

Day 4 - Counters 5 Sept

This document is created by Sweety Pinjani. All rights are reserved by the original author.
DO NOT DISTRIBUTE

DAY 4 – 111 DAYS VERIFICATION CHALLENGE

Topic: Counters, Timers

Skill: Digital Electronics

DAY 4 CHALLENGE:

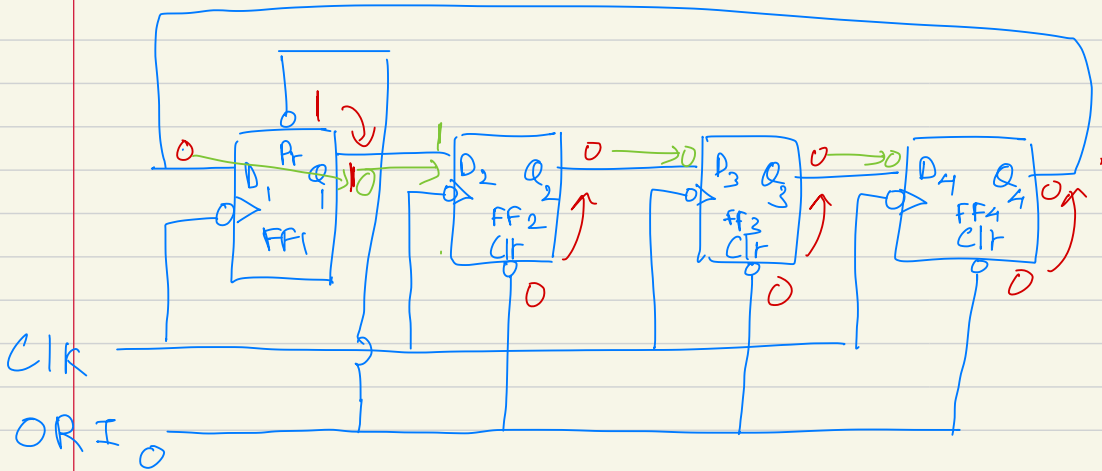
1. Design & explain working of:
 - a. 4-bit Ring counter
 - b. 4-bit Johnson counter
 - c. 3-bit Ripple counter
 - d. Decade counter
2. List the difference between:
 - a. Timers & Counters
 - b. Synchronous & Asynchronous Counter
3. Explain working of 555 Timer IC

©2024 SwitiSpeaksOfficial. All rights reserved. Author - Sweety Pinjani

0:54 / 1:57

1) Ring Counter (4 bit ring counter)

no. of States = no. of flip flops
for a ring counter.
the same value is looped around
& is fed back



Questions: \rightarrow Why have we given Preset to FF1 & reset we are giving CLR (ORI) \rightarrow override Input.

- \rightarrow What are ring counters.
- \rightarrow Why are ring counters used.
- \rightarrow How does ring counter work?

ORI \rightarrow These are Preset & clear.
So if $PR=0 \Rightarrow Q=1$
 $CLR=0 \Rightarrow Q=0$

$\left. \begin{array}{l} \text{active low} \\ \text{ORI always overrides regardless of CLK \& D.} \end{array} \right\}$ use this for above dt.

We only set the first FF to 1 so that we can initialise it to a known state & set the rest of the states as zero

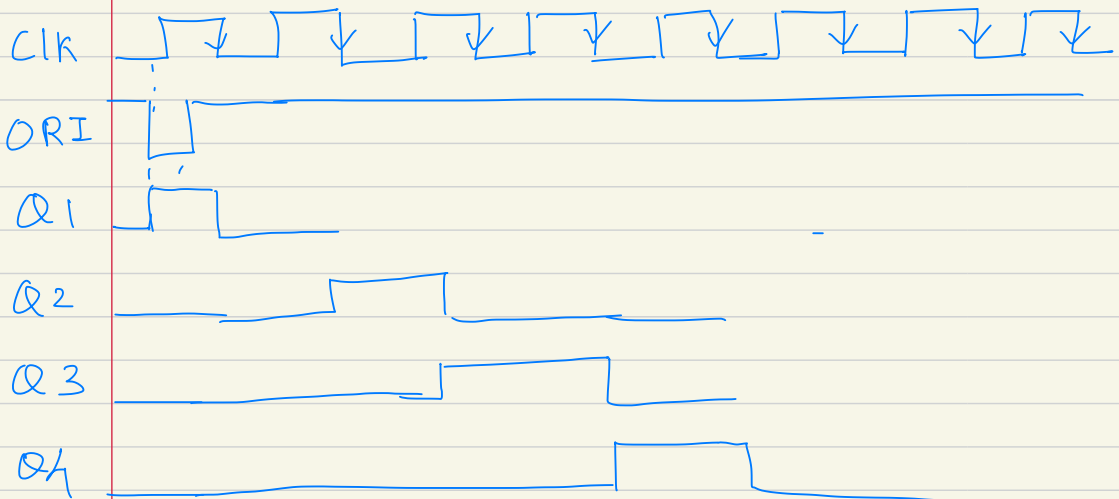
active low
↑

→ Since active low
Clk → falling edge

ORI	Clk	Q ₁	Q ₂	Q ₃	Q ₄
1	x	1	0	0	0
1	↓	0	1	0	0
1	↓	0	0	1	0
1	↓	0	0	0	1
1	↓	1	0	0	0

→ present = 1
Clk = 0

becomes a ring structure.



Applicatⁿ → ring counter helps in sequential control

Just some background on ripple counter (UP)

1) (a) 3-bit ripple counter.

* UP Counter (3 bit) & down counter.

why
→ \bar{Q} connected to CLK

$M=0$	up counting
$M=1$	down counting

→ \bar{Q} is connected to CLK.

$0 \rightarrow 1 \rightarrow 2 \rightarrow 3$ UP counter

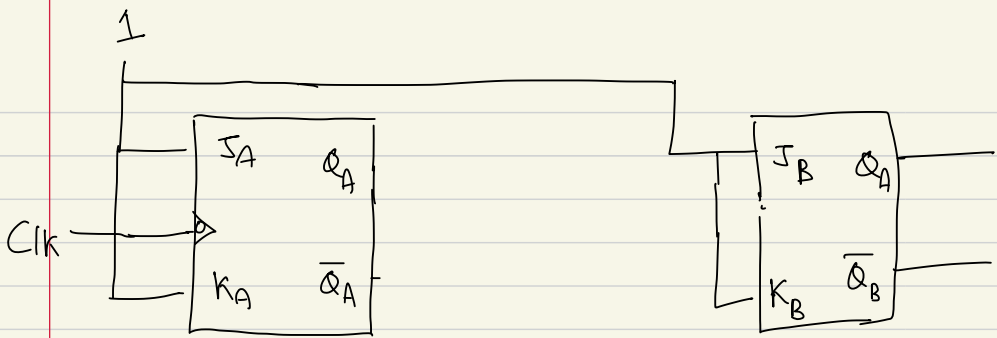
$3 \rightarrow 2 \rightarrow 1 \rightarrow 0$ Down counter.

Asynchronous Flip Flop Counter:

(Ripple means asynchronous).

why should JK be in Toggle mode?
otherwise it won't be counting in binary as the nos.
↑

M	Q	\bar{Q}
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

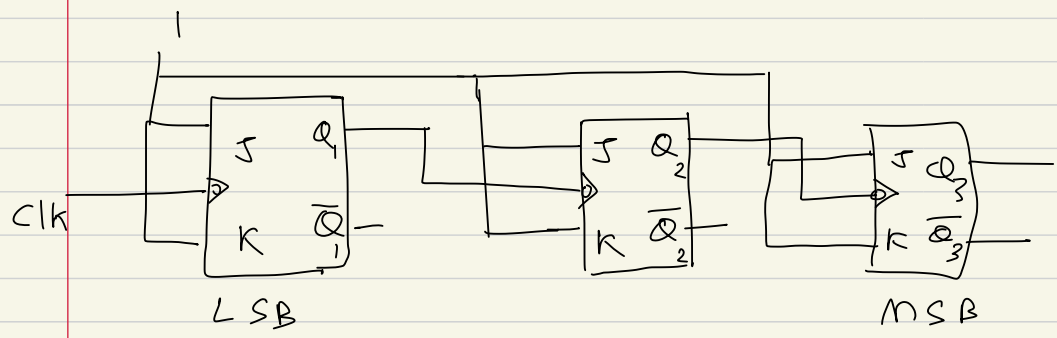


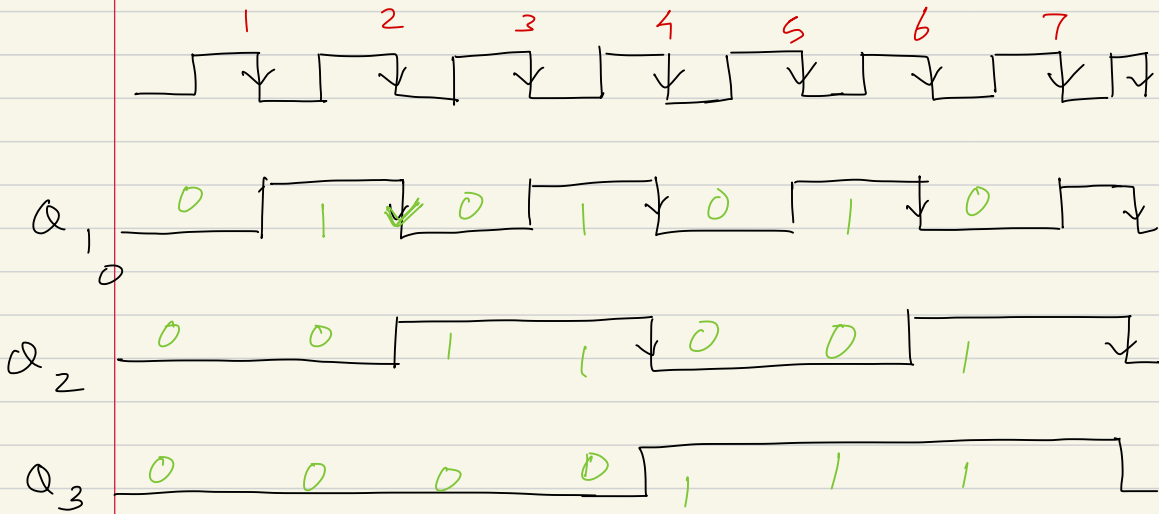
What

why

how?

UP Counter Asynchronous





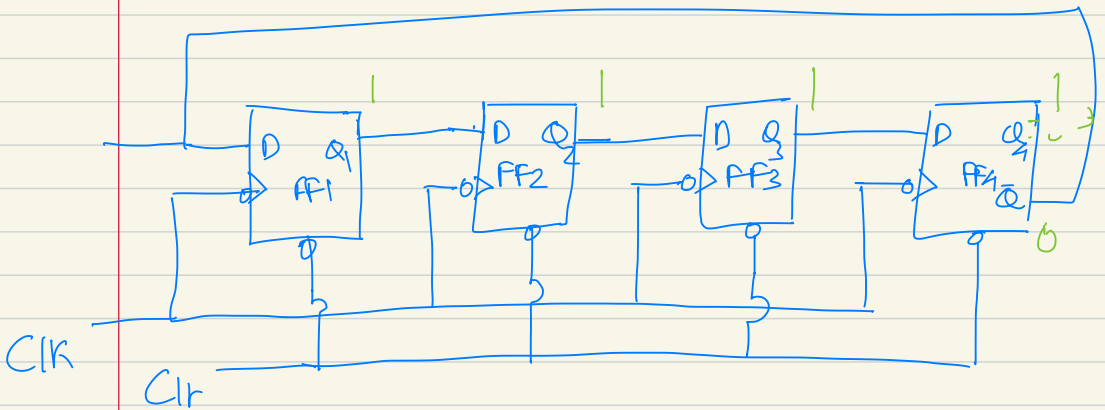
Initial 1st edge

Q_3	Q_2	Q_1
0	0	0
0	0	1
0	1	0
1	0	0
1	0	1
1	1	0

This means it is an up counter.

(b) 4 bit Johnson's Counter

The difference b/w 4-bit Johnson's Counter & ripple Counter is that there is no initial state condⁿ req^d for Johnson's Counter as opposed to ripple Counter.

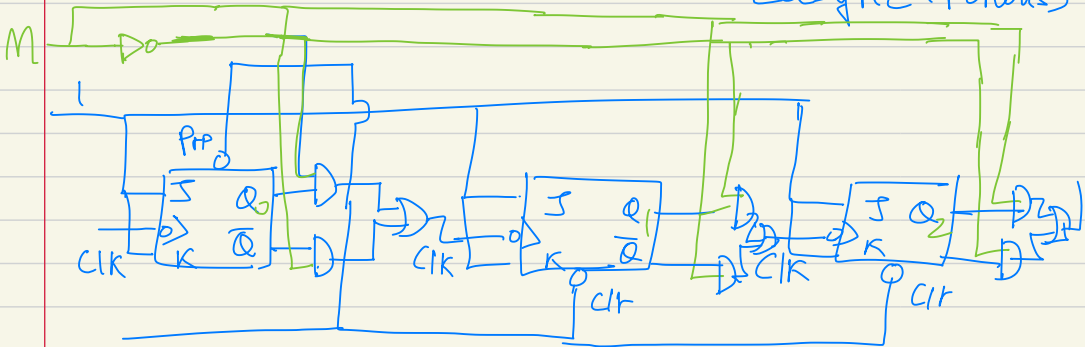


initially all the FFs are initialized at 0.

Clr	CLK	Q_1	Q_2	Q_3	Q_4	no. of States
0	X	0	0	0	0	1
1	↓	1	0	0	0	2
1	↓	1	1	0	0	3
1	↓	1	1	1	0	4
1	↓	1	1	1	1	5
"	"	0	1	1	1	6
"	"	0	0	1	1	7
"	"	0	0	0	1	8
"	"	0	0	0	0	

Hence, for 4 flip flops we can see 16 states.

(c) 3 bit ripple up/down counter (asynchronous)



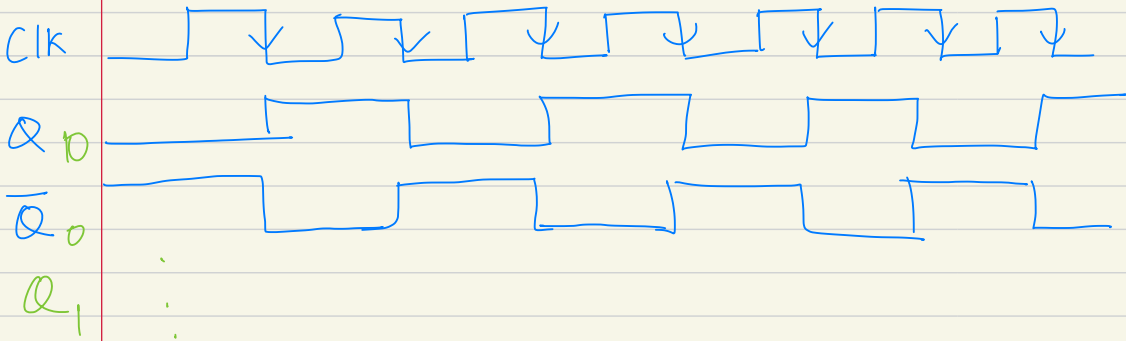
for $M=0$; y follows Q
 $M=1$; y follows \bar{Q}

M	Q	Q	Q	y	
0	0	0	0	0	} up counter
0	0	0	1	0	
0	1	0	0	1	
0	1	1	0	1	
1	0	0	0	0	} down counter
1	0	0	1	1	
1	1	0	0	0	
1	1	1	0	1	

$y =$

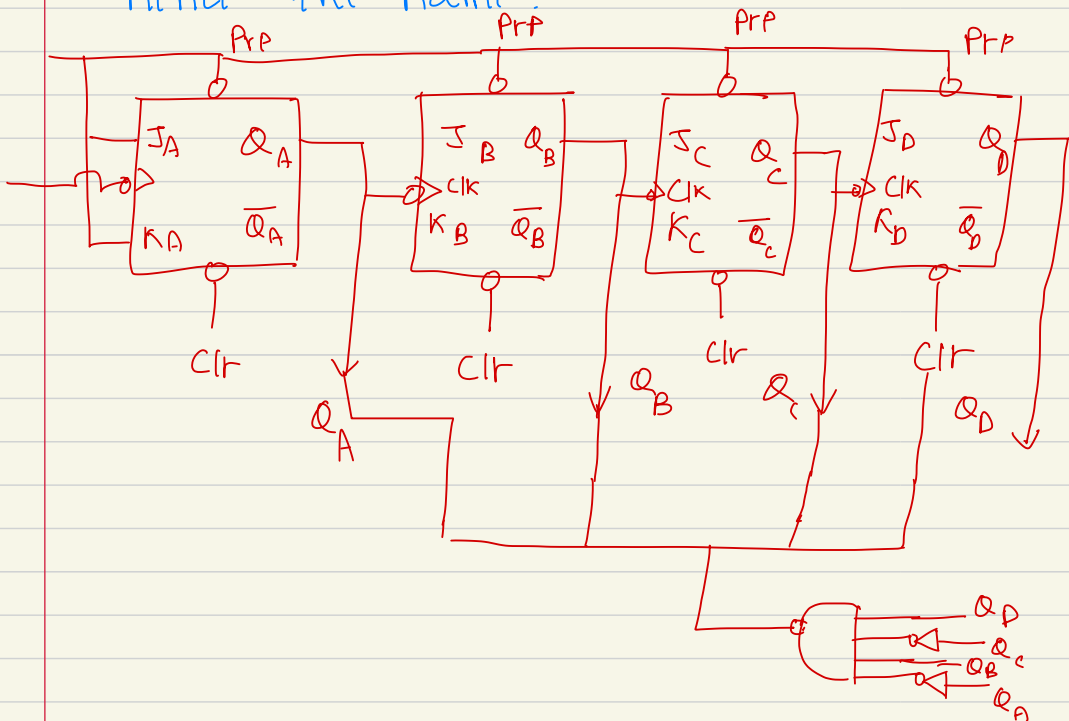
	Q_2	Q_1	Q_0	y
0	0	0	0	0
1	0	0	1	0
2	0	1	0	1
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	0
7	1	1	1	1

$$y = M\bar{Q} + \bar{M}Q$$



(d) Decade Counter

This counter counts from 0 to 9, hence the name.



CLK	Q_D	Q_C	Q_B	Q_A
Initial	0	0	0	0
1 st	0	0	0	1
2 nd	0	0	1	0
⋮	0	0	1	1
⋮	1	0	1	0
⋮	1	1	1	1
n th	1	1	1	1

← 'decade counter'

→ 10

The mini clk resets after a count of 10 ∴ it is a decade counter.

2 (a) Timers vs Counters

→ Timers : It basically evaluates the amount of time passed by & it is triggered based on clock S/N.

→ Counters : It helps to count the occurrences of inp events. Counter counts based on the external clocks.

2 (b)

* Asynchronous Counter :

The clock of the next F/F is based on the input from the prev stage. i.e., not all FFs are given the clocks synchronously.

* Synchronous Counter :

All the clocks are given the signal at the same time.