

Digital Systems- IT131

Dr. Alaa Eldeen S. Mohamed

Lecture 06: Combinational Circuits

Contents

- Combinational circuits
- Design procedure
- Half adder
- Full adder
- Parallel Adder
- Adder subtractor

Combinational Circuits

Combinational Circuits

- Logic circuits for digital systems may be **Combinational circuit** or **Sequential circuit**
- **Combinational circuit**: logic circuit whose outputs at any time are determined directly and only from the present input combination.
- A combinational circuit performs a specific information-processing operation fully specified logically by a set of **Boolean functions**.
- **Sequential circuit** : depends on combination of inputs and state of the storage elements.

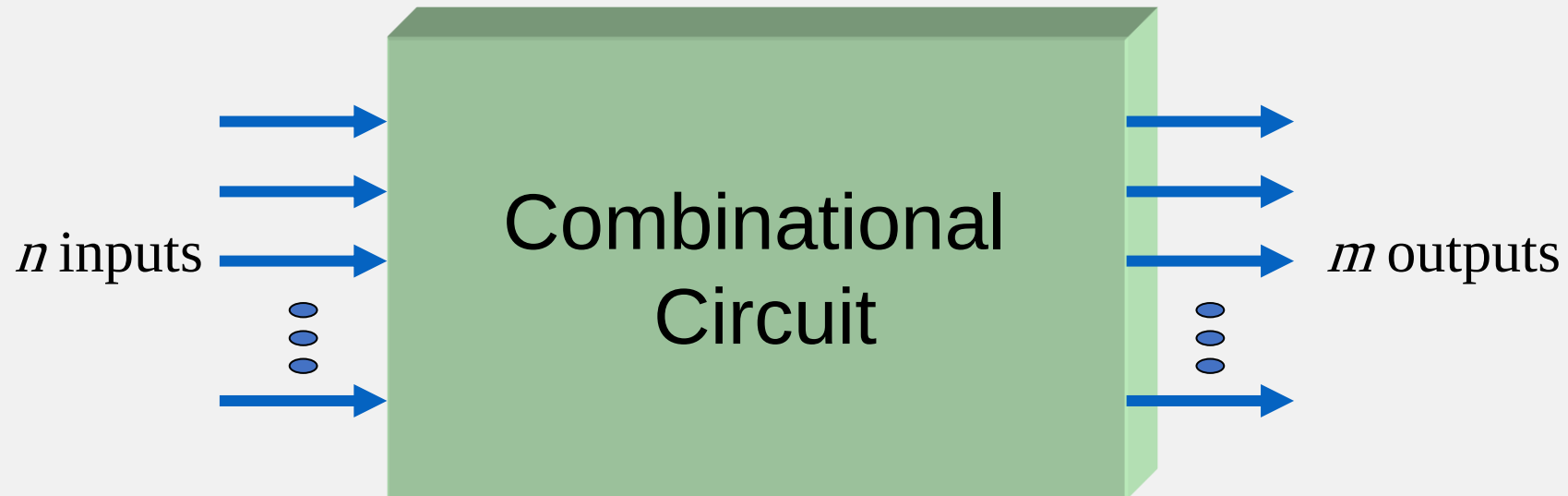
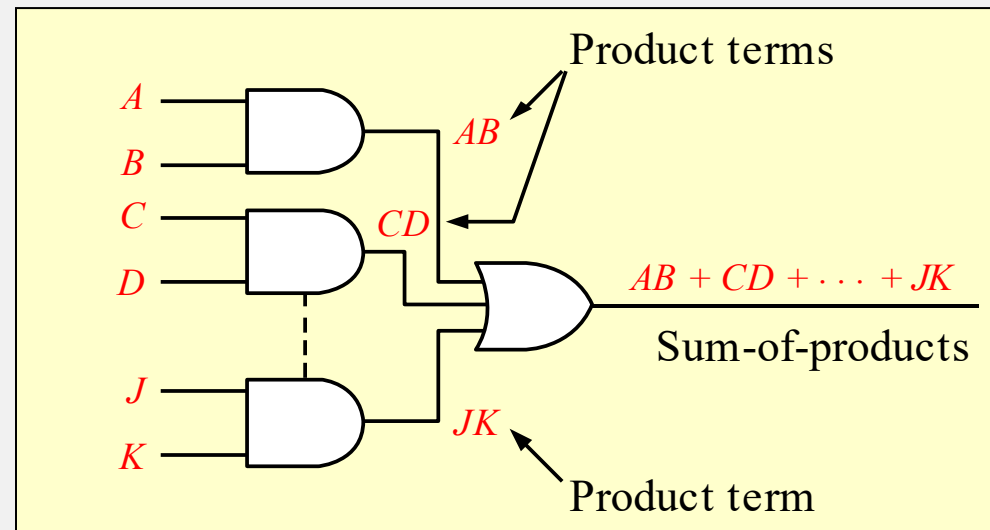


Fig.: Block Diagram of Combinational Circuit

Combinational Circuits

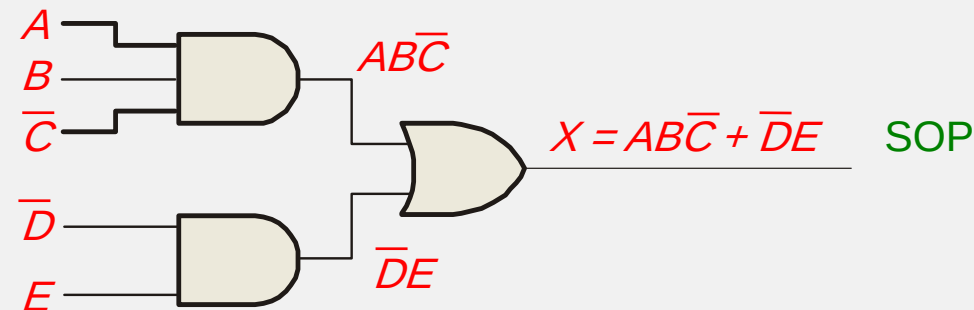
Combinational Circuits

- In Sum-of-Products (SOP) form, basic combinational circuits can be directly implemented with AND-OR combinations if the necessary complement terms are available.

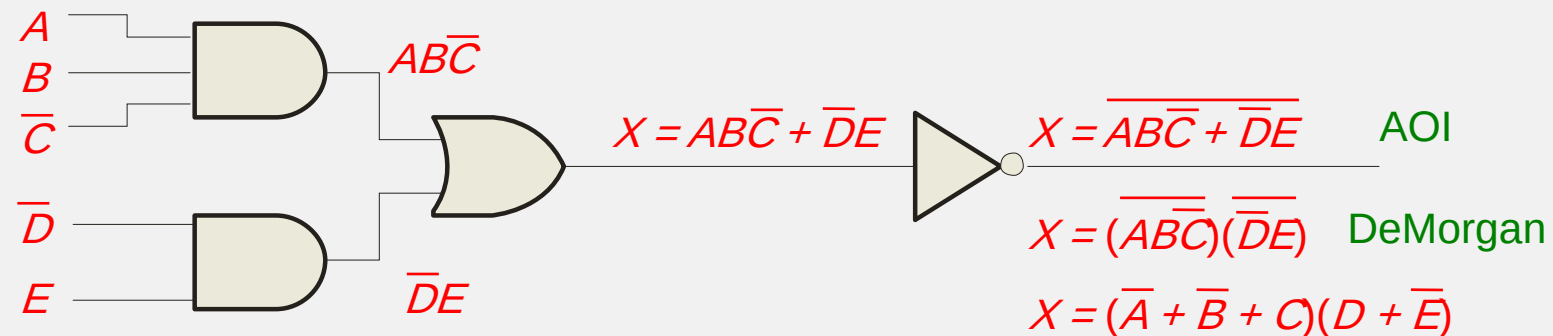


Combinational Circuits

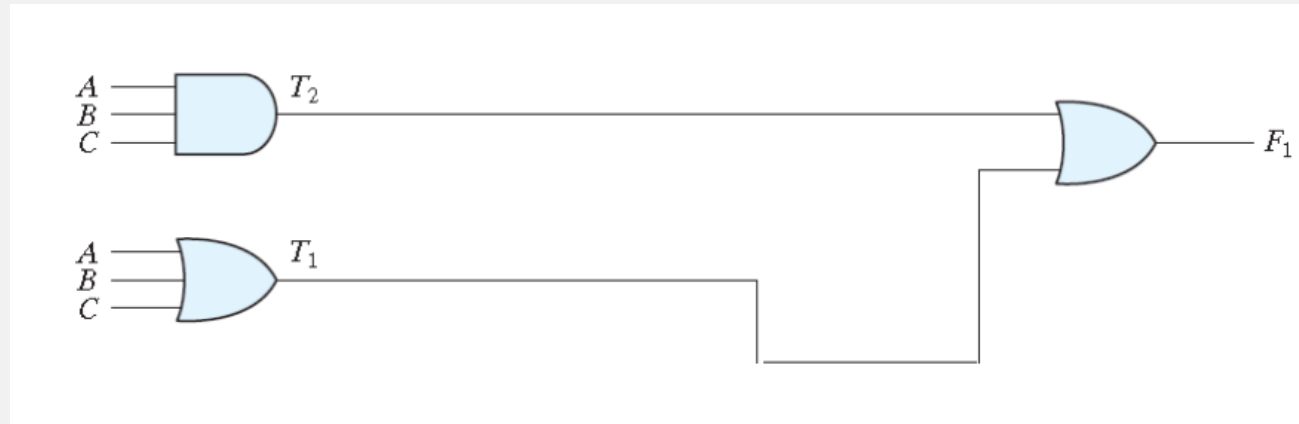
- An example of an SOP implementation is shown. The SOP expression is an AND-OR combination of the input variables and the appropriate complements.



- When the output of a SOP form is inverted, the circuit is called an AND-OR-Invert circuit (AOI). The AOI configuration lends itself to product-of-sums (POS) implementation.
- An example of an AOI implementation is shown. The output expression can be changed to a POS expression by applying DeMorgan's theorem twice.



- Exercise: Analyze the following circuit (give the equivalent equation)



- $T_2 = A.B.C$
- $T_1 = A+B+C$
- $F_1 = T_2 + T_1 = A.B.C + (A+B+C)$

Combinational Circuits

Design Procedure

- The design procedure for combinational circuits is:
 1. Define the problem.
 2. Determine the number of input/output variables.
 3. Assign letter symbols to input/output variables.
 4. Derive the truth table that defines the required relationships between inputs & outputs.
 5. Obtain the simplified Boolean function for each output as a function of the input variables.
 6. Draw the logic diagram and verify the correctness of the design.

Combinational Circuits

Example: Exclusive-OR (XOR) Logic

- Problem Definition
 - XOR" an abbreviation for "Exclusively-OR." The simplest XOR gate is a **two-input combinational circuit** that **outputs** a logical "1" if the two input values differ, i.e., its output is a logical "1" if either of its inputs are 1, but not at the same time (exclusively).
- Inputs / outputs
 - The Boolean expression for a two-input XOR gate, with inputs A and B and output X:

Combinational Circuits

Example: Exclusive-OR (XOR) Logic

- Truth Table:
 - The symbol and truth table for an XOR is shown in Figure.

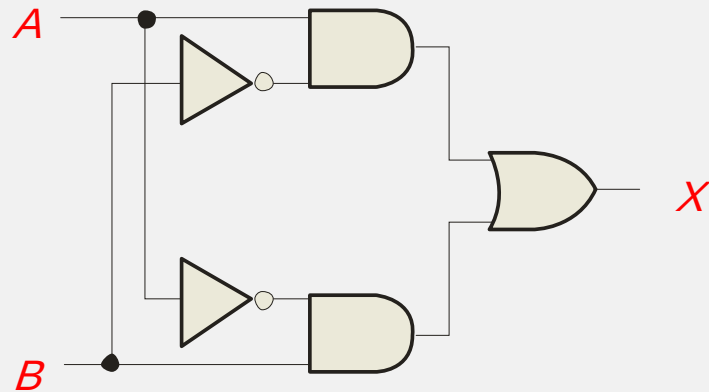
Inputs		Output
A	B	X
0	0	0
0	1	1
1	0	1
1	1	0

- Obtain the simplified Boolean function for the output as a function of the input variables.

- $$X = A' \cdot B + A \cdot B'$$

- Draw the logic diagram and verify the correctness of the design.

The circuit can be drawn as



Symbols:



$$A \oplus B$$

Combinational Circuits

Example: Exclusive-NOR (XNOR) Logic

- Truth Table:
 - The symbol and truth table for an XNOR is shown in Figure.

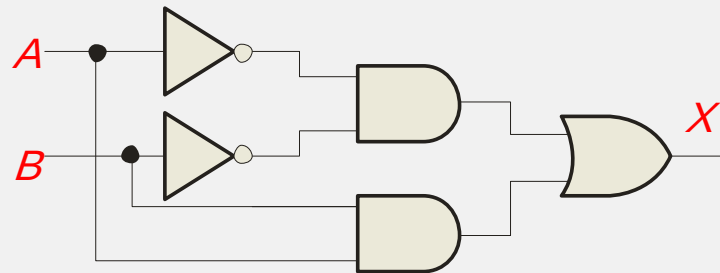
Inputs		Output
A	B	X
0	0	1
0	1	0
1	0	0
1	1	1

- Obtain the simplified Boolean function for the output as a function of the input variables.

- $$X = A \cdot B + A' \cdot B'$$

- Draw the logic diagram and verify the correctness of the design.

The circuit can be drawn as



Symbols:

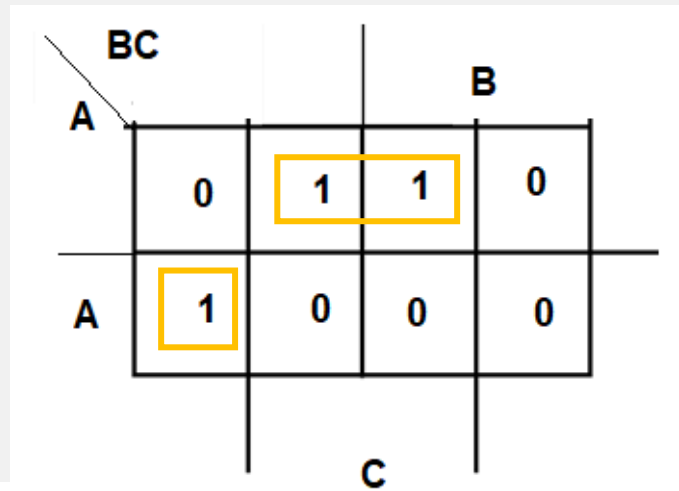


Combinational Circuits

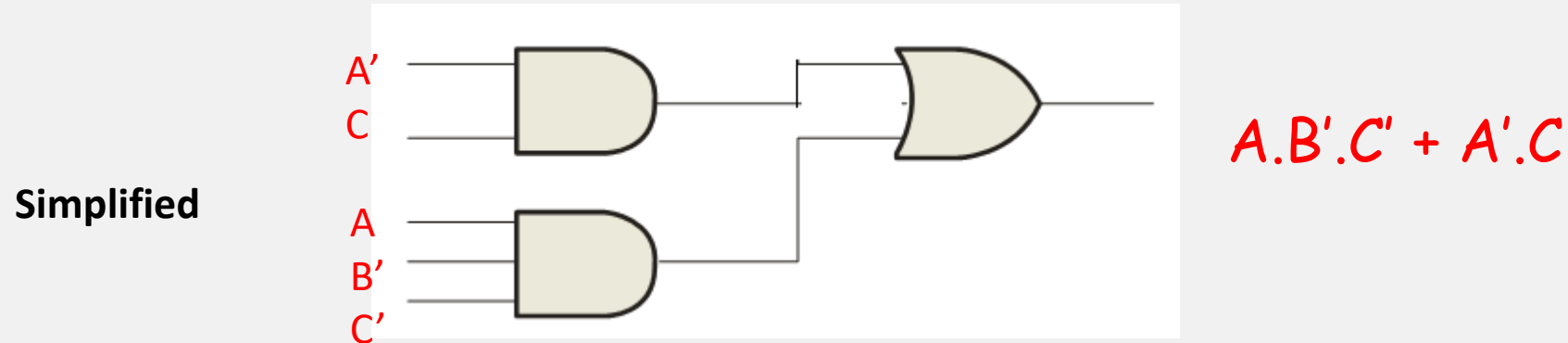
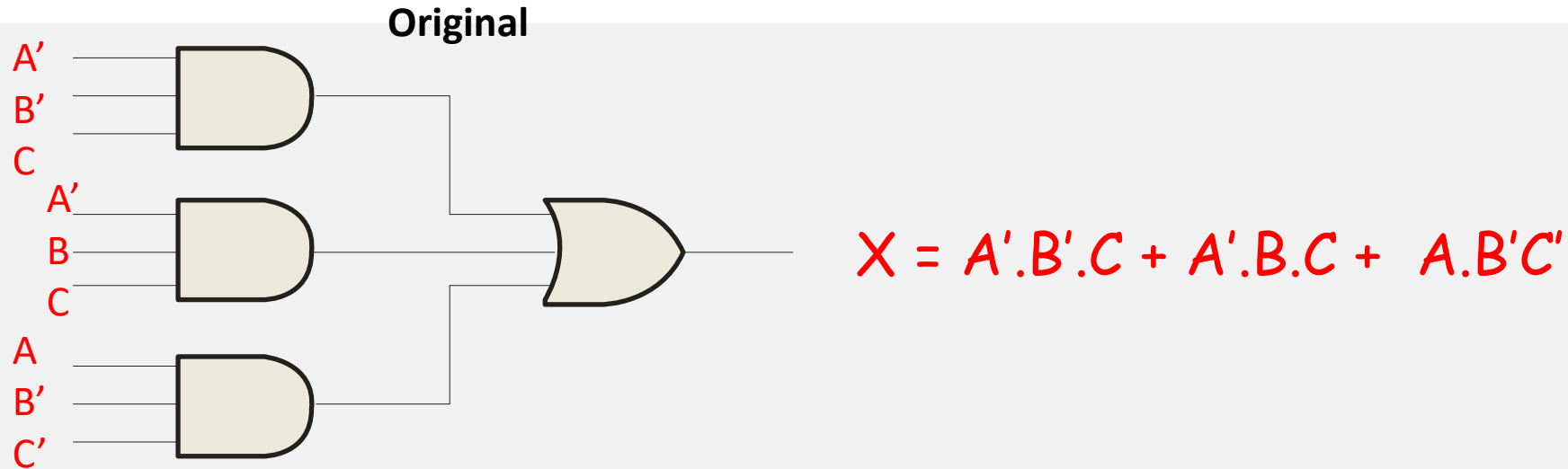
- Example: Design the combinational function shown in the following truth table

- $F(A,B,C) = A'.B'.C + A'.B.C + A.B'.C'$
- Obtain the simplified Boolean function using K Map:

$$F(A,B,C) = A.B'.C' + A'.C$$



A	B	C	F
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0



- Exercise

Design a combinational circuit with three input and one output.

- a. The output is 1 when the binary value of the inputs is less than 3 the output is 0 otherwise.

Solution

- Inputs / outputs

3 inputs, (A,B,C) and one output (F)

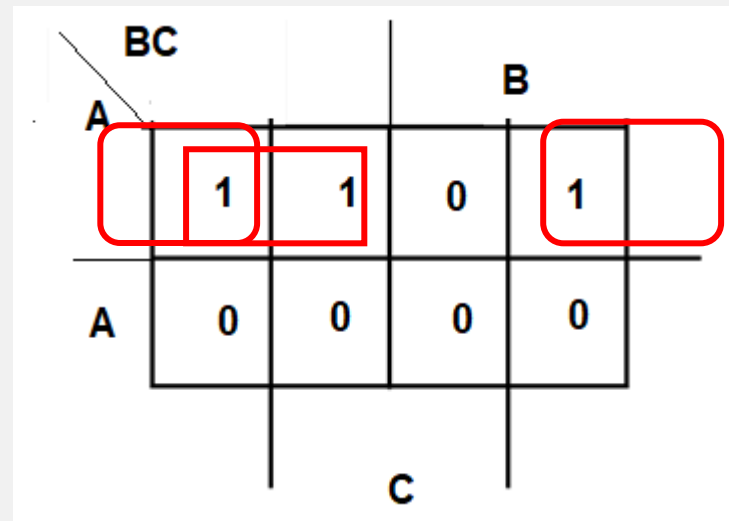
Combinational Circuits

- Truth Table:

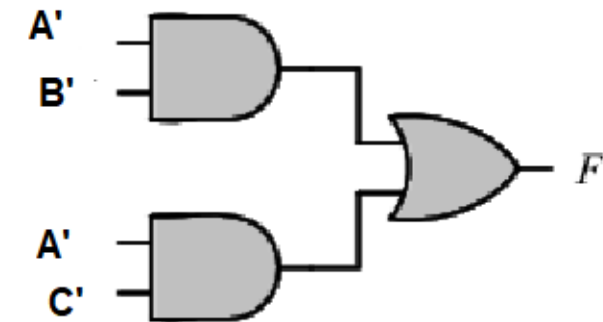
a. The output is 1 when the binary value of the inputs is less than 3. the output is 0 otherwise.

A	B	C	F
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

Obtain the simplified Boolean function for the output as a function of the input:



$$F(A,B,C) = A'.B' + C'.A'$$



- **Exercise:**
- A majority circuit is a combinational circuit whose output is equal to 1 if the input variables have more 1's than 0's. The output is 0 otherwise.
- Design a three inputs majority circuit by finding the circuits truth table. Boolean equation and a logic diagram.

- **Solution**

- Inputs / outputs

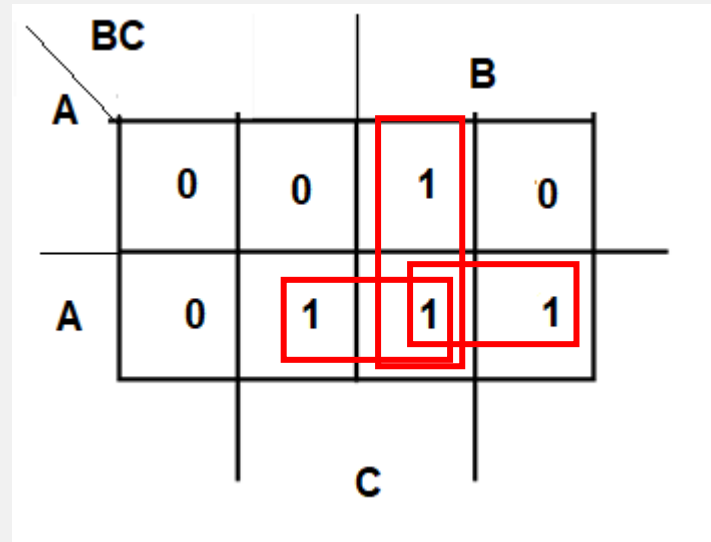
3 inputs, (A,B,C) and one output (F)

Combinational Circuits

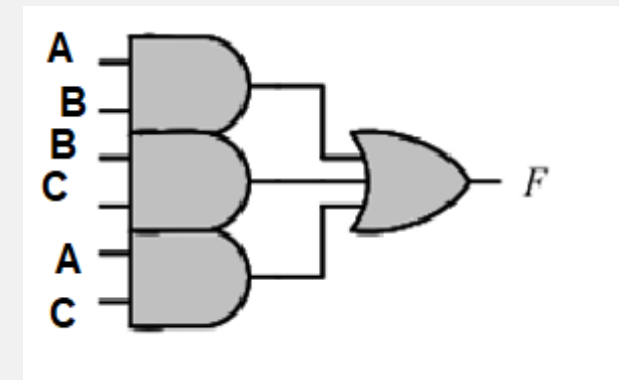
- Truth Table:

A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Obtain the simplified Boolean function for the output as a function of the input:



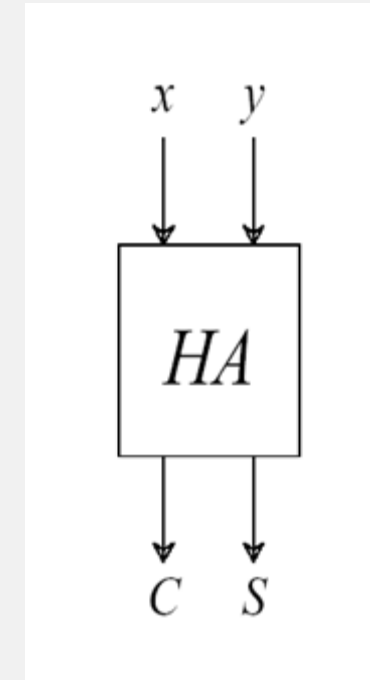
$$F(A,B,C) = A.B + B.C + A.C$$



Combinational Circuits

Half Adder Circuit

- The most basic arithmetic operation is the addition of 2 bits. A combinational circuit that performs this operation is called a *half-adder*.
- 2 inputs and two outputs
 - Inputs : X , Y
 - Outputs: S , C

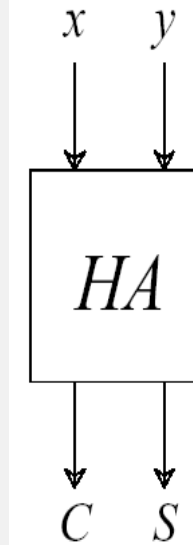


Combinational Circuits

Half Adder Circuit

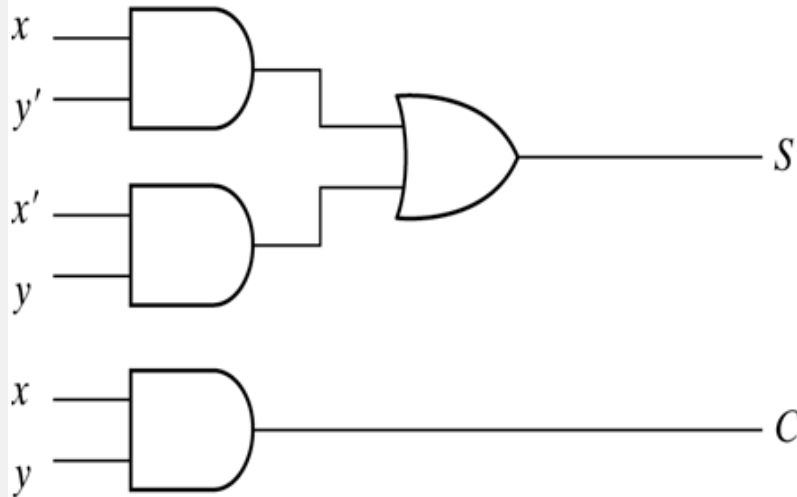
- Truth Table.
- Equations:
 - $S = xy' + x'y = x \oplus y$
 - $C = xy$
- The **sum** can be found using the XOR operation and the **carry** using the AND operation.

x	y	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

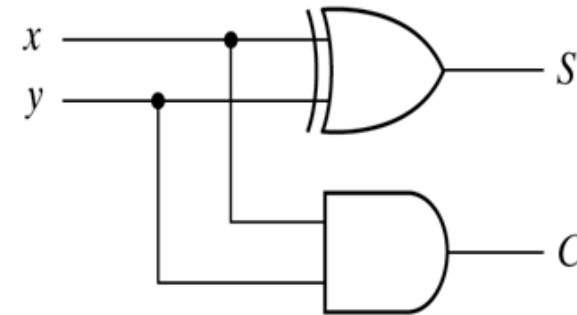


Combinational Circuits

Half Adder



$$(a) \begin{aligned} S &= xy' + x'y \\ C &= xy \end{aligned}$$



$$(b) \begin{aligned} S &= x \oplus y \\ C &= xy \end{aligned}$$

Implementation of Half-Adder

Combinational Circuits

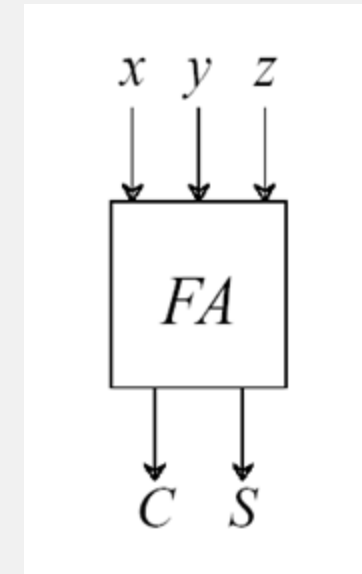
Full Adder

➤ A combinational circuit that performs the addition of 3 bits is called a *full-adder*, which can be implemented by 2 half-adders.

➤ 3 inputs and two outputs

➤ Inputs : X, Y, Z

➤ Outputs: C, S



Combinational Circuits

Full Adder

➤ Truth Table.

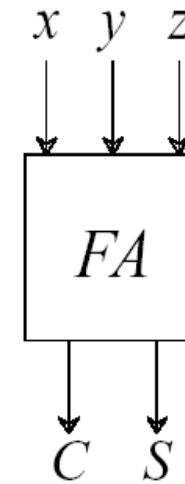
➤ Equations:

➤ Sum:

$$S = xy'z' + x'y'z + xyz + x'y'z$$

➤ Carry:

$$C = x'yz + xy'z + xyz' + xyz$$



x	y	z	S	C
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

Combinational Circuits

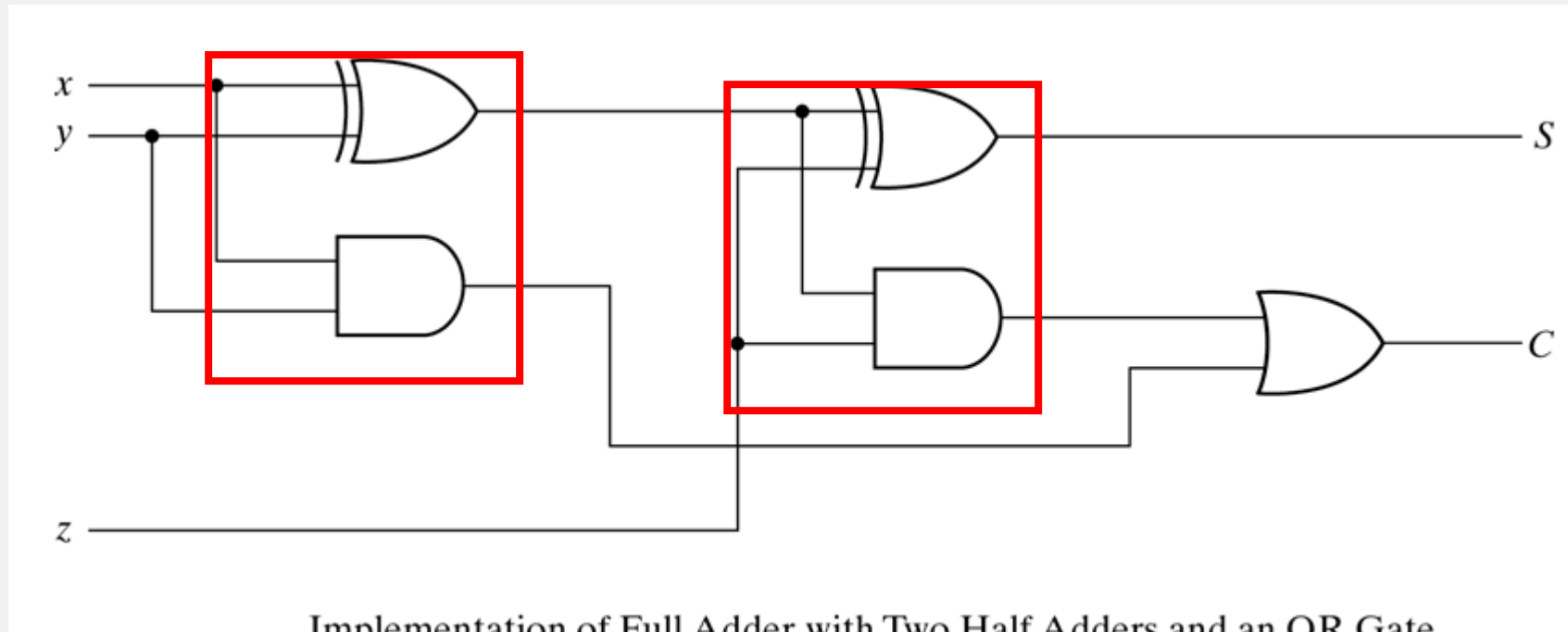
Full Adder

$$\begin{aligned}
 S &= xy'z' + x'y'z' + xyz + x'y'z = \\
 &\quad z'(xy' + x'y) + z(xy + x'y') \\
 &= z'(xy' + x'y) + z(xy' + x'y)' \\
 &= z \oplus (x \oplus y)
 \end{aligned}$$

$$\begin{aligned}
 C &= x'y'z + xy'z + xyz' + xyz \\
 &= x'y'z + xy'z + xy \\
 &= z(x'y + xy') + xy \\
 &= z(x \oplus y) + xy
 \end{aligned}$$

$$S = z \oplus (x \oplus y)$$

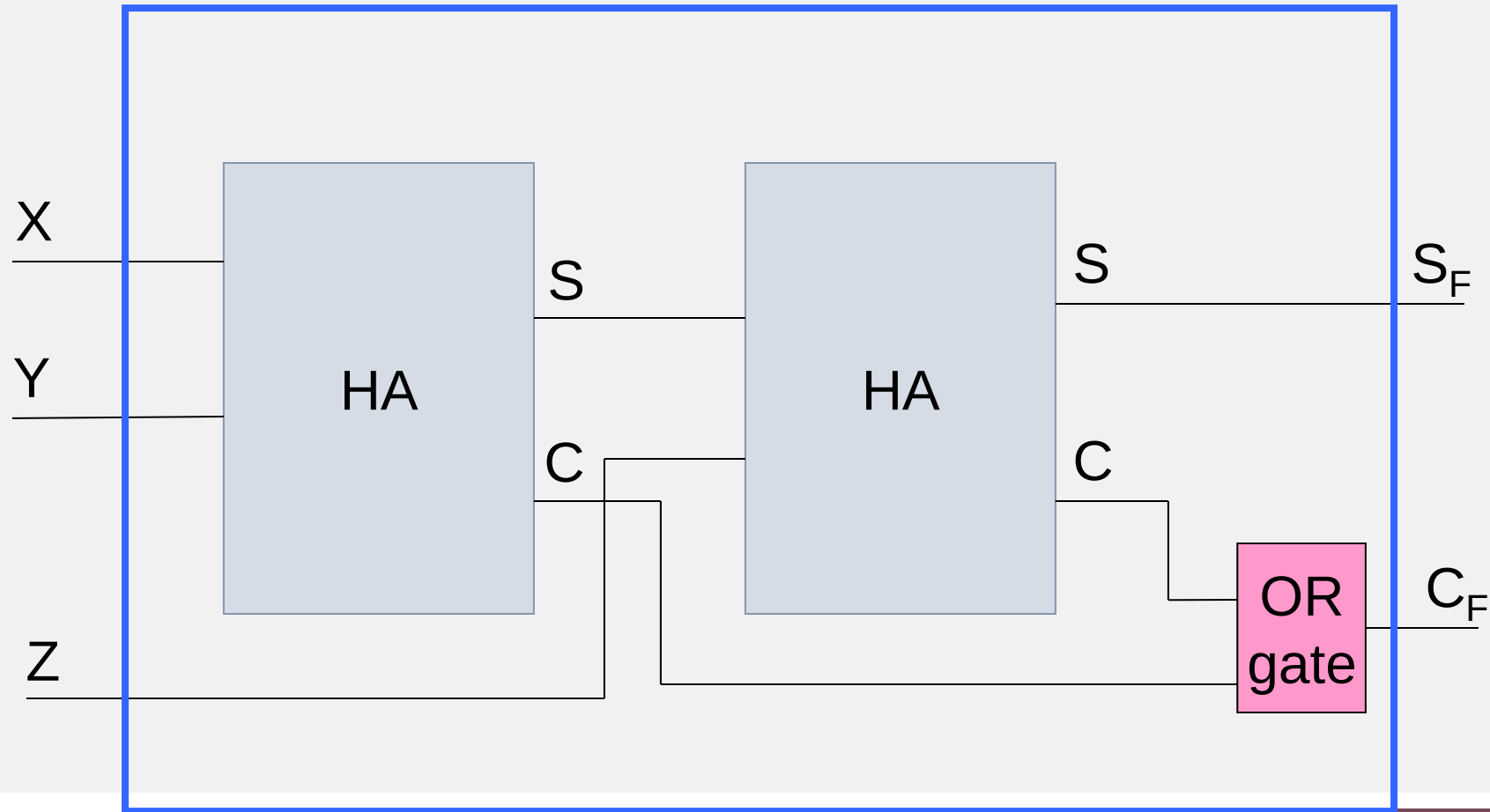
$$C = z(x \oplus y) + xy$$



Implementation of Full Adder with Two Half Adders and an OR Gate

Combinational Circuits

Full Adder



Combinational Circuits

4-Bit Binary Adder (parallel adder)

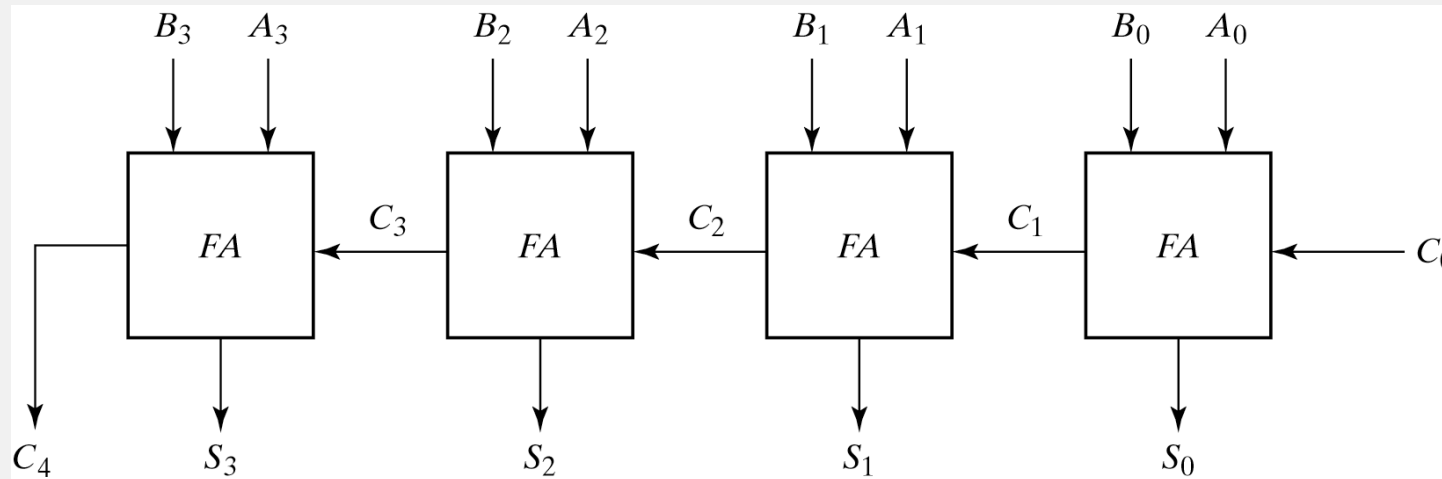
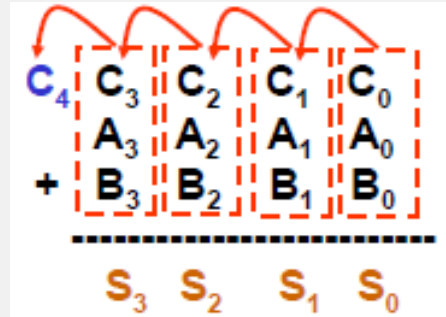


Fig. 4-9 4-Bit Adder

Combinational Circuits

4-Bit parallel Binary Adder

- The logic symbol for a 4-bit parallel adder is shown. This 4-bit adder includes 4 Full Adder. As shown in the following graph, the 4-bit adder has 9 inputs (8 inputs for the 4-bit A and B + one input for C_0) and 5 outputs (4 outputs for the 4-bit sum and one output for the carry C_4).

What will be the case if we have 3-bit parallel adder or 5-bits parallel Adder?

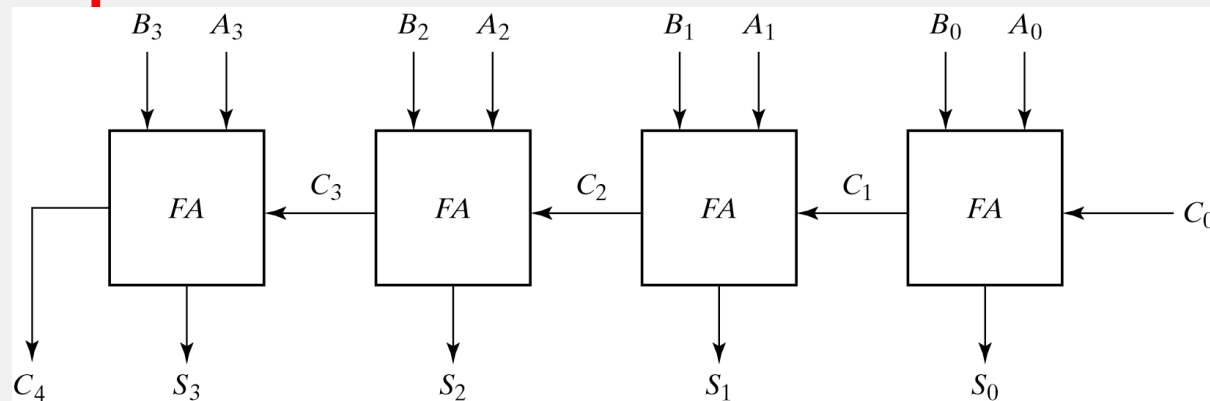
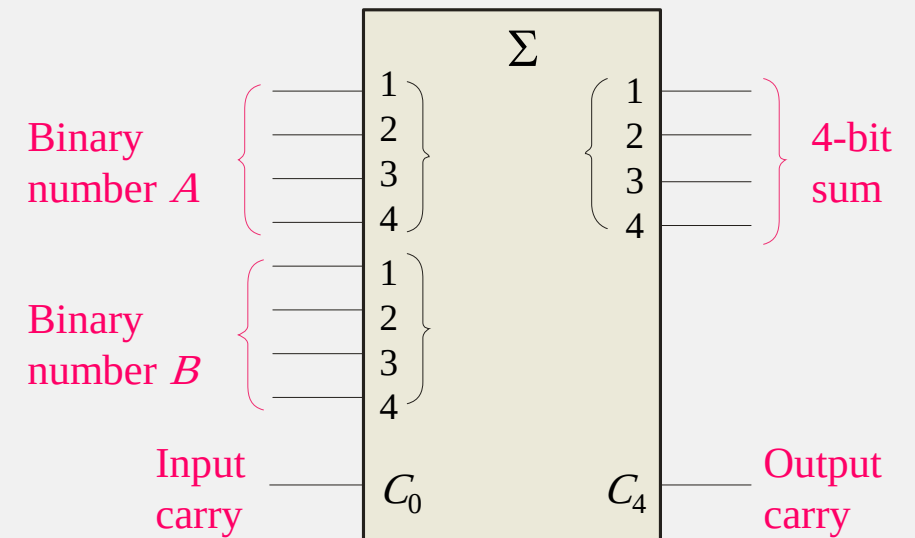


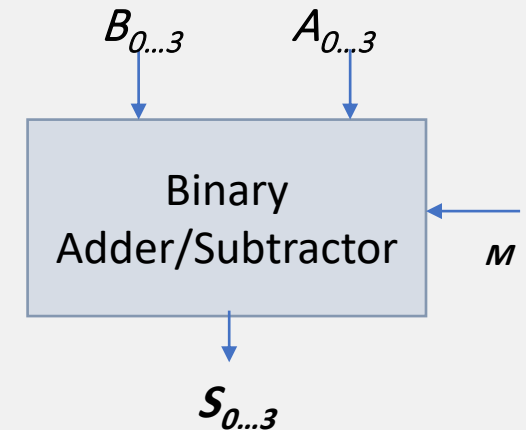
Fig. 4-9 4-Bit Adder



Combinational Circuits

2's Complement Adder/Subtractor

- **Subtraction** can be done by addition of the 2's Complement.
- $A - B = A + B' + 1$
 1. Complement each bit of B (1's Complement.) $[B \rightarrow B']$
 2. Add 1 to the result to get 2's complement of B. $(B' + 1)$
 3. $A - B = A + B' + 1$
- Actually, we can use an adder for **both** addition and subtraction:
 - Complement subtrahend for subtraction
 - Do not complement subtrahend for addition
- Thus, to form an adder-subtractor circuit, we only need a selective complemener and an adder.



XOR TRUTH TABLE

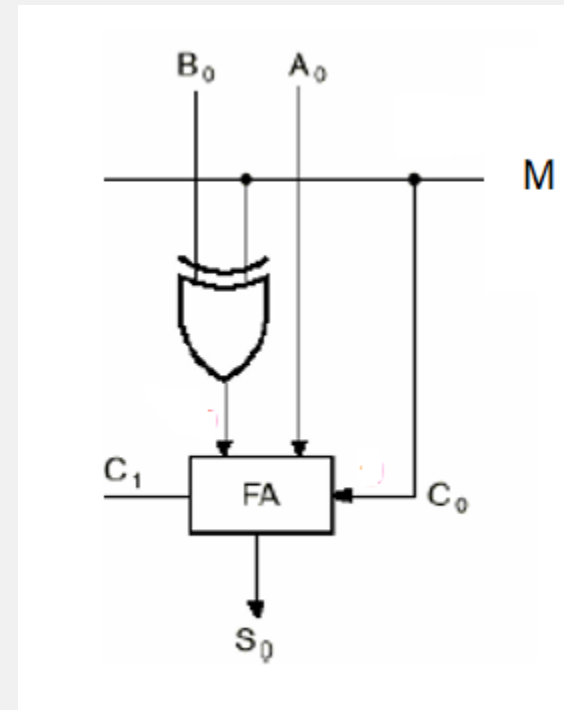
M	B	OUTPUT
0	0	0
0	1	1
1	0	1
1	1	0

B

B'

Here XOR act as a **programmable inverter**:

- 1- when $M=0$ the input is kept the same
- 2- when $M=1$ the input is inverted or complemented

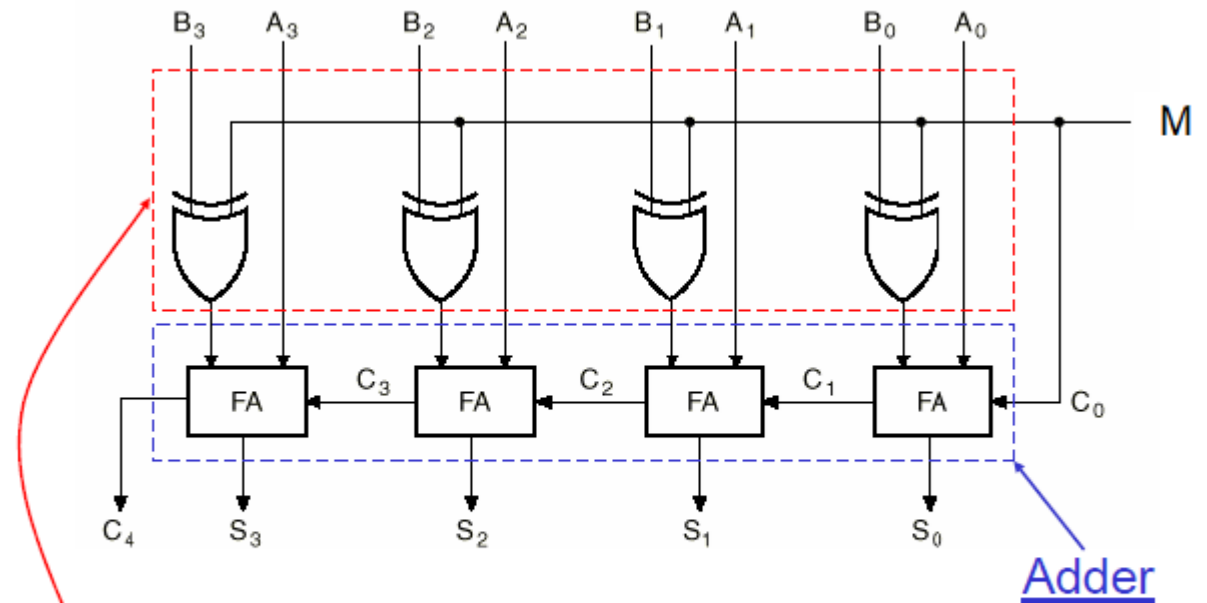


Combinational Circuits

4-bit Binary Adder-Subtractor using 2's Complement

When $M = 0$, the circuit performs $A + B$.
The carry in is 0, and the XOR gates simply pass B untouched.

When $M = 1$, the circuit performs $A - B$.
 $A - B = A + 2\text{'s comp. of } (B) = A + (B' + 1)$



Selective complementer:
XOR gates act as programmable inverters

Combinational Circuits

Addition cases and overflow

00	01	11	10	00	11
0010	0011	1110	1101	0010	1110
0011	0110	1101	1010	1100	0100
-----	-----	-----	-----	-----	-----
0101	1001	1011	0111	1110	0010

2	3	-2	-3	2	-2
3	6	-3	-6	-4	4
5	-7	-5	7	-2	2

OFL

OFL

Combinational Circuits

Addition cases and overflow

As a conclusion:

- If the last two carries are Different, then there is an overflow, and the sum result is incorrect
- If the last two carries are the same, then there is no overflow and the sum result is correct

End of Chapter 6