

Sorting circuit (Originally CEDT 2023 Final Project)

ให้นิสิตสร้างวงจรโดยใช้ Digital ที่ทำการเรียกข้อมูลจาก ROM memory และไปเก็บข้อมูลที่เรียกเส็จแล้วใน RAM memory เมื่อวงจรทำงานเสร็จสิ้นให้แสดงผลลัพธ์การเรียงค่าจากน้อยไปมากผ่านทาง

7-segment (สองตัว) เป็นเลขฐาน 16

ข้อมูลเข้าจะอยู่ใน File ซึ่งสามารถ Load ได้ผ่านทาง ROM (hex file) โดยเมื่อเริ่มวงจรให้มี input คือ

- N - จำนวนข้อมูลเข้า ($N < 512$)
- reset - ให้ reset การทำงานของวงจร
- start - ให้เริ่มทำงาน
- ROM file - เป็นข้อมูลที่ load เข้าก่อนเริ่ม simulation

และมี output คือ

- done - ขึ้นเป็น 1 เมื่อการทำงานเสร็จสิ้น (รวมการแสดงผล)
- valid - เป็นสัญญาณบอกว่าค่าที่ display ใน 7-segment มีผลลัพธ์ที่ถูกต้อง สามารถเป็น 1 สลับกับ 0 ได้ โดยระบบจะถือว่าเป็นค่าตอบที่เราไปตรวจสอบที่ valid เป็น 1 ไม่ต้องรอให้เรียงลำดับเสร็จก็สามารถหยุด display ได้เลย
- 7-segment 2 ตัว แสดงค่าผลลัพธ์ที่อยู่ใน RAM เป็นเลขฐาน 16 โดยจะแสดงค่าที่ถูกต้องที่สัญญาณ valid เป็น 1
- จำนวน Clock ที่ใช้ในการเรียงตัวเลข (รวม Clock ในช่วงที่วงจรแสดงผล) โดยนับเริ่มจากไดร์บสัญญาณ start
- RAM - ขนาดเท่ากับ ROM, RAM ที่นิสิตเลือกใช้สามารถเป็นอันใหม่ก็ได้ที่นิสิตจะเลือก

สามารถใช้อุปกรณ์ อะไหล่ได้เพิ่มเติมที่มีอยู่ใน Digital แบบมาตรฐาน ([เช่น ROM Dual Port](#))

โดยการทำงานของโปรแกรมจำนวนข้อมูลจะมีอยู่ในช่วง 0-511 ค่า

ค่าตัวเลขที่อยู่ใน ROM เป็นเลขอยู่ในช่วง 0-255 (8-bit) เช่นกัน โดยให้คำตอบที่อยู่ใน RAM เรียงจากน้อยไปมาก โดยให้สมมุติว่าค่าที่อยู่ใน ROM เป็นเลขจำนวนเต็ม

การทำงาน

นิสิตสามารถปรึกษา กันได้แต่ห้ามคัดลอกหรือตามคำダメใน Internet แต่สามารถค้นหาข้อมูลได้ใน Internet Search

การให้คะแนน (คะแนน ใน Lab นั้น)

- เอกสารด้านเทคนิค เช่น ASM Chart หรือ FSM Chart การทำงานของโปรแกรมไม่เกิน 5 หน้า (10 คะแนน)
- ทำงานถูกต้องในกรณีของค่าที่อยู่ใน ROM เรียงตัวกันอยู่แล้ว (10 คะแนน)
- ทำงานถูกต้องในกรณีของ $N \leq 4$ (10 คะแนน)
- ทำงานถูกต้องในกรณีของ $N \leq 64$ (10 คะแนน)
- ทำงานถูกต้องในกรณีของ $N \leq 128$ (10 คะแนน)
- ทำงานถูกต้องในกรณีของ $N \leq 256$ (10 คะแนน)
- ทำงานถูกต้องในกรณีของ $N < 512$ (10 คะแนน)
- ทำงานถูกต้องทุกกรณีและเวลาการทำงานไม่เกิน 8,000 clocks (5 คะแนน)
- ทำงานถูกต้องทุกกรณีและเวลาการทำงานไม่เกิน 4,000 clocks (10 คะแนน)
- ทำงานถูกต้องทุกกรณีและเวลาการทำงานไม่เกิน 2,000 clocks (10 คะแนน)
- การแสดงผลของ Clock ถูกต้อง (สามารถ +/- ได้ไม่เกิน 1) (5 คะแนน)
- ทุกๆ สมาชิกที่เกินกว่า 4 คน จะถูกหักคะแนนละ 5 คะแนน (-5 คะแนน)

ตัวอย่างการทำงาน

สมมุติให้ข้อมูลที่เก็บไว้ใน ROM เป็น 0x1F, 0x10, 0xFF, 0x00 เมื่อ สัญญาณ start เป็น 1 ให้เริ่มเรียงข้อมูล เมื่อเสร็จให้แสดง ข้อมูลผ่าน 7-segment เป็น 00, 10, 1F และ FF การแสดงผลลัพธ์อาจไม่ต่อเนื่องได้โดยใช้สัญญาณ valid =1 และแสดงว่าเป็นการแสดงผลลัพธ์ค่านั้นๆ เป็นค่าที่ถูกต้อง เช่น ในการนี้ที่แล้ว user อาจแสดงผลเป็น (7-Segment, valid) = (00, 1), (00, 0), (10, 1), (10, 0), (1F, 1), (FF, 1) เป็นต้น

หลังจากการแสดงผลเสร็จสิ้นให้ แสดง clock จำนวน clock ที่ใช้ในการทำงานและเปลี่ยนสัญญาณ done = 1 และให้ค่าทุกอย่างคงเดิมจนกระทั้งมีการกด reset และ start ใหม่

Template

Template อยู่ใน MCV, User สามารถเปลี่ยน RAM เป็น อุปกรณ์ RAM ชนิดอื่นได้ตามความเหมาะสม ค่าที่อยู่ใน ROM สามารถ Edit เพื่อใส่ค่าที่ต้องการหรือ ใช้ “Reload model at start” ใน Advanced tab ที่ต้องเตรียม Binary File ก่อน

Extra Credit

- กลุ่มที่วงจรทำงานเร็วที่สุดและได้คะแนนเต็ม 3 คนแรกจะได้รางวัลพิเศษ

** Please set your ROM to be “Program Memory” (in Advanced tab) for grader**

ตัวอย่าง Program ที่ใช้ในการ Generate Binary File สำหรับ ROM

```
#include <stdio.h>
int main(int argc, const char **argv) {
    const int N = 4;
    unsigned char buf[] = {0x1f, 0x10, 0xff, 0x00};
    FILE *fo = fopen("test.bin", "wb");
    fwrite(buf, N, 1, fo );
    fclose(fo);
    return 0;
}
```