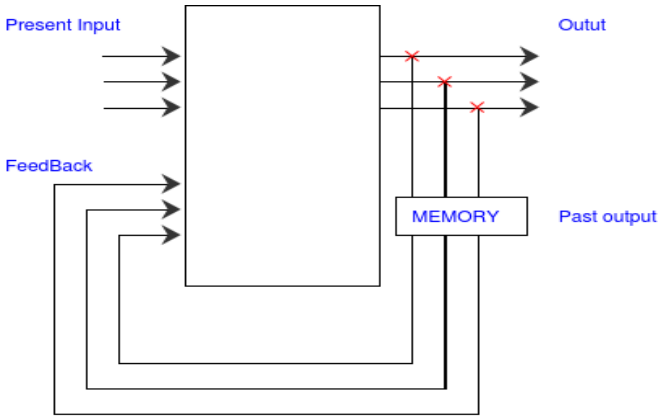
INTRODUCTION

The logic gates (AND, OR, NAND, NOR, INVERT) used in **combinational logic**. we will deal with data storage circuitry that will **latch** on to (remember) a digital state (1 or 0).
This new type of digital circuitry is called **sequential logic**



because it is controlled by and used for controlling other circuitry in a specific sequence dictated by a control clock or enable/disable control signals.
The simplest form of data storage is the Set–Reset (*S*-*R*) flip-flop.

The types of logic circuit

1-Synchronous …one of the inputs is clock …no change in output without (CLK) pulses.

2- ASynchronous …No clock pulses in inputs … output change dependent to inputs pulses.

S-*R* Flip-Flop

The *S*-*R* **flip-flop** is a data storage circuit that can be constructed using the basic gates Using a cross-coupling scheme with two NOR gates, we can form the flip-flop shown in Figure 1.

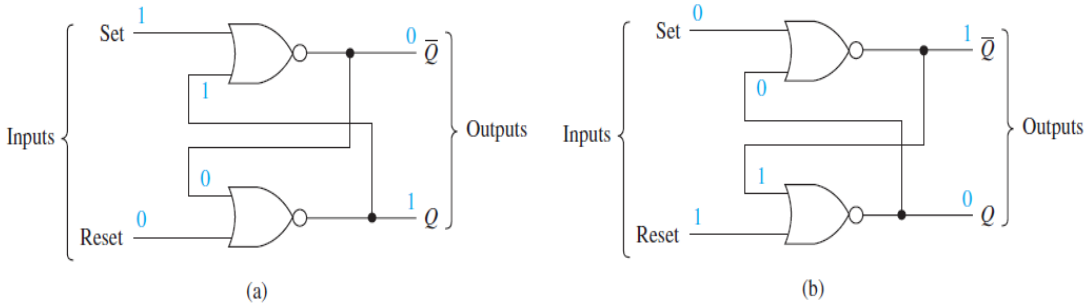


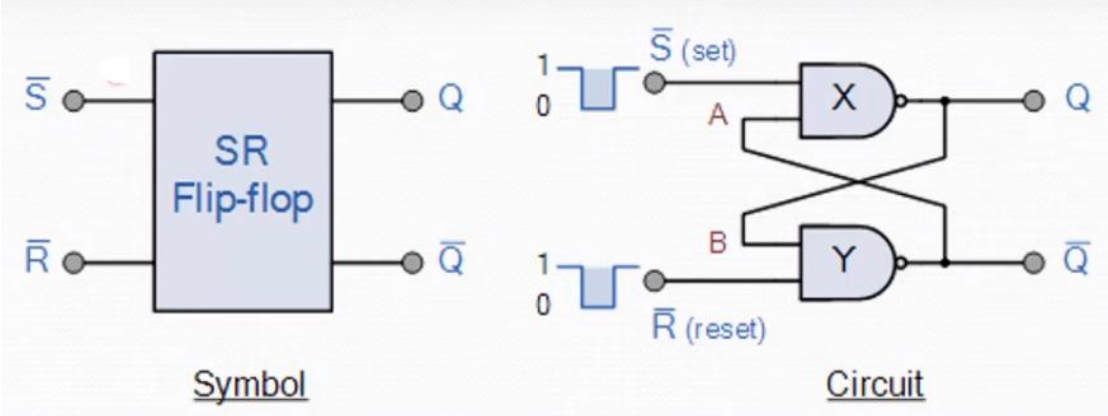
Figure 1 Cross-NOR S-R flip-flop: (a) Set condition; (b) Reset condition.

TABLE 1 Function Table for Figure 1-1

	S	R	Q	$\overline{Q}$	Comments
(b)	0	0	Q	$\overline{Q}$	*Hold* condition (no change)
(a)	1	0	1	0	Flip-flop Set
(c)	0	1	0	1	Flip-flop Reset
	1	1	0	0	Not used

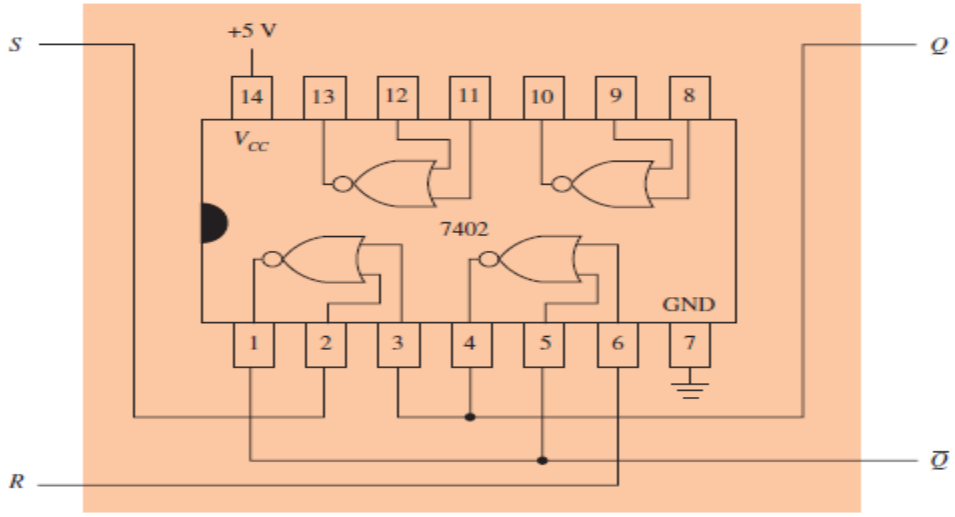
1-(case a)when S=1 , and R=0 .then Q=1,and $\overline{Q}$ =0
If we change only(S=0), and still R=0 ,IT IS
CALLED NO CHANGE (CASE b)

2-(case c)when S=0 , and R=1 .then Q=0,and $\overline{Q}$ =1
If we change only(R=0), and still S=0 ,IT IS
CALLED NO CHANGE (CASE b)
Q=0,and $\overline{Q}$ =1



Figure–2 Cross-NAND S-R flip-flop.

State	S	R	Q	$\overline{Q}$	Description
Set	1	0	0	1	Set $\overline{Q}$ » 1
	1	1	0	1	no change
Reset	0	1	1	0	Reset $\overline{Q}$ » 0
	1	1	1	0	no change
Invalid	0	0	1	1	Invalid Condition

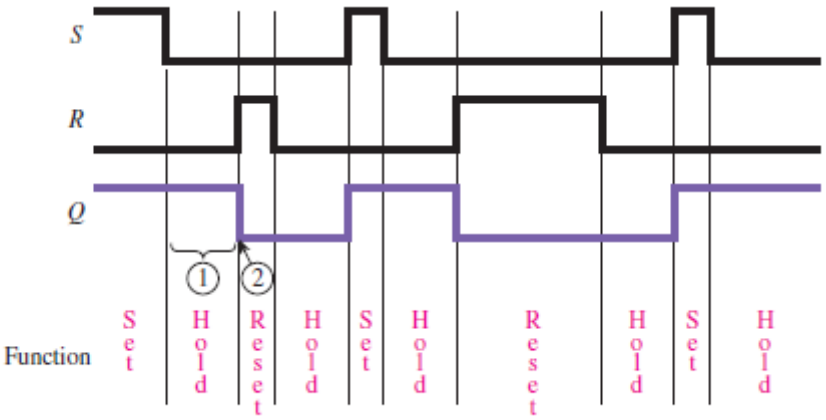


S-R flip-flop connections using a 7402 TTL IC.

EXAMPLE–1

To the *S-R* flip-flop shown in Figure, we connect the *S* and *R* waveforms given in Figure. Sketch the *Q* output waveform that will result.

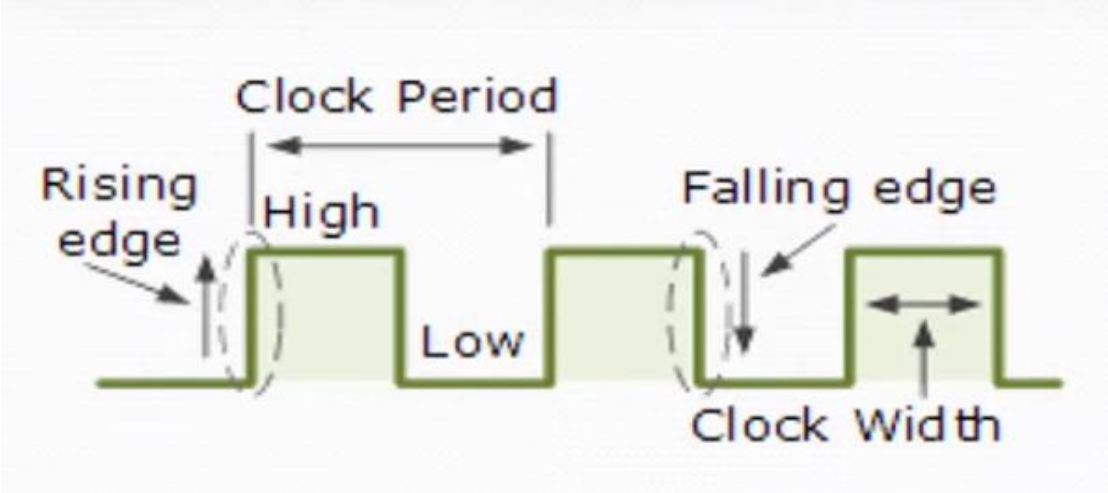
Solution:



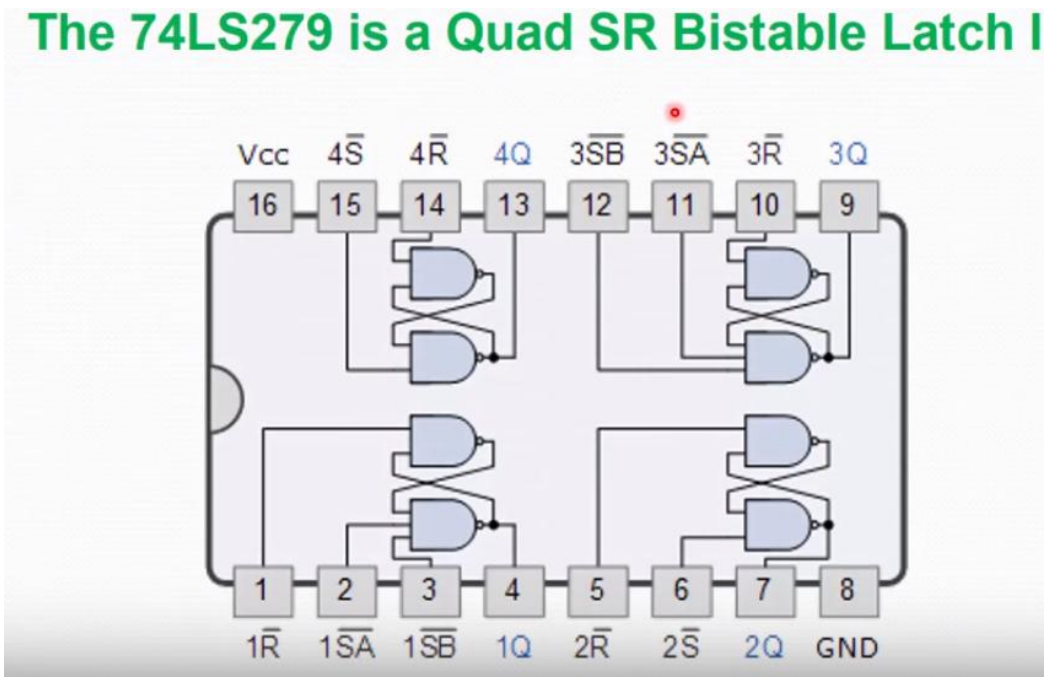
Notes:

1. The flip-flop is latched (held) in the Set condition even after the HIGH is removed from the *S* input.

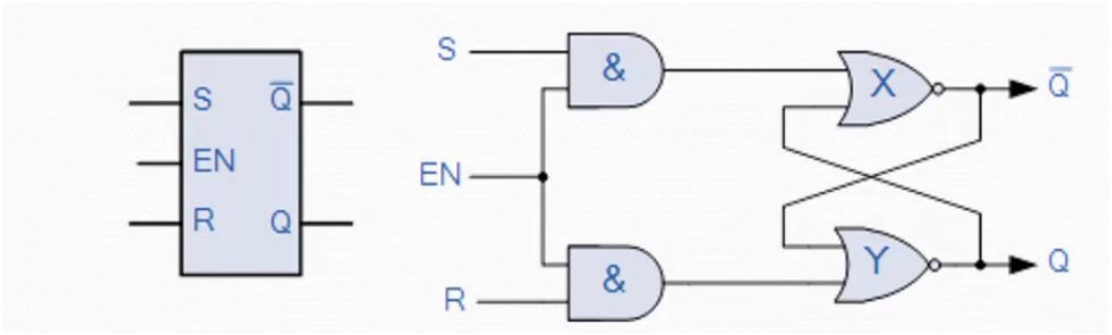
2. The flip-flop is considered transparent because the *Q* output responds immediately to input changes.



Switching diagram



Gated SR Flip-flop

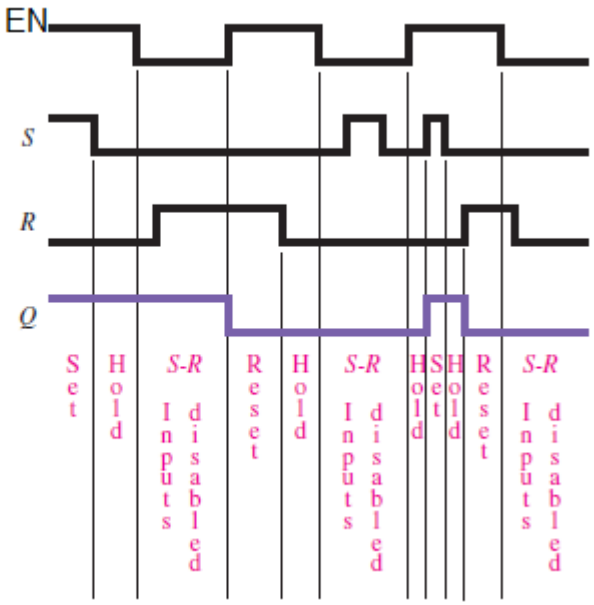


IF enable(EN=1) IT will be ACTIVE
EN=0 not active
To read switching diagram
1-when S=0, R=0 ,CIK=0 ,…then Q=O
2- when S=1, R=0 ,CIK=1 ,…then Q=1….still to another clk pulls
3- when S=0, R=1 ,CIK=0 ,…then Q=1…for the same pulls clk in 2
4-when the new CLK pulls come and when S=0, R=1
CIK=1,…then Q=O
5- when the new CLK pulls come and when S=1, R=0
CIK=1,…then Q=1
6- when the new CLK pulls come and continues then
When change S OR R changing Q…if S=1 ,R=0 , Q=1
And if S=0 ,R=1 , Q=0

EXAMPLE -2

Feed the EN, S, and R inputs in Figure into the gated S-R flip-flop sketch the output wave at Q, and list the flip-flop functions.

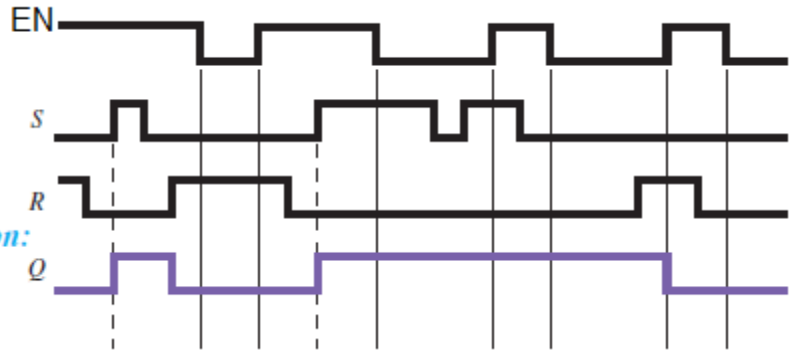
Solution:

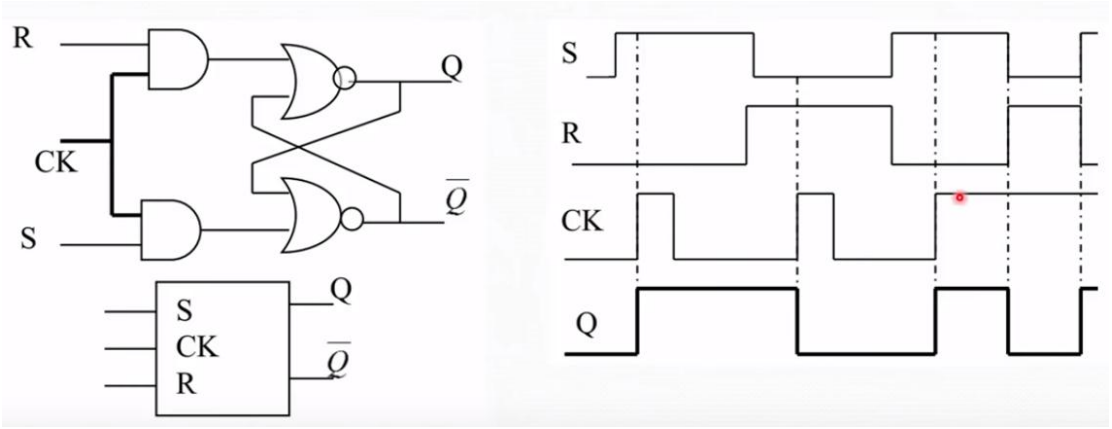


EXAMPLE–3

Feed the *EN*, *S*, and *R* inputs in Figure into the gated *S*-*R* flip-flop, and sketch the output wave at *Q*.

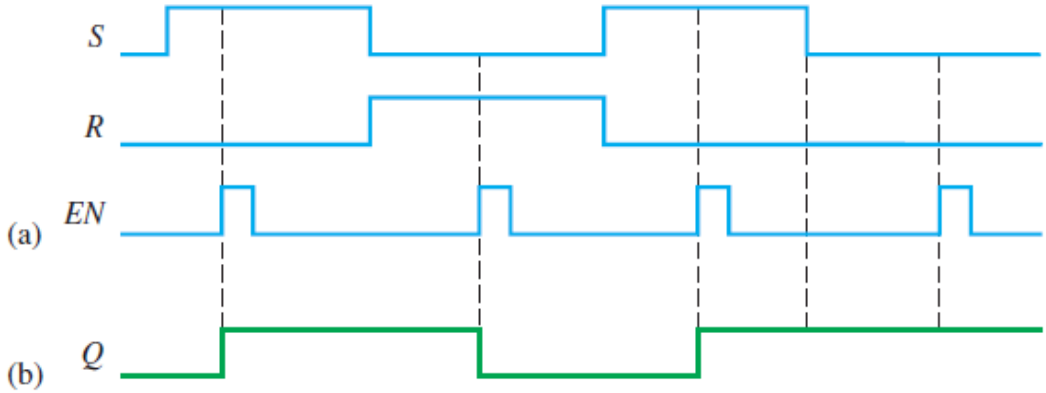
Solution:





EXAMPLE-4

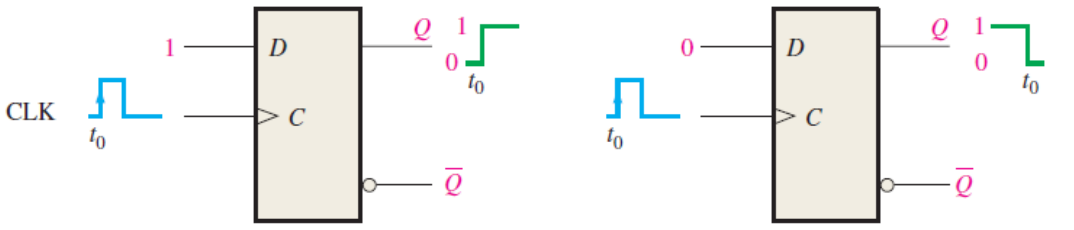
Determine the *Q* output waveform if the inputs shown in Figure are applied to a gated S-R latch that is initially RESET.



D-FLIP-FLOP

D Flip-flop

Data Latch
Delay flip-flop



(a) $D = 1$ flip-flop SETS on positive clock edge. (If already SET, it remains SET.)

(b) $D = 0$ flip-flop RESETS on positive clock edge. (If already RESET, it remains RESET.)

FIGURE Operation of a positive edge-triggered D flip-flop.

TABLE

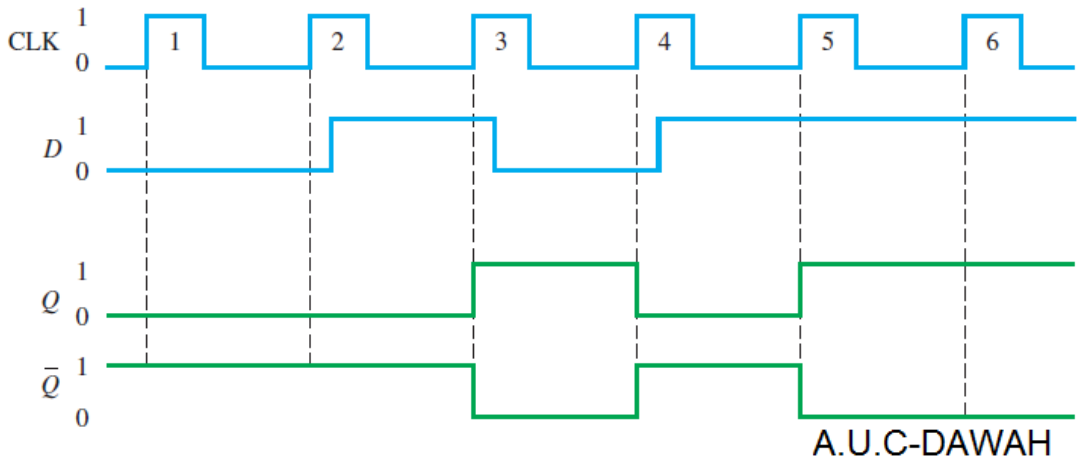
Truth table for a positive edge-triggered D flip-flop.

Inputs		Outputs		
D	CLK	Q	$\overline{Q}$	Comments
0	↑	0	1	RESET
1	↑	1	0	SET

↑ = clock transition LOW to HIGH

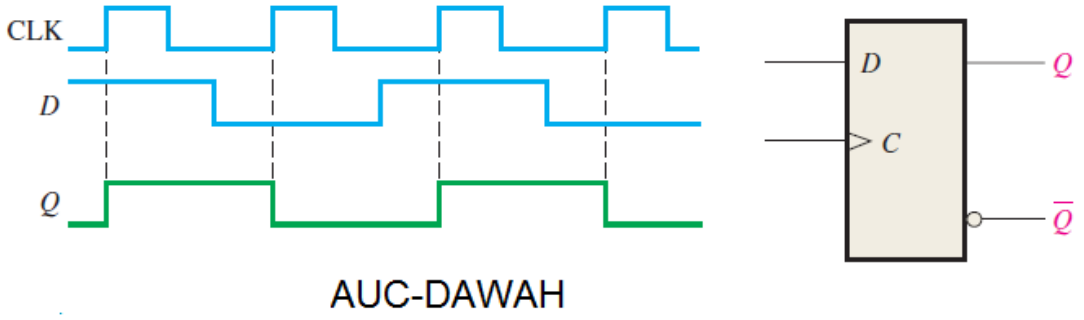
EXAMPLE -5

Determine the Q and $\overline{Q}$ output waveforms of the flip-flop for the D and CLK inputs in Figure Assume that the positive edge-triggered flip-flop is initially RESET.



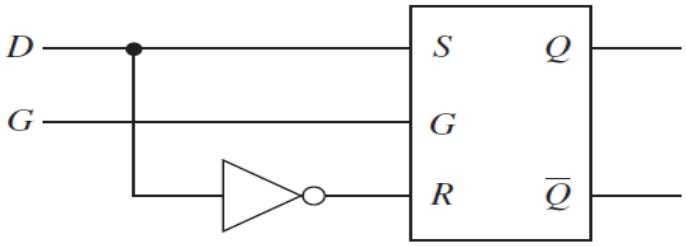
EXAMPLE-6

Given the waveforms in Figure for the *D* input and the clock, determine the *Q* output waveform if the flip-flop starts out RESET.



Gated D Flip-Flop

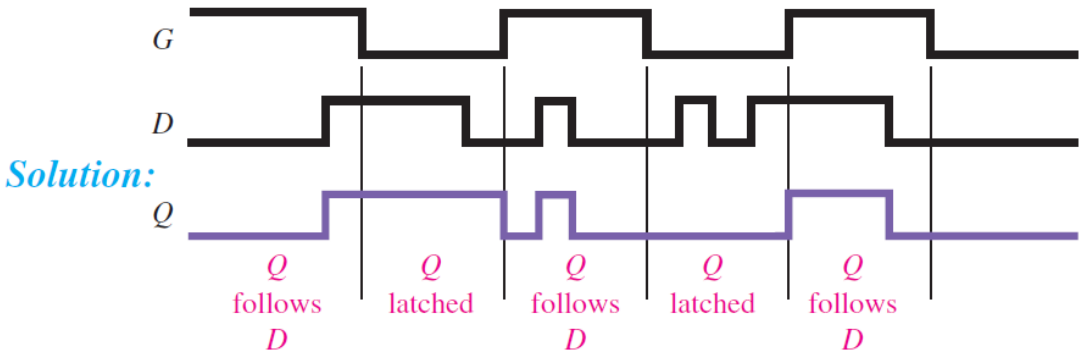
Another type of flip-flop is the D flip-flop (Data flip-flop). It can be formed from the gated S-R flip-flop by the addition of an inverter. This enables just a single input (D) to both Set and Reset the flip-flop In Figure, we see that S and R will be complements of each other, and S is connected to a single line labeled D (Data). The operation is such that Q will be the. same as D while G is HIGH, and Q will remain latched when G goes LOW. (Latched means that Q remains constant regardless of changes in D.)

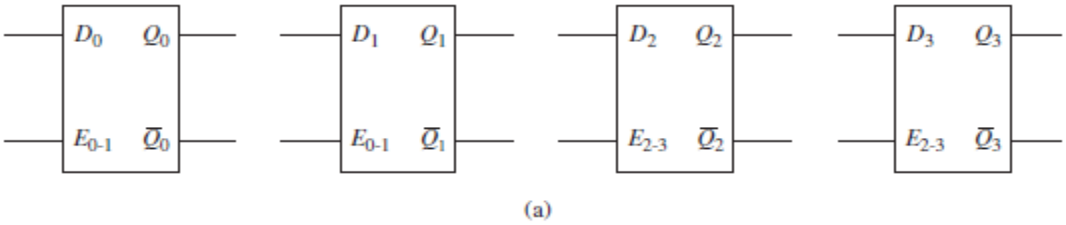


Gated *D* flip-flop.

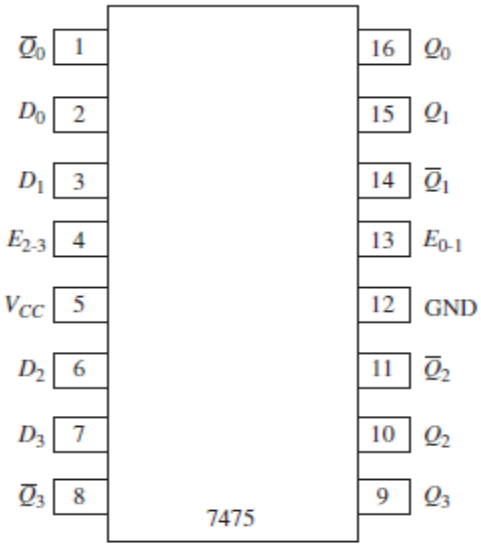
EXAMPLE–7

Sketch the output waveform at *Q* for the inputs at *D* and *G* of the gated *D* flip-flop in Figure.





(a)



(b)

The 7475 quad bistable *D* latch: (a) logic symbol; (b) pin configuration.

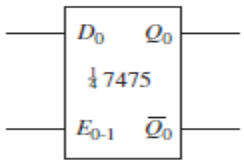
TABLE Function Table for a 7475[a]

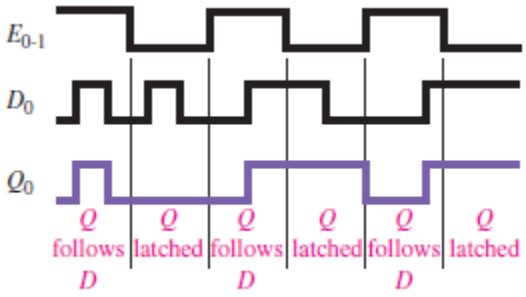
	Inputs		Outputs	
Operating Mode	***E***	***D***	***Q***	$\overline{Q}$
Data enabled	H	L	L	H
Data enabled	H	H	H	L
Data latched	L	x	q	$\overline{q}$

[a] q = state of Q before the HIGH-to-LOW edge of E; x = don't care.

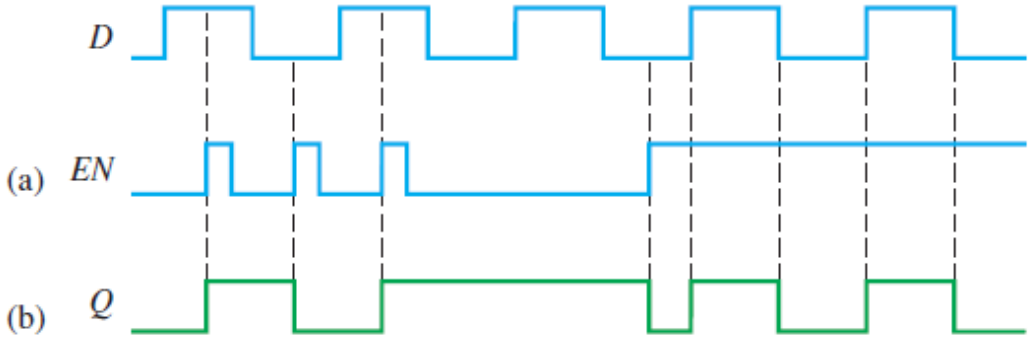
EXAMPLE–8

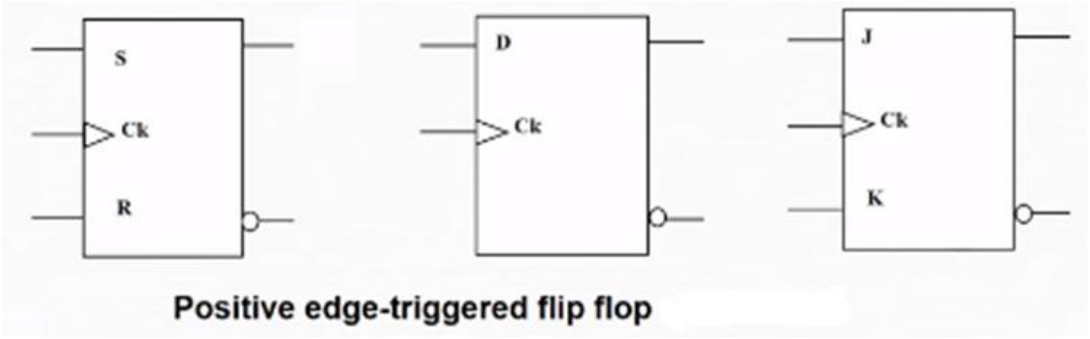
For the inputs at D_0 and E_0 –1 for the 7475 *D* latch shown in Figure sketch the output waveform at Q_0 in Figure

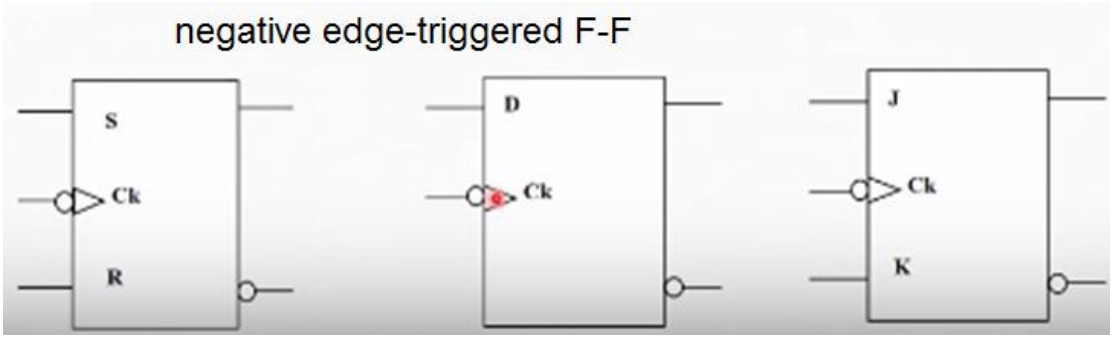


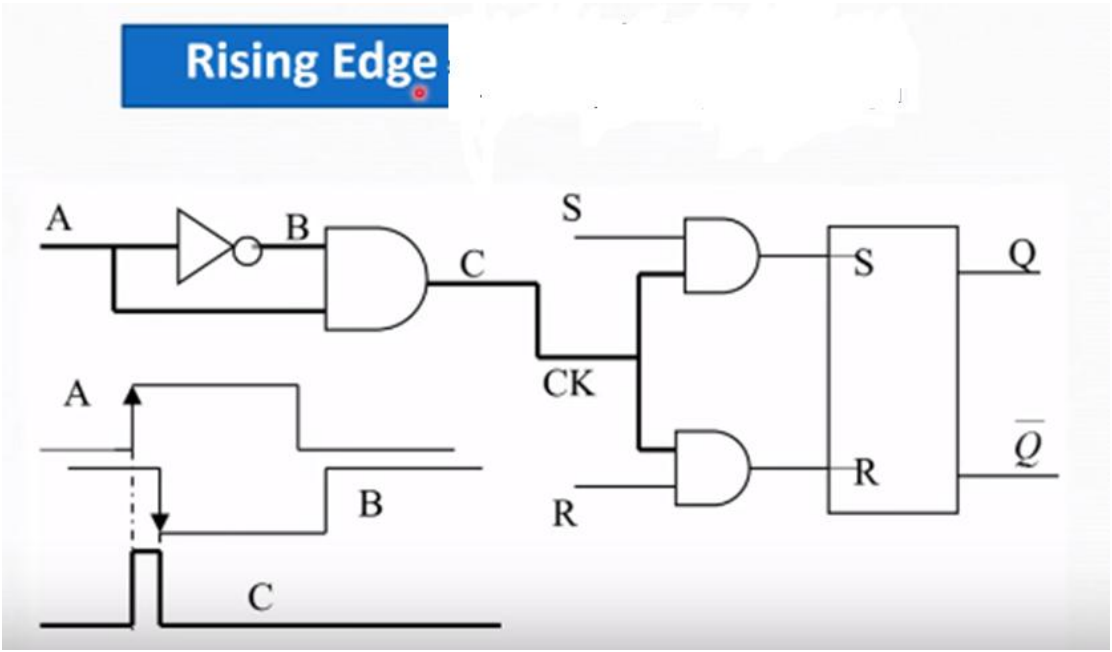


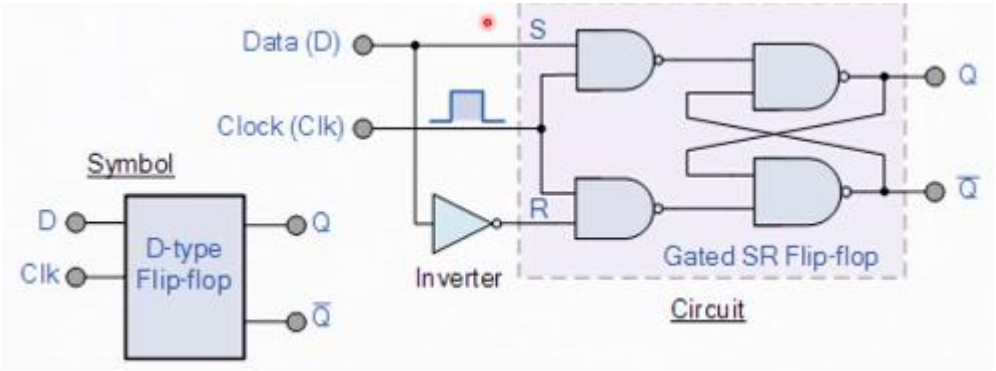
EXAMPLE-9-Determine the *Q* output waveform if the inputs shown in Figure are applied to a gated D latch, which is initially RESET.



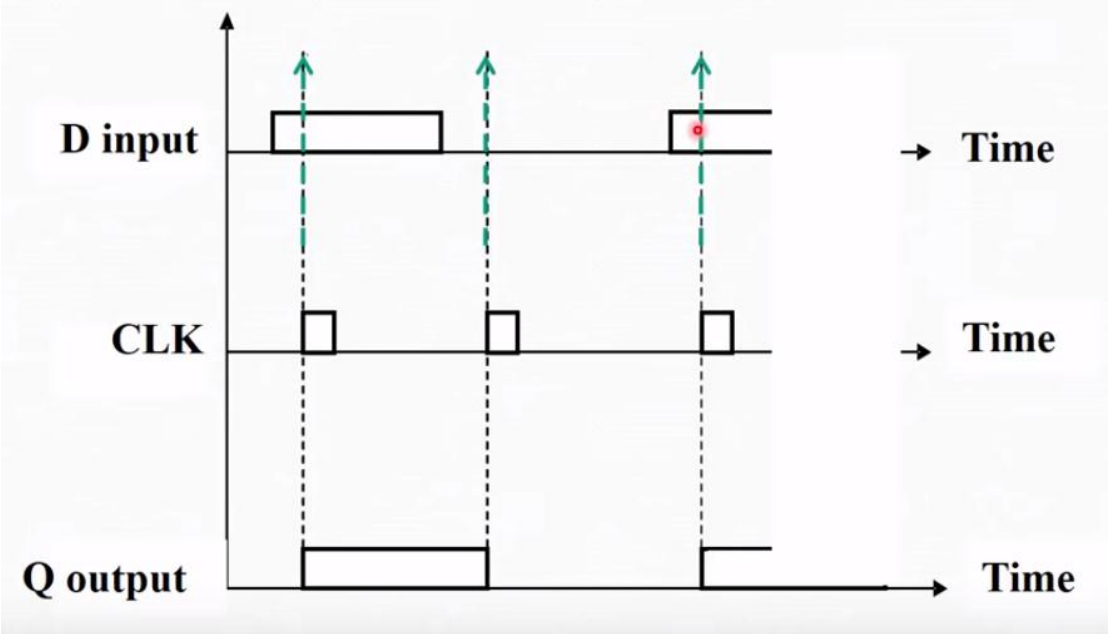

S
Ck
R
D
Ck
J
Ck
K
Positive edge-triggered flip flop


negative edge-triggered F-F
S
Ck
R
D
Ck
J
Ck
K

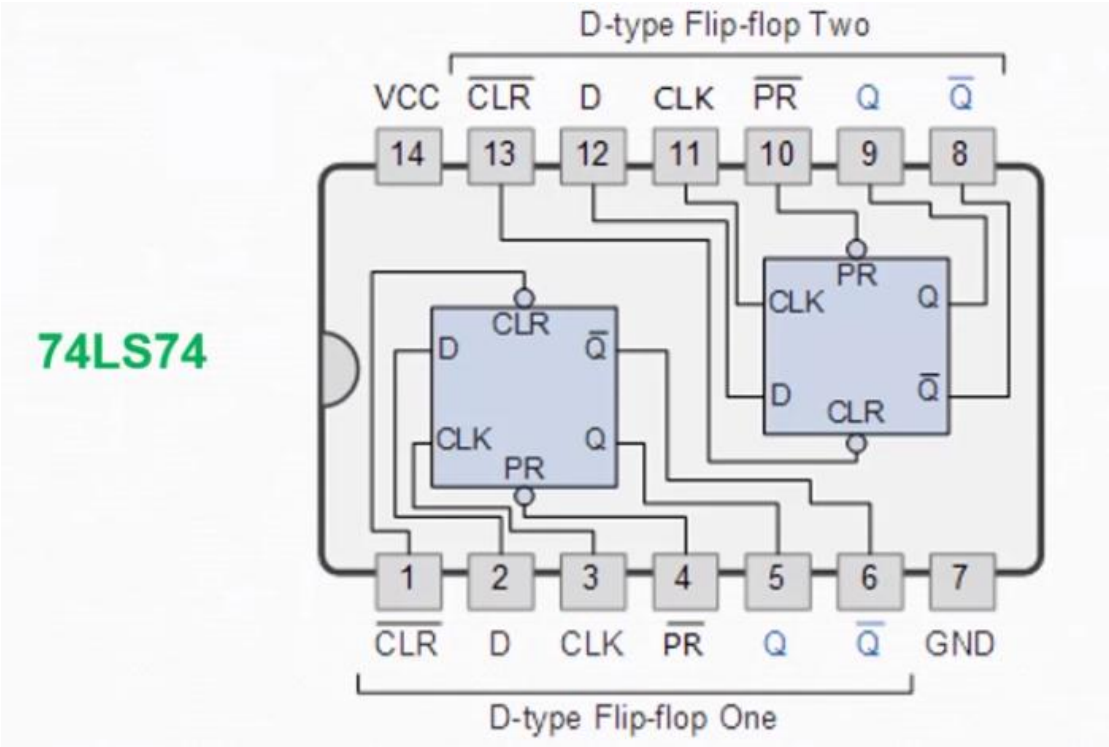

Rising Edge
A
B
C
S
CK
R
S
Q
R
A
B
C

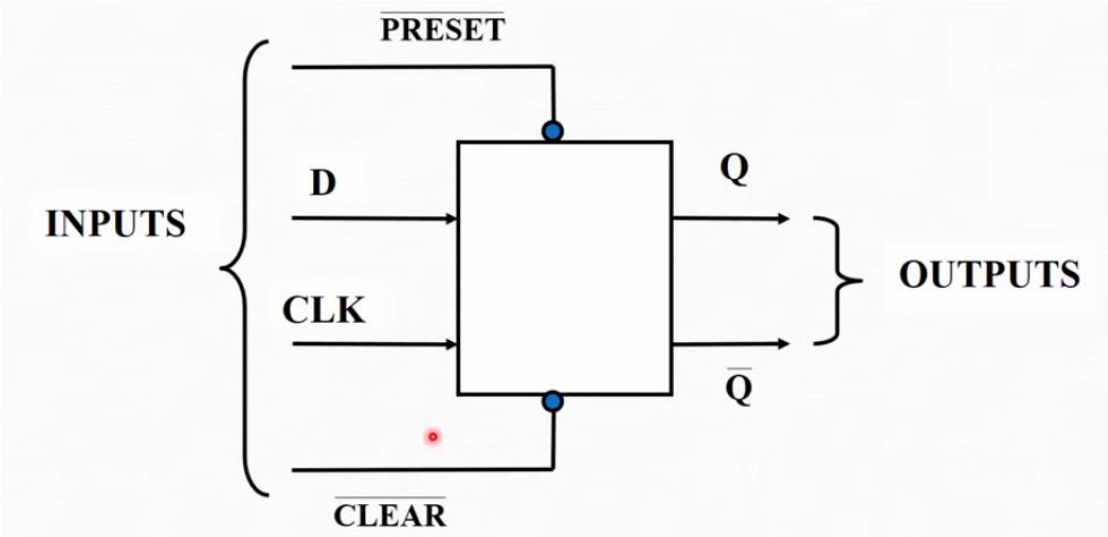


Clk	D	Q	$\overline{Q}$	Description
↓ » 0	X	Q	$\overline{Q}$	Memory no change
↑ » 1	0	0	1	Reset Q » 0
↑ » 1	1	1	0	Set Q » 1



74LS74 is a Dual D flip-flop
74LS174 HEX D flip-flop with direct clear input
74LS175 Quad D flip-flop with complementary outputs
74LS273 Octal D flip-flop containing eight D flip-flops with a clear input



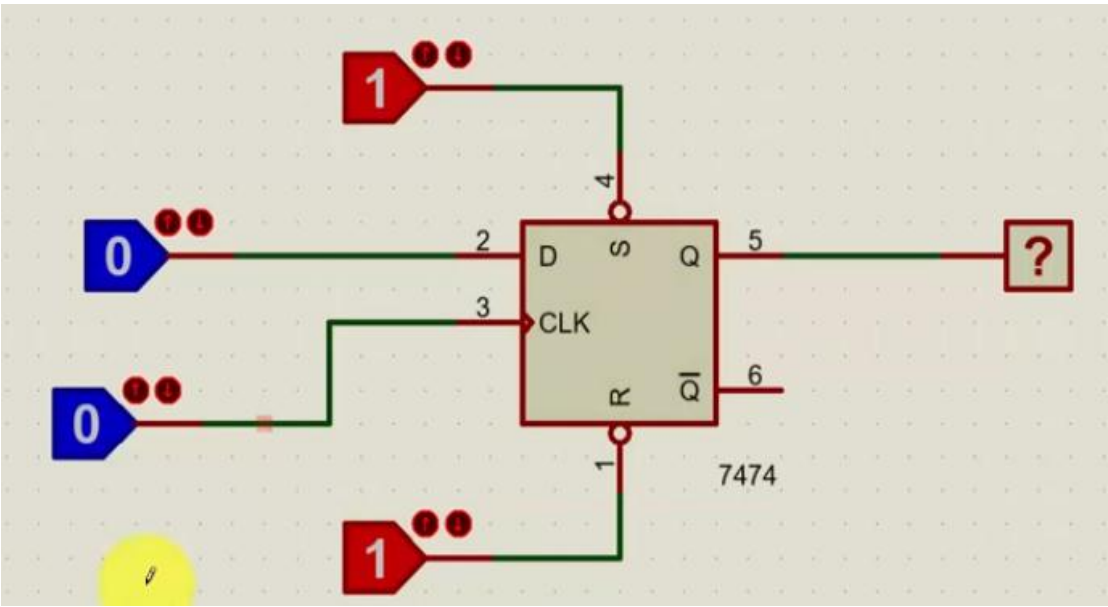


PRESET (PR) and **CLEAR** (CLR) ,,they are inputs no effect to clock in it .used to make the(F-F) in initial value

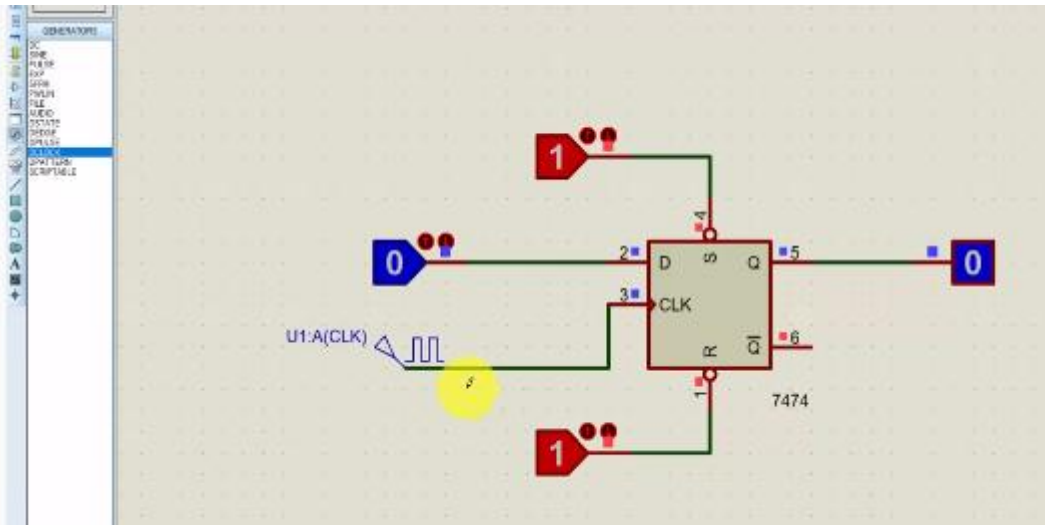
When PR=0 , then Q=1

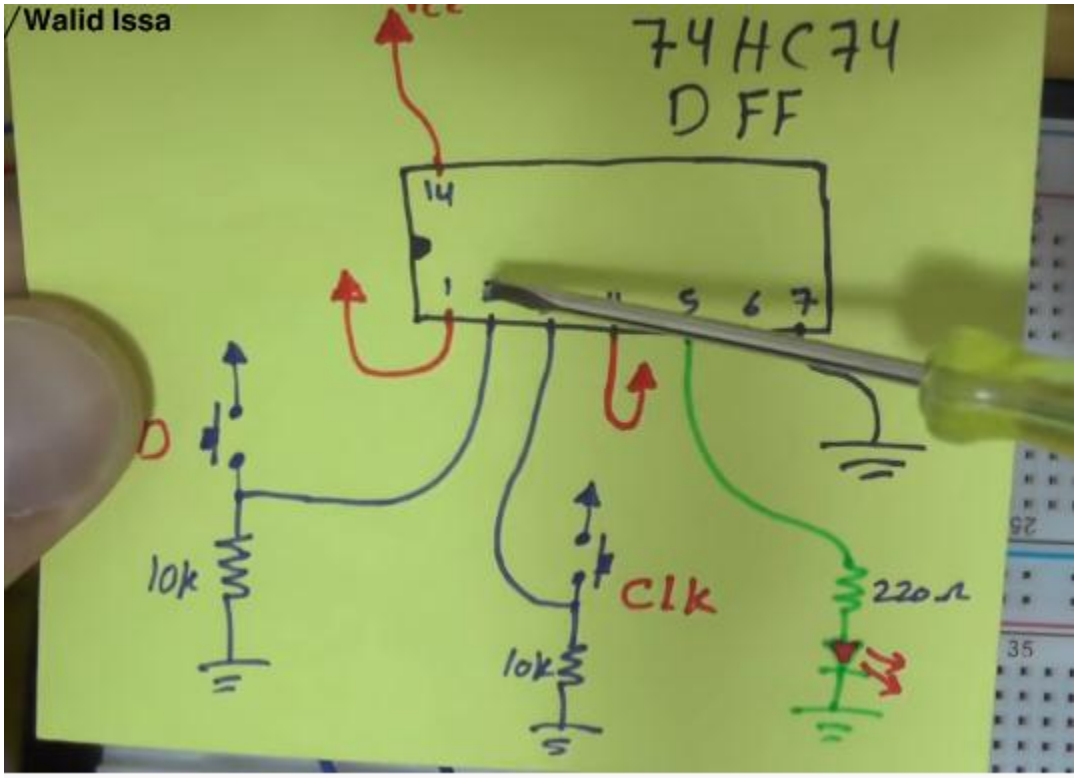
When CLR=0, then Q=0

PR=0 ,,, CLR=0…invalid condition




P
L
DEVICES
7474
LOGICPROBE (BIG)
LOGICSTATE
[7474]


1
0
2
D
S
Q
5
0
3
CLK
U1:A(CLK)
R
Q
6
7474
1


Walid Issa
74HC74
D FF
14
1
5
6
7
D
10k
Clk
10k
220Ω