

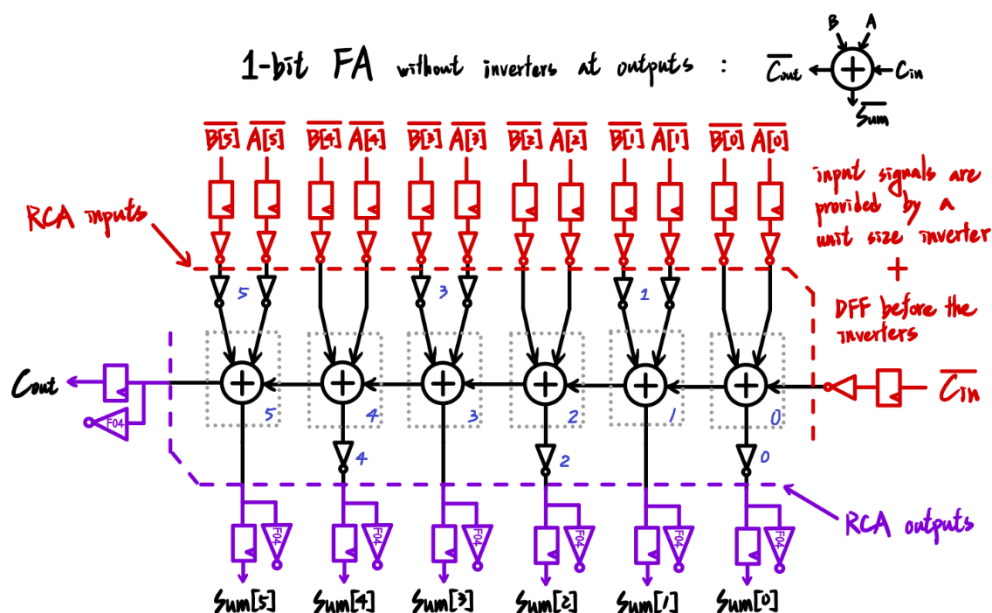
## Digital Integrated Circuits homework 4 電子所 陳柏翔 313510156

1. A **6-bit** one stage pipelining **ripple adder** as shown at Fig.4.1(a) is designed with **Fully Complementary Static Logic Gate** for the 1-bit FA as shown at Fig.4.1(c) and the D-register as shown at Fig.4.1(b). **Input signals are A[5:0], B[5:0] and Cin** which are provided by a unit size inverter. **Outputs are Cout@Sum [6:0]** with **loading of 4 unit size inverters (FO4) connected in parallelism**. (You shall provide SPICE simulation results of **timing** and **power** waveforms.)

- (1) Try your best to design the **fastest adder without pipelining registers**. First, show your **block diagrams** in terms of the **1-bit Full-Adder(FA)**. Second, show the **circuit schematic** of each block. Use **logic effort concepts** (you do not have to write down the procedure) **to design transistor widths** (in table form). Describe your design concept. (40%)

- Block diagram in terms of the 1-bit FA :

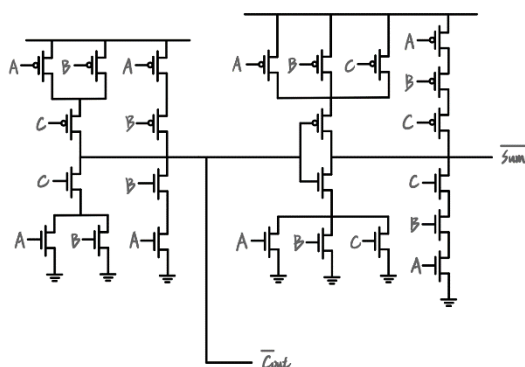
據題目要求，為了設計最快的 Ripple-Carry Adder，我採用與講義上 Fig. 10.12 相同的架構(如下圖 Block diagram)，其中每個 1-bit FA 在輸出端都不具有 Inverter，這樣就能在 Carry 的傳遞路徑上少 6 個 Inverters，因此有較短的 Delay (但相對的需要在某些輸入與輸出端加上 Inverters 以修正電路邏輯)。



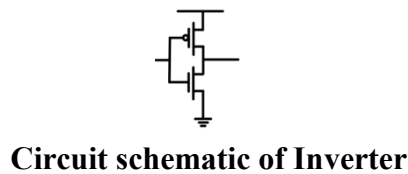
在整個 Ripple-Carry Adder 的輸入端與輸出端上，按照題目(以及助教在作業討論區所述)的要求，加上了 DFF 與 Inverters。此外，由於輸入端的要求是需要加上 Inverters，因此模擬時給到輸入端 DFF 的訊號是相反的。

- **Circuit schematic of each block :**

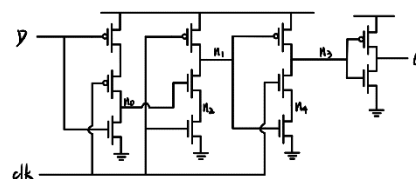
對於上圖Block diagram中的所有Block (即 $\oplus$ 號)，都代表著1-bit FA (without inverter at outputs)、以及Inverter與DFF (D-register)如下所示：



**Circuit schematic of 1-bit Full adder  
(without inverter at outputs)**

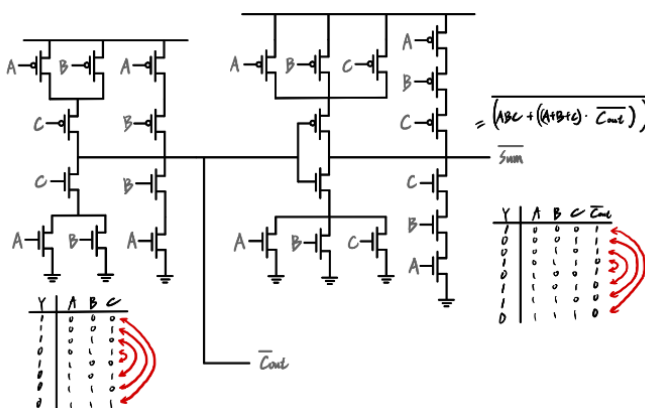


**Circuit schematic of Inverter**



**Circuit schematic of DFF**

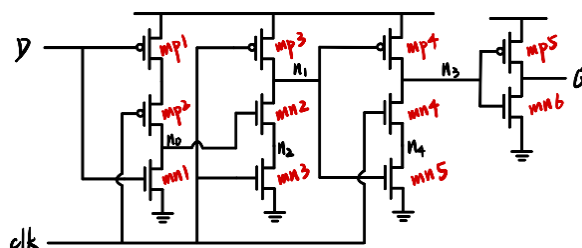
接著我重新從Transistor-level驗證電路邏輯的正確性，如附圖所示，可以發現當輸入訊號相反時，其對應的輸出結果正好也是相反的，因此在前面的Block diagram中，接在輸入端與輸出端的Inverters是交替出現在不同的FA上，而Carry的傳遞路徑上則不需要加上任何Inverters。



- **Design transistor widths :**

根據先前Homework 3中的實驗結果，NMOS與PMOS的比例應保持為1:1 才能讓一個Inverter的Logical Threshold更靠近  $(1/2)V_{DD}$  的位置，因此我將以此為基礎來決定出電路中個個電晶體的大小(寬度)。

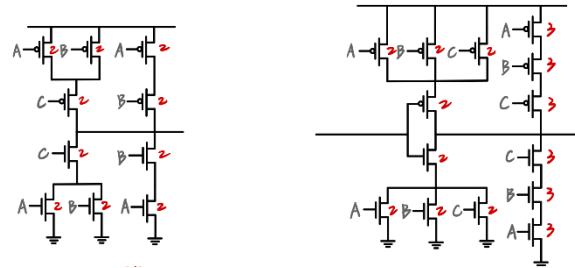
首先要先決定DFF電路中個電晶體大小，才能方便後續的計算順利進行，考量到電晶體的串並聯結構，以及輸入端所看到的電容希望能越小越好的情況下，再經過HSPICE測試微調mp4, mn4, mn5使  $T_{pcq0}$  與  $T_{pcq1}$  接近一致，所得電晶體大小結果如下 (DFF的模擬結果請見第(2)小題)：



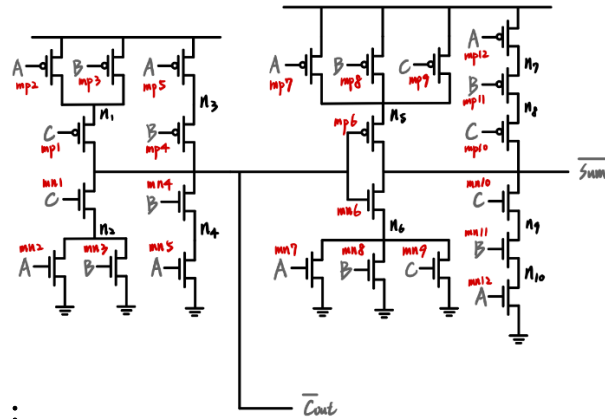
### Transistor widths of all DFFs :

MOS	mp1	mp2	mn1	mp3	mn2	mn3	mp4	mn4	mn5	mp5	mn6
nfin	1	2	1	1	2	2	2	1	1	1	1

接著將一個FA拆分為  $\overline{C_{out}}$  電路與  $\overline{Sum}$  電路兩個部分，右圖為拆分後的初步預設大小，後續將根據這樣的大小進行每一級的縮放。



為了讓電路有最快的速度，以Logical effort的概念來看，希望Critical Path (從FA0輸入端的Unit size inverter，到FA5的sum輸出端外所接的DFF+FO4電容)上的Delay能夠最小，使每一級的  $\hat{f}$  能夠盡量趨近於一致的大小。據題目所述，此部分不需要敘述計算過程，因此下方是我計算後的最終結果：



### Transistor widths of all FAs :

nfin	mp1-5	mn1-5	mp6-9	mn6-9	mp10-12	mn10-12
FA0	2	2	2	2	3	3
FA1	2	2	2	2	3	3
FA2	2	2	2	2	3	3
FA3	2	2	2	2	3	3
FA4	2	2	2	2	3	3
FA5	2	2	2	2	3	3

由於此計算結果顯示每個FA的大小皆設置相同寬度，直覺上不容易反應出此結果，因此以下簡述此結果的計算過程：

透過計算第二長的Critical path(由FA0  $C_{in}$  到FA5  $C_{out}$ )，假設每一級1-bit FA電路與下一級FA電路有  $x$  倍的大小縮放關係，經過的每一級  $\overline{C_{out}}$  電路所形成的  $G = 1 \times 2^6$ ,  $B = ((7x + 2)/2x)^5$ ,  $H = 5/1$ ，經過疊代求解  $F = GBH = \hat{f}^6 = (g_i h_i)^6$ ，可以反推算出當每一級的  $h_i = 4.5$ ，對應的解  $x = 1$ 。此結果表示不同級FA之間不需要進行比例縮放。

最後還要決定交替出現在FA輸入與輸出上的Inverters，考量到FA的大小與輸出端負載趨近於FO4，我決定將所有的Inverters都設為Unit size。

**Transistor widths of all Inverters (inside RCA) :**

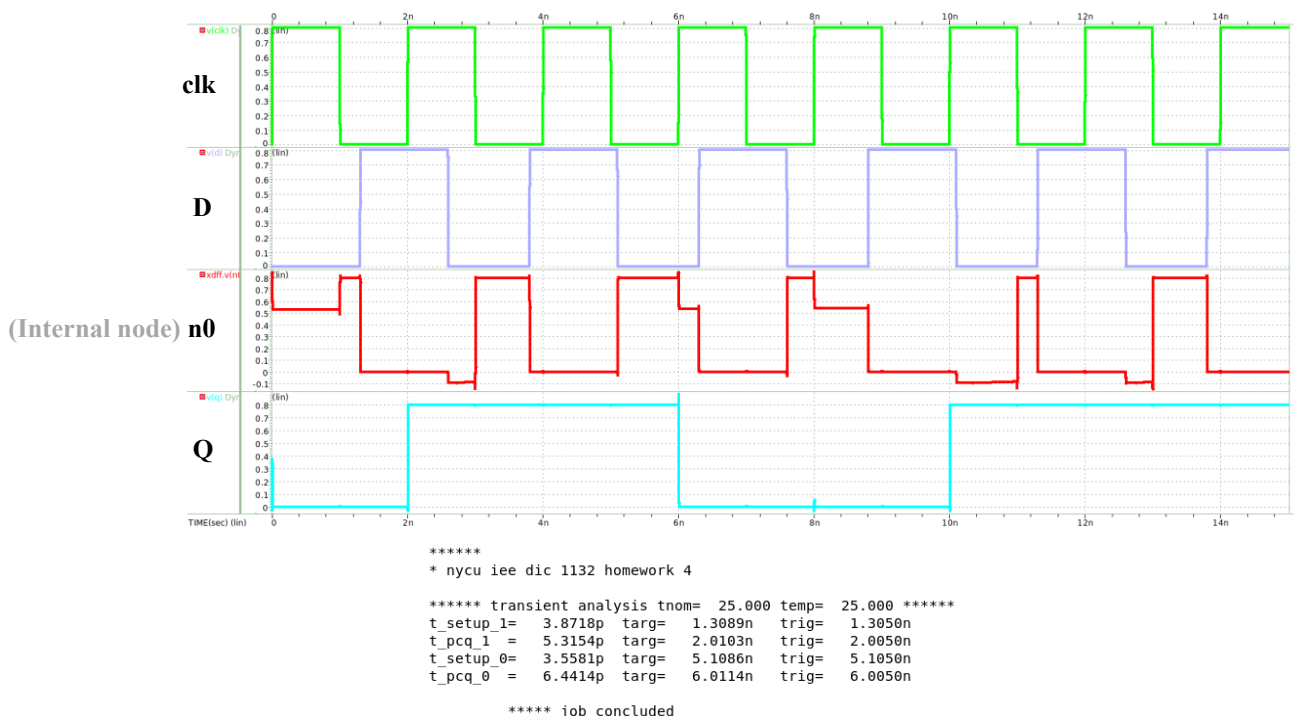
nfin	INV0	INV1a	INV1b	INV2	INV3a	INV3b	INV4	INV5a	INV5b
PMOS	1	1	1	1	1	1	1	1	1
MMOS	1	1	1	1	1	1	1	1	1

對於以上所有設置的Transistor widths，我還有再用HSPICE實驗數種其他設置方式(如每一級的FA寬度遞增)，但結果是此設置方法相較其他設置都來得更快，因此驗證了以上的計算與設置方式的正確性。

- (2) Based on the design of (1), run SPICE to find the **propagation delay time** (with pattern from **000000111110** to **000000111111** (A[5:0]@B[5:0]@Cin). Determine the **minimum clock cycle time** with the delay time estimated by SPICE. (20%)

• **Setup time & Clock-to-Q propagation delay of DFF :**

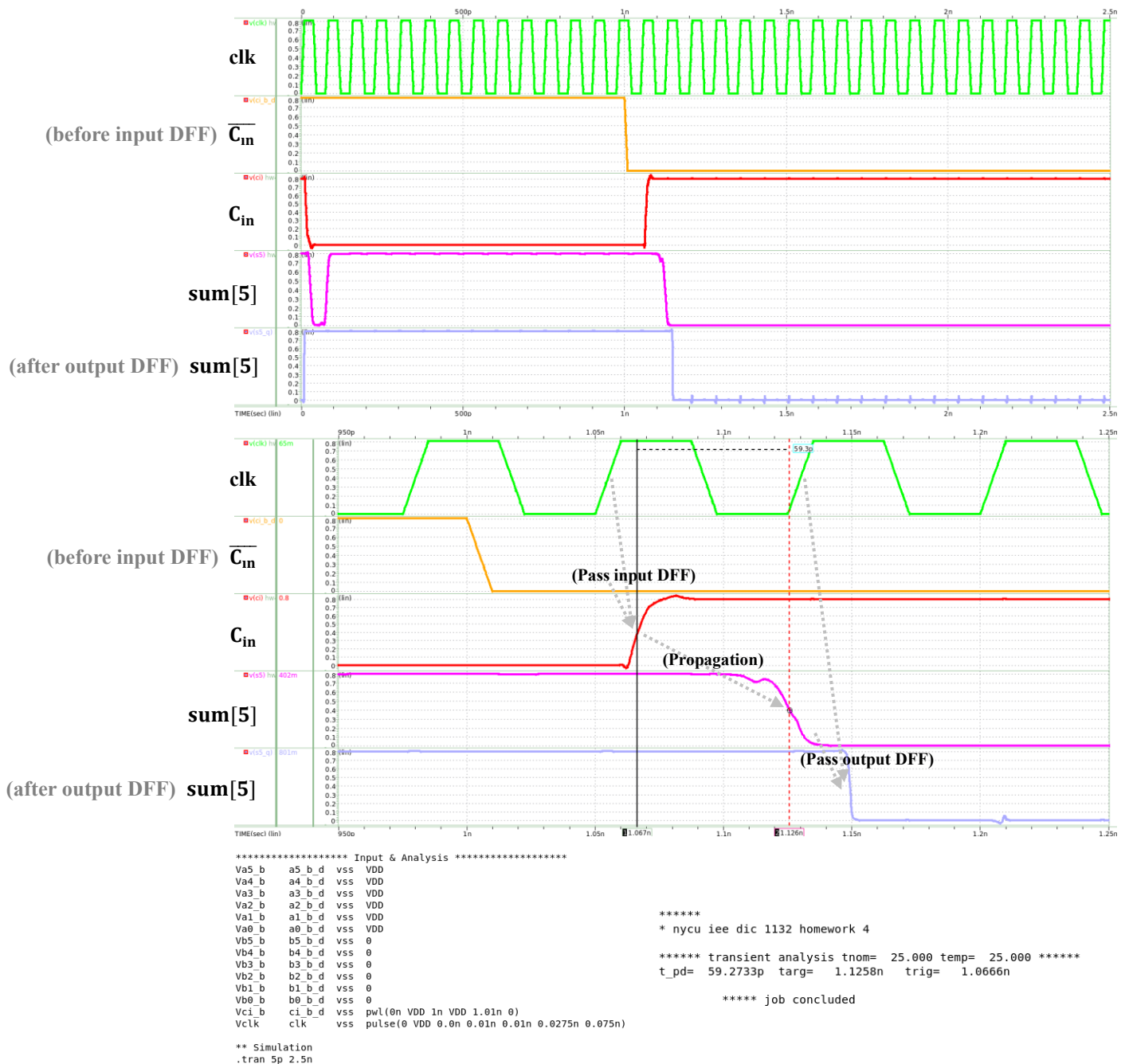
為了後續決定出最小的Cycle time，此處需要先測量DFF的時間資訊，詳細的各種時間分析在講義Ch6.5中可以找到。測量Setup time的方式是從輸入D至第一個儲存點n0所花費的時間，Clock-to-Q propagation delay則是從Clock訊號改變至Q改變所花費的時間。



	$t_{setup1}$	$t_{pcq1}$	$t_{setup0}$	$t_{pcq0}$
<b>Time</b>	3.87ps	5.31ps	3.56ps	6.44ps

- **Propagation delay time & Minimum clock cycle time :**

由於訊號太多，因此以下只截圖主要改變的訊號以及Critical path從輸入 $C_{in}$ 到FA5的sum輸出訊號：



Propagation delay time ( $C_{in}$ to $sum[5]$ )	Minimum clock cycle time
59.27ps	0.075ns (=75ps)

以上結果可以發現符合先前所學的公式： $t_{pd} \leq T_c - (t_{setup} + t_{pcq})$

帶入測量結果： $59.27ps \leq 75 - (3.56 + 6.44) = 65ps$

考量到輸入DFF與RCA的 $C_{in}$ 輸入端之間還有一個Inverter，因此還有約5ps的Delay造成Cycle time沒辦法低於75ps，如果將Cycle time設置得更小則會造成 $sum[5]$  (after output DFF)往後一個Cycle才改變為0。



- (3) Run SPICE to get the **average, peak** and **leakage power dissipation** and **energy/bit**, respectively of this adder with loading (FO4) when working at the **maximum working frequency**. (20%)

• **Input pattern:**

Average & Peak power	Leakage power
A[5:0] = 6'd0, 6'd1, ..., 6'd63	A[5:0] = 6'b000000
B[5:0] = 6'd63, 6'd62, ..., 6'd0	B[5:0] = 6'b111111
C <sub>in</sub> is 0/1 by turns	C <sub>in</sub> = 0, clk=0

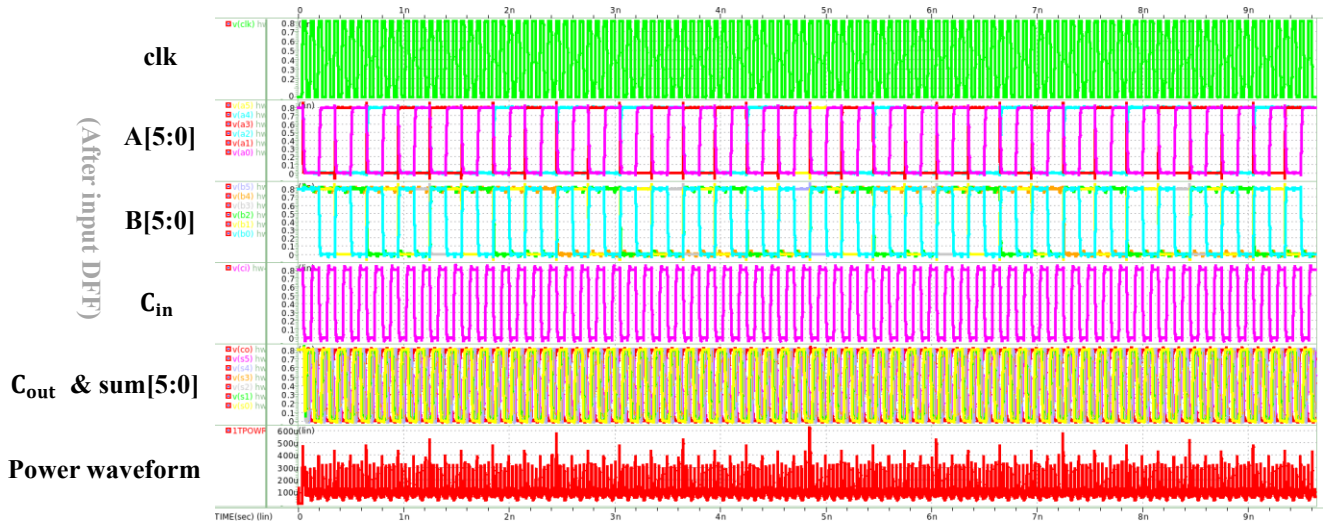
(Input pattern for average & peak power)

```
***** Input & Analysis *****
Va5_b a5_b_d vss pulse(0 VDD 0n 0.01n 0.01n 'CYCLE*64 - 0.01n' 'CYCLE*128')
Va4_b a4_b_d vss pulse(0 VDD 0n 0.01n 0.01n 'CYCLE*32 - 0.01n' 'CYCLE*64')
Va3_b a3_b_d vss pulse(0 VDD 0n 0.01n 0.01n 'CYCLE*16 - 0.01n' 'CYCLE*32')
Va2_b a2_b_d vss pulse(0 VDD 0n 0.01n 0.01n 'CYCLE*8 - 0.01n' 'CYCLE*16')
Va1_b a1_b_d vss pulse(0 VDD 0n 0.01n 0.01n 'CYCLE*4 - 0.01n' 'CYCLE*8')
Va0_b a0_b_d vss pulse(0 VDD 0n 0.01n 0.01n 'CYCLE*2 - 0.01n' 'CYCLE*4')
Vb5_b b5_b_d vss pulse(0 VDD 'CYCLE*64' 0.01n 0.01n 'CYCLE*64 - 0.01n' 'CYCLE*128')
Vb4_b b4_b_d vss pulse(0 VDD 'CYCLE*32' 0.01n 0.01n 'CYCLE*32 - 0.01n' 'CYCLE*64')
Vb3_b b3_b_d vss pulse(0 VDD 'CYCLE*16' 0.01n 0.01n 'CYCLE*16 - 0.01n' 'CYCLE*32')
Vb2_b b2_b_d vss pulse(0 VDD 'CYCLE*8' 0.01n 0.01n 'CYCLE*8 - 0.01n' 'CYCLE*16')
Vb1_b b1_b_d vss pulse(0 VDD 'CYCLE*4' 0.01n 0.01n 'CYCLE*4 - 0.01n' 'CYCLE*8')
Vb0_b b0_b_d vss pulse(0 VDD 'CYCLE*2' 0.01n 0.01n 'CYCLE*2 - 0.01n' 'CYCLE*4')
Vcl_b cl_b_d vss pulse(0 VDD 0n 0.01n 0.01n 'CYCLE*1 - 0.01n' 'CYCLE*2')
Vclk clk vss pulse(0 VDD 'CYCLE/2' 0.01n 0.01n 'CYCLE/2 - 0.01n' 'CYCLE')

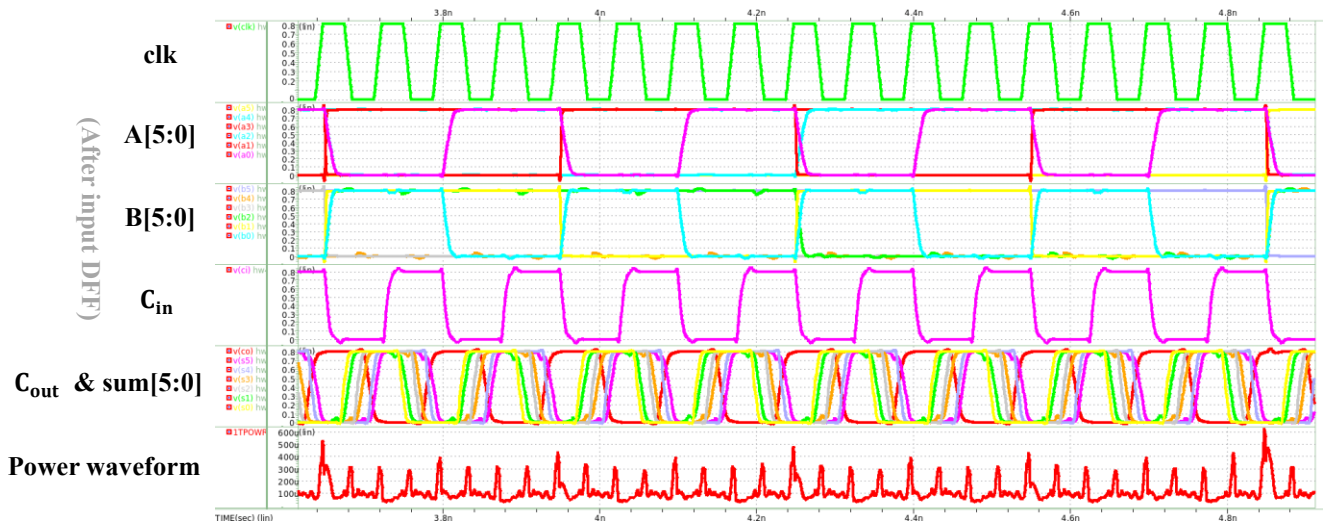
** Simulation
.tran 5p 'CYCLE*128 + CYCLE/2'

** Power measurement
.meas tran Avg power AVG POWER
.meas tran Peak power MAX POWER
```

• **Average & Peak power – Timing & Power waveforms:**



• **Average & Peak power - Timing & Power waveforms (Zoomed in):**



Simulation Time (Total 128.5 cycles)

$$= 0.075ns \times 128.5 = 9.6375ns$$

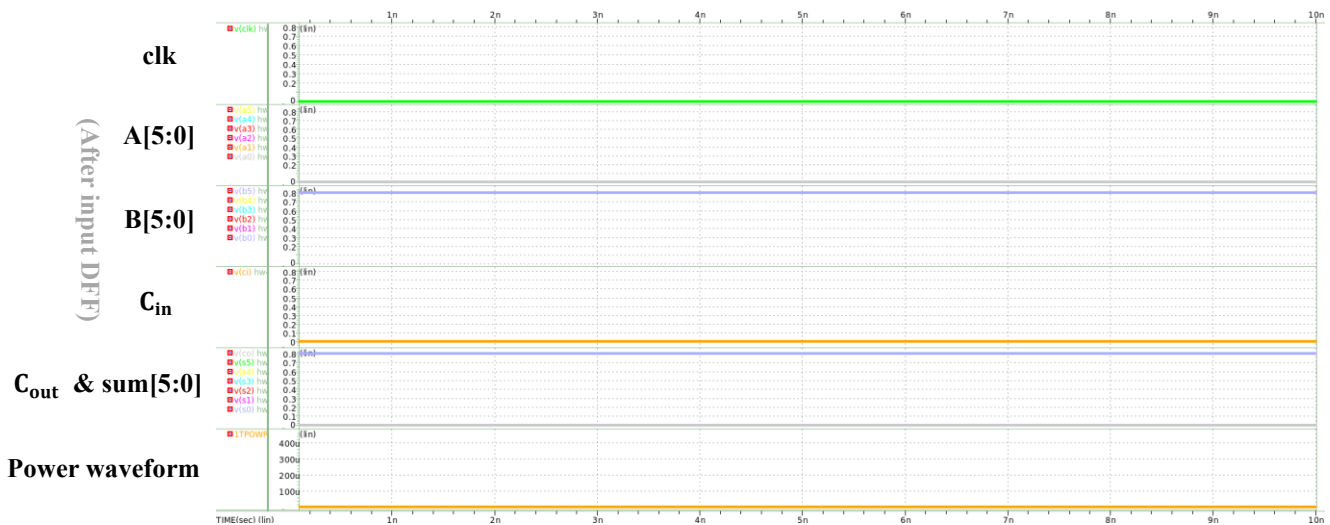
(多0.5 cycle是為了錯開clk posedge)

```
*****
* nycu iee dic 1132 homework 4

***** transient analysis tnom= 25.000 temp= 25.000 *****
avg_power = 115.3683u from= 0. to= 9.6375n
peak_power= 618.3058u at= 4.8472n from= 0. to= 9.6375n

***** job concluded
```

- **Leakage Power – Timing & Power waveforms:**



Simulation Time = 10ns

(模擬時由於鎖定clk=0會無法傳遞A,B與C<sub>in</sub>訊號，因此我將clk設定為跳動一次後再定為0，並且Power從訊號穩定的0.1ns後才做測量，經過這個過程才能正確測得Input pattern所對應的結果。)

- **Power analysis of RCA with maximum working frequency:**

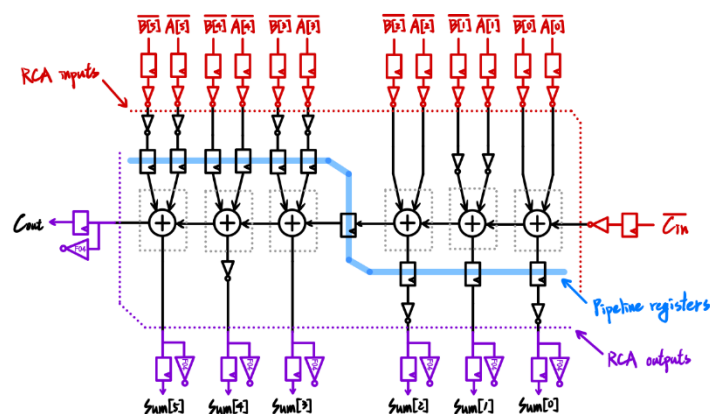
計算 Energy/Bit : (Average power × Simulation time) / Bits

$$= 115.3683 \times 10^{-6} \times 9.6375 \times 10^{-9} / 6 = 1.8531 \times 10^{-13} = 185.31 \text{ fJ}$$

Average power	Peak power	Leakage power	Energy/Bit
115.37μW	618.31μW	28.09nW	185.31fJ

(4) **Add one pipelining stage** using the designed D register (Fig.4.1(b)) into the 6-bit ripple adder as shown at Fig.4.1(a). Run SPICE to find the the **propagation delay time** (with pattern from **0000001111110** to **0000001111111** (A[5:0]@B[5:0]@C<sub>in</sub>) between pipelining stages to determine the **maximum working frequency** of the clock with the delay time estimated by SPICE.

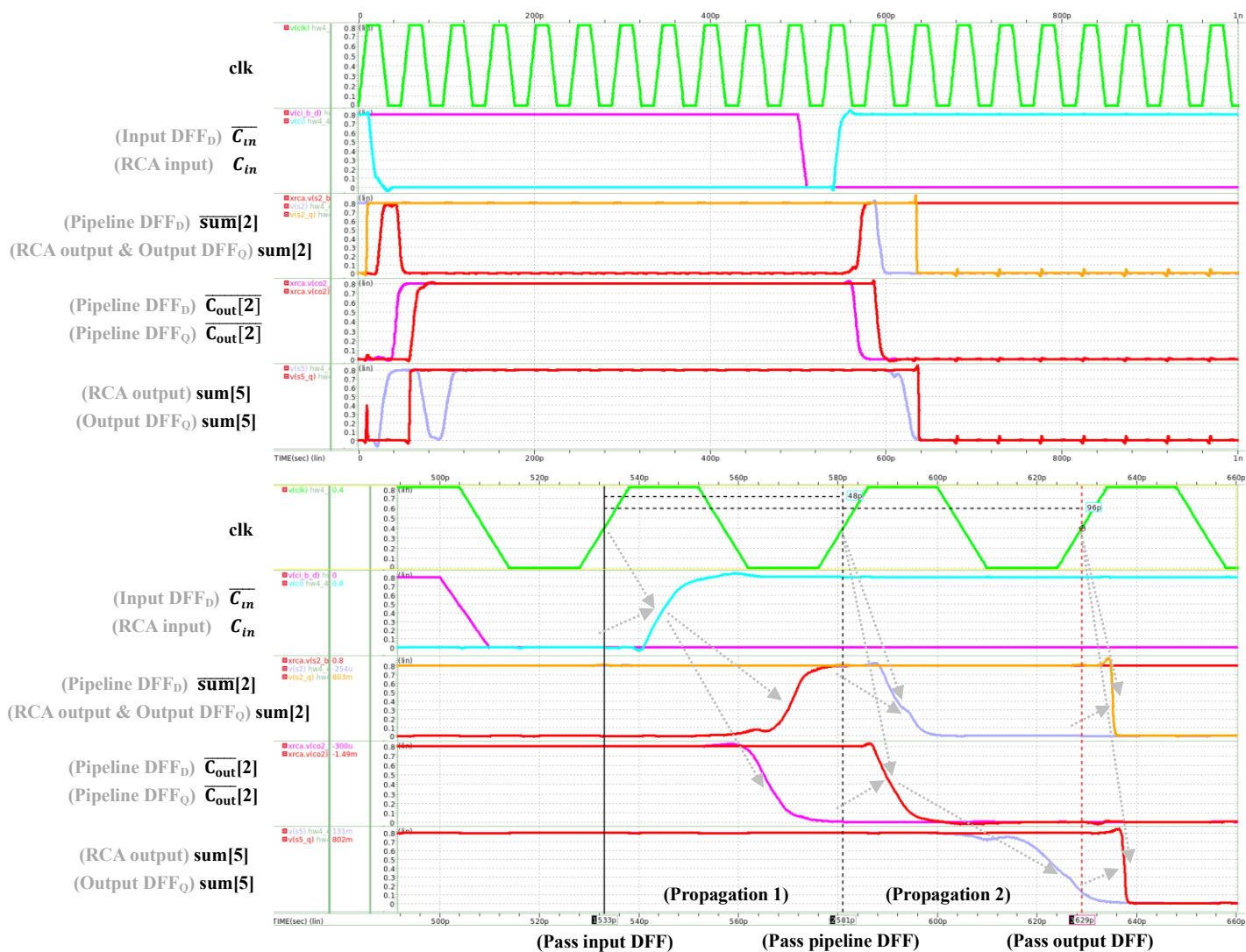
- **Pipelined (2-stage) RCA block diagram:**



經過Pipelined後的設計，輸出會再延遲一個Cycle才給出原本的計算結果，如上一頁附圖所示，我在6個FAs的中間(FA2, 3之間)加上了Pipeline register，由於考量到資料傳遞的正確性，我也在FA0, FA1, FA2的輸出以及FA3, FA4, FA5的輸入端加上了Pipeline registers。

因此原本最長的Critical path從 $C_{in}$ 到sum[5]變成了兩段不同Stage的Critical paths：Stage 1<sup>st</sup>的是從 $C_{in}$ 到sum[2]，而Stage 2<sup>nd</sup>的則是從中間Pipeline register輸出的 $\overline{C_{out}}[2]$ 到sum[5]。

以下為模擬後的波形圖以及Propagation delay time測量結果：



同(2)小題方法，由公式估算 $T_c$ 後我不斷調小Cycle time，最終調到Clock cycle time = 48ps。

```
*****
* nycu iee dic 1132 homework 4
***** transient analysis tnom= 25.000 temp= 25.000 *****
t_pd_stg1= 26.2748p targ= 570.8069p trig= 544.5322p
t_pd_stg2= 33.3675p targ= 623.9904p trig= 590.6228p
***** job concluded
```

Propagation delay time ( $C_{in}[0]$ to sum[2])	Propagation delay time ( $\overline{C_{in}}[3]$ to sum[5])	Maximum working frequency
26.27	33.37ps	20.83GHz (=1/48ps)