

JK LAKSHMIPAT UNIVERSITY

DIGITAL CIRCUIT AND SYSTEMS  
(EE1120)

Activity 05

Half Adder and Full Adder using VHDL language.

Date : 20th February 2023

Name : Bobby Sharma

Roll no. = 2023BTECH023

AIM:Design and Simulation of Half adder and Full Adder using VHDL language using Xilinx ISETool.

SOFTWARE REQUIRED: Xilinx ISE tool in your device.

THEORY:

VHDL stands for very high-speed integrated circuit hardware description language. It is a programming language used to model a digital system by dataflow, behavioral and structural style of modeling. This language was first introduced in 1981 for the department of Defense (DoD) under the VHSIC program.

In this modeling style, the flow of data through the entity is expressed using concurrent (parallel) signal. The concurrent statements in VHDL are WHEN and GENERATE. Besides them, assignments using only operators (AND, NOT, +, \*, sll, etc.) can also be used to construct code. Finally, a special kind of assignment, called BLOCK, can also be employed in this kind of code.

In concurrent code, the following can be used −

* Operators
* The WHEN statement (WHEN/ELSE or WITH/SELECT/WHEN);
* The GENERATE statement;
* The BLOCK statement

**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.

**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.

OBSERVATION: The observed outputs of all the basic gates are as follows:

* HALF ADDER :

VHDL Code: RTL Diagram:

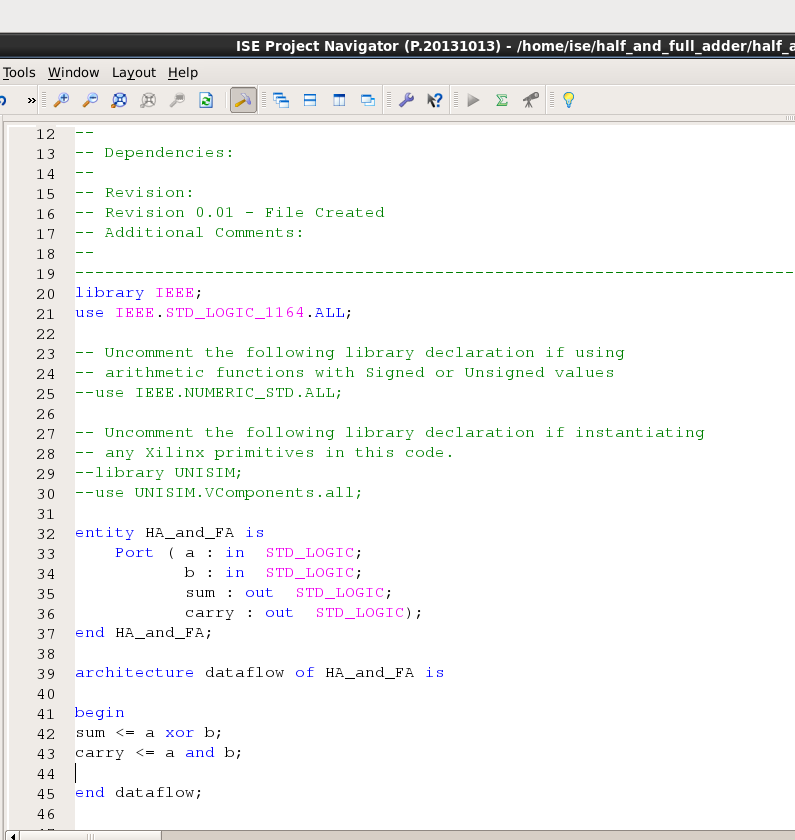
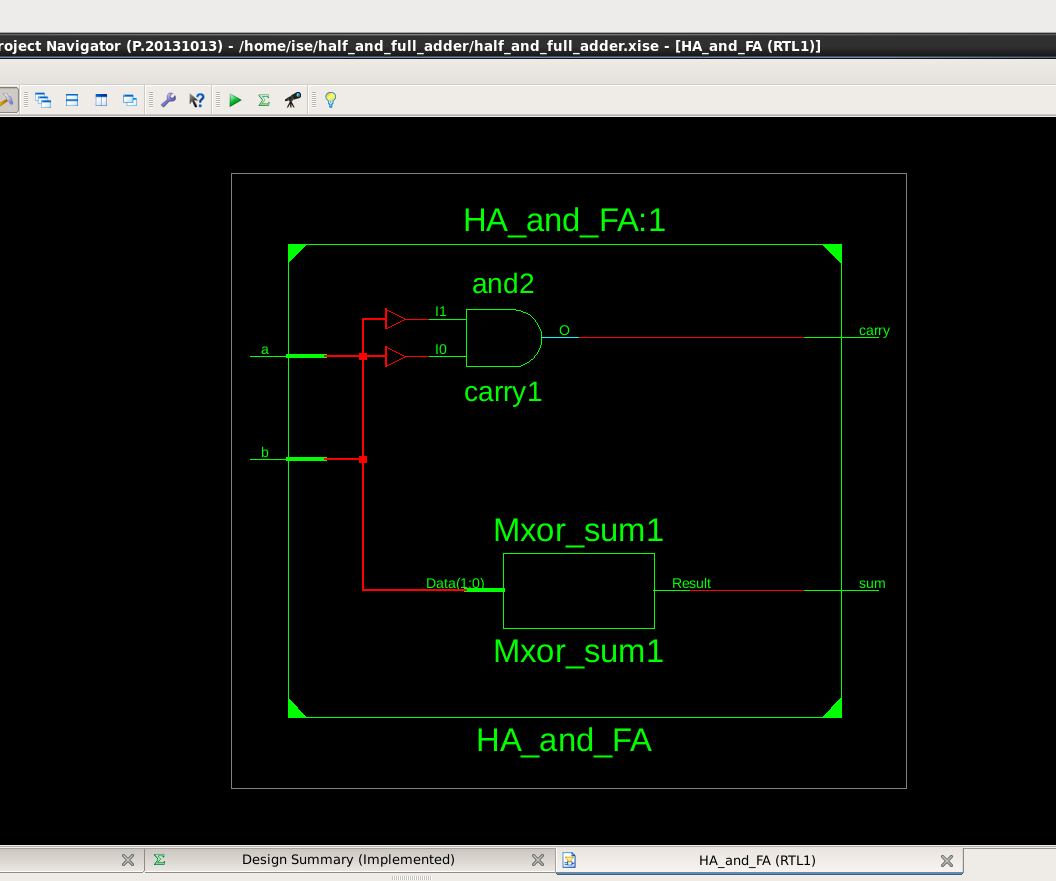
 

Figure 1 Figure 2

Test Bench Code: Waveform:

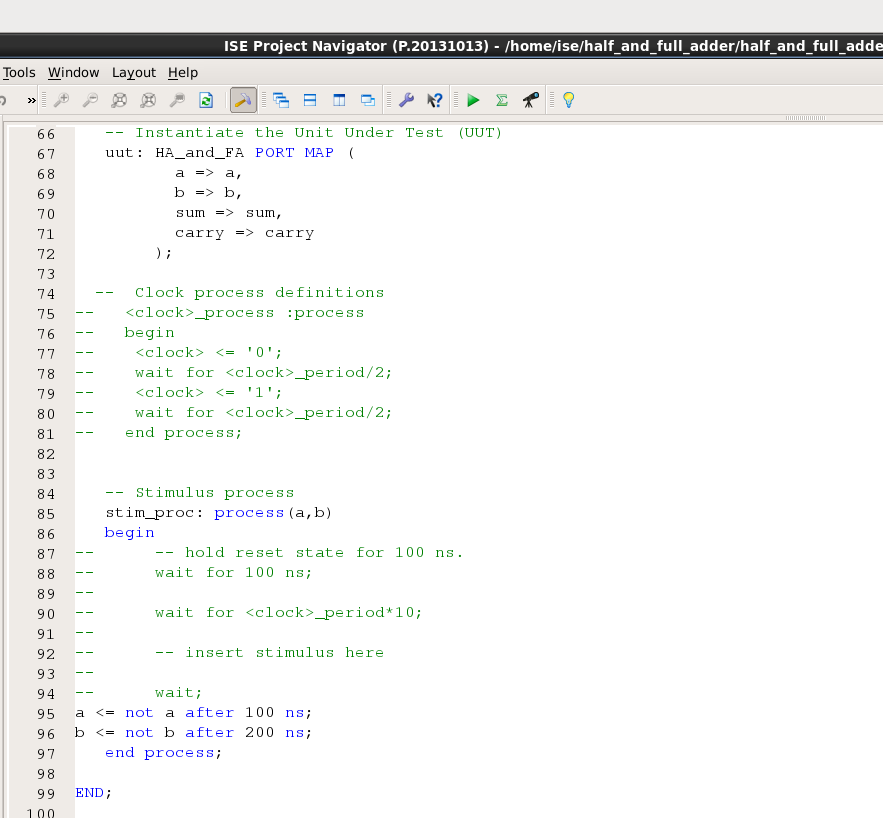
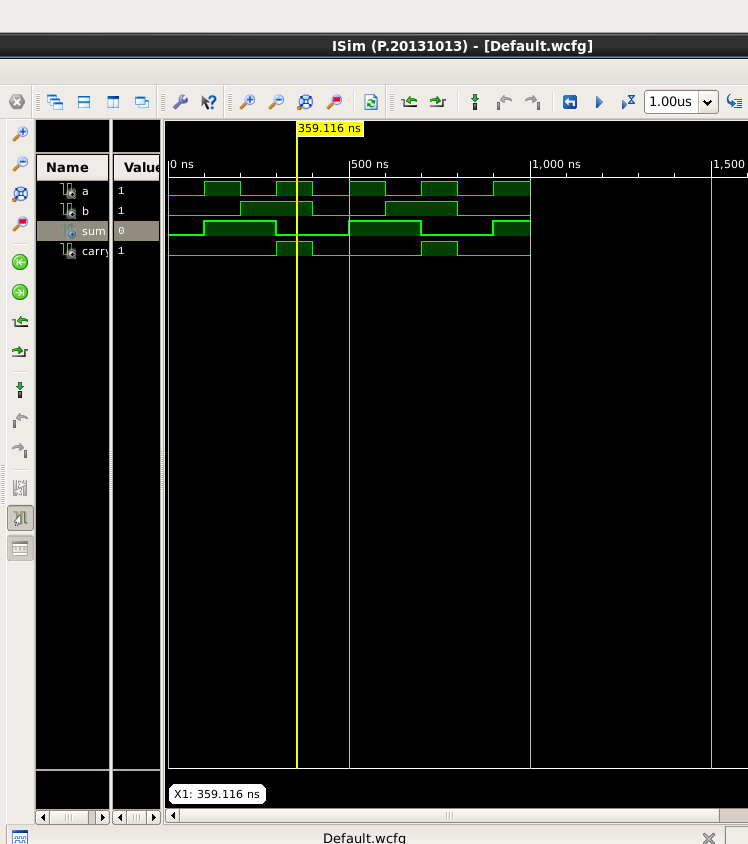
 

Figure 3 Figure 4

Here the yellow line in figure 4 represents the input (a=1 and b=1) and sum =0 and carry =1, which signifies the working of

**Sum = a xor b,**

**Carry = a and b**

respectively. In which we can see the change of input signals of and b after every 100 and 200 nano seconds.

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

Table 1

* FULL ADDER:

VHDL Code: RTL Diagram:

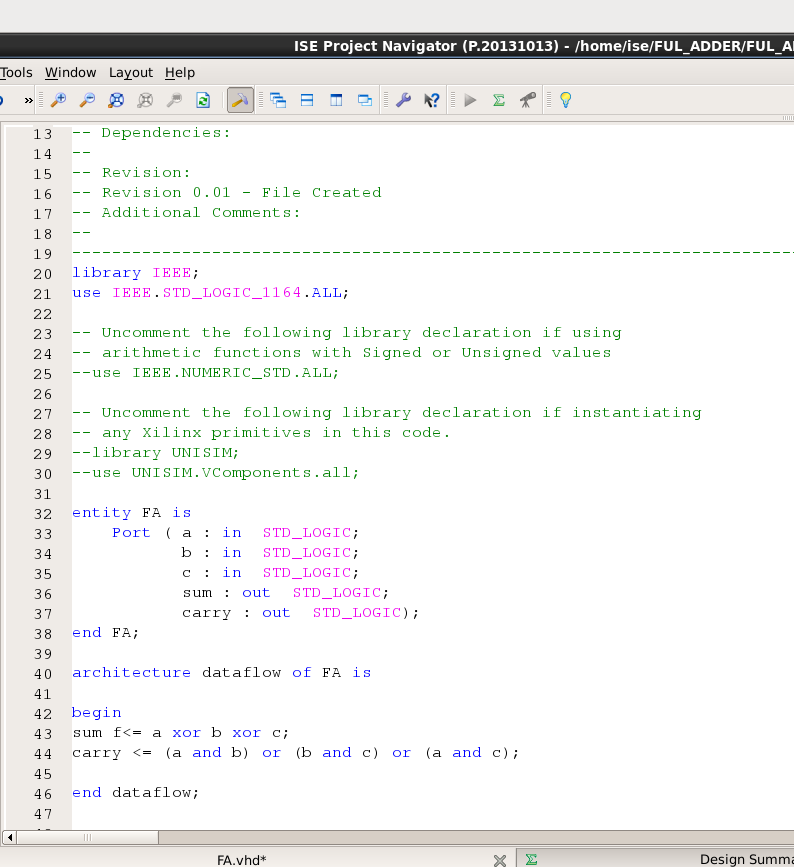
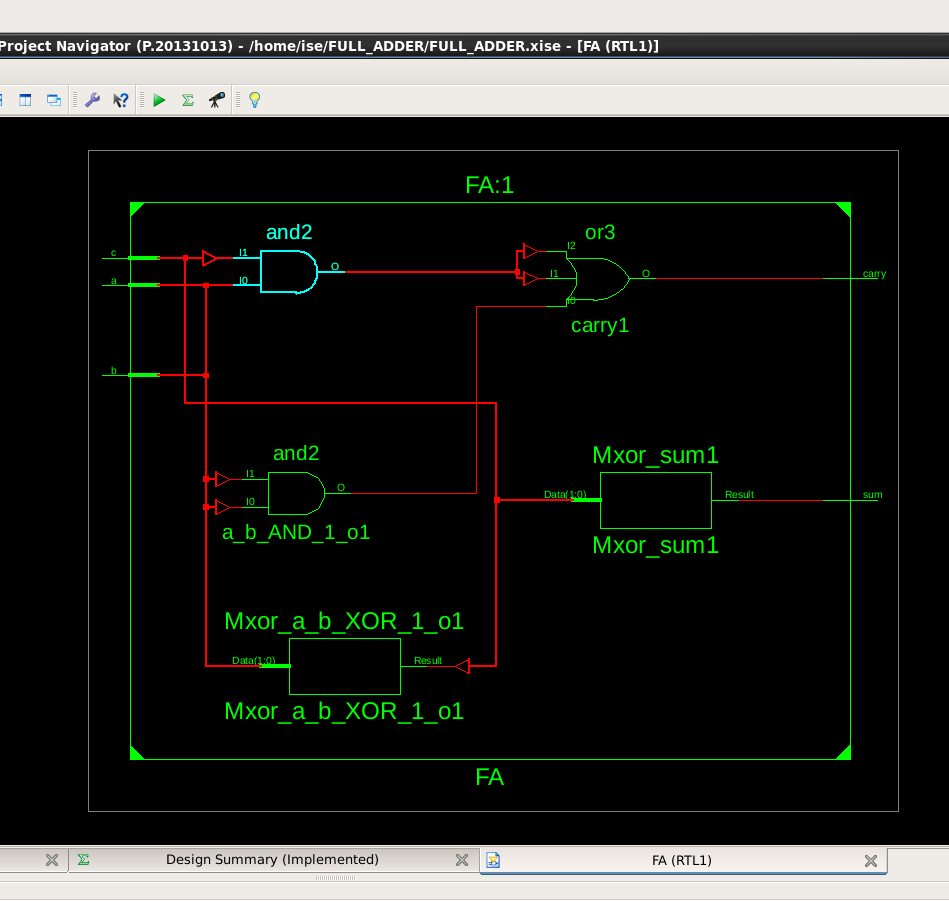
 

Figure 6 Figure 7

Test Bench Code:

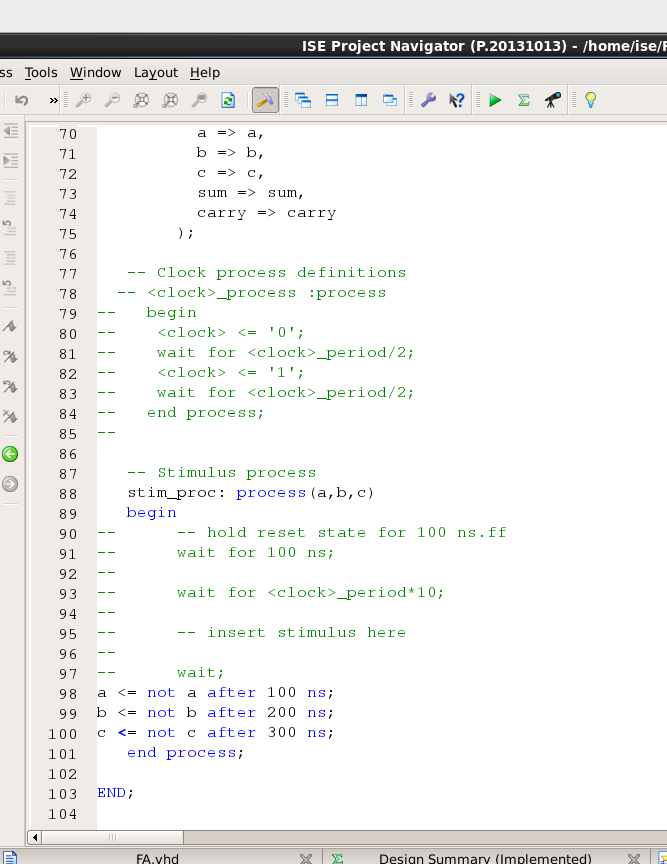


Figure 7

Waveform:

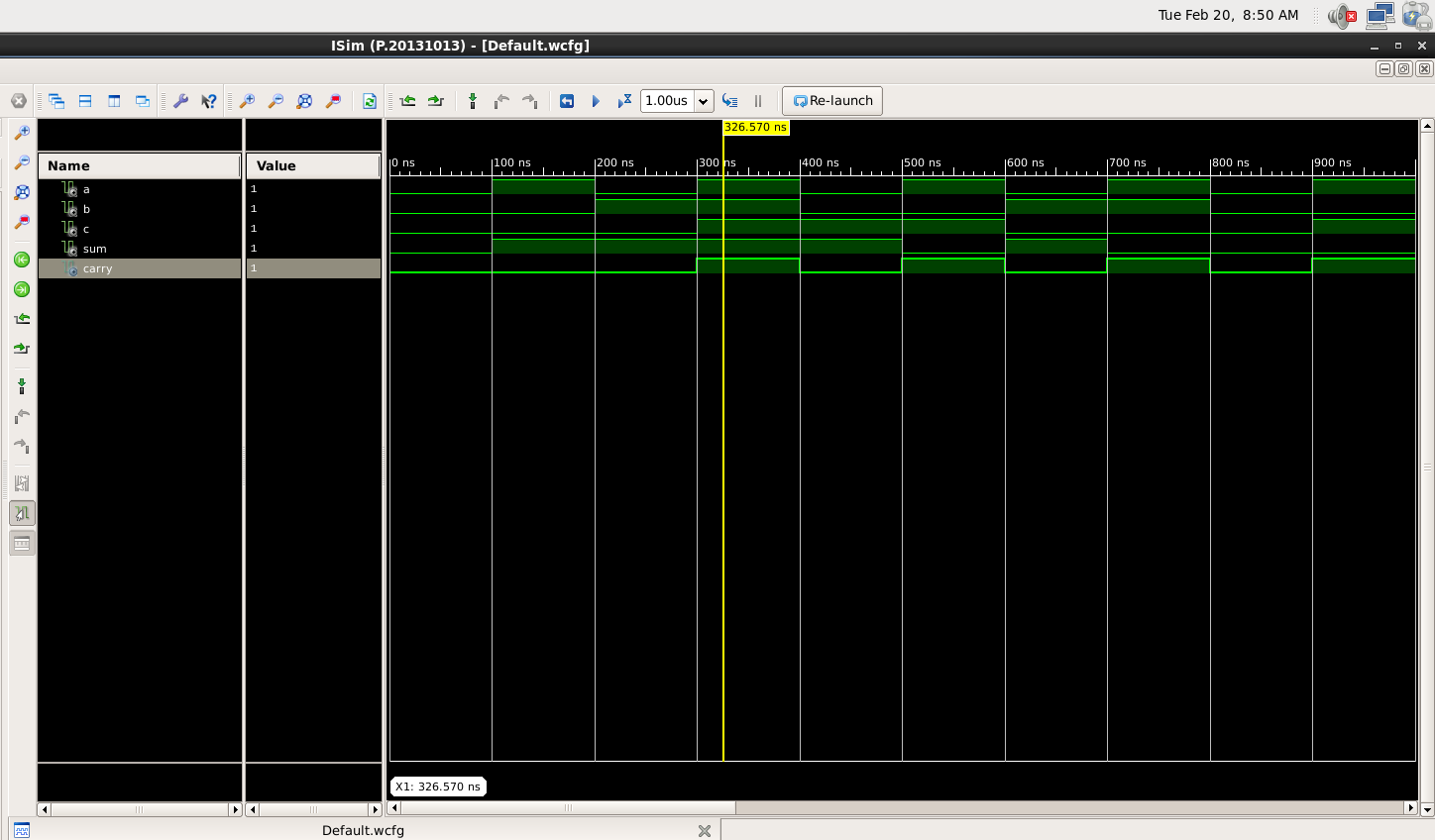


Figure 8

Here the yellow line in figure 8 represents the input (a=1, b=1 and carry(in)=1) and sum = 1 and carry(out) = 1, which signifies the working of

**Sum = a xor b xor c,**

**Carry(out) = (a and b) or (b and c) or (a and c)**

respectively. In which we can see the change of input signals of and b after every 100, 200 and 300 nano seconds respectively.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 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 |

Table 2

RESULT: We have concluded all the truth tables of half adder and full adder using VHDL language in Xilinx Vivado tool.

APPLICATION IN DAILY LIFE:

Half adders and full adders are fundamental building blocks in digital electronics, specifically in the design of digital circuits and arithmetic operations. While they may not have direct applications in daily life for the average person, they play a crucial role in various electronic devices and technologies that we encounter regularly. Here are some applications:

* Binary Arithmetic in Computers: Half adders and full adders are extensively used in the arithmetic logic unit (ALU) of a computer processor. They perform binary addition, which is fundamental to all computer operations, including calculations, data manipulation, and processing.
* Digital Signal Processors (DSPs): Digital signal processors use adders to perform various mathematical operations quickly and efficiently. This is crucial in applications such as audio signal processing, image processing, and telecommunications.
* Memory Addressing: Binary addition is used in memory addressing systems. Computers and microcontrollers utilize adders to calculate memory addresses for data retrieval and storage.