Multiplication of Binary numbers

4-bit i/p

$a_3\ a_2\ a_1\ a_0$ A

$\times\quad b_3\ b_2\ b_1\ b_0$ B

Partial Products

$a_3b_0\quad a_2b_0\quad a_1b_0\quad a_0b_0$

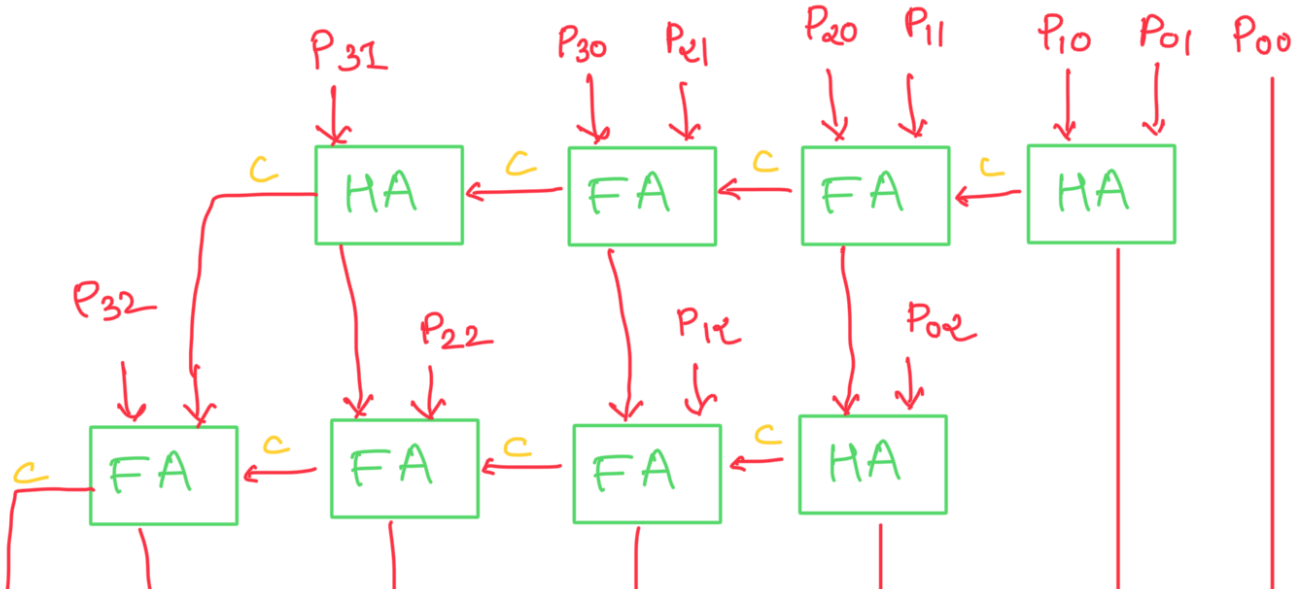
$+\quad a_3b_1\quad a_2b_1\quad a_1b_1\quad a_0b_1\quad X$

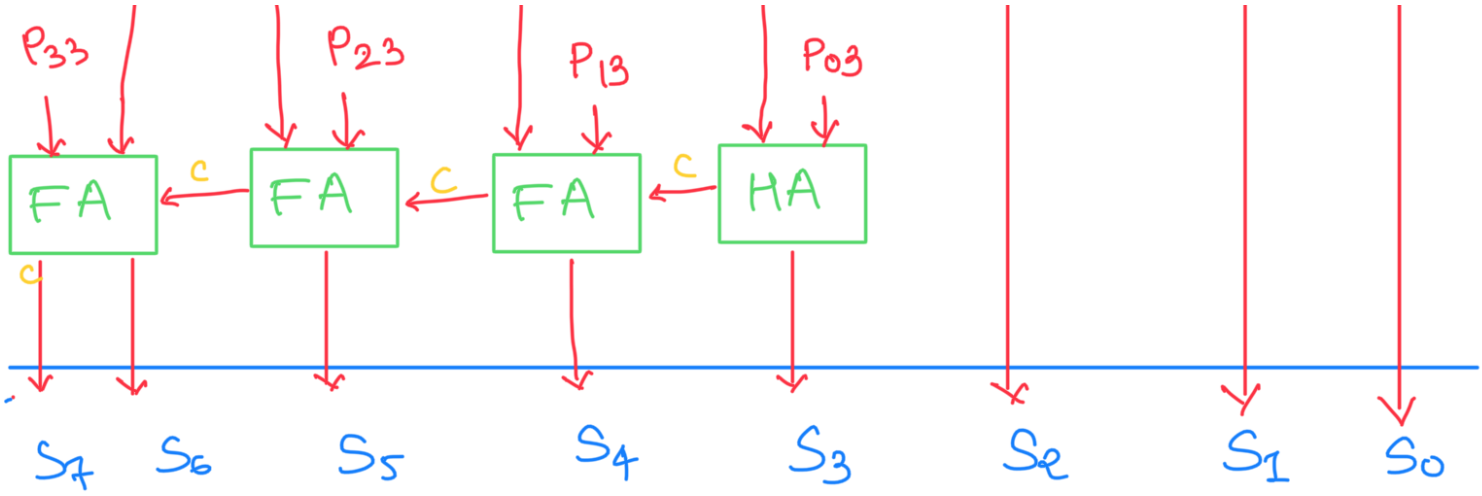
$+\quad a_3b_2\quad a_2b_2\quad a_1b_2\quad a_0b_2\quad X\quad X$

$+\ a_3b_3\quad a_2b_3\quad a_1b_3\quad a_0b_3\quad X\quad X\quad X$

And gate

Let's call the partial product for bit A_i & B_j, P_{ij}. Now for addition we have to use Full adders (FAs) or half adders (HAs) according to the needs.





Final Ans. : S[7:0]

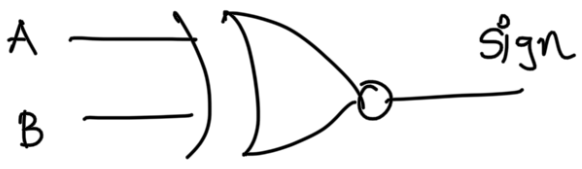
Now if we want to add the signed bit for multiplication purpose, we can provide an additional signed bit for both inputs, which then propogate through logic as the sign of the output.

Let's Assume, 0 bit ⟶ "−"ve number

1 bit ⟶ "+"ve number

Truth table:

A	B	sign
0	0	1
0	1	0
1	0	0
1	1	1



$A \overline{\oplus} B = sign$

2. If we use 2'-complement as input

But if ... number, it is very important to do sign extention upto M.S.B. for each term of multiplication.

At the end of the multiplication, We ignore the extra bits to the left corner and treat left bit M.S.B. as a sign of the answer.

Note : But for this method, when multiplier be the negative number, it won't give the correct answer.

→ To get the correct answer, we have to take 2's complement of the last multiplicative factor and then we can get correct result by adding it to the answer. (Or to say subtract the last factor insted of adding it.)

Logic : So we will take 2's complement for both the cases cause it won't effect the answer for positive multiplier.

$a_3\ a_2\ a_1\ a_0$ A

$\otimes$ $b_3\ b_2\ b_1\ b_0$ B

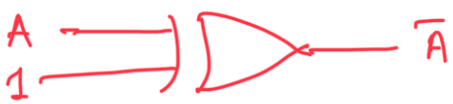
a_3b_0	a_3b_0	a_3b_0	a_3b_0	a_3b_0	a_2b_0	a_1b_0	a_0b_0
a_3b_1	a_3b_1	a_3b_1	a_3b_1	a_2b_1	a_1b_1	a_0b_1	X
a_3b_2	a_3b_2	a_3b_2	a_2b_2	a_1b_2	a_0b_2	X	X
$\overline{a_3b_3}$	$\overline{a_3b_3}$	$\overline{a_2b_3}$	$\overline{a_1b_3}$	$\overline{a_0b_3}$	X	X	X
⊕ 0	0	0	0	1	X	X	X
S_7	S_6	S_5	S_4	S_3	S_2	S_1	S_0

Optimization to avoid additional Hardware Problem ?

Here, if we can make the extended signed bit "0" by adding or subtracting something, it may lead us to optimize the hardware use.

But How! ; Let's say the M.S.B. is A.

$$A \oplus 1 = \bar{A}$$

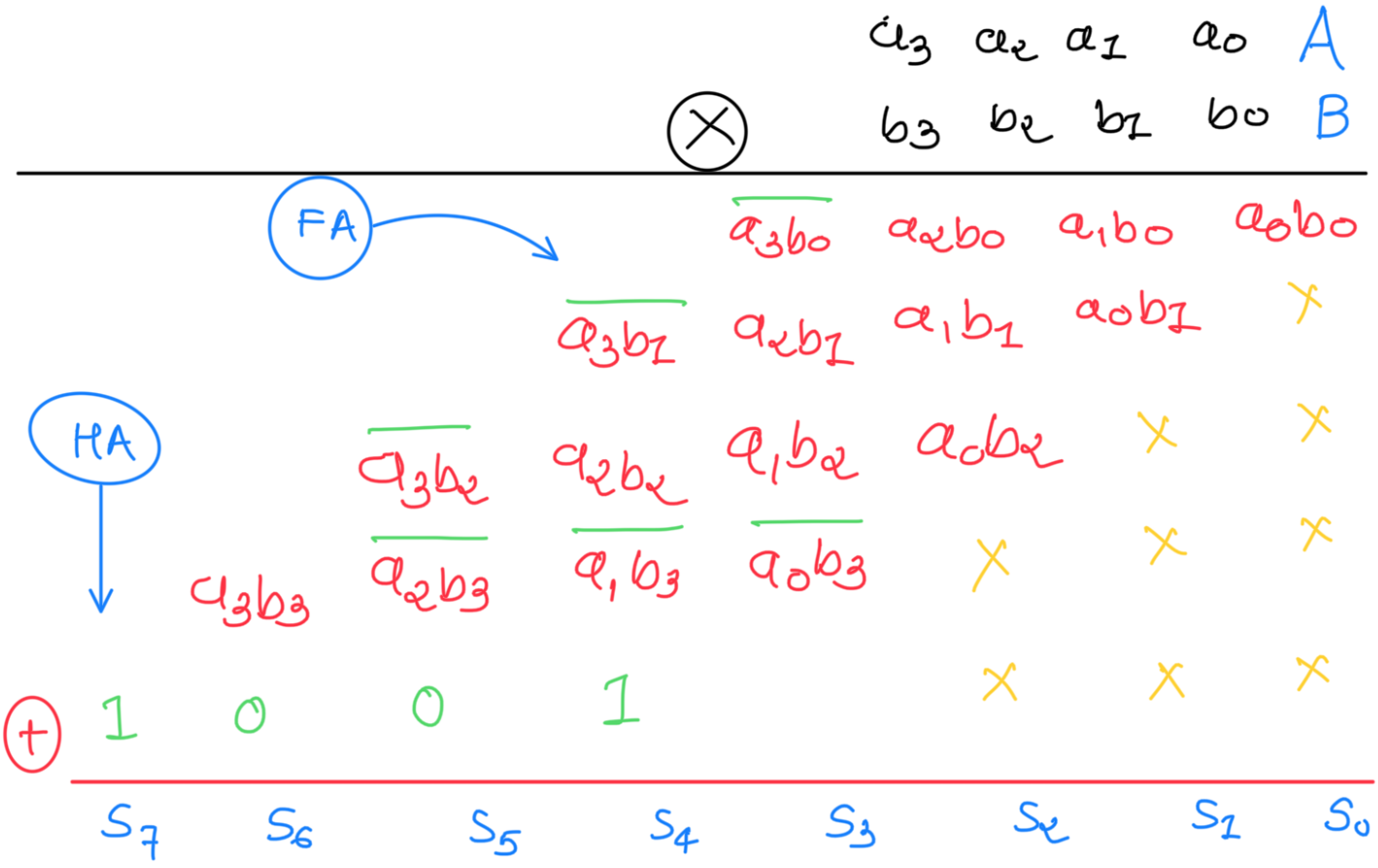


So by adding "1" to the MSB, we can modify all the sign extentation to "0" and hence no need to add it in the answer.

⇒ we need to add (+1) to all the signed extention bits. And adding "1" means taking complement.

Because we are adding the numbers, we also have to substract or say add 2's complement at the end.

Result : so, the final result will look like this :



⇒ By only addition of 2 adders, we are able to make signed multiplier for

4-bit x 4-bit binary multiplication.