ADDER AND SUBTRACTOR

INTRODUCTION

OBJECTIVE:

- HALF ADDER
- FULL ADDER
- BINARY PARALLEL ADDER
- HALF SUBTRACTOR
- FULL SUBTRACTOR

What is Adder?

- **Adder** : In electronics an adder is digital circuit that perform addition of numbers. In modern computer adder reside in the arithmetic logic unit (ALU).

Adders :

- Adders are important not only in the computer but also in many types of digital systems in which the numeric data are processed.

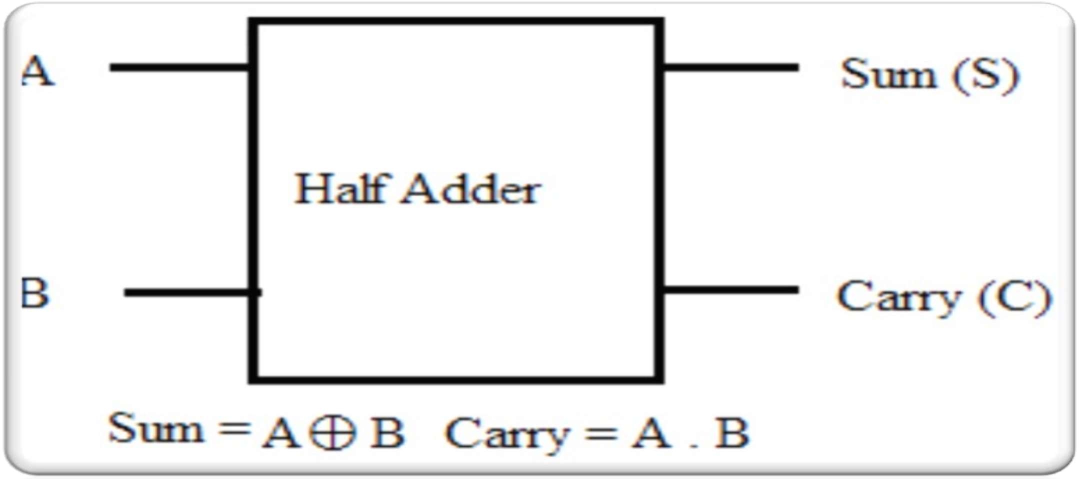
- **Types of adder:**

- **Half adder**

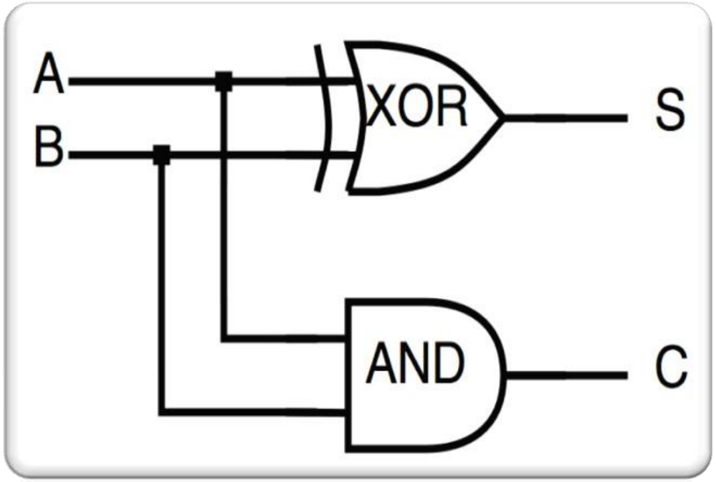
- Full adder

- **Half adder :** The half adder accepts two binary digits on its inputs and produce two binary digits outputs, a sum bit and a carry bit.

- The half adder is an example of a simple, functional digital circuit built from two logic gates. The half adder adds to one-bit binary numbers (AB). The output is the sum of the two bits (S) and the carry (C).



- Note that how the same two inputs are directed to two different gates. The inputs to the XOR gate are also the inputs to the AND gate. The input "wires" to the XOR gate are tied to the input wires of the AND gate; thus, when voltage is applied to the A input of the XOR gate, the A input to the AND gate receives the same voltage.



Half adder truth table

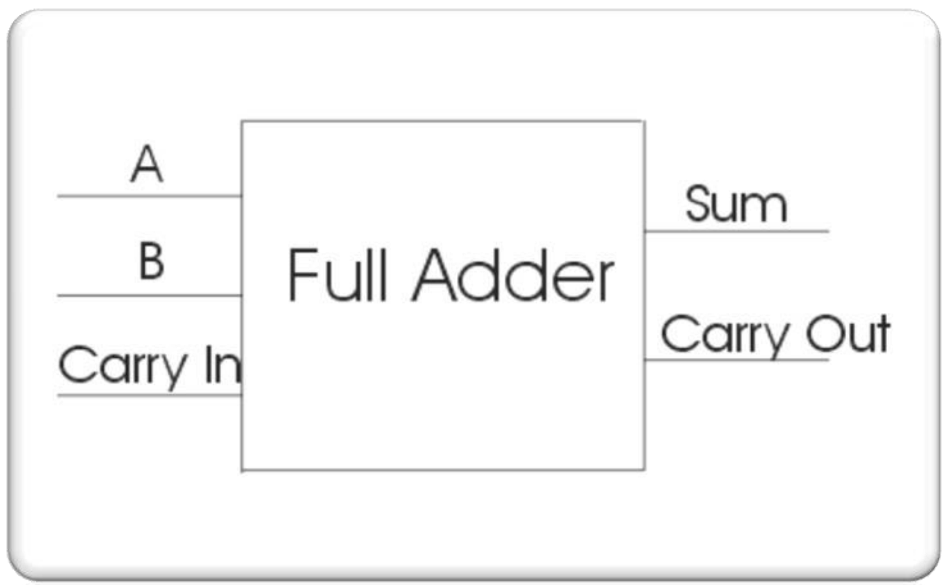
A	B	Sum	Carry-Out
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

$S=A \oplus B$ (Exclusive OR)

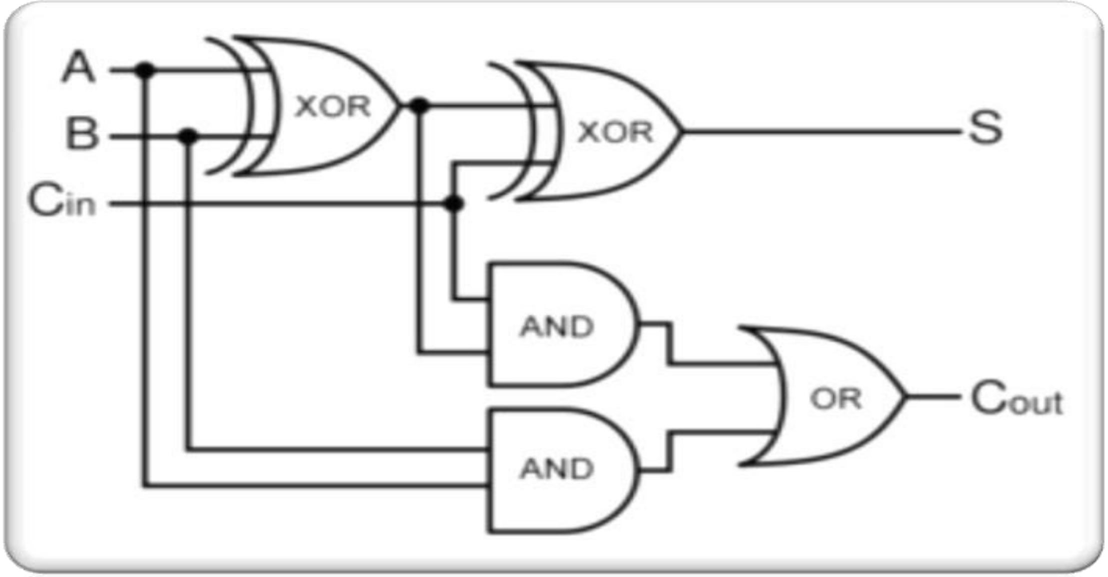
$C=A.B$ (AND)

- **Full adder :** The full adder accepts two inputs bits and an input carry and generates a sum output and an output carry.

- The full-adder circuit adds three one-bit binary numbers (Cin, A ,B) and outputs two one-bit binary numbers, a sum (S) and a carry (Cout). The full-adder is usually a component in a cascade of adders, which add 8, 16, 32, etc. binary numbers.



- If you look closely, you'll see the full adder is simply two half adders joined by an OR.
- We can implement a full adder circuit with the help of two half adder circuits. The first half adder will be used to add A and B to produce a partial Sum. The second half adder logic can be used to add CIN to the Sum produced by the first half adder to get the final S output. If any of the half adder logic produces a carry, there will be an output carry. Thus, COUT will be an OR function of the half-adder Carry outputs.



Full adder truth table

$S = A \oplus B \oplus Cin$

$C = AB + Cin\,(A \oplus B)$

A	B	Carry-In	Sum	Carry-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

What is the difference between half adder and a full adder circuit?

- The main difference between a half-adder and a full-adder is that the full-adder has three inputs and two outputs. The first two inputs are A and B and the third input is an input carry designated as CIN. When a full adder logic is designed we will be able to string eight of them together to create a byte-wide adder and cascade the carry bit from one adder to the next.
- The output carry is designated as COUT and the normal output is designated as S.

BINARY PARALLEL ADDER

Parallel Adder

- A circuit , consisting of n full adders , that will add n-bit binary numbers.
- The output consists of n sum bits and a carry bit.
- Cout of one full adder is connected to Cin of the next full adder.

STRUCTURE OF PARALLEL ADDER

- Parallel adder nothing but a cascade of several full adders.
- The number of full adders used will depend on the number of bits in the binary digits which require to be added.

BLOCK DIAGRAM OF N-bit BINARY PARALLEL ADDER

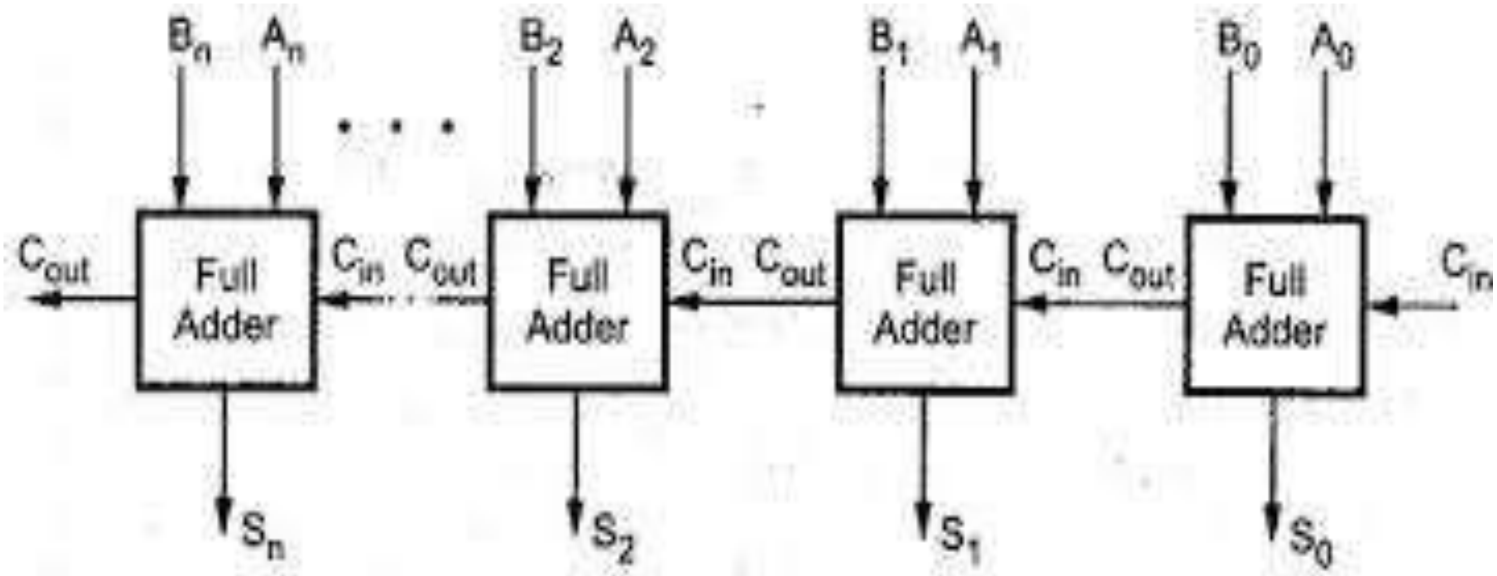


Fig. 3.25 Block diagram of n-bit parallel adder

When an n-bit binary number is added to another , each column generates a sum and a 1 or 0 carry to the next higher order column.

Procedure

- The bits are added with full-adder.
- Starting from the least significant position to form the sum and carry.
- The input carry CI1 in the least significant position must be zero.
- The value of CIi+1 in a given significant position is the output carry CO of the full adder.
- This value is transferred into the input carry CI of the full-adder that adds the bits one higher significant position to left.
- The sum bits are thus generated starting from the rightmost position and are available as soon as the corresponding previous carry bit is generated.

For an example:

A = 1011

B = 0011

S = 1110

Demonstration:

subscript	4	3	2	1	
Input carry	0	1	1	0	Ci
Augend (A)	1	0	1	1	Ai
Addend (B)	0	0	1	1	Bi
Sum	1	1	1	0	Si
Output carry	0	0	1	1	Ci+1

Carry propagation delay

- The sum and the output carry of any stage cannot be produced until the input carry occurs ,this causes a time delay, called ***the carry propagation delay***

•The carry propagation delay for each full-adder is the time from the application of the input carry until the output carry occurs, assuming that the A and B inputs are already present.

SUBTRACTOR

❖ What is Subtractor ?

Subtractor is an electronic logic circuit for calculating the difference between two binary numbers which provides the difference and borrow as output.

❖ Types of Subtractor

- Half Subtractor
- Full Subtractor

➢ Half subtractor

Half Subtractor is used for subtracting one single bit binary number from another single bit binary number.

It has two inputs; Minuend (A) and Subtrahend (B) and two outputs; Difference (D) and Borrow (B_{out}).

Truth Table

Input		Output	
A	B	Difference (D)	Borrow (B_{out})
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

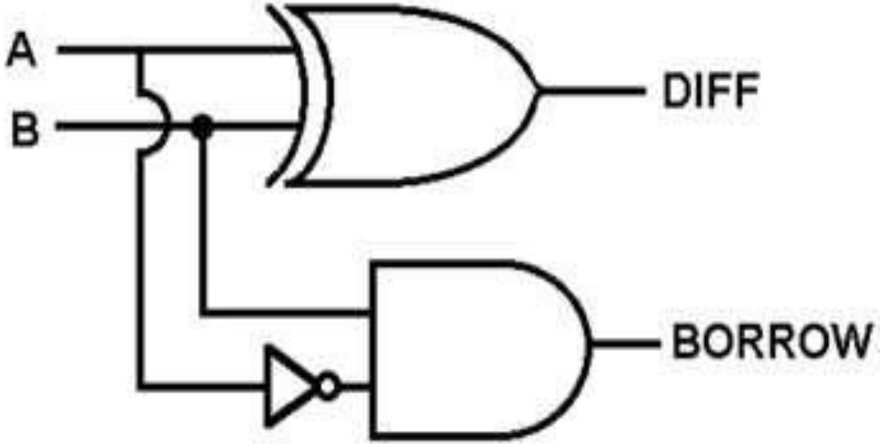
❑ Boolean Expression

From the truth table and K-map, the Boolean Expression can be derived as:

Difference (D) = $\bar{A}.B + A.\bar{B} = A \oplus B$

Borrow $(B_{out}) = \bar{A}.B$

❑ Logical Circuit



➢ FULL Subtractor

- A logic Circuit Which is used for subtracting three single bit binary numbers is known as Full Subtractor.
- It has three inputs; Minuend (A), Subtrahend (B) and following Subtrahend (C) and two outputs; Difference (D) and Borrow (B_{out}).

Truth Table

Input			Output	
A	B	C	D	$B_{(out)}$
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

❑ BOOLEAN EXPRESSION

From the Truth Table The Difference and Borrow will written as,

Difference=A'B'C+A'BC'+AB'C'+ABC

Reducing it we got,

Difference=A $\oplus$B $\oplus$C

Borrow=A'B'C+A'BC'+A'BC+ABC

=A'B'C+A'BC'+A'BC+A'BC+A'BC+ABC

=A'C(B'+B)+A'B(C'+C)+BC(A'+A)

Borrow=A'C+A'B+BC

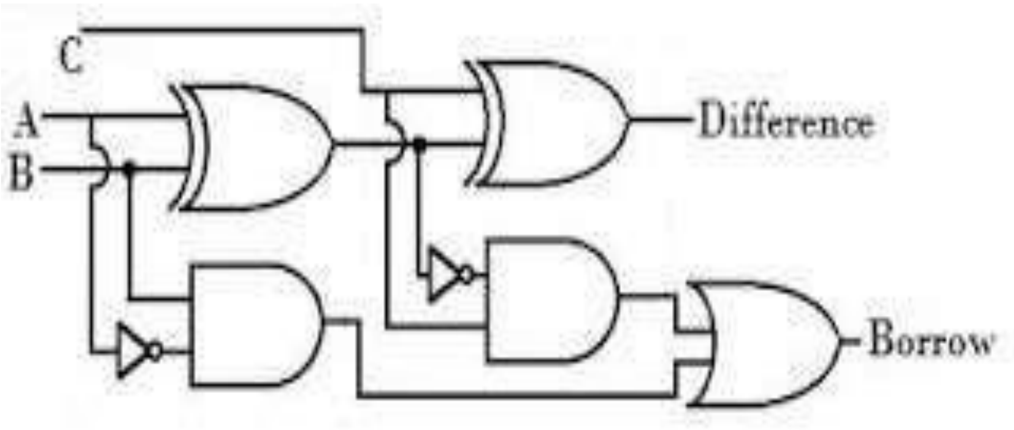
❑ Boolean Expression

From the truth table and k-map minimization, the Boolean Expression can be derived as:

$$D = A \oplus B \oplus C$$

$$B_{(out)} = BC + (B \oplus C) A$$

❑ Logical Circuit



Any question?