

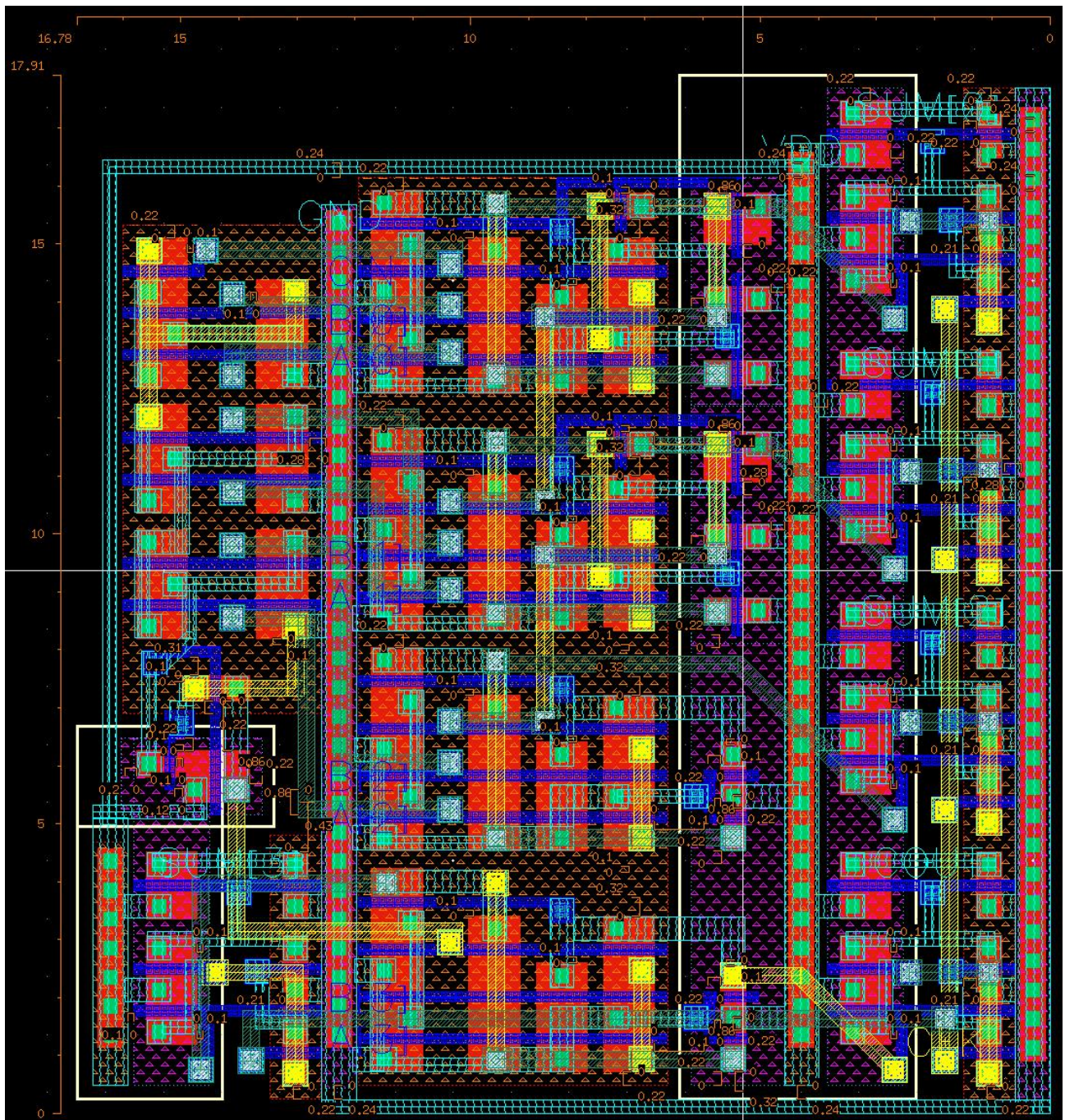
2022 NYCU EE VLSI Lab Report

Lab03 4-Bit Full Adder + DFF

Student ID: 109611070 Name: 郭家均 Date: 2022/11/19

I. Layout result

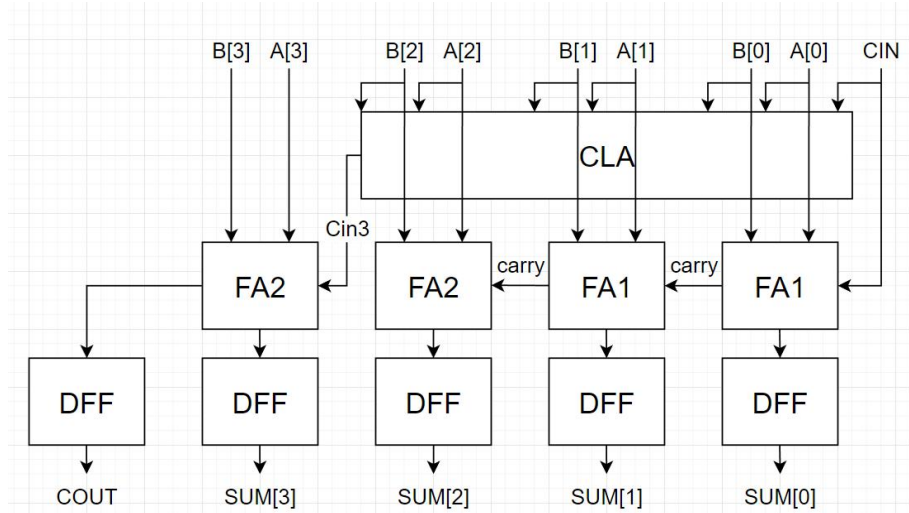
1. Layout picture with ruler (Height: 17.91um, Width: 16.78um)



2. Design concept

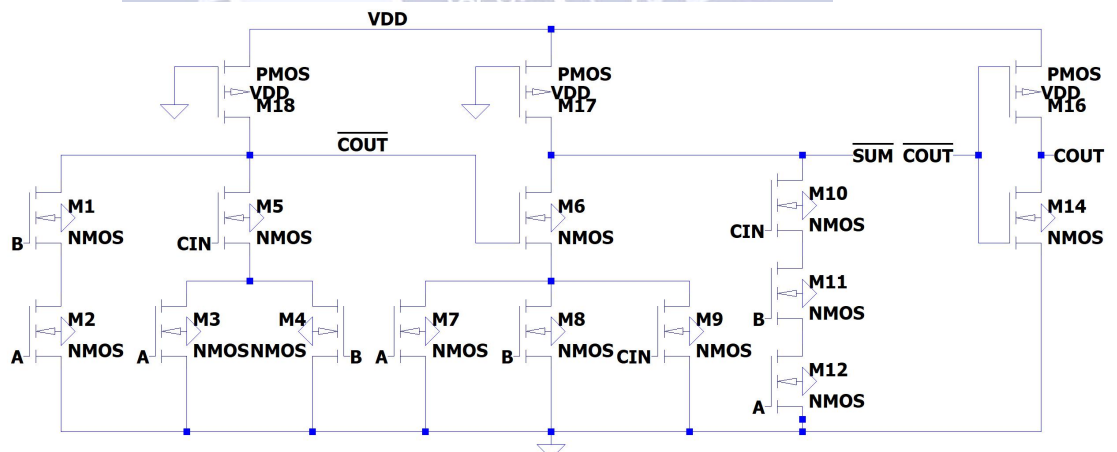
(1) Circuit Schematic

Block diagram :

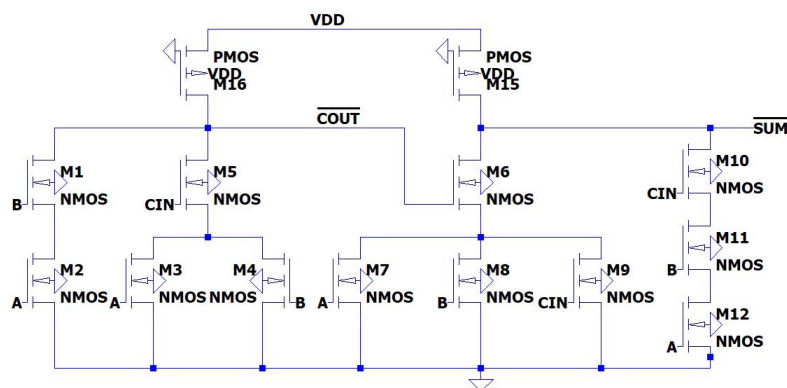


Sub-circuit : FA1, FA2, CLA, DFF

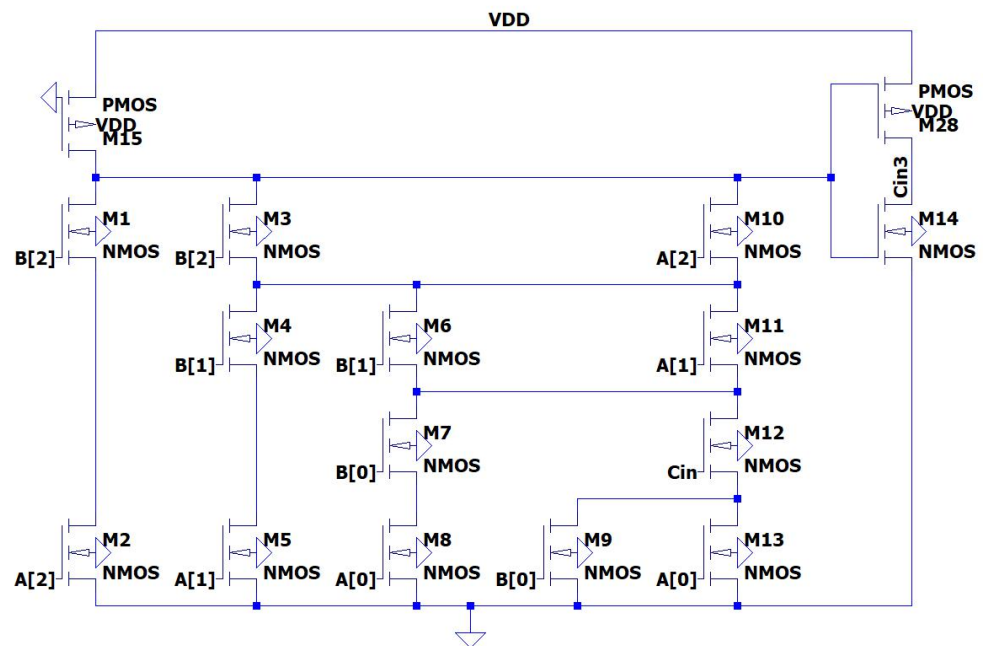
FA1 (since DFF have inverse output, SUM doesn't need inverter)



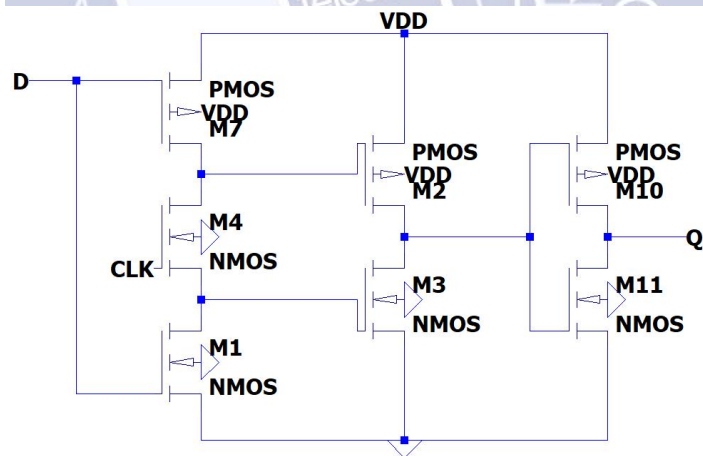
FA2 (since implementing CLA circuit, 2FA don't need COUT)



CLA



High level trig D-latch + Inverter 以下全部簡稱 DFF



(2) Summary of structure (number of transistor / logic gate is used)

此實驗我主要選用 pseudo C-CMOS Adder + pseudo compound gate CLA + DFF 來設計 4bit-FA。全部用了 111 個 Transistors。

簡單來說，前面三級 FA 是 ripple adder 架構，第四級 FA 的 carry-in 由一個 combinational CLA circuit 去運算。縮短 critical path 的時間。

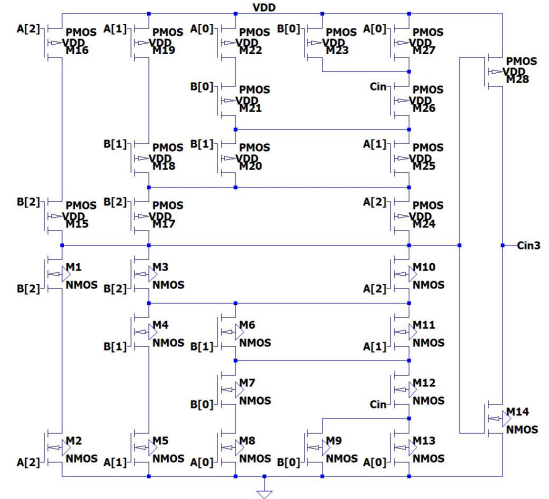
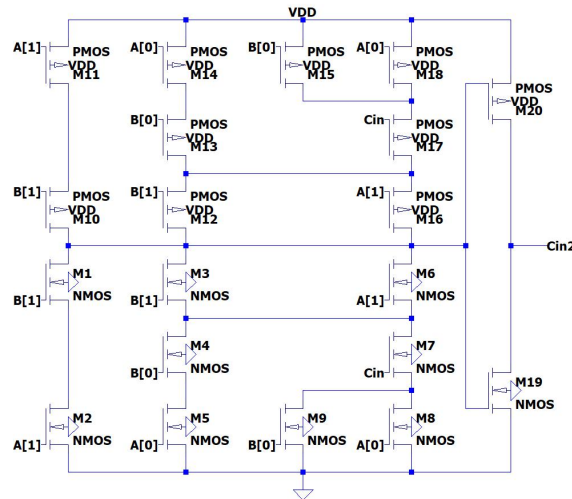
我們必須在 CLK posedge 前讓 FA 的 output stabilize，意即運算速度越快越好，因此在猜想不同架構時去比較不同 critical path 的長短，可以在沒有 err0 的狀況下使 CLK 最快為主要考量。

以下有一些 subckt 的代號解釋

(carry 1 邏輯運算因為和 C-CMOS 的算法相同因此不列入考慮)

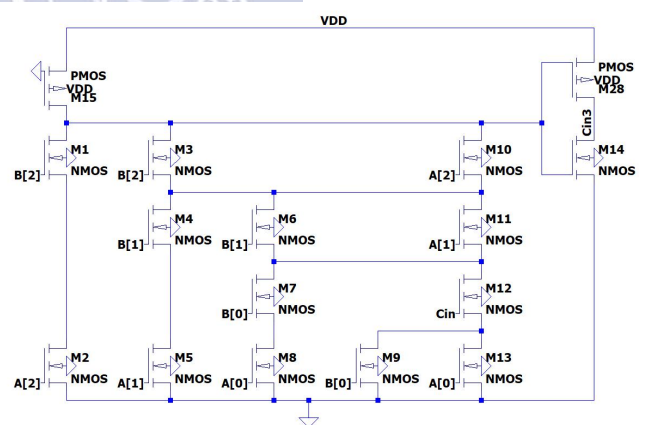
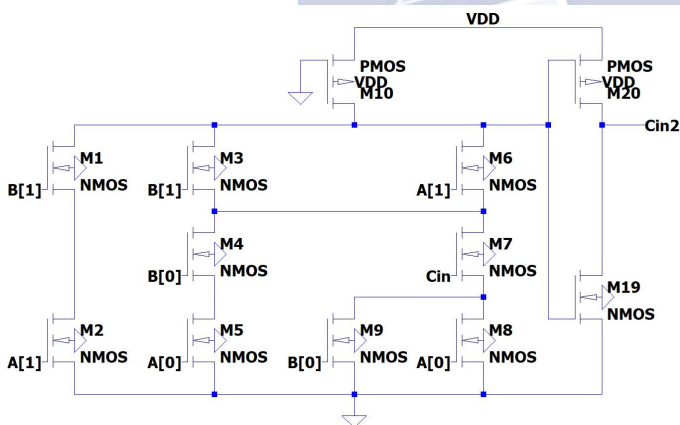
CLA2 : compound gate 架構下 output 得到 Cin2 (下左)

CLA3 : compound gate 架構下 output 得到 Cin3 (下右)



pCLA2 : CLA2 改成 pseudo NMOS 形式(下左)

pCLA3 : CLA3 改成 pseudo NMOS 形式(下右)

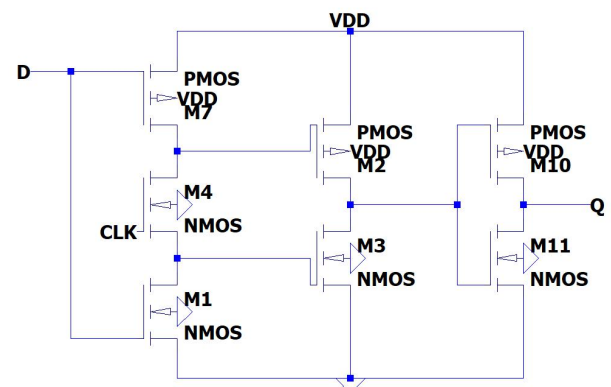
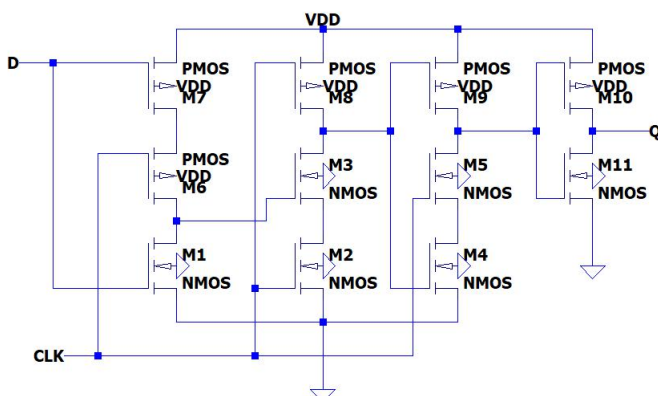


在實驗中我前後採用了兩種不同的 DFF 架構

DFF1 : 11 transistors , D in Q out (下圖左)

DFF2 : 7 transistors , D in Q_bar out (下圖右, output 名稱仍訂為 Q)

實質上為 High-level trigger D-latch + inverter



有測試過以下的幾種架構 (藍色為使用 DFF1, 橘色為使用 DFF2):

Design Structure	Critical Path	CLK	Adder Type
4-bit RCA	A[0] => SUM[3]	>1.66n	C-CMOS
2-bit RCA + CLA2 + 2-bit RCA	A[0] => CLA2 => SUM[3]	>1.66n	C-CMOS
3-bit RCA + CLA3 + FA	A[0] => CLA3 => SUM[3]	>1.66n	C-CMOS
2-bit RCA + pCLA2 + 2-bit RCA	A[0] => pCLA2 => SUM[3]	0.88n	Pseudo C-CMOS
4-bit RCA	A[0] => SUM[3]	0.68n	Pseudo C-CMOS
2-bit RCA + pCLA2 + 2-bit RCA	A[0] => pCLA2 => SUM[3]	0.62n	Pseudo C-CMOS
2-bit RCA + pCLA2 + FA + pCLA3 + FA	A[0] => pCLA3 => SUM[3]	0.52n	Pseudo C-CMOS
3-bit RCA + pCLA3 + FA	A[0] => pCLA3 => SUM[3]	0.52n	Pseudo C-CMOS

經過多次驗證和實驗後，我歸納出影響電路速度的三大個因素。

DFF 架構的差別、使用 pseudo NMOS 架構、使用 CLA 架構

我們先對 DFF1 和 DFF2 進行分析：(D-Latch vs DFF)

DFF1 的電路因為是 D in Q out，因此前方 adder 的電路接到 DFF 前必須先經由 inverter stage 拉為 SUM、COUT 的值。

DFF2 的電路為 D in Q_bar out，因此 adder 的 SUM output 不用加上 inverter stage，COUT 的部分如果再 RCA 的架構下，除了最後一級不用加 inverter stage 外，其他級因為是互相傳遞不會經過 DFF，所以都必須加上 inverter stage。

不過因為 adder 電路都採用 C-CMOS 為原型，因此單看最後一級 FA 可以發現 SUM 的 delay 都會比 COUT 來的大，因此使用 DFF2 的架構速度上會優於 DFF1，因為相當於 critical path 少了一級的 inverter delay。DFF1 的優點在前一級 adder 會經過 inverter stage 做 restore，因此整體比較 robust。

Pseudo NMOS 的架構根據上表可以明顯看出差距，若是用一般 C-CMOS 架構，最基本的 1.66ns 無法達成。Pseudo NMOS 大大改善了 PMOS driving 較小的問題，但相對上 power 的部分就會大很多。原因在於 Pseudo NMOS 的 logic 0 是在 NMOS 導通且 driving 能力為 PMOS 四倍的情況下達成，會有電流從 VDD 流到 GND。不過此次 lab performance 的部分不考慮 power 的問題，且設計的過程採 full custom design，和 stdcell design 取向不同，因此最後使用 pseudo 來做。

在此需要特別注意的是，一般數位電路的設計上會避免 pseudo NMOS 的架構。高度整合的晶片，例如 processor 裡面會有很多運算的單元，如果單位面積有大量電流持續導通，將造成功耗過大、晶片過熱、甚至讓電路整個燒掉。因此 EDA 抓取的 stdcell library 應該不會出現 pseudo 的架構。

CLA 在 4bit adder 下效果並不顯著，原因在於 bit 數實在太少。一般 RCA 的架構如果出現 8bit、16bit 等，其 critical path 會出現在 carry chain，因此在 bit 數多的情況下 CLA 會大大改善電路速度。但這次 lab 只有 4bit，CLA 用一般 complementary 的架構並沒有比 RCA 好多少。即使改成 pCLA 架構，改善幅度也只有 0.1~0.15ns 左右。甚至 pCLA3 + FA 的 delay 經過測試仍比 3bit RCA delay 大，可以從上表最

後一欄看到 critical path 為經過 pCLA3 的路徑，而非 $A[0] \Rightarrow \text{SUM}[2]$ 。

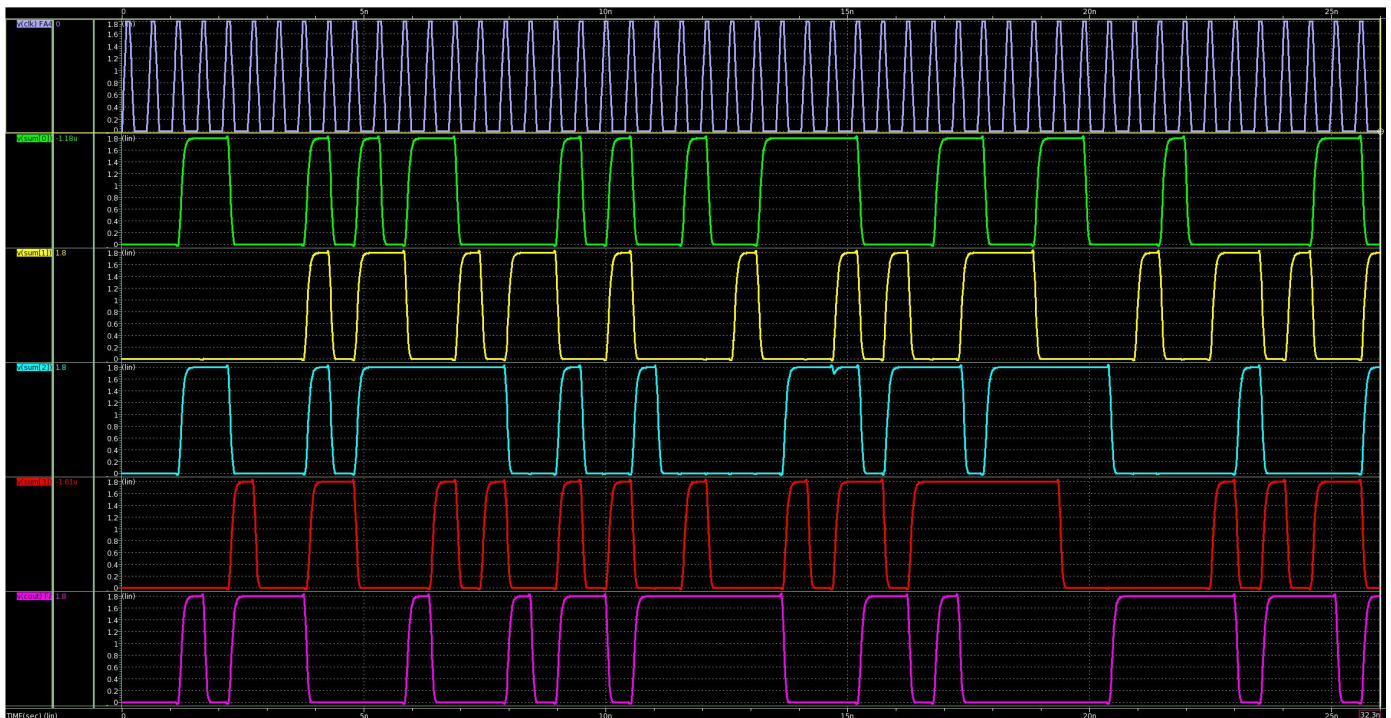
根據上表和分析過程，我最終選用 **3-bit RCA + pCLA3 + FA+DFF2** 的 Design。

上表最後兩個架構測出的 CLK 一樣快(critical path 為同一條)，因此選用 mos 數較小的 design，layout 起來面積也會比較小。

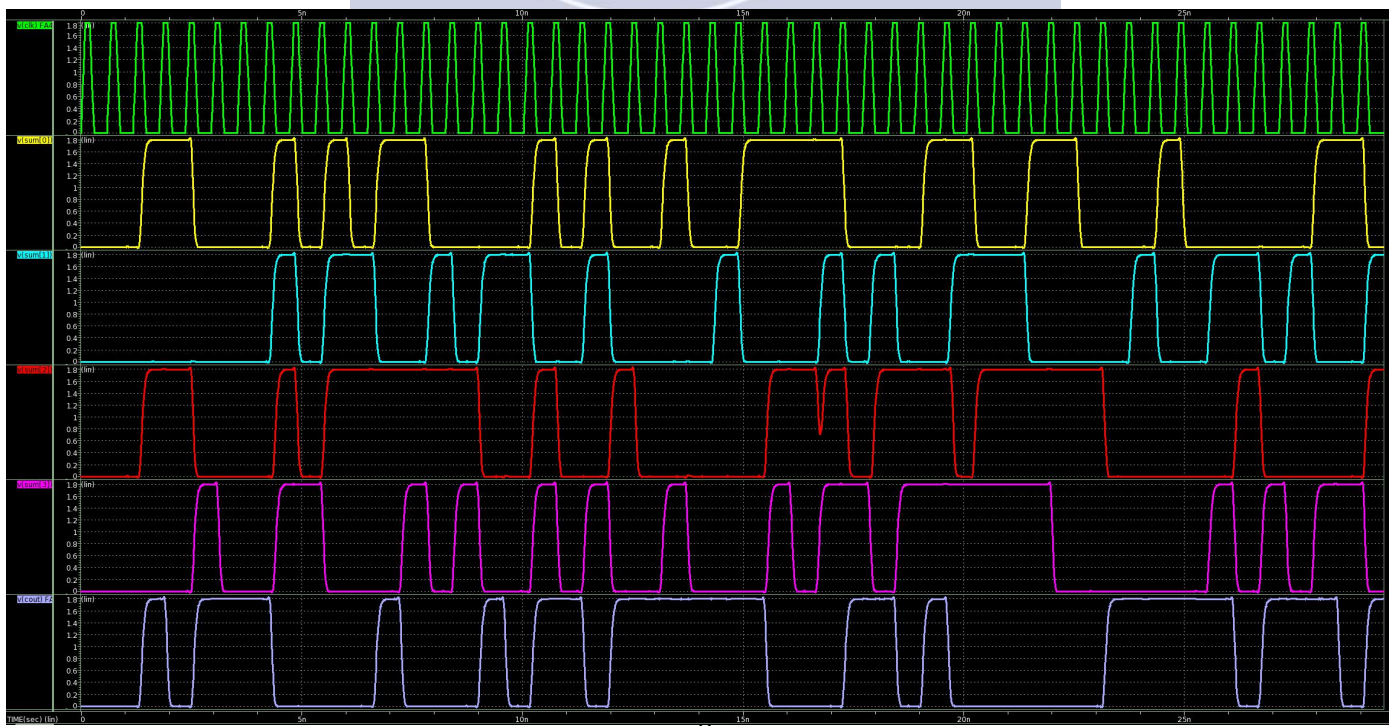
II. Simulation result

1. Output waveform

(1) Pre-sim (Output waveform)



(2) Post-sim (Output waveform)



(3) Performance list (TT case under worst case input pattern)

Maximum operation frequency	Pre-sim: 1.923 GHz
	Post-sim: 1.695 GHz
Average power	Pre-sim: 1.1982m
	Post-sim: 1.2002m
Layout area	16.78 *17.91 = 300.53 (μm^2)
4-bit full adder structure	Pseudo-NMOS C-CMOS Adder
Glitch control (Yes/No)	Presim Yes(< 0.2V) / Postsim No

Post-Sim have Glitch Control(< 0.2V) when CLK period >= 0.62ns

Table 1: Output Simulation Summary

	Spec.	Pre-sim	Post-sim
Worst Rise Time	< 0.8ns	122.8 ps	130.2 ps
Worst Fall time	< 0.8ns	83.0 ps	89.4 ps
Worst Propagation Delay	N/A	292.9 ps	339.5 ps
Average Power	N/A	1.1982 m	1.2002 m

Pre-sim rise/fall

```

.TITLE ***** lab03: 4-bit fa *****
iavg      pavg      worst_prop      rise_1sum[0]
rise_2sum[0] rise_3sum[0] rise_4sum[0] rise_5sum[0]
rise_6sum[0] rise_7sum[0] rise_8sum[0] rise_9sum[0]
rise_10sum[0] rise_11sum[0] rise_12sum[0] rise_13sum[0]
rise_2sum[1] rise_3sum[1] rise_4sum[1] rise_5sum[1]
rise_6sum[1] rise_7sum[1] rise_8sum[1] rise_9sum[1]
rise_10sum[1] rise_11sum[1] rise_12sum[1] rise_13sum[1]
rise_1sum[2] rise_2sum[2] rise_3sum[2] rise_4sum[2]
rise_5sum[2] rise_6sum[2] rise_7sum[2] rise_8sum[2]
rise_9sum[2] rise_10sum[2] rise_11sum[2] rise_12sum[2]
rise_3sum[3] rise_4sum[3] rise_5sum[3] rise_6sum[3]
rise_7sum[3] rise_8sum[3] rise_9sum[3] rise_10sum[3]
rise_11sum[3] rise_12sum[3] rise_13sum[3] rise_1cout
rise_2cout      rise_3cout      rise_4cout      rise_5cout
rise_6cout      rise_7cout      rise_8cout      rise_9cout
rise_10cout     rise_11cout     fall_1sum[0] fall_2sum[0]
fall_3sum[0] fall_4sum[0] fall_5sum[0] fall_6sum[0]
fall_7sum[0] fall_8sum[0] fall_9sum[0] fall_10sum[0]
fall_11sum[0] fall_12sum[0] fall_1sum[1] fall_2sum[1]
fall_3sum[1] fall_4sum[1] fall_5sum[1] fall_6sum[1]
fall_7sum[1] fall_8sum[1] fall_9sum[1] fall_10sum[1]
fall_11sum[1] fall_12sum[1] fall_1sum[2] fall_2sum[2]
fall_3sum[2] fall_4sum[2] fall_5sum[2] fall_6sum[2]
fall_7sum[2] fall_8sum[2] fall_9sum[2] fall_1sum[3]
fall_2sum[3] fall_3sum[3] fall_4sum[3] fall_5sum[3]
fall_6sum[3] fall_7sum[3] fall_8sum[3] fall_9sum[3]
fall_10sum[3] fall_11sum[3] fall_12sum[3] fall_13sum[3]
fall_1cout      fall_2cout      fall_3cout      fall_4cout
fall_5cout      fall_6cout      fall_7cout      fall_8cout
fall_9cout      fall_10cout     temper      alter#
-6.657e-04      1.198e-03      4.643e-10      1.175e-10
1.168e-10      1.174e-10      1.167e-10      1.167e-10
1.170e-10      1.182e-10      1.178e-10      1.189e-10
1.168e-10      1.173e-10      1.171e-10      1.169e-10
1.174e-10      1.175e-10      1.169e-10      1.170e-10
1.183e-10      1.179e-10      1.171e-10      1.175e-10
1.184e-10      1.174e-10      1.172e-10      1.177e-10
1.175e-10      1.185e-10      1.174e-10      1.167e-10
1.171e-10      1.177e-10      1.191e-10      1.173e-10
1.169e-10      1.177e-10      1.168e-10      1.228e-10
1.168e-10      1.174e-10      1.168e-10      1.168e-10
1.182e-10      1.181e-10      1.177e-10      1.168e-10
1.174e-10      1.171e-10      1.171e-10      1.175e-10
1.168e-10      1.167e-10      1.169e-10      1.168e-10
1.171e-10      1.172e-10      1.189e-10      1.171e-10
1.169e-10      1.177e-10      8.283e-11      8.118e-11
8.114e-11      8.241e-11      8.069e-11      8.260e-11
8.127e-11      8.200e-11      8.240e-11      8.181e-11
8.139e-11      8.268e-11      8.117e-11      8.267e-11
8.249e-11      8.264e-11      8.260e-11      8.141e-11
8.196e-11      8.275e-11      8.260e-11      8.250e-11
8.260e-11      8.260e-11      8.286e-11      8.115e-11
8.282e-11      8.067e-11      8.189e-11      8.206e-11
1.138e-11      8.229e-11      8.260e-11      8.096e-11
8.234e-11      8.172e-11      8.240e-11      8.069e-11
8.248e-11      8.131e-11      8.126e-11      8.119e-11
8.068e-11      8.287e-11      8.264e-11      8.253e-11
8.125e-11      8.130e-11      8.288e-11      8.300e-11
8.253e-11      8.226e-11      8.266e-11      8.136e-11
8.278e-11      8.081e-11      25.0000      1

```

Pre_sim tprop

```

.TITLE ***** lab03: 4-bit fa *****
iavg      pavg      tprop0      tprop1
tprop2      tprop3      tprop4      tprop5
tprop6      tprop7      tprop8      tprop9
tprop10      tprop11      tprop12      tprop13
tprop14      tprop15      tprop16      tprop17
tprop18      tprop19      tprop20      tprop21
tprop22      tprop23      tprop24      tprop25
tprop26      temper      alter#
-1.290e-03      2.322e-03      1.838e-10      7.761e-11
2.929e-10      7.717e-11      7.717e-11      1.051e-10
2.618e-10      1.077e-10      1.075e-10      8.685e-11
7.759e-11      1.597e-10      7.978e-11      4.894e-11
7.779e-11      2.593e-10      7.726e-11      4.142e-11
2.024e-10      8.532e-11      7.715e-11      5.196e-11
5.716e-11      1.470e-10      5.691e-11      4.909e-11
1.076e-10      25.0000      1

```

Post_sim tprop

```

.TITLE ***** lab03: 4-bit fa *****
iavg      pavg      tprop0      tprop1
tprop2      tprop3      tprop4      tprop5
tprop6      tprop7      tprop8      tprop9
tprop10      tprop11      tprop12      tprop13
tprop14      tprop15      tprop16      tprop17
tprop18      tprop19      tprop20      tprop21
tprop22      tprop23      tprop24      tprop25
tprop26      temper      alter#
-1.298e-03      2.336e-03      2.286e-10      8.730e-11
3.395e-10      8.922e-11      8.922e-11      1.409e-10
2.946e-10      1.331e-10      1.317e-10      1.097e-10
8.803e-11      2.044e-10      1.074e-10      6.214e-11
8.726e-11      3.017e-10      8.915e-11      4.994e-11
2.368e-10      1.200e-10      8.890e-11      6.311e-11
7.302e-11      1.899e-10      7.131e-11      6.197e-11
1.316e-10      25.0000      1

```

Post_sim rise/fall

```

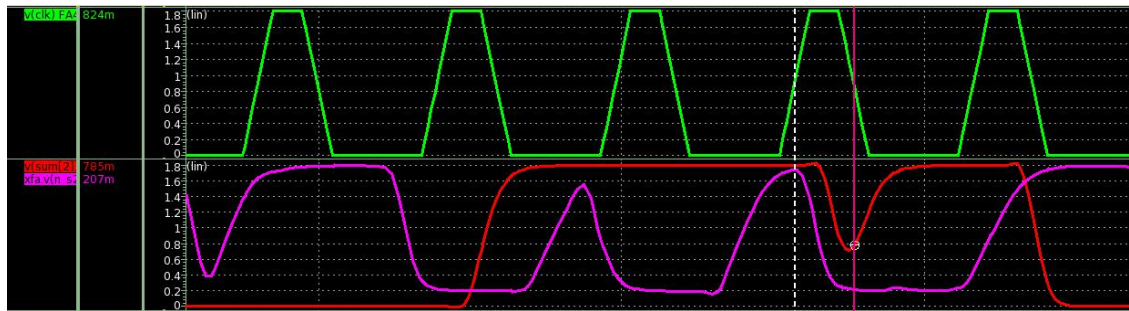
.TITLE ***** lab03: 4-bit fa *****
iavg      pavg      worst_prop      rise_1sum[0]
rise_2sum[0] rise_3sum[0] rise_4sum[0] rise_5sum[0]
rise_6sum[0] rise_7sum[0] rise_8sum[0] rise_9sum[0]
rise_10sum[0] rise_11sum[0] rise_12sum[0] rise_13sum[0]
rise_2sum[1] rise_3sum[1] rise_4sum[1] rise_5sum[1]
rise_6sum[1] rise_7sum[1] rise_8sum[1] rise_9sum[1]
rise_10sum[1] rise_11sum[1] rise_12sum[1] rise_13sum[1]
rise_1sum[2] rise_2sum[2] rise_3sum[2] rise_4sum[2]
rise_5sum[2] rise_6sum[2] rise_7sum[2] rise_8sum[2]
rise_9sum[2] rise_10sum[2] rise_11sum[2] rise_12sum[2]
rise_3sum[3] rise_4sum[3] rise_5sum[3] rise_6sum[3]
rise_7sum[3] rise_8sum[3] rise_9sum[3] rise_10sum[3]
rise_11sum[3] rise_12sum[3] rise_13sum[3] rise_1cout
rise_2cout      rise_3cout      rise_4cout      rise_5cout
rise_6cout      rise_7cout      rise_8cout      rise_9cout
rise_10cout     rise_11cout     fall_1sum[0] fall_2sum[0]
fall_3sum[0] fall_4sum[0] fall_5sum[0] fall_6sum[0]
fall_7sum[0] fall_8sum[0] fall_9sum[0] fall_10sum[0]
fall_11sum[0] fall_12sum[0] fall_1sum[1] fall_2sum[1]
fall_3sum[1] fall_4sum[1] fall_5sum[1] fall_6sum[1]
fall_7sum[1] fall_8sum[1] fall_9sum[1] fall_10sum[1]
fall_11sum[1] fall_12sum[1] fall_1sum[2] fall_2sum[2]
fall_3sum[2] fall_4sum[2] fall_5sum[2] fall_6sum[2]
fall_7sum[2] fall_8sum[2] fall_9sum[2] fall_1sum[3]
fall_2sum[3] fall_3sum[3] fall_4sum[3] fall_5sum[3]
fall_6sum[3] fall_7sum[3] fall_8sum[3] fall_9sum[3]
fall_10sum[3] fall_11sum[3] fall_12sum[3] fall_13sum[3]
fall_1cout      fall_2cout      fall_3cout      fall_4cout
fall_5cout      fall_6cout      fall_7cout      fall_8cout
fall_9cout      fall_10cout     temper      alter#
-6.668e-04      1.200e-03      5.379e-10      1.250e-10
1.239e-10      1.234e-10      1.234e-10      1.225e-10
1.232e-10      1.257e-10      1.229e-10      1.264e-10
1.255e-10      1.231e-10      1.224e-10      1.257e-10
1.248e-10      1.255e-10      1.236e-10      1.245e-10
1.242e-10      1.241e-10      1.261e-10      1.254e-10
1.272e-10      1.278e-10      1.255e-10      1.241e-10
1.257e-10      1.296e-10      1.240e-10      1.247e-10
1.247e-10      1.242e-10      1.302e-10      1.250e-10
1.250e-10      1.243e-10      1.234e-10      1.298e-10
1.267e-10      1.250e-10      1.242e-10      1.243e-10
1.269e-10      1.242e-10      1.254e-10      1.268e-10
1.279e-10      1.238e-10      1.242e-10      1.257e-10
1.235e-10      1.252e-10      1.237e-10      1.245e-10
1.247e-10      1.255e-10      1.275e-10      1.243e-10
1.239e-10      1.243e-10      8.702e-11      8.759e-11
8.743e-11      8.660e-11      8.765e-11      8.665e-11
8.604e-11      8.559e-11      8.742e-11      8.741e-11
8.717e-11      8.695e-11      8.815e-11      8.706e-11
8.809e-11      8.875e-11      8.730e-11      8.734e-11
8.750e-11      8.847e-11      8.695e-11      8.812e-11
8.782e-11      8.875e-11      8.770e-11      8.828e-11
8.792e-11      8.847e-11      8.703e-11      8.768e-11
8.687e-11      8.869e-11      8.784e-11      8.804e-11
8.736e-11      8.941e-11      8.782e-11      8.835e-11
8.708e-11      8.758e-11      8.850e-11      8.678e-11
8.826e-11      8.748e-11      8.745e-11      8.763e-11
8.726e-11      8.715e-11      8.674e-11      8.651e-11
8.703e-11      8.663e-11      8.867e-11      8.723e-11
8.696e-11      8.815e-11      25.0000      1

```


關於這次量測 propagation delay 的部分，理論上只需量測 critical path 的 propagation delay，且若將 clk period 調到過小 (i.e. Period = 0.51ns for presim / 0.58ns for postsim) 就可以看到是哪一組 input 產生 error，進而找到 worst propagation delay，不用每組 input、output 都量測。

以上還是有附上每一組 critical path tprop 的量測，而最後也驗證了以上猜想是正確的，**worst propagation delay 是在 CLK 調過快出現的第一個 err0 處。**

Glitch control：在 Post-Sim 的 waveform 上可以看到出現明顯的 glitch 如下圖。

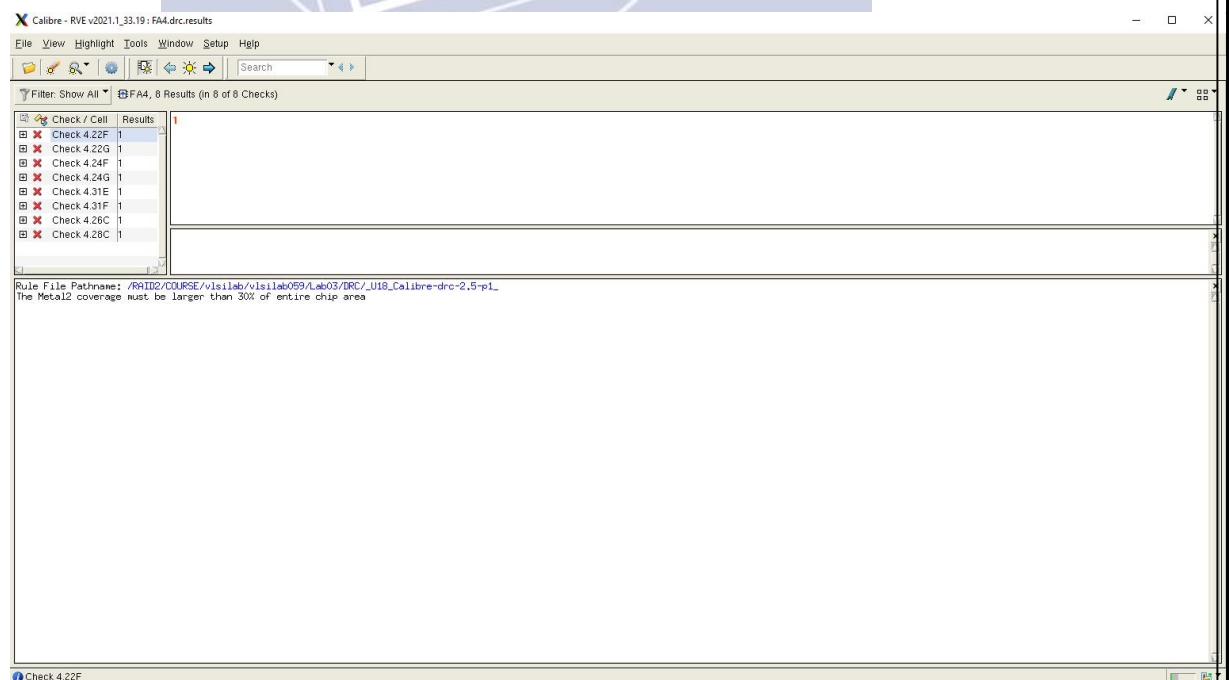


上圖紫色信號為 SUM[2]_bar，由於 glitch 剛好在 clk posedge 觸發的 threshold，因此紅色 output 被拉低，理論上如果架構為 DFF，此結果會使 output SUM[2] 在下一個 clk posedge 前都處於 logic 0 的狀態、導致 err0。但由於今天使用的是 high level trigger 的 D-Latch 架構，因此多了將近 195ps 的時間去使 output 拉回正確的數值，才出現了有 glitch 但沒有 error 的情況。

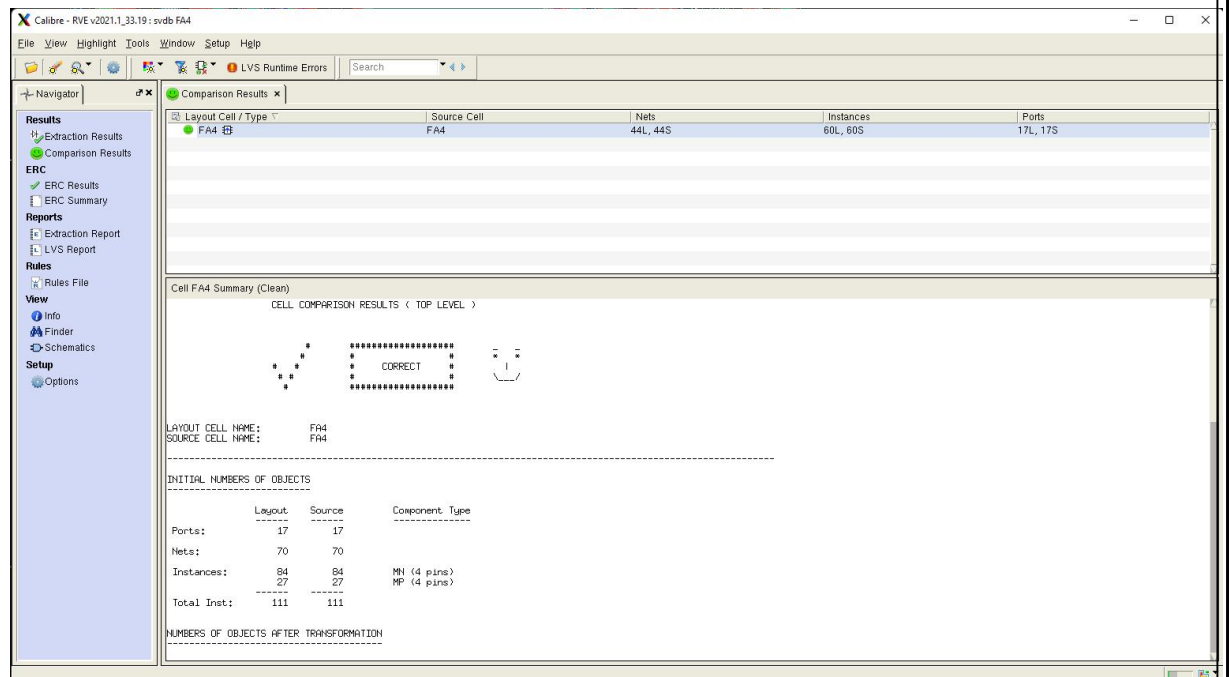
如果將 Post_Sim CLK period 調到 0.62ns，glitch 的情況就會消失。

III. Verification result

DRC



2. LVS



IV. Discussion

1. Compare Full Custom to Cell-based design and list their pros and cons.

Full-custom design: Logic cells、circuit 到最後 layout 全部由工程師設計。

Pros: 設計上有較高的彈性，可以達到最 optimal 的面積和 performance

Cons: 設計流程繁瑣、成本高，大規模晶片整合 time-to-market 不切實際

Cell-based design: 使用已經預先設計好的 component (stdcell) 作為 building block。而 layout 的拉線部分是 customize (大規模由 EDA 做 routing)。

Pros: 設計速度快，較符合 time-to-market。由於 stdcell 通常高度固定，很方便由 EDA tool 去做 place & route。因此對於大規模的晶片設計會使用 cell-based design + EDA。

Cons: 由於 stdcell 已固定，因此無法達到最優化的 performance 和面積。整合度不會像 full-custom 那麼高。

2. How to eliminate performance variation between Post-sim and Pre-sim.

Post-Sim 的 netlist 因為經過 layout LPE 後會將繞線的導線、poly 等寄生電容電阻值考慮進去，因此 layout 的一些準則如果達到整體 post_sim 的 performance 會離 pre-sim 小一些。以下列入幾點：

盡量多打 contact。diffusion、GND、VDD、metal 間銜接都需要 contact。而打一個 contact 相當於銜接處接上一個寄生電阻。因此打越多 contact 代表並聯的寄生電阻越多=>等效寄生電阻值會趨近於零，減少 IR drop。

避免 latch-up。P-substrate/N-substrate 電位要拉到 GND/VDD。因此有 VDD GND 拉線的部分，應盡可能加入 diffusion 去使 body 的準位固定。

能共用的 diffusion tap 應盡量共用，如此減少中間連線使用 metal 帶來的 delay。metal 拉線有過長、或多附載的情形應考慮加寬，可以降低訊號在傳遞上的 IR drop 太明顯。

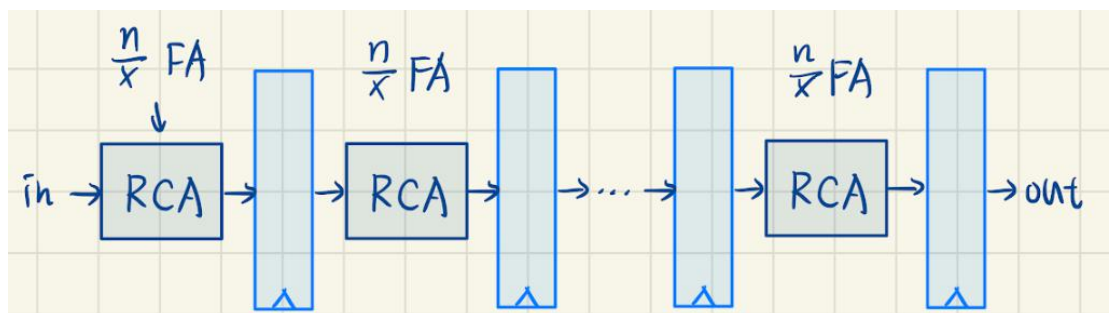
3. Please discuss what has influence on speed while pipeline regs are placed at different location.

如果考慮到要做出 32bit、64bit 以上的 adder 時，比起用非常龐大的 CLA 電路，切 pipeline 可能會更實際一些。

假設今天做一個 n-bit RCA，且做 x 級的 pipeline：

可預想 n-bit RCA 的 critical path 就是由 input 經過 n 個 FA 到最後一級的 output。最好的 pipeline 可以將 total Tdelay 縮小 x 倍。意即每經過 n/x 個 FA 後都在中間擋一層 pipeline reg。每一級的 pipeline 時間會幾乎相等，critical path 就被縮短了。(如下圖)

如果今天切 pipeline 十分不平均，有的大有的小，可以推測出 critical path 會被 pipeline 最大的那一級 latch 住，如此反而在切同樣級數的 pipeline 下得到較差的 performance。



4. Summary (Optional)

此次實驗的設計過程相較 Lab02 來說順利許多。基本上設計的 flow 都是先參閱很多不同的 paper、講義。在比較過不同電路的架構後，可以算出該電路會經過的 critical path、和大概經過幾級的 logic gate 等。初步判斷後選出認為最佳的幾個架構(不確定彼此間的 performance 情況下)利用方便的 HSpice 跑 presim 去驗證自己的想法和選擇最終的電路架構。

在 layout 的部分，因為面積的判定包含 N-Well，因此雖為 full-custom 的 design，我認為在畫 stdcell 的一些概念可以引進，也有助於優化面積。不同的 subckt 之間，應該要盡量使 PMOS network 對接，如此 N-Well 才可以圈在一起。再來是 GND、VDD 的共用、加寬。基本上每個 subckt 我都盡量使用 metal1 去繞線，這樣在銜接不同 subckt 的過程，才可以有很多空間給 metal2、metal3 去做繞線。

這次 lab 比較可惜的是，一開始採用 DFF (edge trigger) 的架構。但後來對 DFF、D-latch 的判定有些誤解，後來為了增加 CLK frequency 而不小心使用了 D-latch 的結構。我認為在電路的 design 上(尤其現在是數位電路)，不應該使用 D-latch。在追求 performance 的同時，我認為要怎麼做出一個很 robust 的電路更為重要。18 的製程下，如果今天我做出的 4bit FA 速度極快、但會有 glitch 的發生，那我不如用較慢的 CLK frequency，或是改用更 robust 的電路架構，如此一來我設計出來的 adder 還會更實用一些。

不過今天如果只是考慮製作一個單純的 4bit FA，沒有要拿來做成其他電路的 building block，單純以 full-custom 的角度來說，這樣 pseudo-NMOS 大 power 的問題、使用 latch 還是 DFF 就會比較不那麼重要。單看今天電路是扮演甚麼樣的一個角色吧。

總之，此次的 Lab 還是滿好玩的，尤其是最後看到自己猜想的架構能正常運作，還是很有成就感的！