าตามตารางที่ 4.9 เมื่อเปรียบเทียบกับตารางค่าความจริงของเกตแอนด์ (ตารางที่ 2.2) และตารางค่าความจริงของเกตออร์ (ตารางที่ 2.3) ในบทที่ 2 แล้วได้ค่าตรงกัน แสดงว่าค่าที่ได้จากการทดลองนั้นถูกต้องตามทฤษฎีทางตรรกศาสตร์

ตารางที่ 4.9 ตารางค่าความจริงของสมการ(IP1andIP2)or(IP3andIP4)

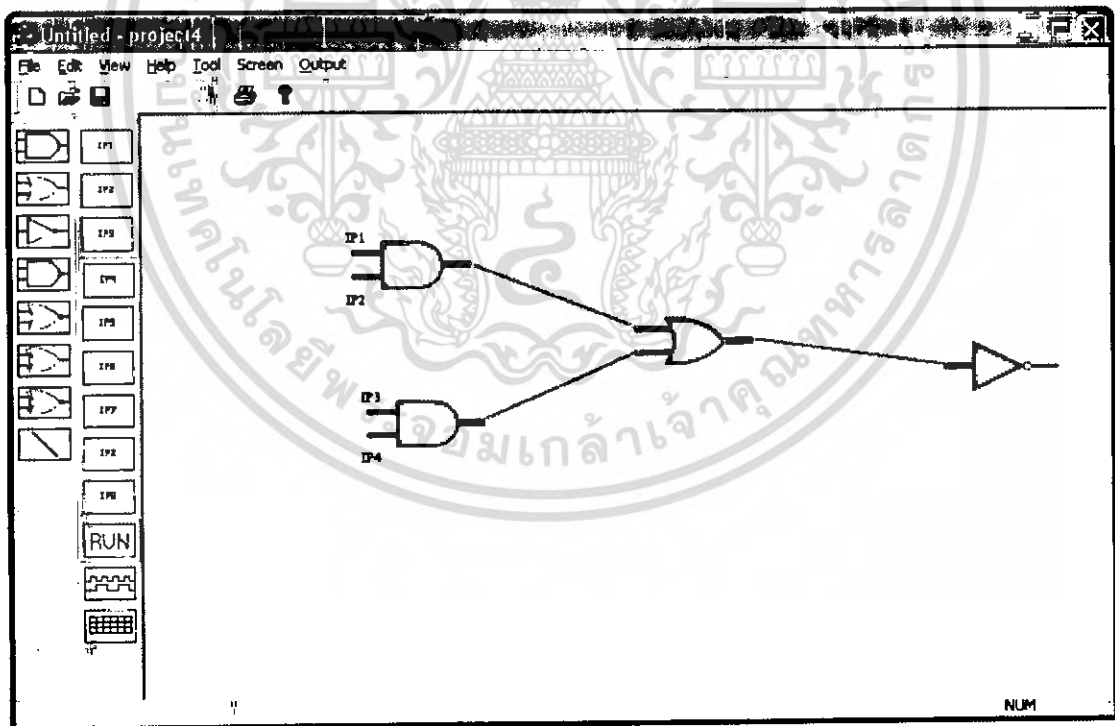
IP1	IP2	IP3	IP4	IP1andIP2	IP3andIP4	Output
0	0	0	0	0	0	0
1	0	0	0	0	0	0
0	1	0	0	0	0	0
1	1	0	0	1	0	1
0	0	1	0	0	0	0
1	0	1	0	0	0	0
0	1	1	0	0	0	0
1	1	1	0	1	0	1
0	0	0	1	0	0	0
1	0	0	1	0	0	0
0	1	0	1	0	0	0
1	1	0	1	1	0	1
0	0	1	1	0	1	1
1	0	1	1	0	1	1
0	1	1	1	0	1	1
1	1	1	1	1	1	1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.2.7.8.3 ตัวอย่างที่3 (IP1andIP2)or(IP3andIP4)

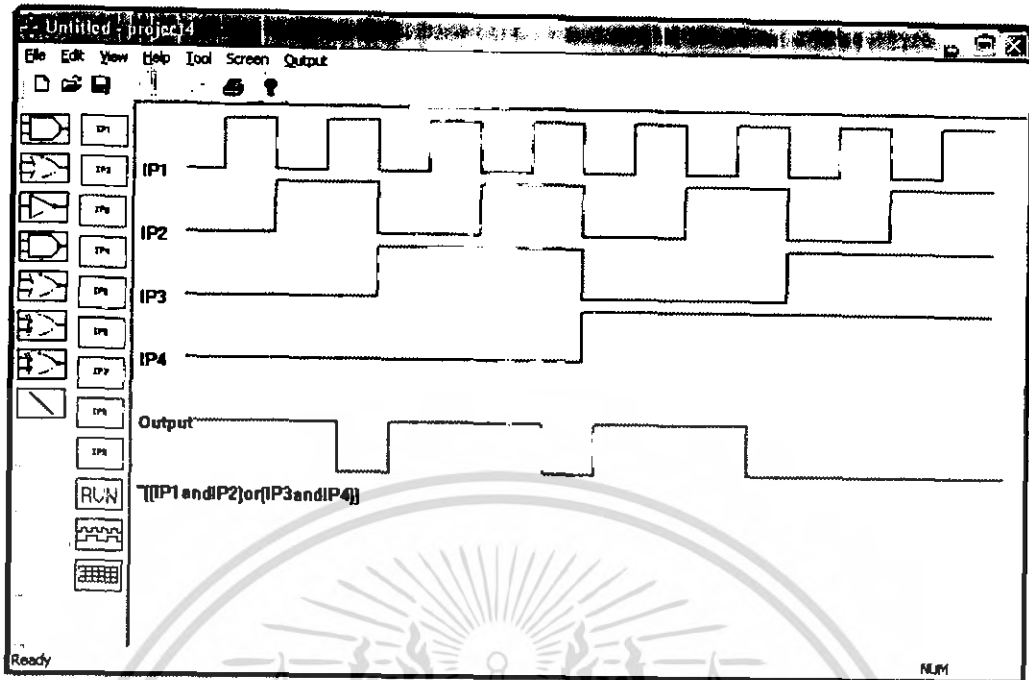


ภาพที่ 4.50 ตัวอย่างการจำลองการทำงานวงจรตรรกโดยการป้อนสมการ

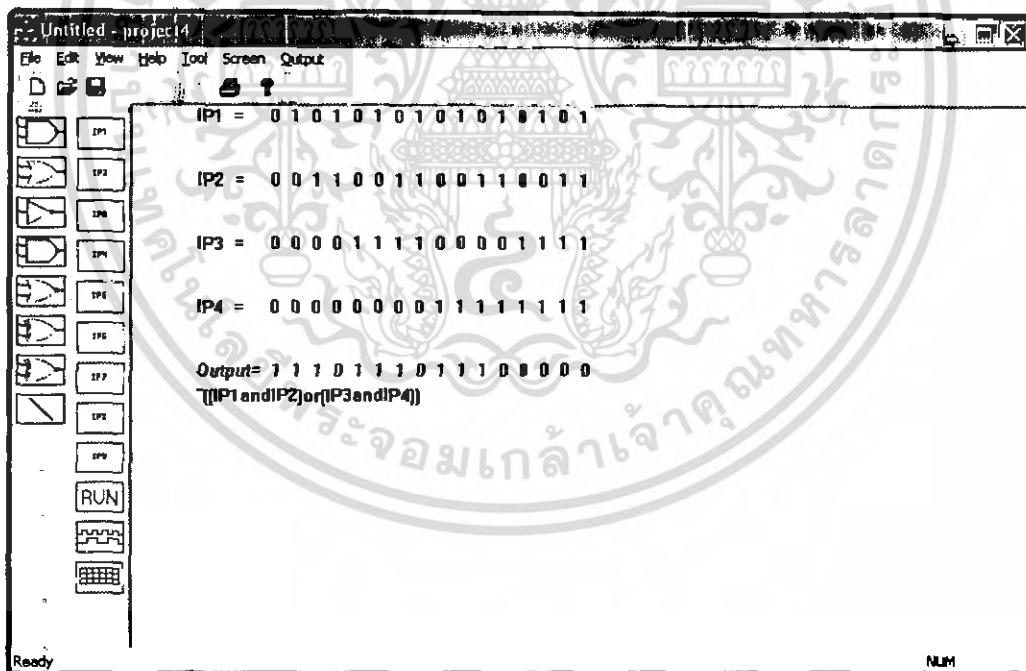


ภาพที่ 4.51 ตัวอย่างการจำลองการทำงานวงจรตรรกโดยการรูปร่างวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.52 เอาท์พุทของผลการทดลองในรูปแบบของแผนภูมิเวลา



จากการทดลองพบว่าเมื่อมีการป้อนอินพุตในรูปแบบของสมการ หรือรูปวงจรรวมของตัวอย่างที่ 3 แล้ว ค่าเอาต์พุตที่ได้ทั้งตารางค่าความจริงและแผนภูมิเวลานั้นมีค่าตามตารางที่ 4.10 เมื่อเปรียบเทียบกับตารางค่าความจริงของเกตแอนด์ (ตารางที่ 2.2) ตารางค่าความจริงของเกตออร์ (ตารางที่ 2.3) และตารางค่าความจริงของเกตนอท (ตารางที่ 2.4) ในบทที่ 2 แล้วได้ค่าตรงกัน แสดงว่าค่าที่ได้จากการทดลองนั้นถูกต้องตามทฤษฎีทางตรรกศาสตร์

ตารางที่ 4.10 ตารางค่าความจริงของสมการ  $\sim((IP1 \text{ and } IP2) \text{ or } (IP3 \text{ and } IP4))$

IP1	IP2	IP3	IP4	IP1andIP2	IP3andIP4	(IP1andIP2)or(IP3andIP4)	Output
0	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
0	1	0	0	0	0	0	1
1	1	0	0	1	0	1	0
0	0	1	0	0	0	0	1
1	0	1	0	0	0	0	1
0	1	1	0	0	0	0	1
1	1	1	0	1	0	1	0
0	0	0	1	0	0	0	1
1	0	0	1	0	0	0	1
0	1	0	1	0	0	0	1
1	1	0	1	1	0	1	0
0	0	1	1	0	1	1	0
1	0	1	1	0	1	1	0
0	1	1	1	0	1	1	0
1	1	1	1	1	1	1	0

### 4.3 สรุปผลการทดลอง

เมื่อทำการทดลองโปรแกรมจำลองการทำงานวงจรตรรกะที่เขียนบนโปรแกรม Microsoft Visual C++ เสร็จแล้วเราพบว่าค่าเอาต์พุตที่อยู่ในรูปตารางค่าความจริง และแผนภูมิเวลาที่ได้จากการป้อนสมการ หรือจากการป้อนรูปวงจรตรรกะนั้นถูกต้องตามทฤษฎีต่างๆที่ได้กล่าวมาแล้วในบทที่ 2 ซึ่งการทดลองนี้เป็นบางส่วนของฟังก์ชันการทำงานที่เป็นไปได้ และถ้าต้องการขยายขอบเขตการทำงานของฟังก์ชันให้มากกว่านี้ก็สามารถทำได้โดยการป้อนสมการหรือป้อนรูปวงจรตรรกะตามที่ใช้ต้องการ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

# บทสรุปและข้อเสนอแนะ

### 5.1 บทสรุป

ในการออกแบบวงจรตรรกแล้วจะทำการทดสอบการทำงานของวงจรที่ออกแบบ โดยใช้วิธีนำอุปกรณ์ต่างๆมาเชื่อมต่อกันนั้นเป็นวิธีที่ยุ่งยาก และสิ้นเปลืองค่าใช้จ่าย นอกจากนี้ยังทำให้เสียเวลา ดังนั้นการประยุกต์ใช้โปรแกรมคอมพิวเตอร์ในการจำลองการทำงานของวงจรตรรกนั้น จึงมีประโยชน์มาก เนื่องจากผู้ที่ออกแบบสามารถตรวจสอบการทำงานของวงจรได้สะดวก และไม่ต้องใช้อุปกรณ์หรือเครื่องมือวัดใดๆเลย และถ้าพบว่าวงจรทำงานไม่ถูกต้อง หรือต้องการเปลี่ยนแปลงแก้ไขการทำงานของวงจร ก็สามารถทำได้โดยทันที ซึ่งจะทำให้เราสามารถลดปัญหาดังข้างต้นได้ ในปัจจุบันโปรแกรมที่ใช้ในการจำลองการทำงานของวงจรตรรกนั้นมีอยู่มากมาย ซึ่งแต่ละโปรแกรมก็มีข้อดีและข้อเสียต่างกัน โดยส่วนใหญ่แล้วจะมีฟังก์ชันการทำงานของตัวอุปกรณ์จำกัด และไม่สามารถแก้ไขได้ แต่ในโปรแกรมนี้ผู้ใช้สามารถที่จะกำหนดฟังก์ชันการทำงานของตัวอุปกรณ์ต่างๆเองได้

ในปฏิญานี้ฉบับนี้ได้ออกแบบโปรแกรมดังกล่าวนี้โดยแบ่งออกเป็น 2 ส่วนใหญ่ๆ คือ โปรแกรมสำหรับการรับอินพุตซึ่งประกอบด้วยการป้อนสมการหรือการป้อนรูปร่างวงจรตรรก และโปรแกรมการจำลองการทำงานของวงจรตรรก โปรแกรมในส่วนแรกนั้นได้ออกแบบให้ผู้ใช้สามารถใช้โปรแกรมได้ง่ายและสะดวก โดยได้พัฒนาคำสั่งและฟังก์ชันต่าง ๆ ให้ใช้งานได้สะดวก ซึ่งผู้ใช้งานจะต้องป้อนสมการในรูปแบบของการพิมพ์สมการ หรือการวาดรูปร่างที่ต้องการเท่านั้น สำหรับโปรแกรมการจำลองการทำงานของวงจรตรรกนั้นจะเป็นส่วนของการประมวลผลแล้ว จะแสดงเอาท์พุตออกมาในรูปของตารางค่าความจริง (True Table) หรือในรูปของแผนภูมิเวลา (Timing Diagram) โดยมีการจัดให้แผนภูมิเวลาดังกล่าวอยู่ในแนวนอน เพื่อให้ผู้ใช้สะดวกในการอ่านแผนภูมิเวลา

### 5.2 ข้อเสนอแนะ

ในการออกแบบโปรแกรมจำลองการทำงานของวงจรตรรกนั้นได้ใช้โปรแกรม Microsoft Visual C++ 2005 ซึ่งมีความสะดวกในการติดต่อกับผู้ใช้ คือผู้ใช้สามารถวาดรูปร่างที่ต้องการได้ในโปรแกรมเลย ทั้งนี้โปรแกรมนี้ยังมีความสามารถในการเขียนโปรแกรมแบบโครงสร้าง และการเขียนโปรแกรมแบบเชิงวัตถุ (OOP : Object Oriented Programming) ซึ่งมีประโยชน์ต่อการเขียน

โปรแกรม คือ เราจะมองการทำงานต่างๆในโปรแกรมเป็นในรูปออบเจกต์ซึ่งเป็นการง่ายต่อการออกแบบสำหรับการใช้งาน

แต่อย่างไรก็ดีในบางส่วนของโปรแกรมจำลองการทำงานของวงจรตรรกะนั้น ยังมีข้อบกพร่องอยู่บ้าง เช่น

1. สามารถใส่อินพุตได้แค่ 8 อินพุตเท่านั้น
2. ควรมีการพัฒนาการวาดรูปวงจรตรรก์ให้มีความสวยงามกว่านี้ เช่น ควรมีการวาดเส้นของแต่ละเกตไม่ให้ทับกันโดยอัตโนมัติ

3. โปรแกรมสามารถที่จะประมวลผลได้ 8 อินพุต แต่การแสดงค่าอินพุตและเอาต์พุตไม่ว่าจะเป็นตารางค่าความจริง หรือแผนภูมิเวลานั้นสามารถแสดงค่าได้มากที่สุด 4 อินพุต เพราะว่าพื้นที่หน้าจอวินโดว์นั้นไม่เพียงพอ ดังนั้นควรจะมีการทำสกอลบาร์ (Scroll bar) เพื่อให้สามารถเลื่อนสกอลบาร์ไปดูค่าต่างๆที่ต้องการได้

4. ไม่สามารถจำลองการทำงานฟังก์ชันฟลิปฟล็อป (Flip-Flop), วงจรนับ (Counter), ชิพรีจิสเตอร์ (Shift Register) ทั้งนี้เพราะการสร้างฟังก์ชันของโปรแกรมจำลองการทำงานของวงจรตรรก์นั้นต้องอาศัยฟังก์ชันที่กำหนดจากโปรแกรม Microsoft Visual C++ เพียงอย่างเดียว ดังนั้นจะต้องมีการสร้างฟังก์ชันเพิ่มเติมขึ้นมาสำหรับฟังก์ชันประเภทนี้

ซึ่งสิ่งเหล่านี้จะเป็นแนวทางสำหรับผู้ที่จะทำการศึกษา หรือพัฒนาโปรแกรมให้มีประสิทธิภาพมากขึ้นกว่าเดิมต่อไป

## บรรณานุกรม

- [1] สมศักดิ์ มิตรถา.การออกแบบวงจรดิจิทัลและวงจรตรรก. กรุงเทพฯ : ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สจล., 2543
- [2] ประภาพร ช่างไม้.คู่มือการเขียนภาษา C ฉบับผู้เริ่มต้น. นนทบุรี : อินโฟเพรส , 2545
- [3] ยุทธนา กิตาพัฒน์กุล.เริ่มต้นการเขียนโปรแกรมภาษา C++. กรุงเทพฯ :ดวงกมลสมัย , 2547
- [4] นิรุช อำนวยศิลป์.การออกแบบชุดคำสั่งและการประยุกต์ C++. กรุงเทพฯ : Thaidev.Com, 2548
- [5] ยุทธนา กิตาพัฒน์กุล.คู่มือการเขียนโปรแกรมและใช้งาน Visual C++ 6.0 ฉบับโปรแกรมเมอร์. นนทบุรี : อินโฟเพรส, 2544
- [6] นิรุช อำนวยศิลป์.สร้างโปรแกรมบน Window ด้วย Microsoft Visual C++ Version 6.0. กรุงเทพฯ : ชัคเชสมีเดีย, 2544
- [7] ยุทธนา กิตาพัฒน์กุล.คู่มือการเขียนโปรแกรมและใช้งาน Visual C++.NET ฉบับสมบูรณ์. นนทบุรี : อินโฟเพรส, 2546

## ภาคผนวก



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## วิธีการใช้โปรแกรม

ผู้ใช้งานจะต้องเลือกรูปแบบการป้อนอินพุตก่อนว่าจะป้อนเป็นสมการ หรือจะป้อนเป็นรูปวงจรรรค

1. หากผู้ใช้งานต้องการป้อนอินพุตเป็นสมการ

1.1 ผู้ใช้งานจะต้องป้อนสมการที่ต้องการประมวลผลลงในเท็กซ์บ็อกซ์

1.2 ผู้ใช้ต้องกดปุ่ม OK

1.3 ผู้ใช้ต้องกดปุ่ม Run ในเมนู Output หรือในเมนูภาพเพื่อประมวลผล

1.4 ถ้าผู้ใช้งานต้องการเอาท์พุตเป็นตารางค่าความจริงผู้ใช้งานต้องกดปุ่มเมนู TrueTable ในเมนู Output หรือในเมนูภาพ

1.5 ถ้าผู้ใช้งานต้องการเอาท์พุตเป็นแผนภูมิเวลาผู้ใช้งานต้องกดปุ่มเมนู TimingDiagram ในเมนู Output หรือในเมนูภาพ

2. หากผู้ใช้งานต้องการป้อนอินพุตเป็นรูปวงจรรรค

2.1 ผู้ใช้ต้องกดปุ่มเมนู Tool หรือเมนูภาพเกิดเพื่อเลือกฟังก์ชันการทำงานของเกดที่ต้องการ

2.2 ผู้ใช้จะต้องลากเส้นจากเอาท์พุตของเกดตัวหนึ่ง ไปยังอินพุตของเกดอีกตัวหนึ่ง การลากเส้นนั้นจะต้องกดปุ่มเมนู Tool ก่อนแล้วเลือกฟังก์ชัน line หรือเมนูภาพเส้น จากนั้นจะลากเส้นได้โดยการคลิกเมาท์ 1 ครั้งที่จุดเริ่มต้น และดับเบิ้ลคลิกเมาท์ที่จุดสุดท้ายของเส้นที่ต้องการ

2.3 ผู้ใช้จะต้องป้อนอินพุต IP1-IP9 ใส่ที่ขาอินพุตของเกดที่ต้องการให้เป็นเกดเริ่มต้น

2.4 ผู้ใช้จะต้องตรวจสอบว่า

- แต่ละเกดที่มีการต่อกันนั้นมีเส้นมาต่อกันครบหรือเปล่า
- แต่ละขาของเกดที่เป็นเกดเริ่มต้นนั้นมีการใส่ค่า IP ครบหรือยัง
- เกดเอาท์พุตสุดท้ายจะต้องมีแค่ 1 ตัวเท่านั้น

2.5 เมื่อผู้ใช้งานตรวจสอบว่าการวาดวงจรรรคนั้นถูกต้องตามที่ผู้เขียนกำหนดไว้แล้ว ผู้ใช้ต้องกดปุ่ม Run ในเมนู Output หรือในเมนูภาพเพื่อประมวลผล

2.6 ถ้าผู้ใช้งานต้องการเอาท์พุตเป็นตารางค่าความจริงผู้ใช้งานต้องกดปุ่มเมนู TrueTable ในเมนู Output หรือในเมนูภาพ

2.7 ถ้าผู้ใช้งานต้องการเอาท์พุตเป็นแผนภูมิเวลาผู้ใช้งานต้องกดปุ่มเมนู TimingDiagram ในเมนู Output หรือในเมนูภาพ

3. เมื่อกดปุ่ม TrueTable หรือ TimingDiagram แล้วโปรแกรมจะแสดงเอาท์พุตสุดท้ายเป็นตารางค่าความจริงหรือแผนภูมิเวลาตามที่ผู้ใช้งานต้องการ

4. เมื่อผู้ใช้งานต้องการจะป้อนสมการตรรก หรือจะป้อนรูปวงจรรรคใหม่นั้น ผู้ใช้จะต้องกดปุ่ม Screen แล้วเลือกกดปุ่ม Clear เพื่อเคลียร์หน้าจอก่อนจากนั้นผู้ใช้ก็กดปุ่ม File แล้วเลือกกดปุ่ม New Program แล้วโปรแกรมก็จะเรียกไดอะล็อกการรับแบบการป้อนอินพุตขึ้นมาอีกครั้ง