Department of Computer Science and Engineering
Pandit Deendayal Energy University
Digital Electronics and Computer Organization Lab – **20CP203P**

Lab 4: Design a Half Adder, and Full Adder using Half Adder

Question 1 Write Verilog code for half adder. Test using waveform.

Question 2. Design a four-bit combinational circuit 2's complement using exclusive-OR gates and half adder. Write Verilog code in Edaplayground and then test using waveform.

Question 3. Write Verilog code for full adder. Test using waveform.

Question 4. Write Verilog code for full adder using module instantiation of half adder. Test using waveform.

Question 5. Write Verilog code for half adder using only NAND gate. Test using waveform.

Question 6. Write Verilog code for half adder using only NOR gate. Test using waveform.

Description:

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

Table 4.1: Truth table for "Sum" and "Carry".

Here we perform two operations Sum and Carry, thus we need two K-maps one for each to derive the expression.

Logical Expression:

For Sum:

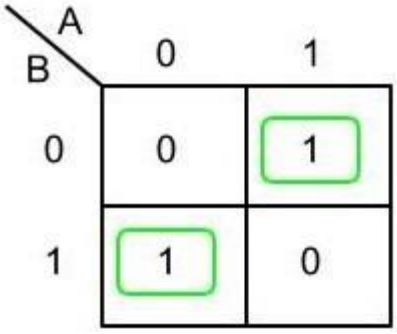

Fig 4.1: K-map for “Sum”

Sum = A XOR B

For Carry:

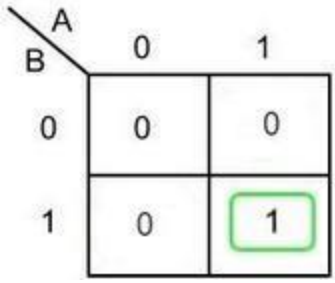

Fig 4.2: K-map for “Carry”

Carry = A AND B

Implementation:

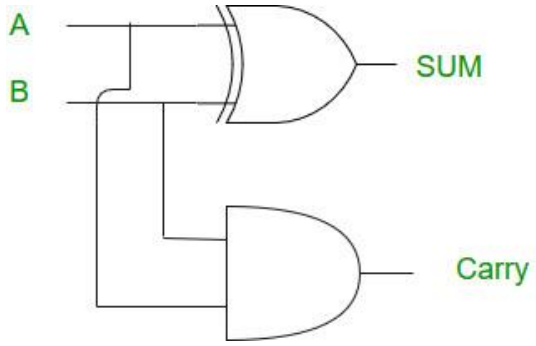

Fig 4.3: Circuit Diagram of "Half-Adder".

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. 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.

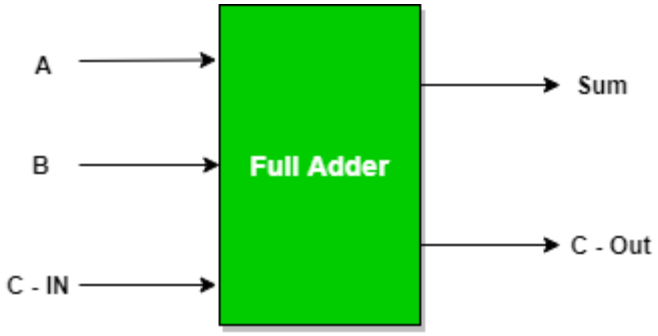

Fig 4.4: Black box view of "Full Adder".

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

Table 4.2: Truth table of "Full Adder"

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)

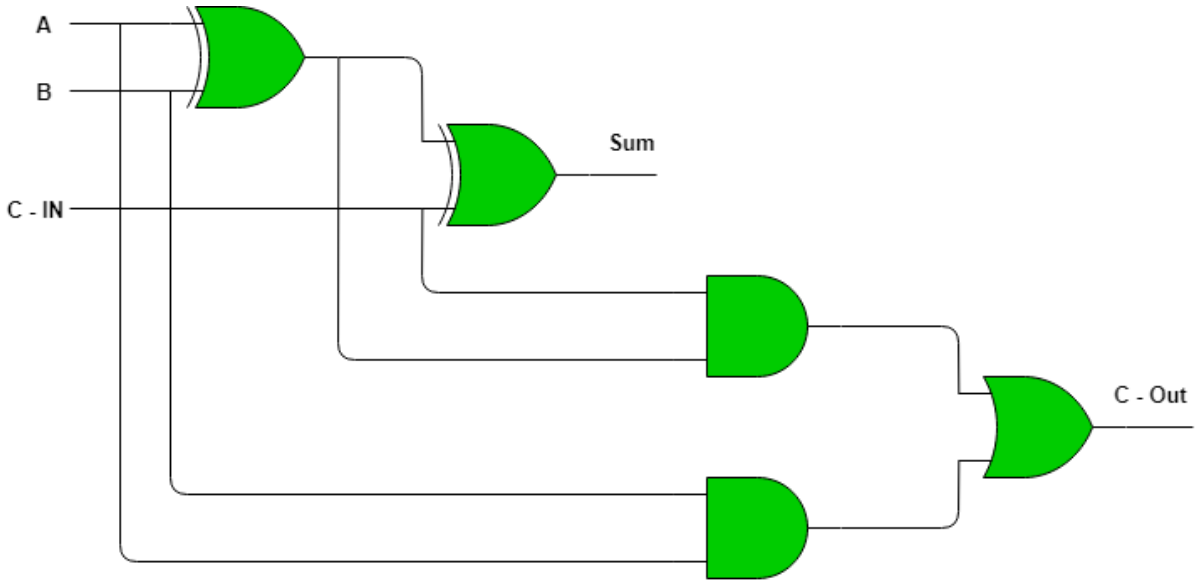

Fig 4.5: Circuit Diagram of "Full Adder"

Implementation of Full Adder using Half Adders:

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

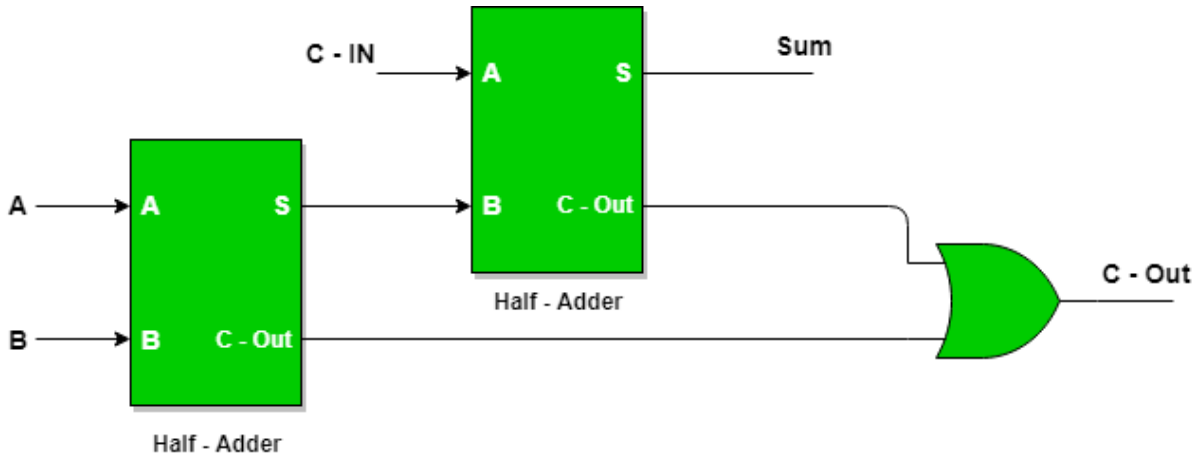

Fig 4.6: Circuit Diagram of “Full Adder” using two “Half-Adder”.

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.