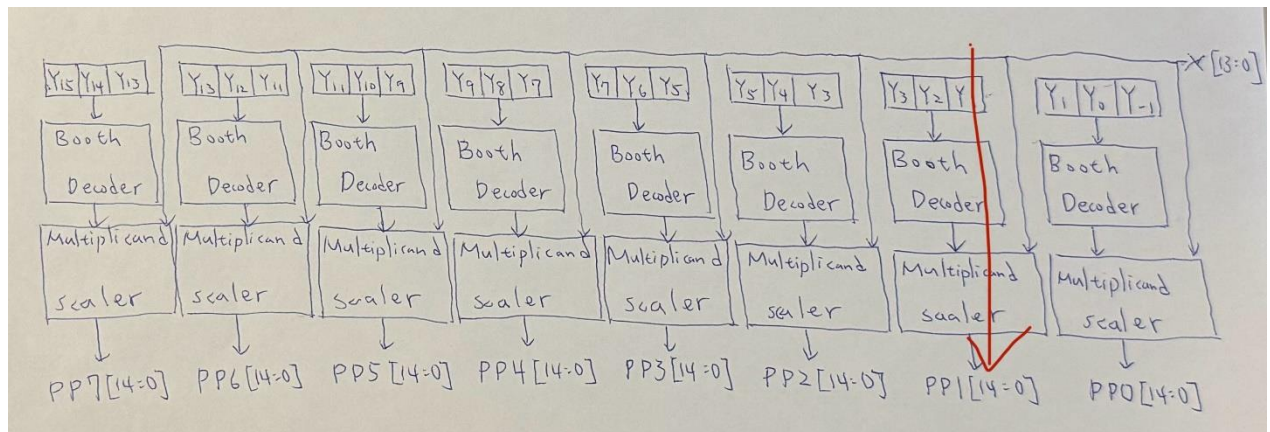


312510239 王則惟

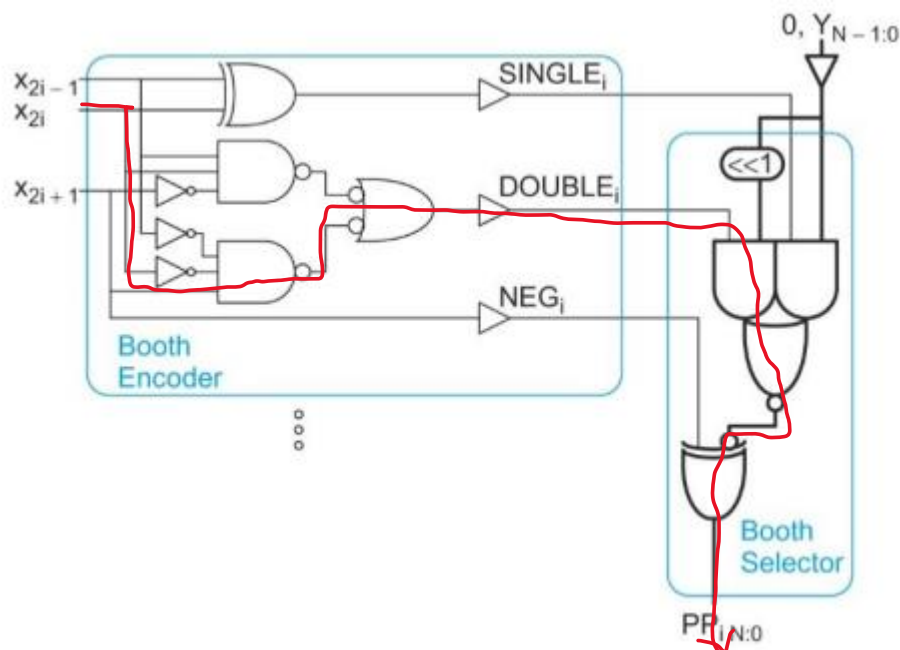
Design a 14-bit unsigned multiplier $P=X \cdot Y$ by using modified Radix-4 Booth multipliers. The goal is to have minimum critical path delay time with the least gate count.

- (1) Show your block diagram as shown in Fig.1 below (example for the case of 8-bit; Booth decoder and Multiplicand scale corresponding to Booth Encoder and Booth selector of Fig.10.80). For each block, you shall show its logic design diagram. Indicate the critical path. Explain your design concepts (30).

下圖是電路的 block diagram，其中紅色路徑是 critical path。



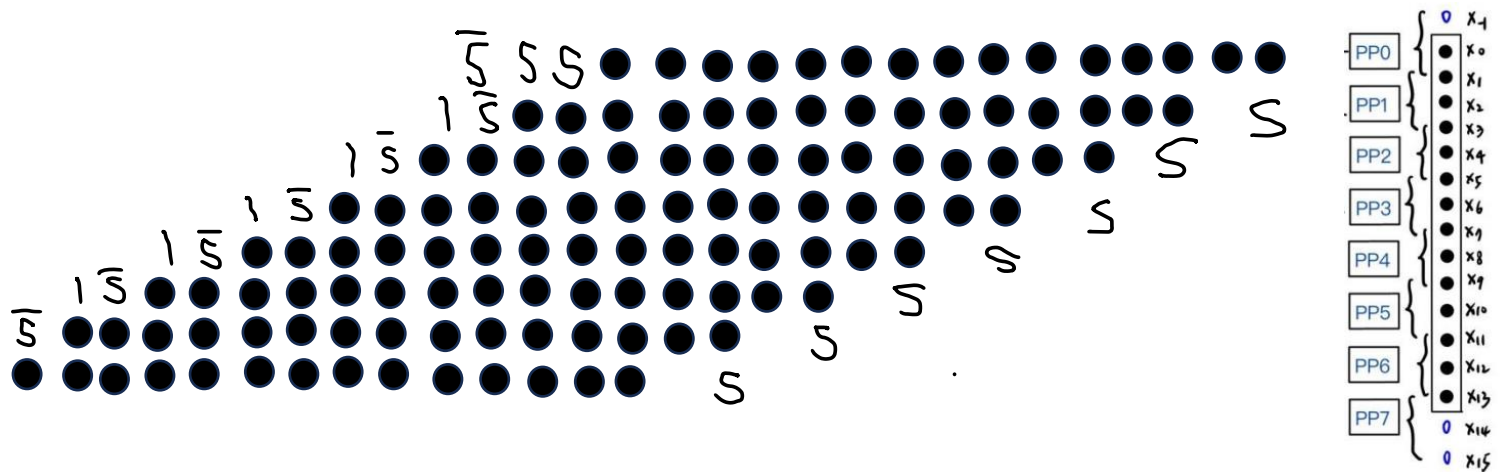
題目要求使用 Radix-4 Booth multiplier 來設計這個 14-bit 的乘法器，而 Booth algorithm 是要一次看三個 bits，而此題因為是 14 個 bits，因此為了達成運算需要多出 Y_{-1} 、 Y_{14} 、 Y_{15} ，其中為了確保乘法器正確運作，所以這三個數皆補 0，而如上圖的 block diagram 所示，我們會有 8 組的 booth decoder 和 multiplicand scaler，而這 8 組會平行處理，因此 critical path 如下圖紅線所標示。



將一次 scan 完的三個 bits 總共 8 組，傳進 Booth Decoder，會產生出 SINGLE_i、DOUBLE_i、NEG_i 三個訊號，其中 SINGLE_i= $Y_{2i-1} * \overline{Y_{2i}} + Y_{2i} * \overline{Y_{2i-1}}$ ，DOUBLE_i= $Y_{2i-1} * Y_{2i} * \overline{Y_{2i+1}} + Y_{2i+1} * \overline{Y_{2i-1}} * Y_{2i}$ ，NEG_i= Y_{2i+1} ，此三個訊號會再與 multiplicand 一起傳入 Booth Selector，再根據不同的輸入訊號產生出 0、X、2X、-2X、-X 五種不同的 partial product。下表即為對 multiplier scan 的三個 bits 對應到的 SINGLE_i、DOUBLE_i、NEG_i 以及 partial product 的 truth table。

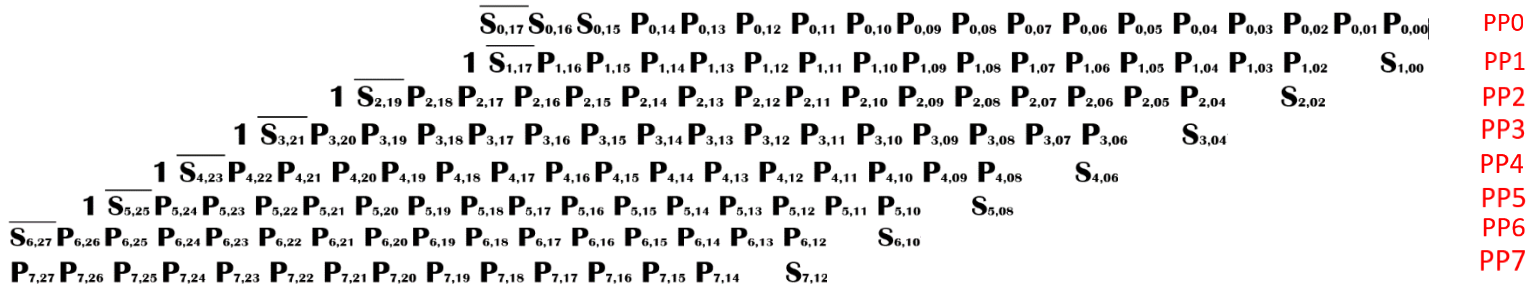
Y_{2i+1}	Y_{2i}	Y_{2i-1}	PP _i	SINGLE _i	DOUBLE _i	NEG _i
0	0	0	0	0	0	0
0	0	1	+X	1	0	0
0	1	0	+X	1	0	0
0	1	1	+2X	0	1	0
1	0	0	-2X	0	1	1
1	0	1	-X	1	0	1
1	1	0	-X	1	0	1
1	1	1	0	0	0	1

(2) Show the Radix Booth-4 encoded partial products with simplified sign extension like that shown in Fig.10.82 of B5_Array and Recoded Multiplier. Draw another diagram with the dots or S that replace pij where i is the ith row and j is the bit location . (20)



PP0~PP6 有 2X、-2X 的可能性，所以有 15-bit。而 PP7 僅有 0、+X 的可能性，所以為 14-bit。

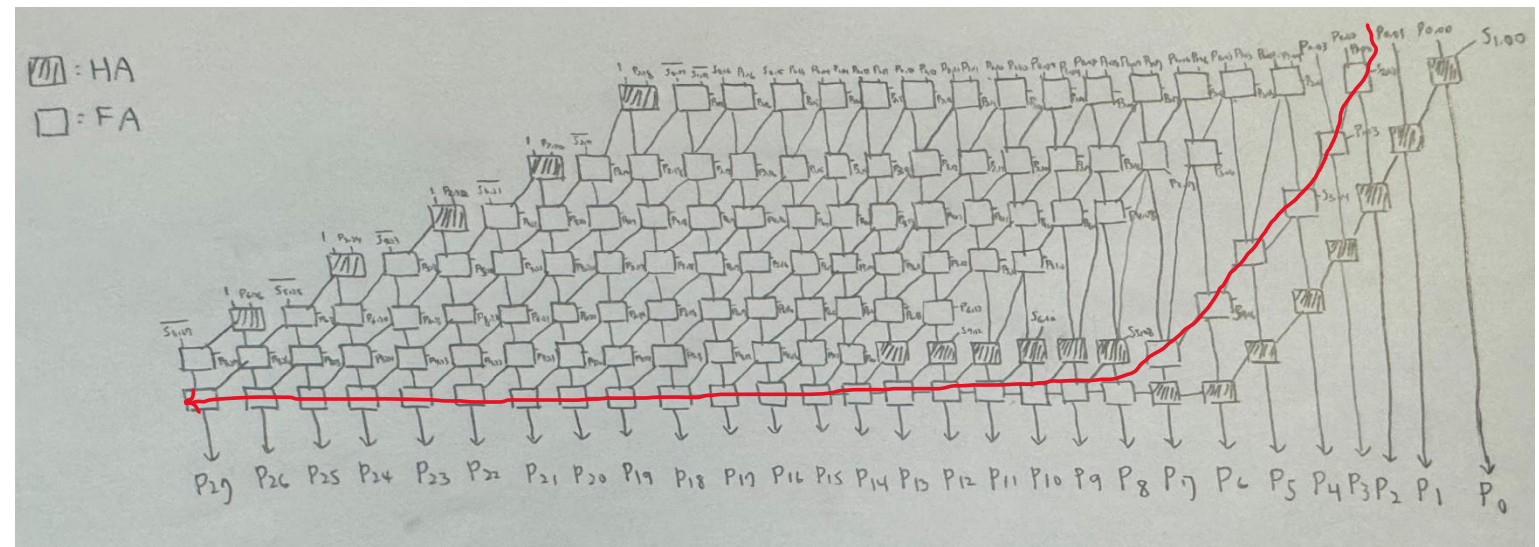
下圖為將 dots 換成 P_{ij} 的圖，其中 sign extension 一樣用 S_{ij} 來表示。



- (3) Design the partial products with array multiplier as that shown in Fig.5.2. Draw the block diagram in terms of FA and HA and p_{ij} as inputs. Show the critical path and indicate the delay time in terms of number of HA and FA. For area, show the number of FA and HA used. Explain your design concepts. (25)

Design concept :

由第二小題所畫的圖來設計電路，其中如果 **ADDER** 的 **INPUT** 只有兩個就使用 **HALF ADDER**，反之則用 **FULL ADDER**，這樣能有效的減少面積，而下圖為我設計的電路圖，其中紅色路徑為 **critical path**。



下表為使用到的 Full adder 以及 Half adder 的數量以及 critical path。

Full adder	Half adder	Critical path
110 個	19 個	$26t_{FA}$

- (4) Design the partial products with Dadda method shown in Fig.5.19. To fairly compare with array multiplier, Carry-ripple adder is used in the last stage of CPA. Show the critical path and indicate the delay time in terms of number of HA and FA. For area, show the number of FA and HA used. Explain your design concepts (25)

Design concept:

在 LEVEL 0 時，由講義 5-22 可得知，因為一開始的 partial products array 有 8 層，透過公式推導可得知，將 2、3、4、6 級合併，可得到最佳解。

在 LEVEL1 時，此時由層合併成 6 層，因此若是 column 超過六層，則使用 Full adder 或是 Half adder 合併，並會產生 carry 和 sum，因此下一次計算層數時要加上前一級所產生的 carry 和 sum，而 LEVEL2 和 LEVEL3 也是如此，以此類推，從 6 層變 4 層，再從 4 層變 3 層。

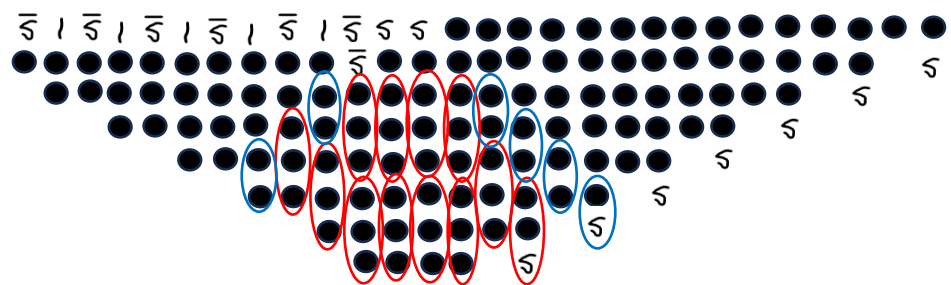
而在最後的 LEVEL4，題目規定要使用 carry ripple adder，來算出最後的 sum，因此 critical path 也在這一級。

下方為我畫的 Dadda Tree，紅色路徑為 Critical path。

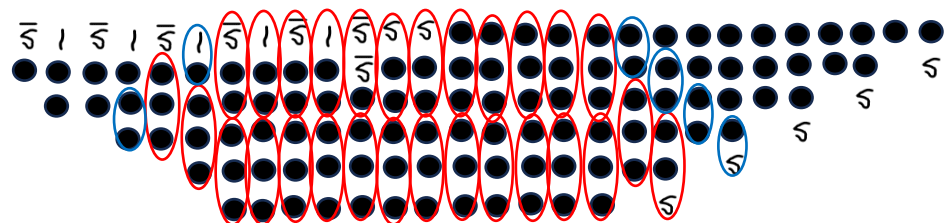
FA ○

HA ○

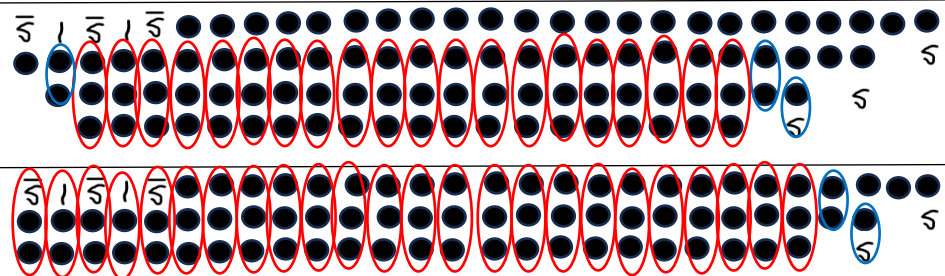
LEVEL 0



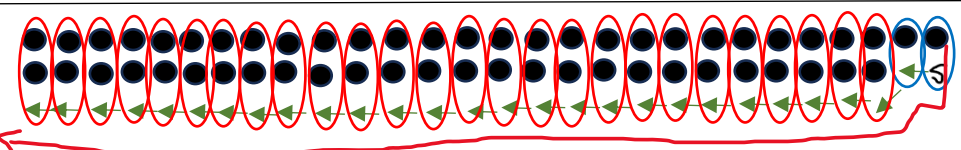
LEVEL 1



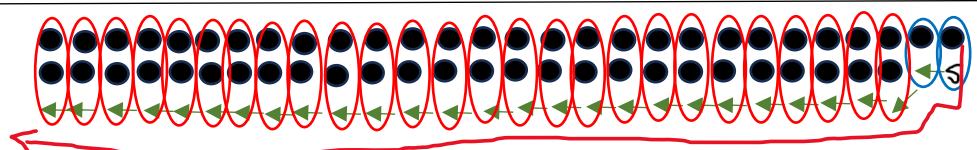
LEVEL 2



LEVEL 3



LEVEL 4



下表為我畫的 Dadda Tree 在各 level 所使用到的 full adder 和 half adder 數量。

LEVEL	FULL ADDER	HALF ADDER
LEVEL0	12	6
LEVEL1	28	6
LEVEL2	20	3
LEVEL3	24	2
LEVEL4	26	2
TOTAL	110	19

而下表為 Array Multiplier 和 Dadda Tree 的 critical path 比較。

	Array Multiplier	Dadda Tree
Critical path	$26t_{FA}$	$26t_{FA}+2t_{HA}$

由上表可發現，這次題目所設計的 14-bit unsigned multiplier，在使用 Array Multiplier 和 Dadda Tree 的兩種情況下，所使用到的 Full adder 和 Half adder 個數相同，不過使用 Dadda Tree 的 critical path 比使用 Array Multiplier 還多兩個 Half adder 的時間，我想這是因為 Dadda Tree 把很多相加的都留到最後才加，而我們也沒有加速最後一層的 carry ripple adder 的運算，所以在 critical path 上才會來得比 Array Multiplier 還高，不過通常 Dadda Tree 可以做到減少面積的目標，只是剛好這次題目所設計的 14-bit unsigned multiplier 沒有什麼可以省略的運算，因此才會造成犧牲了 delay time 卻也沒有減少面積的情況。