ADDER & SUBTRACTOR

Adder

- An adder is a digital logic circuit in electronics that implements addition of numbers.
- In many computers and other kinds of processors, adders are used not only in the arithmetic logic units, but also in other parts of the processor, where they are used to calculate addresses, increment and decrement operators, and similar operations.
- Adders are classified into two types: 1)half adder.
  2) full adder.

Let us first take a look at the addition of single bits.

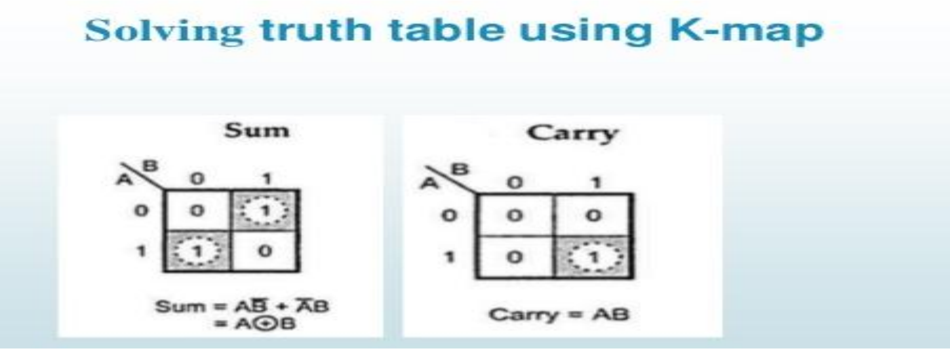
- 0+0 = 0
- 0+1 = 1
- 1+0 = 1
- 1+1 =10 (i.e. 1+1=0 with carry = 1)

Half Adder

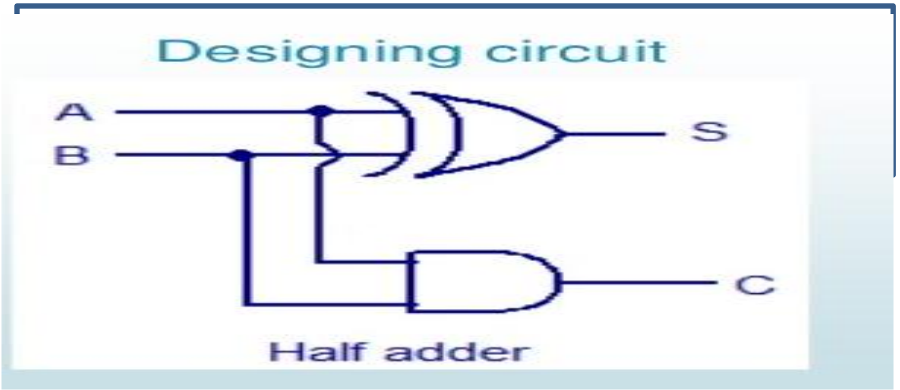
- The half adder adds two single binary digits *A* and *B*.
- It has two outputs, sum (*S*) and carry (*C*).
- The carry signal represents an overflow into the next digit of a multi-digit addition.

Truth Table

INPUTS		OUTPUTS	
A	B	SUM	CARRY
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1



Solving truth table using K-map
Sum
Carry
Sum = AB̄ + ĀB
= A⊕B
Carry = AB



Designing circuit
A
B
S
C
Half adder

Full Adder

- A full adder adds binary numbers and accounts for values carried in as well as out.
- The main difference between a half-adder and a full-adder is that the full-adder has three inputs and two outputs.
- A one-bit full adder adds three one-bit numbers, often written as A, B, and C_{in}.
- It has two outputs, sum (S) and carry (C_{out}).

Truth Table

INPUTS			OUTPUTS	
A	B	CIN	COUT	Sum
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

Solving Truth Table using K-Map

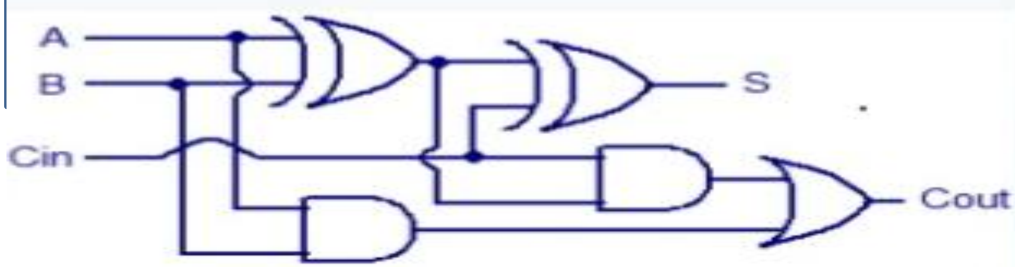
K-map for Sum (S)

$S = A \oplus B \oplus C_{in}$

K-map for Carry (C_{out})

$C_{out} = (A \cdot B) + (C_{in} \cdot (A \oplus B))$

Designing circuit



Full adder

Subtractor

- An Subtractor is a digital logic circuit in electronics that implements subtraction of numbers.
- In many computers and other kinds of processors, Subtractor are used not only in the arithmetic logic units, but also in other parts of the processor, where they are used to calculate addresses, increment and decrement operators, and similar operations.
- Substractor are classified into two types: 1)half Subtractor.
  2) full Subtractor.

Let us first take a look at the subtraction of single bits.

- 0-0 = 0
- 0-1 = 11 (i.e. 0-1 = 1 with borrow = 1)
- 1-0 = 1
- 1-1 = 0

Half Subtractor

- The half Subtractor subtracts two single binary digits *A* and *B*.
- It has two outputs, Difference (*D*) and borrow (*B*).
- The borrow signal represents an overflow into the next digit of a multi-digit subtraction.

Truth Table

INPUTS		OUTPUTS	
A	B	DIFF	BORROW
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

Solving truth table using K-map

B \ A	0	1
0	0	0
1	1	0

For Borrow

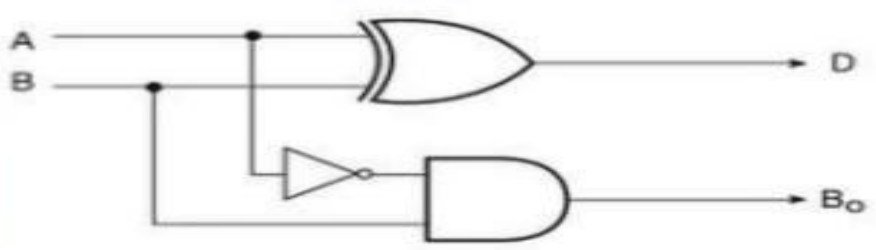
Borrow = $\bar{A}.B$

B \ A	0	1
0	0	1
1	1	0

For Difference

Difference = $A \oplus B$

Designing circuit



Full Subtractor

- A full Subtractor subtracts binary numbers and accounts for values borrowed in as well as out.
- The main difference between a half- Subtractor and a full- Subtractor is that the full- Subtractor has three inputs and two outputs.
- A one-bit full Subtractor subtracts three one-bit numbers, often written as A, B, and B_{in}.
- It has two outputs, Difference (D) and borrow (B).

Truth Table

INPUTS			OUTPUTS	
A	B	BIN	BOUT	Difference
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	0
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Solving Truth Table using K-Map

A \ BC	00	01	11	10
0	0	1	1	1
1	0	0	1	0

For Borrow

A \ BC	00	01	11	10
0	0	1	0	1
1	1	0	1	0

For Difference

Designing circuit