

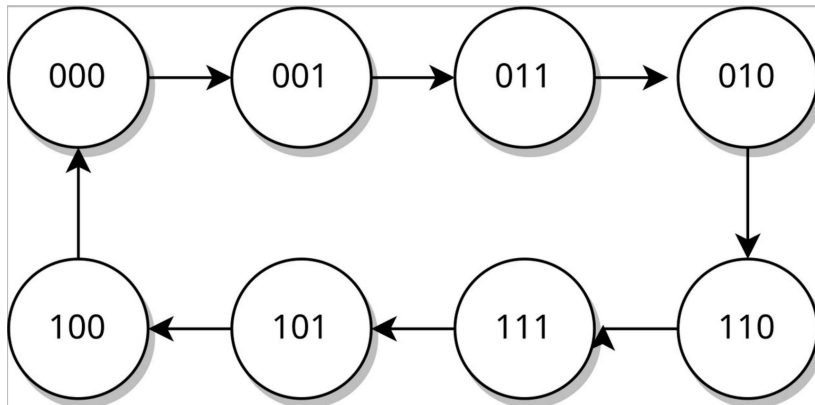
วิธีการดำเนินการสอบครั้งที่ 2 วิชา 2110252 Digital Computer Logic ปีการศึกษา 2566 เทอมต้น

วันจันทร์ที่ 2 ตุลาคม 2566 เวลาสอบ 09:00 - 11:30

- นิสิตสามารถนำกระดาษ A4 เข้าได้ 2 แผ่น เขียนหรือพิมพ์ก็ได้ เป็นเอกสารที่เตรียมด้วยตัวเองหรือสำเนาจากผู้อื่นก็ได้ หรือไม่เขียนอะไรเป็นกระดาษเปล่าๆก็ได้
- ไม่อนุญาตให้เปิด application ใดๆ นอกจาก digital, espresso, basic text editor, VS Code และ browser หนึ่งตัว
 - โดย browser จะมีได้เพียงแค่ tab ที่เป็นการใช้เกรดเดอร์ของวิชานี้และโจทย์ pdf ของวิชานี้เท่านั้น
 - Basic text editor เช่น notepad สำหรับการสร้างไฟล์ .pla ที่ใช้ใน espresso เท่านั้น
 - VS Code สำหรับสร้างไฟล์ .pla ที่ใช้ใน espresso เท่านั้น
- ไม่สามารถเปิดวงจรเก่าที่เคยเขียนไว้
- อนุญาตให้ใช้ espresso เพื่อช่วยลตรูปวงจร
- อนุญาตให้ใช้ gate หรือ IC หรือเทคนิคอื่นใดที่อยู่ในโปรแกรม digital ได้ (รวมทั้งการสร้างวงจรจาก Truth Table)
- ไม่อนุญาตให้ปรึกษาหรือดูผู้อื่น หรือค้นจากอินเทอร์เน็ต/ข้อมูลในเครื่อง เพิ่มเติม ให้ค้นหาได้เฉพาะจาก A4 ที่นำเข้ามาเท่านั้น
- บนโต๊ะที่ใช้ทำสอบจะมีได้แค่โน้ตบุ๊ก, กระดาษ A4 ที่นำเข้ามา, ระเบียบการสอบที่ผู้คุมสอบแจกให้ และอุปกรณ์เครื่องเขียนเท่านั้น ที่เหลือให้วางไว้โต๊ะ
- ต้องอัด video บันทึกหน้าจอตลอดการสอบ หากไม่มี video ส่งหลังการสอบ จะถือว่าได้คะแนนเป็น 0 **ทั้งนี้กรณีที่นิสิตไม่สามารถใช้โปรแกรมที่ทางวิชาแนะนำเพื่ออัดหน้าจอ นิสิตสามารถใช้โปรแกรมอื่นสำหรับอัดหน้าจอได้**
- หากพบว่าทุจริต ไม่ว่าจะเป็นผู้กระทำผิดหรือมีส่วนร่วมในการกระทำผิด นิสิตจะได้ U ทันที และอาจจะ
 - ให้พ้นสถานินิสิต
 - ให้ถอนรายวิชาอื่นทั้งหมดที่นิสิตผู้นั้นลงทะเบียนไว้ในภาคการศึกษานั้น
 - อ้างอิงจากหมวด 8 ข้อ 60 ของ [ข้อบังคับจุฬาลงกรณ์มหาวิทยาลัยว่าด้วยการศึกษาระดับปริญญาตรี](#)
- หลังสอบเสร็จจะมีลิงก์สำหรับส่งวิดีโอที่แถบ announcement ของ grader
- สอบวันที่ 2 ตุลาคม 2566 เวลา 09:00 - 11:30 โดยให้มาถึงห้องสอบเพื่อเตรียมพร้อมในเวลา 08:30 หากเข้าสายเกิน 08:45 จะไม่อนุญาตให้เข้าห้องสอบ โดยห้องสอบ เลขที่นั่งสอบและผังที่นั่งสอบ สามารถดูได้จากลิงก์นี้ [ทั้งนี้เมื่อสอบเสร็จและส่งไฟล์วิดีโอแล้ว ให้นิสิตปิดคอมพิวเตอร์ และรอในห้องสอบจนกว่าจะหมดเวลาสอบ](#)
- ในกรณีที่นิสิตคิดว่าทำวงจรได้ถูกต้องแล้ว แต่เกรดเดอร์ตรวจให้ผิดทุกข้อ ขอให้นิสิตแจ้งมาตอนส่งลิงค์วิดีโอว่า มีความประสงค์ให้อาจารย์ตรวจพิจารณาเพิ่มเติม โดยอาจารย์จะพิจารณาตรวจให้เฉพาะกรณีที่เกรดเดอร์ให้ผิดทุกข้อเท่านั้น ถ้าถูกต้องบาง Test Case จะไม่ตรวจพิจารณาเพิ่มเติมให้

Problem 1: Counter

จงสร้าง Counter ที่มีสัญญาณเข้าเป็น clock (**clk**) 1 ตัว, และ สัญญาณ input **reset** และมี output เป็น A, B และ C ซึ่งมี output ดังต่อไปนี้



โดยกำหนดให้ เมื่อ $\text{reset} = 1$ ค่าจะเริ่มต้นที่ $\text{CBA} = 000$, (reset เป็น synchronous) เมื่อนำค่า $\text{reset} = 0$, CBA จะเปลี่ยนเป็น 001, 011, 010, 110, 111, 101, 100 แล้ววนมาที่ 000 ตามลำดับ

ตัวอย่าง Testcase อยู่ใน template_01.dig

ข้อมูลนำเข้า

- Input: reset
- Clock: clk

ข้อมูลส่งออก

- Output: A, B, และ C

ชุดข้อมูลทดสอบ

- ตัวอย่างชุดข้อมูลทดสอบมีอยู่ใน template_01.dig

คะแนน

คะแนนเต็ม 100 คะแนน โดยมีจาก Grader 90 คะแนน และ ถ้าถูกต้องทุก Case ภายใน 2 ชม. จะได้อีก 10 คะแนน

Problem 2: Square Root

สร้างวงจรหารากที่สองของ input โดยให้ input เป็นเลข **A** 16 บิต, **start** 1 บิต และมี output **sqrt** 8 บิตเป็นค่ารากที่สองของ **A** ที่เป็นจำนวนเต็มมากที่สุดที่น้อยกว่าค่ารากที่สองของ **A** (ตัวอย่างเช่น **A** = 10, **sqrt**= 3)

วงจรจะเริ่มทำงานเมื่อมีการเปลี่ยน **start** จาก 0 เป็น 1 ระหว่างการทำงานให้ **busy** เป็น 1 และเมื่อคำนวณเสร็จสิ้นแล้วให้แสดงค่ารากที่สองใน **sqrt** และให้ค่า **busy** กลับเป็น 0

หมายเหตุ 1 : ช่วงระหว่างคำนวณอยู่ไม่ต้องสนใจสัญญาณ **start** และให้แสดงค่ารากที่สองของ **A** ค้างไว้จนกว่าจะมีการเปลี่ยน **start** จาก 0 เป็น 1 ใหม่จึงสามารถเปลี่ยนค่าได้

หมายเหตุ 2 : input **A** มีค่าตั้งแต่ 0 เป็นต้นไป และจะไม่เปลี่ยนแปลงในช่วงที่ **busy** เป็น 1

หมายเหตุ 3: ตัวตรวจจะรอสัญญาณไม่เกิน 500 cycles (ดังตัวอย่างใน **template_02.dig**)

Hint: สามารถใช้ อุปกรณ์ **Multiply** หรือ **Division** ซึ่งอยู่ใน Arithmetic ได้

ตัวอย่าง Testcase อยู่ใน **template_02.dig**

ข้อมูลนำเข้า

- Input: **A** ขนาด 16 Bit
- Input: **start** ขนาด 1 Bit
- Clock: **clk**

ข้อมูลส่งออก

- Output: **sqrt** ขนาด 8 Bit
- Output: **busy** ขนาด 1 Bit

ชุดข้อมูลทดสอบ

ตัวอย่างชุดข้อมูลทดสอบมีอยู่ใน **template_02.dig**

คะแนน

คะแนนเต็ม 100 คะแนน โดยมีจาก Grader 90 คะแนน และ ถ้าถูกต้องทุก Case ภายใน 2 ชม. จะได้ไปอีก 10 คะแนน

Problem 3: Mean

จงสร้างวงจรหาค่าเฉลี่ยของตัวเลข โดยสมมุติสามารถรับค่าได้ผ่านทาง input เป็นเลขฐานสอง **A** 16 บิต, **add** 1 บิตและ **reset** 1 บิต และมี output เป็นเลขฐานสอง **mean** 16 บิตและ **busy** 1 บิต

เมื่อ **add** มีค่าเปลี่ยนจาก 0 เป็น 1 ให้เก็บค่าจาก **A** และแสดงค่าเฉลี่ยของตัวเลขใน **mean** (ปัดเศษลง) หากวงจรใช้เวลาในการหาค่าเฉลี่ยให้มีค่า **busy** เป็น 1 จนกระทั่งทำงานเสร็จ ถ้า **reset** เป็น 1 จะทำการเริ่มต้นวงจรใหม่และวงจรจะทำงานก็ต่อเมื่อ **reset** เป็น 0

หลังจาก **reset** ในแต่ละครั้งจะมีการ **add** ไม่เกิน 15 ครั้ง

หมายเหตุ 1 : input **A** จะไม่เปลี่ยนแปลงในช่วงที่ **busy** เป็น 1

หมายเหตุ 2: จะรับประกันว่า **reset** จะเป็น 1 อย่างน้อย 1 cycle เมื่อเริ่มโปรแกรม

หมายเหตุ 3: ตัวตรวจจะรอสัญญาณไม่เกิน 1000 cycles ต่อการรับค่าเพิ่มหนึ่งค่า (ดังตัวอย่างใน **template_03.dig**)

Hint: สามารถใช้ อุปกรณ์ **Multiply** หรือ **Division** ซึ่งอยู่ใน Arithmetic ได้

ตัวอย่าง Testcase อยู่ใน **template_03.dig**

ข้อมูลนำเข้า

- A ขนาด 16 Bit
- input ขนาด 1 Bit
- add ขนาด 1 Bit

ข้อมูลส่งออก

- mean ขนาด 16 Bit
- busy ขนาด 1 Bit

ชุดข้อมูลทดสอบ

ตัวอย่างชุดข้อมูลทดสอบมีอยู่ใน **template_03.dig**

คะแนน

คะแนนเต็ม 100 คะแนน โดยมีจาก Grader 90 คะแนน และ ถ้าถูกต้องทุก Case ภายใน 2 ชม. จะได้อีก 10 คะแนน