LabVIEW report

COMMUNICATION DEPARTMENT

NAME	ID
Mahmoud Khaled Mohamed Ahmed	20011788
Asmaa Hassan Mokhtar Aboushady	19015430
Hassan Ibrahim Hassan Ibrahim	20010484
Mazen Mohamed Mahmoud Ali	20011162
Abdelrahman Mahmoud Mohamed Eldesokey	20010860

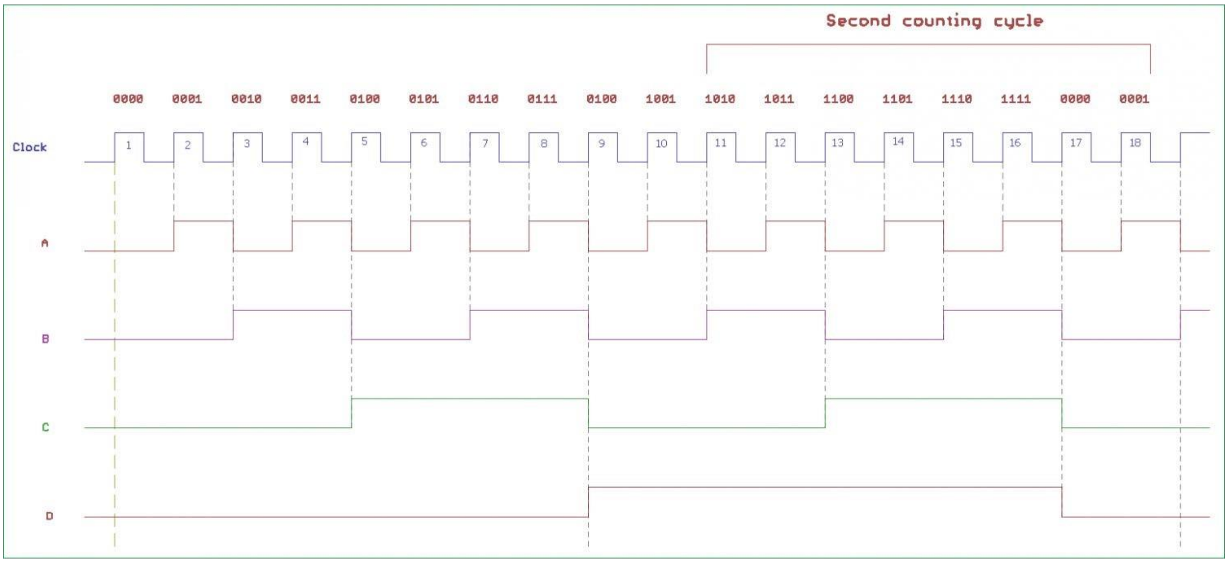
4-Bit Synchronous Counter

How does a counter work?

A device that can count any given event based on the number of times it occurs is known as a counter. Depending on a clock signal, this counter can count and store the number of times a particular event or process has occurred in a digital logic system or computer. The sequential digital logic circuit, which has multiple outputs and a single clock input, is the most prevalent type of counter. Binary or binary-coded decimal numbers are the outputs. The number can either rise or fall with each clock pulse.

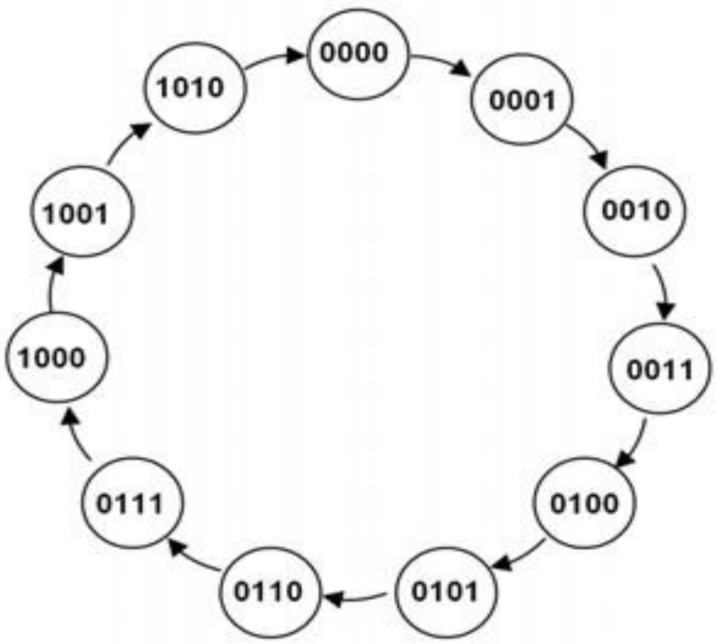
Synchronous generally refers to something that is time-synchronized with others. All clocks follow the same reference clock when synced signals occur at the same rate.

Synchronous Counter Timing Diagram:

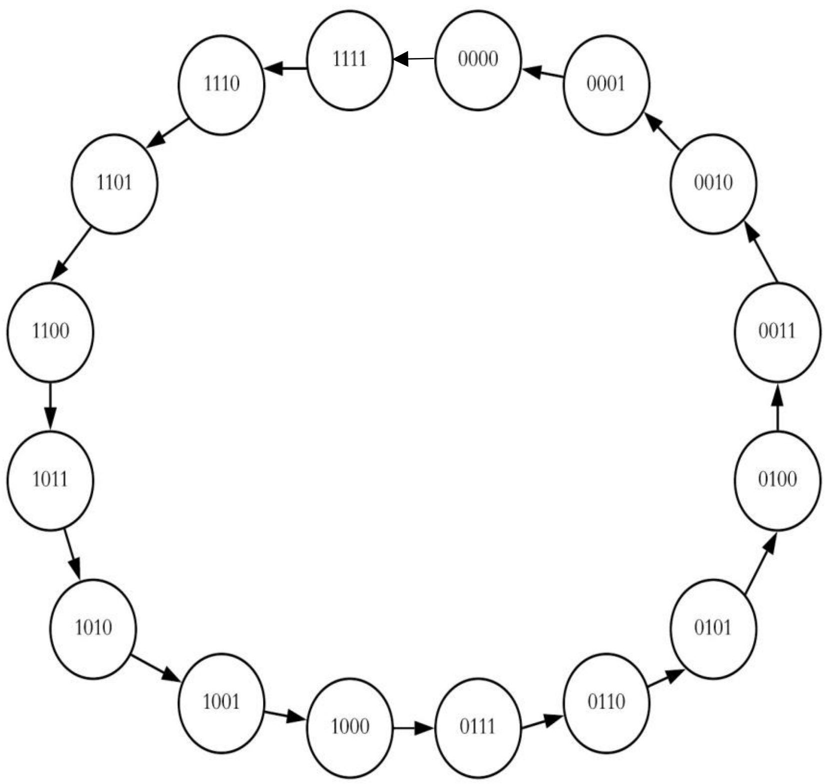


The state diagram of the 4-bit counter:

UP COUNTER:



DOWN COUNTER:



The next state table using D flip-flop :

UP COUNTER:

Q3	Q2	Q1	Q0	Q3`	Q2`	Q1`	Q0`	D3	D2	D1	D0	N
0	0	0	0	0	0	0	1	0	0	0	1	0
0	0	0	1	0	0	1	0	0	0	1	0	1
0	0	1	0	0	0	1	1	0	0	1	1	2
0	0	1	1	0	1	0	0	0	1	0	0	3
0	1	0	0	0	1	0	1	0	1	0	1	4
0	1	0	1	0	1	1	0	0	1	1	0	5
0	1	1	0	0	1	1	1	0	1	1	1	6
0	1	1	1	1	0	0	0	1	0	0	0	7
1	0	0	0	1	0	0	1	1	0	0	1	8
1	0	0	1	1	0	1	0	1	0	1	0	9
1	0	1	0	1	0	1	1	1	0	1	1	10
1	0	1	1	1	1	0	0	1	1	0	0	11
1	1	0	0	1	1	0	1	1	1	0	1	12
1	1	0	1	1	1	1	0	1	1	1	0	13
1	1	1	0	1	1	1	1	1	1	1	1	14
1	1	1	1	0	0	0	0	0	0	0	0	15

DOWN COUNTER:

Q3	Q2	Q1	Q0	Q3`	Q2`	Q1`	Q0`	D3	D2	D1	D0	N
1	1	1	1	0	0	0	0	0	0	0	0	15
1	1	1	0	1	1	1	1	1	1	1	1	14
1	1	0	1	1	1	1	0	1	1	1	0	13
1	1	0	0	1	1	0	1	1	1	0	1	12
1	0	1	1	1	1	0	0	1	1	0	0	11
1	0	1	0	1	0	1	1	1	0	1	1	10
1	0	0	1	1	0	1	0	1	0	1	0	9
1	0	0	0	1	0	0	1	1	0	0	1	8
0	1	1	1	1	0	0	0	1	0	0	0	7
0	1	1	0	0	1	1	1	0	1	1	1	6
0	1	0	1	0	1	1	0	0	1	1	0	5
0	1	0	0	0	1	0	1	0	1	0	1	4
0	0	1	1	0	1	0	0	0	1	0	0	3
0	0	1	0	0	0	1	1	0	0	1	1	2
0	0	0	1	0	0	1	0	0	0	1	0	1
0	0	0	0	0	0	0	1	0	0	0	1	0

From the next-state table the equation of the flip-flops are :

UP COUNTER:

Q1Q0 \ Q3Q2	00	01	11	10
00	0	0	1	1
01	0	0	1	1
11	0	1	0	1
10	0	0	1	1

$D3 = Q3Q2' + Q3Q0' + Q3Q1' + Q3'Q2Q1Q0$

$= Q3 \oplus (Q2\ Q1\ Q0)$

Q1 Q0 \ Q3 Q2	00	01	11	10
00	0	1	1	0
01	0	1	1	0
11	1	0	0	1
10	0	1	1	0

$D2 = Q2Q1' + Q2Q0' + Q2'Q1Q0$

$= Q2 \oplus (Q1\ Q0)$

Q1 Q0 \ Q3 Q2	00	01	11	10
00	0	0	0	0
01	1	1	1	1
11	0	0	0	0
10	1	1	1	1

$D1 = Q1'Q0 + Q1Q0' = Q1 \oplus Q2$

Q1 Q0 \ Q3 Q2	00	01	11	10
00	1	1	1	1
01	0	0	0	0
11	0	0	0	0
10	1	1	1	1

$D0 = Q0'$

DOWN COUNTER:

Q1 Q0 \ Q3 Q2	00	01	11	10
00	1	1	0	0
01	1	1	0	0
11	1	0	1	0
10	1	1	0	0

$D3 = Q3'Q2' + Q3'Q1Q0' + Q3'Q1' + Q3Q2Q1Q0$

$= Q2 \oplus (Q1\,Q0) = Q3 \oplus (Q2\,Q1\,Q0)$

Q1 Q0 \ Q3 Q2	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	0	1	1	0
10	1	0	0	1

$D2 = Q2'Q1' + Q2'Q0' + Q2Q1Q0$

Q_1, Q_0 \ Q_3Q_2	00	01	11	10
00	0	0	0	0
01	1	1	1	1
11	1	1	1	1
10	0	0	0	0

$D1 = Q1'Q0' + Q1Q0$

$= Q1 \oplus Q0$

Q1 Q0 \ Q3 Q2	00	01	11	10
00	1	1	1	1
01	0	0	0	0
11	1	1	1	1
10	0	0	0	0

$D0 = Q0$

Problem Analysis:

1. Initializing the appropriate virtual instruments in the LabVIEW.
2. Determining the basic way to initialize the components like the clock and the flip-flop and the multiplexer.
3. Make the output of the counter in many forms like the 7-segment display and the decimal numeric indicator.
4. Make the user have the freedom to choose the speed of the counter. And whether he wants the counter to act as an up counter or a down counter and the ability to reset the counter at any time.
5. Converting the S-R flip flop to D flip-flop by the analysis of their truth tables.
6. Make use of the coding in the while loop to make the seven segment works.
7. The analysis of the state diagram and the next state table of the 4-bit counter to make it works properly.

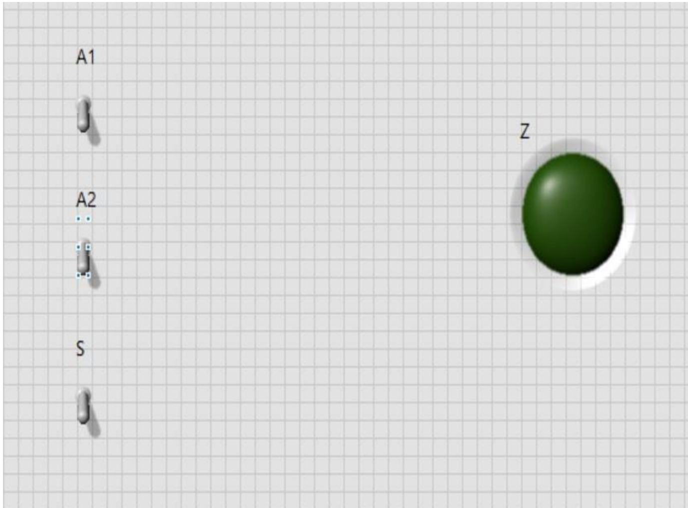
Virtual Instruments:

- 1-bit multiplexer.
- 7 segment display.
- Clock.
- D type flip-flop.
- 4-bit counter.

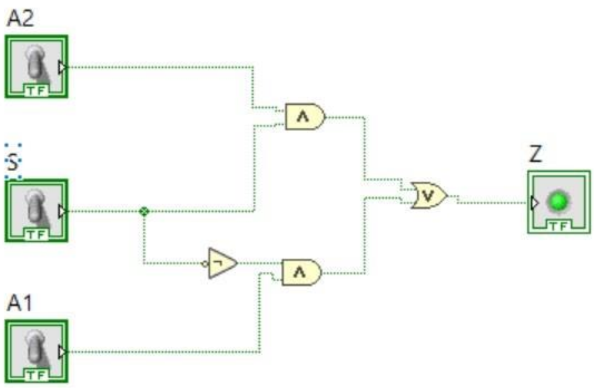
<u>1. 2-to-1 multiplexer</u>:

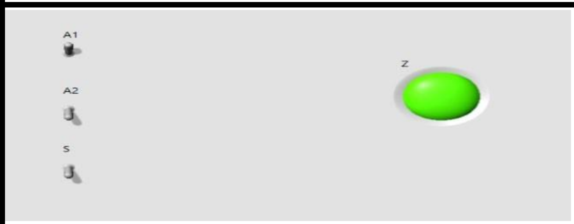
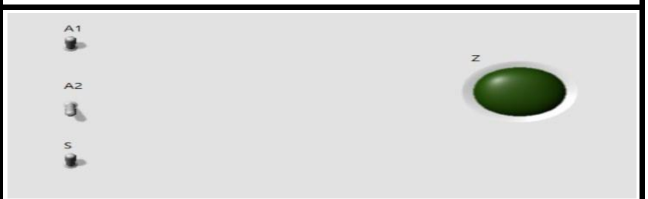
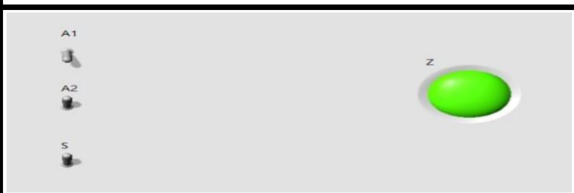
This VI is used to determine whether the counter will count up or down.

<u>front panel view:</u>



<u>Block diagram view:</u>



S	A1	A2	The output	
0	1	x		
0	0	X		
1	x	0		
1	x	1		

The analysis of the VI:

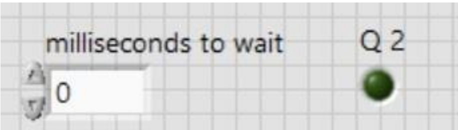
The aim of the mux is to use it as a digital switch. It will accept the binary information from the two input lines (A1 and A2). And depending on the state of the select line a particular input line is routed onto the output line.

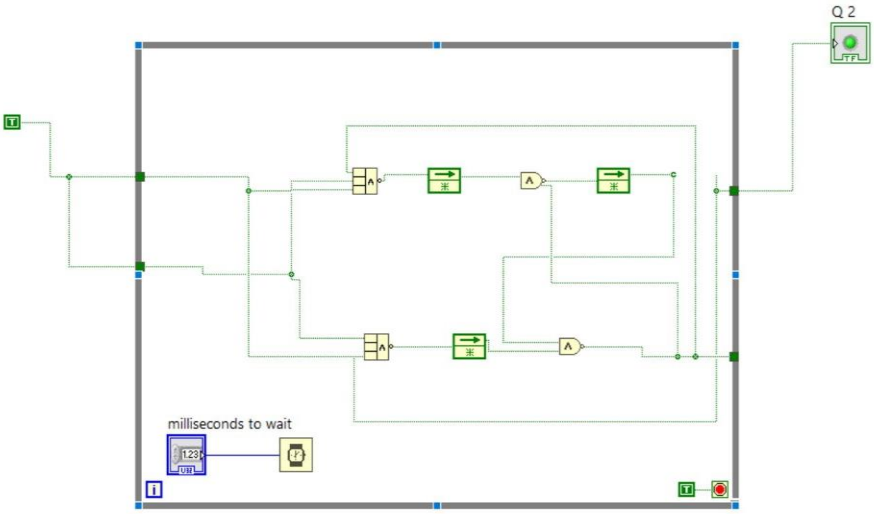
The components of the VI:

- Vertical toggle switch.
- Boolean (AND).
- Boolean (NOT).
- Boolean (OR).
- Boolean (round LED).

2.Clock:

the clock signals control the output of the counter.





The analysis of the VI:

An input high is applied to the JK flip-flop to make the flip-flop in the toggle mode. which will make it acts as an inverter with every new loop. as it toggles the signals by a given time interval (milliseconds to wait). And to control the time delay of clock a while loop is used.

- **The while loop:**

It is a structure used to execute a block of LabVIEW code repeatedly until a given condition is met.

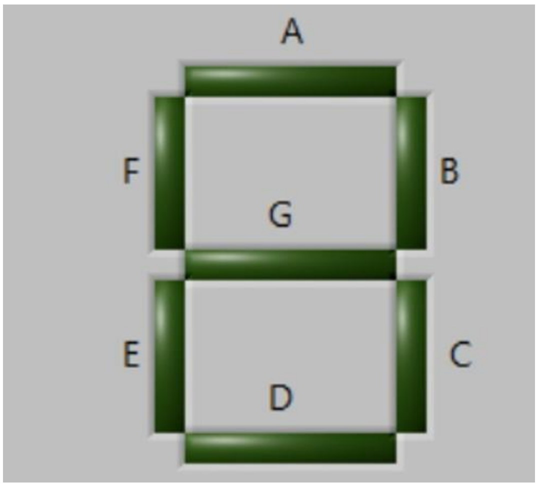
It consists of:

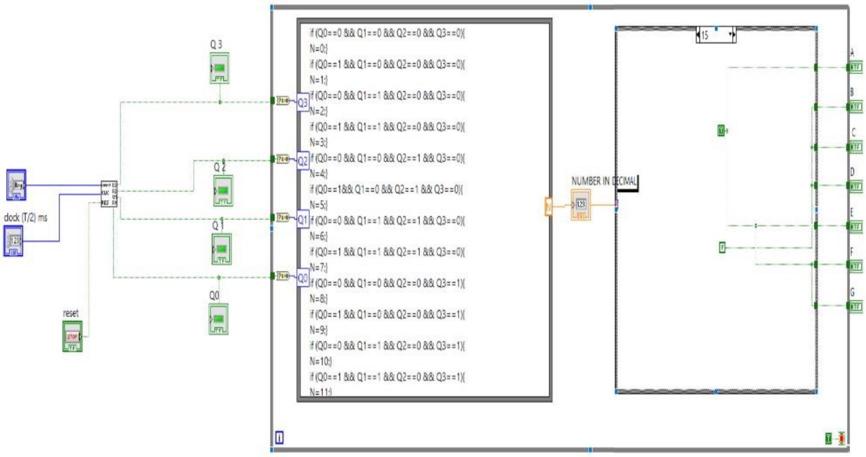
- Loop iteration: the number of repetitions starting from zero.
- Loop condition: the condition of terminating the loop and accepts Boolean values.
  - ✓ Stop if true: terminates if (1).
  - ✓ Continue if true: terminates if (0) as used in the VI.
- Milliseconds to wait: It is used to adjust the time between two consecutive loops.

3.7-segment display:

A seven-segment display is a form of electronic display device for displaying decimal numerals that is an alternative to the more complex dot matrix displays . And is used to display the output of the counter in an easy graphical form.

Front panel view: **block diagram view :**

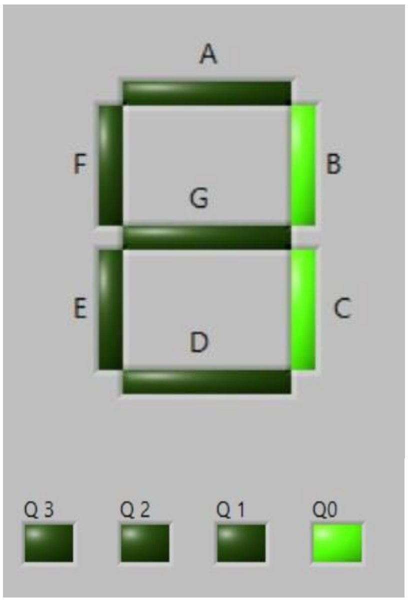


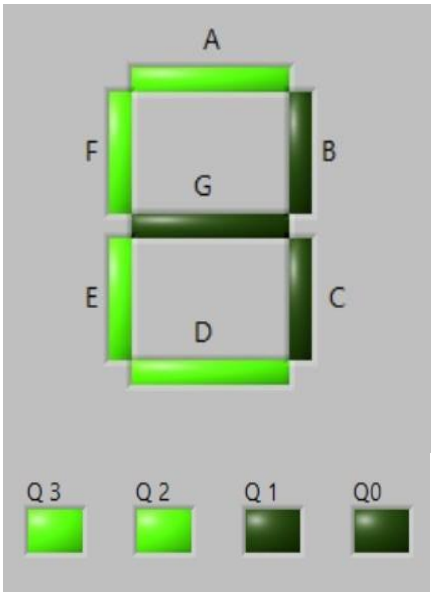


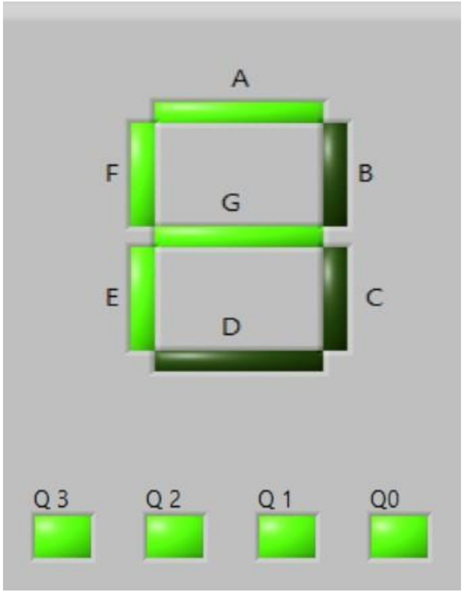
The analysis of the VI :

- The 7-segment require four binary inputs (Q0, Q1, Q2, Q3) which are the outputs of the counter to form the 4-bit binary number that will appear in the display.
- Each of the four inputs is connected to (If) function to check the states of each input.
- Example:
  If the values of (Q0, Q1, Q2, Q3) are (1,1,1,0) this indicating a decimal value Of (7).
- This decimal value is transferred to the 7-segment and the appropriate LEDS will be on.
- The whole VI inserted in a while loop to repeat the process every time the input changes and the while loop is connected to the clock to make the inputs and outputs synchronized.

Samples on the output of the 7-segment with their equivalent inputs:



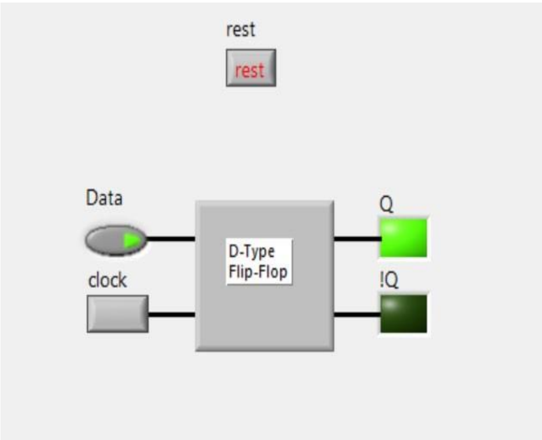


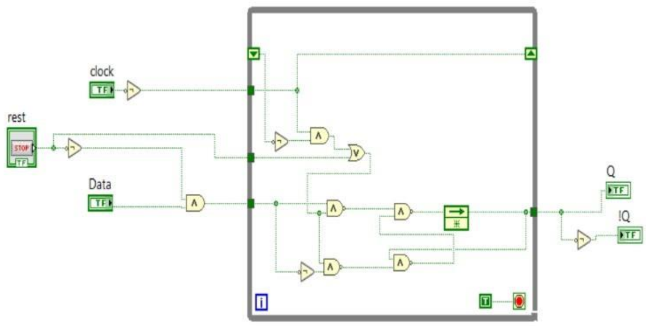


4. D-Flip-flop:

The flip-flop is an essential part in the design of the counter. it works as a memory element of the present state which will be used to know the next state of the output, and this makes the counter works properly and in right order.

front-panel view: **block diagram view:**





D	**QN**	**QN+1**	**S**	**R**
0	0	0	0	X
0	1	0	0	1
1	0	1	1	0
1	1	1	X	0

The procedures to convert the SR flip-flop to D flip-flop:

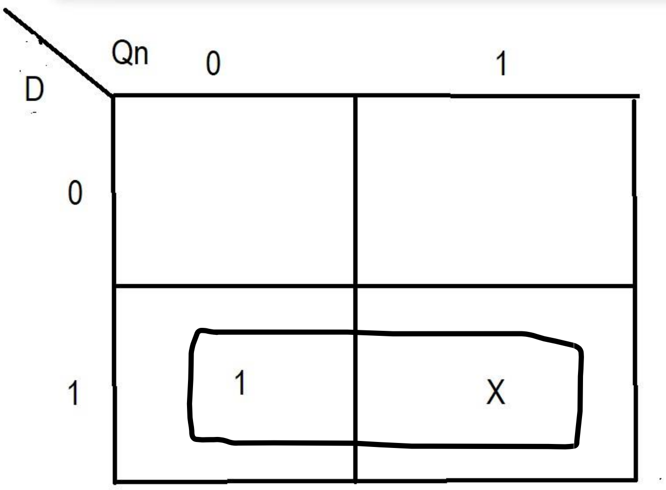
Truth table of the d flip flop:

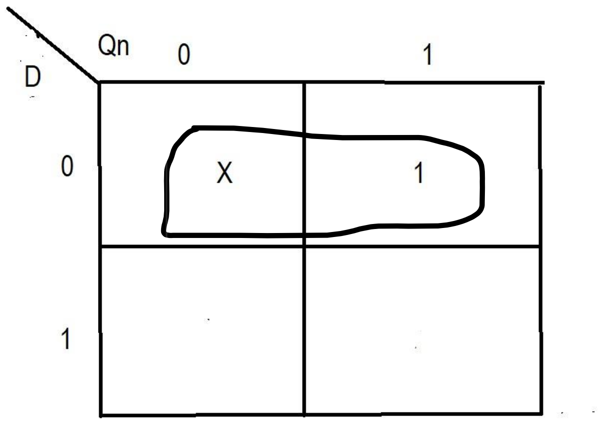
D	QN	QN+1
0	X	0
1	X	1

The excitation table of SR flip flop:

QN	QN+1	S	R
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0

KARNAUGH MAPS:

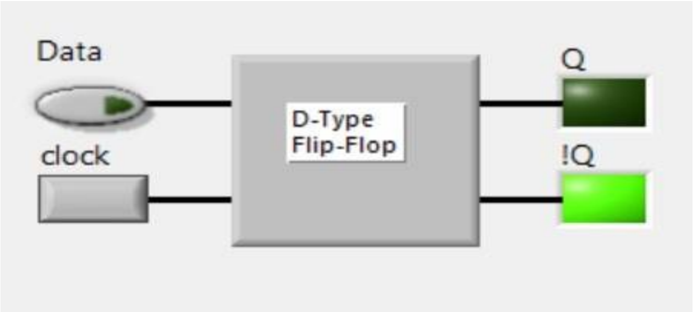


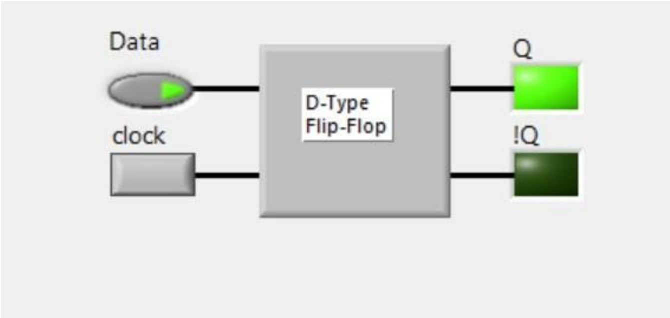


D=S

D=R’

Samples on the output of the flip-flop :

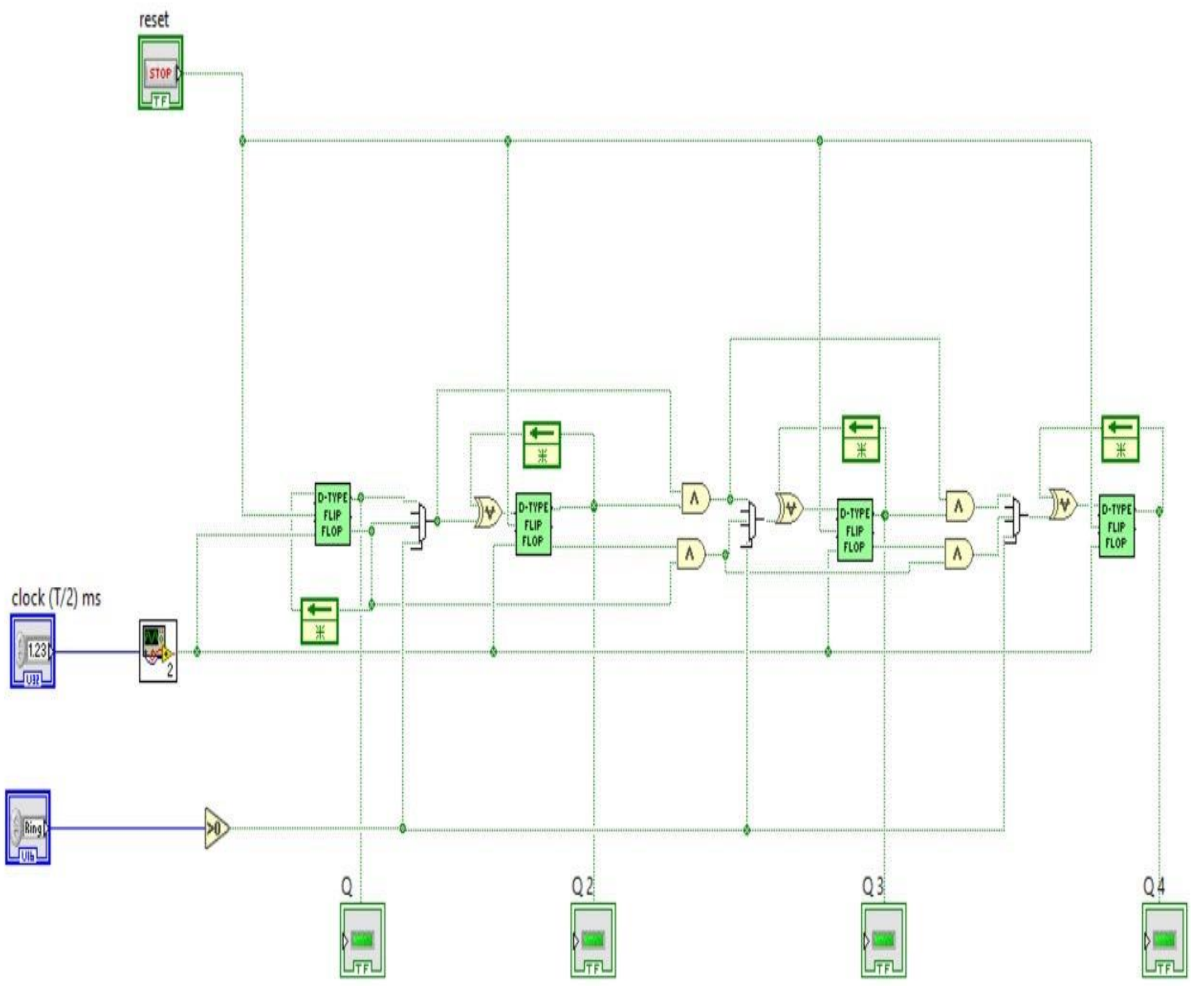




4-bit counter:

It is a device that counts a hexadecimal number depends on Filb-Flob that stored the previous number count for the next number for up or down.

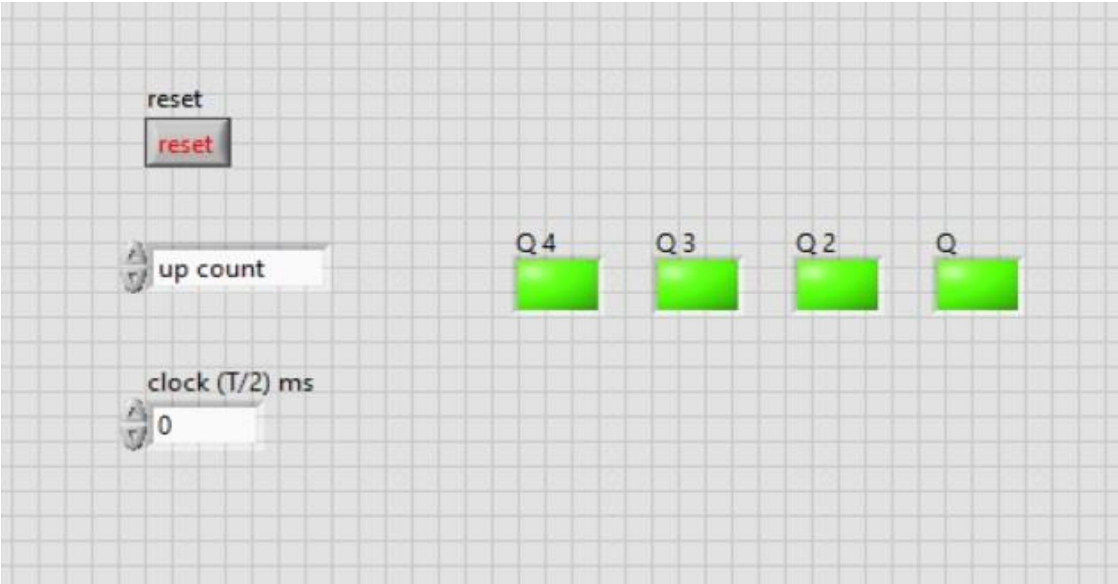
- Implementation of 4-bit counter in LabVIEW:



Comment:

We use the mux to change between the up state to down state and the reverse correctly.

- The user interface:



NOTE: the reset button to go to zero number at any number state

Samples of result:

Number state	Result in program
0000	
0001	
0011	
1111