**Designing Combinatorial Circuits**

The design of a combinational circuit starts from the verbal outline of the problem and ends with a logic circuit diagram or a set of Boolean functions from which the Boolean function can be easily obtained. The procedure involves the following steps:

- The problem is stated

- The number of available input variables and required output variables is determined.

- The input and output variable are assigned their letter symbol

- The truth table that defines the required relationship between the inputs and the outputs is derived.

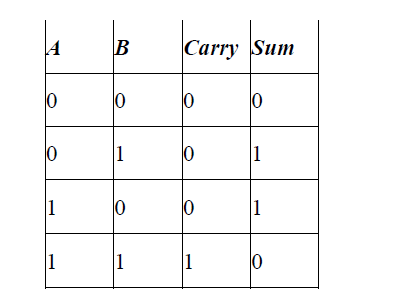
- The simplified Boolean function for each output is obtained

- The logic diagram is drawn.

**Example of combinational circuit**

**Adders**

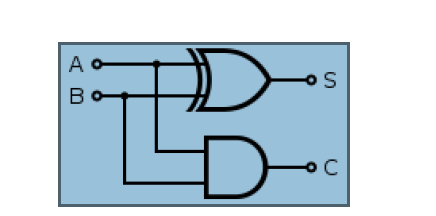
In electronics, an adder or summer is a digital circuit that performs addition of numbers. In modern computers adders reside in the arithmetic logic unit (ALU) where other operations are performed. Although adders can be constructed for many numerical representations, such as Binary-coded decimal or excess-3, the most common adders operate on binary numbers. In cases where twos complement or ones complement is being used to represent negative numbers, it is trivial to modify an adder into an adder-subtracter. Other signed number representations require a more complex adder. **-Half Adder** A half adder is a logical circuit that performs an addition operation on two binary digits. The half adder produces a sum and a carry value which are both binary digits. A half adder has two inputs, generally labelled A and B, and two outputs, the sum S and carry C. S is the two-bit XOR of A and B, and C is the AND of A and B. Essentially the output of a half adder is the sum of two one-bit numbers, with C being the most significant of these two outputs. The drawback of this circuit is that in case of a multibit addition, it cannot include a carry. Following is the truth table for a half adder:



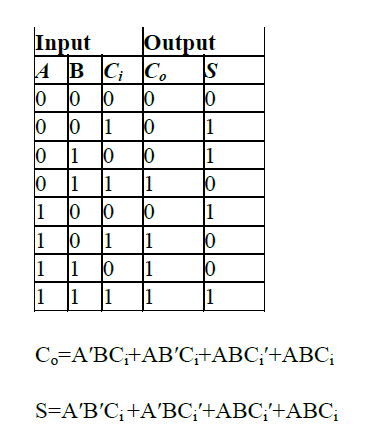
Equation of the Sum and Carry. Sum=A′B+AB′ Carry=AB

One can see that Sum can also be implemented using XOR gate as

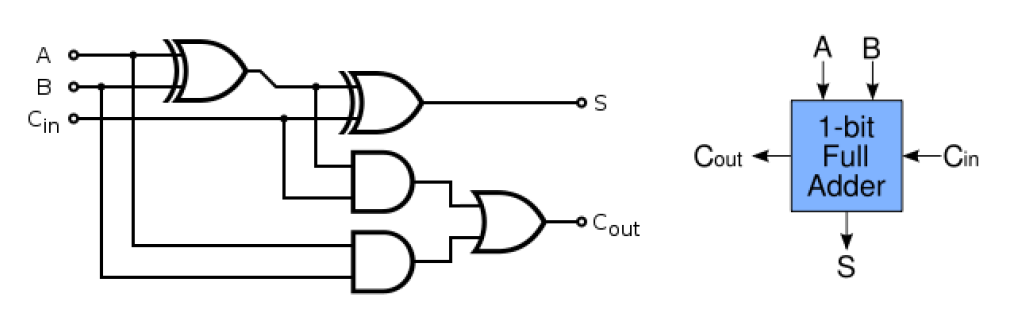




**-Full Adder.** A full adder has three inputs *A*, *B*, and a carry in *C*, such that multiple adders can be used to add larger numbers. To remove ambiguity between the input and output carry lines, the carry in is labelled *Ci* or *Cin* while the carry out is labelled *Co* or *Cout*. A full adder is a logical circuit that performs an addition operation on three binary digits. The full adder produces a sum and carry value, which are both binary digits. It can be combined with other full adders or work on its own.

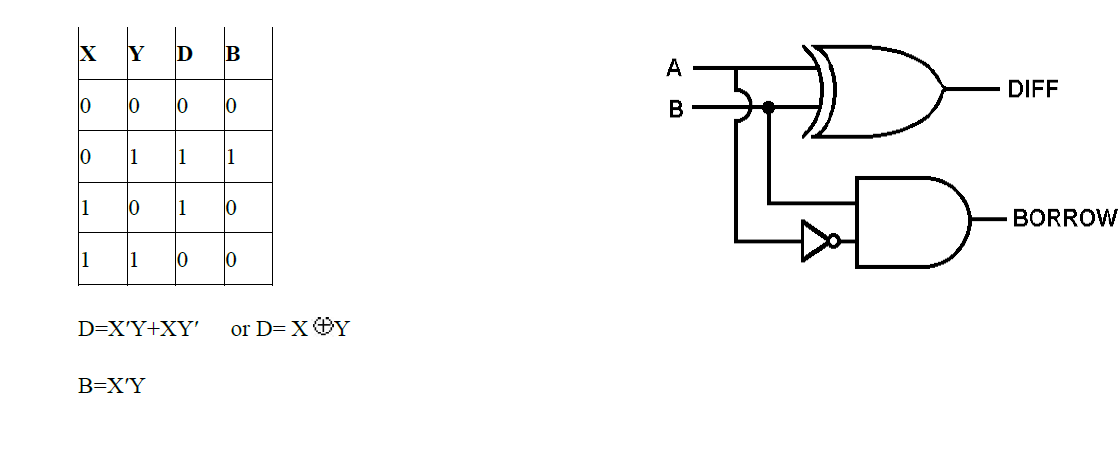


A full adder can be constructed from two half adders by connecting A and B to the input of one half adder, connecting the sum from that to an input to the second adder, connecting Ci to the other input and OR the two carry outputs. Equivalently, S could be made the three-bit xor of A, B, and Ci and Co could be made the three-bit majority function of A, B, and Ci. The output of the full adder is the two-bit arithmetic sum of three one-bit numbers.

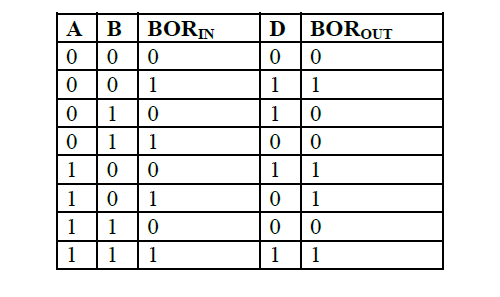


**Subtractor**

In electronics, a subtractor can be designed using the same approach as that of an adder. The binary subtraction process is summarized below. As with an adder, in the general case of calculations on multi-bit numbers, three bits are involved in performing the subtraction for each bit: the minuend (Xi), subtrahend (Yi), and a borrow in from the previous (less significant) bit order position (Bi). The outputs are the difference bit (Di) and borrow bit Bi + 1. **Half subtractor** The half-subtractor is a combinational circuit which is used to perform subtraction of two bits. It has two inputs, X (minuend) and Y (subtrahend) and two outputs D (difference) and B (borrow). Such a circuit is called a half-subtractor because it enables a borrow out of the current arithmetic operation but no borrow in from a previous arithmetic operation. The truth table for the half subtractor is given below.



**Full Subtractor** As in the case of the addition using logic gates , a *full subtractor* is made by combining two half-subtractors and an additional OR-gate. A full subtractor has the borrow in capability (denoted as **BORIN** in the diagram below) and so allows *cascading* which results in the possibility of **multi-bit subtraction**. The final truth table for a full subtractor looks like:



Find out the equations of the borrow and the difference The circuit diagram for a full subtractor is given below.

