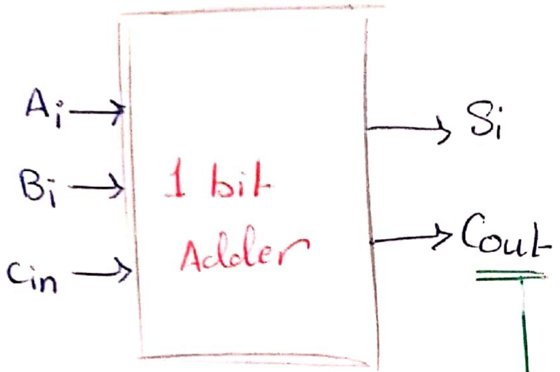
[Adders are example for Iterative combinational circuits]

[4 bits + 4 bits = 5 bits

ال carry الأخير راح يكون جزء من الجواب ✓



it will become the C_{in} (the carry) for the next adder if exists ✓

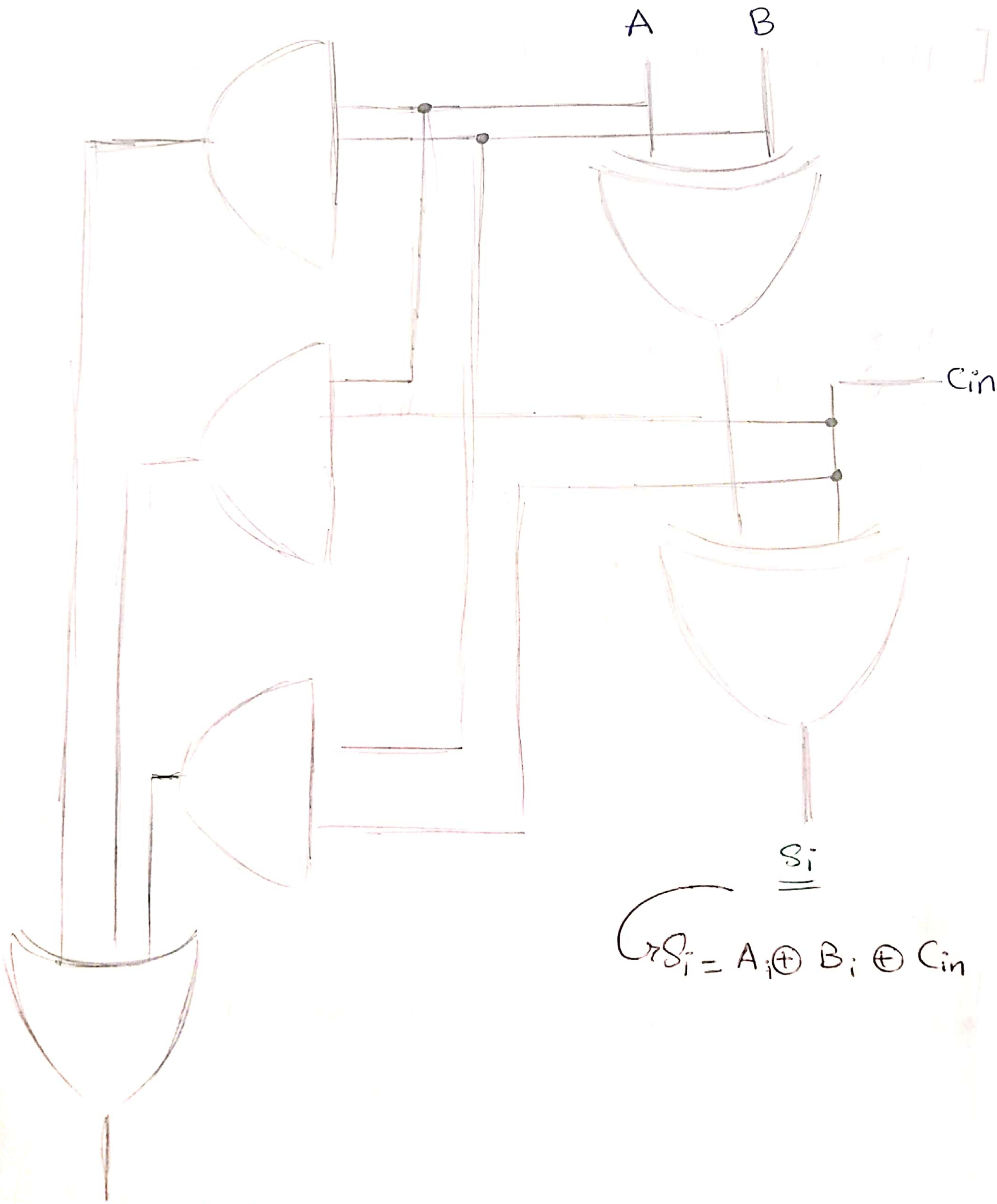
A_i	B_i	C_{in}	S_i	C_{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

After doing the k-map, and performing equations we find:

[↳ S_i (sum equation) = XOR function / the odd function ✓

$= A_i \oplus B_i \oplus C_{in}$

C_{in} = And gates ✓



$S_i = A_i \oplus B_i \oplus C_{in}$

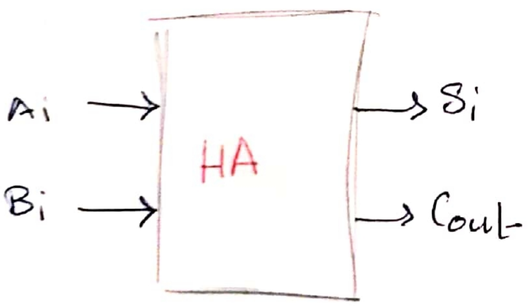
$C_{out} \rightarrow A_iB_i + A_iC_{in} + B_iC_{in}$

[you do the single bit adder
then you design the vector

Single bit → HALF ADDER 2 input-bit addition block
Single bit → FULL ADDER 3 input-bit addition block

[(HA) helps to design the (FA) circuit with less cost]

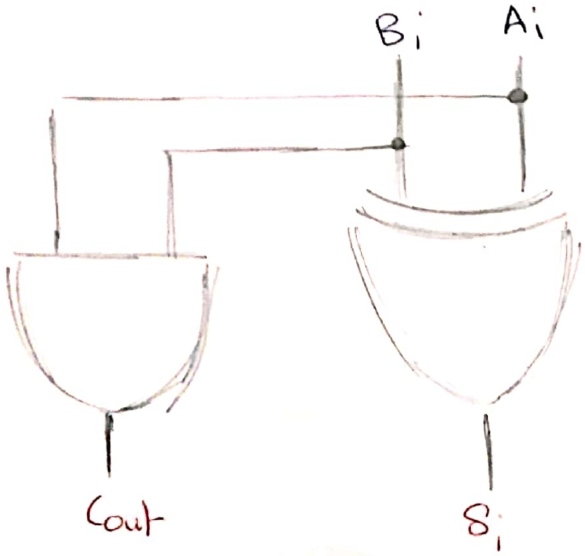
HALF ADDER design:



A_i	B_i	S_i	Cout
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

$S_i = A_i \oplus B_i$

$C_{out} = A_i B_i$ ✓



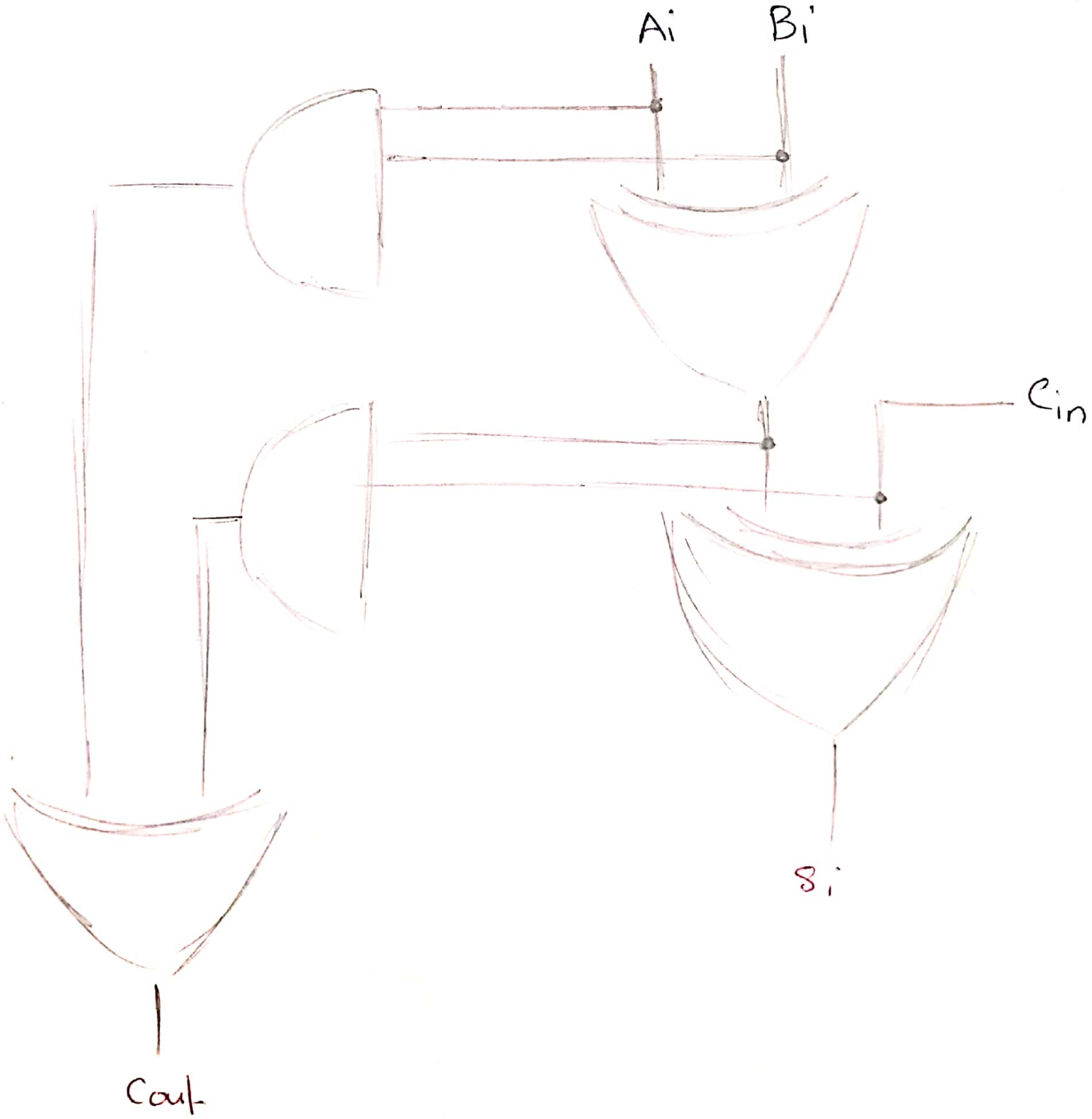
العلاقة بين ال FA وال HA من خلال د. الرياضي

FA equations

$$S_i = A_i \oplus B_i \oplus C_{in}$$

$$C_{out} = A_iB_i + A_iC_{in} + B_iC_{in} = A_iB_i + C_{in}(A_i \oplus B_i)$$

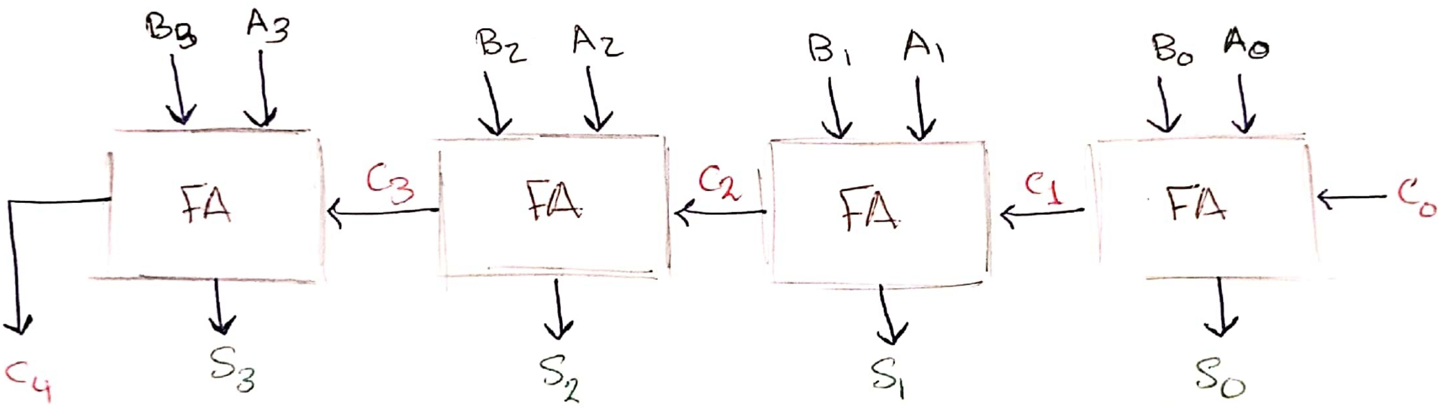
هاي معادلة ال HA الخاصة بال sum بنستخدمها عشان نقلل عدد ال And Gates



After Designing the internal adder block you design the vector (Ripple-carry Adder)

ال Cout الي طالع من block ، بصير ال Cin للبلوك الي بعده ✓

4-bit Ripple-Carry Adder :-



Scanned with CamScanner

[Subtraction]

$$\begin{array}{l} N - \\ M \\ \hline \end{array}$$

↳ if $N < M$ → ① do the subtraction and the dummy borrow

② do the correction.

[To Summarize]

end borrow = 1

$(N < M)$

→ do the subtraction with dummy borrow and will result → $(N-M)+2^n$

→ do the correction

$2^n - ((N-M)+2^n)$

and will result the negative number ✓

end borrow = 0

$(N \geq M)$

→ do the subtraction normally, for a positive answer

[The Complements]

↳ Diminished Radix Complement

→ you simply take the complement to the max number

also

$(r-1)'s = r^n - 1 - N$

↳ Radix complement.

→ find the $(r-1)'s$ comp. then add (1)

also

$r's = r^n - N$

→ Binary Complements

$(r-1)$'s → only flip the bits

① 1's complement can be found by NOT (inverter)

② 2's → 1's + 1

or ↳ use the fast method [تثبيت الارقام لعند أول (1) بعدين بنقلب باقي ال (bits)]

علاقة ال complement بال Subtraction

↲ بيعطيني طريقة جديدة للطرح

$$N - M = N + 2\text{'s of } M$$

needs correction

↳ find 2's M and add it to N

↳ the end carry will be 0

↳ the correction is done as follows:

find the 2's for the result and make it negative.

does not need correction

↳ find 2's M and add it to N

↳ the end carry will be 1 and won't be taken with the result.

↳ the result should be taken as is.