

CS5120 Homework Assignment 02

Student ID: 107065507

Name: 盧允凡

1. Design Concept

Goal: 根據 hw01 的 8-bit GCD engine，擴充為 16-bit GCD engine。input, output 為 8-bit signals。

修改 FSM 為 5 個 states，在 READDATA 停留 2 個 cycles，1st cycle 輸入 A[15:8], B[15:8]，2nd cycle 輸入 A[7:0], B[7:0]。原來的 FINISH 拆成 FINISH 和 FINISH_2，第一個輸出 Y[15:8]，第二個輸出 Y[7:0]。並且搭配 DONE 看 output。

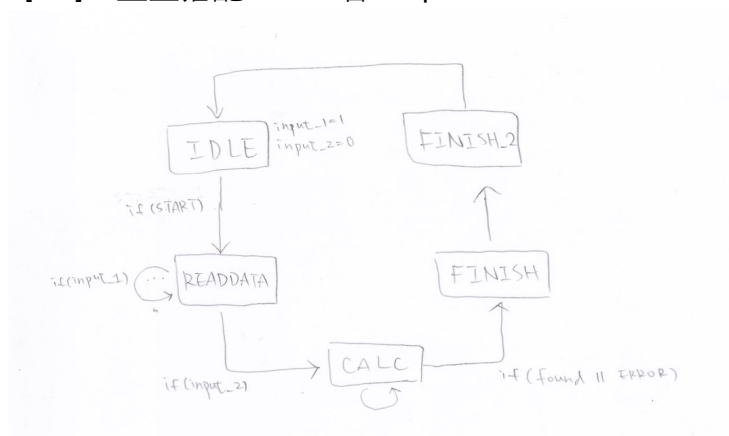


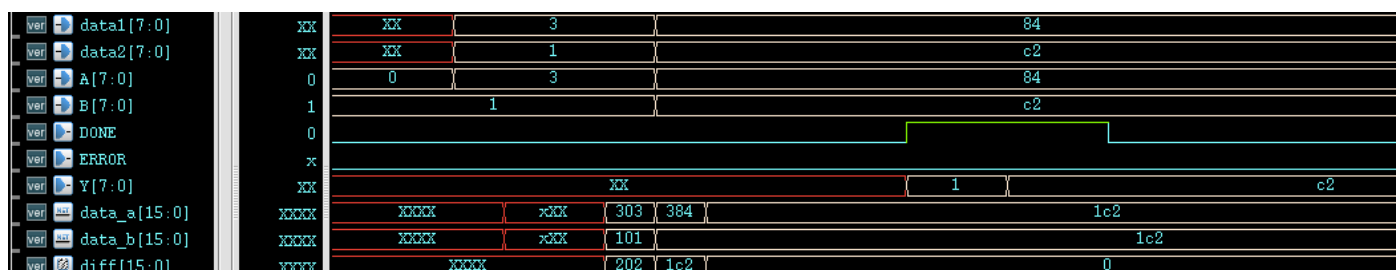
Figure 1. FSM

2. Simulation and Discussion

Case: RTL simulation

首先先看 RTL code 的 simulation 結果

Input: gcd(900,450)



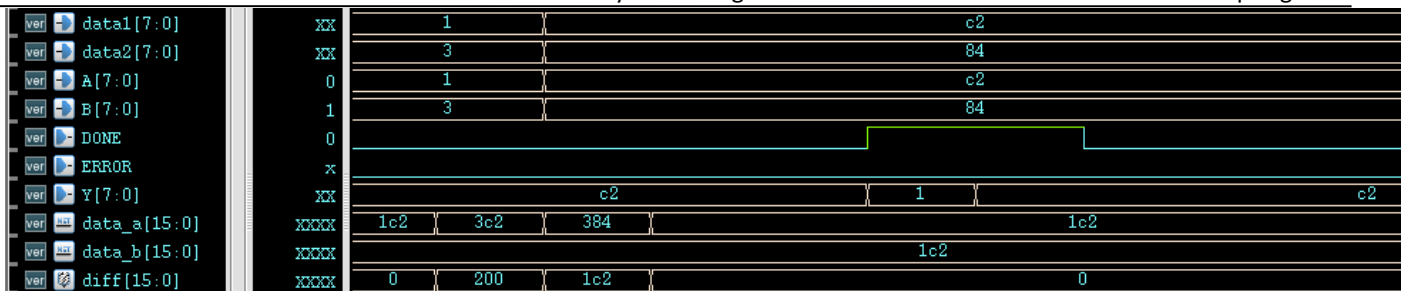
When DONE==1, Y=1 in the first cycle, and Y=c2 in the second cycle.

outputs: 1 and c2

即從 DONE=1 的 2 cycles 可以讀到 gcd(900, 450) = 450。

Input: gcd(450, 900)

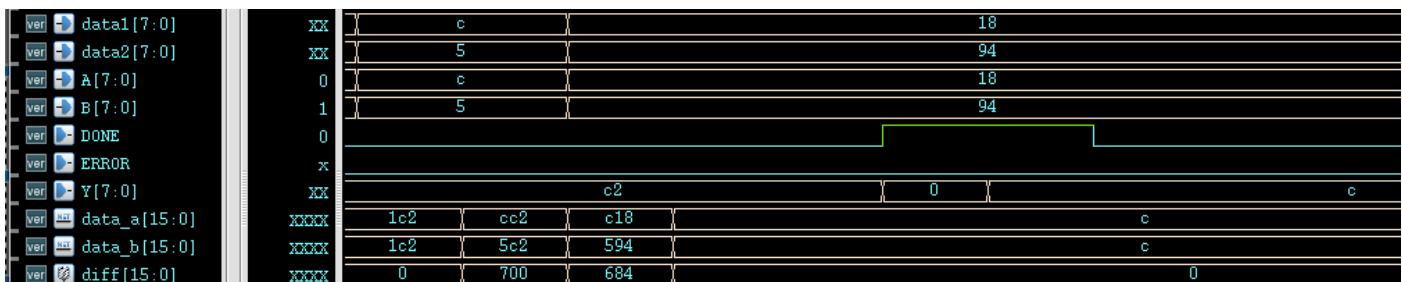
交換 A, B 的次序



outputs: 1 and c2

$\text{gcd}(450, 900)=450$

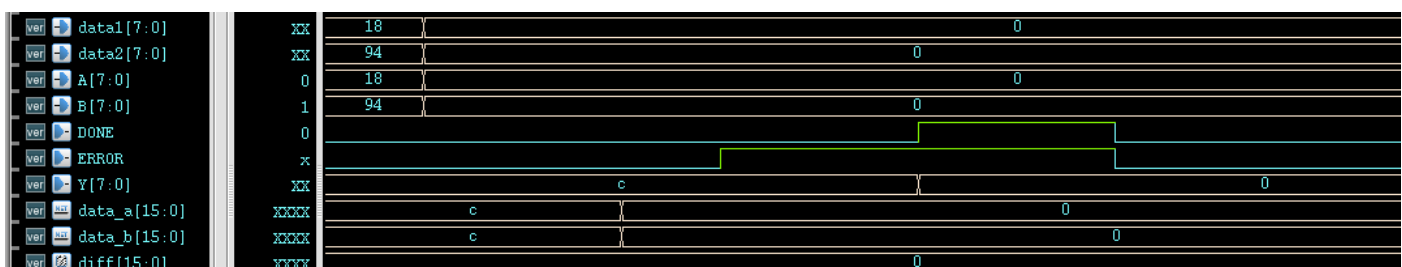
Input: gcd(3096, 1428)



outputs: 0 and c

$\text{gcd}(3096, 1428)=12$

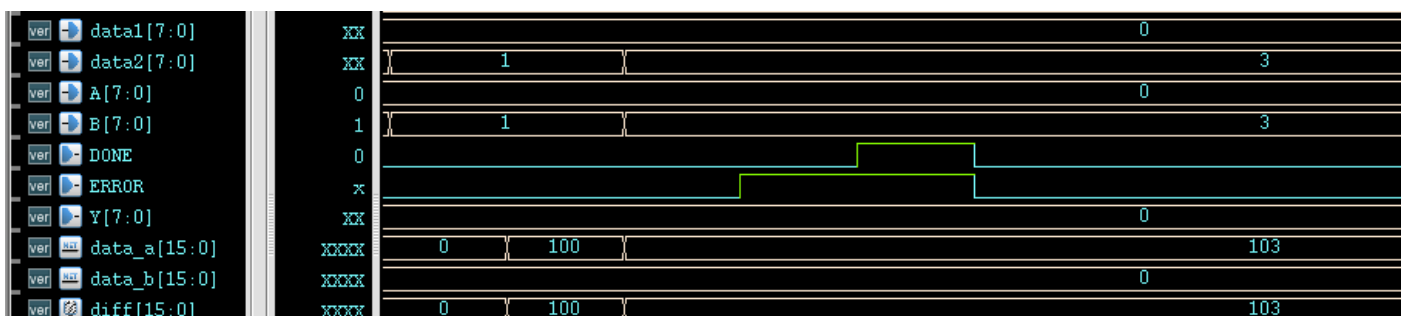
Input(0, 0)



outputs: ERROR

因為 inputs=0, ERROR singal=1

Input: (0, 259)



outputs: ERROR

其中一個 input=0 也會造成 ERROR

Synthesis

使用 Synopsys 的 design_vision 軟體，可以幫助我們合成。

根據講義上步驟依序操作，可以得到 hw02_syn.sd 以及 hw02_syn.v 的檔案。

如果都沒有出現錯誤訊息，表示我的 RTL code 是可合成的。

也可以向軟體要求 Report:

Report Area

```
Number of ports:                79
Number of nets:                 400
Number of cells:                317
Number of combinational cells:  258
Number of sequential cells:     58
Number of macros/black boxes:   0
Number of buf/inv:              87
Number of references:           32

Combinational area:             2254.147223
Buf/Inv area:                   322.505995
Noncombinational area:          1429.210802
Macro/Black Box area:           0.000000
Net Interconnect area:          undefined (No wire load specified)

Total cell area:                 3683.358025
Total area:                      undefined
design_vision>
design_vision>
```

Report Timing Path

```
*****
Report : timing
       -path full
       -delay max
       -max_paths 1
       -sort_by group
Design : GCD_1
Version: K-2015.06-SP1
Date   : Fri Mar 22 14:54:30 2019
*****

Operating Conditions: slow   Library: slow
Wire Load Model Mode: top

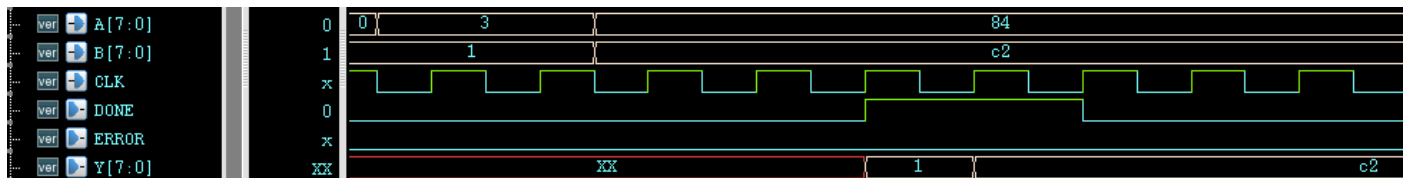
Startpoint: ERROR_reg (rising edge-triggered flip-flop clocked by CLK)
Endpoint:   ERROR (output port)
Path Group: (none)
Path Type:  max

Point                Incr      Path
-----
ERROR_reg/CK (DFFRX1) 0.00    0.00 r
ERROR_reg/Q (DFFRX1)  0.50    0.50 f
ERROR (out)            0.00    0.50 f
data arrival time      0.50
-----
(Path is unconstrained)
```

接下來可以使用 ncverilog 和 nWave 來看 gate-level simulation 是否與 RTL simulation 的結果一致。

Gate-level simulation

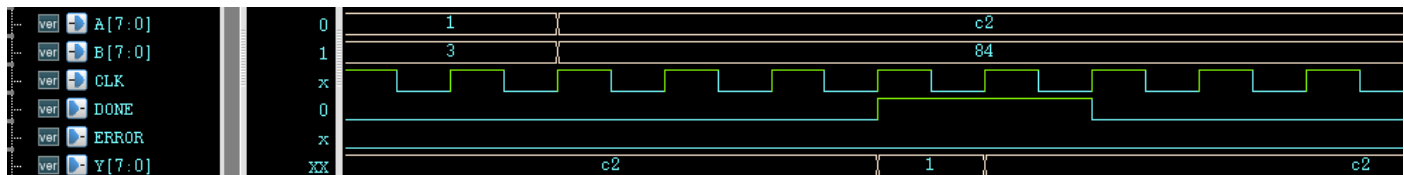
Input: gcd(900,450)



outputs: 1 and c2

$\text{gcd}(900, 450)=450$

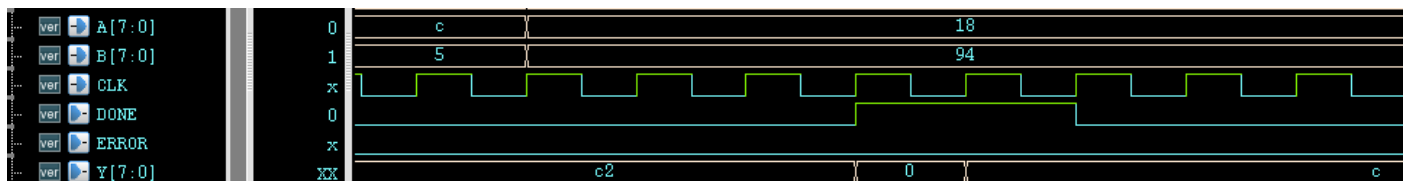
Input: $\text{gcd}(450, 900)$



outputs: 1 and c2

$\text{gcd}(900, 450)=450$

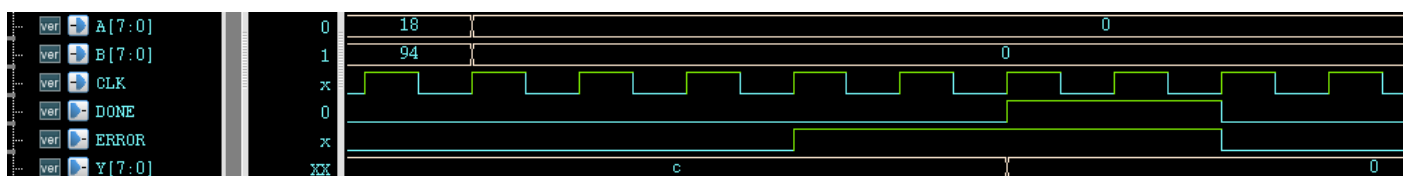
Input: $\text{gcd}(3096, 1428)$



outputs: 0 and c

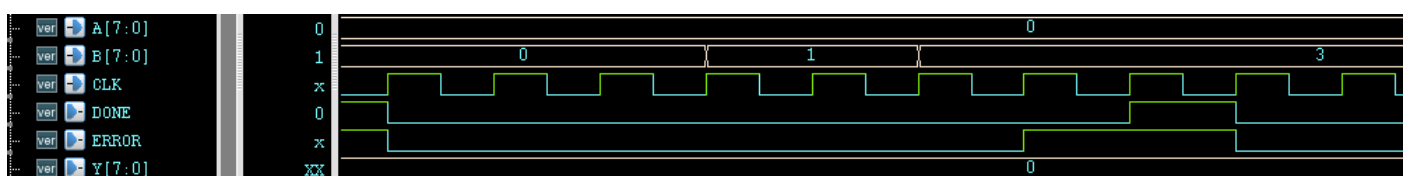
$\text{gcd}(3096, 1428)=12$

Input(0, 0)



outputs: ERROR

Input: (0, 259)



outputs: ERROR

經過檢查後，合成後的 gate-level simulation 與 RTL simulation 結果一致。

3. Summary

將 8-bit GCD engine 擴充為 16-bit 看似簡單，以為只需要簡單修改前一份作業即可。實際做起來卻很困難，花了相當多時間才把 cycle 搞定！之後 Synthesis 也是覺得照步驟做下來即可，但實際做起來也發生了無法合成的問題，得回去修該 RTL code。

也有發生 simulation 過程卡住，後來把 always 改寫為 assign，程式順利執行，推測 assign 會判定為 combinational circuit，所以 ncverilog compile 過程會變比較簡單。

總結，這次 lab 學到了簡易合成的步驟，design_vision 的使用，以及 Synthesis 的概念。