CS5120 Final Project

Student ID: 107065507 Name: 盧允凡

1. Design Concept

Goal:

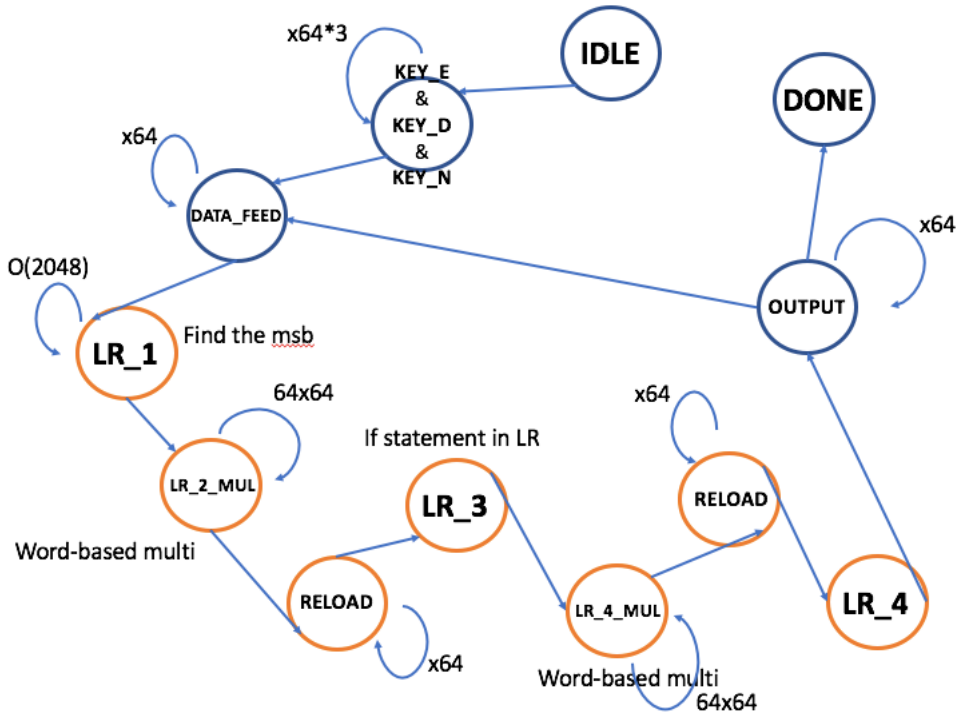
Design a RSA engine，public key = (E, N), private key = (D, N), input data = A。E, D, N and A are all 2048-bit. The IO width is 32-bit. i.e., to compute A^E (mod N).

我使用 LR algorithm 實作 RSA，因為要處理大數運算，所以使用 word-based multiplication 來實作乘法的部分。

LR Algorithm

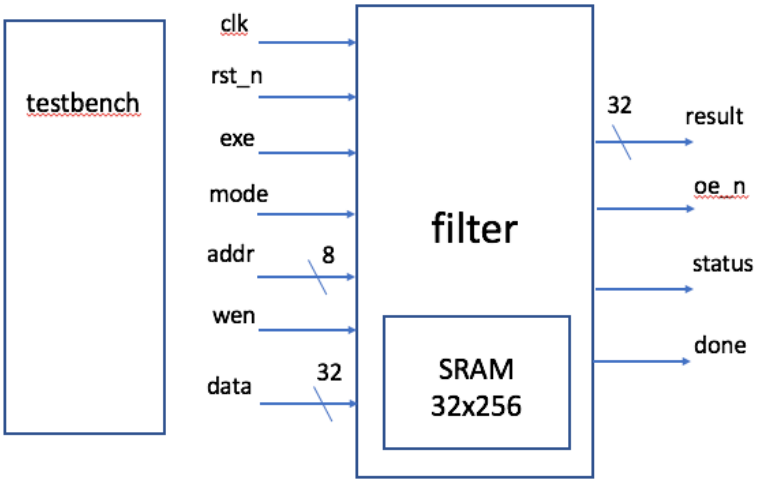
LR Binary Method
Input: M, e, n
Output: $C := M^e \bmod n$
1. **if** $e_{h-1} = 1$ **then** $C := M$ **else** $C := 1$
2. **for** $i = h - 2$ **downto** 0
2a. $C := C \cdot C \pmod n$
2b. **if** $e_i = 1$ **then** $C := C \cdot M \pmod n$
3. **return** C

Finite State Machine (FSM)



KEY_E, KEY_D, KEY_N and DATA_FEED: 為 2048-bit 的資料，花費 64 cycles 輸入
LR algorithm (圖中橘色部分):
LR_1: 找到 most significant bit (msb)，需花費 O(2048)
LR_2_MUL and LR_4_MUL: word-based multiplication，需花費 64*64 cycles
RELOAD: 將完成乘法後的新 C 值，重新 load 回 register，需花費 64 cycles
LR_3: if statement in LR algorithm (演算法中第 2b 行)
LR_4: 演算法是否結束的判別
OUTPUT: 將加密後的 text 輸出，需花費 64 cycles

Block Diagram



SRAM
使用 32-bit*256 的 single-port memory
面積估測:
The area of a single-port 32*256 SRAM is 43400 um2.

Cycle Analysis
Input key: 3*64
N data feeding: N*64
LR algorithm: N*(O(2048)*64*64+64)
Output: N*64

Word-Based Multiplication

因為 2048-bit 很大，直接做乘法的 multiplier 會太大，所以使用 word-based 的方式做乘法。
將 2048-bit 切成 64 個 32-bit 的 word，需花費 64*64 cycles 做乘法。

- Word-Based

$$
\begin{array}{rcc}
 & A_H & A_L \\
\times & B_H & B_L \\
\hline
 & A_HB_L & A_LB_L \\
A_HB_H & A_LB_L & \\
\hline
\end{array}
$$

2. Simulation and Discussion

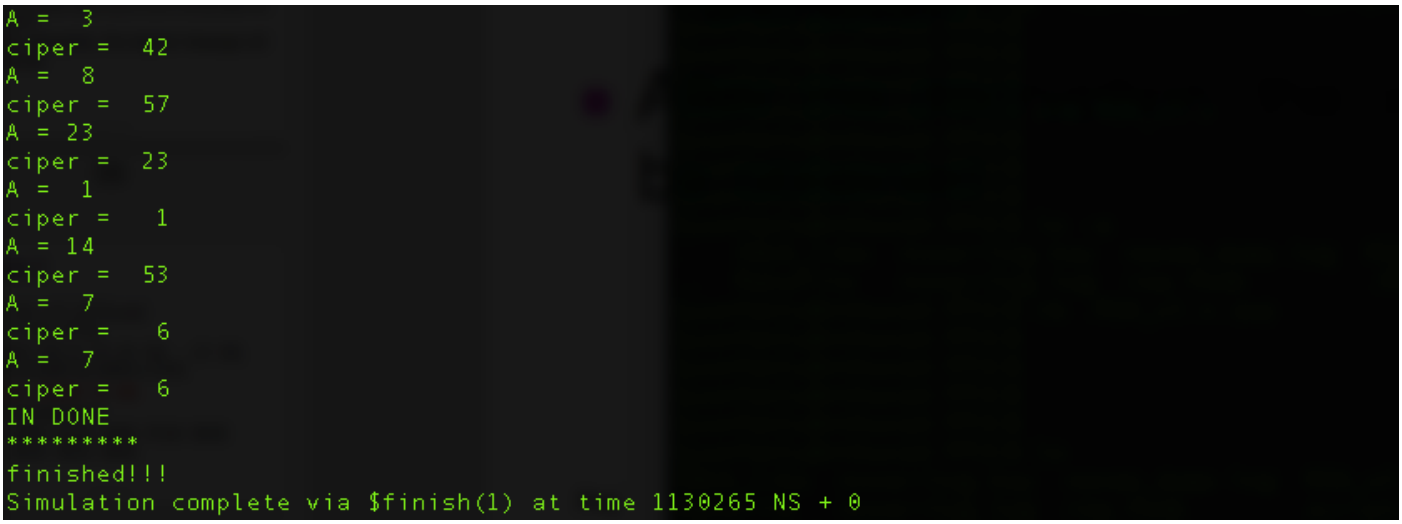
我的 RTL simulation 可以成功，但是無法 synthesis (合成時間過久，2 小時後 Design_Vision 回報合成失敗)。推測原因如下：
雖然做了 word-based 去處理乘法，但是 LR algorithm 中還需要做 modulo 運算，這個部分也是 2048-bit。所以這是造成除法器過大的原因。

RTL simulation

Example 1

Public key: (E, N) = (7, 143)

Message: 03, 08, 23, 01, 14, 07



The cipher text is: 42, 57, 23, 01, 53, 06

Private key: (D, N) = (103, 143)

Message: 42, 57, 23, 01, 53, 06

```
A = 42
ciper =   3
A = 57
ciper =   8
A = 23
ciper =  23
A =  1
ciper =   1
A = 53
ciper =  14
A =  6
ciper =   7
A =  6
ciper =   7
IN DONE
*********
finished!!!
Simulation complete via $finish(1) at time 2628105 NS + 0
```

After the decryption, the message is: 03, 08, 23, 01, 14, 07

Example 2

Public key: (E, N) = (157, 2773)

Message: 948

```
A = 948
ciper = 920
A = 948
ciper = 920
IN DONE
*********
finished!!!
Simulation complete via $finish(1) at time 26165 NS + 0
```

The cipher text is 920

3. Summary

因為我 modulo 的問題尚未解決，所以沒有合成後的 gate-level code。
後來發現 Montgomery algorithm 可以解決一次 multiplication 與 modulo 的問題，或許一開始就要用這個方法而不是用 LR 演算法，會比較簡單。但很遺憾因為時間的關係，沒辦法重做了。
其中有遇到 testbench 多筆 data 要 input 64 cycles，數量很大，所以用 script language (python)寫了簡單的腳本語言幫忙自動化工作。

因為我的 RTL code 合成失敗，所以無法提供 area, timing 以及 power reports。

總結這學期的 VLSI 課程受益良多，從完全不會 Verilog 到可以設計出一些 projects，也學習到了 VLSI 和 IC design flow 的相關知識。