Universal Shift Register Using SR Flip Flop

Tabish Ahmad
Zakir Hussain College of Engineering and Technology, Aligarh Muslim University

THEORY:

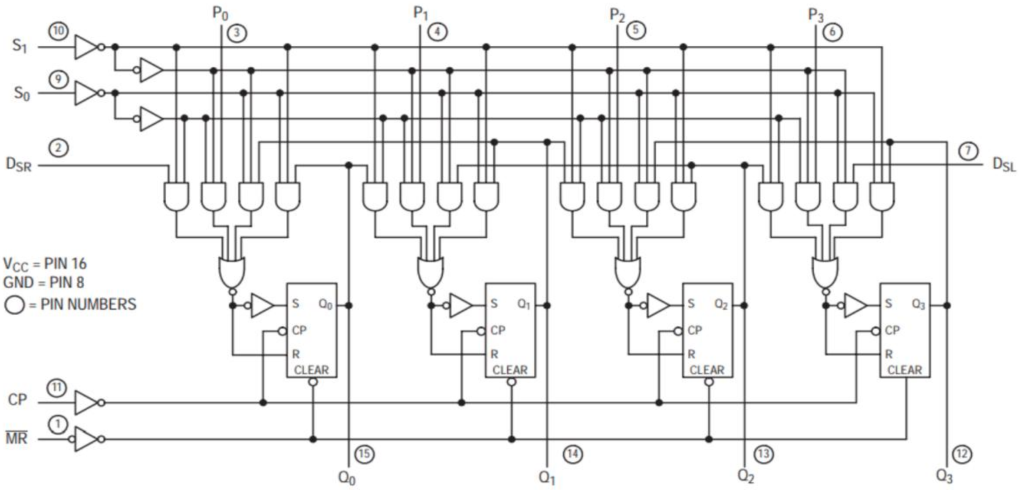
Universal shift register is a sequential circuit made of basic SR flip flops and is used to store data or shift the data. In this type of register we have basically 3 main operations that are parallel loading of data, right shift of data, or left shift of data. While shift operation, data is loaded serially through DSR and DSL inputs (in circuit diagram) during right shift and left shift respectively. For these operations this circuit has select lines to select a particular operation to be performed. Below is the table showing assigned operation for each select input.

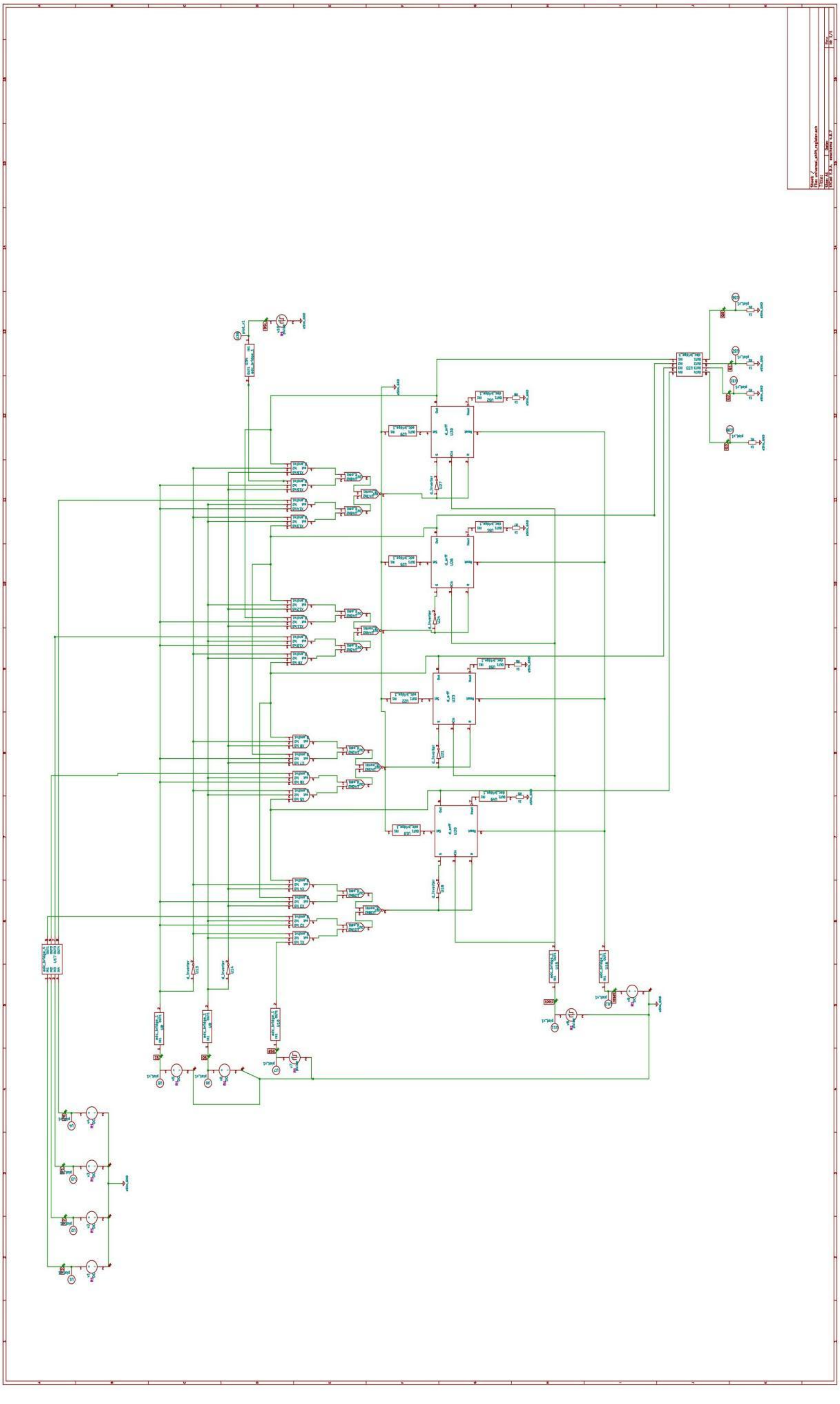
Mode Control		**Register Operation**
S_1	*S_0*	
0	0	No change
0	1	Shift Right
1	0	Shift Left
1	1	Parallel Load

For select input ‘00’ hold operation takes place i.e. whatever data is stored at output will remain there unchanged.
For select input ‘01’ right shift operation occur and data is shifted right bit by bit at every positive going transition of clock (for positive edge triggered flip flop).
For select input ‘10’ left shift operation occur and data is shifted left bit by bit at every positive going transition of clock (for positive edge triggered flip flop).
For select input ‘11’ parallel loading takes place and whatever data is at input, gets loaded at output parallelly.

All flip flops used in this circuit are synchronous with each other as a single clock is connected to every flip flop.

CIRCUIT DIAGRAM:





ESim schematic

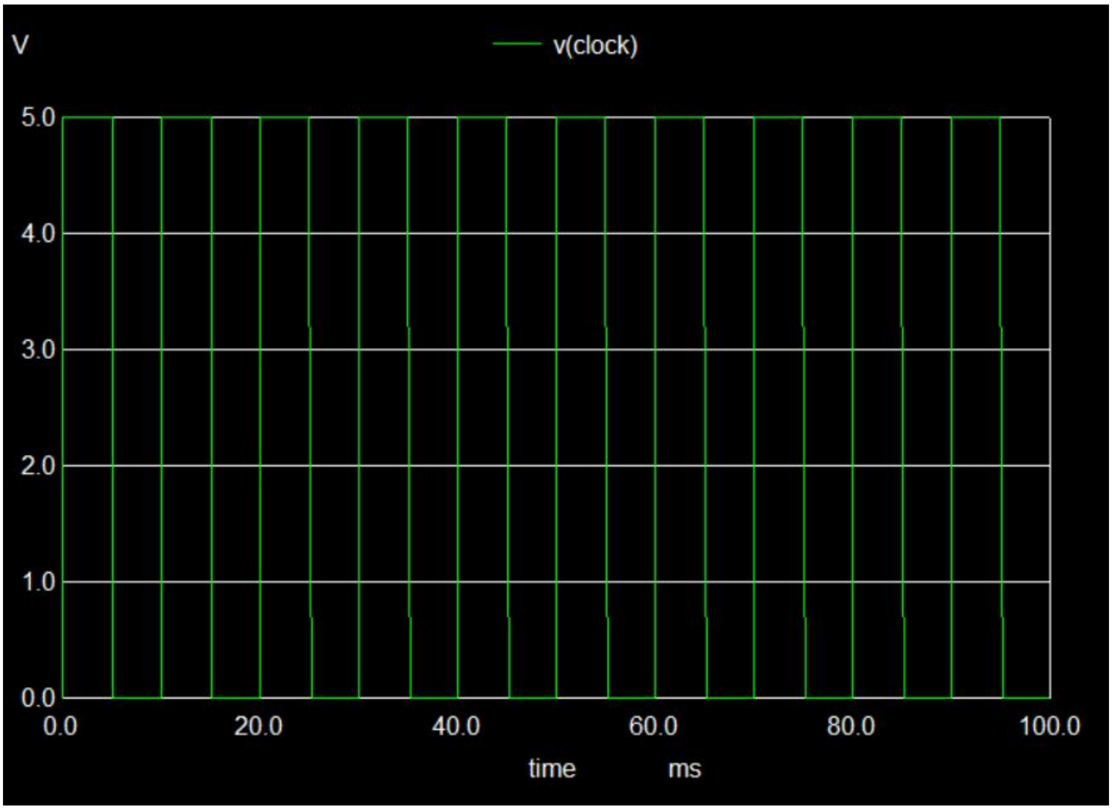
TERMS USED IN ESIM SCHEMATIC:

P0,P1,P2,P3	INPUT DATA(P0[LSB], P3[MSB])
q0,q1,q2,q3	OUTPUT DATA(q0[LSB], q3[MSB])
Clock	Clock input(positve edge triggered)
Reset	Reset command
S0,S1	Select input
Dsr	Input for right shift
Dsl	Input for left shift

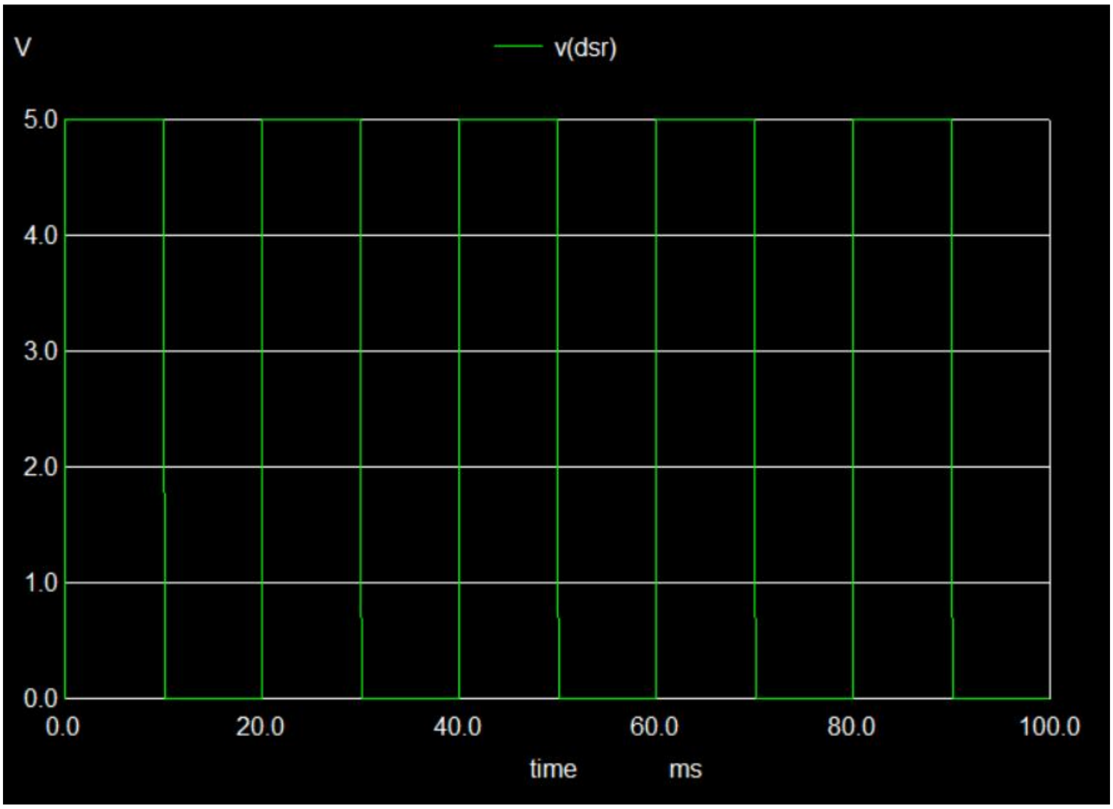
RESULTS:

1) NgSpice plots

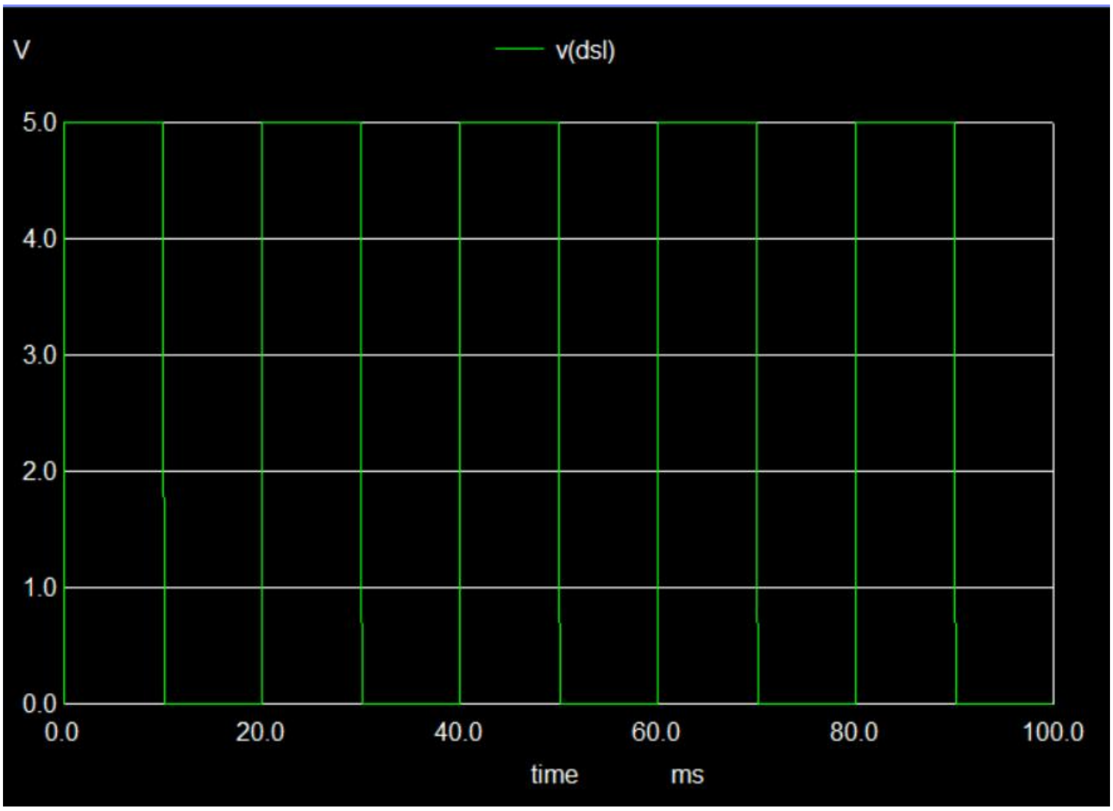
CLOCK



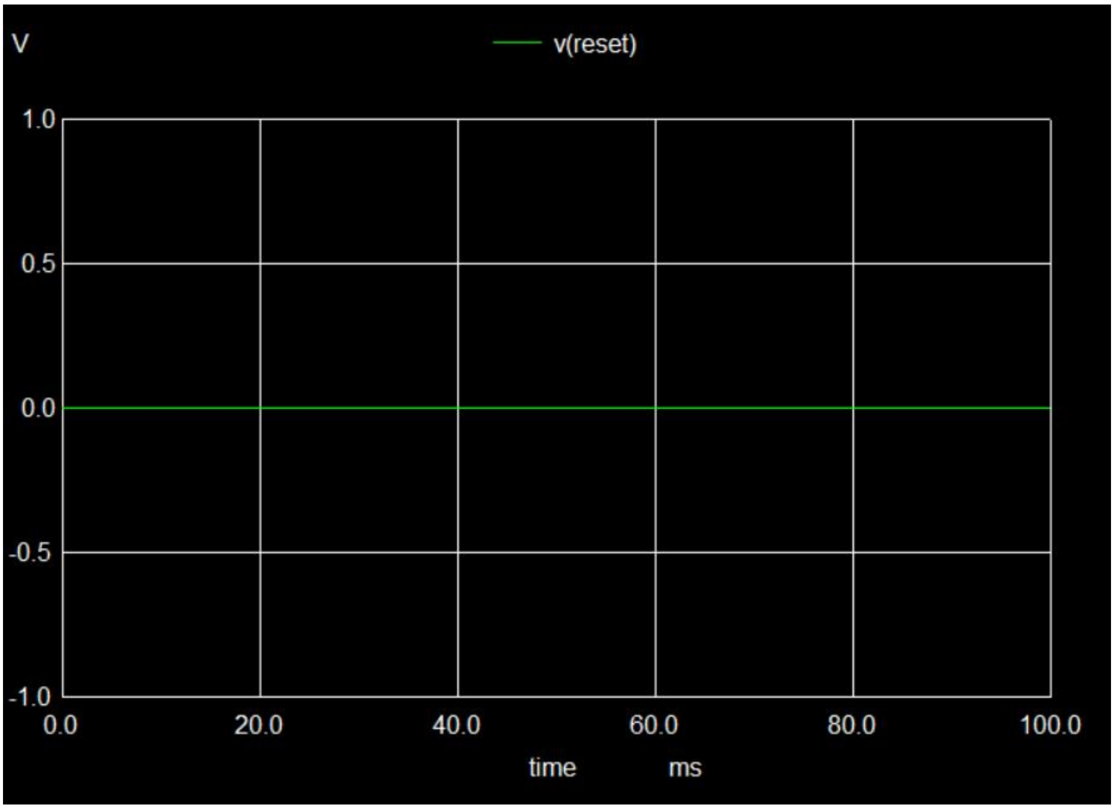
Dsr


V
v(dsr)
5.0
4.0
3.0
2.0
1.0
0.0
0.0
20.0
40.0
60.0
80.0
100.0
time
ms

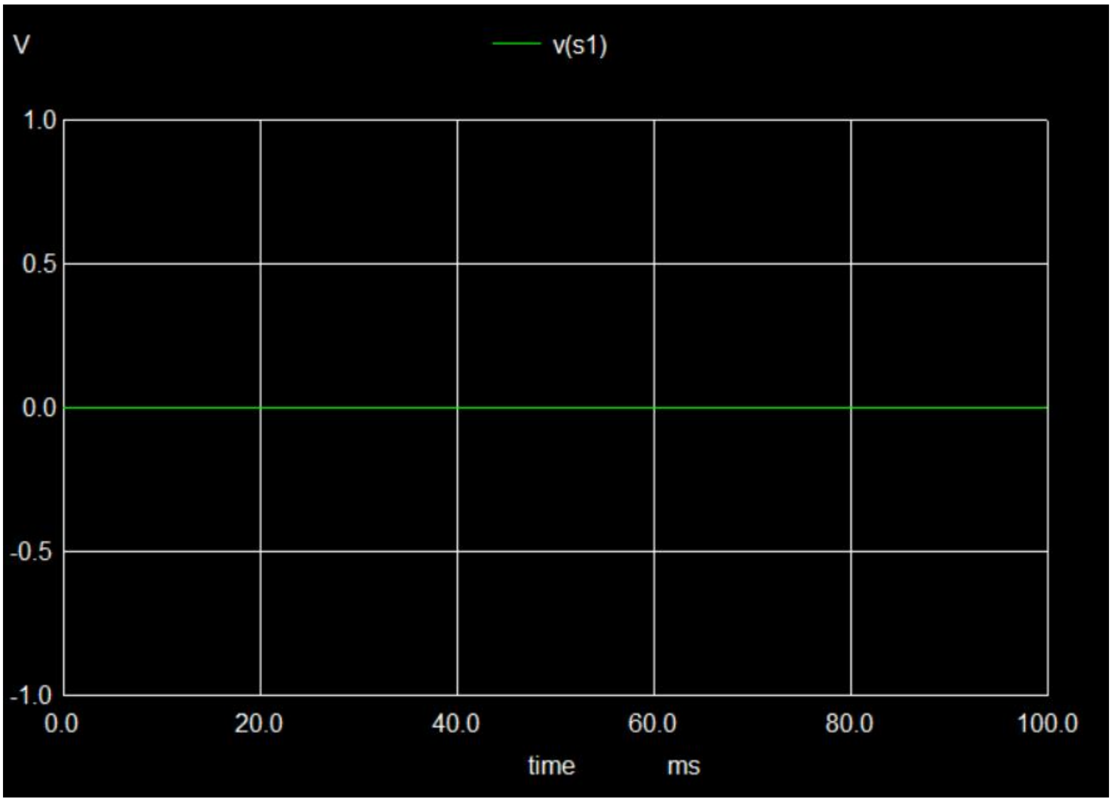
Dsl

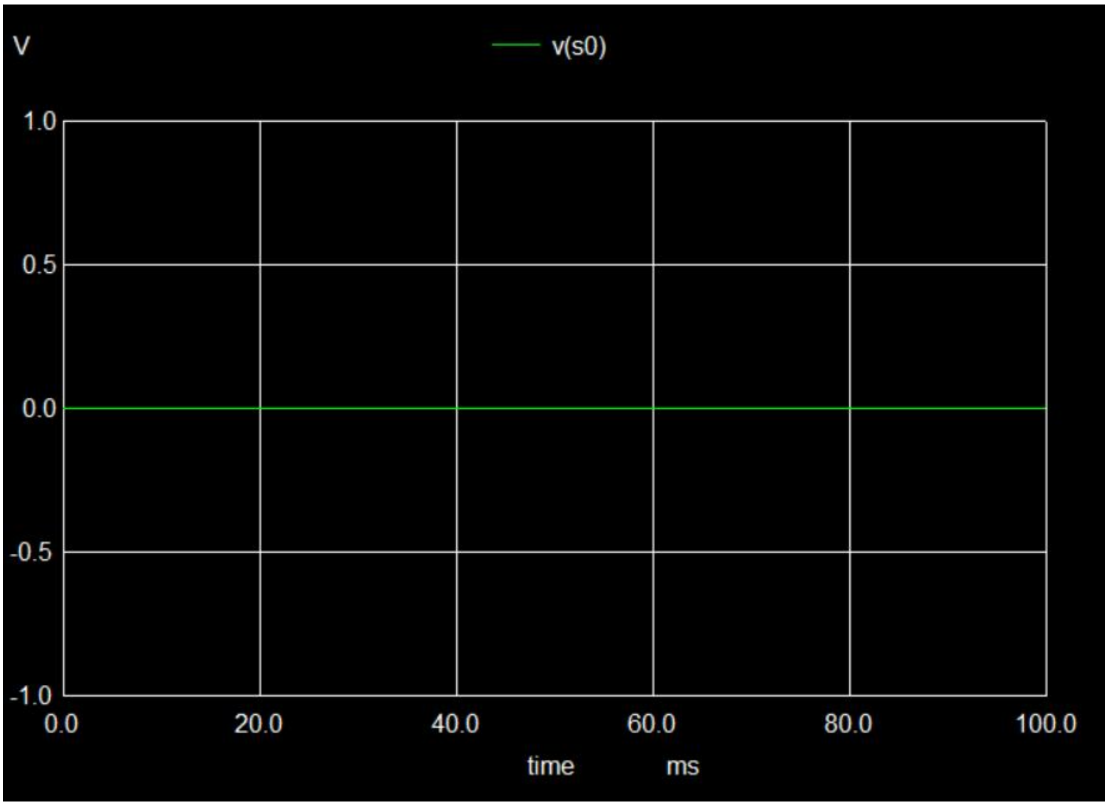

V
v(dsl)
5.0
4.0
3.0
2.0
1.0
0.0
0.0
20.0
40.0
60.0
80.0
100.0
time
ms

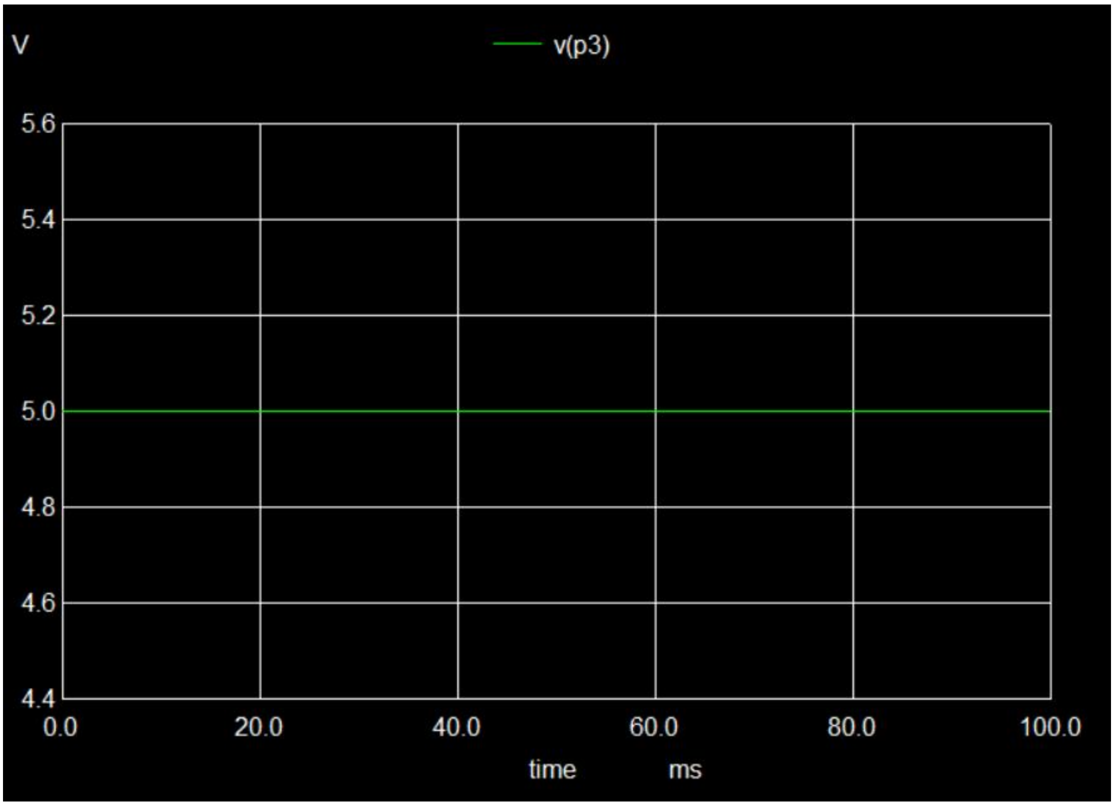
RESET

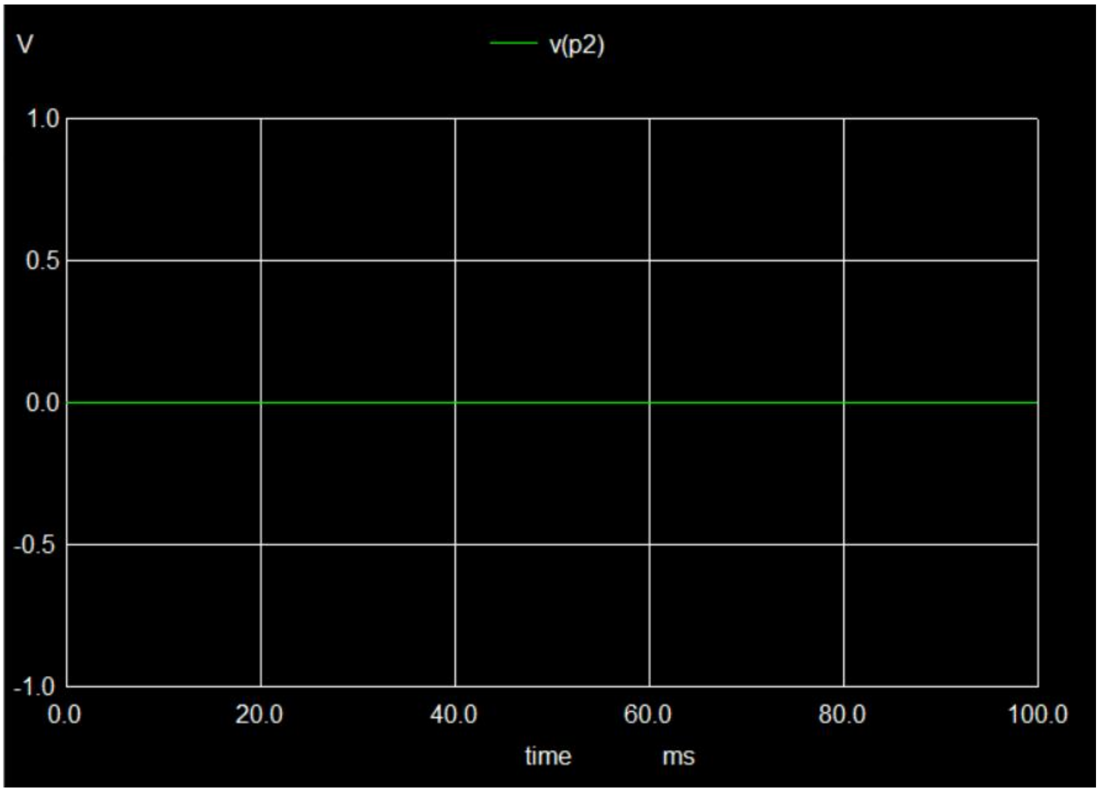


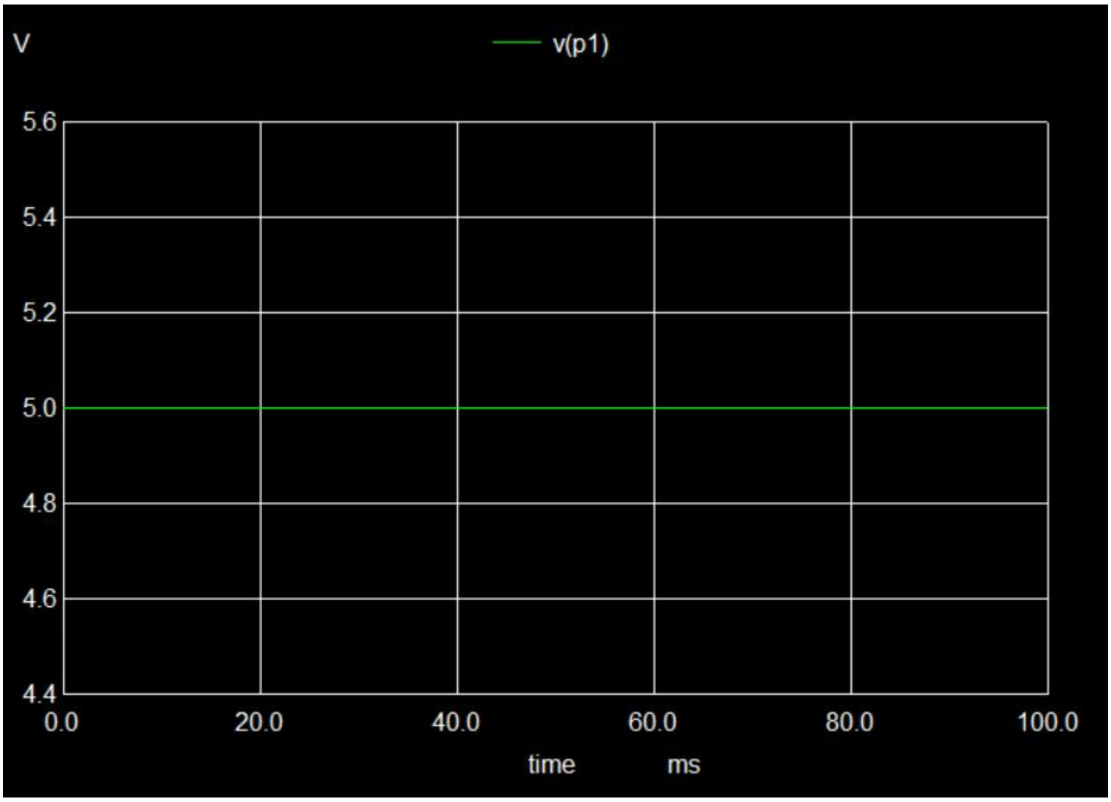
A) For input data - 1010
Select input - 00 (NO CHANGE)
Reset - 0

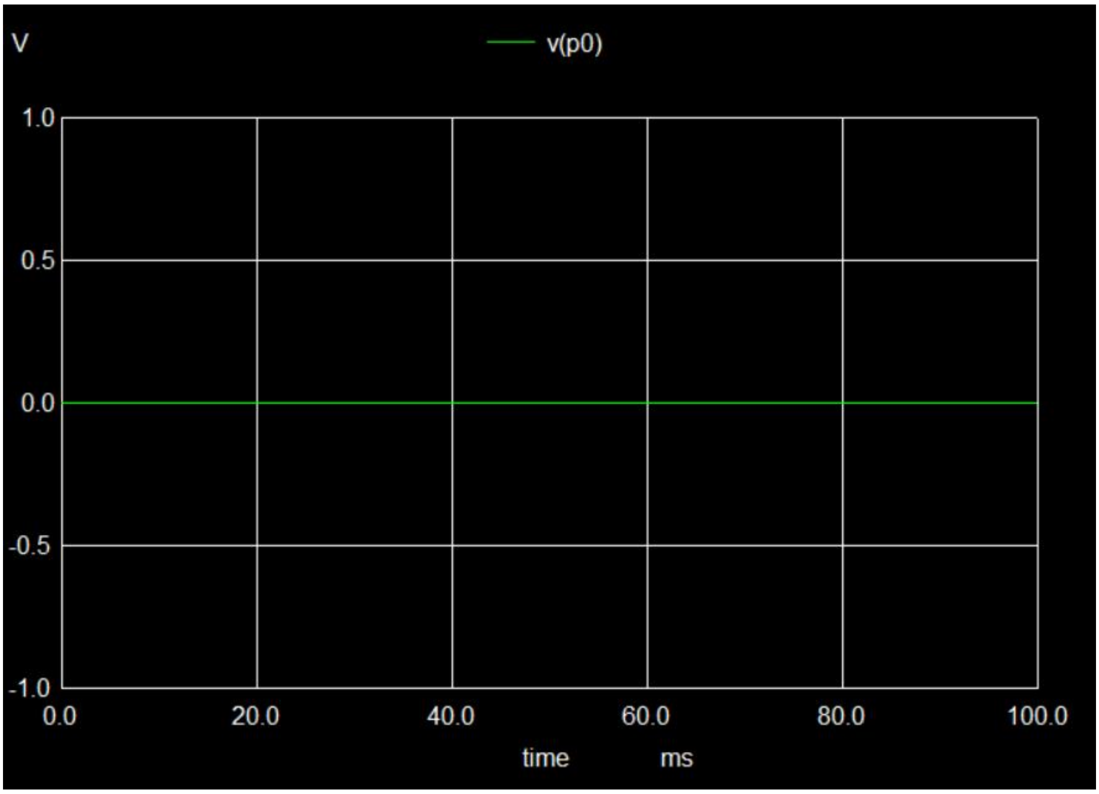


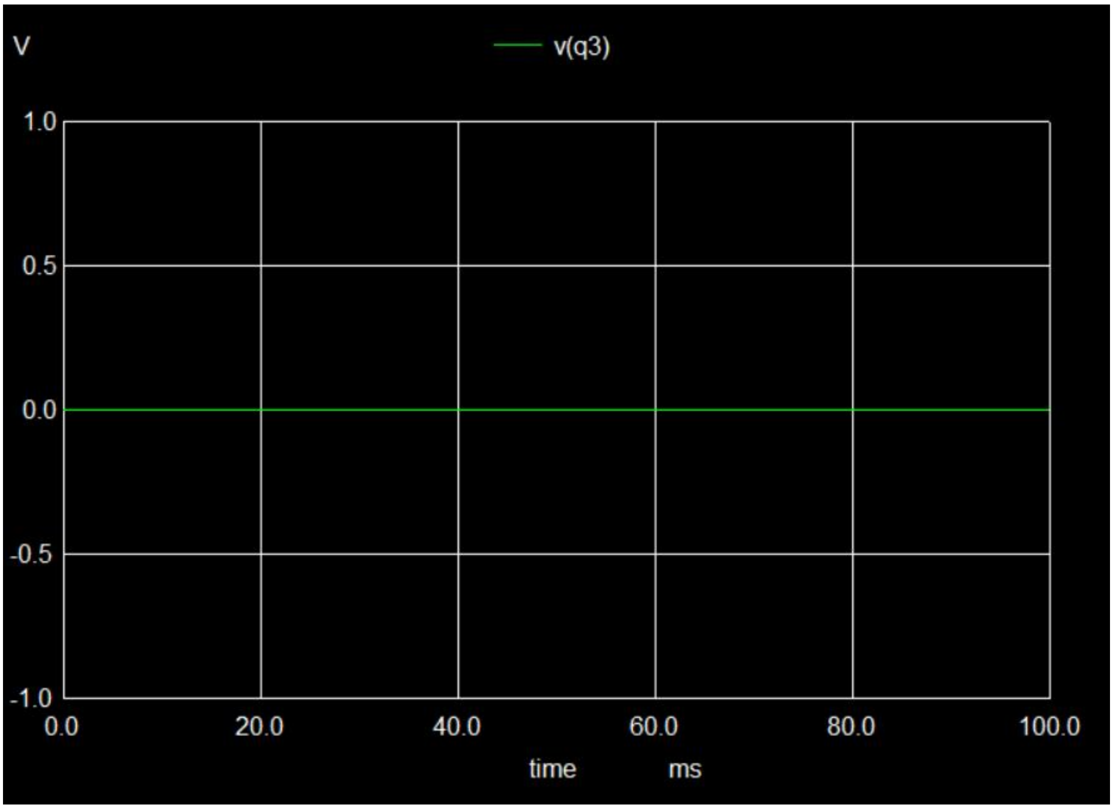

V
v(s0)
1.0
0.5
0.0
-0.5
-1.0
0.0
20.0
40.0
60.0
80.0
100.0
time
ms

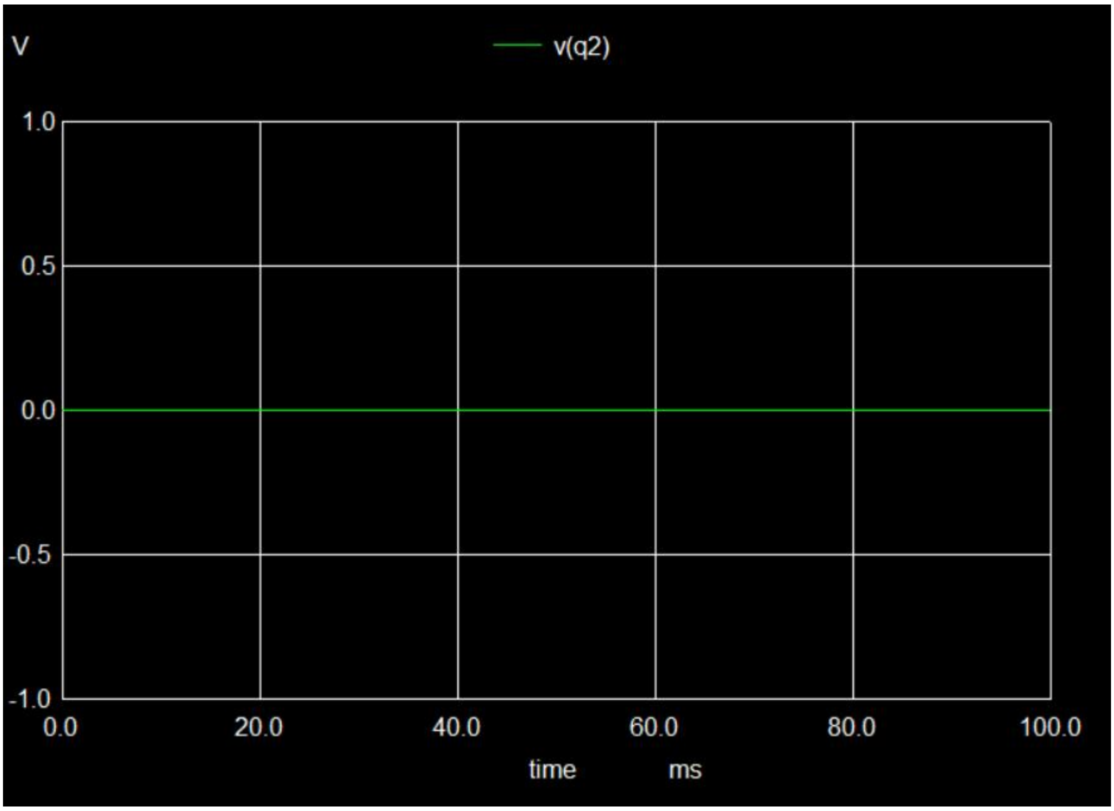

V
v(p3)
5.6
5.4
5.2
5.0
4.8
4.6
4.4
0.0
20.0
40.0
60.0
80.0
100.0
time
ms

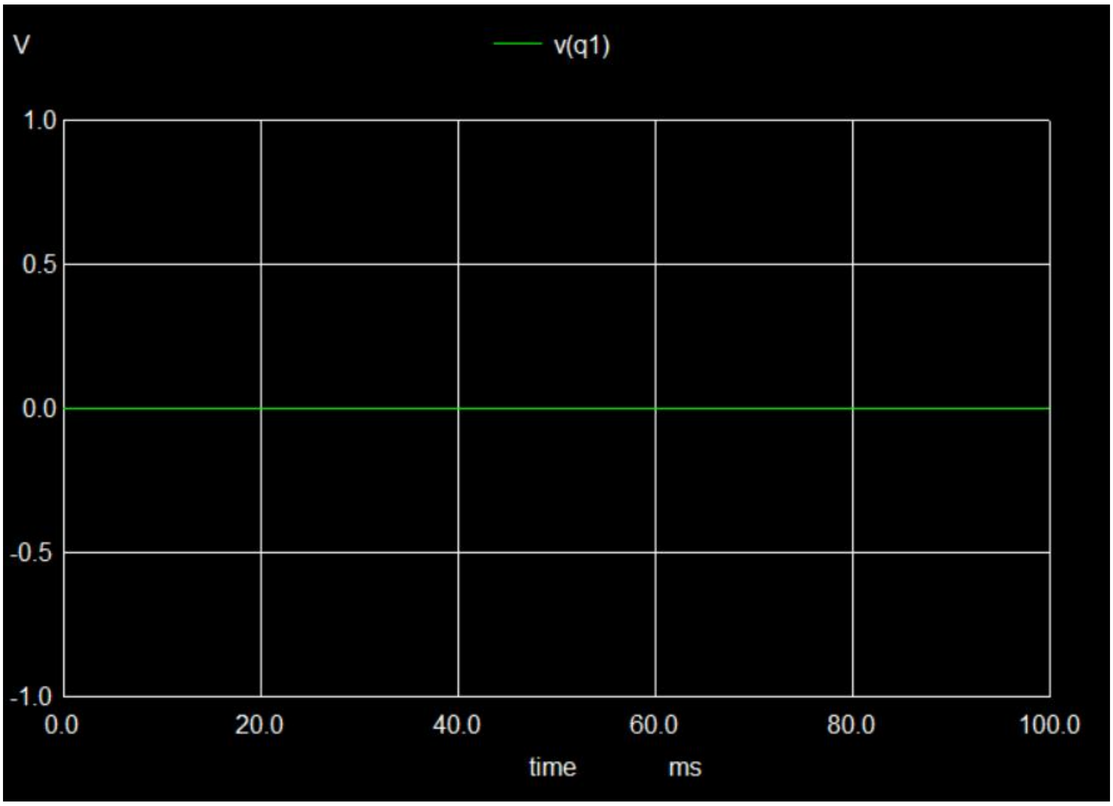

V
v(p2)
1.0
0.5
0.0
-0.5
-1.0
0.0
20.0
40.0
60.0
80.0
100.0
time
ms

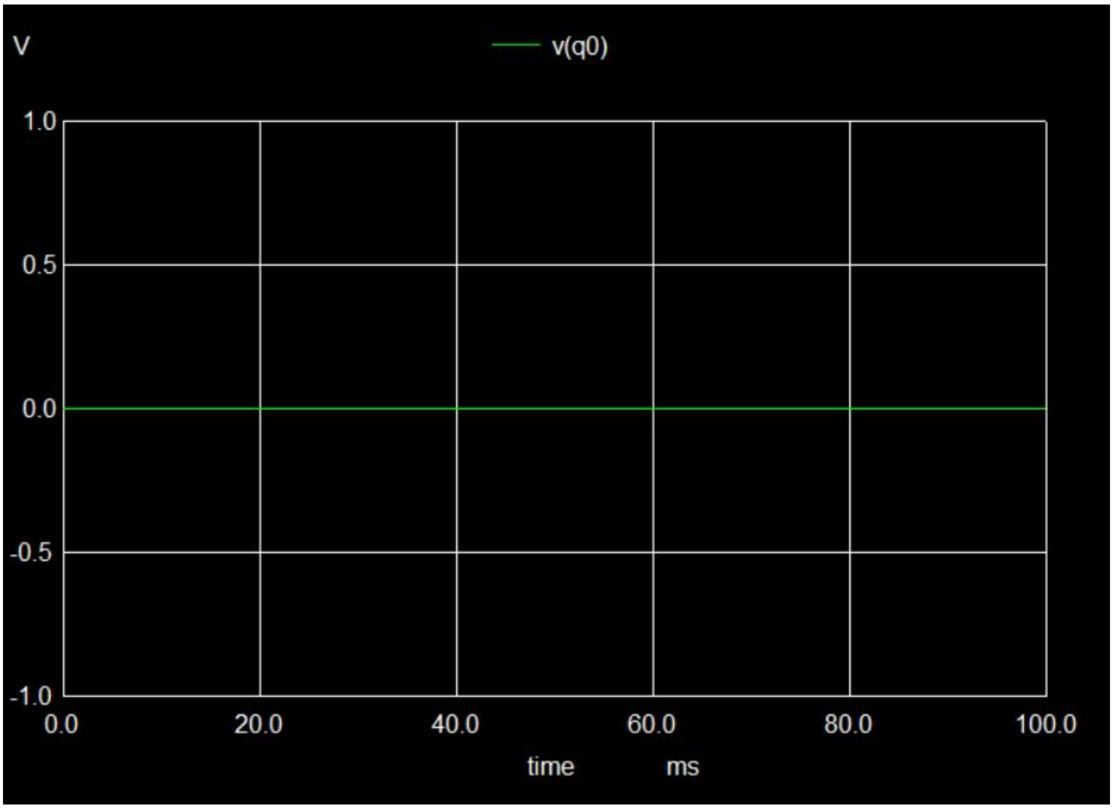

V
v(p1)
5.6
5.4
5.2
5.0
4.8
4.6
4.4
0.0
20.0
40.0
60.0
80.0
100.0
time
ms


V
v(p0)
1.0
0.5
0.0
-0.5
-1.0
0.0
20.0
40.0
60.0
80.0
100.0
time
ms

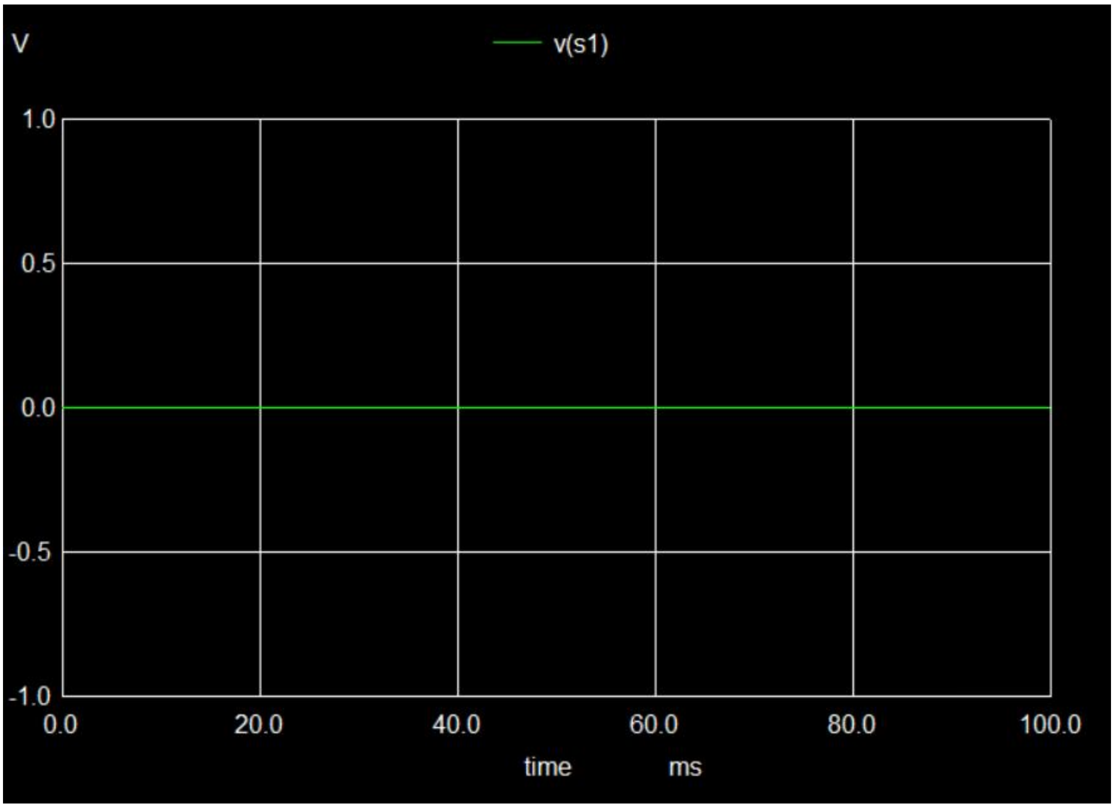

V
v(q3)
1.0
0.5
0.0
-0.5
-1.0
0.0
20.0
40.0
60.0
80.0
100.0
time
ms

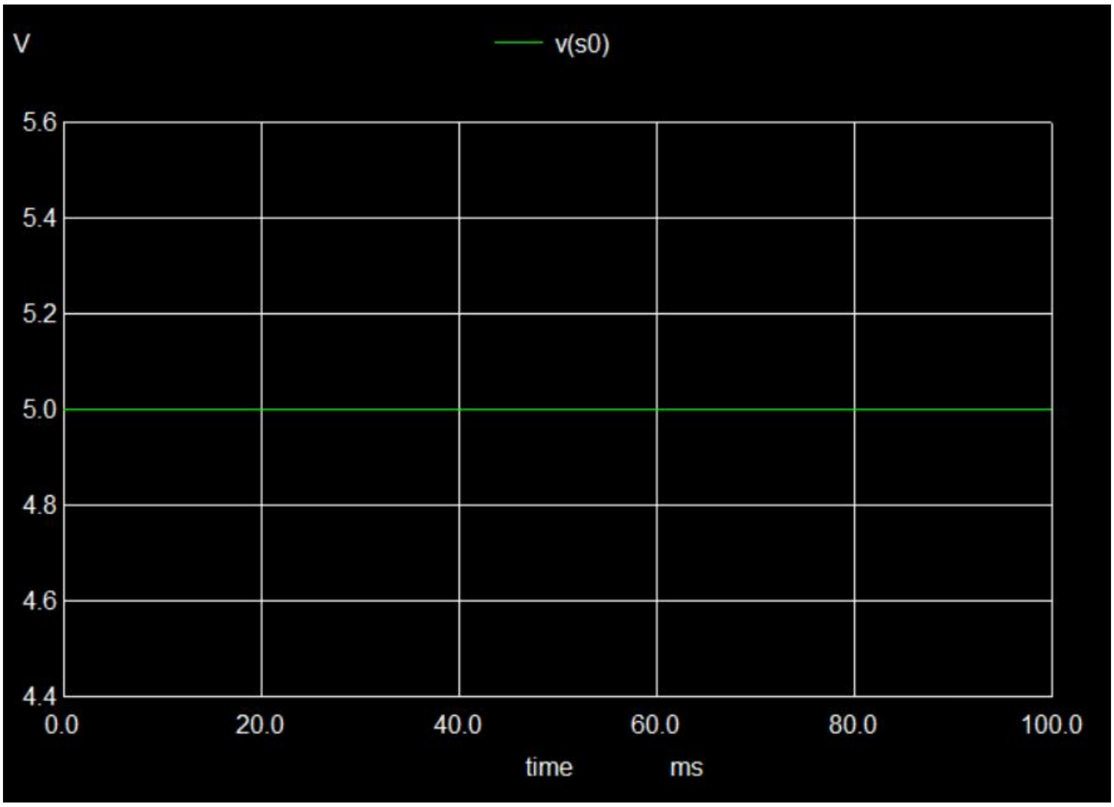

V
v(q2)
1.0
0.5
0.0
-0.5
-1.0
0.0
20.0
40.0
60.0
80.0
100.0
time
ms

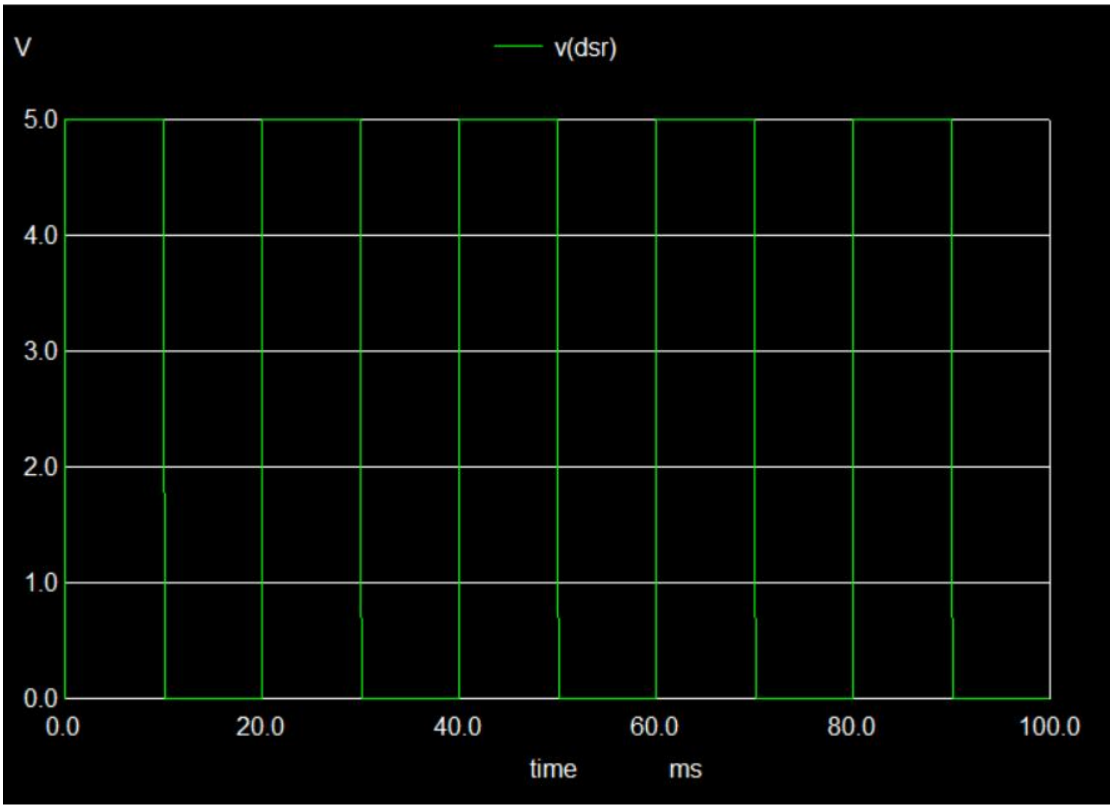

V
v(q1)
1.0
0.5
0.0
-0.5
-1.0
0.0
20.0
40.0
60.0
80.0
100.0
time
ms

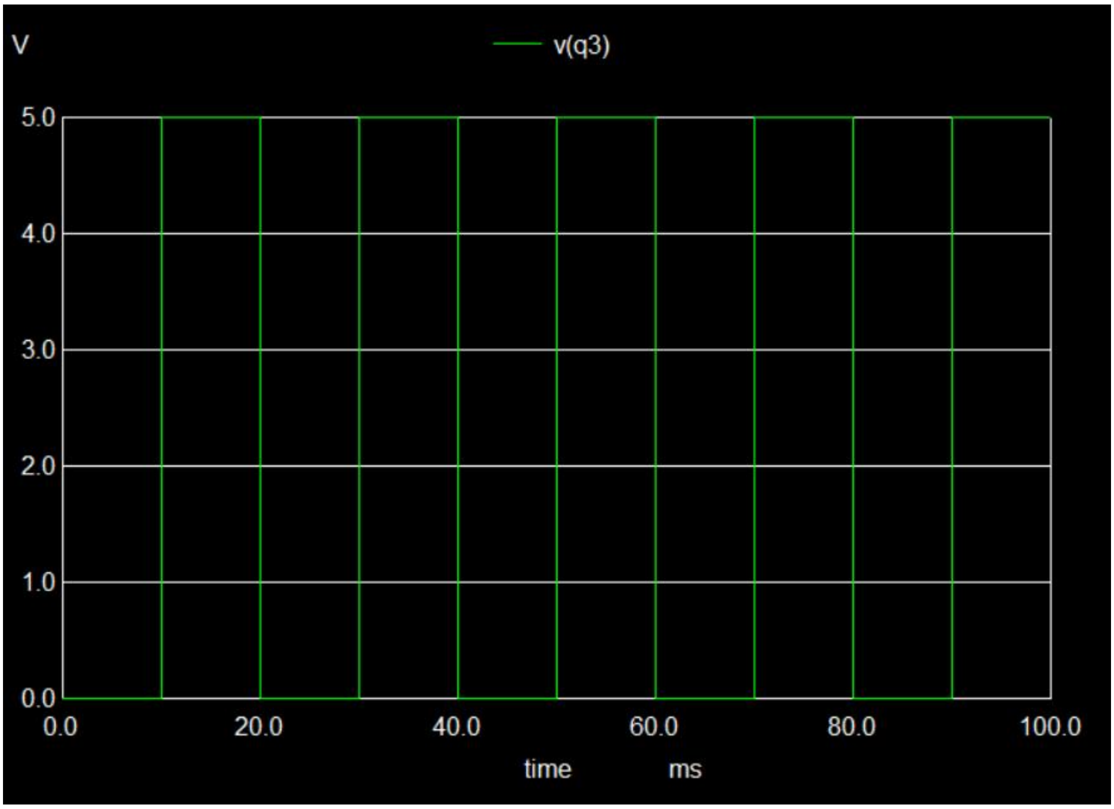


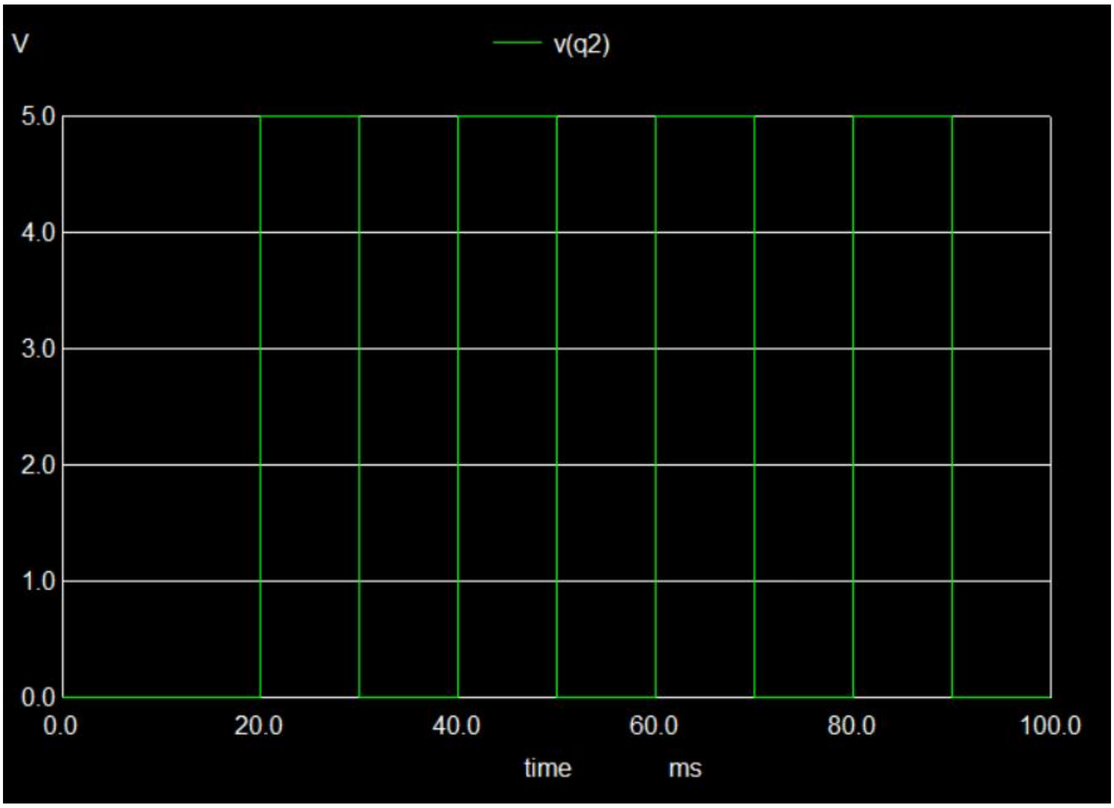
B) For input data - 1010
Select input - 01 (RIGHT SHIFT)
Reset - 0

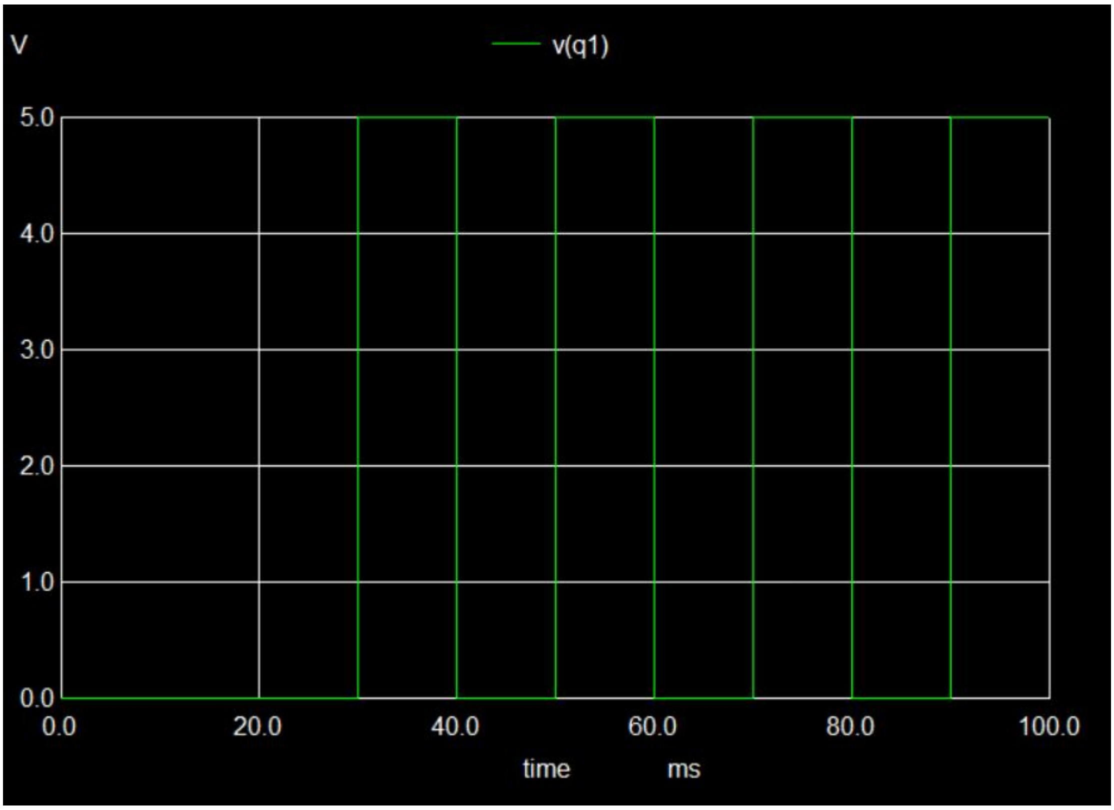


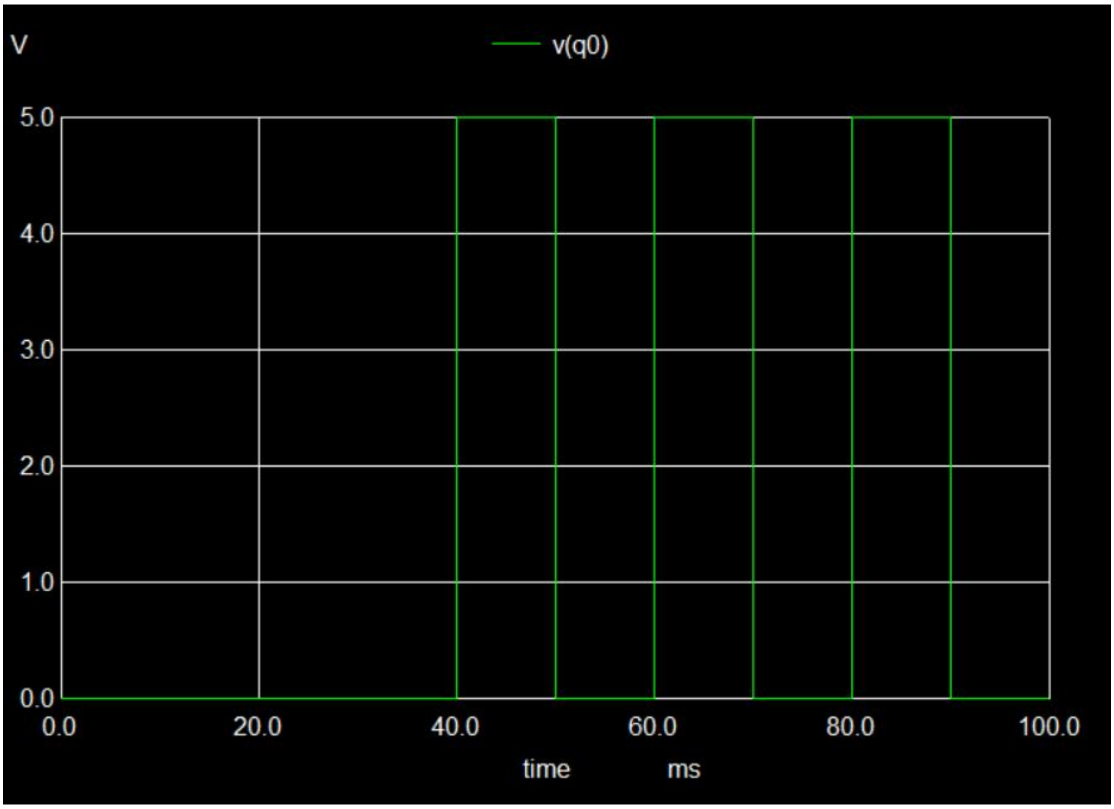

V
v(s0)
5.6
5.4
5.2
5.0
4.8
4.6
4.4
0.0
20.0
40.0
60.0
80.0
100.0
time
ms


V
v(dsr)
5.0
4.0
3.0
2.0
1.0
0.0
0.0
20.0
40.0
60.0
80.0
100.0
time
ms

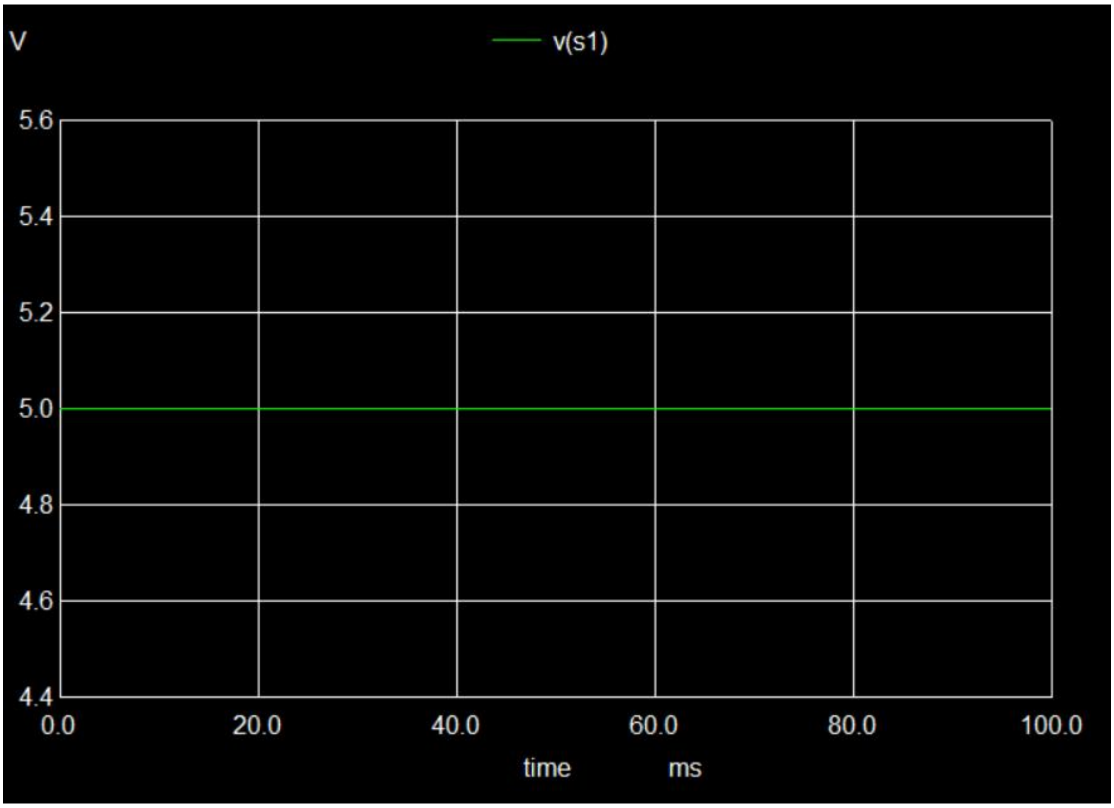

V
v(q3)
5.0
4.0
3.0
2.0
1.0
0.0
0.0
20.0
40.0
60.0
80.0
100.0
time
ms

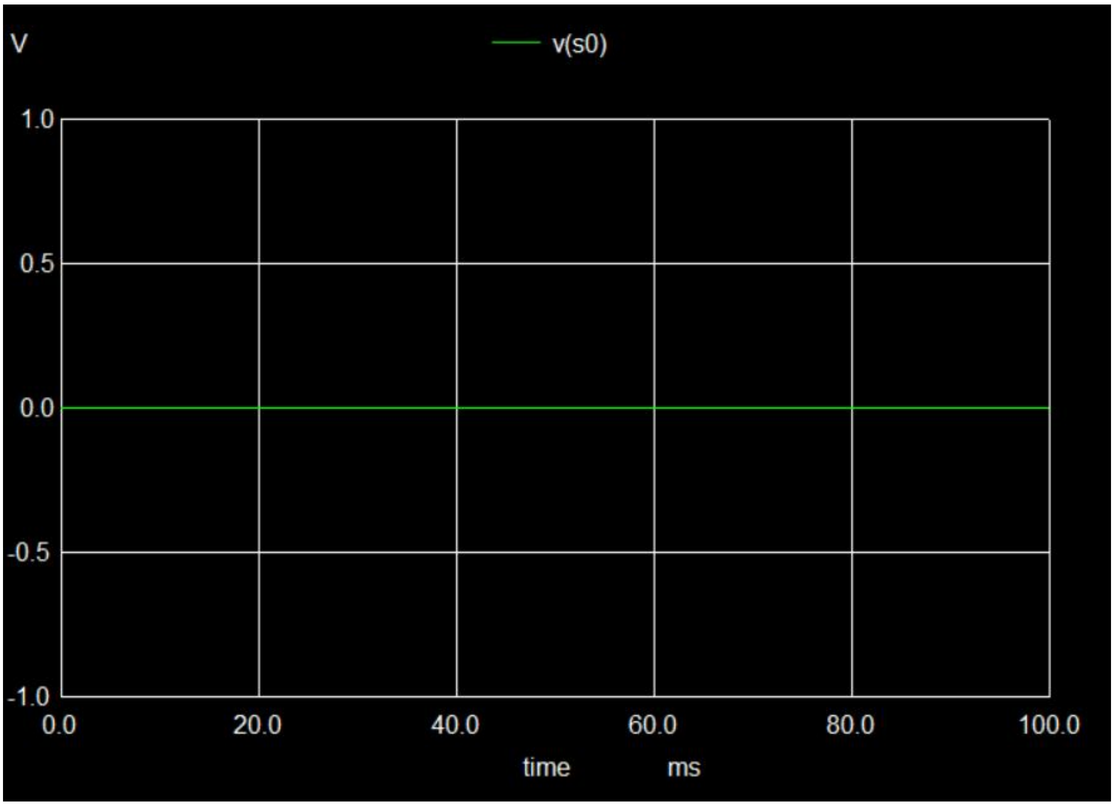

V
v(q2)
5.0
4.0
3.0
2.0
1.0
0.0
0.0
20.0
40.0
60.0
80.0
100.0
time
ms

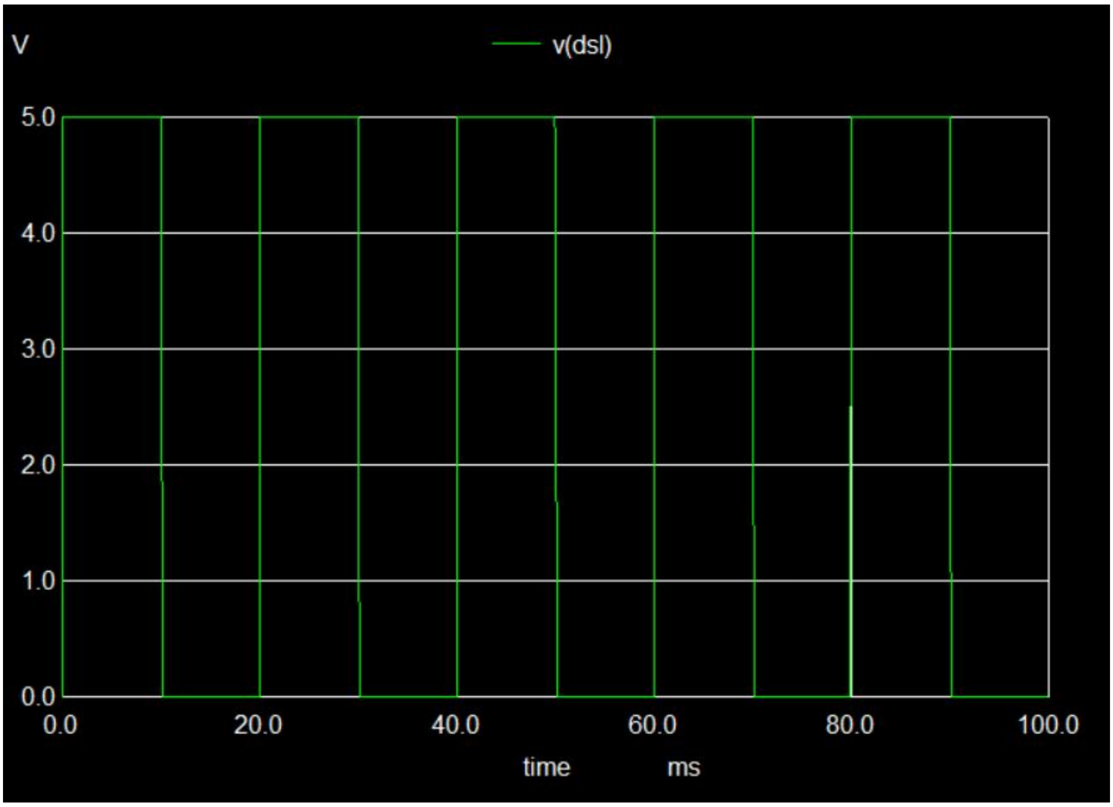

V
v(q1)
5.0
4.0
3.0
2.0
1.0
0.0
0.0
20.0
40.0
60.0
80.0
100.0
time
ms

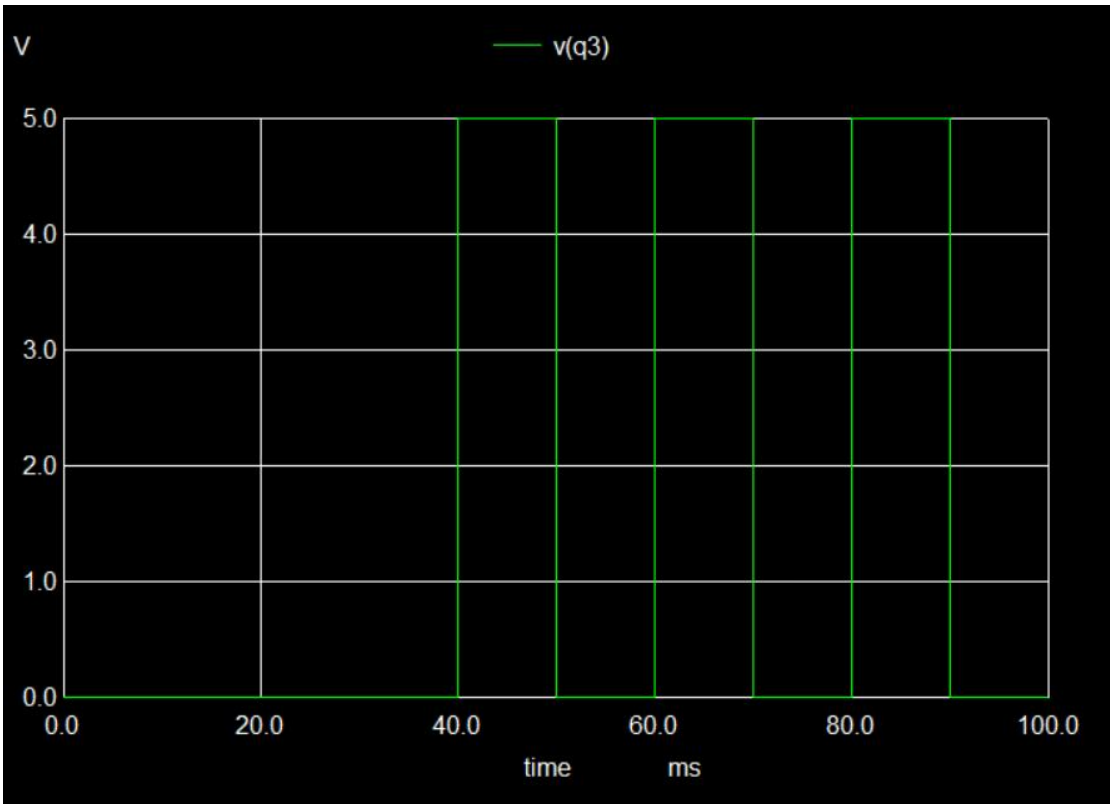

V
v(q0)
5.0
4.0
3.0
2.0
1.0
0.0
0.0
20.0
40.0
60.0
80.0
100.0
time
ms

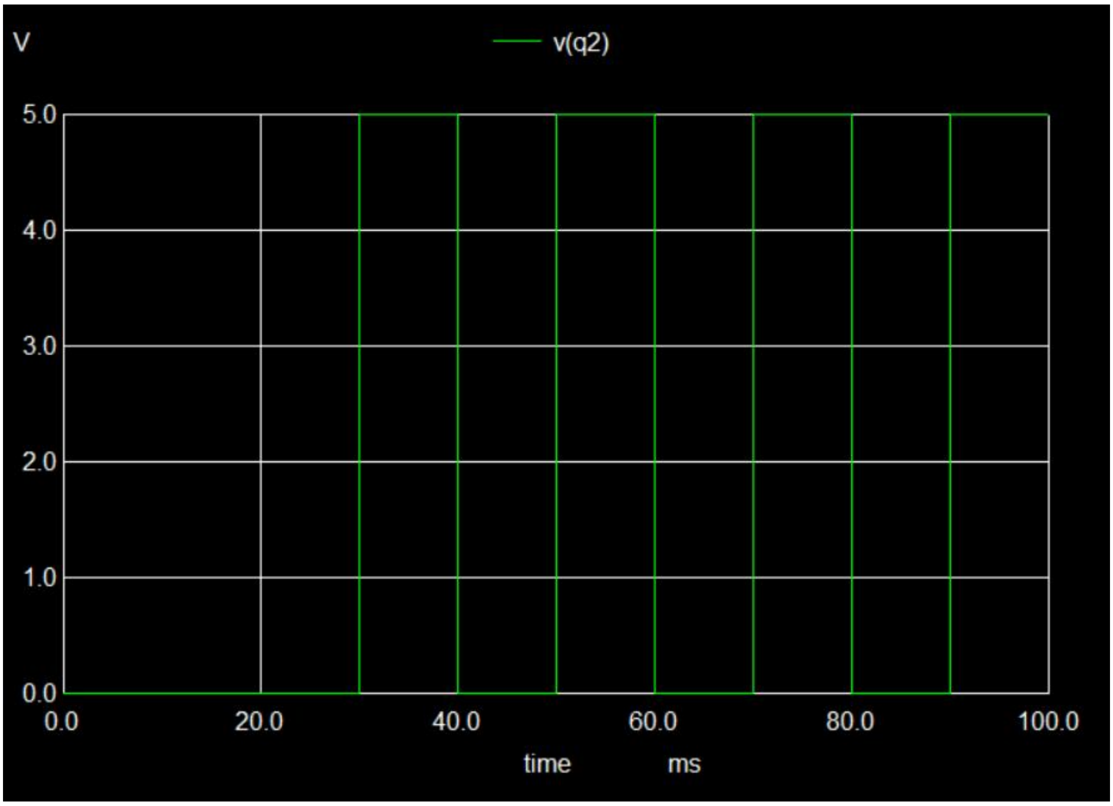
C) For input data - 1010
Select input - 10 (LEFT SHIFT)
Reset - 0

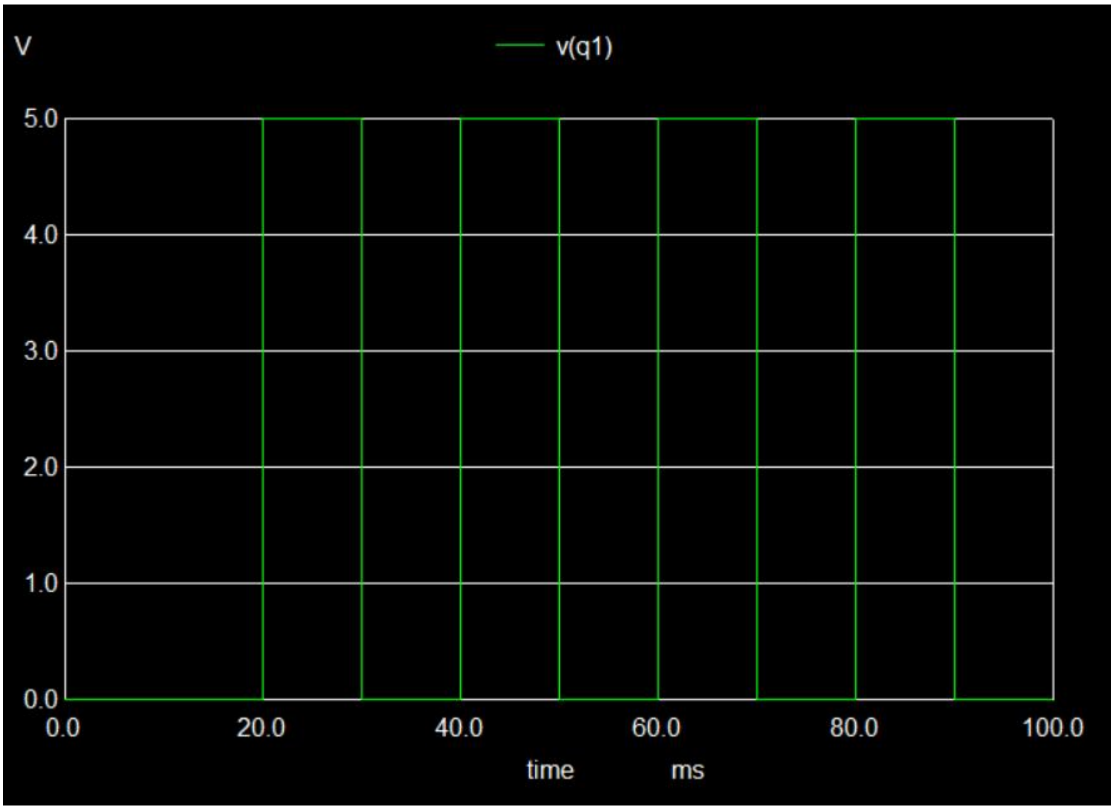


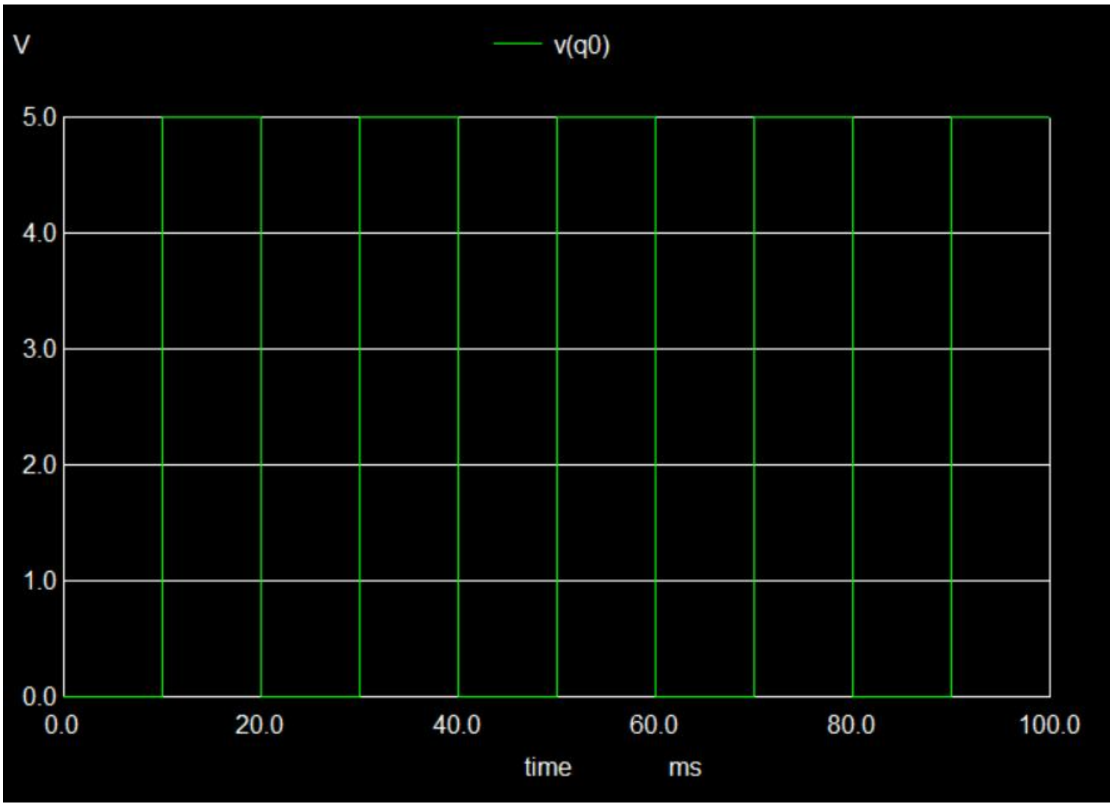



V
v(dsl)
5.0
4.0
3.0
2.0
1.0
0.0
0.0
20.0
40.0
60.0
80.0
100.0
time
ms

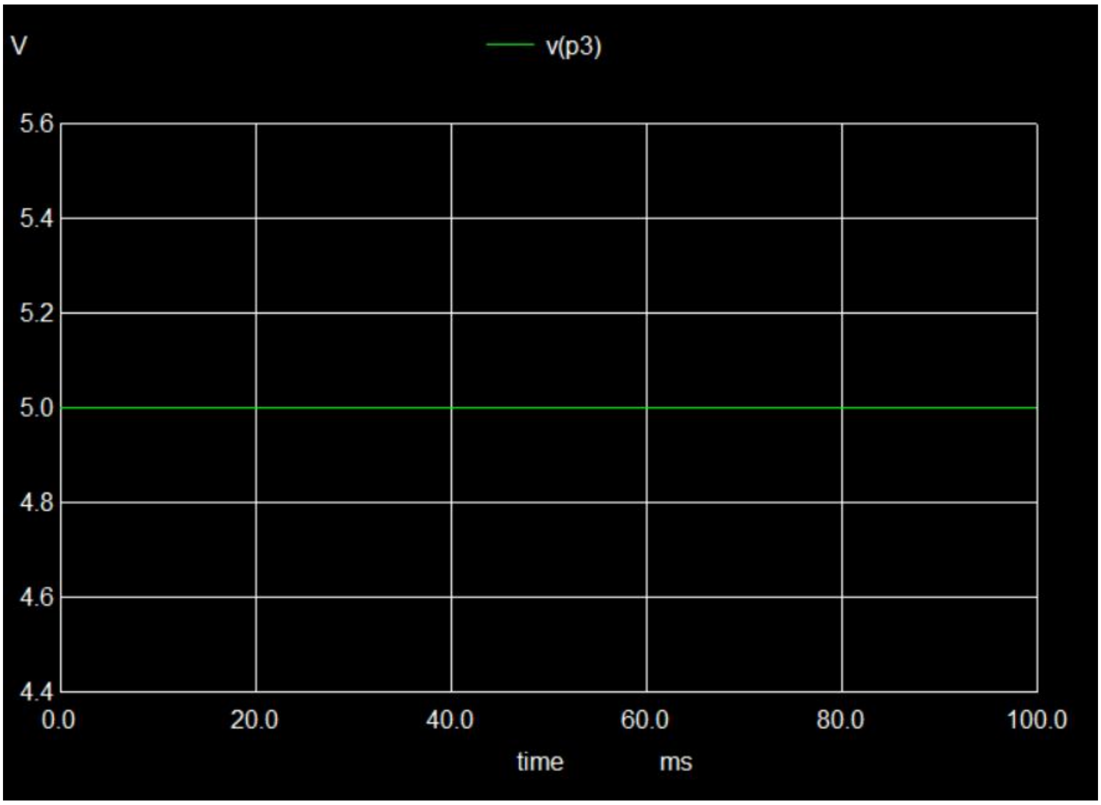

V
v(q3)
5.0
4.0
3.0
2.0
1.0
0.0
0.0
20.0
40.0
60.0
80.0
100.0
time
ms

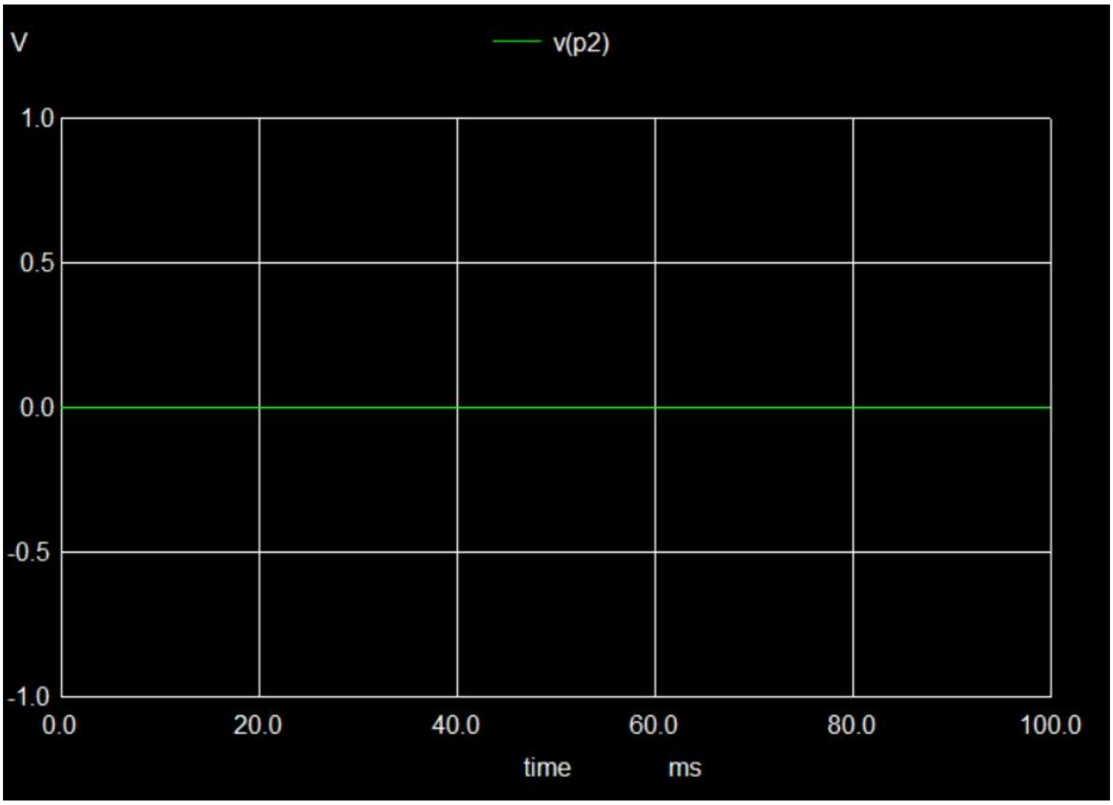

V
v(q2)
5.0
4.0
3.0
2.0
1.0
0.0
0.0
20.0
40.0
60.0
80.0
100.0
time
ms

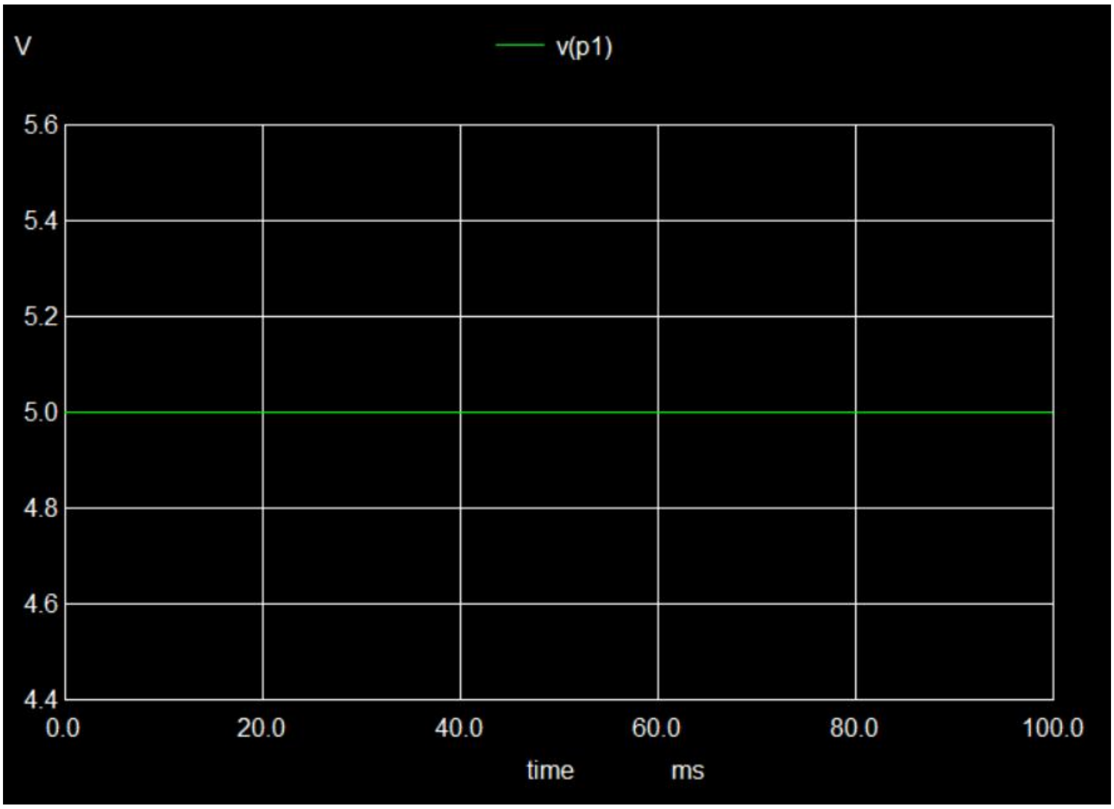

V
v(q1)
5.0
4.0
3.0
2.0
1.0
0.0
0.0
20.0
40.0
60.0
80.0
100.0
time
ms

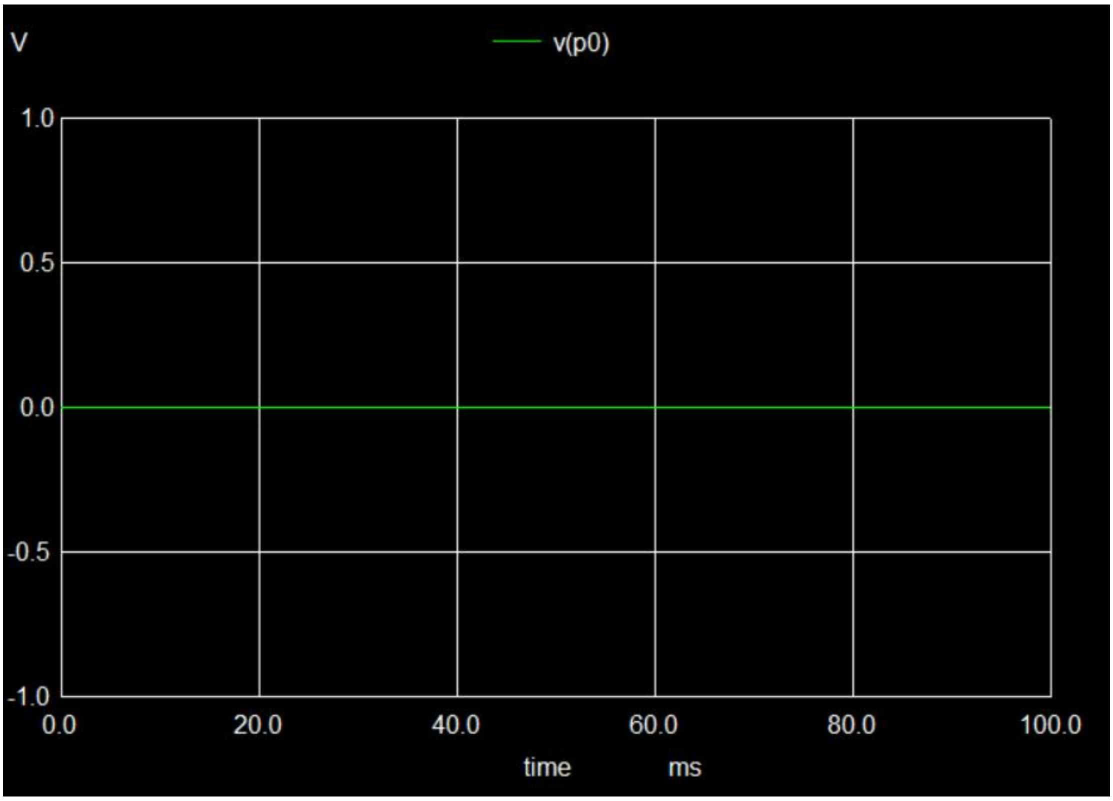


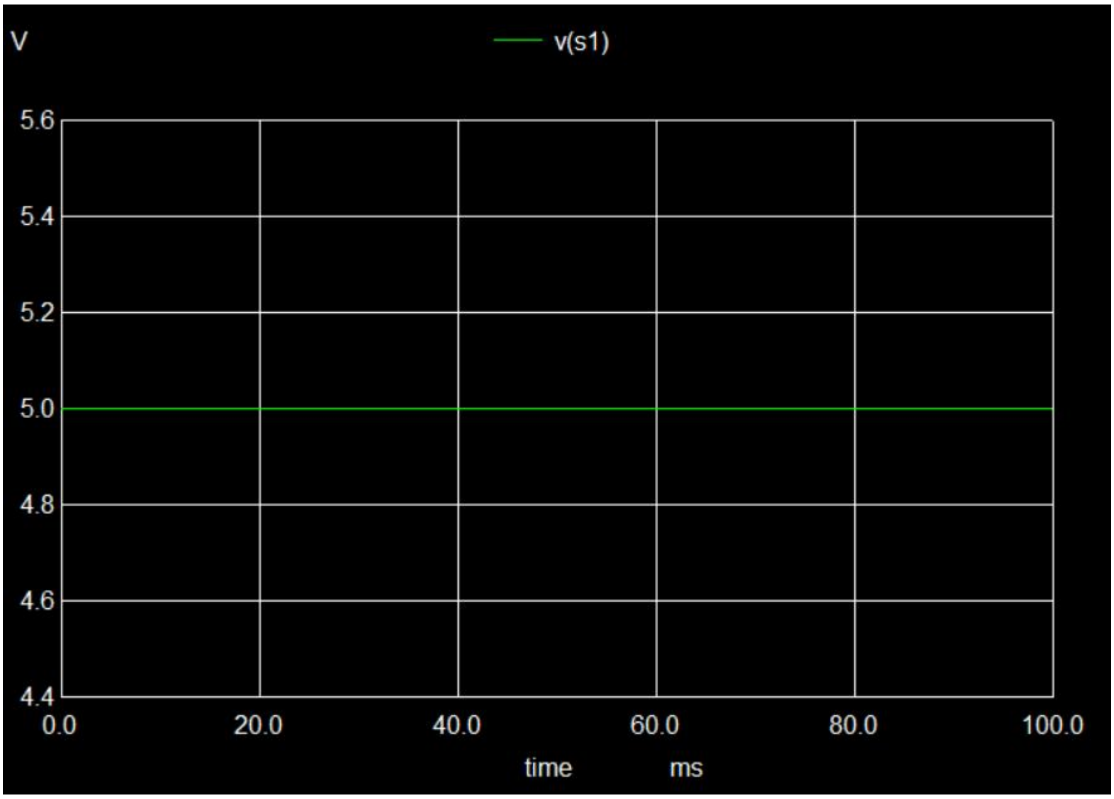
D) For input data - 1010
Select input - 11 (PARALLEL LOADING)
Reset - 0

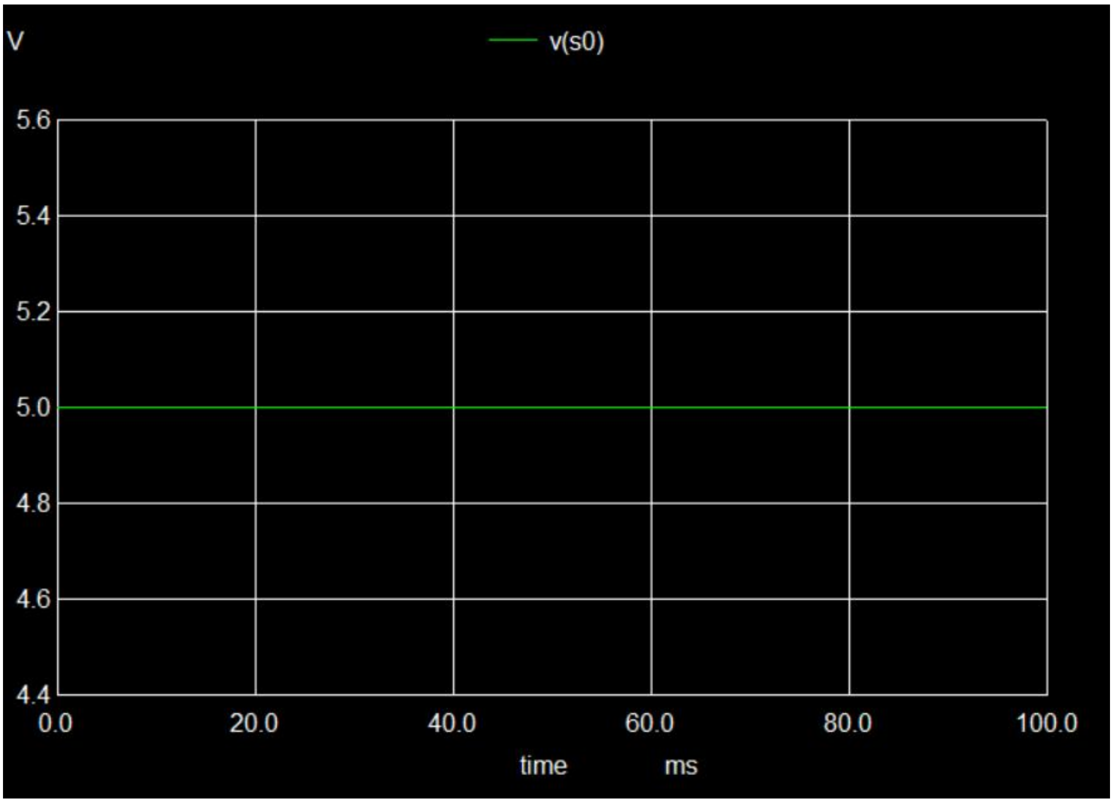


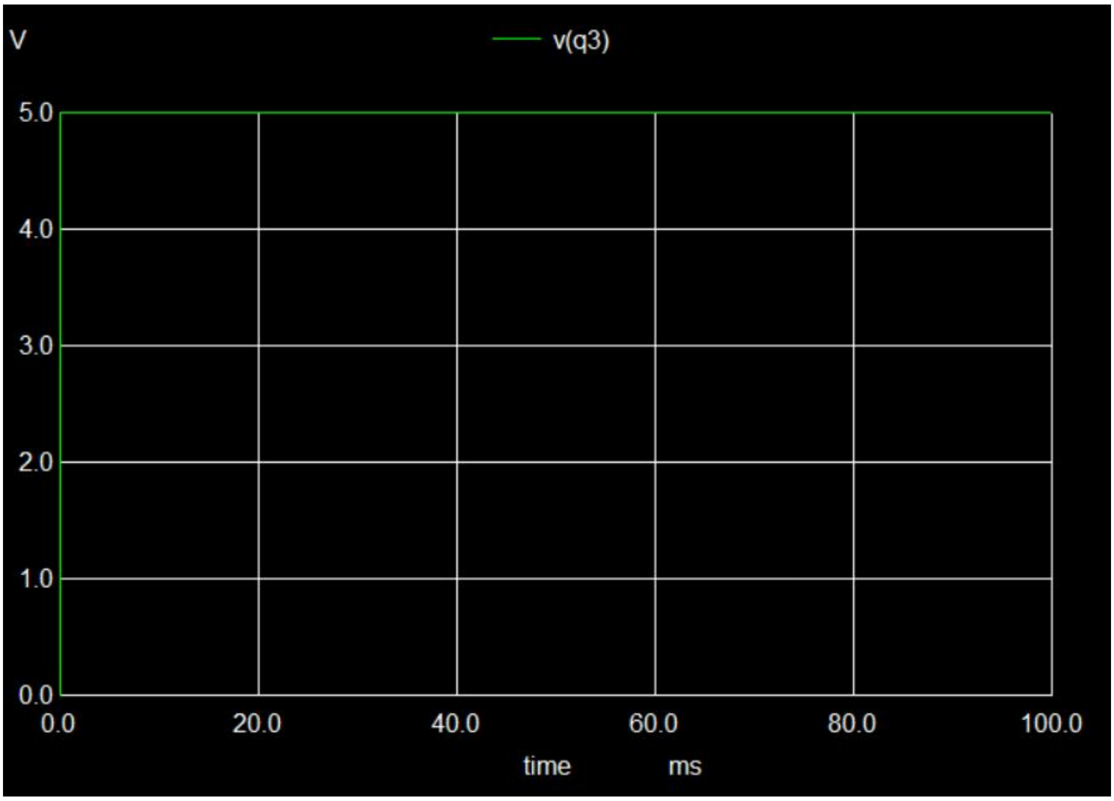

V
v(p2)
1.0
0.5
0.0
-0.5
-1.0
0.0
20.0
40.0
60.0
80.0
100.0
time
ms

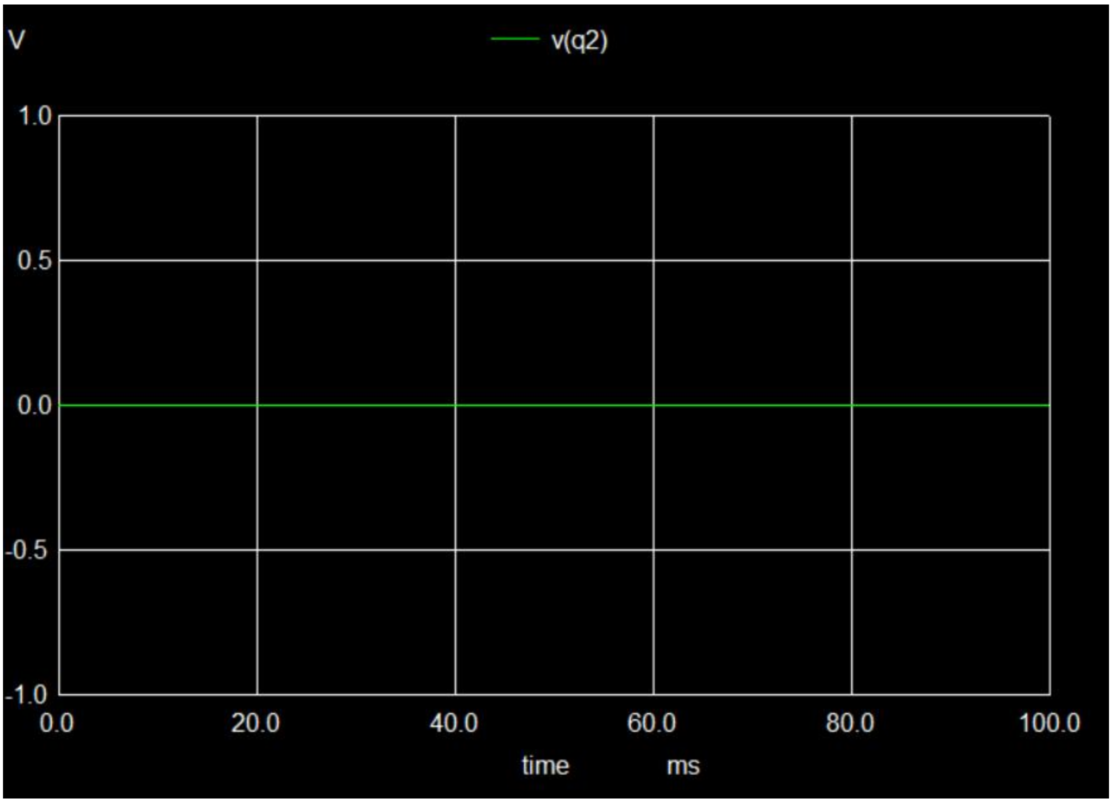

V
v(p1)
5.6
5.4
5.2
5.0
4.8
4.6
4.4
0.0
20.0
40.0
60.0
80.0
100.0
time
ms

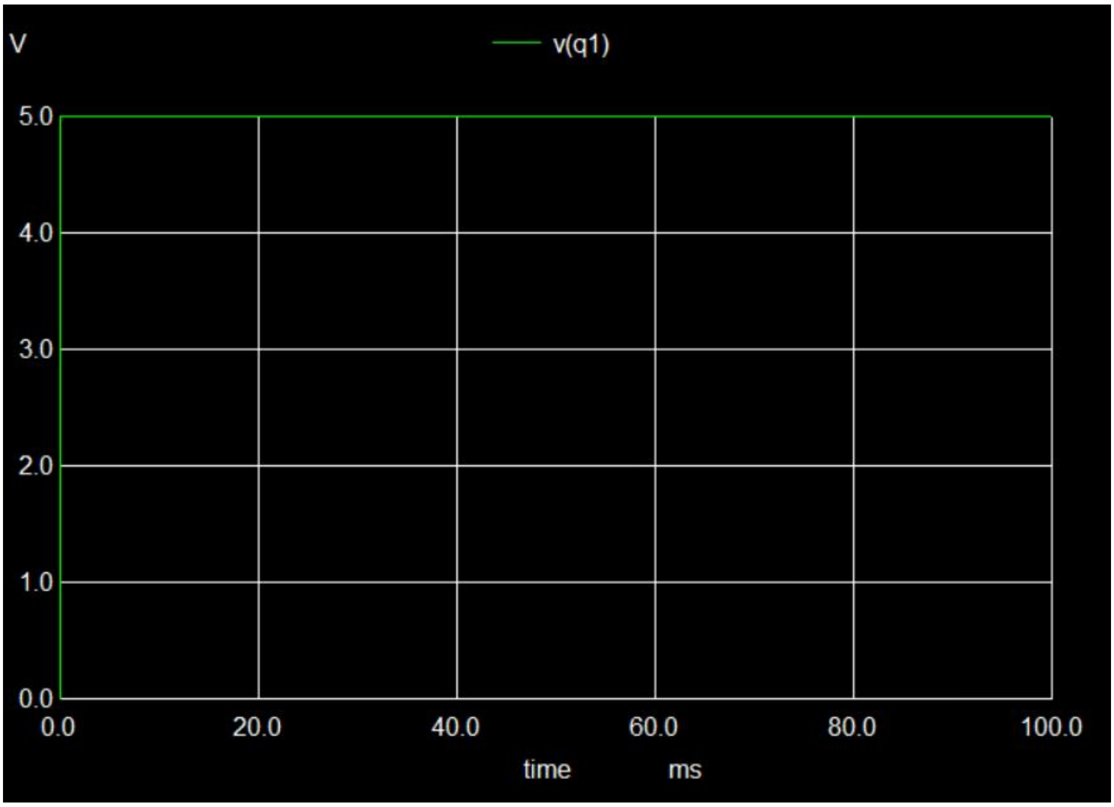

V
v(p0)
1.0
0.5
0.0
-0.5
-1.0
0.0
20.0
40.0
60.0
80.0
100.0
time
ms

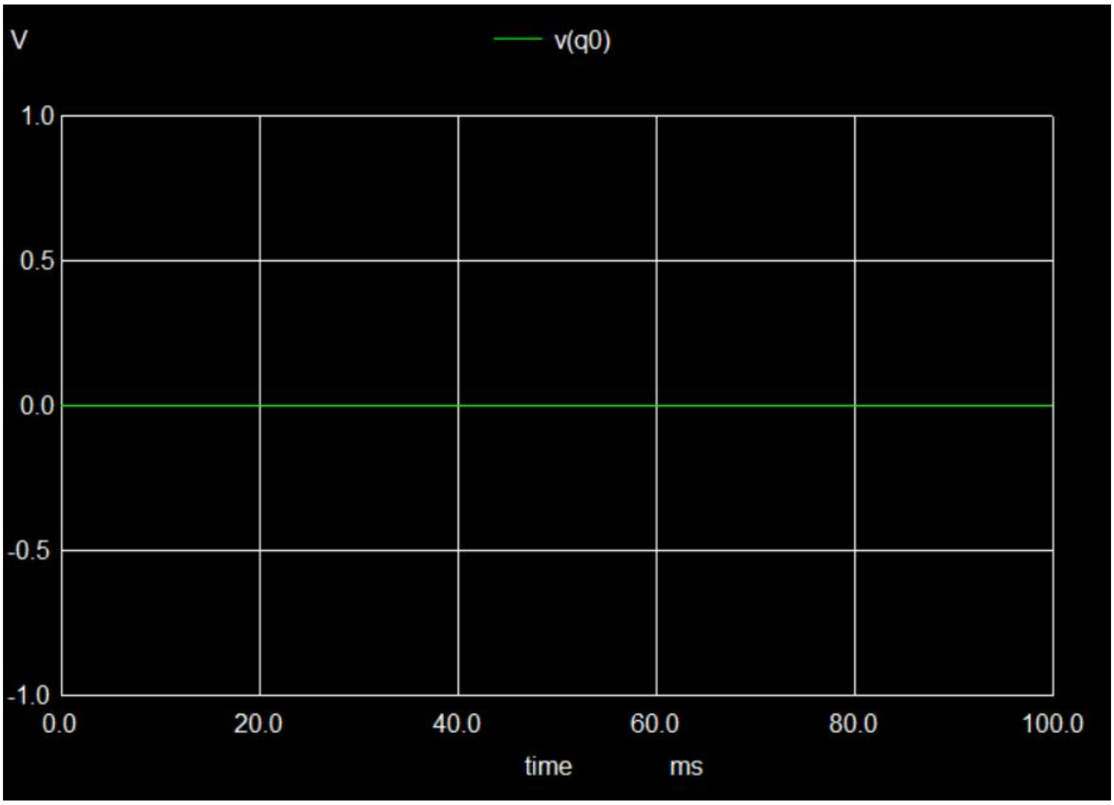

V
v(s1)
5.6
5.4
5.2
5.0
4.8
4.6
4.4
0.0
20.0
40.0
60.0
80.0
100.0
time
ms


V
v(s0)
5.6
5.4
5.2
5.0
4.8
4.6
4.4
0.0
20.0
40.0
60.0
80.0
100.0
time
ms


V
v(q3)
5.0
4.0
3.0
2.0
1.0
0.0
0.0
20.0
40.0
60.0
80.0
100.0
time
ms

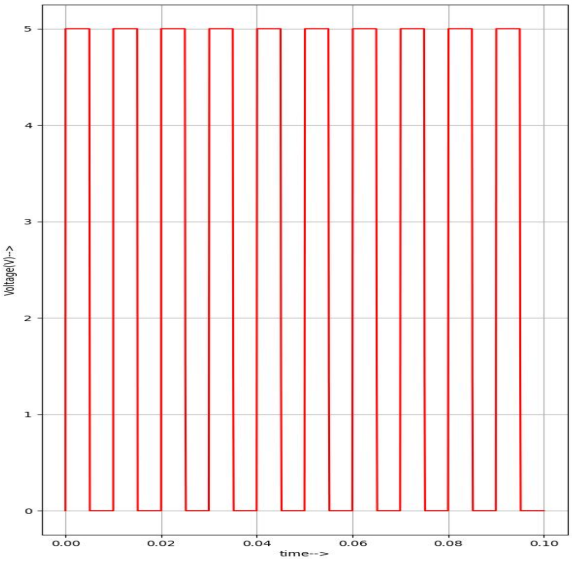

V
v(q2)
1.0
0.5
0.0
-0.5
-1.0
0.0
20.0
40.0
60.0
80.0
100.0
time
ms


V
v(q1)
5.0
4.0
3.0
2.0
1.0
0.0
0.0
20.0
40.0
60.0
80.0
100.0
time
ms

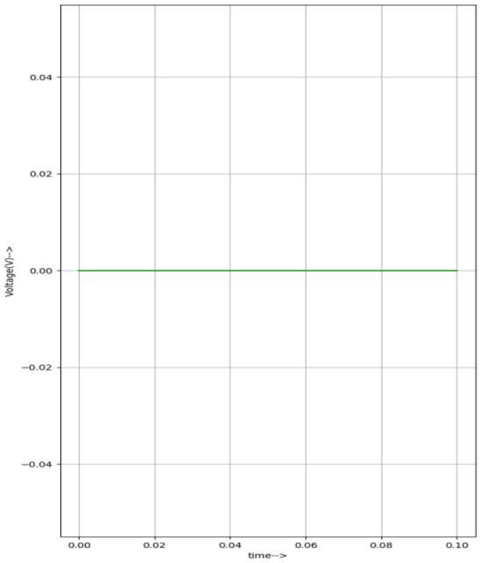


2) Python Plots

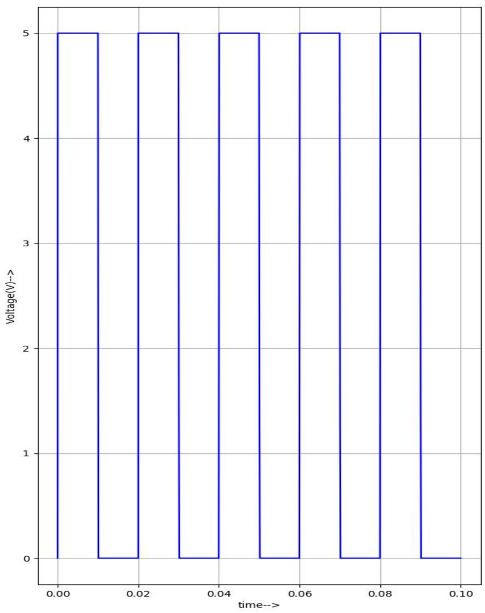
CLOCK



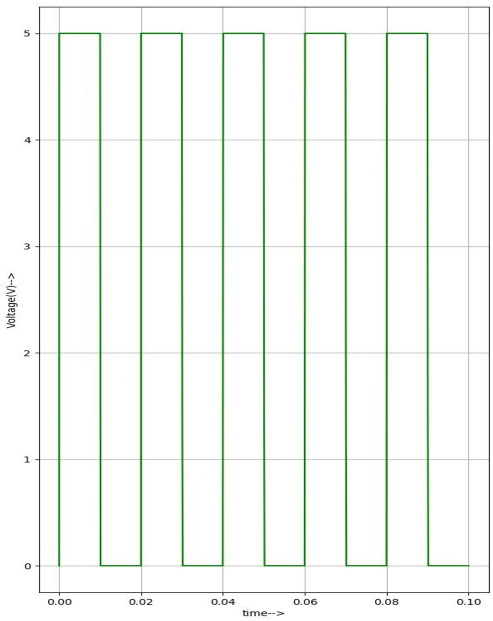
RESET



DSL

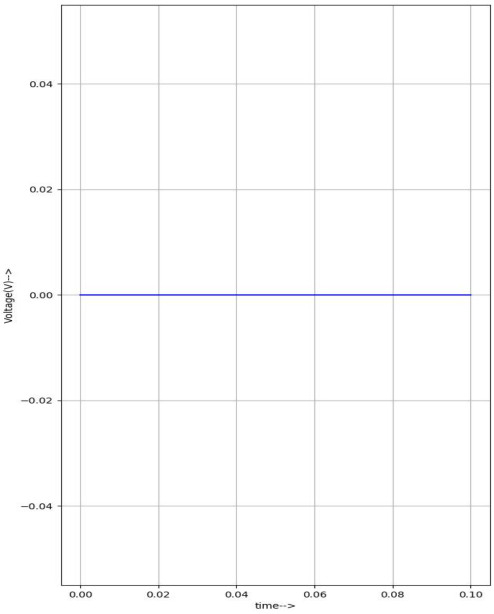


DSR

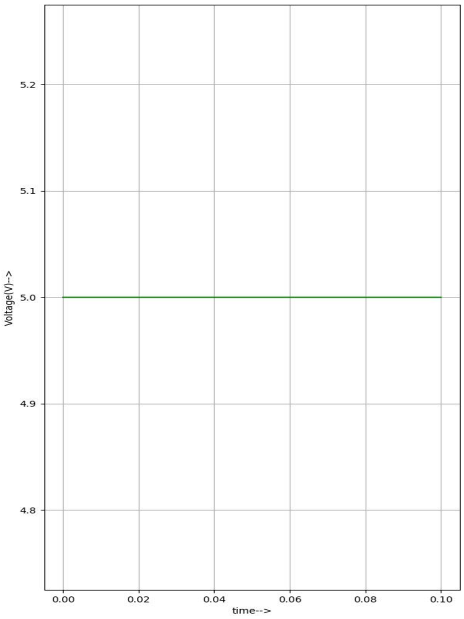


A) For input data - 1010
Select input - 00 (NO CHANGE)
Reset - 0

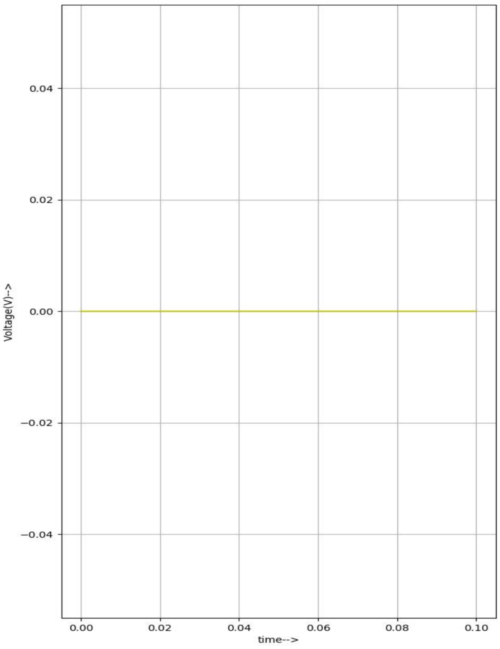
INPUT P0



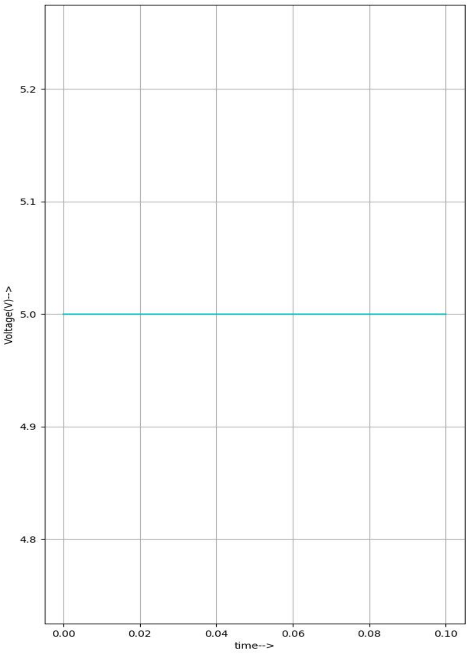
INPUT P1



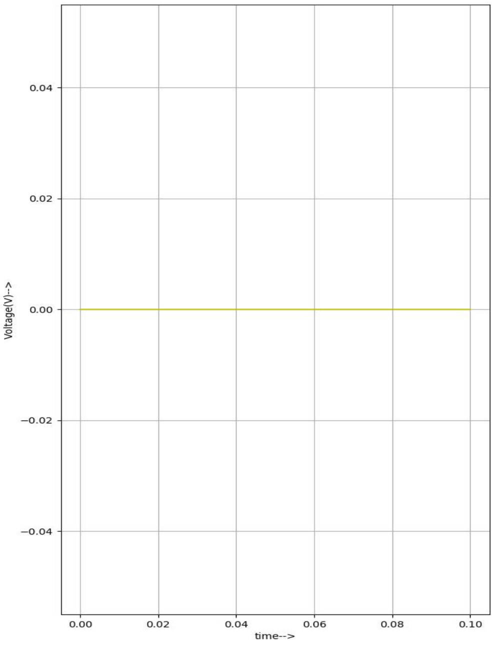
INPUT P2



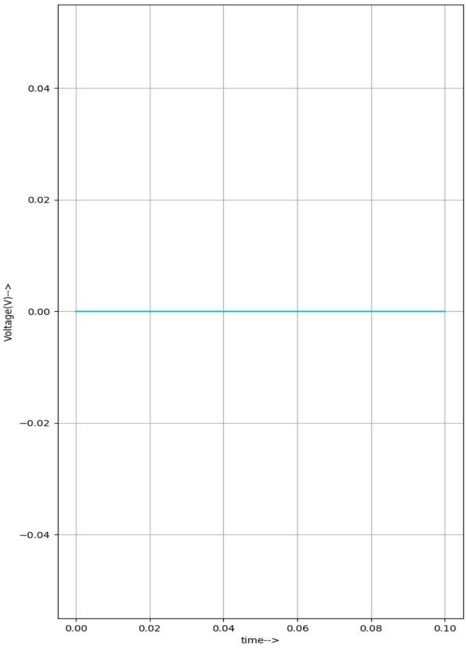
INPUT P3



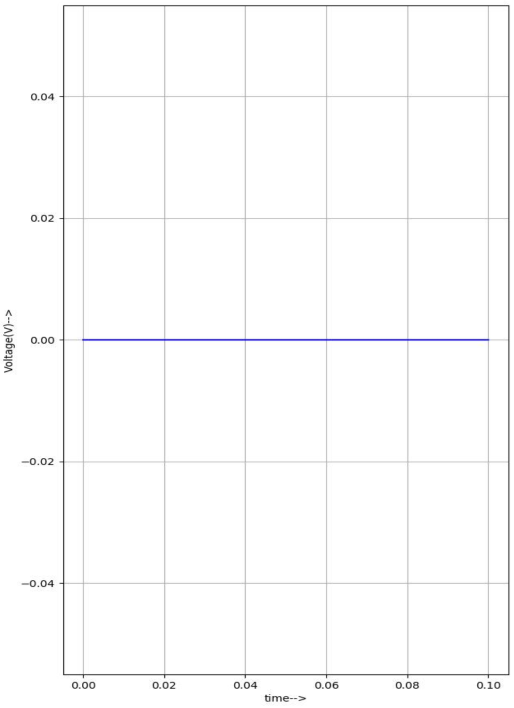
S1



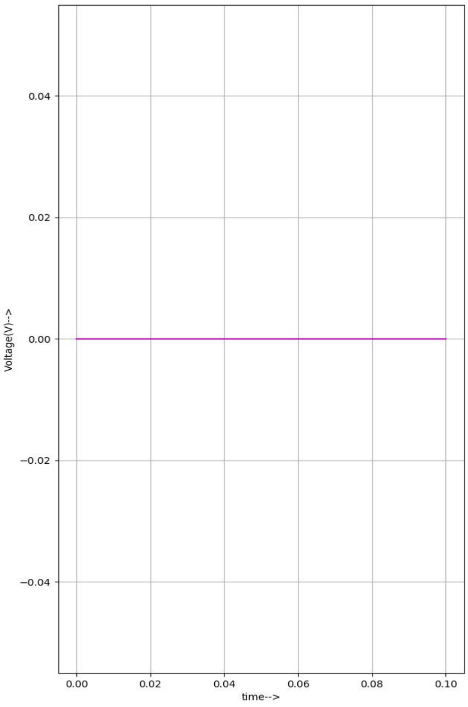
S0



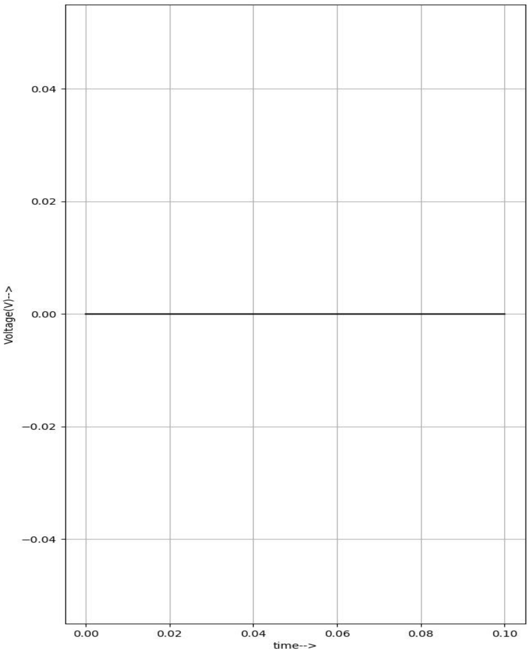
OUTPUT q3



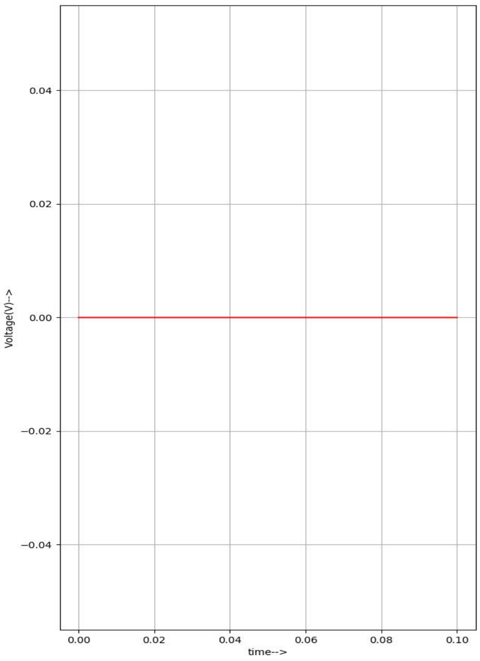
OUTPUT q2



OUTPUT q1

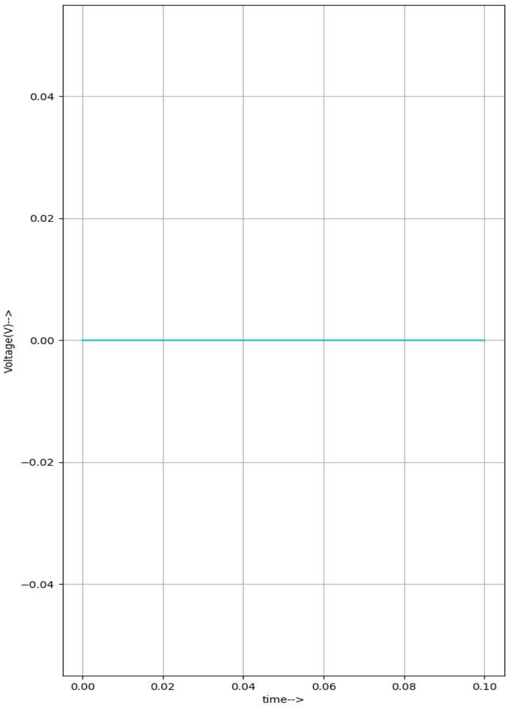


OUTPUT q0

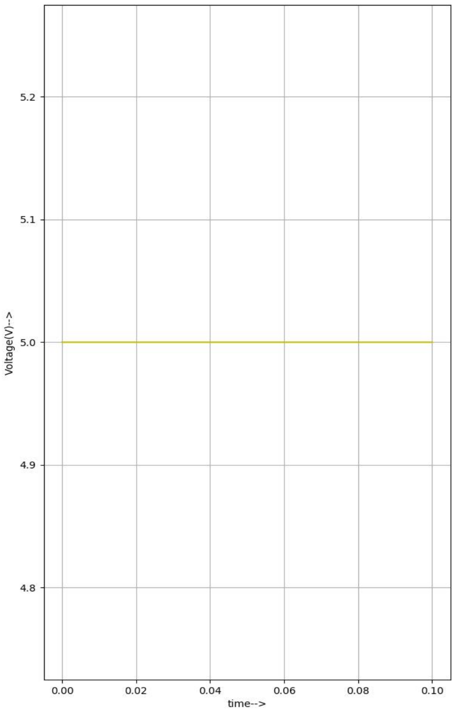


B) For input data - 1010
Select input - 01 (RIGHT SHIFT)
Reset - 0

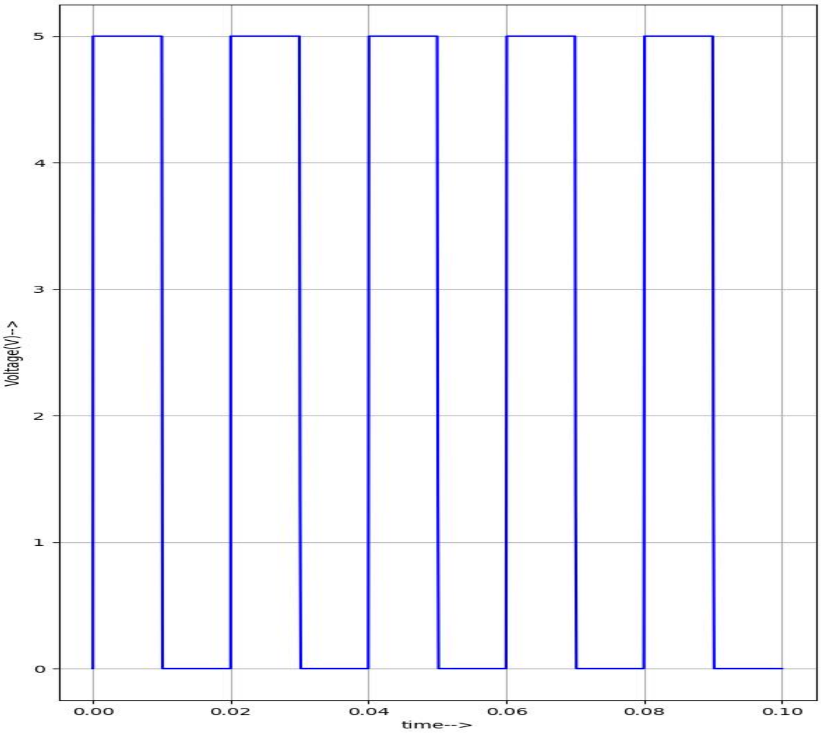
S1



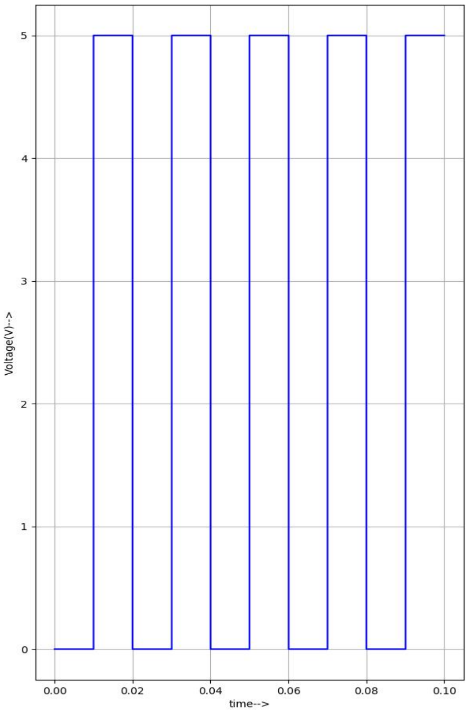
S0



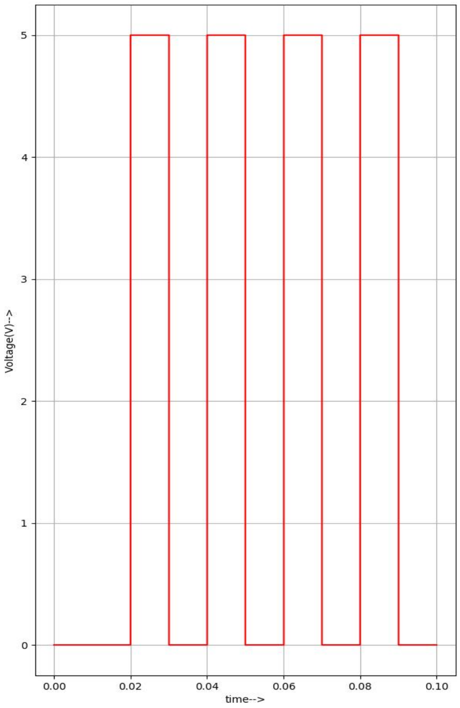
DSR



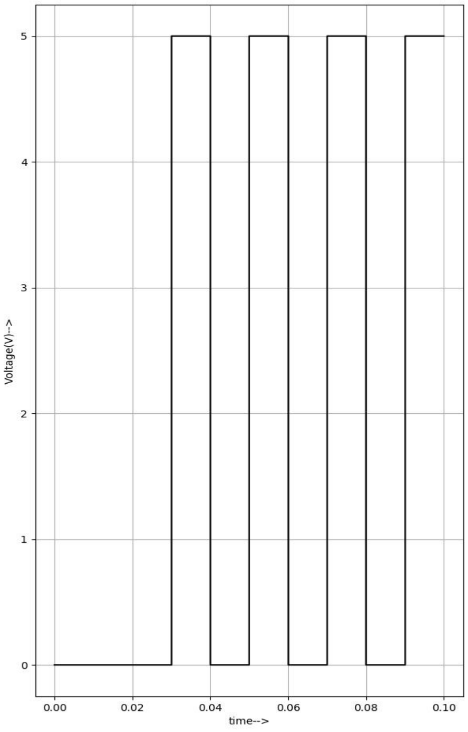
OUTPUT q3



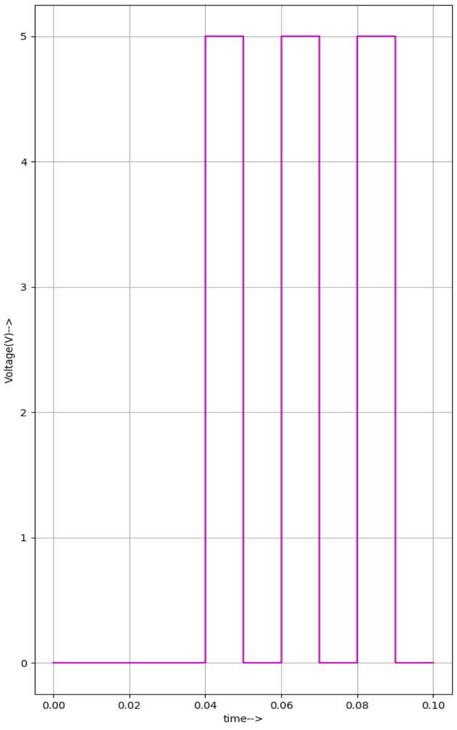
OUTPUT q2



OUTPUT q1

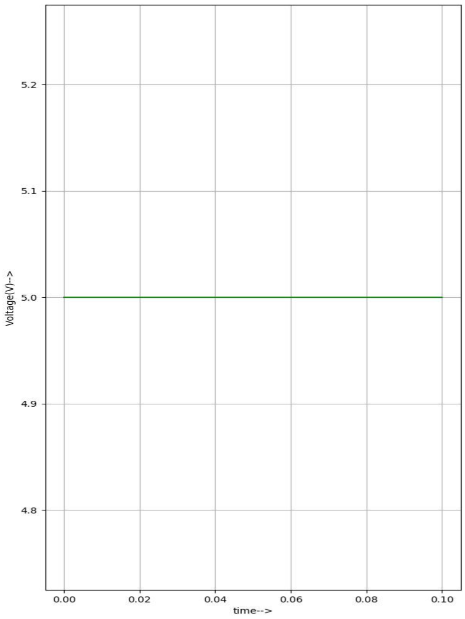


OUTPUT q0

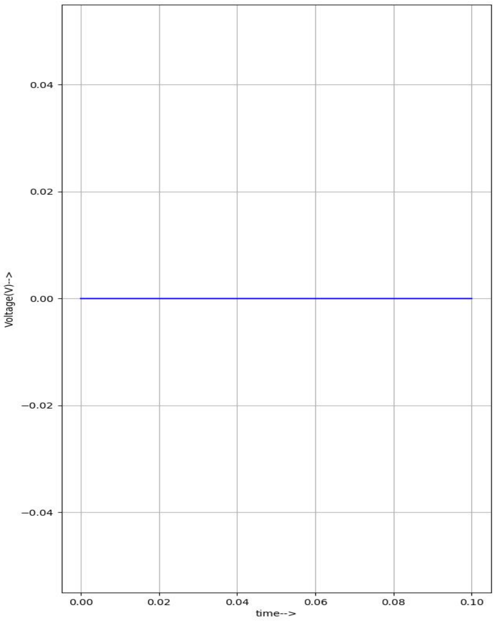


C) For input data - 1010
Select input - 10 (LEFT SHIFT)
Reset - 0

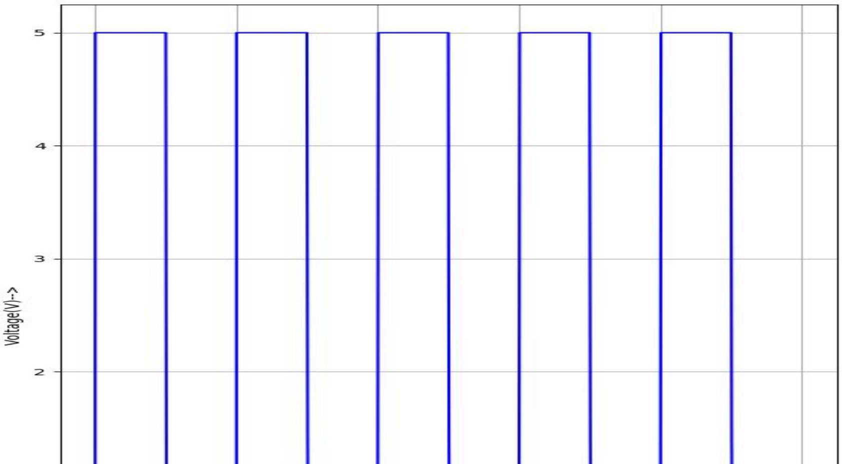
S1



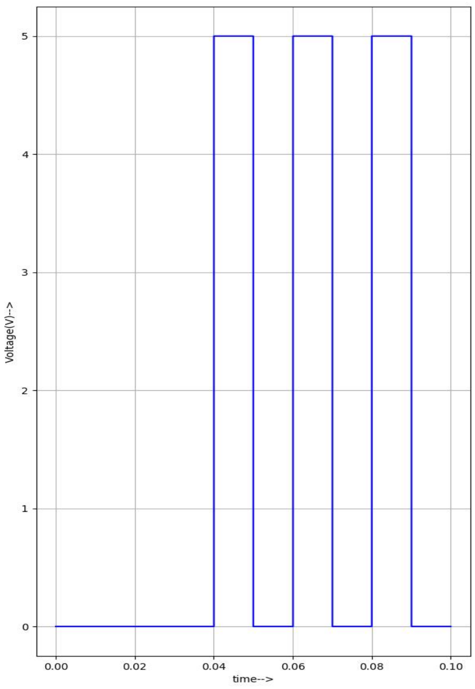
S0



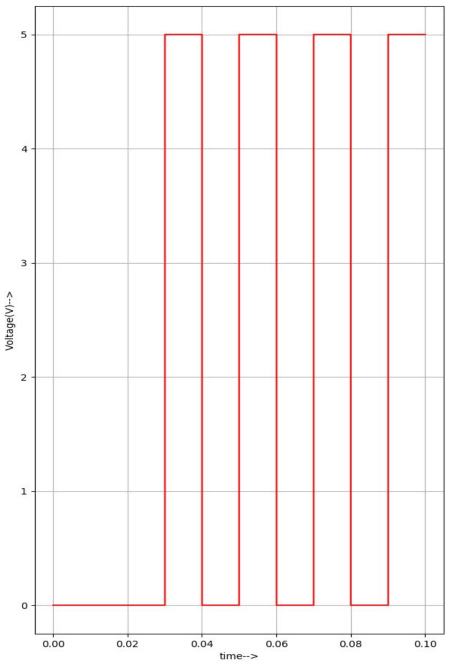
DSL



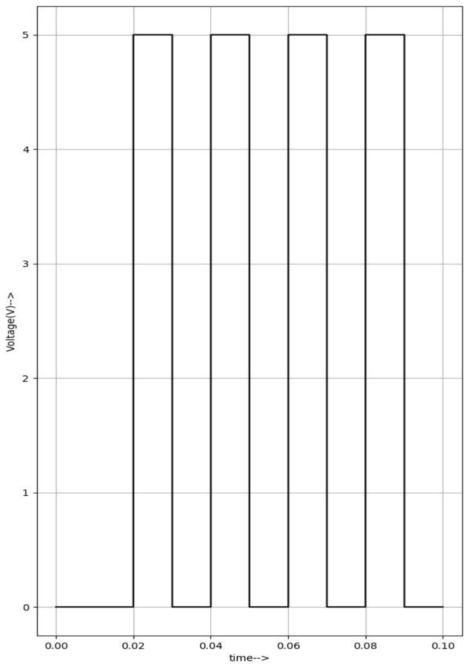
OUTPUT q3



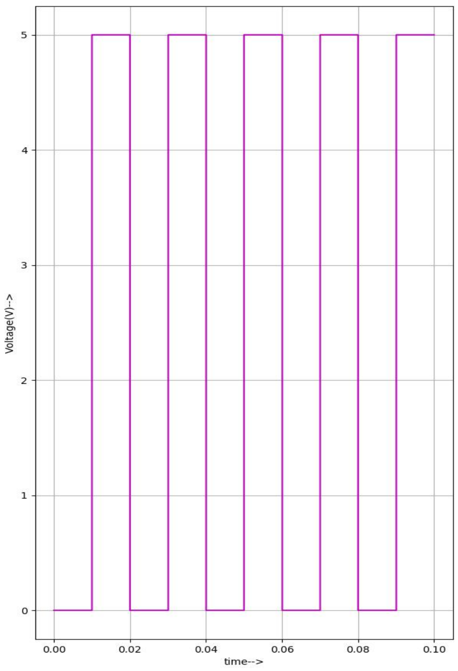
OUTPUT q2



OUTPUT q1

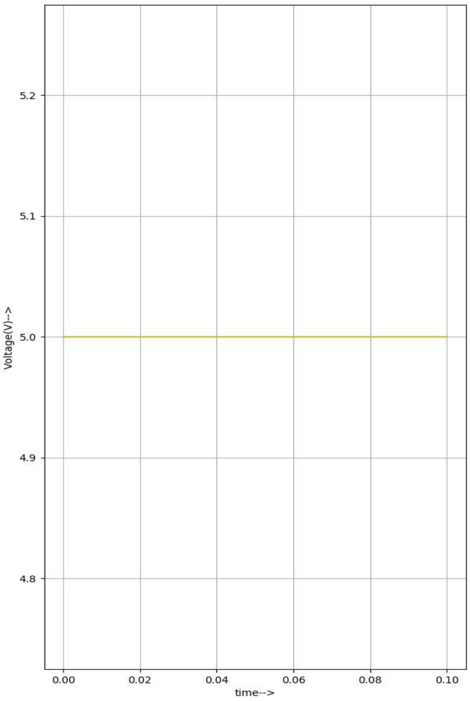


OUTPUT q0

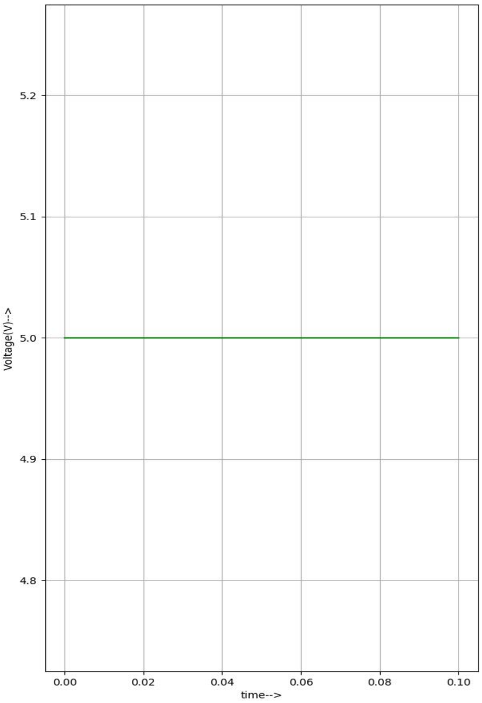


D) For input data - 1010
Select input - 11 (PARALLEL LOADING)
Reset - 0

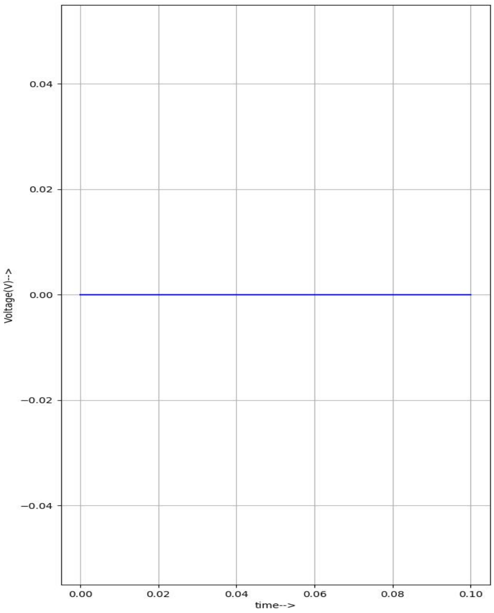
S1



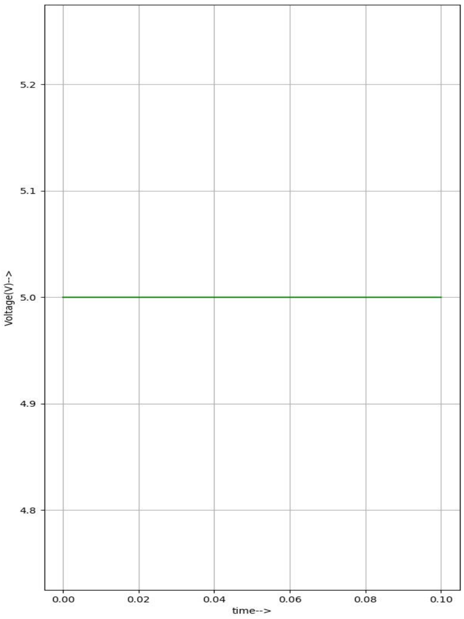
S0



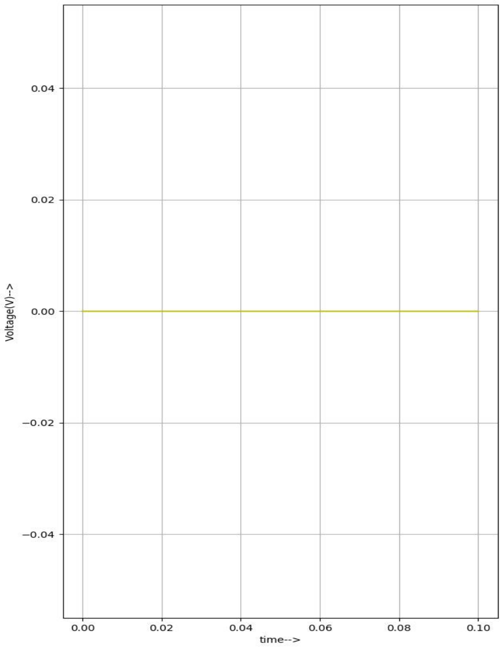
INPUT P0



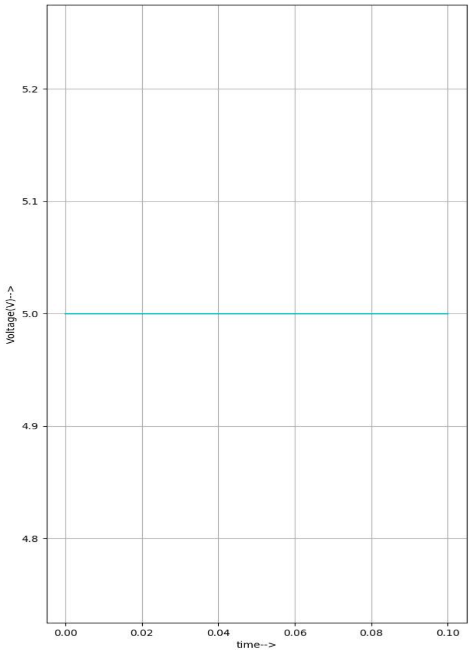
INPUT P1



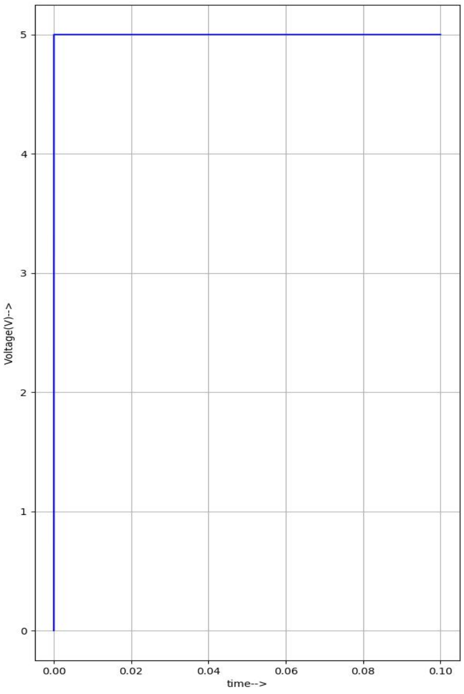
INPUT P2



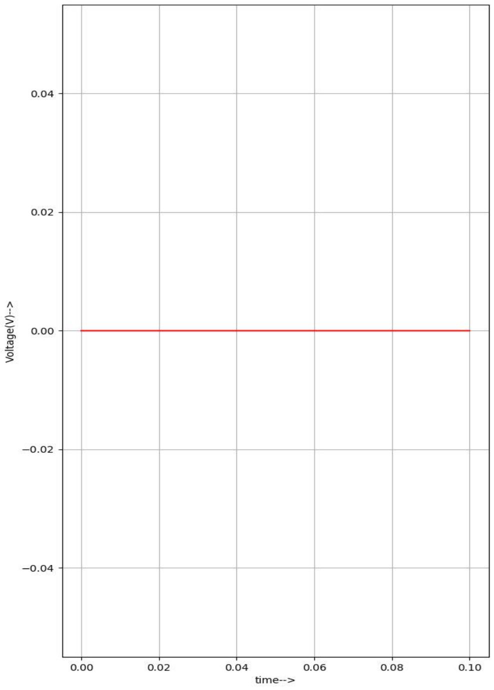
INPUT P3



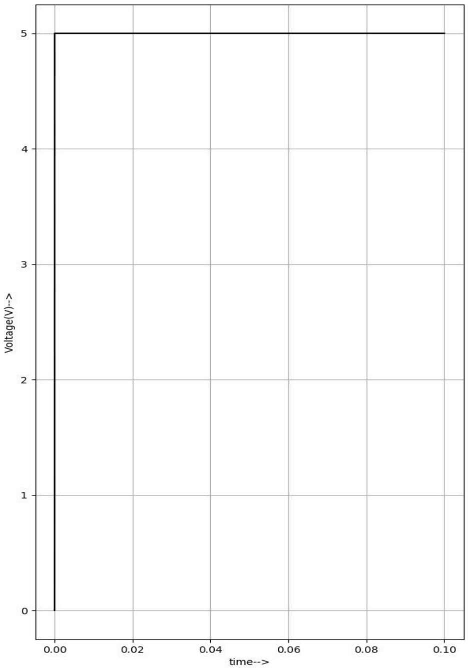
OUTPUT Q3



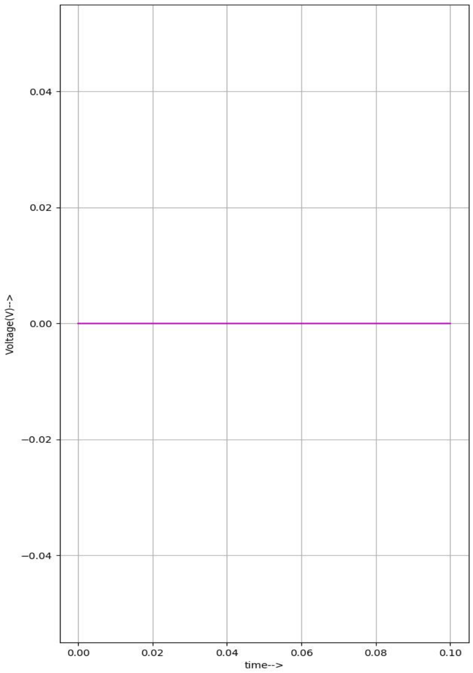
OUTPUT Q2



OUTPUT Q1



OUTPUT Q0



REFRENCES:

https://www.elprocus.com/what-is-universal-shift-register-its-working/

https://learn.adafruit.com/digital-circuits-4-sequential-circuits/shift-registers