

## เพื่อให้ผู้เรียนฝึกหัดการออกแบบและสร้างวงจรเชิงตรรกะ Sequential

การทดลอง(อนุญาตให้ใช้อุปกรณ์อะไรก็ได้ แต่ห้ามใช้วิธี table lookup)

ออกแบบโดยใช้ FSM Chart และสร้างวงจรหาตัวหารร่วมมาก (GCD) โดยวงจรมี input คือเลขฐานสอง 8 บิต A และ B และสัญญาณ Start และมี output คือค่าหาร แสดงด้วยผลด้วย Hex Display สองตัว และสัญญาณ Busy (output)

วงจรจะเริ่มทำงานเมื่อมีการเปลี่ยน Start จาก 0 เป็น 1 โดยนำ A และ B มาคำนวณหาหาร ระหว่างหาให้ Busy เป็น 1 และเมื่อคำนวณเสร็จสิ้นแล้วให้แสดงหาร พร้อมทั้งให้ Busy กลับเป็น 0 ด้วย

เมื่อเริ่มวงจร จะกำหนดให้อย่างน้อยมีสัญญาณ clear เป็น 1 อย่างน้อย 1 รอบสัญญาณ clock

หมายเหตุ 1 : ช่วงระหว่างคำนวณอยู่ไม่ต้องสนใจสัญญาณ Start และให้แสดงค่าหาร ค้างไว้จนกว่าจะมีการกด Start ใหม่จึงสามารถเปลี่ยนค่าได้

หมายเหตุ 2 : input A, B มีค่าตั้งแต่ 1 เป็นต้นไป และจะไม่เปลี่ยนแปลงในช่วงที่ Busy เป็น 1

หมายเหตุ 3: จะรับประกันว่า clear จะเป็น 1 อย่างน้อย 1 cycle ก่อนเริ่ม start

**ตัวอย่าง** Algorithm ในการหาหาร.

```
// a, b >= 1

int GCD(int a, int b) {
    while ( a != b) {
        if (a>b)
            a = a - b;
        else
            b = b - a;
    }
    return (a>b)?a:b;
}
```

### ข้อมูลนำเข้า

- A ขนาด 8 Bit
- B ขนาด 8 Bit
- input ขนาด 1 Bit
- clear ขนาด 1 Bit

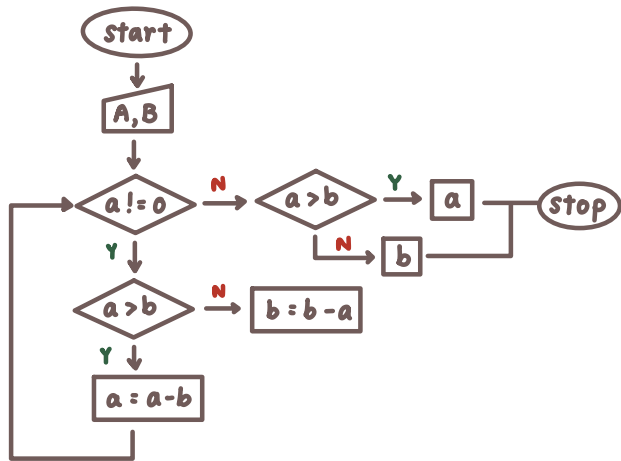
### ข้อมูลส่งออก

- output ขนาด 8 Bit
- busy

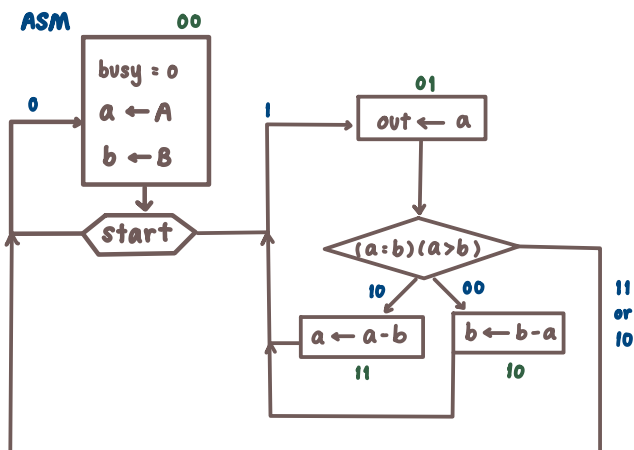
### ชุดข้อมูลทดสอบ

- 100% โปรแกรมทำงานถูกต้องตาม Input ทุกรูปแบบ

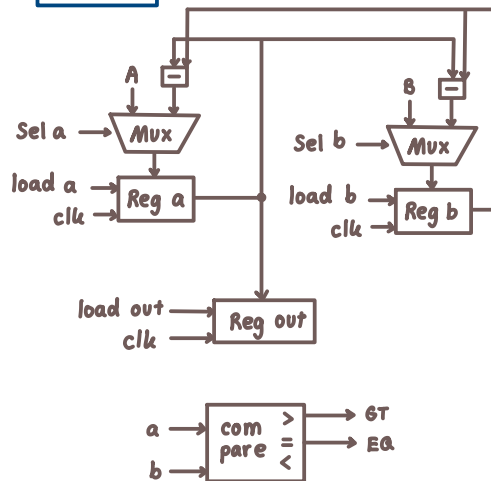
## flowchart



## ASM



## Data Path



## control

load a  
load b  
load out  
sel a  
sel b  
busy

$Q_a Q_b$	$Q_a^+ Q_b^+$	
00	00	start = 0
	01	start = 1
01	10	EQ = 0, GT = 0
	11	EQ = 0, GT = 1
	00	EQ = 1, GT = 1
		EQ = 1, GT = 0
10	01	None
11	01	None

