Circuit Simulation Project

https://esim.fossee.in/circuit-simulation-project

Name of the participant : Arnab Mondal

Project Guide: Dr. Maheswari.R

Title of the circuit : Full Adder using NAND Gates

Theory/Description :

Adders are digital circuits that carry out addition of numbers. Adders are a key component of Arithmetic Logic Unit (ALU) inside any CPU. Adders can be constructed for most of the numerical representations like Binary Coded Decimal (BDC), Excess – 3, Gray code, Binary etc. Out of these, binary addition is the most frequently performed task by most common adders. Apart from addition, adders are also used in certain digital applications like table index calculation, address decoding etc. Binary addition is similar to that of decimal addition. Add the first digits of a number and if the count exceeds binary 2, then carry '1' to the next row. Some basic binary additions are shown below. The adder that performs simple binary addition must have two inputs (augend and addend) and two outputs (sum and carry). The device which performs above task is called a Half Adder. A Full Adder is another circuit which can add three numbers (two bits from the numbers and one carry bit from previous sum).

A Full Adder is a Combinational Logic Circuit which performs binary addition on two-digit numbers. Full adders are complex and difficult to implement when compared to half adders. Full adder is a digital circuit used to calculate the sum of three binary bits, which is the main difference between this and half adder. Two of the three bits are same as before which are A, the augend bit and B, the addend bit. The additional third bit is carry bit from the previous stage and is called Carry–in, generally represented by C_{IN}. It calculates the sum of three bits along including the carry. The output carry is called Carry–out and is represented by C_{OUT}.

Truth Table:

INPUT			OUTPUT	
A	**B**	**Cin**	**Sum**	**Cout**
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

Equation for Sum as follows:

$S = A\ B\ C_{IN} + A\ B\ C_{IN} + A\ B\ C_{IN} + A\ B\ C_{IN}$

$= C_{IN}\ (A\ B + A\ B) + C_{IN}\ (A\ B + A\ B)$

$= C_{IN}$ (A Ex-NOR B) + C_{IN} (A Ex-OR B)

$= C_{IN}\ (A \oplus B) + C_{IN}\ (A \oplus B)$

Therefore, $S = C_{IN} \oplus (A \oplus B)$

Equation for C_{OUT} as follows:

$C_{OUT} = A\ B + A\ C_{IN} + B\ C_{IN}$

$= A\ B + A\ C_{IN} + B\ C_{IN}\ (A + A)$

$= A\ B + A\ C_{IN} + A\ B\ C_{IN} + A\ B\ C_{IN}$

$= A\ B\ (1 + C_{IN}) + A\ C_{IN} + A\ B\ C_{IN}$

$= A\,B + A\,C_{IN} + A\,B\,C_{IN}$

$= A\,B + A\,C_{IN}\,(B + B) + A\,B\,C_{IN}$

$= A\,B + A\,B\,C_{IN} + A\,B\,C_{IN} + A\,B\,C_{IN}$

$= A\,B\,(1 + C_{IN}) + C_{IN}\,(A\,B + A\,B)$

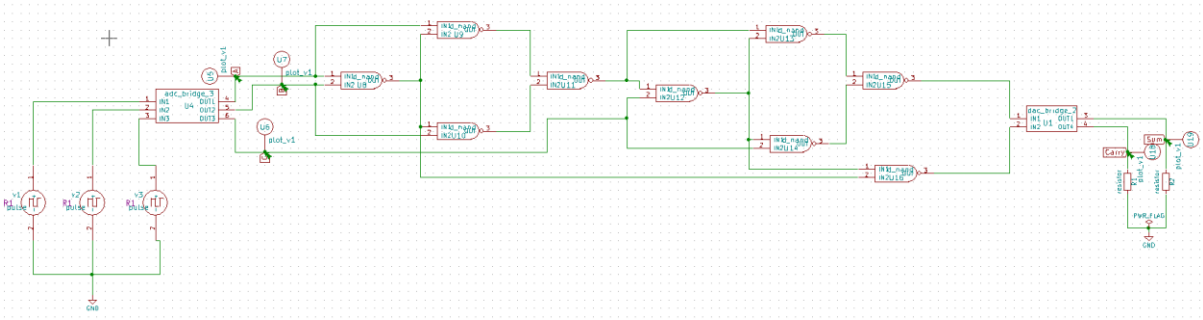
$= A\,B + C_{IN}\,(A\,B + A\,B)$

$= A\,B + C_{IN}\,(A \text{ Ex-OR } B)$

Therefore, $C_{OUT} = A\,B + C_{IN}\,(A \oplus B)$

Circuit Diagram(s) :

The circuit of full adder using only NAND gates is shown below.



Results (Input, Output waveforms and/or Multimeter readings) :

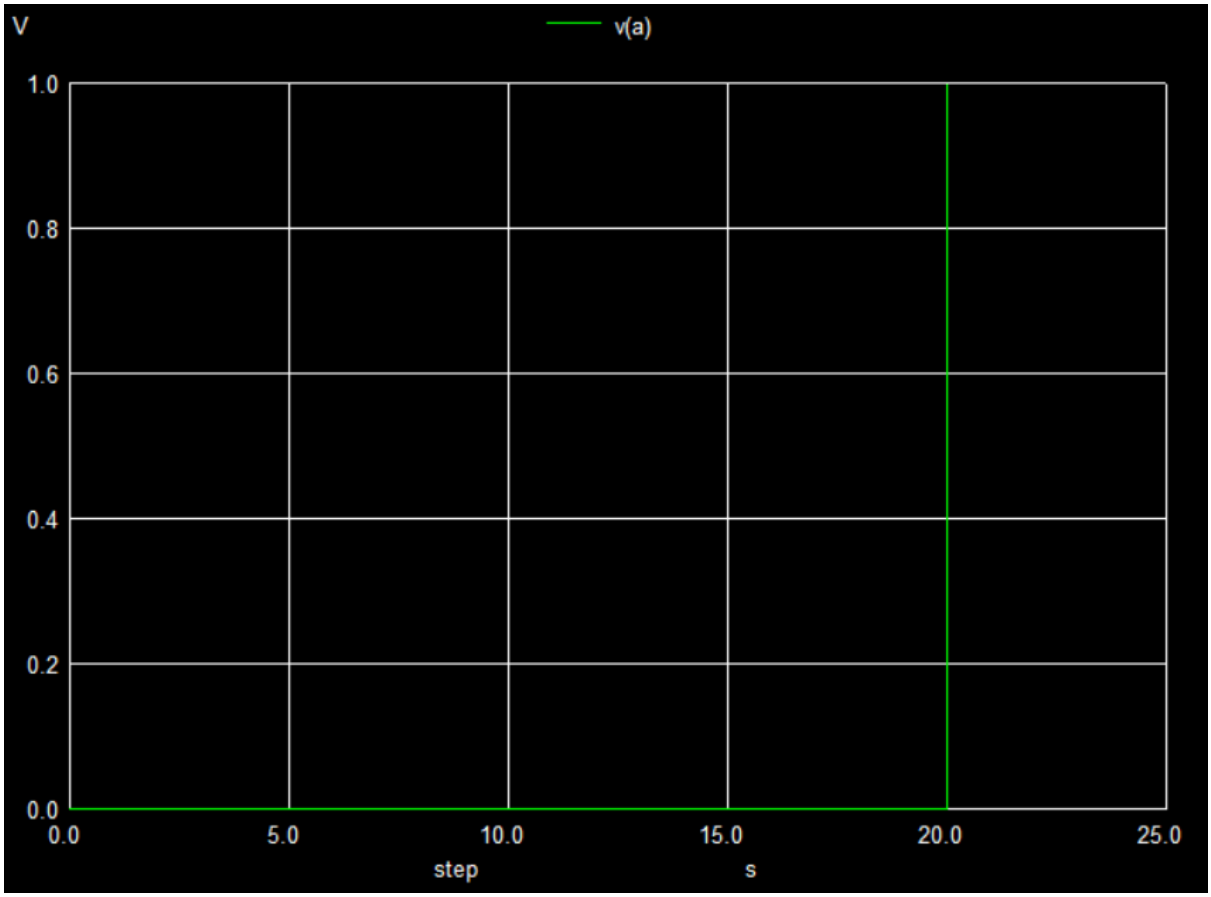


Fig. 2a: Analog signal for A

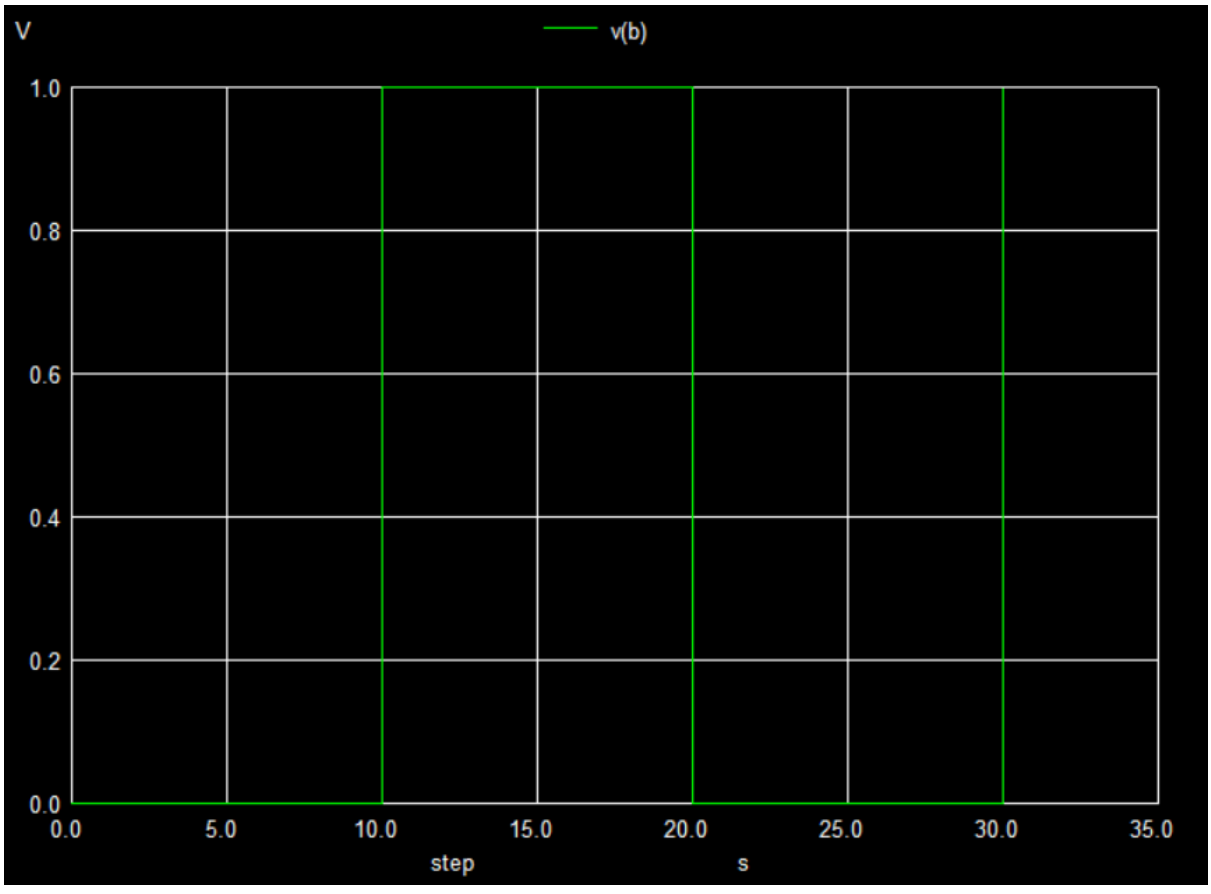


Fig. 2b: Analog signal for B

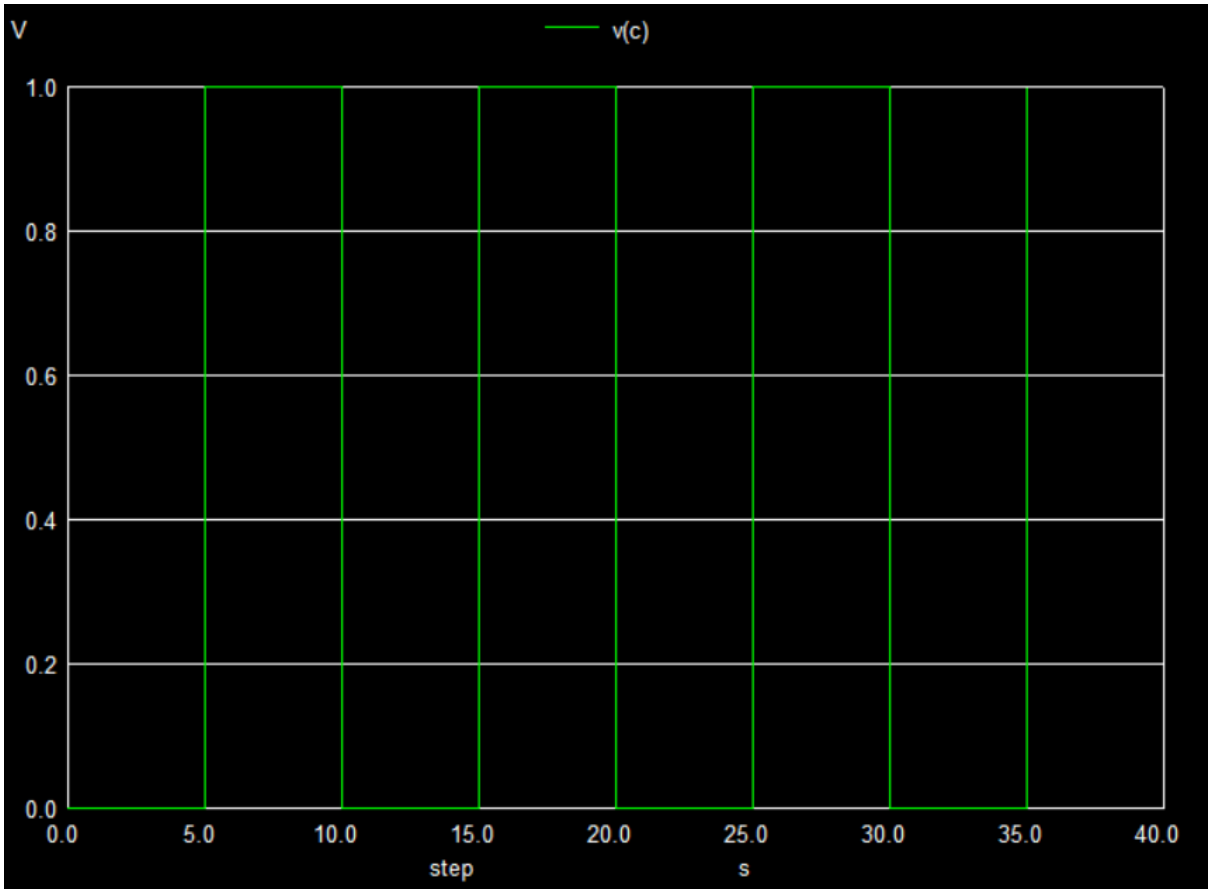


Fig. 2c: Analog signal for C

Figures 2a trough 2b show the analog signals for each of the input bits.

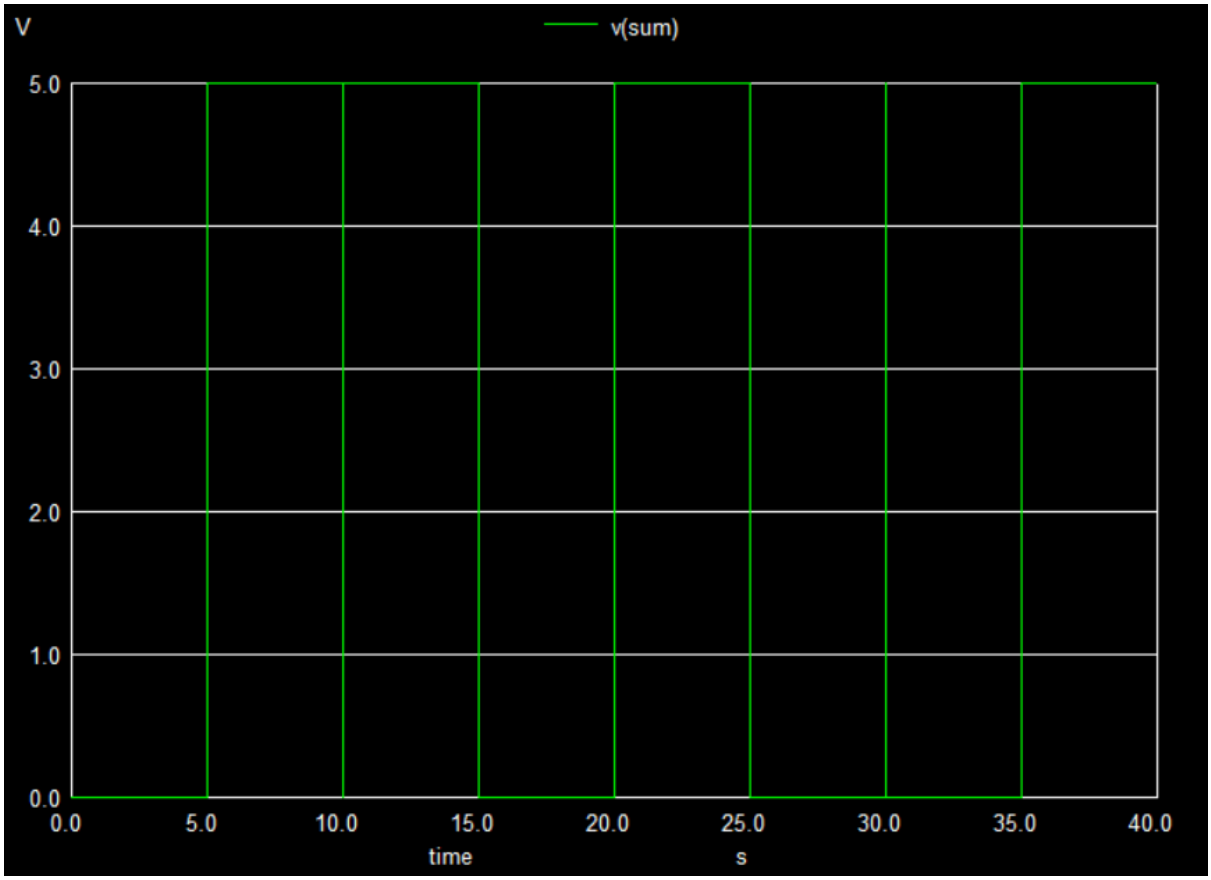


Fig. 3a: Analog signal for Sum

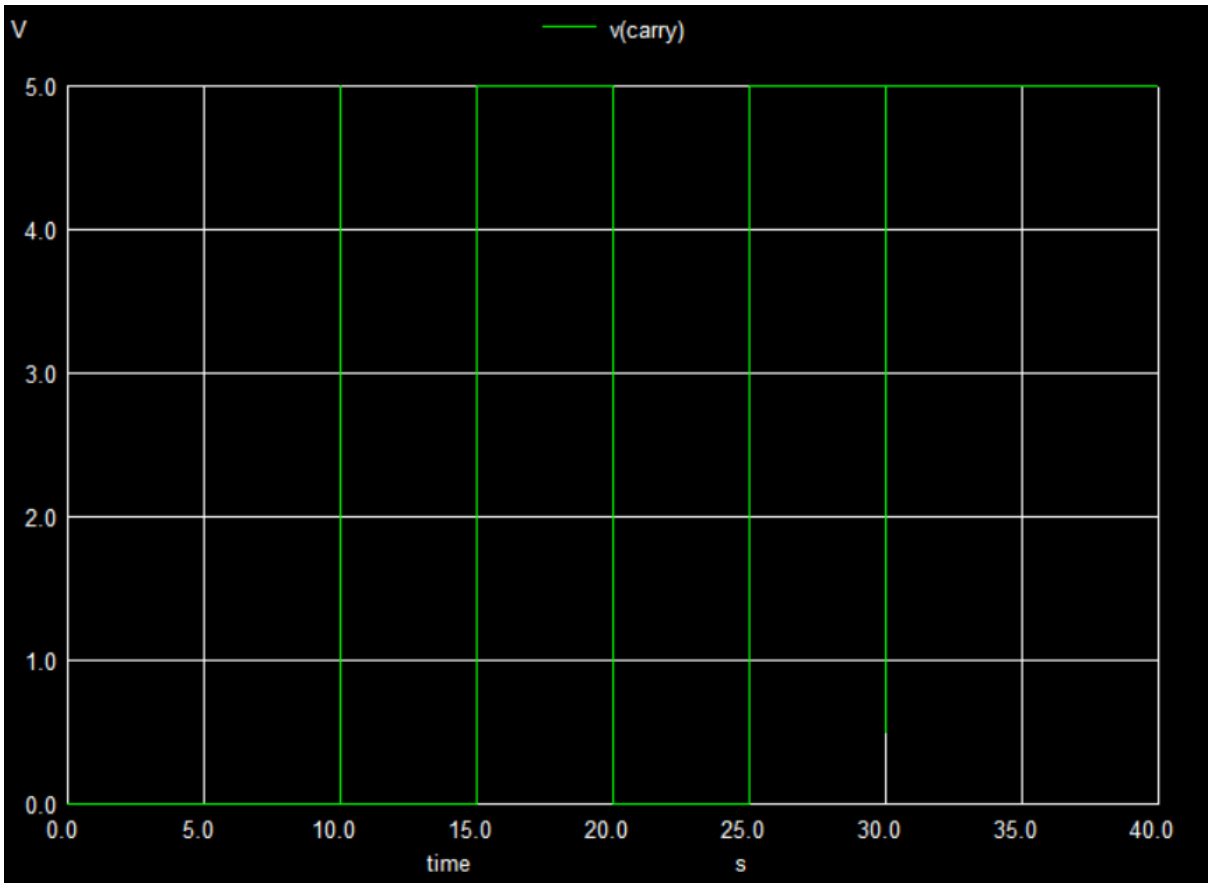


Fig. 3b: Analog signal for Carry

Figures 3a through 3b display show the output analog signals for each of the output bits.

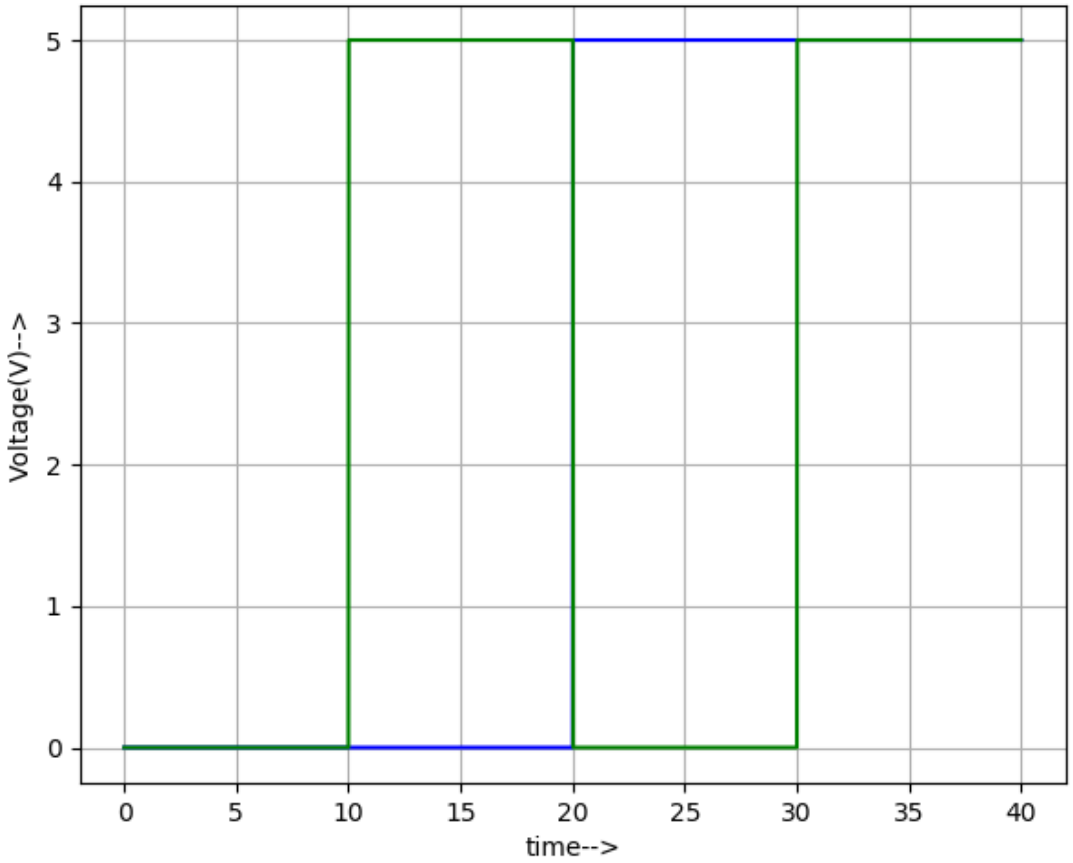


Fig. 4a: Python plots for A and B

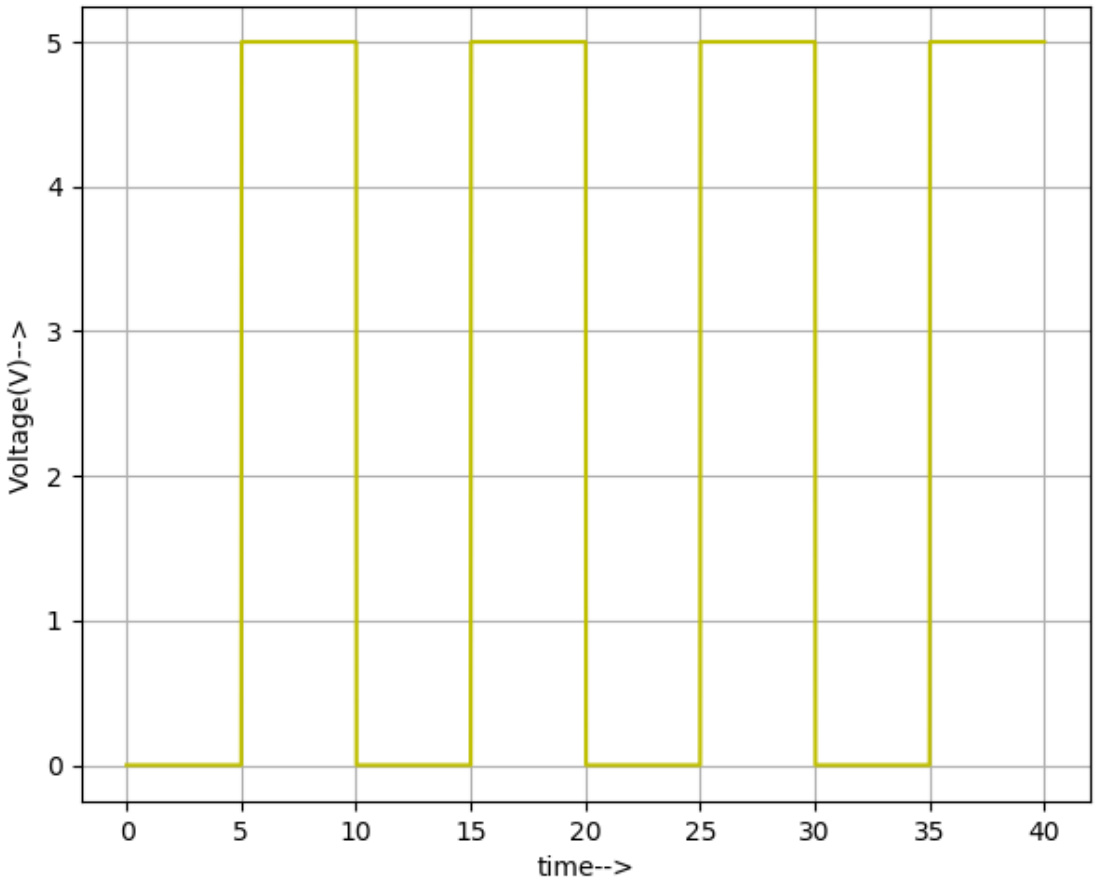


Fig. 4b: Python plot for C

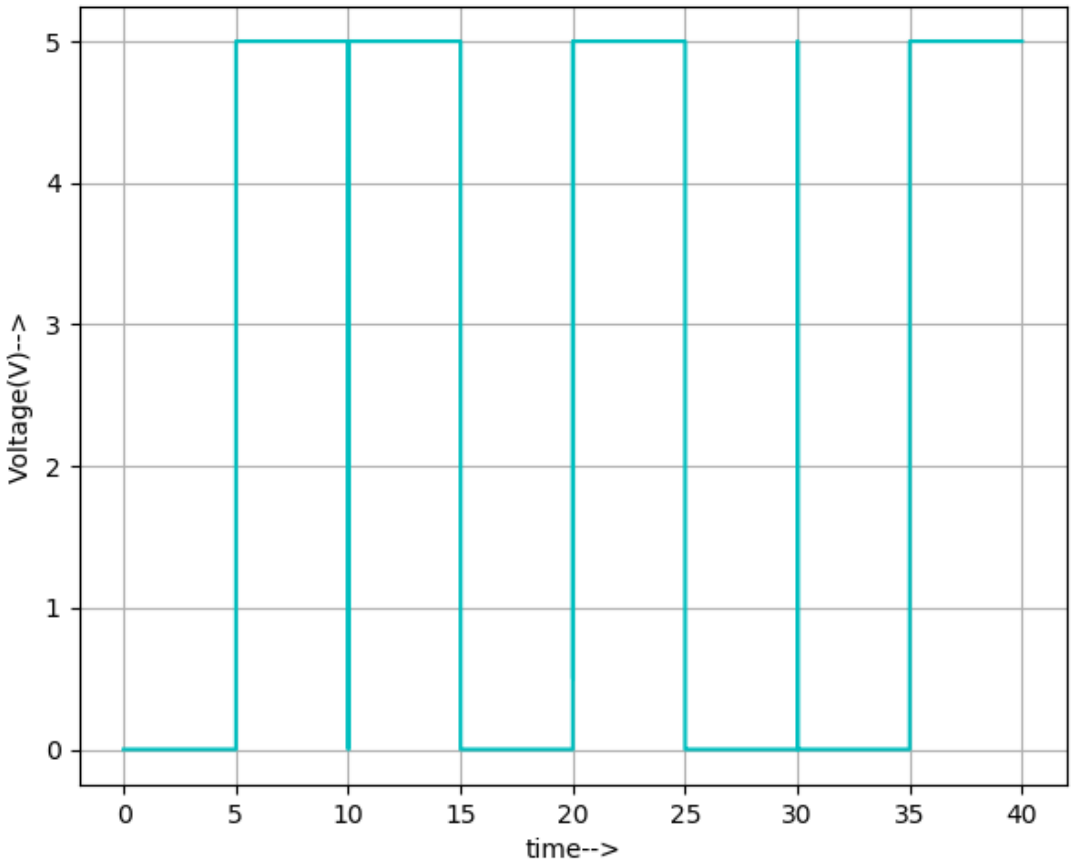


Fig. 4c: Python plot for Sum

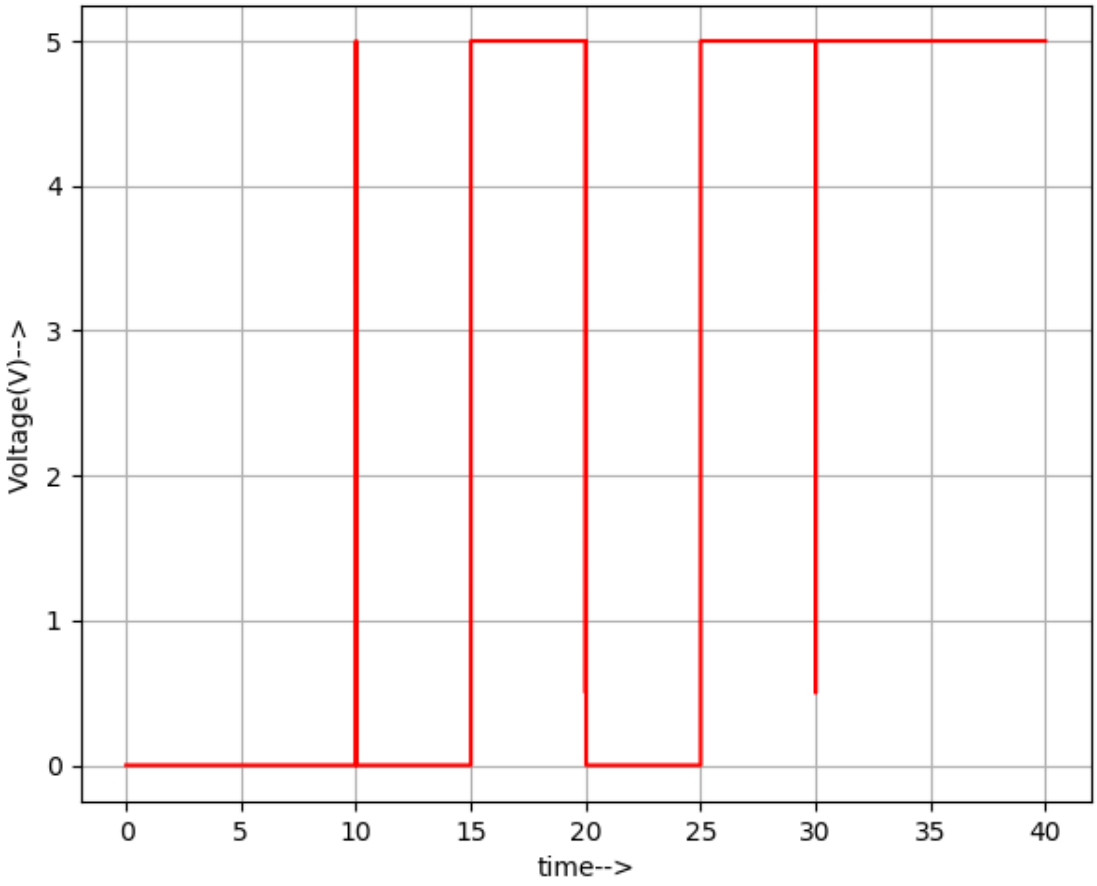


Fig. 4d: Python plot for Carry

Figures 4a through 4d show the python plots for the same signals for better visualisation.

Simulation parameter for reference:

Fig. 5a

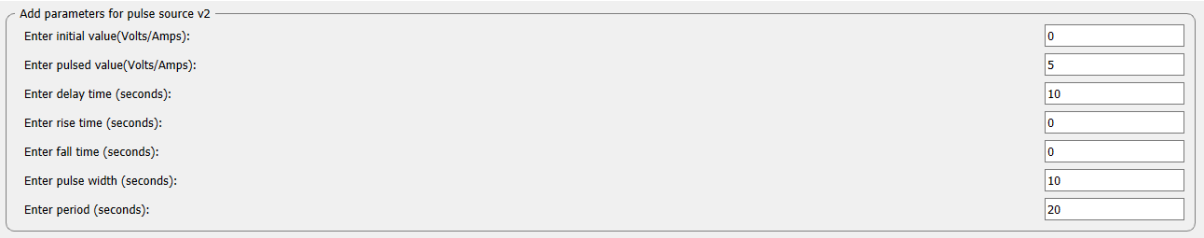


Fig. 5b

Fig. 5c

Source/Reference(s) :

https://www.electronicshub.org/half-adder-and-full-adder-circuits/

(Full Adder using NAND Gates)