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

Name of Participant: Ajith Suresh

Project Guide: Dr. Maheswari R

Title: FULL SUBTRACTOR USING 1:4 DEMUX CIRCUIT

THEORY:

A full subtractor is a combinational circuit that performs subtraction of two bits, one is minuend and other is subtrahend, taking into account borrow of the previous adjacent lower minuend bit. This circuit has three inputs and two outputs**.** The three inputs A, B and Bin, denote the minuend, subtrahend, and previous borrow, respectively. The two outputs, D

and B out represent the difference and output borrow, respectively.

We can implement a full subtractor using a 1×8 de multiplexer. Let's first see the truth table for full subtractor.

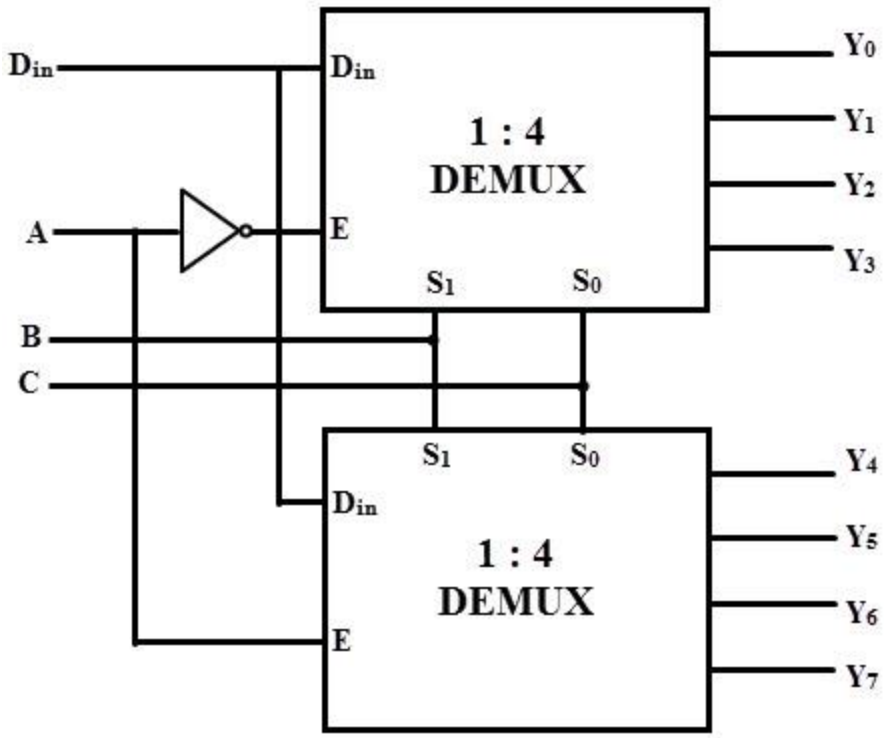
Inputs			Outputs	
X	Y	B_i	D	B_o
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

SOP form expression for the difference and borrow bit.

Difference-> $D(X, Y, B_i) = \Sigma m(1, 2, 4, 7)$

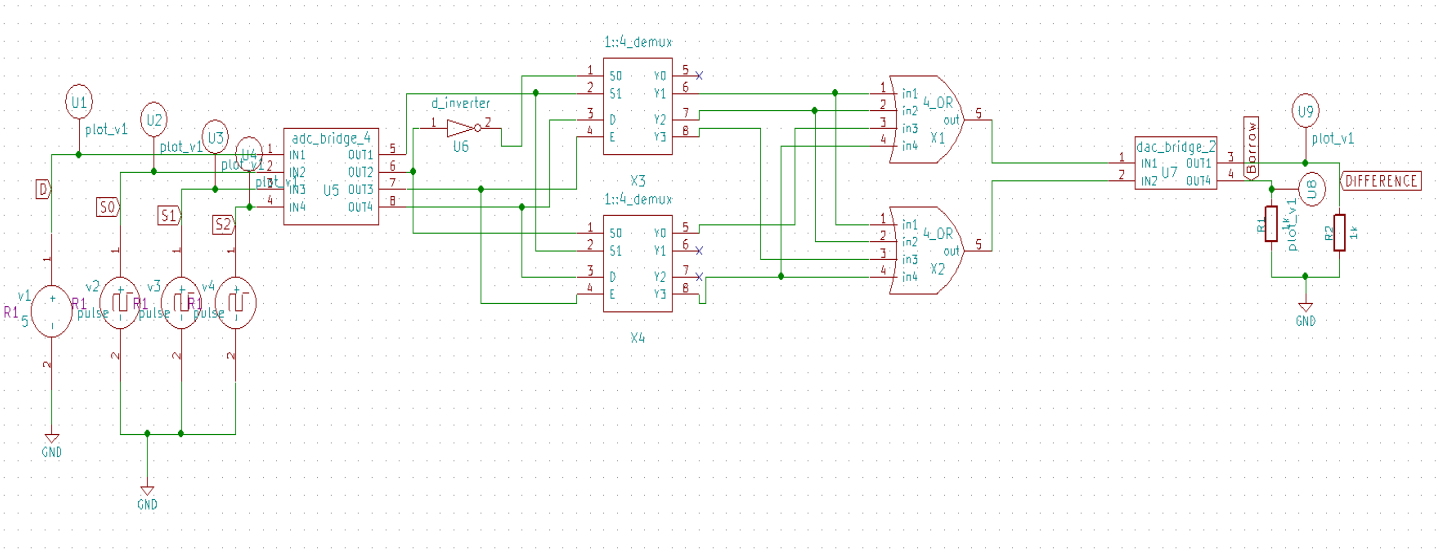
Borrow-> $B(X, Y, B_i) = \Sigma m(1, 2, 3, 7)$

Two 1x4 de multiplexers can make up a 1x8 de multiplexer.

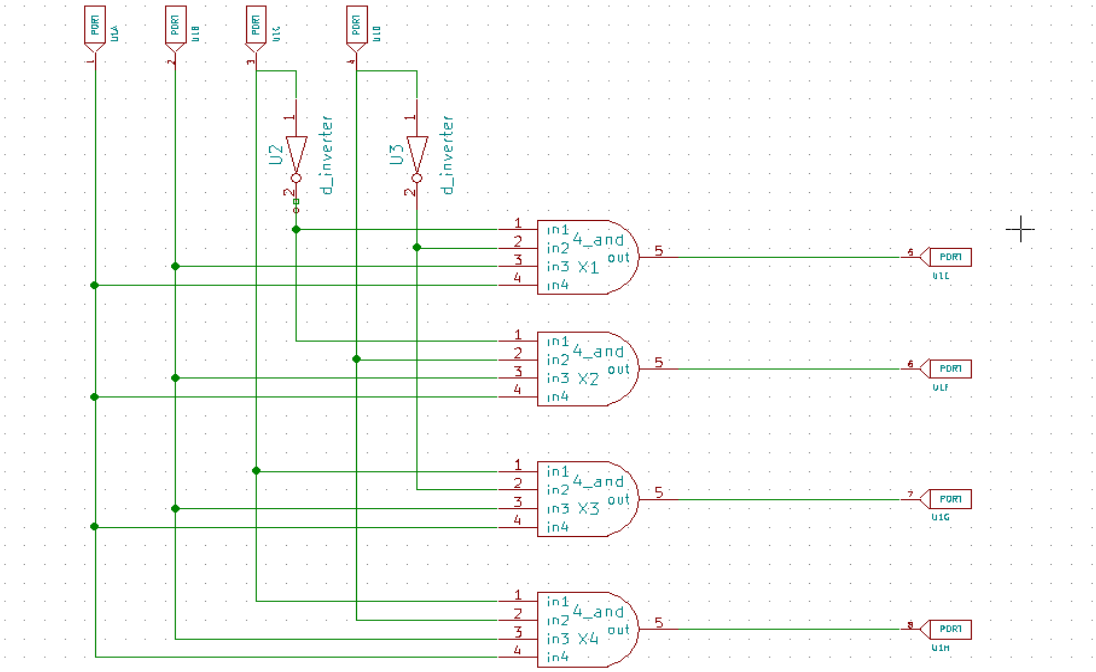


Circuit Diagrams:

The schematic diagram of full subtractor using two 1x4 demultiplexers:

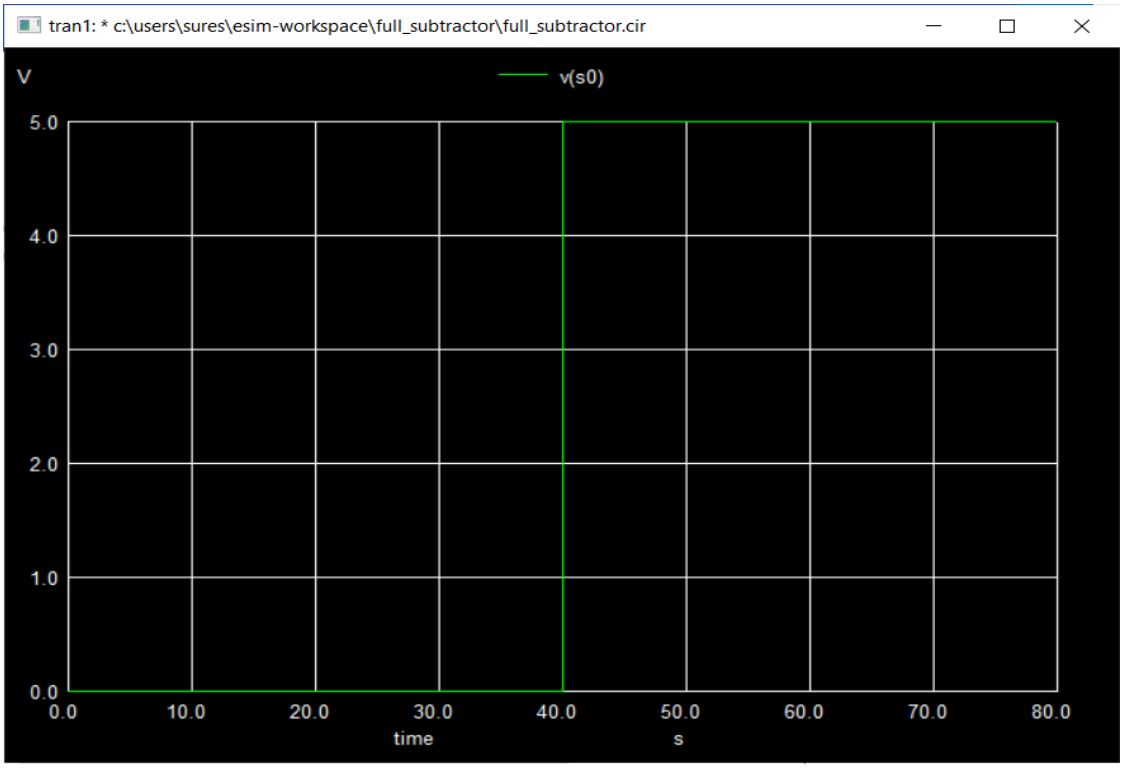


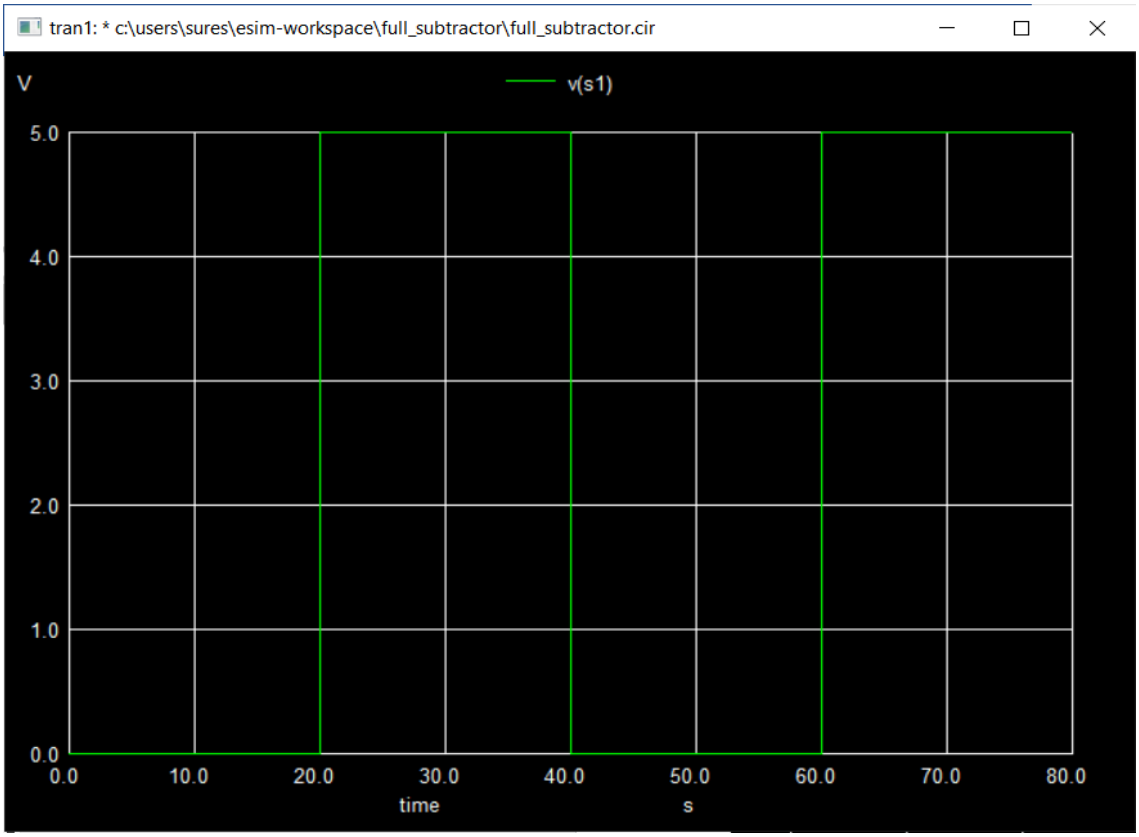
Here the sub-circuit used for 1x4 demultiplexers is:

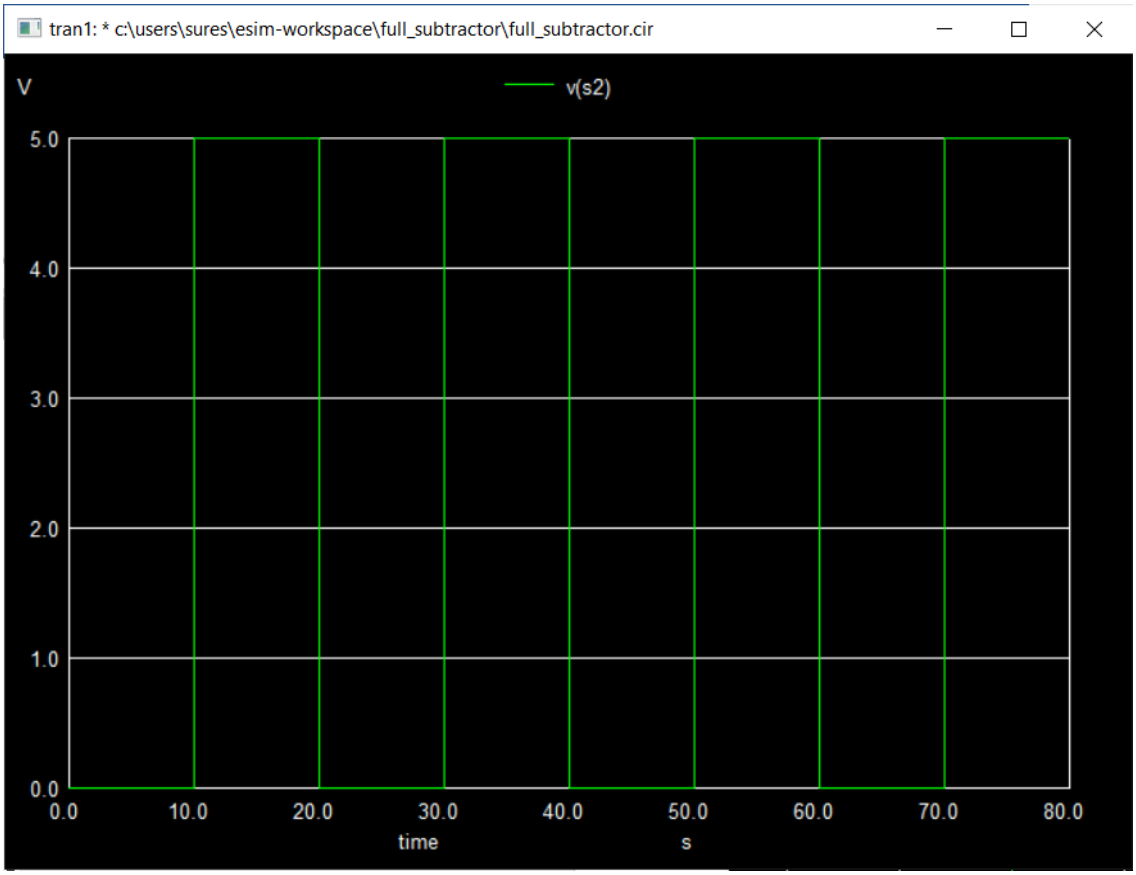


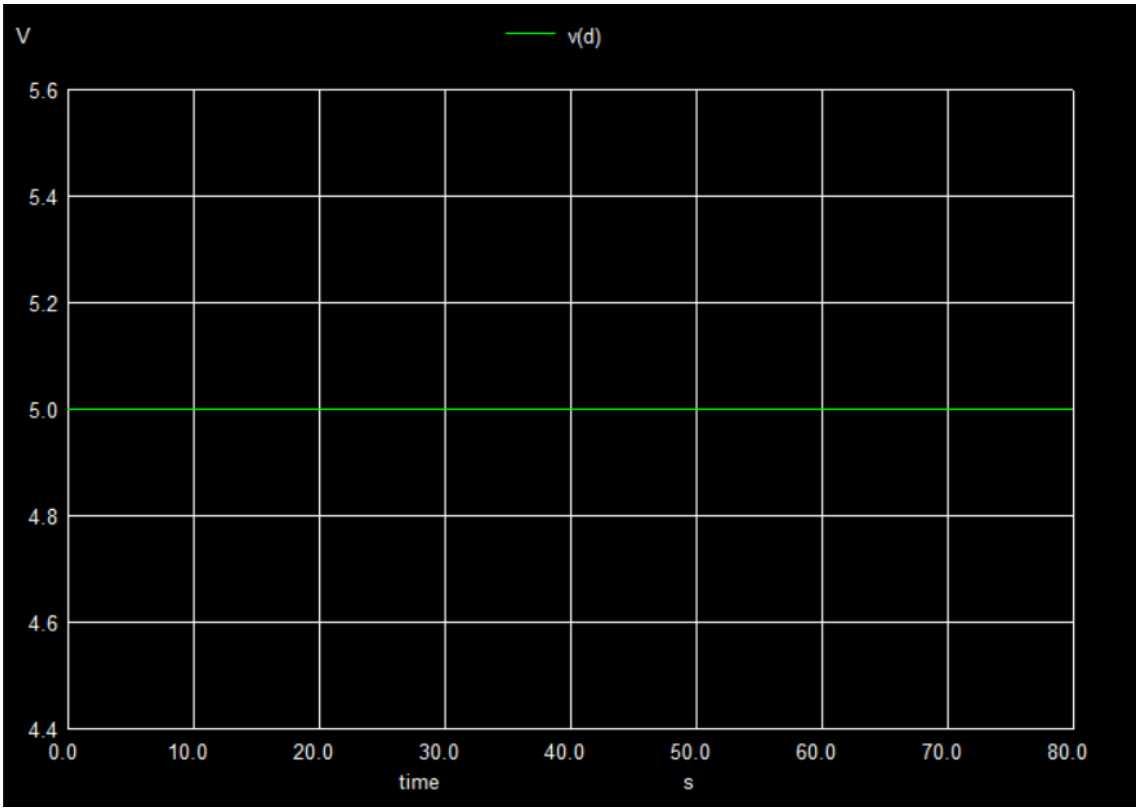
NGSPICE PLOTS

INPUT PLOTS:


tran1: * c:\users\sures\esim-workspace\full_subtractor\full_subtractor.cir
V
v(s0)
5.0
4.0
3.0
2.0
1.0
0.0
0.0
10.0
20.0
30.0
40.0
50.0
60.0
70.0
80.0
time
s

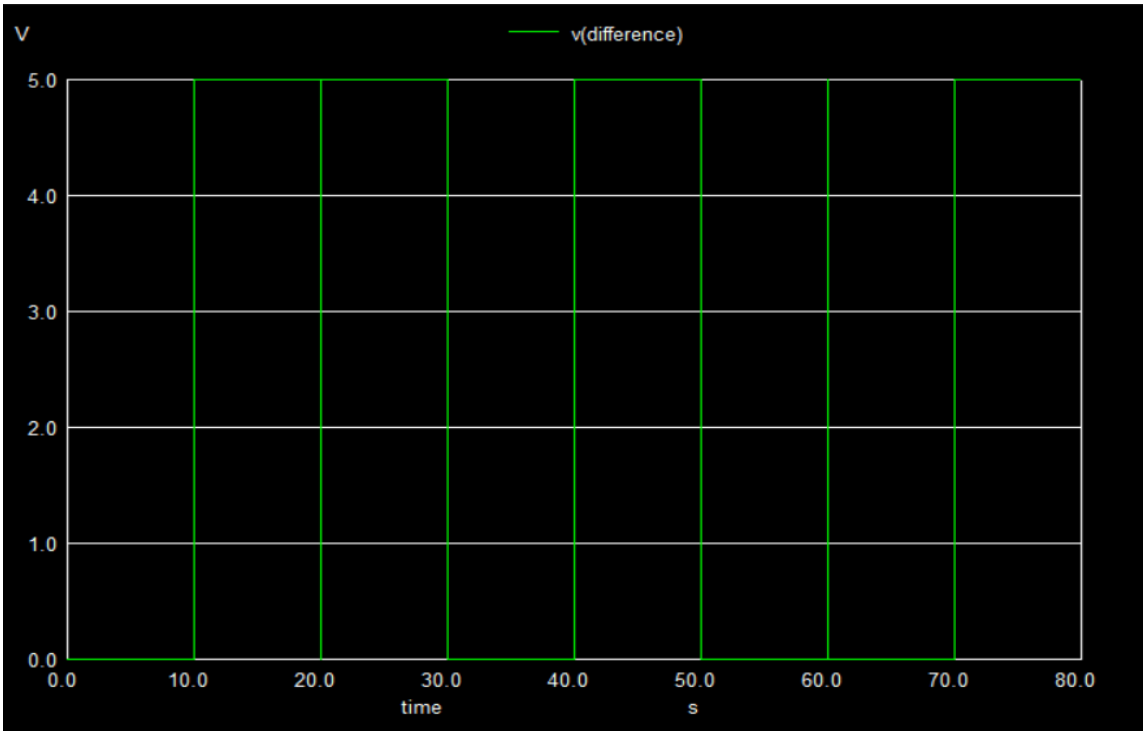

tran1: * c:\users\sures\esim-workspace\full_subtractor\full_subtractor.cir
V
v(s1)
5.0
4.0
3.0
2.0
1.0
0.0
0.0
10.0
20.0
30.0
40.0
50.0
60.0
70.0
80.0
time
s


tran1: * c:\users\sures\esim-workspace\full_subtractor\full_subtractor.cir
V
v(s2)
5.0
4.0
3.0
2.0
1.0
0.0
0.0
10.0
20.0
30.0
40.0
50.0
60.0
70.0
80.0
time
s

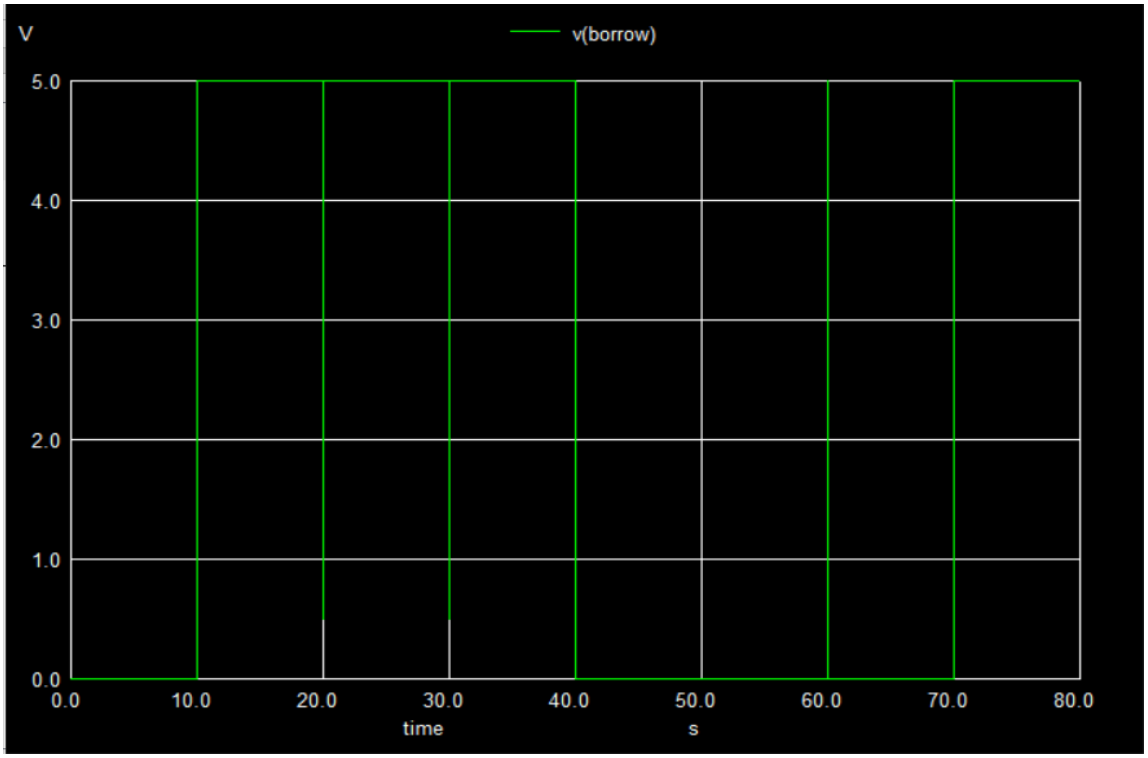

V
v(d)
5.6
5.4
5.2
5.0
4.8
4.6
4.4
0.0
10.0
20.0
30.0
40.0
50.0
60.0
70.0
80.0
time
s

OUTPUT PLOTS:

DIFFERENCE



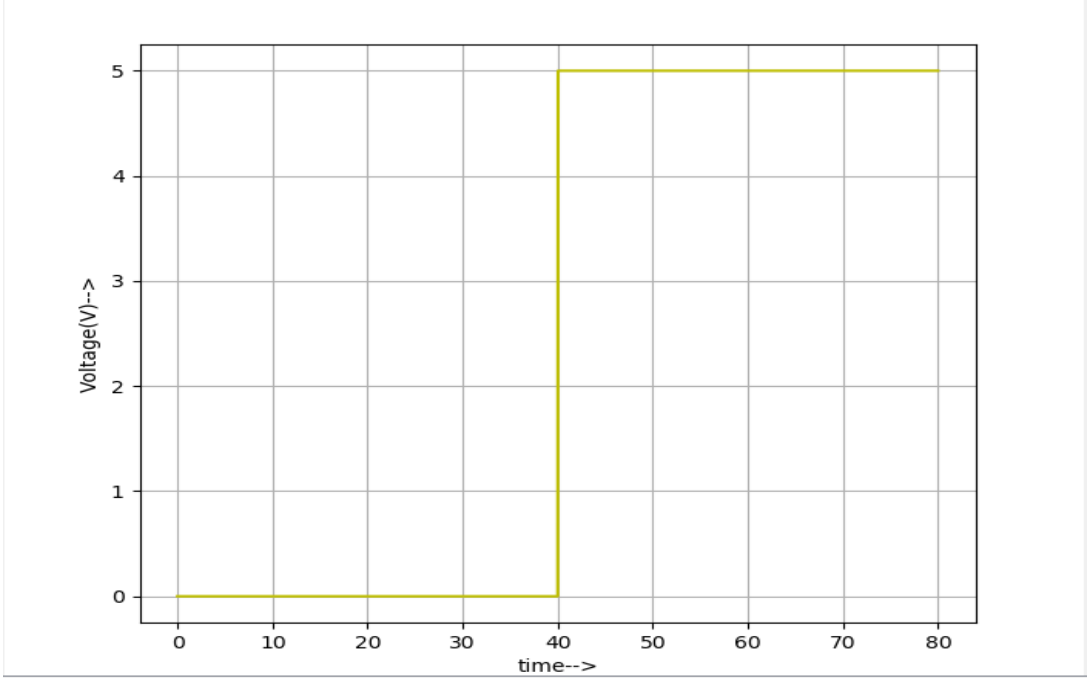
BORROW



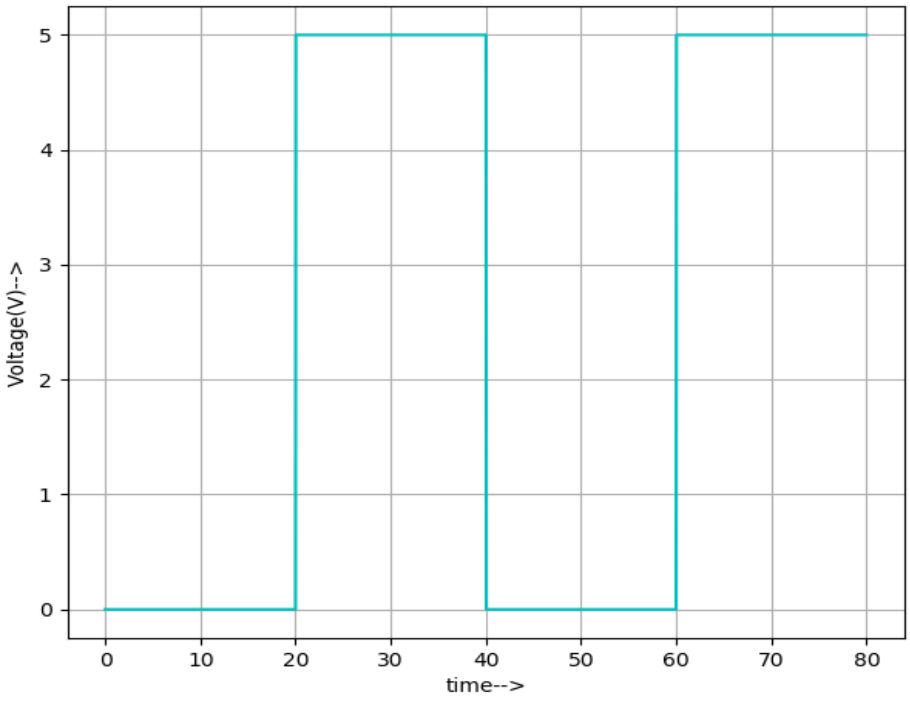
PYTHON PLOTS:

INPUT

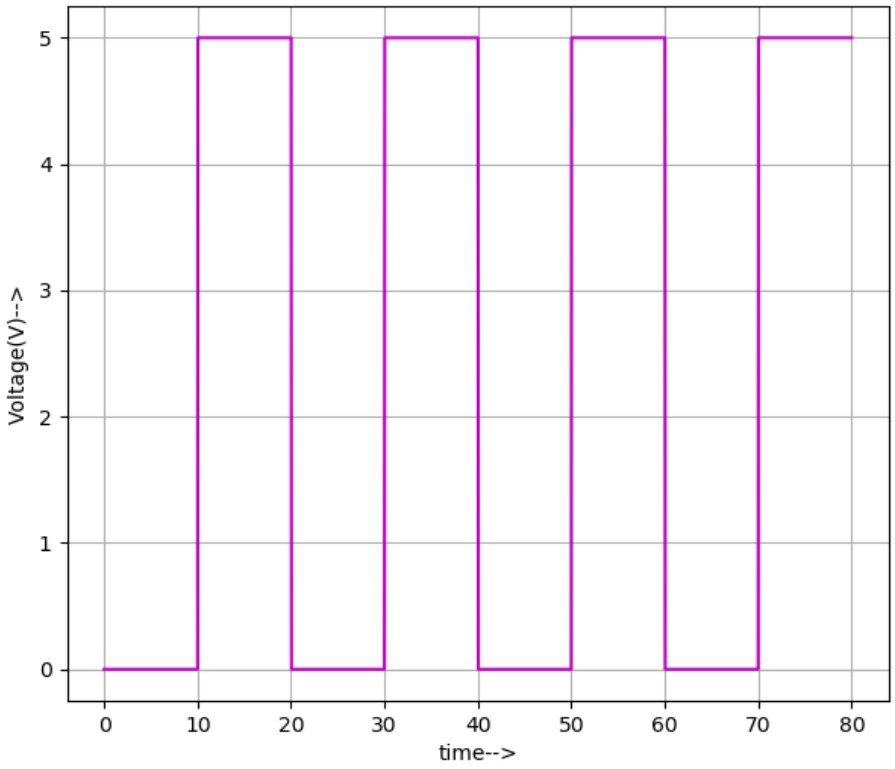
1. S0



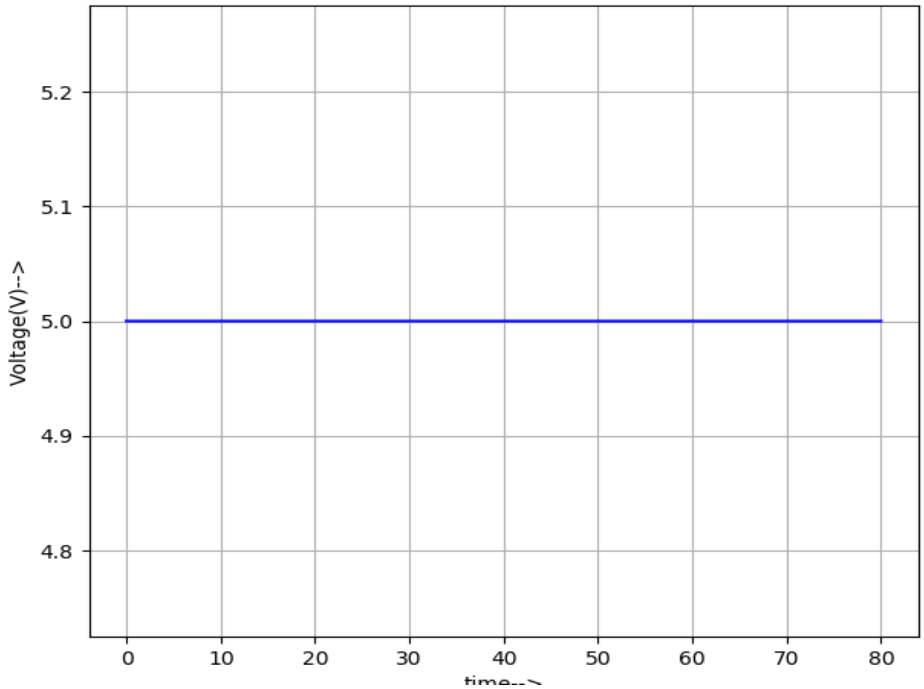
2. S1



3. S2

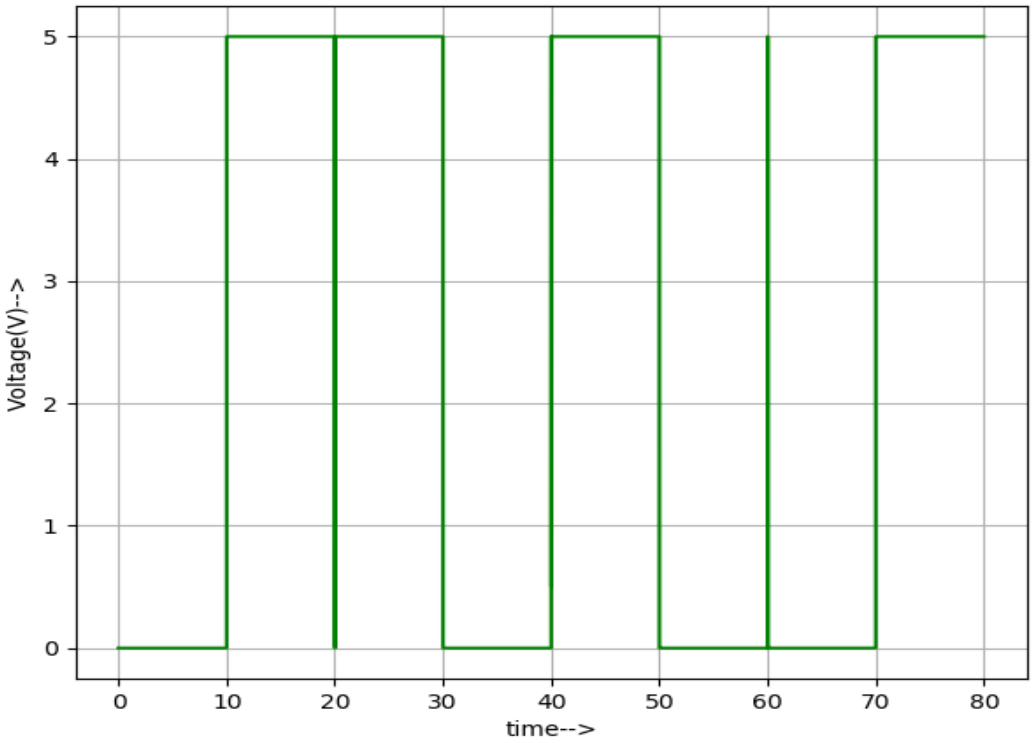

Voltage(V)-->
time-->
0
1
2
3
4
5
0
10
20
30
40
50
60
70
80

4. D (INPUT)


Voltage(V)-->
4.8
4.9
5.0
5.1
5.2
0
10
20
30
40
50
60
70
80

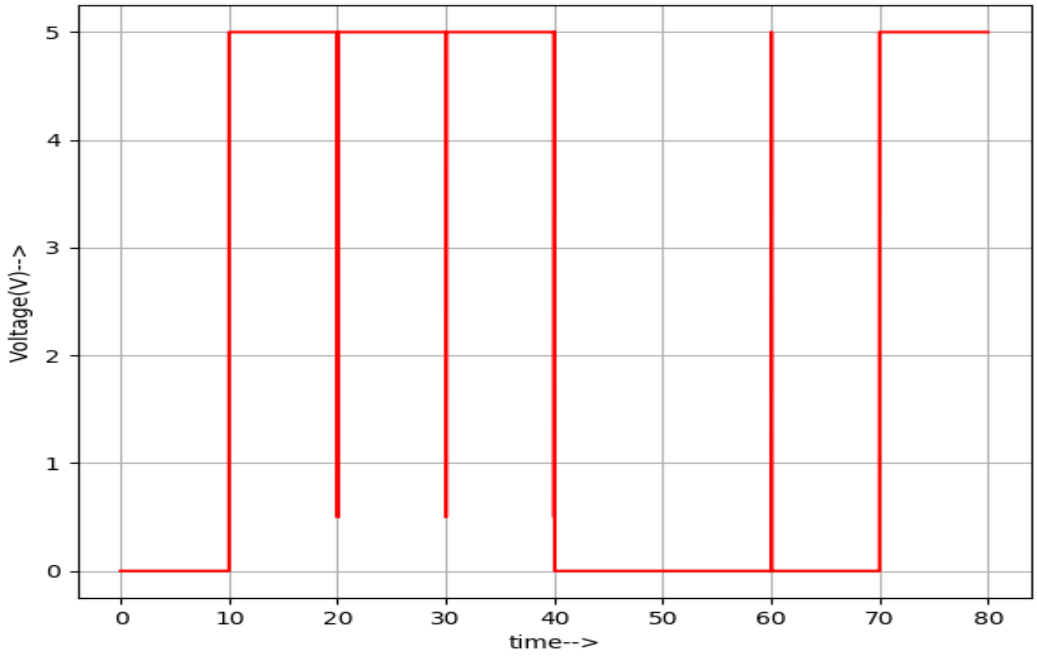
OUTPUT:

DIFFERENCE



The required waveform for the difference is same as the truth table.

BORROW



The required waveform for borrow is same as truth table.

Reference:

https://www.geeksforgeeks.org/full-subtractor-in-digital-logic/

https://electronics-fun.com/demultiplexer-and-its-application/

https://www.youtube.com/watch?v=-9s3xBlcBVM

https://www.electronicshub.org/demultiplexerdemux/