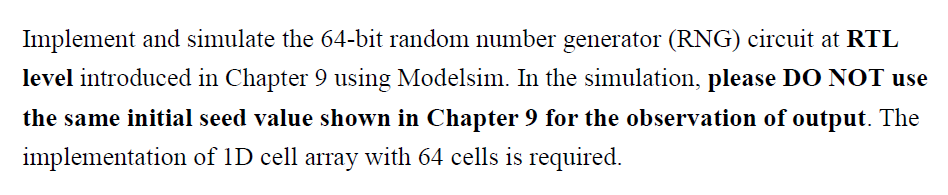
HW4 M10707617 梁松澤

**第一部分**

**My output as a textfile：**



1. 程式碼以及程式內容概述

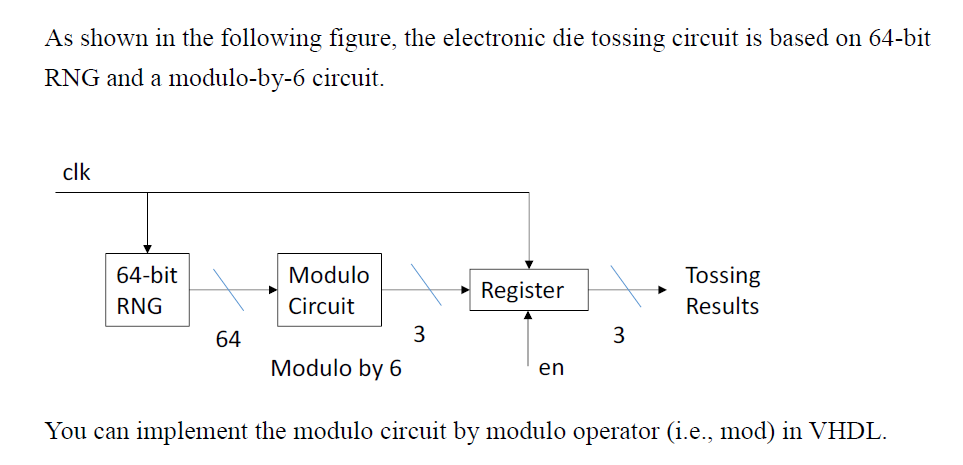
* VHDL Code

|  |
| --- |
| **Cellular Automata** |
|  |
| **Random Number Generator** |
|  |

|  |
| --- |
| **Test Bench** |
|  |

作業4的第一部分是做有 RNG 亂數產生器的 Test Bench 練習，參考老師的 code 後，我有稍微修該並且自己設計亂數產生的規則，每個 Cells 四個輸入部分我改成 {a, b, c, d} = {-2, -1, 0, 3}，而 Look-Up-Table 我在 Cellular Automata 的程式裡面有自行稍微修改一下輸出 0 跟 1 產生的規則，這部分還好還有老師講義裡的程式可以做參考，所以修改起來還算是簡單。

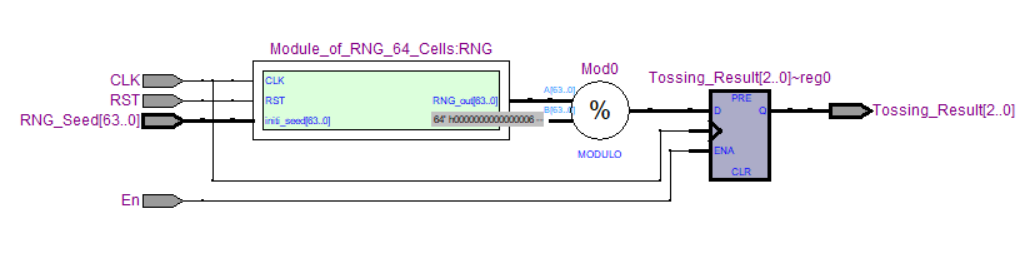
**第二部分**



第二部分的作業為佳芬選項

1. 程式碼以及程式內容概述

* RTL Viewer 合成電路圖



* VHDL Code

|  |
| --- |
| **Electronic Dice Circuit** |
|  |

1. 波形圖(Simulation results)

* 自行設計驗證波型數值：(Is Correct)

|  |
| --- |
| 一張含有 桌 的圖片  自動產生的描述 |
| 這個部分我有把 RNG\_Seed 特別拉出來，這樣方便更改 Seed，而在 Reset(RST) = ‘1’的時候，Seed 就是 RNG 亂數的初始值，En(Enable) 則分別在 1.8 us ~ 2.1 us 和 4.8 us ~ 5.1 us 時是 ‘0’，所以Output 在這段時間裡都維持 著 3 和 5 不變，其他時間 En 則維持著 ‘1’ 使得 Register 可以一直進行讀取，而每次亂數跳時跟暫存器讀取值都是在 Clock 正緣時才會觸發，所以可以看到結果 Tossing\_Result 都是在 0 ~ 5 之間亂跳，代表著電子骰子的六個面。  而在做這個作業時，我覺得最重要的地方是 Clock Rate 的部分，由於之前一直是在 < 1 us 的 Clock Rate 跑 WaveForm，所以一直不知道為什麼有時候 CLK 正緣時，最後的 Output 一直會出現 6 跟 7 的數值，由於 Output 是長度為 3 的向量，所以最大值就會出現 7 (111)，於是我覺得很不合理，好像 RNG 產生的亂數根本沒有做 MOD的計算，所以一直反覆糾結在 Modulo 電路的部分，在不斷地重新設計和驗證結果後，發現到 Modulo 電路在把 Clock Rate 拉長後，用滿多組作業第一部份的 Textfile 的值丟進去後，其 MOD 6 的結果都是對的，故最後終於找到了 Output 出錯的原因：Clock Rate 不夠大所導致的，以至於 Modulo 計算後的結果來不及在下一次的 Clock 正緣觸發前送去給 Register。 |

* 使用助教提供範例做測試：(Is Wrong)

|  |
| --- |
| 一張含有 桌 的圖片  自動產生的描述 |
| 順便補充一下，我也有使用助教提供的範例裡的 Clock Rate = 100 ns/clk做測試，發現有的時候會出現 6 跟 7 的錯誤，所以推測原因可能是因為我 RNG 的 Look-Up-Table 和 Cells 的四個 inputs 連接有重新自己設計的關係，跟老師講義中的 RNG 範例不太一樣，導致助教的 Clock Rate 對我的電路來說還是算太小了，以至於 Output 發生出錯，但把 Clock Rate 改成 150 ns/clk 後，就沒有再發生 Output 有跑出 6 跟 7 的問題了。 |

1. 心得與問題討論(corresponding discussions)

作業4 除了練習亂數產生器(RNG)的設計外，還練習了如何寫 Test Bench，然後老師與再額外出一份加分題，是使用亂數產生器(RNG)設計一顆電子骰子，看電路是滿簡單的，但在測試 WaveForm 的時候，因為 Output 一直不如預期，就算不斷修改程式 debug 還是找不到問題的點，甚至一度找到有點崩潰，本來還想說放棄了呢，但還好，在不斷地測試之下，偶然發現到把 Clock 拉長後，試驗的幾組數值，其 Output 都是對的，所以想起了老師之前在上課時有稍微提到 Clock Rate 的問題，後來把 Clock Rate 加大之後，其所呈現的亂數終於沒有再看到有 6 (110) 跟 7 (111) 的數值了，這時我猜測我應該是做對了，還好最後能夠成功，沒有枉費我熬了多天的夜在 debug。