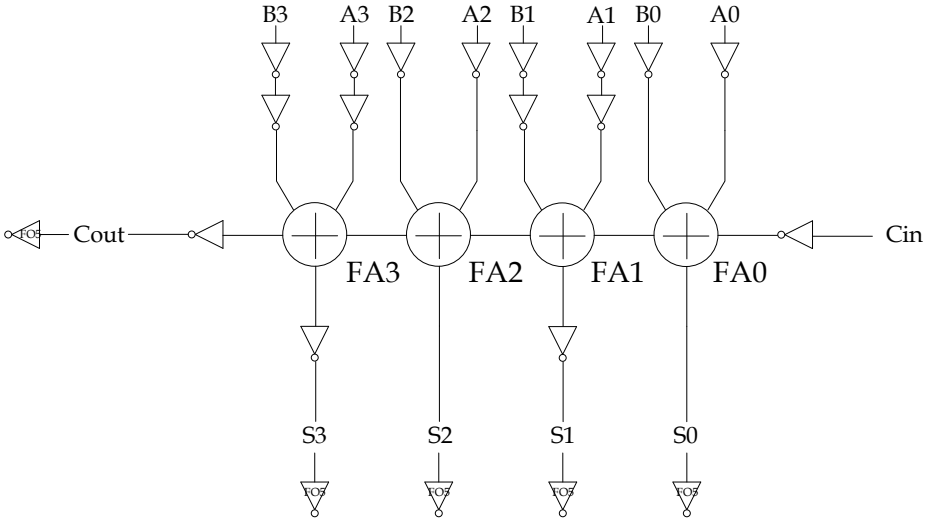
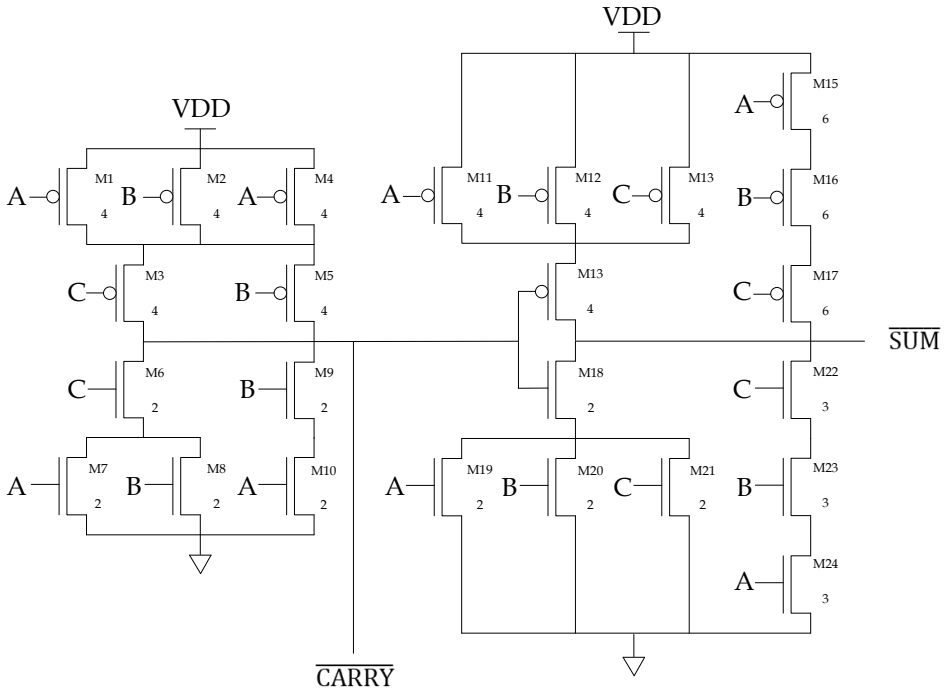
A 4-bit ripple carry adder shown in Fig. 1 is designed with fully complementary static logic gate of full adder (FA). Input signals are A[3:0], B[3:0], and C_{in}. Outputs are SUM[3:0] and C_{out}.

(1) Try to design the fastest adder when each input signal is driven by one unit size inverter. For the output loads, 5 unit size inverters are used for each output signal. First, show your block diagrams in terms of the 1-bit FA. Second, show the circuit schematic of the FA. Use logic effort concepts (do not have to write down the procedure) to design transistor widths. Describe your design concept. (40%)

(a) The block diagram in terms of 1-bit FA:



(b) The circuit schematic of my FA:



我把每個 mos 的 nfin 都標在上面了。

(c) Design concept:

- 利用 FA 的 inversion property 省去 FA 與 FA 之間的 inverter 以減少 critical path 上的 delay。
- 晚到的訊號 (即 C) 要放靠近輸出的地方，可以減少 delay。
- 並聯的部分要放靠近 VDD 與 GND，以減少寄生電容對 output 訊號的影響。
- 我設定 PMOS 和 NMOS 的 size 比例為 2 : 1。
- RCA 的 critical path 就是由每個 FA 的 carry out 串接起來再加上 FA3 的 sum。
  - FA 左半部用來產生 carry out 的 mos 要調大，這樣才推得動下一個 FA。
  - FA 右半部用來產生 sum 的 mos 要調小，讓 carry out 的 loading 小一點。
  - RCA 最尾端的 carry out 後面又接了 FO5。也就是說整個 critical path 的 effort delay 是 5。
  - 所以我 size 的設計如下:
    - FA0 設為 unit size 的 1 倍。
    - FA1 設為 unit size 的 2 倍。
    - FA2 設為 unit size 的 3 倍。
    - FA3 設為 unit size 的 4 倍。
    - 以 FA3 為例，下圖是 FA3 的 code:

```
.subckt FA3 A B C carry_bar sum_bar
* begin of carry
    M1          n1          A vdd! vdd! pmos_rvt nfin = '4 * 4'
    M2          n1          B vdd! vdd! pmos_rvt nfin = '4 * 4'
    M3   carry_bar          C   n1 vdd! pmos_rvt nfin = '4 * 4'
    M4          n2          A vdd! vdd! pmos_rvt nfin = '4 * 4'
    M5   carry_bar          B   n2 vdd! pmos_rvt nfin = '4 * 4'
    M6   carry_bar          C   n3 gnd! nmos_rvt nfin = '2 * 4'
    M7          n3          A gnd! gnd! nmos_rvt nfin = '2 * 4'
    M8          n3          B gnd! gnd! nmos_rvt nfin = '2 * 4'
    M9   carry_bar          B   n4 gnd! nmos_rvt nfin = '2 * 4'
    M10         n4          A gnd! gnd! nmos_rvt nfin = '2 * 4'
* end of carry
* begin of sum
    M11         n5          A vdd! vdd! pmos_rvt nfin = 4
    M12         n5          B vdd! vdd! pmos_rvt nfin = 4
    M13         n5          C vdd! vdd! pmos_rvt nfin = 4
    M14   sum_bar carry_bar   n5 vdd! pmos_rvt nfin = 4
    M15         n6          A vdd! vdd! pmos_rvt nfin = 6
    M16         n7          B   n6 vdd! pmos_rvt nfin = 6
    M17   sum_bar           C   n7 vdd! pmos_rvt nfin = 6
    M18   sum_bar carry_bar   n8 gnd! nmos_rvt nfin = 2
    M19         n8          A gnd! gnd! nmos_rvt nfin = 2
    M20         n8          B gnd! gnd! nmos_rvt nfin = 2
    M21         n8          C gnd! gnd! nmos_rvt nfin = 2
    M22   sum_bar           C   n9 gnd! nmos_rvt nfin = 3
    M23         n9          B  n10 gnd! nmos_rvt nfin = 3
    M24        n10          A gnd! gnd! nmos_rvt nfin = 3
* end of sum
.ends
```

  - 然後 FA3 去推 FO5 (5 倍的 unit size inverter)
  - 這樣做的好處是我每一級負責的推力較為平均。

(d) Size for each mos:

Unit size inverter

MOS Type	**size**
PMOS	2
NMOS	1

FA0:

Index of MOS	**size**
M1	$4 \times 1 = 4$
M2	$4 \times 1 = 4$
M3	$4 \times 1 = 4$
M4	$4 \times 1 = 4$
M5	$4 \times 1 = 4$
M6	$2 \times 1 = 2$
M7	$2 \times 1 = 2$
M8	$2 \times 1 = 2$
M9	$2 \times 1 = 2$
M10	$2 \times 1 = 2$
M11	4
M12	4
M13	4
M14	4
M15	6
M16	6
M17	6
M18	2
M19	2
M20	2
M21	2
M22	3
M23	3
M24	3

FA1:

Index of MOS	**size**
M1	$4 \times 2 = 8$
M2	$4 \times 2 = 8$
M3	$4 \times 2 = 8$
M4	$4 \times 2 = 8$
M5	$4 \times 2 = 8$
M6	$2 \times 2 = 4$
M7	$2 \times 2 = 4$
M8	$2 \times 2 = 4$
M9	$2 \times 2 = 4$
M10	$2 \times 2 = 4$
M11	4
M12	4
M13	4
M14	4
M15	6
M16	6
M17	6
M18	2
M19	2
M20	2
M21	2
M22	3
M23	3
M24	3

FA2:

Index of MOS	**size**
M1	$4 \times 3 = 12$
M2	$4 \times 3 = 12$
M3	$4 \times 3 = 12$
M4	$4 \times 3 = 12$
M5	$4 \times 3 = 12$
M6	$2 \times 3 = 6$
M7	$2 \times 3 = 6$
M8	$2 \times 3 = 6$
M9	$2 \times 3 = 6$
M10	$2 \times 3 = 6$
M11	4
M12	4
M13	4
M14	4
M15	6
M16	6
M17	6
M18	2
M19	2
M20	2
M21	2
M22	3
M23	3
M24	3

FA3:

Index of MOS	**size**
M1	$4 \times 4 = 16$
M2	$4 \times 4 = 16$
M3	$4 \times 4 = 16$
M4	$4 \times 4 = 16$
M5	$4 \times 4 = 16$
M6	$2 \times 4 = 8$
M7	$2 \times 4 = 8$
M8	$2 \times 4 = 8$
M9	$2 \times 4 = 8$
M10	$2 \times 4 = 8$
M11	4
M12	4
M13	4
M14	4
M15	6
M16	6
M17	6
M18	2
M19	2
M20	2
M21	2
M22	3
M23	3
M24	3

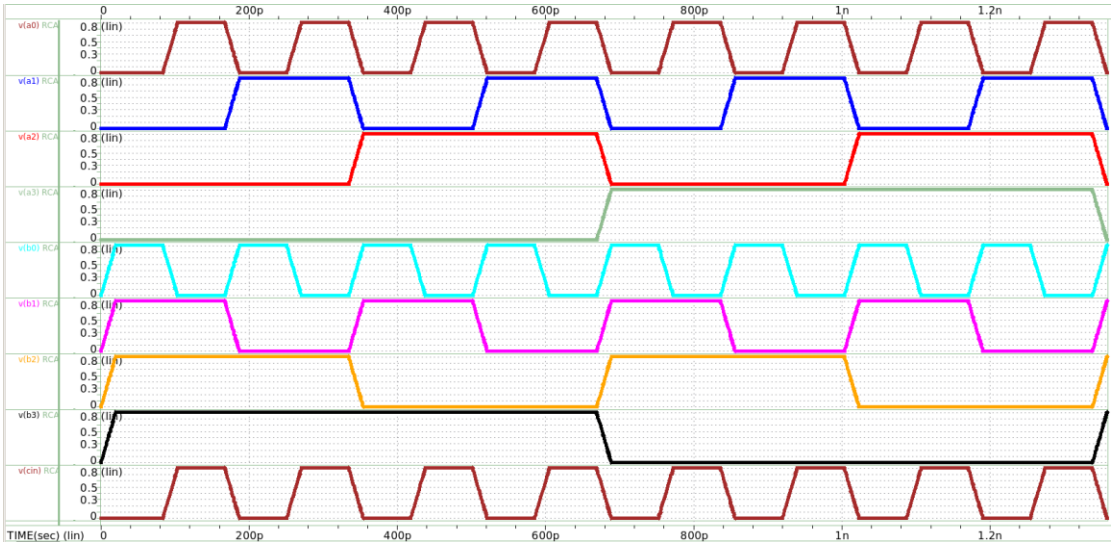
(2) Based on the design of (1), run SPICE to find the propagation delay (As shown in Fig. 2, with pattern from IA [3:0] = 4'b1111, IB[3:0] = 4'b0000, IC_{in} = 1'b1 to IA[3:0] = 4'b1111, IB[3:0] = 4'b0000, IC_{in} = 1'b0). Determine the maximum propagation delay of the Ripple Carry Adder (exclude the load inverters). (20%)

Here's the waveform of C_{in} and S3:

Therefore, the maximum propagation delay = 63.6ps.

(3) Run SPICE of the ripple carry adder with each output load equals to 4 unit size inverter. Determine the average, peak, and leakage power dissipation and energy per bit, respectively when simulating at the working frequency in (2). (20%)

(a) Here's my pattern of input:



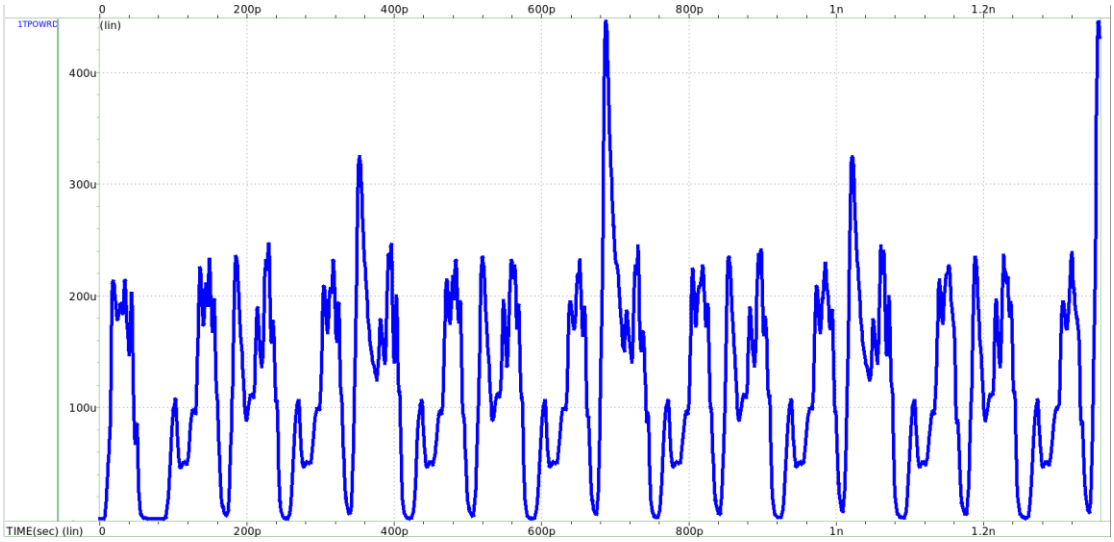
Note that the period of single pattern is 63.6ps.

Here's the simulation result of average power and peak power:

```
****** transient analysis tnom=  25.000 temp=  25.000 ******
avgpwr= 115.8002u  from=   0.            to=   1.3576n
peakpwr= 445.6248u    at= 687.3335p
            from=   0.            to=   1.3576n
```

Therefore, average power = 115.8002uW and peak power = 445.6248uW.

(b) Here's the waveform of average power and peak power:



(c) Here's my code measuring leakage power:

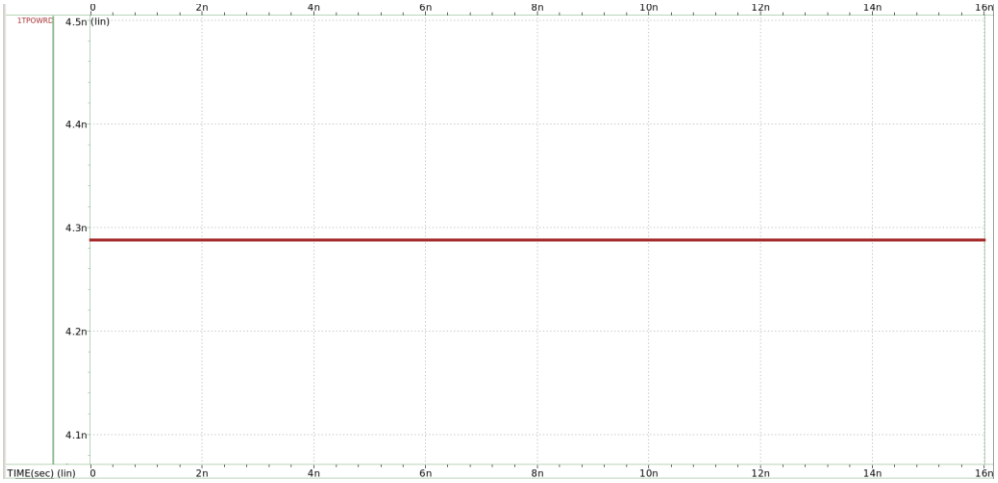
```
.trans 1ps 16ns
V0 A0 gnd! 0v
V1 A1 gnd! 0v
V2 A2 gnd! 0v
V3 A3 gnd! 0v
V4 B0 gnd! xvdd
V5 B1 gnd! xvdd
V6 B2 gnd! xvdd
V7 B3 gnd! xvdd
V8 Cin gnd! 0v

.meas tran LeakagePower AVG power from = 0ns to = 16ns
```

(d) Here's the simulation result of leakage power:

```
****** transient analysis tnom=  25.000 temp=  25.000 ******
leakagepower=   4.2875n  from=   0.            to=  16.0000n
```

Therefore, leakage power = 4.2875nW

(e) Here's the waveform of the leakage power:



(f) Energy/bit:
Energy/bit = (avg_pwr * sim_time) / bit = (115.8002uW * 1.3576ns) / 9 = 17.4678fJ

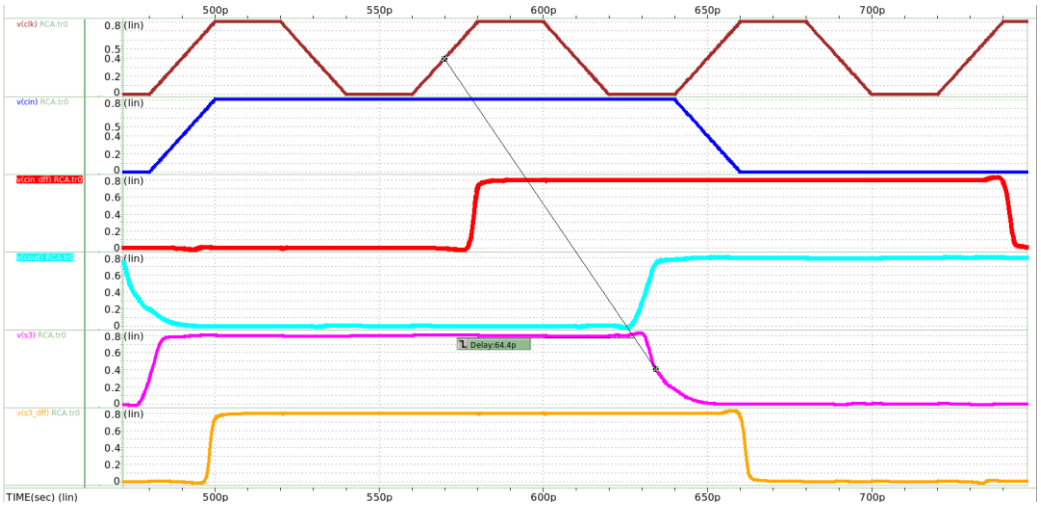
(g) Average power, peak power, leakage power, and energy/bit in table form:

Average power	115.8002uW
Peak power	445.6248uW
Leakage power	4.2875nW
Energy/bit	17.4678fJ

(4) Add pipelining stages as shown in Fig. 4 into the 4-bit ripple carry adder, with the D register given in Fig. 3. Run SPICE to find the propagation delay time (with pattern from IA [3:0] = 4'b0000, IB[3:0] = 4'b1111, IC_{in} = 1'b0 to IA[3:0] = 4'b0000, IB[3:0] = 4'b1111, IC_{in} = 1'b1) between pipelining stages to determine the maximum working frequency of the clock.

(a) Critical path is clk -> cin_dff -> s3.

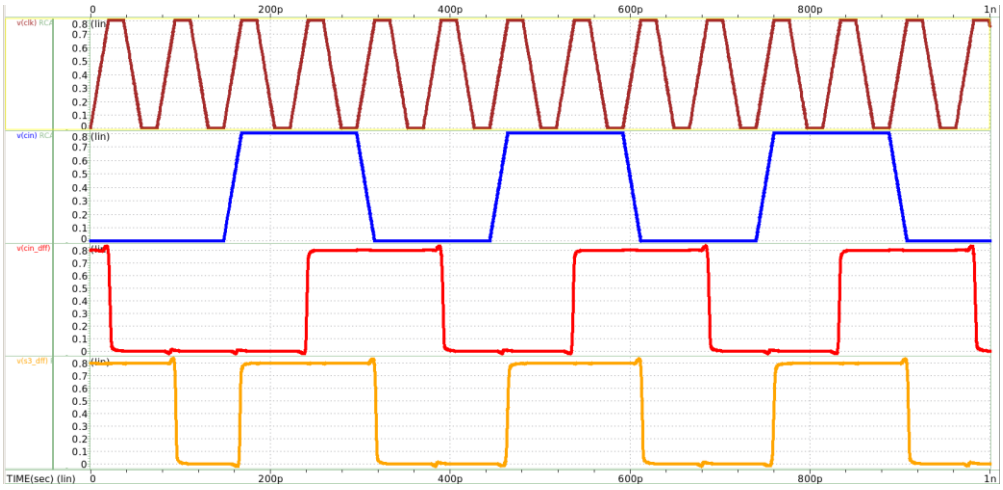
(b) Here's the waveform including clk, cin, cin_dff, s3 and s3_dff:



Therefore, the max propagation delay = 64.4ps

(c) 因為 propagation delay = 64.4ps, 所以 clk period 至少要大於 64.4 ps。在我反覆測試了幾次後，最終測出了 clk period 是 74ps 時 RCA 的功能會正確。.

下圖是 clk period = 74ps 時的波型圖，可以看到 RCA 的功能是正確的。



(d) Maximum working frequency of the clock:

$$f = \frac{1}{T} = \frac{1}{74\text{ps}} = 13.513\text{GHz}$$

Therefore, the maximum working frequency of the clock = 13.513GHz