# SNS COLLEGE OF ENGINEERING

Kurumbapalayam (Po), Coimbatore – 641 107

**An Autonomous Institution**

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

## DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

COURSE NAME : 19EC306 – Digital Circuits

II YEAR / III SEMESTER

Unit I- **COMBINATIONAL CIRCUITS**

Topic : Design procedure – Half adder

# Design procedure – Half adder

A half adder is a digital logic circuit that performs binary addition of two single-bit binary numbers. It has two inputs, A and B, and two outputs, SUM and CARRY. The SUM output is the least significant bit (LSB) of the result, while the CARRY output is the most significant bit (MSB) of the result, indicating whether there was a carry-over from the addition of the two inputs. The half adder can be implemented using basic gates such as XOR and AND gates.

Sure, here's a more in-depth explanation of the half adder circuit:
The half adder is a basic building block for more complex adder circuits such as full adders and multiple-bit adders. It performs binary addition of two single-bit inputs, A and B, and provides two outputs, SUM and CARRY.
The SUM output is the least significant bit (LSB) of the result, which is the XOR of the two inputs A and B. The XOR gate implements the addition operation for binary digits, where a "1" is generated in the SUM output only when one of the inputs is "1".
The CARRY output is the most significant bit (MSB) of the result, indicating whether there was a carry-over from the addition of the two inputs. The CARRY output is the AND of the two inputs A and B. The AND gate generates a "1" in the CARRY output only when both inputs are "1".

**Half Adder (HA):**

Half adder is the simplest of all adder circuits. Half adder is a combinational arithmetic circuit that adds two numbers and produces a sum bit (s) and carry bit (c) both as output. The addition of 2 bits is done using a combination circuit called a Half adder. The input variables are augend and addend bits and output variables are sum & carry bits. A and B are the two input bits.

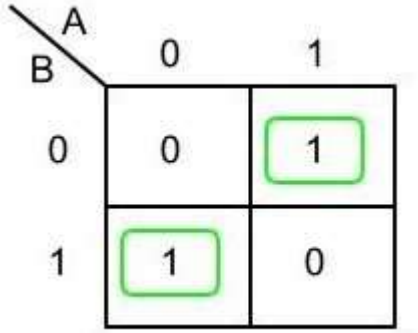
let us consider two input bits A and B, then sum bit (s) is the X-OR of A and B. it is evident from the function of a half adder that it requires one X-OR gate and one AND gate for its construction.

**Truth Table:**

| A | B | Sum | Carry |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |

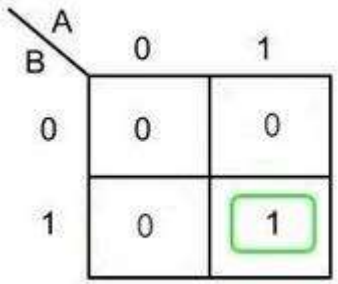
**Logical Expression:**

**For Sum:**



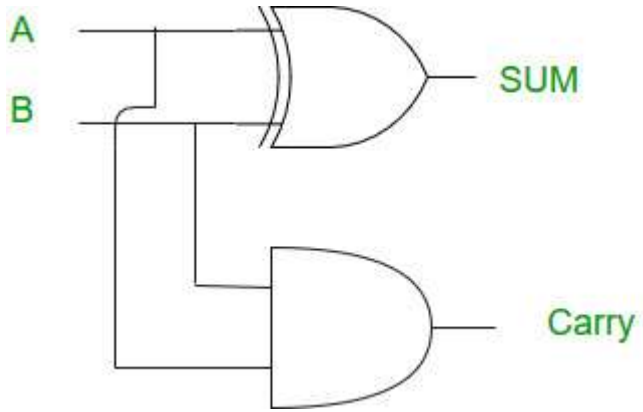

**Sum = A XOR B**

**For Carry:**




**Carry = A AND B**

**Implementation:**




**Note:** Half adder has only two inputs and there is no provision to add a carry coming from the lower order bits when multi addition is performed.

*Advantages and disadvantages of Half Adder in Digital Logic :*

**Advantages** of Half Adder in Digital Logic :

Simplicity, Speed

**Disadvantages** of Half Adder in Digital Logic :

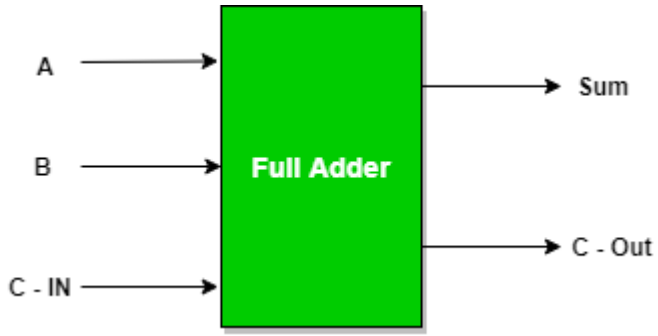
Limited Usefulness, Lack of Convey Info, Propagation Deferral

**Application** of Half Adder in Digital Logic

Arithmetic circuits, Data handling, Address unraveling, Encoder and decoder circuits, Multiplexers and demultiplexers, Counters

# Full Adder

- Full Adder is the adder that adds three inputs and produces two outputs. The first two inputs are A and B and the third input is an input carry as C-IN. The output carry is designated as C-OUT and the normal output is designated as S which is SUM. The C-OUT is also known as the majority 1's detector, whose output goes high when more than one input is high. A full adder logic is designed in such a manner that can take eight inputs together to create a byte-wide adder and cascade the carry bit from one adder to another. we use a full adder because when a carry-in bit is available, another 1-bit adder must be used since a 1-bit half-adder does not take a carry-in bit. A 1-bit full adder adds three operands and generates 2-bit results.



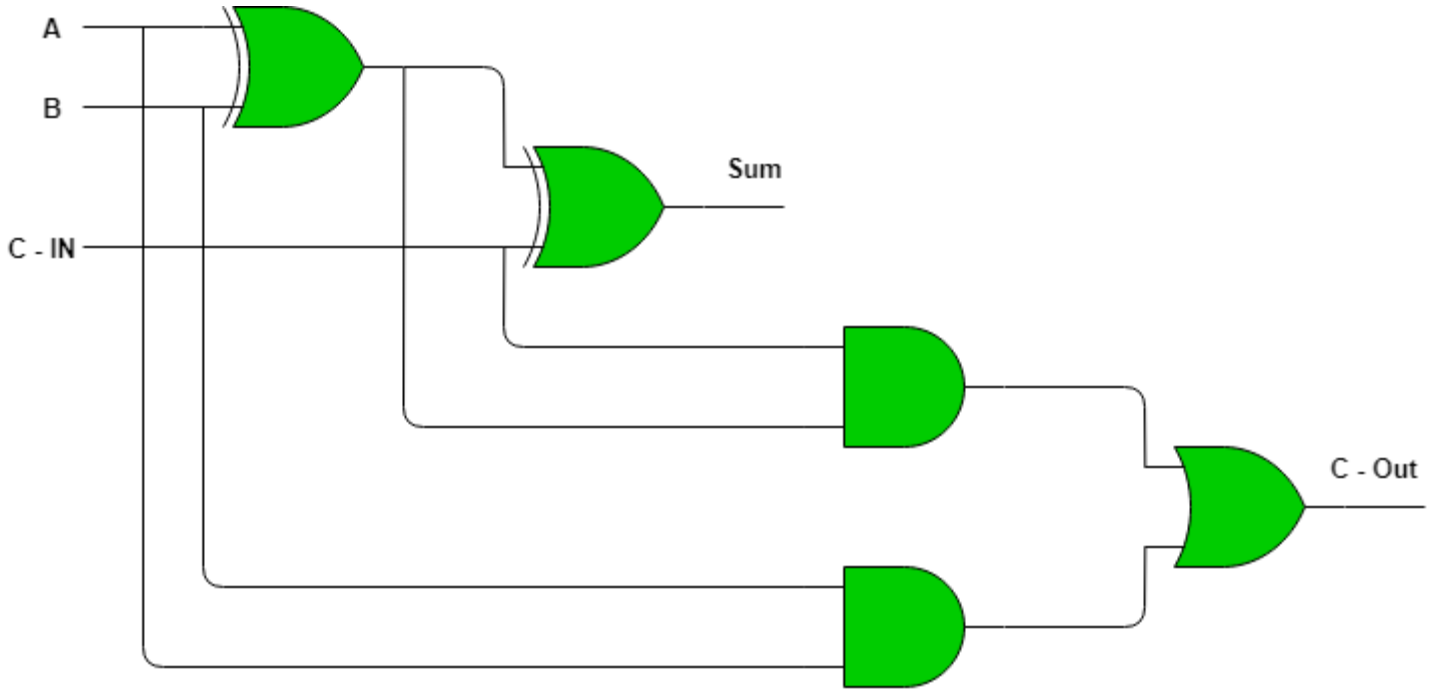
## Adder Truth Table:

| Inputs | | | Outputs | |
|---|---|---|---|---|
| A | B | C – IN | Sum | C - 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 |

**Logical Expression for SUM:** = A’ B’ C-IN + A’ B C-IN’ + A B’ C-IN’ + A B C-IN = C-IN (A’ B’ + A B) + C-IN’ (A’ B + A B’) = C-IN XOR (A XOR B) = (1,2,4,7)

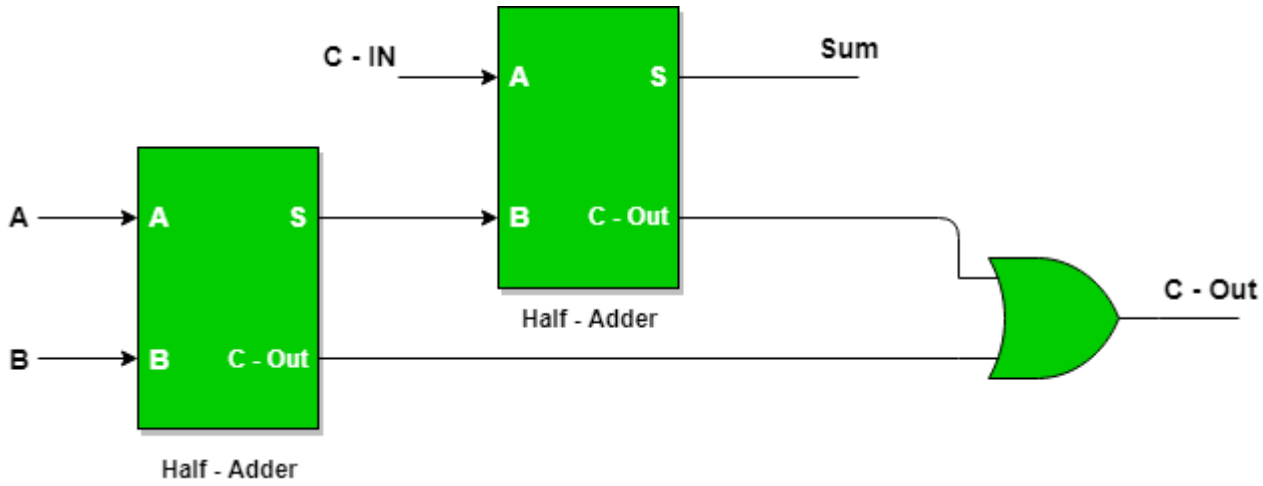
**Logical Expression for C-OUT:** = A’ B C-IN + A B’ C-IN + A B C-IN’ + A B C-IN = A B + B C-IN + A C-IN = (3,5,6,7)

**Another form in which C-OUT can be implemented:** = A B + A C-IN + B C-IN (A + A') = A B C-IN + A B + A C-IN + A' B C-IN = A B (1 +C-IN) + A C-IN + A' B C-IN = A B + A C-IN + A' B C-IN = A B + A C-IN (B + B') + A' B C-IN = A B C-IN + A B + A B' C-IN + A' B C-IN = A B (C-IN + 1) + A B' C-IN + A' B C-IN = A B + A B' C-IN + A' B C-IN = AB + C-IN (A' B + A B')
Therefore COUT = AB + C-IN (A EX – OR B)



**Implementation of Full Adder using Half Adders:**

2 Half Adders and an OR gate is required to implement a Full Adder.




With this logic circuit, two bits can be added together, taking a carry from the next lower order of magnitude, and sending a carry to the next higher order of magnitude.

Any Query????

Thank you……