Circuit Simulation Project

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

Name of the participant: Arjun Vishanth

Project Guide: Dr. R. Maheswari

Title of the Project: BCD to Excess 3 Code Conversion

Theory & Description:

The Excess-3 binary code is an example of a self-complementary BCD code. A self-complementary binary code is a code which is always complimented in itself. By replacing the bit 0 to 1 and 1 to 0 of a number, we find the 1's complement of the number. The sum of the 1'st complement and the binary number of a decimal is equal to the binary number of decimal 9.

The process of converting BCD to Excess-3 is quite simple from other conversions. The Excess-3 code can be calculated by adding 3, i.e., 0011 to each four-digit BCD code. Below is the truth table for the conversion of BCD to Excess-3 code. In the below table, the variables A, B, C, and D represent the bits of the binary numbers. The variable 'D' represents the LSB, and the variable 'A' represents the MSB. In the same way, the variables w, x, y, and z represent the bits of the Excess-3 code. The variable 'z' represents the LSB, and the variable 'w' represents the MSB. The 'don't care conditions' is expressed by the variable 'X'.

In this circuit a BCD to Excess-3 code converter has been implemented and simulated using the AND, NAND and OR gates.

Truth table:

Decimal Number	BCD Code				Excess-3 Code			
	A	B	C	D	W	X	Y	Z

0	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**1**
1	**0**	**0**	**0**	**1**	**0**	**1**	**0**	**0**
2	**0**	**0**	**1**	**0**	**0**	**1**	**0**	**1**
3	**0**	**0**	**1**	**1**	**0**	**1**	**1**	**0**
4	**0**	**1**	**0**	**0**	**0**	**1**	**1**	**1**
5	**0**	**1**	**0**	**1**	**1**	**0**	**0**	**0**
6	**0**	**1**	**1**	**0**	**1**	**0**	**0**	**1**
7	**0**	**1**	**1**	**1**	**1**	**0**	**1**	**0**
8	**1**	**0**	**0**	**0**	**1**	**0**	**1**	**1**
9	**1**	**0**	**0**	**1**	**1**	**1**	**0**	**0**
10	**1**	**0**	**1**	**0**	**X**	**X**	**X**	**X**
11	**1**	**0**	**1**	**1**	**X**	**X**	**X**	**X**
12	**1**	**1**	**0**	**0**	**X**	**X**	**X**	**X**
13	**1**	**1**	**0**	**1**	**X**	**X**	**X**	**X**
14	**1**	**1**	**1**	**0**	**X**	**X**	**X**	**X**
15	**1**	**1**	**1**	**1**	**X**	**X**	**X**	**X**

From the above truth table we observe,

W = A + BC + BD
X = B'C + B'D + BC'D'
Y = CD + C'D'
Z = D'

Circuit Diagram:

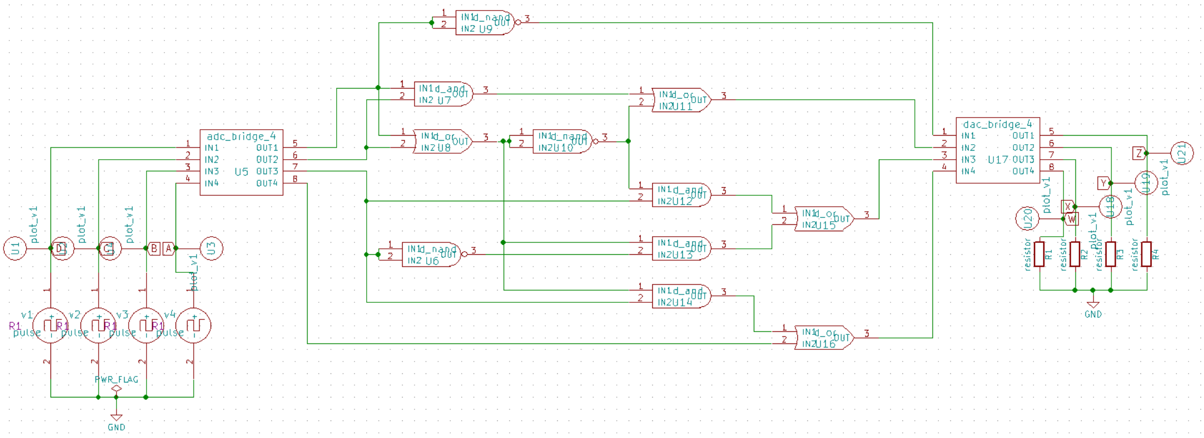

Fig. 1: Main circuit

Fig. 1 represents the schematic for the main circuit. The pulse sources feed the inputs from 0000 to 1111. The ADC bridge is used to convert the analog voltage source inputs into digital bits to be used with the digital gates. The DAC bridge is used to convert the digital signals back to analog so that they can be plotted and viewed as output.

Results (Input, Output waveforms):

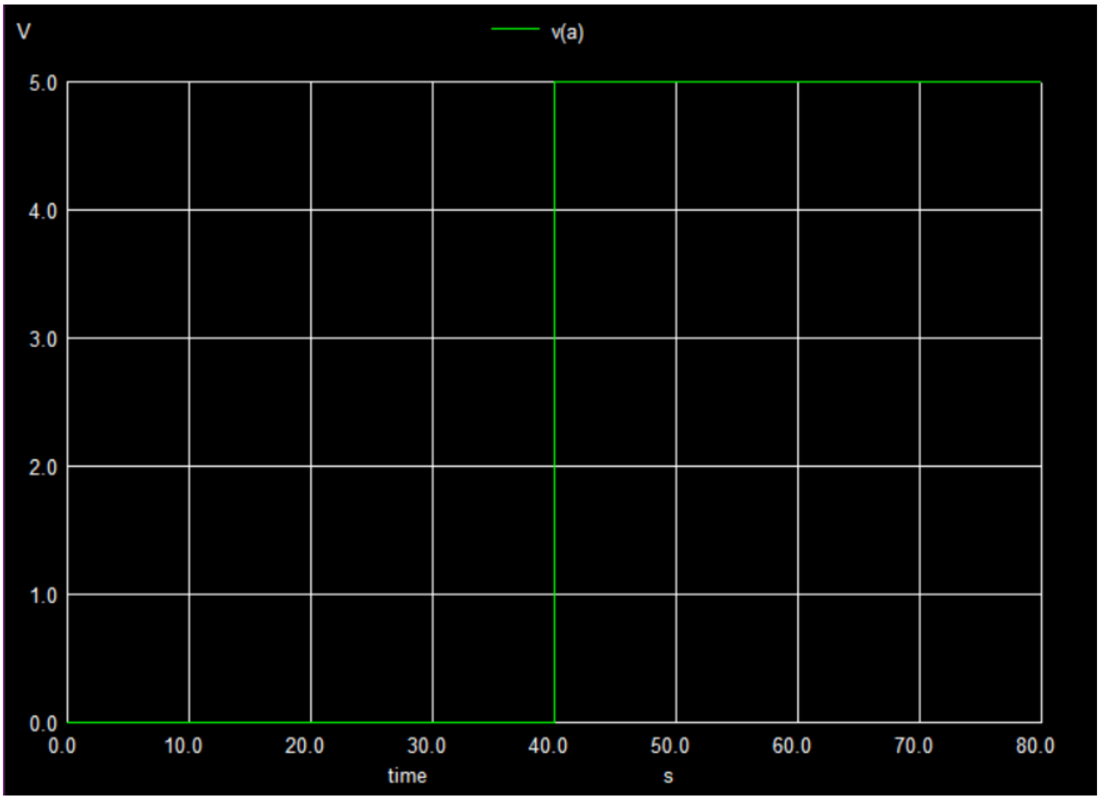

Fig. 2a: Analog signal for A (MSB)

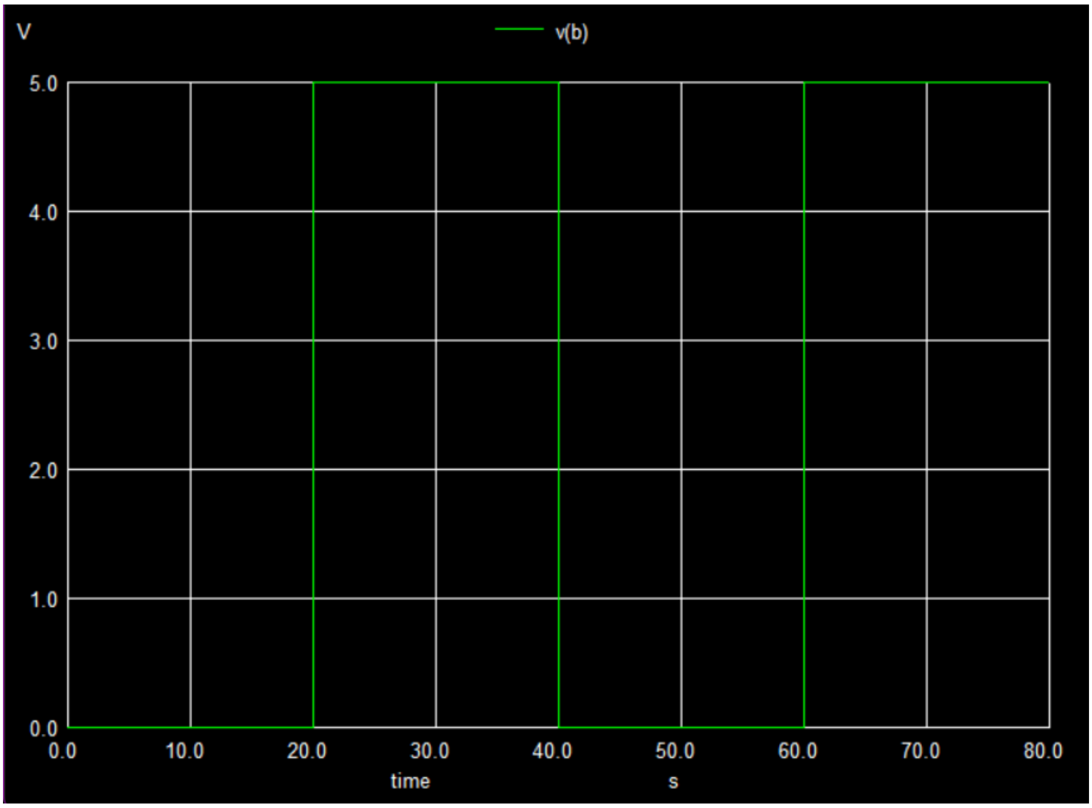

Fig. 2b: Analog signal for B

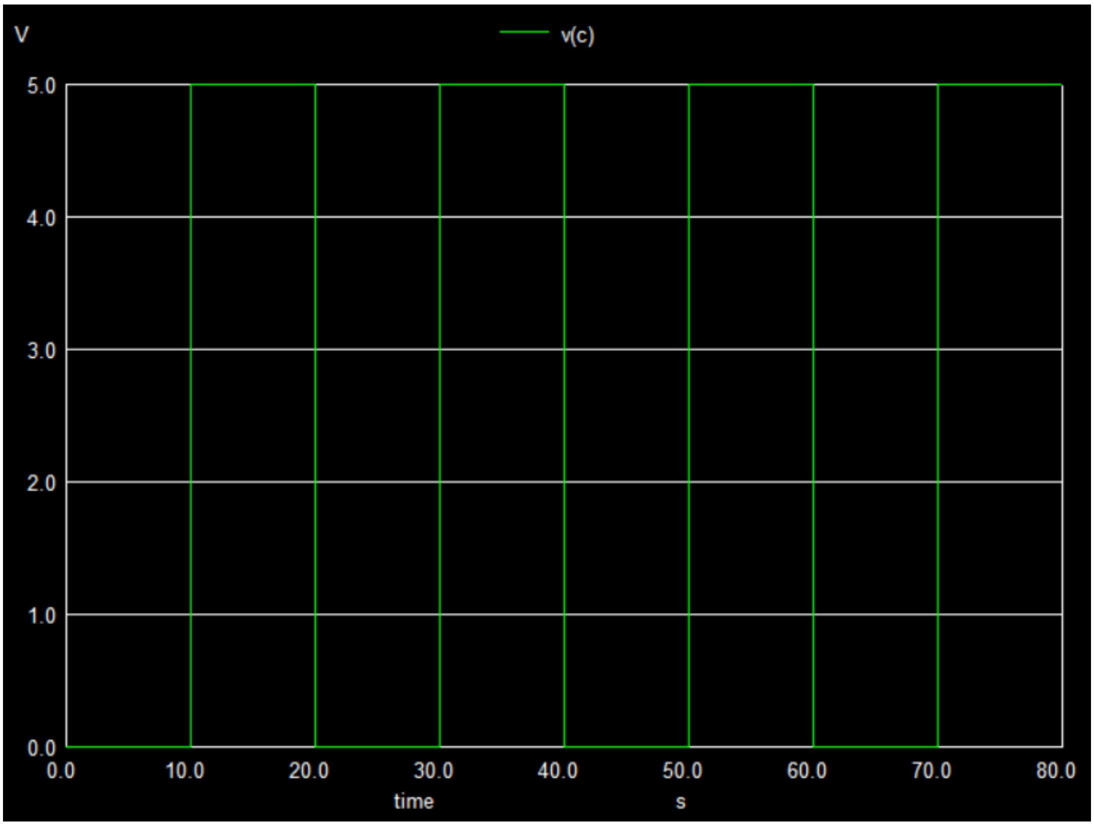

Fig. 2c: Analog signal for C

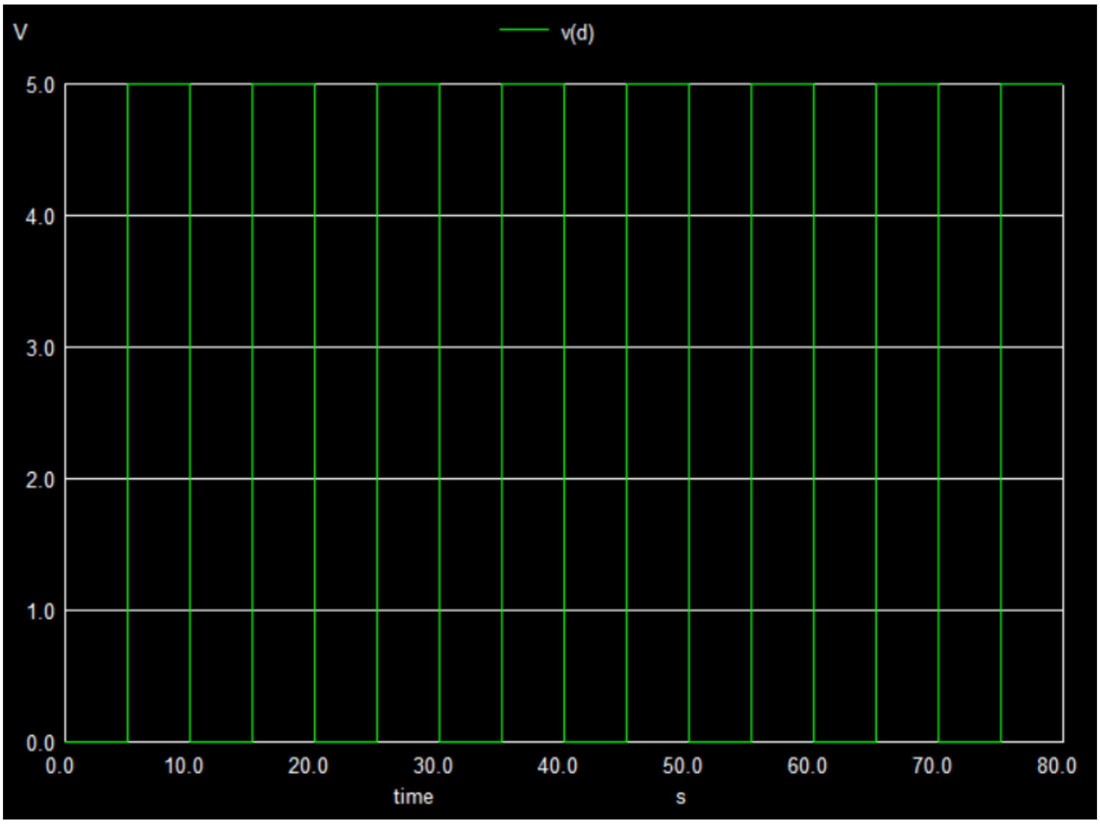

Fig. 2d: Analog signal for D (LSB)

Figures 2a through 2d show the analog signals for each of the inputs bits (BCD code).

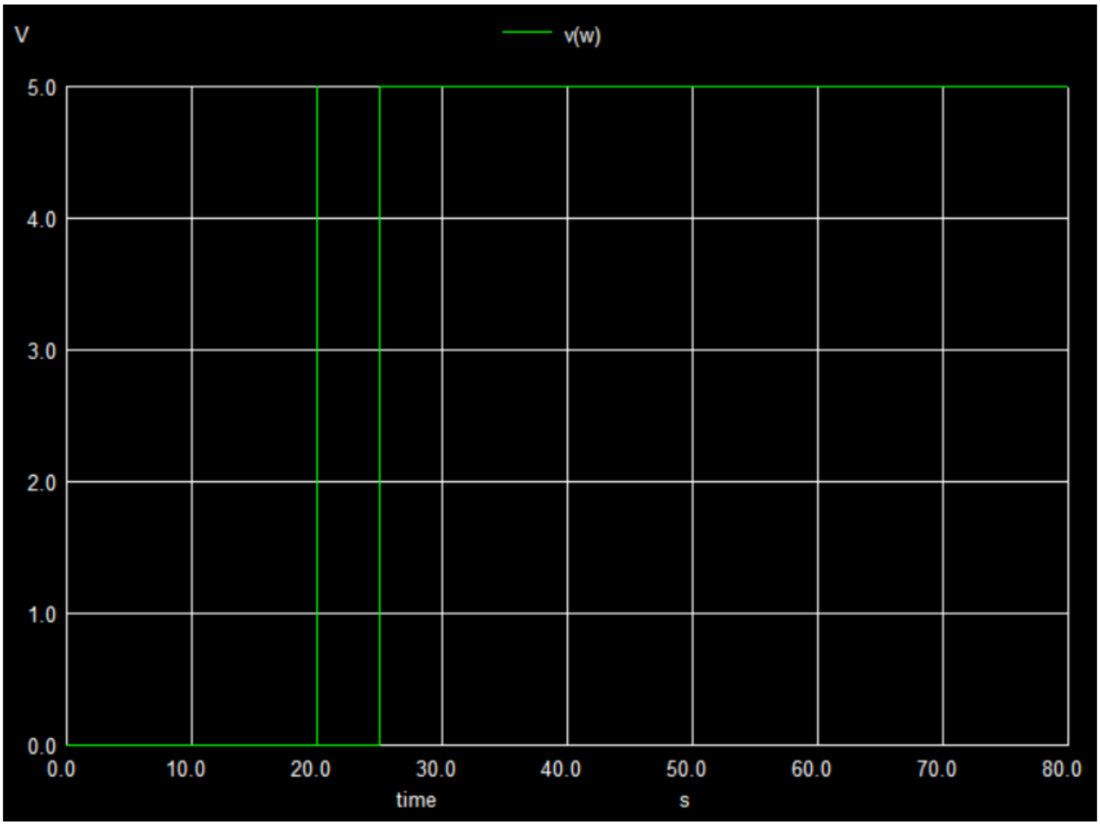

Fig. 3a: Analog signal for W (MSB)

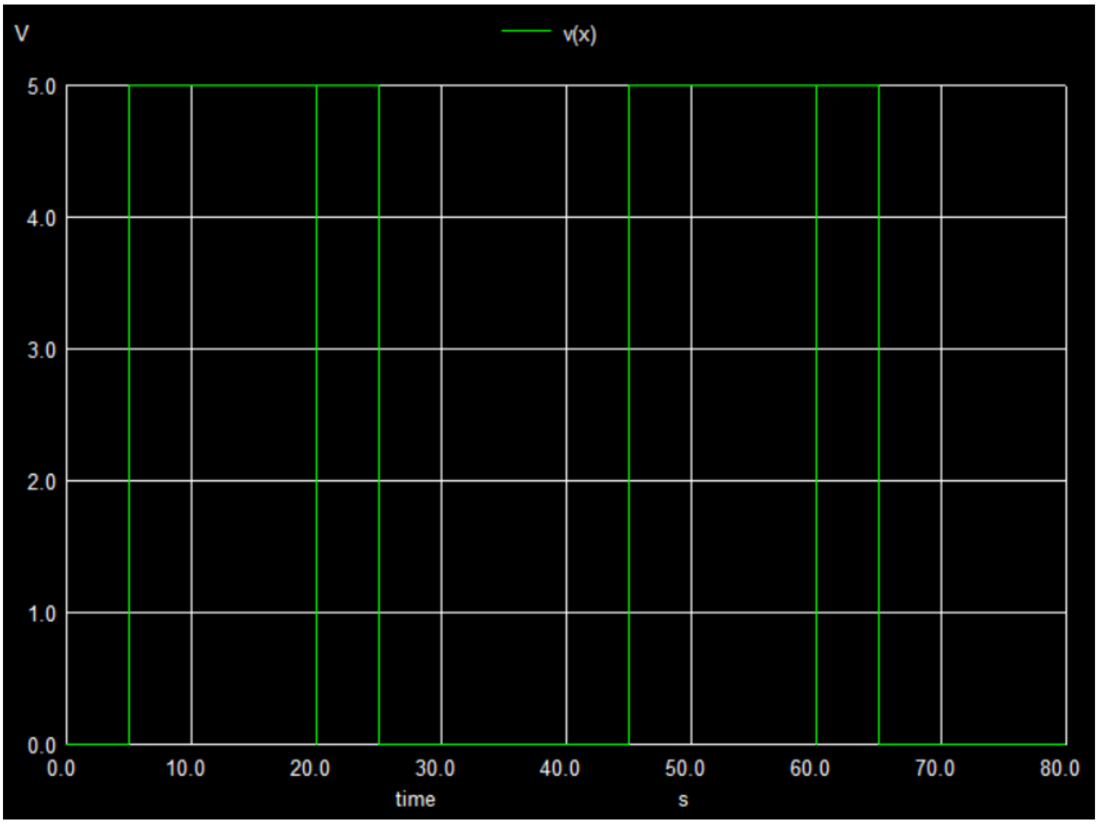

Fig. 3b: Analog signal for X

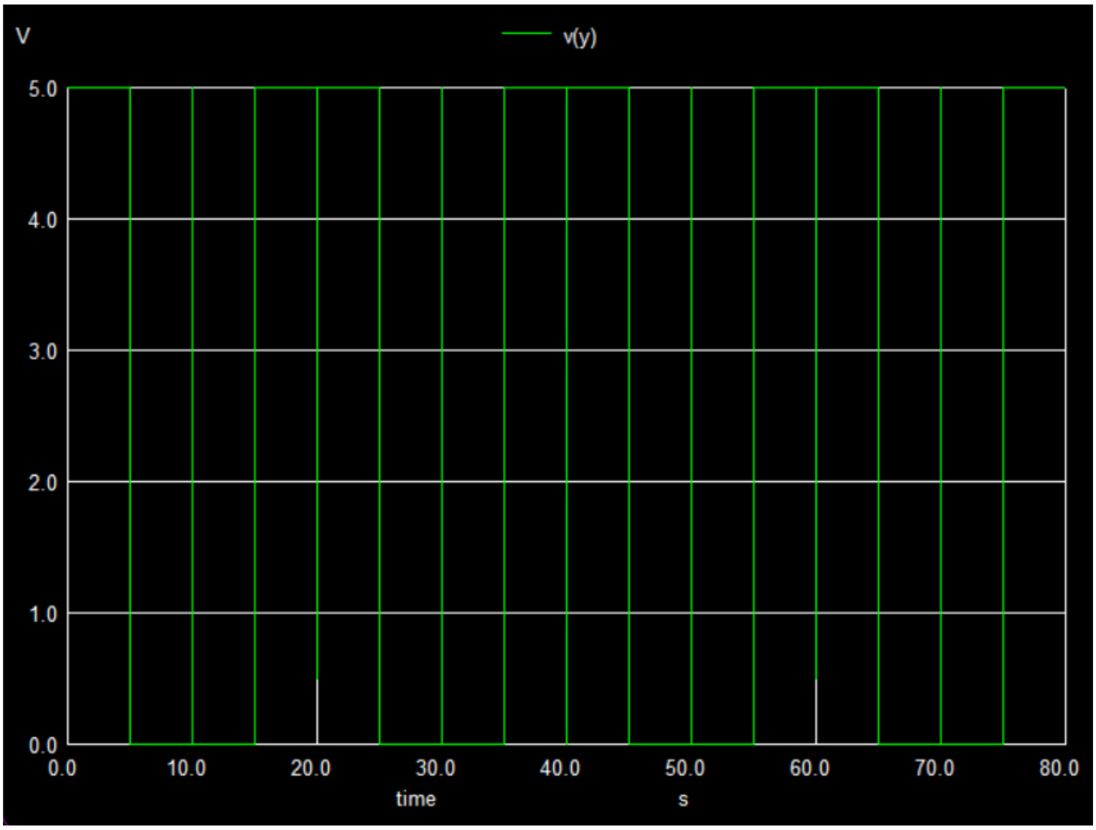

Fig. 3c: Analog signal for Y

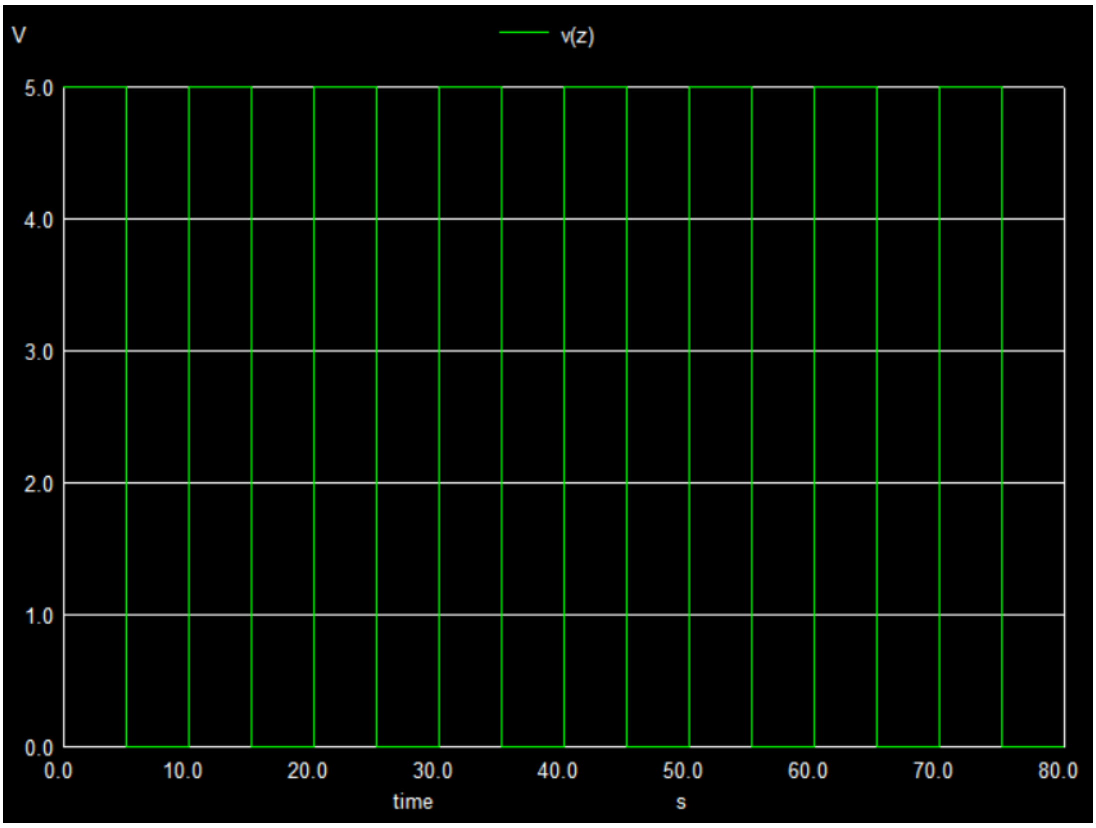

Fig. 3d: Analog signal for Z (LSB)

Figures 3a through 3d display show the output analog signals for each of the output bits (Excess-3 code).

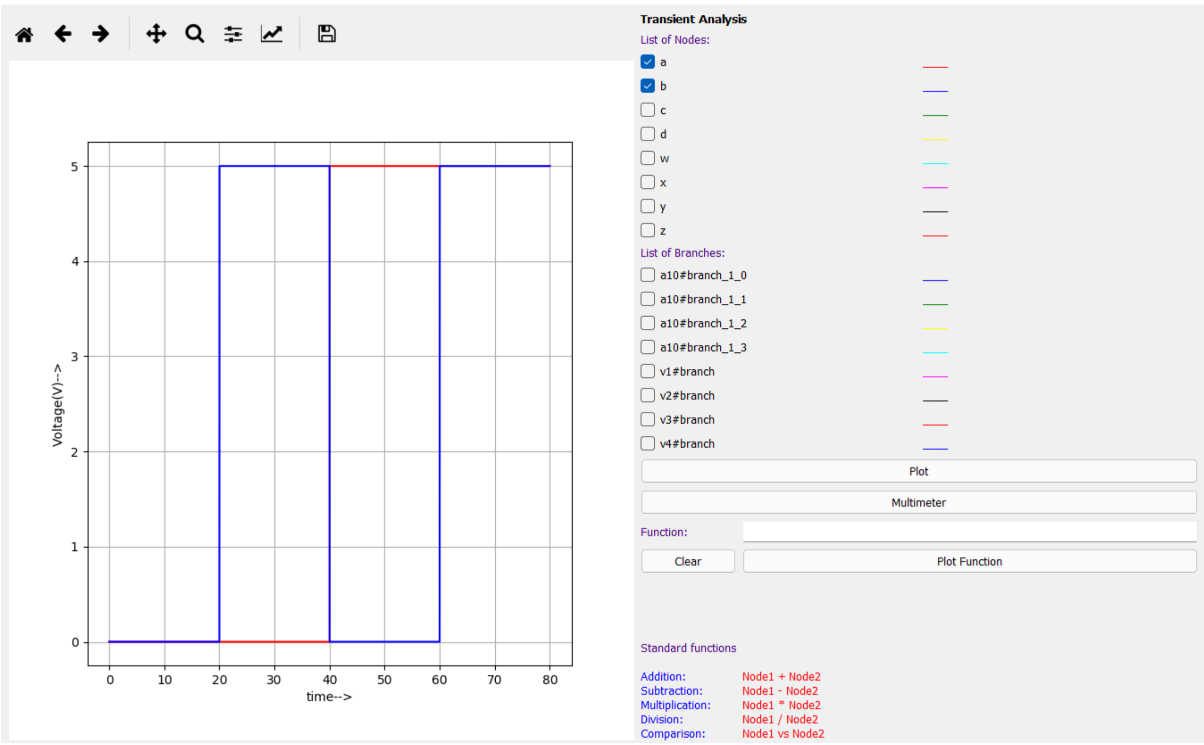

Fig. 4a: Python plots for A and B

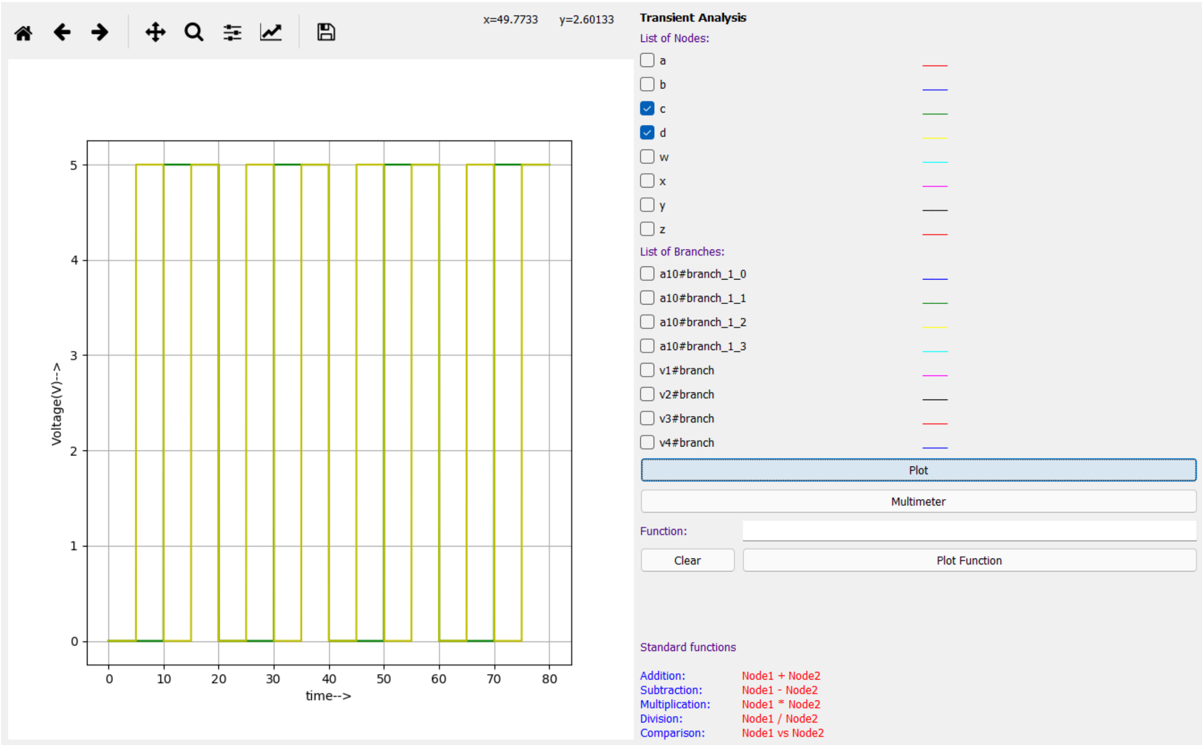

Fig. 4b: Python plots for C and D

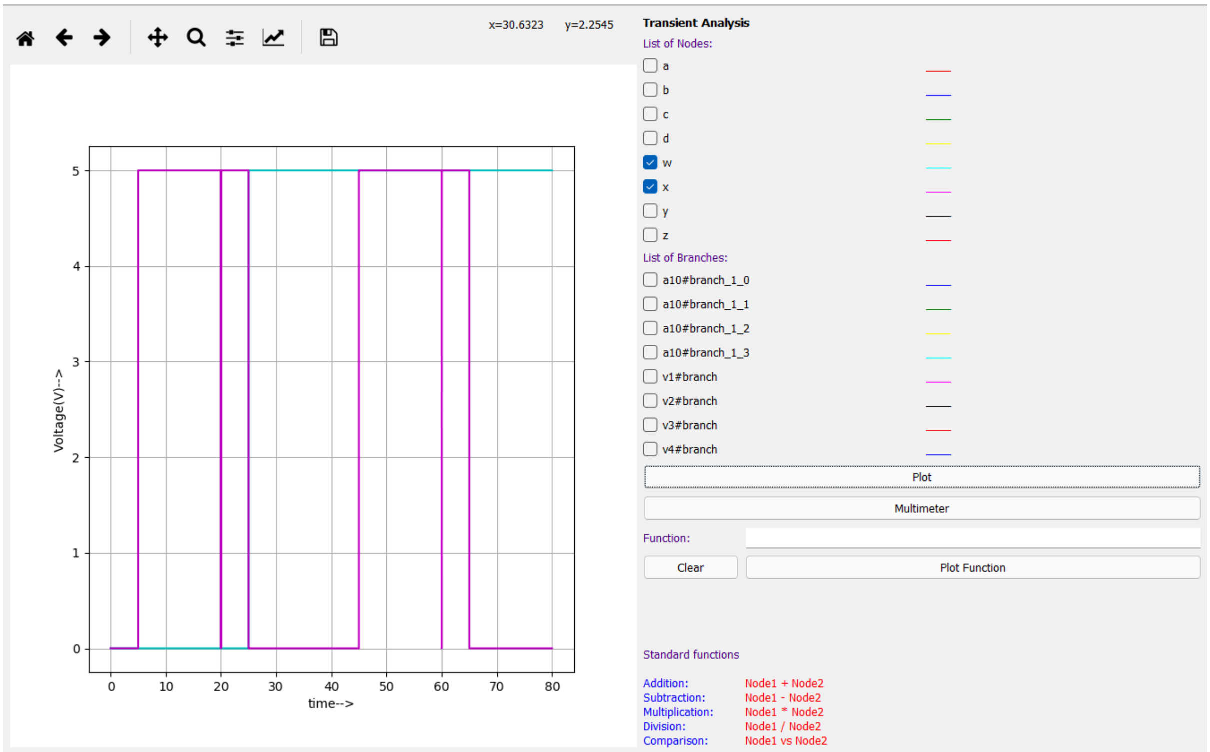

Fig. 4c: Python plots for W and X

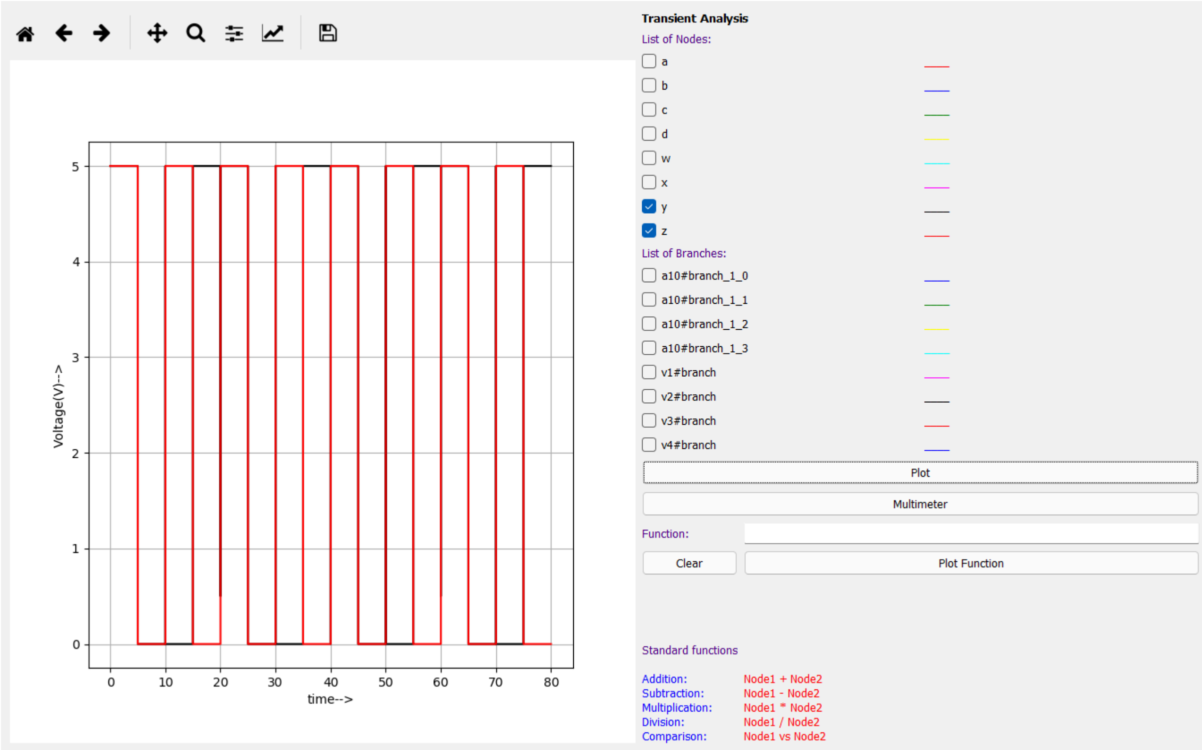

Fig. 4d: Python plots for Y and Z

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

Simulation parameters for reference:

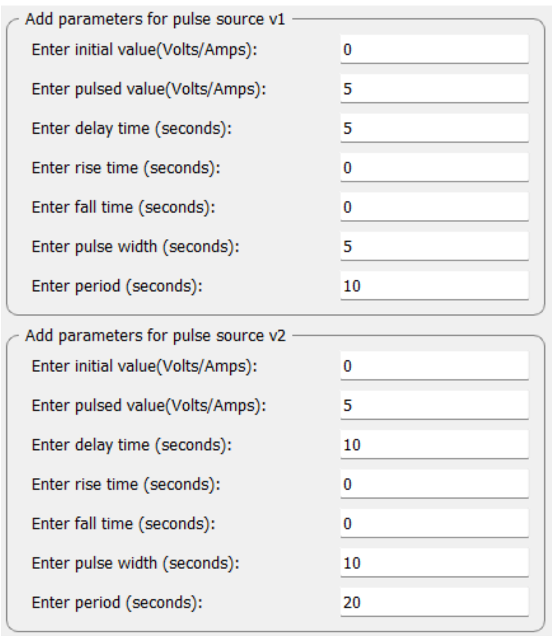

Fig. 5a

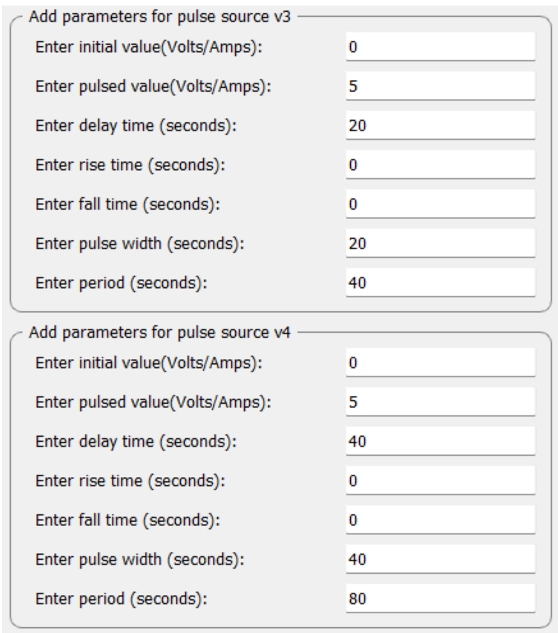

Fig. 5b

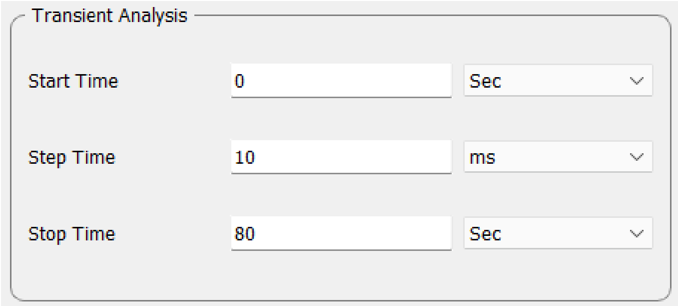

Fig. 5c

Source / Reference:

https://www.javatpoint.com/conversion-of-bcd-to-excess-3-code-in-digital-electronics
(BCD to Excess-3 conversion)