

จุฬาลงกรณ์มหาวิทยาลัย

ชื่อ _____

คณะวิศวกรรมศาสตร์

เลขประจำตัว _____

ภาควิชาวิศวกรรมคอมพิวเตอร์

หมายเลขเครื่อง _____

2110-263 DIGITAL COMPUTER LOGIC LAB I

วันที่ _____

1. การใช้โปรแกรมจำลองการทำงานของวงจรตรรกะ (Simulator)

วัตถุประสงค์

1. เพื่อให้นิสิตสามารถใช้โปรแกรมเพื่อสร้างวงจรตรรกะ
2. เพื่อให้นิสิตสามารถใช้โปรแกรมเพื่อจำลองการทำงานของวงจรตรรกะ

บทนำ

การทดลองและศึกษาการออกแบบและการทำงานของวงจรตรรกะอาจทำได้โดยสร้างวงจรขึ้นมาโดยใช้อุปกรณ์ (IC) แล้วใช้เครื่องมือเช่น Oscilloscope หรือ logic probe วัดผล

อีกวิธีหนึ่งที่ได้คือใช้โปรแกรมจำลองการทำงาน หรือ Simulator ซึ่ง เป็น โปรแกรมที่สามารถแสดงผลการทำงานของวงจรจาก “รายละเอียดวงจร” (Design description) การกำหนดลักษณะของวงจรทำได้หลายแบบ เช่น ใช้การวาดสัญลักษณ์ของ logic gate (schematic capture) หรือ เขียนเป็น text ที่แสดงการเชื่อมต่อของแต่ละองค์ประกอบหรือ การทำงานของวงจร (hardware description languages) จากนั้น simulator จะแสดงผลที่ได้จากการทำงาน หรือ output ของวงจรที่เกิดจาก input ที่ผู้ใช้กำหนด

- Simulator มีอยู่หลายประเภทแต่ที่เราจะทดลองใช้ในวิชานี้จะมีสองรูปแบบคือ

Logic simulator จำลองการออกแบบเป็นการเชื่อมต่อของ logic gate ที่ให้ค่าออกมาเป็น 0 และ 1 (และค่าอื่นๆที่พบในตอนหลัง) นิสิตจะใช้โปรแกรมเพื่อตัดสินใจว่า ตารางความจริง (Truth table) ของวงจรที่สร้างขึ้นตรงกับ input/output ที่ต้องการหรือไม่ สำหรับวงจรง่ายๆ เราสามารถทดลองได้ทุกกรณีของ input แต่ถ้าเป็นวงจรที่ซับซ้อนเราอาจทดสอบได้เฉพาะบางกรณีเท่านั้น

- Timing simulator จะคล้ายกับ Logic simulator แต่จะจำลองการทำงานของ logic gate ให้เหมือนการทำงานจริงยิ่งขึ้น โดยจะยอมให้แต่ละ gate มี delay (ช่วงเวลาที่เกิดขึ้นระหว่างการเปลี่ยนแปลงของ output หลังจากที่มีการเปลี่ยนแปลงของ input) หรืออีกนัยหนึ่งโปรแกรมไม่เพียงแต่บอกว่า output จะเป็นอย่างไรจาก input แต่จะ

บอกได้ว่า output ที่ได้จะได้เมื่อไร โดยทั่วไปผลของ timing simulator จะเป็นในรูปของ waveform (แผนภูมิแสดงค่าของ logic 0 และ 1 เทียบกับเวลา)

การจำลองการทำงานนั้น ไม่สามารถจะทำงานได้ “เหมือนจริง” ในทุกๆ ด้าน ผู้ใช้จำเป็นต้องคำนึงถึงขีดจำกัดของโปรแกรมจำลองการทำงานคือ

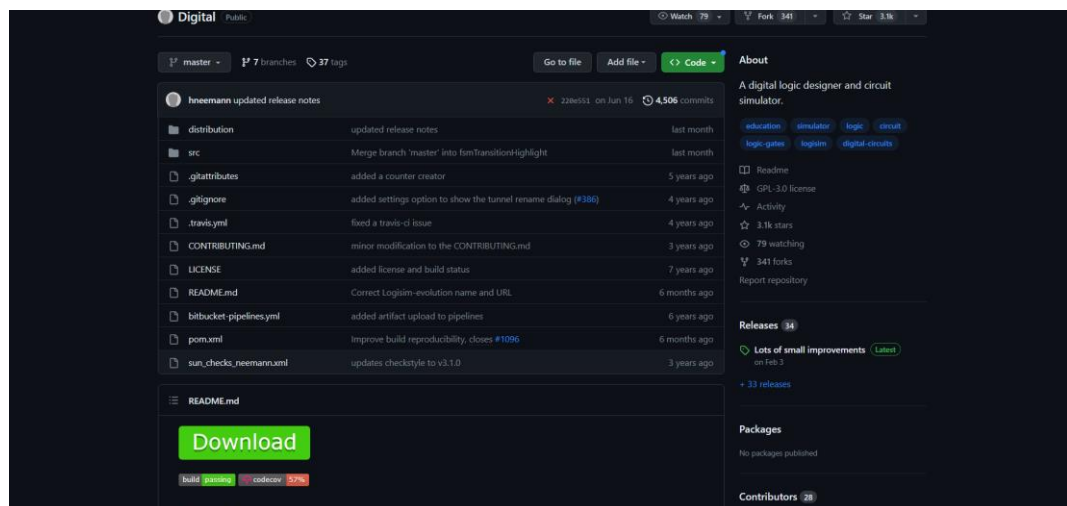
Logic simulator จะใช้ “แบบจำลอง” (Model) ของวงจร ดังนั้นถึงแม้ว่าวงจรจะทำงานได้อย่างถูกต้องใน simulator แต่เมื่อ “ต่อ” วงจรจริงอาจจะทำงานไม่ถูกต้องก็ได้ เช่น ในวงจรจริงอาจมีปัญหาเกี่ยวกับกระแสไฟฟ้าที่ใช้ไม่เพียงพอ หรือการรบกวนของสัญญาณเกิดขึ้น

ผลที่ได้จากการจำลองการทำงานขึ้นอยู่กับ input ที่ผู้ใช้ทดลองใช้ ในการทำงานจริงอาจจะมี input ที่อยู่นอกเหนือสิ่งที่ทดลองก็ได้ซึ่งวงจรอาจทำงานผิดพลาดสำหรับ input ที่ผู้ใช้ไม่ได้ทดสอบนี้

ทฤษฎี

โปรแกรมที่สามารถให้ผู้ใช้สร้างวงจรและจำลองการทำงานของวงจรตรรกะที่จะใช้ในวิชานี้ นั้นมีชื่อว่า Digital โดยเป็นโปรแกรมแบบ open source สามารถดาวโหลดได้จาก github :

<https://github.com/hneemann/Digital>

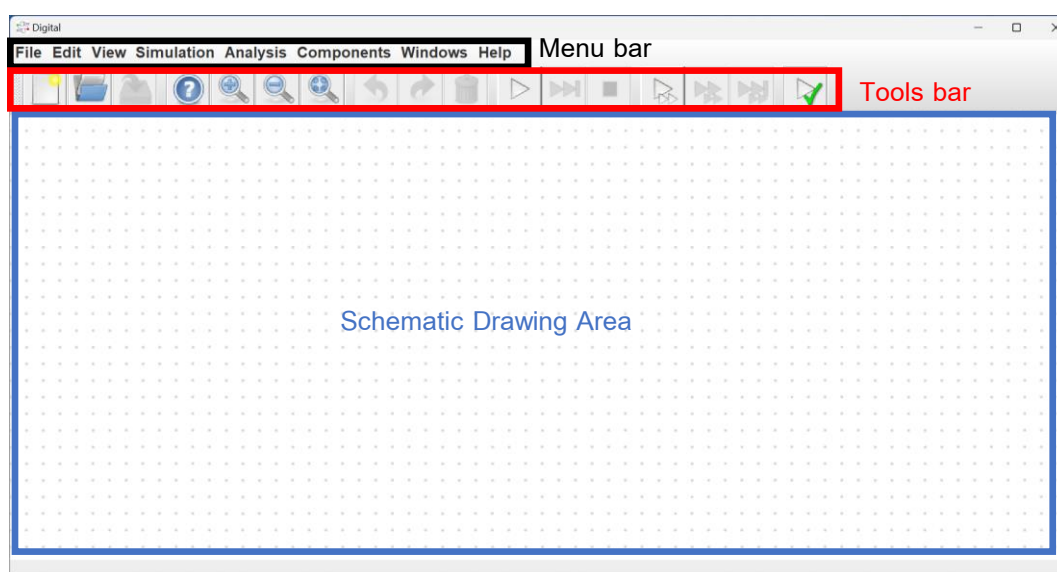


เมื่อดาวน์โหลดแล้วแตกไฟล์ จะสามารถเข้าโปรแกรมโดย double click ที่ icon Digital

docu	2023/06/30 23:15	File folder	
examples	2023/06/30 23:15	File folder	
lib	2023/06/30 23:15	File folder	
linux	2023/06/30 23:15	File folder	
Digital	2023/06/30 23:15	Application	46 KB
Digital	2023/06/30 23:15	Executable Jar File	3,715 KB
Digital	2023/06/30 23:15	SH Source File	1 KB
Digital_noD3D	2023/06/30 23:15	Application	47 KB
icon	2023/06/30 23:15	Microsoft Edge HTM...	14 KB
install	2023/06/30 23:15	SH Source File	2 KB
ReleaseNotes	2023/06/30 23:15	Text Document	22 KB
Version	2023/06/30 23:15	Text Document	1 KB

เมื่อเข้าสู่โปรแกรมโดยปกติจะเห็นหน้าจอประกอบด้วยส่วนต่างๆ ดังนี้

1. Schematic Drawing Area บริเวณสำหรับวาดวงจร
2. Tools Bar มีไว้สำหรับเลือกเครื่องมือต่างๆ
3. Menu bar แถบเมนูคำสั่งต่าง ๆ





Toolbar มีไว้สำหรับเลือกเครื่องมือต่างๆ โดย



ใช้สำหรับจัดการไฟล์



ใช้สำหรับการย่อ/ขยาย



ใช้สำหรับการย้อนกลับ ไปข้างหน้า และ ลบ Components



ใช้สำหรับการทำ Simulation



ใช้สำหรับทำ Single gate steps



ใช้สำหรับรัน Test Cases

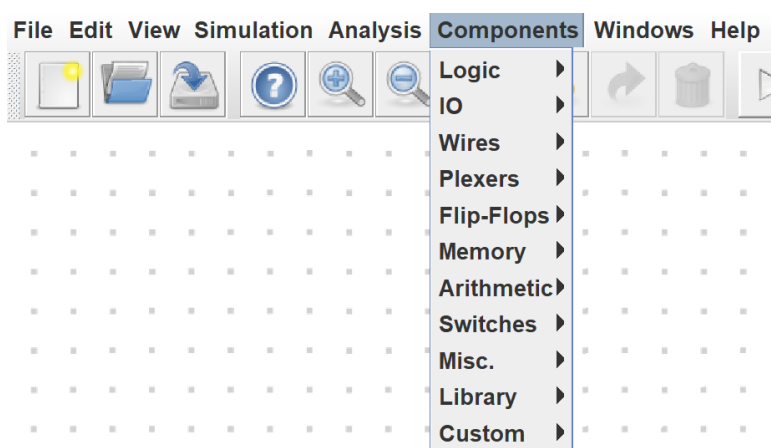
เริ่มต้นสร้างวงจร

ให้เลือกจากเมนู File → New หรือกดที่  จะปรากฏบริเวณวาดวงจรใหม่ขึ้น (ปกติเมื่อเปิดโปรแกรมครั้งแรก โปรแกรมจะแสดง New Design ให้เองอยู่แล้ว)

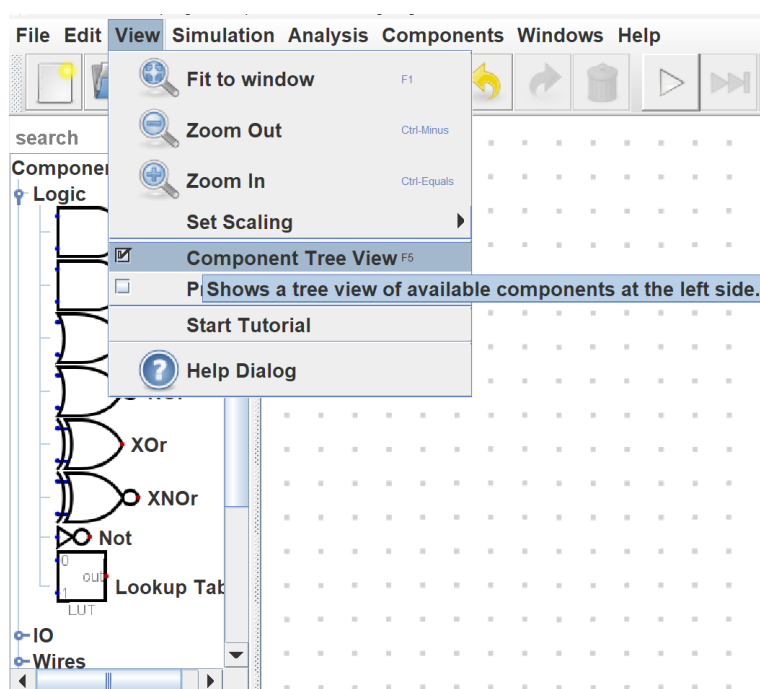
การเลือกและวางอุปกรณ์

การเลือกอุปกรณ์ สามารถทำได้สองวิธี

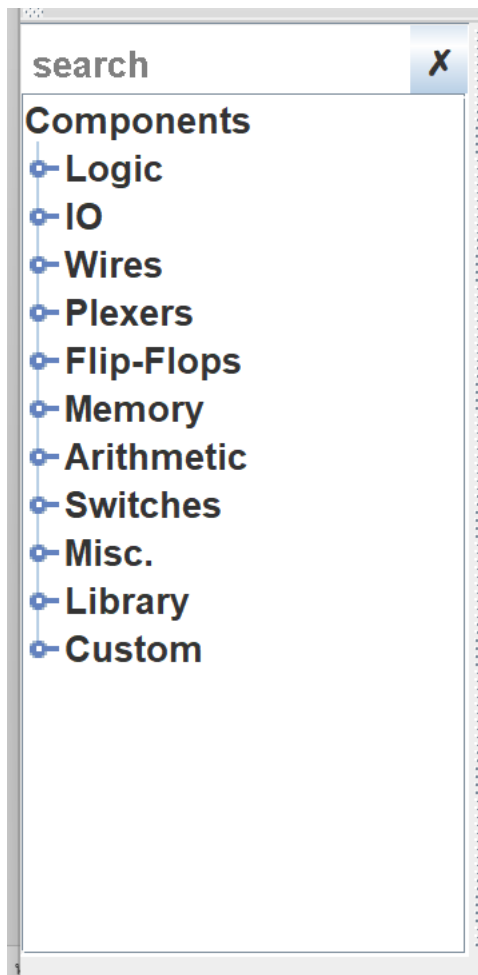
- 1) เลือกจากเมนู Components



- 2) เลือกจาก Component Tree View โดยเราสามารถเปิด Component Tree View โดยกดที่เมนู View → Component Tree View หรือกด F5 แล้วจะมี Component Tree View ปรากฏขึ้นที่ด้านซ้าย



ในหน้าต่าง Component Tree มีไว้สำหรับให้ผู้ใช้เลือกอุปกรณ์ต่างๆ ซึ่งจัดไว้เป็น Folder



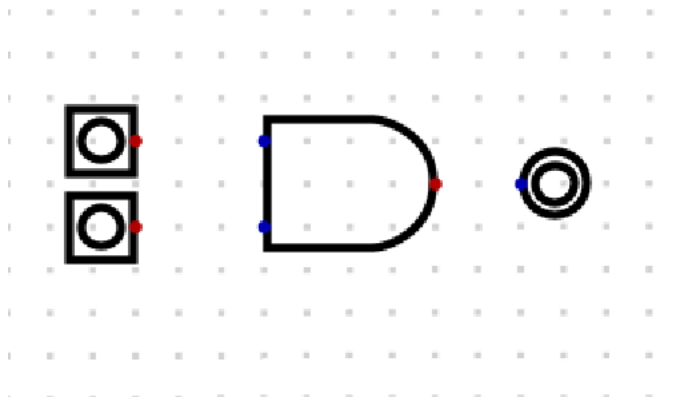
เราสามารถกดที่ชื่อ Folder เพื่อดูชื่อของอุปกรณ์ทั้งหมดที่อยู่ใน Folder นั้นๆ หากรู้ชื่อของอุปกรณ์ที่จะใช้แล้วก็สามารถพิมพ์ชื่อบางส่วนลงในช่อง Search ได้ เฉพาะอุปกรณ์ที่ชื่อมีส่วนเหมือนกับข้อความในช่อง Search จะปรากฏใน List วิธีนี้สามารถทำให้หาอุปกรณ์ที่ต้องการได้รวดเร็ว

สำหรับการทดลอง อุปกรณ์ที่ใช้ส่วนใหญ่จะอยู่ใน Folder ต่างๆ ดังนี้

- Logic เป็น Folder ของอุปกรณ์ logic เบื้องต้นเช่น AND , OR , NOT เป็นต้น
- IO เป็น Folder ของอุปกรณ์ Input/Output เบื้องต้น
- Wires เป็น Folder ของอุปกรณ์ที่เกี่ยวข้องกับสายสัญญาณ

การวางอุปกรณ์

เราสามารถวางอุปกรณ์ที่เราต้องการ โดยให้กดคลิกที่อุปกรณ์แล้วเลื่อน cursor ของ mouse ไปที่ Schematic Drawing Area แล้ว cursor จะลากรูปอุปกรณ์นั้น เมื่อต้องการวางอุปกรณ์นี้ไว้ที่ตำแหน่งใด ให้คลิกซ้ายที่บริเวณนั้น และก่อนวางเรายังสามารถหมุนอุปกรณ์ได้โดยการกด r

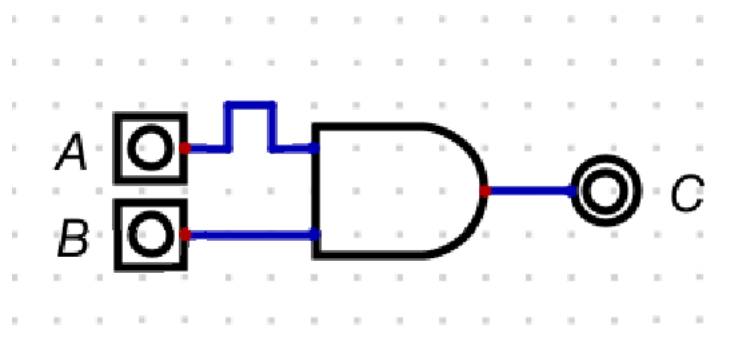


เมื่อเราวางอุปกรณ์เราจะสังเกตเห็นว่าตัวอุปกรณ์ต่างๆจะมีจุดสีแดงหรือสีน้ำเงิน

- จุดสีแดงแสดงถึง Output pin หรือขาเอาต์พุตของอุปกรณ์
- จุดสีน้ำเงินแสดงถึง Input pin หรือขาอินพุตของอุปกรณ์

การลากเส้นเชื่อมต่อระหว่างอุปกรณ์

การลากเส้น (คือ wire หรือ สายสัญญาณ) เส้นที่สร้างจะต้องเริ่มจาก “ขาของอุปกรณ์” (Pin) หรือ ต่อจากเส้นเดิมที่มีอยู่แล้ว ให้นำ cursor มาชี้ที่ขาอุปกรณ์ แล้วคลิกซ้ายแล้วเลื่อน mouse ไปตามทิศทางที่ต้องการจะมีเส้นเกิดขึ้น เมื่อได้เส้นที่ต่ออุปกรณ์ทั้งสองแล้วให้คลิกซ้ายอีกที การสร้างเส้นนั้นจะทำได้ในแนวนอนหรือแนวตั้ง และจะ “หัก” เส้น (เปลี่ยนทิศทางเป็นมุมฉาก) ได้หนึ่งครั้ง และถ้าต้องการหักเส้นหลายหนก็ต้องทำทีละตอนก่อนเชื่อมโยงกับขาอุปกรณ์

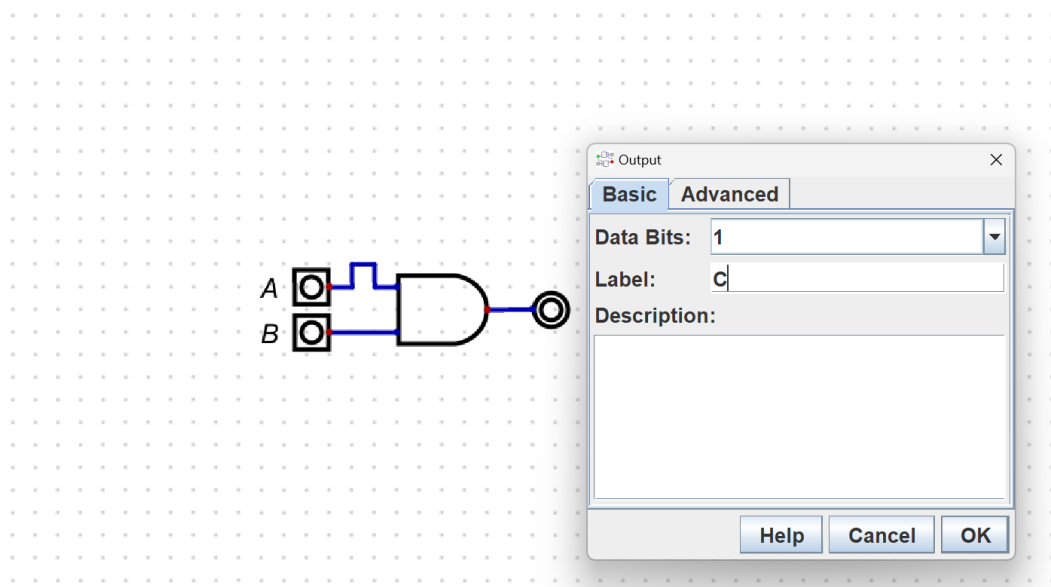


ข้อควรระวัง เราไม่ควรต่อขาอินพุตกับขาอินพุต หรือต่อขาเอาต์พุตกับขาเอาต์พุต โดยไม่ได้ต่อกับขาอินพุตเลย เพราะอาจทำให้เกิด Error ในตอนทำ Simulation ยกเว้นในบางอุปกรณ์ที่สามารถทำได้



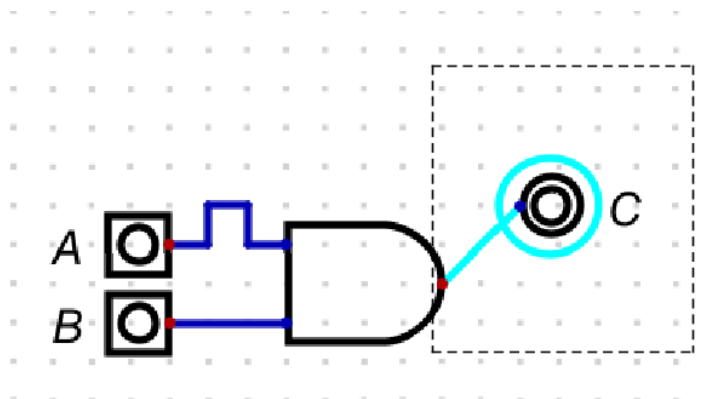
การตั้งชื่ออุปกรณ์

ทำได้โดยคลิกขวาที่อุปกรณ์ที่เราต้องการตั้งชื่อ แล้วใส่ชื่อในช่อง Label




การย้ายอุปกรณ์

ทำได้โดยคลิกซ้ายที่อุปกรณ์ที่เราต้องการย้ายแล้วลาก cursor ไปจุดที่เราต้องการวางอุปกรณ์ที่เราจะย้ายแล้วกดคลิกซ้ายอีกรอบ ในกรณีที่เรากำลังย้ายอุปกรณ์หลายตัว ให้ใช้ cursor ลากคลุมกลุ่มอุปกรณ์ที่เราต้องการ




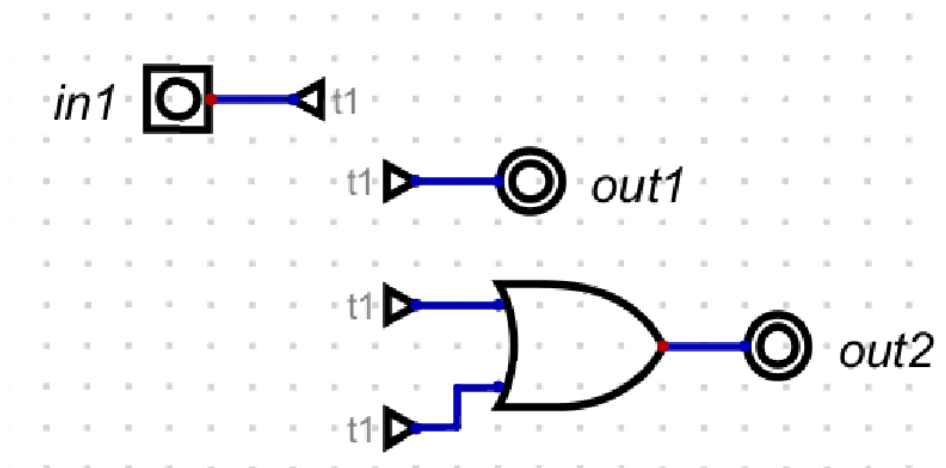
การลบอุปกรณ์หรือสายสัญญาณ

ทำได้โดยกดคลิกค้างแล้วลากคลุมอุปกรณ์หรือสายสัญญาณที่เราต้องการลบ แล้วกดปุ่ม delete หรือ Backspace ที่แป้นพิมพ์ หรือกด  ในหน้าต่าง Toolbar

การใช้งาน Tunnel เพื่อช่วยเชื่อมโยงสายสัญญาณ

ในการออกแบบวงจร เราอาจจะมีบางส่วนของวงจรที่มีสายสัญญาณเชื่อมต่อเข้ามาเป็นจำนวนมากหรือ อุปกรณ์ที่ต้องเชื่อมต่อกันนั้นอยู่ห่างจากกันมาก ทำให้จำนวนสายเชื่อมต่อมีความหนาแน่น, ไม่เป็นระเบียบ ซึ่งอาจส่งผลให้เกิดความยากลำบากในการแก้ไขหากวงจรเกิดข้อผิดพลาดขึ้น ฉะนั้นเราจึงควรจะใช้ Tunnel ในการช่วยออกแบบวงจร เพื่อช่วยให้วงจรของเราเป็นระเบียบมากยิ่งขึ้น

โดยเราจะสามารถหา Tunnel ได้จาก Folder Wires -  Tunnel เมื่อเราได้ Tunnel มา ให้เรานำ Tunnel มาแล้วเชื่อมต่อกับตัว Part ที่เราต้องการ โดย Tunnel ที่มีชื่อเดียวกันจะเชื่อมต่อกัน

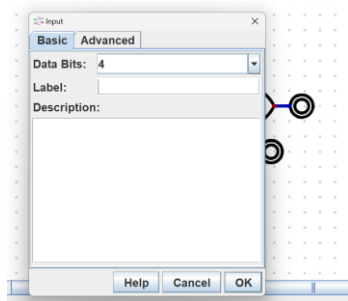


Data Bus

สายสัญญาณที่มีคุณสมบัติเหมือนกัน และมีจำนวนมากหรือเป็นกลุ่ม เราเรียกว่า บัส (Bus) เช่น บัสข้อมูล (Data Bus) บัสแอดเดรส (Address Bus) เป็นต้น การลากสายสัญญาณเหล่านั้นมักมีอุปสรรคไม่น้อย และเสียเวลามาก จึงมีการรวมสายเหล่านั้นเป็นสายสัญญาณบัสสายเดียวแทน เพื่อให้ง่ายต่อการลากสายสัญญาณ โดยเราสามารถรวมสายสัญญาณเข้าหากันได้โดยใช้ Splitter

Data Bits

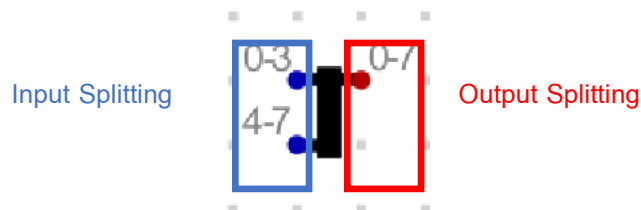
Data Bits คือจำนวนบิตของ Data ในสายสัญญาณนั้น โดยในโปรแกรม Digital เราสามารถกำหนดจำนวน Data Bits ของขาอุปกรณ์ต่างๆได้ ทำให้สายสัญญาณที่ต่อกับขานั้นมีค่า Data Bits เท่ากับขานั้น และ ขาของอุปกรณ์อื่นที่ต่อกับสายนั้นก็จะต้องมี Data Bits เท่ากันด้วยไม่เช่นนั้นจะเกิด Error ในตอนทำ Simulation โดยเราสามารถตั้งค่า Data bits ได้โดยการคลิกขวาที่อุปกรณ์



การใช้ Splitter/Merger

Splitter สามารถใช้ในการรวมสายสัญญาณ แยกสายสัญญาณ หรือทั้งรวมและแยกสายสัญญาณได้ เช่น เราสามารถสร้างการเชื่อมต่อแบบ 16-bit โดยไม่เดินต้องสายสัญญาณแบบ 1-bit จำนวน 16 เส้น แต่ใช้ Splitter รวมเป็นสายสัญญาณเส้นเดียวแบบ 16-bit แทน

โดย Splitter หาได้จาก Folder Wire โดยจะสังเกตว่า Splitter มีส่วนสีแดงและสีน้ำเงิน สีน้ำเงินคือส่วน Input Splitting และ สีแดงคือส่วน Output Splitting ตัวเลขที่กำกับอยู่ที่ขของ Splitter แสดงถึง index ที่ขนั้นกำกับอยู่ที่ขนั้น



เราสามารถกำหนดค่าของ Splitter และจำนวน Data Bus ของแต่ละขาได้ โดยคลิกขวาที่ Splitter แล้วใส่ตัวเลขตัวเดียวหรือตัวเลขสองตัวคั่นด้วย “-” แล้วคั่นด้วย “,” โดยตัวเลขตัวเดียวแสดงถึงจำนวน Data bits ของขานั้น และ ตัวเลขสองตัวคั่นด้วย “-” แสดงถึงช่วง index ของขานั้นๆ การคั่นด้วย “,” เป็นการแบ่งขาของ Splitter

การจำลองการทำงานของวงจรตรรกะ (Simulation)

สามารถทำ Logic Simulation ได้ในหน้าต่างที่วาดวงจรมันเลย โดยอุปกรณ์พื้นฐานที่ใช้ในการจำลองการทำงานของวงจรตรรกะได้แก่ Input, Output และ Clock Input ซึ่งอยู่ใน folder IO





Input ทำหน้าที่สร้างสัญญาณ input เป็นเลขฐานสองเช่น 0,1,10 เป็นต้น ถ้าเราต้องการให้อุปกรณ์สร้าง input ที่มีมากกว่า 1 บิต เราต้องแก้ไขค่า Data Bits ทำได้โดยใช้ cursor คลิกขวาที่อุปกรณ์จะเพื่อเปลี่ยนค่าได้



Output เป็นอุปกรณ์ที่แสดงค่าของ logic โดยจะแสดงค่าเป็นเลขฐานสองเช่น 0,1,10 เป็นต้น Data Bits ของ Output จะต้องเท่ากับ Data Bits ของอุปกรณ์ที่นำไปต่อ เราสามารถแก้ไข Data Bits ได้โดยใช้ cursor คลิกขวาที่อุปกรณ์

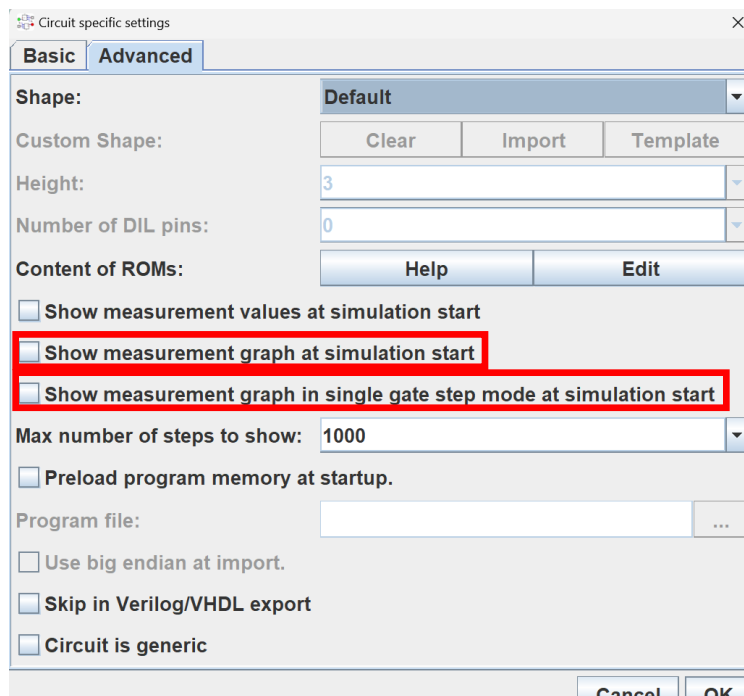


Clock Input เป็นอุปกรณ์ที่ทำหน้าที่เป็น input ที่ให้ค่า 0 และ 1 อย่างต่อเนื่อง ซึ่งสามารถกำหนดความถี่ได้โดยใช้ cursor ลากศรคลิกขวาที่ Clock อันที่ต้องการ ดึงถูกที่ Start real time clock แล้วใส่ค่าที่ต้องการในช่อง Frequency/Hz

เราสามารถเริ่มทำ Simulation ได้โดยกดที่  ถ้าวงจรที่เราสร้างไม่มี Error โปรแกรมจะเริ่มทำการ Simulation เมื่อเราต้องการหยุดการ Simulation ให้กดที่ 

Timing diagram เป็นการแสดงค่าของสัญญาณในวงจรเทียบกับเวลา โดยจะแสดงในหน้าต่างอีกหน้าต่างหนึ่ง สัญญาณที่จะแสดงได้นั้นจะมาจากอุปกรณ์จาก Folder IO ที่มีกำกับชื่อเท่านั้น

เราสามารถเปิด Timing diagram โดยคลิกที่เมนู Edit แล้วเลือก circuit specific settings แล้วกดที่แท็บ Advanced แล้วติ๊กที่ Show measurement graph at simulation start หรือ Show measurement graph in single gate step mode at simulation start



- Show measurement graph at simulation start
เมื่อ Simulation เริ่มจะได้ Timing diagram ที่ไม่มี gate delay
- Show measurement graph in single gate step mode at simulation start
เมื่อ Simulation เริ่มจะได้ Timing diagram ที่มี gate delay

ตัวอย่างการใช้ Timing diagram

