

Problem 2: Any Root

สร้างวงจรหารากที่ R ของ input โดยให้ input เป็นเลข A 16 บิต, R 4 บิต, input 1 บิต และมี output nroot 16 บิตเป็นค่า root ที่ R ของ A ที่เป็นจำนวนเต็มมากที่สุดที่น้อยกว่าเท่ากับค่ารากที่ R ของ A ตัวอย่างเช่น

$$A = 10, R = 3, \text{nroot} = 2 \rightarrow 2*2*2 = 8 < 10 \text{ (if } \text{nroot}=3 \text{ then } 3*3*3 = 27 > 10)$$

$$A = 256, R = 4, \text{nroot} = 4 \rightarrow 4*4*4*4 = 256$$

ในช่วงเริ่มต้นให้ busy มีค่าเป็น 0 และ nroot มีค่าเป็นอะไรก็ได้ โดยวงจรจะเริ่มทำงานเมื่อ busy มีค่าเป็น 0 และมีการเปลี่ยน input จาก 0 เป็น 1 ระหว่างการทำงานให้ busy เป็น 1 และเมื่อคำนวณเสร็จแล้วให้แสดงค่ารากที่ R ใน nroot และให้ค่า busy กลับเป็น 0 โดยที่ค่า R ในทุก testcase จะมีค่ามากกว่า 0 เสมอ โดยอาจจะมีการ input ให้วงจรมีการทำงานมากกว่า 1 ครั้งต่อ 1 ได้

หมายเหตุ 1 : ช่วงระหว่างคำนวณอยู่ไม่ต้องสนใจสัญญาณ input และให้แสดงค่ารากที่สองของ A ค้างไว้จนกว่าจะมีการเปลี่ยน input จาก 0 เป็น 1 ใหม่จึงสามารถเปลี่ยนค่าได้

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

หมายเหตุ 3: ตัวตรวจจะรอสัญญาณไม่เกิน 255 cycles (ดังตัวอย่างใน template_02.dig)

คะแนน

- 30 คะแนนสำหรับคำตอบที่ถูกต้อง ในกรณี R = 2
- 70 คะแนนสำหรับคำตอบที่ถูกต้อง ในทุกๆกรณี

Hint 1: สามารถใช้ อนุกรม Multiplication หรือ Division ซึ่งอยู่ใน Arithmetic ได้

Hint 2: เลขยกกำลังสามารถทำได้โดยการแตกการคูณ เช่น

$$5^{10} = 5^8 * 5^2$$

ตัวอย่าง Testcase อยู่ใน template_02.dig

ข้อมูลนำเข้า

- Input: A ขนาด 16 Bit
- Input: R ขนาด 8 Bit
- Input: input ขนาด 1 Bit
- Clock: clk

ข้อมูลส่งออก

- Output: nroot ขนาด 16 Bit
- Output: busy ขนาด 1 Bit

ชุดข้อมูลทดสอบ

ตัวอย่างชุดข้อมูลทดสอบมีอยู่ใน template_02.dig

คะแนน

คะแนนเต็ม 100 คะแนน