Problem 2: Square Root

สร้างวงจรหารากที่สองของinput โดยให้ input เป็นเลข **A** 16 บิต, **input** 1 บิต และมี output **sqrt** 8 บิตเป็นค่ารากที่สองของ **A** ที่เป็นจำนวนเต็มมากที่สุดที่น้อยกว่าค่ารากที่สองของ **A** (ตัวอย่างเช่น **A** = 10, **sqrt**= 3)
วงจรจะเริ่มทำงานเมื่อมีการเปลี่ยน **input** จาก 0 เป็น 1 ระหว่างการทำงานให้ **busy** เป็น 1 และเมื่อคำนวณเสร็จสิ้นแล้วให้แสดงค่ารากที่สองใน **sqrt** และให้ค่า**busy** กลับเป็น 0

หมายเหตุ 1 : ช่วงระหว่างคำนวณอยู่ไม่ต้องสนใจสัญญาณ **input** และให้แสดงค่ารากที่สองของ **A** ค้างไว้จนกว่าจะมีการเปลี่ยน start จาก 0 เป็น 1 ใหม่จึงสามารถเปลี่ยนค่าได้
หมายเหตุ 2 : input A มีค่าตั้งแต่ 0 เป็นต้นไป และจะไม่เปลี่ยนแปลงในช่วงที่ busy เป็น 1
หมายเหตุ 3: ตัวตรวจจะรอสัญญาณไม่เกิน 80 cycles (ดังตัวอย่างใน **template_02.dig**)
คะแนน
- 50 คะแนนสำหรับคำตอบที่ถูกต้อง ในกรณี A < 6400
- 20 คะแนนสำหรับคำตอบที่ถูกต้อง ในกรณี 6400 <= A < 12800
- 20 คะแนนสำหรับคำตอบที่ถูกต้อง ในกรณี A >= 12800

Hint 1: สามารถใช้ อุปกรณ์ **Multiply** หรือ **Division** ซึ่งอยู่ใน Arithmetic ได้
Hint 2: It is easier to get full score on the problem 3 more than this one. But 50 is easy.

Note from the instructor There are many methods for finding square root, fast. If you are adventurous, you can start with Newton-Raphson, but if you start doing research, you will go down in a blackhole of numerical method and a lot of magic. For this problem, there is a not so hard way to get a full credit. **Here is another hint. Hardware works in parallels, you can have as many comparators and multipliers you would like (Grader should be able to accept 1Mbytes now.....)**

ตัวอย่าง Testcase อยู่ใน template_02.dig

ข้อมูลนำเข้า

- Input: **A** ขนาด 16 Bit
- Input: **input** 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 90 นาที จะได้อีก 10 คะแนน