جامعة المستقبل
AL MUSTAQBAL UNIVERSITY
كلية الهندسة والتقنيات الهندسية

Communication Technical Engineering Department

1st Stage

Digital Logic- UOMU028021

Lecture 6 – Half-adder & Full-adder

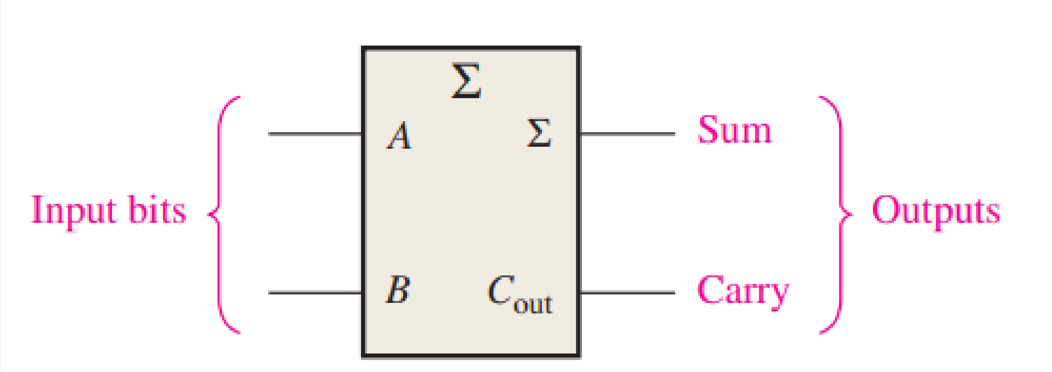
Dr. Mohammed Fadhil
PhD in Computer Networks
Email: mohammed.fadhil1@uomus.edu.iq

Half-adder & Full-adder

- **Adders** are important in computers and also in other types of digital systems in which numerical data are processed.
- A **half-adder** adds **two bits** and produces a *sum and an output carry.*
- The operations are performed by a logic circuit called a **half-adder.**

$$0 + 0 = 0$$
$$0 + 1 = 1$$
$$1 + 0 = 1$$
$$1 + 1 = 10$$

- **The half-adder accepts two binary digits on its inputs and produces two binary digits on its outputs—a sum bit and a carry bit**.
- A half-adder is represented by the logic symbol in the following Figure



Half-Adder Logic

- From the operation of the half-adder as stated in the Table, expressions can be derived for the sum and the output carry as functions of the inputs.
- Notice that the output carry (C_{out}) is a **1** only when **both A and B are 1s**; therefore, C_{out} can be expressed as the **AND** of the input variables.
- $C_{out} = AB$
- Now observe that the sum output $\sum$ is a 1 only if the input variables, A and B, are not equal. The sum can therefore be expressed as the exclusive-OR of the input variables.
- $\sum = A \oplus B$

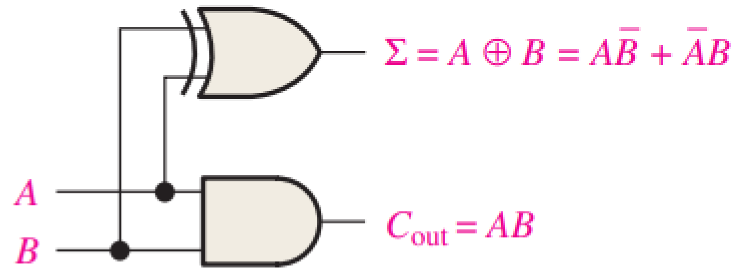
Half-adder truth table.

A	B	C_{out}	Σ
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Σ = sum
C_{out} = output carry
A and B = input variables (operands)

Half-Adder Logic

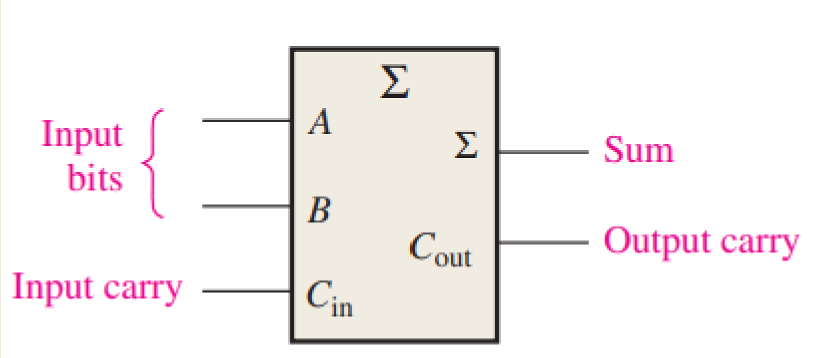
- The logic implementation required for the half-adder function can be developed. The **output carry** is produced with an **AND** gate with **A and B** on the inputs, and the sum output is generated with an **exclusive-OR gate**, as shown in the below Figure.
- Remember that the **exclusive-OR** can be implemented with AND gates, an OR gate, and inverters.



Half-adder logic diagram.

The Full-Adder

- The second category of adder is the full-adder.
- The full-adder accepts **two input bits** and an **input carry** and generates a **sum output and an output carry**.
- **A full-adder has an input carry while the half-adder does not.**
- A logic symbol for a full-adder is shown in Figure



Full-adder Truth Table

A	B	C_{in}	C_{out}	Σ
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

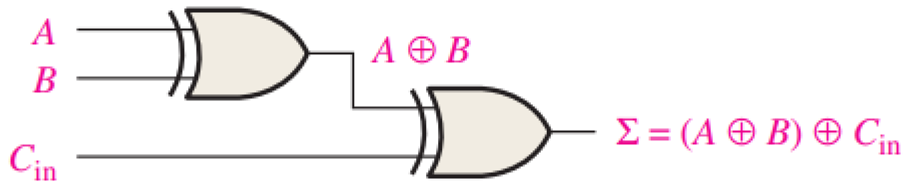
C_{in} = input carry, sometimes designated as CI
C_{out} = output carry, sometimes designated as CO
Σ = sum
A and B = input variables (operands)

Full-Adder Logic

- The full-adder must add the two input bits and the input carry. From the half-adder you know that the sum of the input bits A and B is the exclusive-OR of those two variables, $\boldsymbol{A \oplus B}$.
- For the input carry ($\boldsymbol{C_{in}}$) to be added to the input bits, it must be exclusive-ORed with $\boldsymbol{A \oplus B}$, yielding the equation for the sum output of the full-adder.
- $\sum \quad = (\boldsymbol{A \oplus B}) \oplus \boldsymbol{C_{in}}$

The Full-Adder

- This means that to implement the full-adder sum function, two 2-input exclusive-OR gates can be used.
- The first must generate the term $\boldsymbol{A \oplus B}$, and the second has as its inputs the output of the first XOR gate and the input carry, as illustrated in Figure



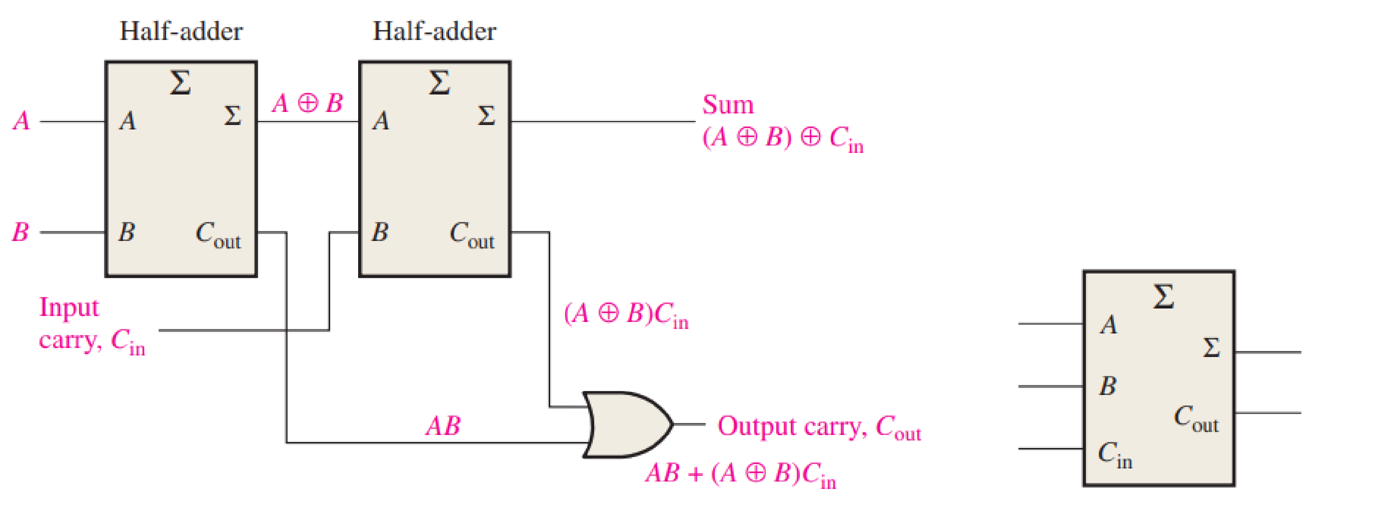
(a) Logic required to form the sum of three bits

The Full-Adder

- The output carry is a 1 when both inputs to the first XOR gate are 1s or when both inputs to the second XOR gate are 1s. You can verify this fact by studying full adder truth Table.
- The output carry of the full-adder is therefore produced by input A ANDed with input B and $\boldsymbol{A \oplus B}$ ANDed with $\boldsymbol{C_{in}}$.
- These two terms are ORed, as expressed in Equation:
$\boldsymbol{C_{out}} = \boldsymbol{AB} + (\boldsymbol{A \oplus B})\, \boldsymbol{C_{in}}$
- This function is implemented and combined with the sum logic to form a complete full-adder circuit.

The Full-Adder

- There are two half-adders, connected as shown in the block diagram of Figure (a), with their output carries ORed. The logic symbol shown in Figure (b) will normally be used to represent the full-adder.

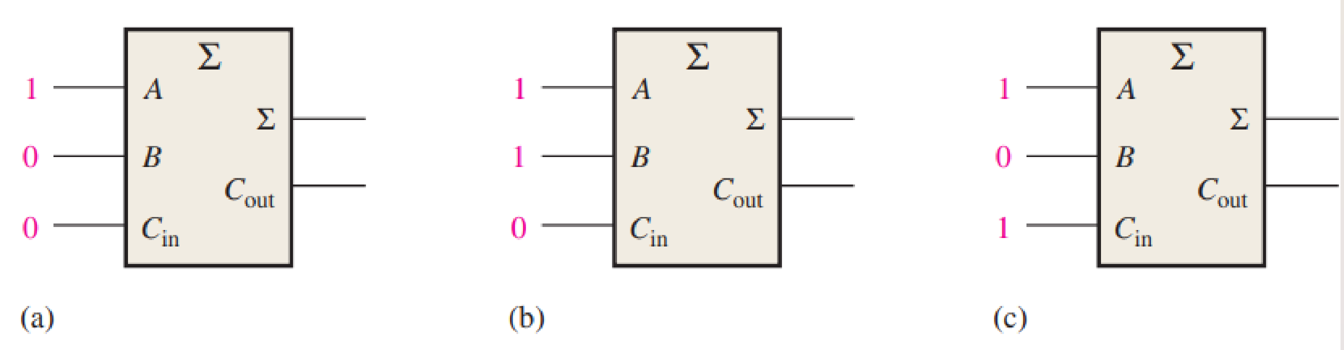


(a) Arrangement of two half-adders to form a full-adder

(b) Full-adder logic symbol

The Full-Adder

- For each of the three full-adders in Figure a,b,c, determine the outputs for the inputs shown.



- **Solution**
  - **(a)** The input bits are **A = 1, B = 0, and C_{in} = 0.**

    1 + 0 + 0 = 1 with no carry

    Therefore, $\sum$ = 1 and C_{out} = 0.
  - **(b)** The input bits are **A = 1, B = 1, and C_{in} = 0.**

    1 + 1 + 0 = 0 with a carry of 1

    Therefore, $\sum$ **= 0 and C_{out} = 1.**
  - **(c)** The input bits are **A = 1, B = 0, and Cin = 1.**

    1 + 0 + 1 = 0 with a carry of 1

    Therefore, $\sum$ **= 0 and C_{out} = 1.**

Checkup!

- Determine the sum ($\sum$) and the output carry (C_{out}) of a half-adder for each set of input bits:
  - **(a)** 01 **(b)** 00 **(c)** 10 **(d)** 11

- A full-adder has C_{in} = 1. What are the sum ($\sum$) and the output carry (C_{out}) when **A** = 1 and **B = 1**?

Solution

- **Solving Part (a): Input Bits 01**
  - **Inputs**: A = 0, B = 1
  - **Calculate Sum (∑):**
    - **A XOR B = 0 XOR 1**
    - **0 XOR 1 = 1**
  - **Calculate Carry-out (C_{out}):**
    - **A AND B = 0 AND 1**
    - **0 AND 1 = 0**
- **Answer**:
  - **Sum (∑) = 1**
  - **C_{out} = 0**

A full-adder has C_{in} = 1. What are the sum ($\sum$) and the output carry (C_{out}) when A = 1 and B = 1?

- **Solving the Full-Adder Problem**
  - **Given:**
    - A = 1 , B = 1 , C_in = 1
  - **Step 1: Calculate A ⊕ B:**
    - A ⊕ B = 1 ⊕ 1 = 0
  - **Step 2: Plug into the formula**
    - Cout=AB+(A⊕B)·Cin
      =(1·1)+(0·1)
      =1+0=1
  - **Sum (Σ) is still:**
    - Σ = A ⊕ B ⊕ Cin
      =1 ⊕ 1 ⊕ 1
      =0 ⊕ 1 = 1
- **Final Answer:**
  - Sum (∑) = 1
  - C_out = 1

THANK YOU ☺