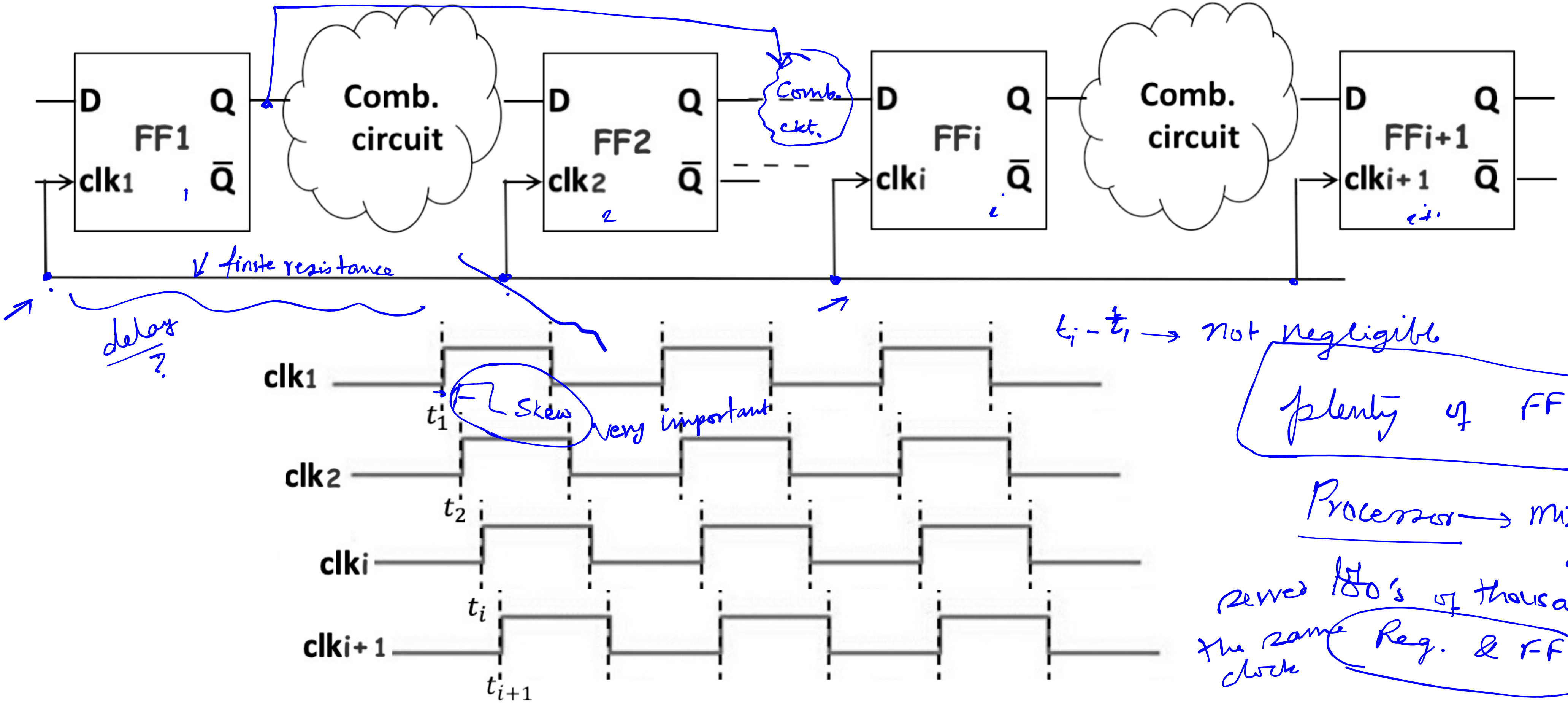
Clock Skew t_{skew} :



$t_i - t_1$ → not negligible

plenty of FF

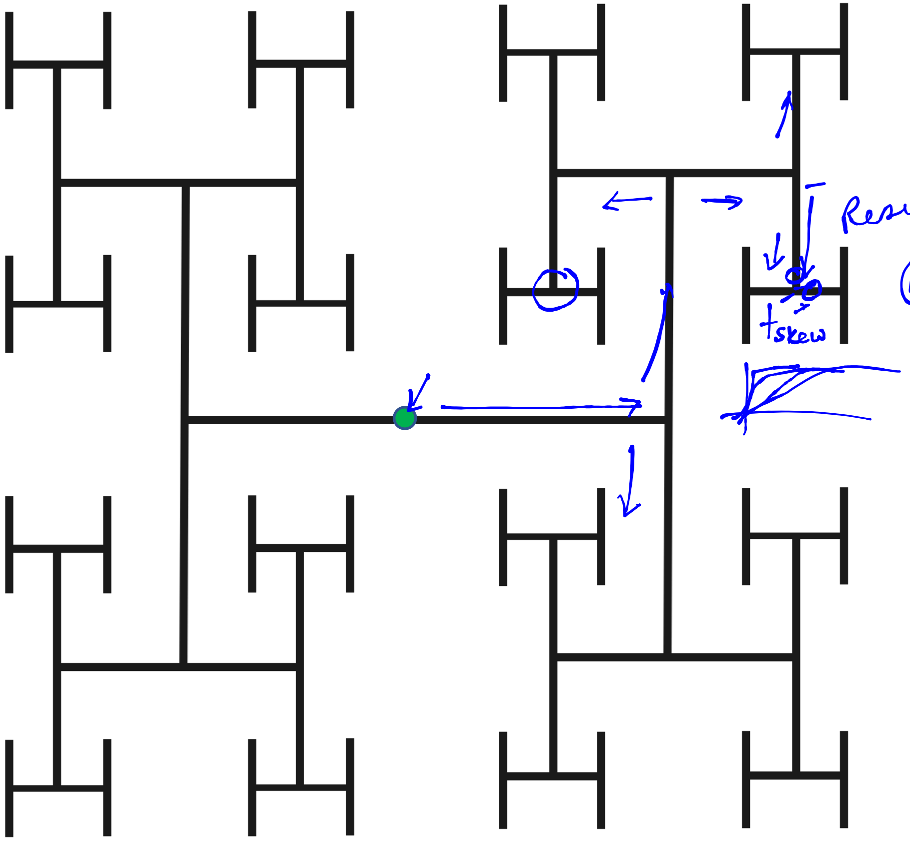
Processor → millions of devices

several 100's of thousands of Reg. & FF

the same clock

The clock to FF1 goes high at time t_1 whereas that for FF2 goes high at time t_2, we then say that there is a clock skew of $t_{skew} = t_2 - t_1$ between FF1 and FF2.

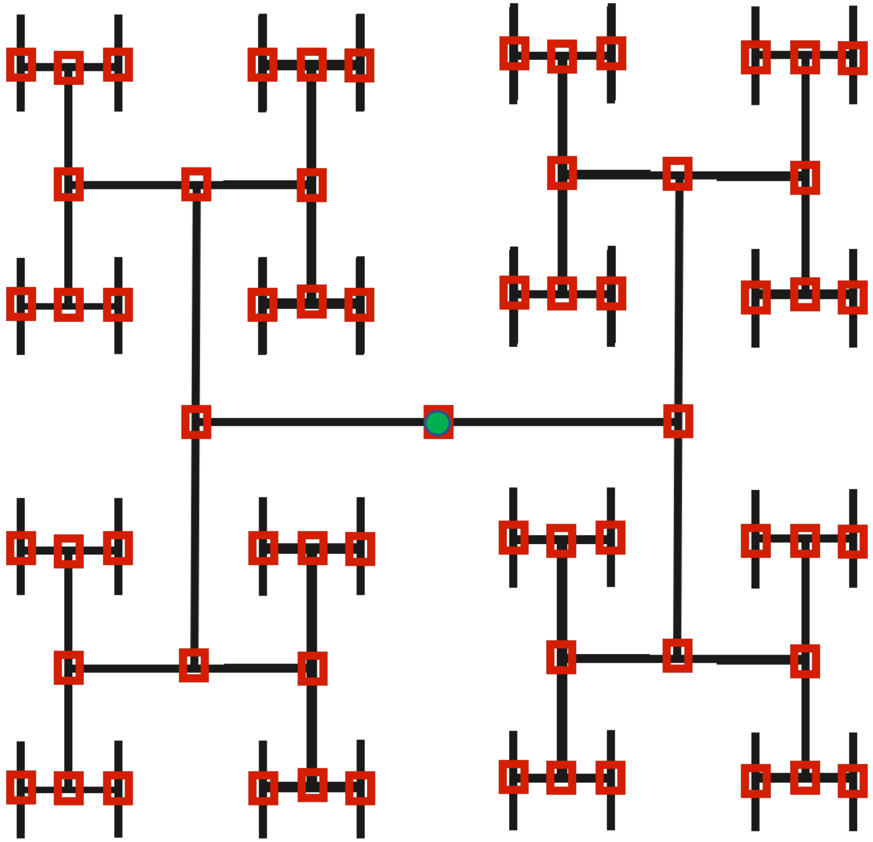
Clock Distribution Tree:

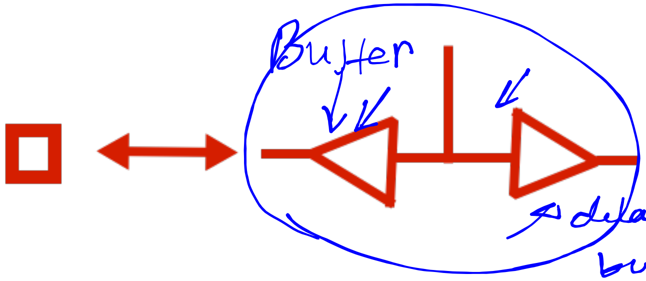


Resistance

① there will be drop across those resistance

② delay due to interconnect





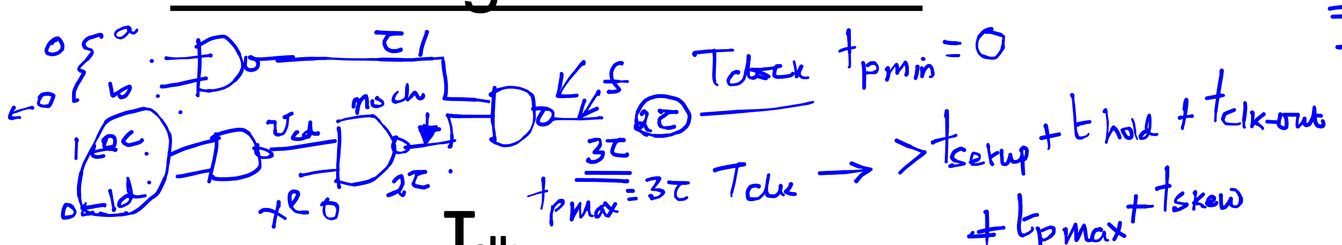
delay of a buffer

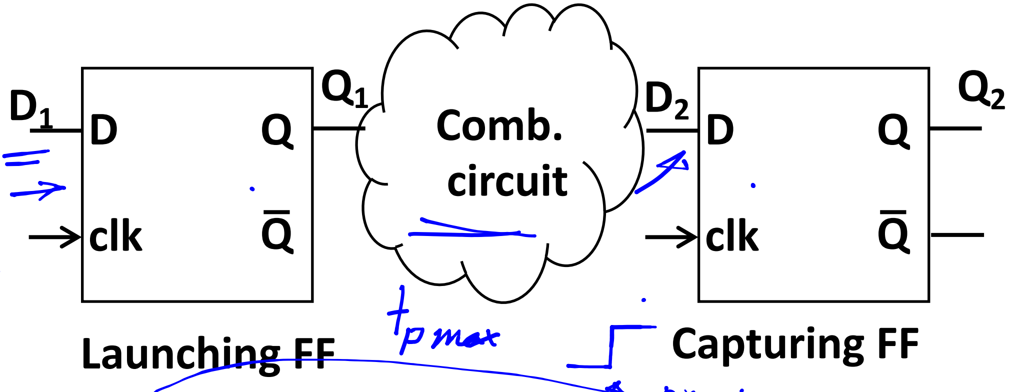
FF Timing Constraints:

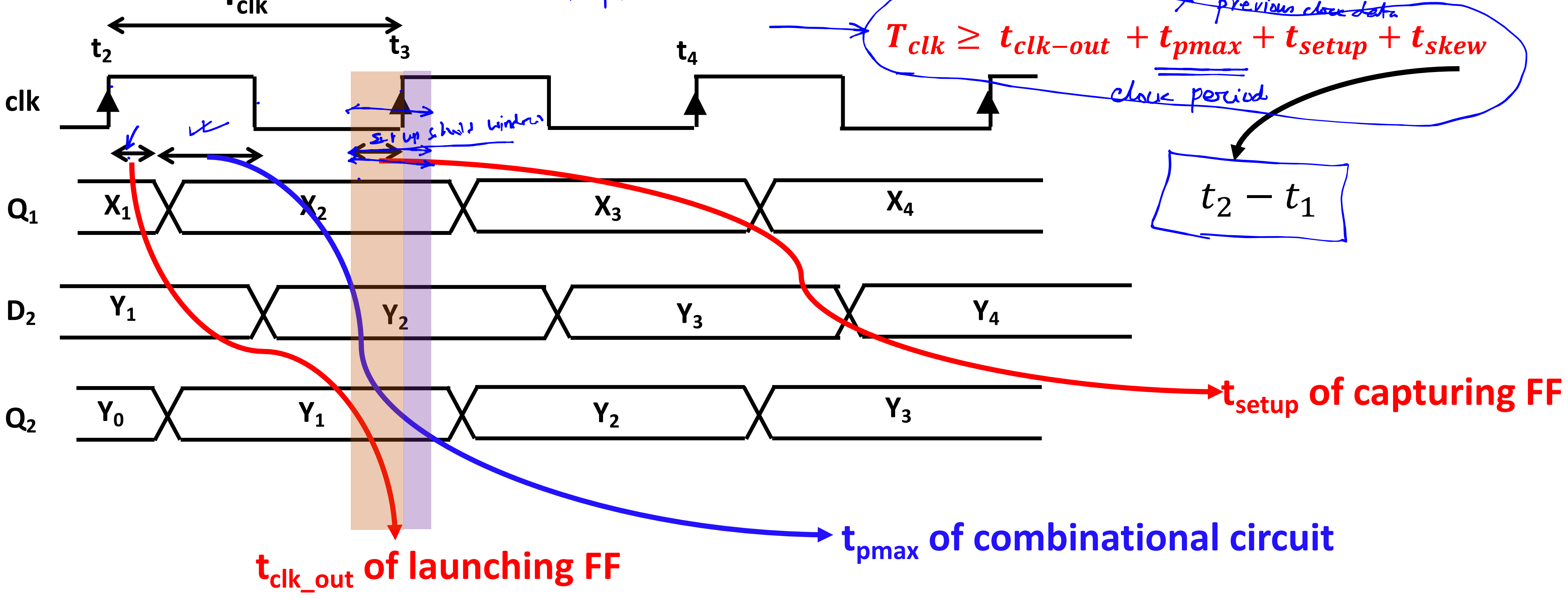
t_p min to t_{pmax}

Tclock $t_{pmin} = 0$

$t_{pmax} = 3\tau$ $T_{clk} \rightarrow\ > t_{setup} + t_{hold} + t_{clk-out} + t_{pmax} + t_{skew}$







$$T_{clk} \geq t_{clk-out} + \underline{t_{pmax}} + t_{setup} + t_{skew}$$

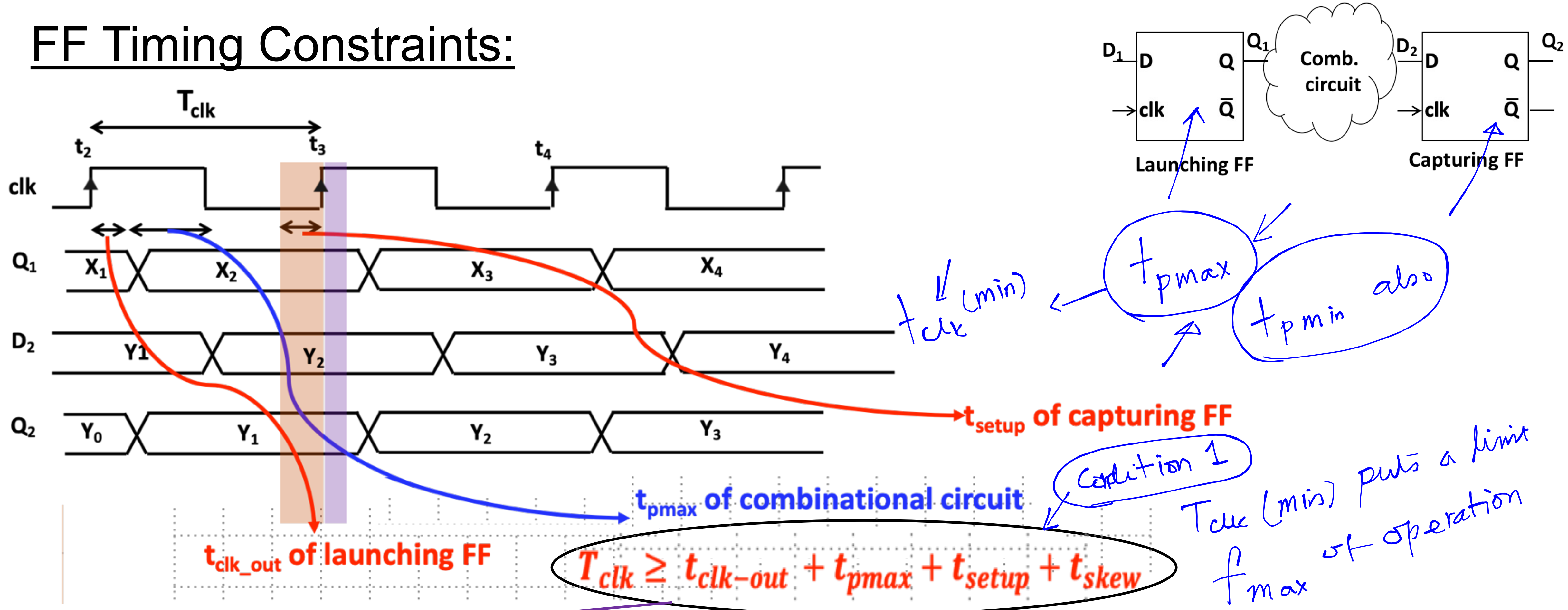
$$\boxed{t_2 - t_1}$$

t_{setup} of capturing FF

t_{pmax} of combinational circuit

t_{clk_out} of launching FF

FF Timing Constraints:



$$T_{clk} \ge t_{clk-out} + t_{pmax} + t_{setup} + t_{skew}$$

t_{clk} (min)

t_{pmax}

t_{pmin} also

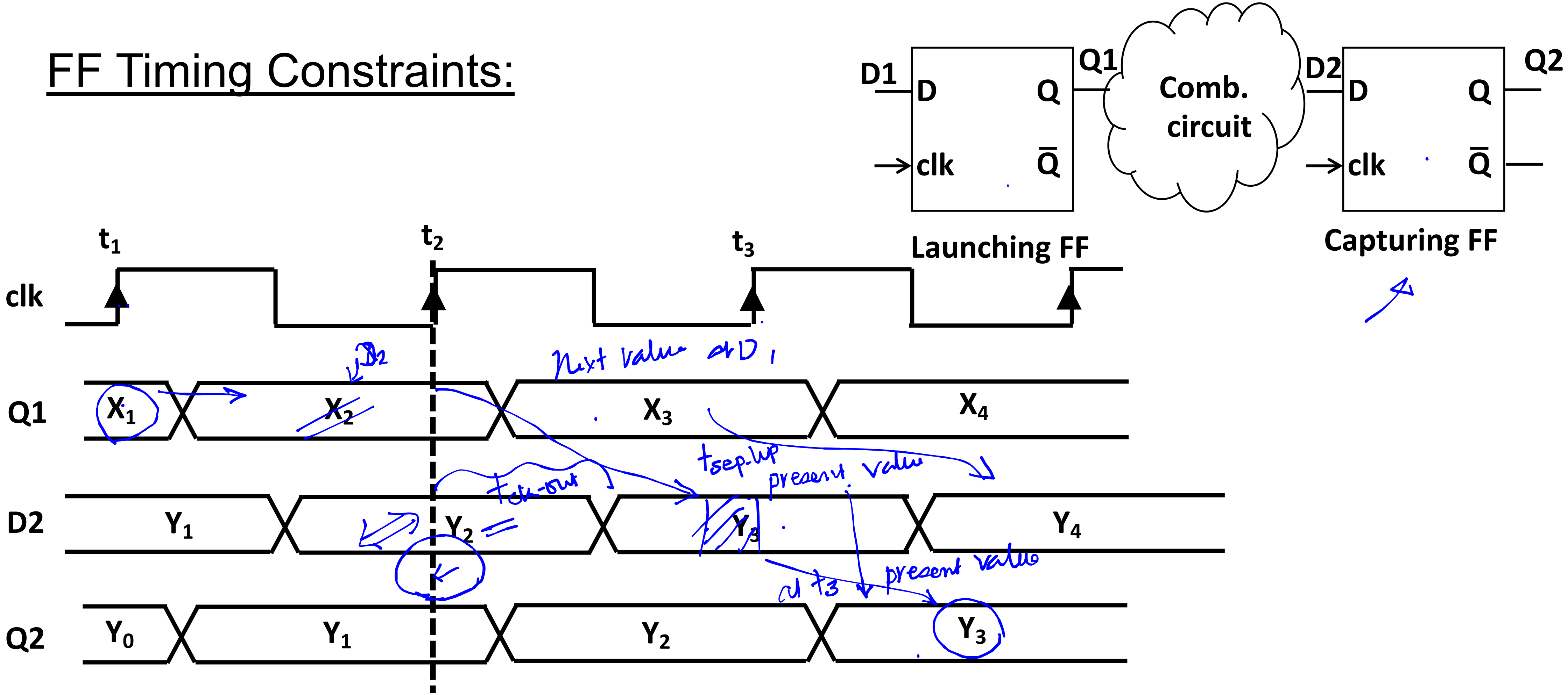
Condition 1

T_{clk} (min) puts a limit

f_{max} of operation

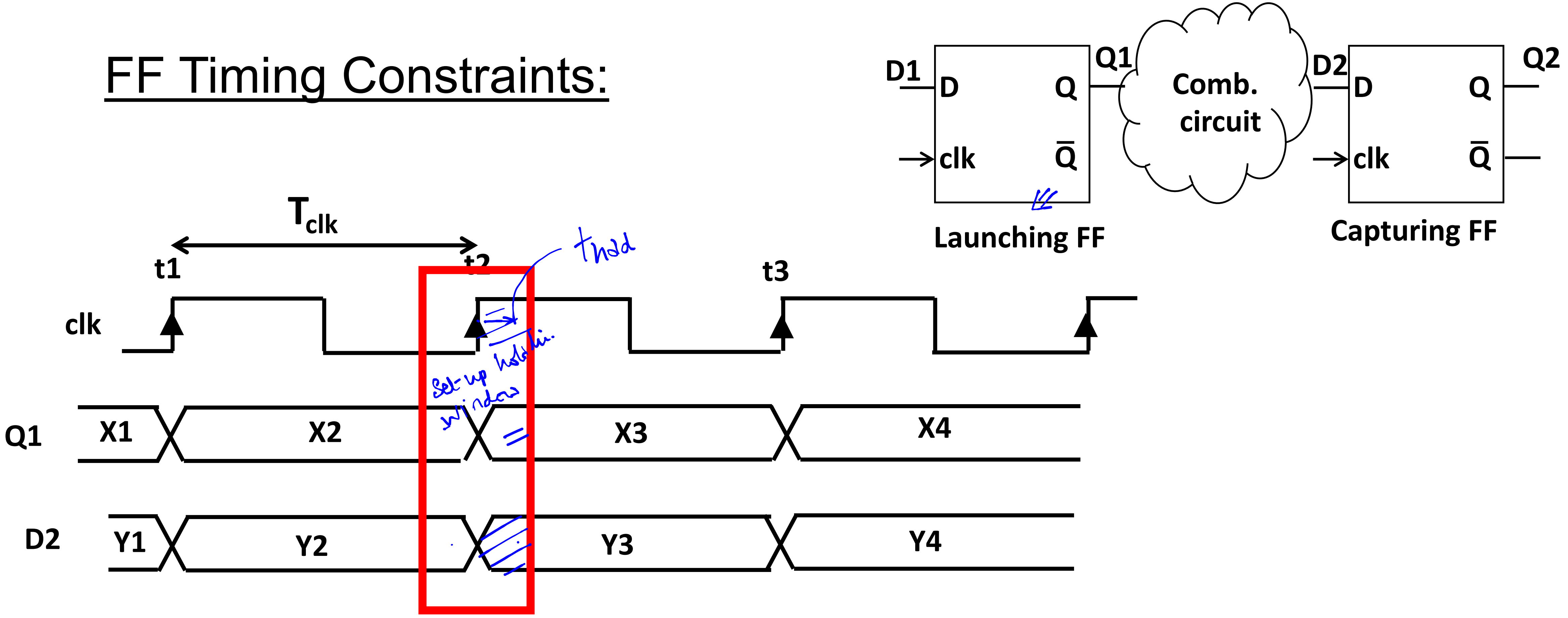
- This equation ensures that if the output of launching FF changes, this will make it to capturing FF before set-up time of the capturimg FF. else it won't be captured
- It also puts a limit on the minimum value of clock period and hence, maximum value of clock frequency hence the speed at which the circuit can operate.

FF Timing Constraints:



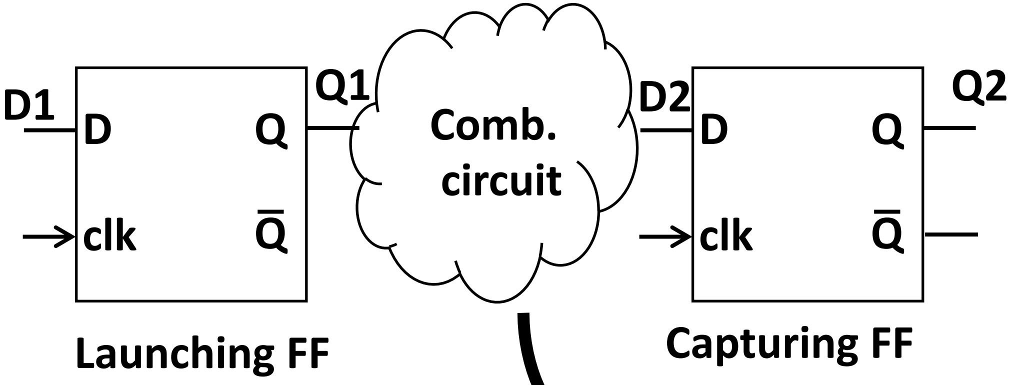
- At t_2, launching FF loads X_3 while capturing FF loads Y_2 which corresponds to input X_2.
- Will it be possible for capturing FF to load Y_3 which corresponds to X_3 at t_2?
- This may happen if Y_3 reaches early enough to capturing FF.

FF Timing Constraints:



- For instance, if $\overset{=0}{t_{clk\text{-}out}}$ and t_{pmax} are zero, then Y_2 may change to Y_3 during hold time of the capturing FF leading to unstable output.
- To avoid this: $\overset{\neq 0}{t_{clk-out}} + t_{pmin} \geq \overset{t_{hold}}{t_{hold}} + t_{skew}$ Condition 2

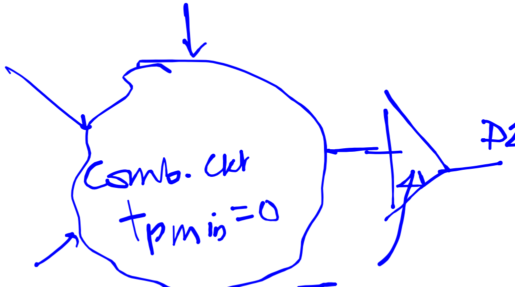
FF Timing Constraints:



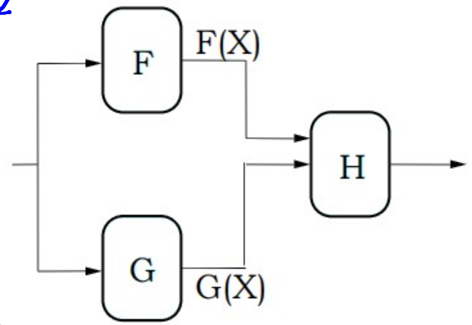
Maximum logic delay constraint

$$T_{clk} \geq t_{clk-out} + t_{pmax} + t_{setup} + t_{skew}$$

$t_{pmax} \leq T_{clk} - t_{clk-out} - t_{setup} - t_{skew}$



T_{burst}

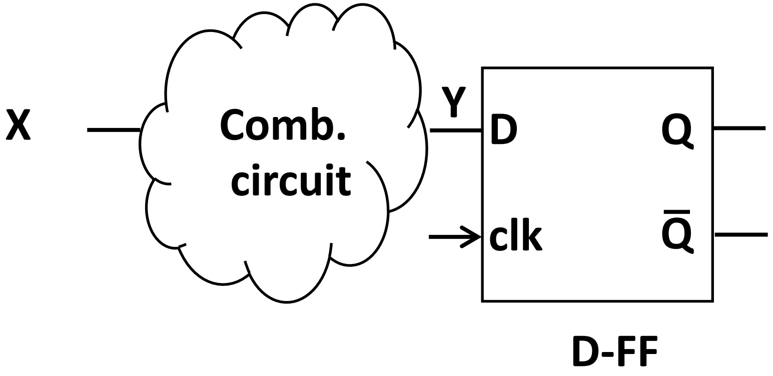


$$t_{clk-out} + t_{pmin} \geq t_{hold} + t_{skew}$$

$t_{pmin} \geq -t_{clk-out} + t_{hold} + t_{skew}$

Minimum logic delay constraint is maintained

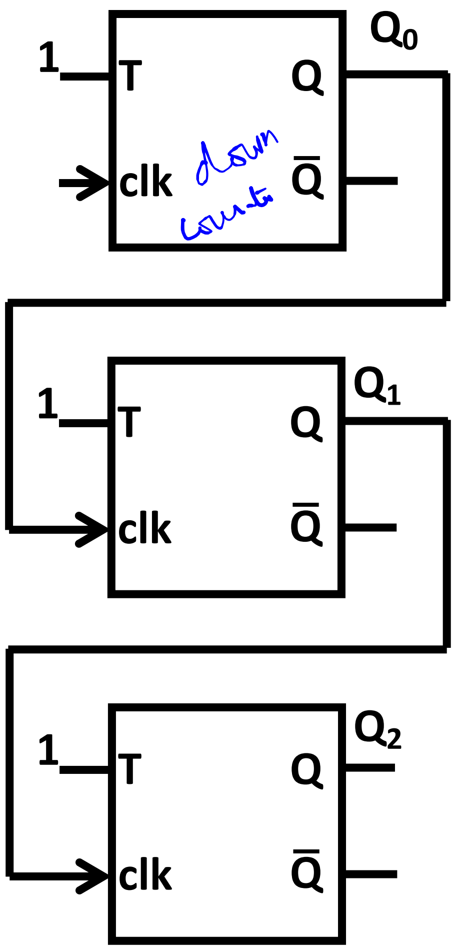
Homework:

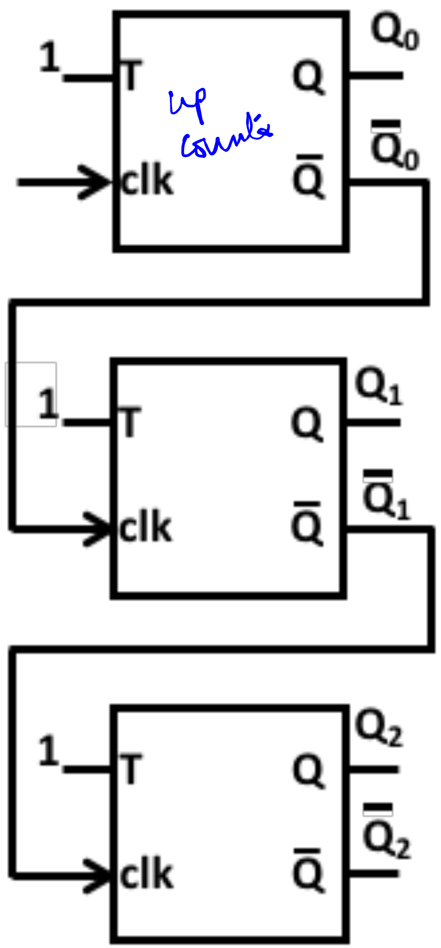


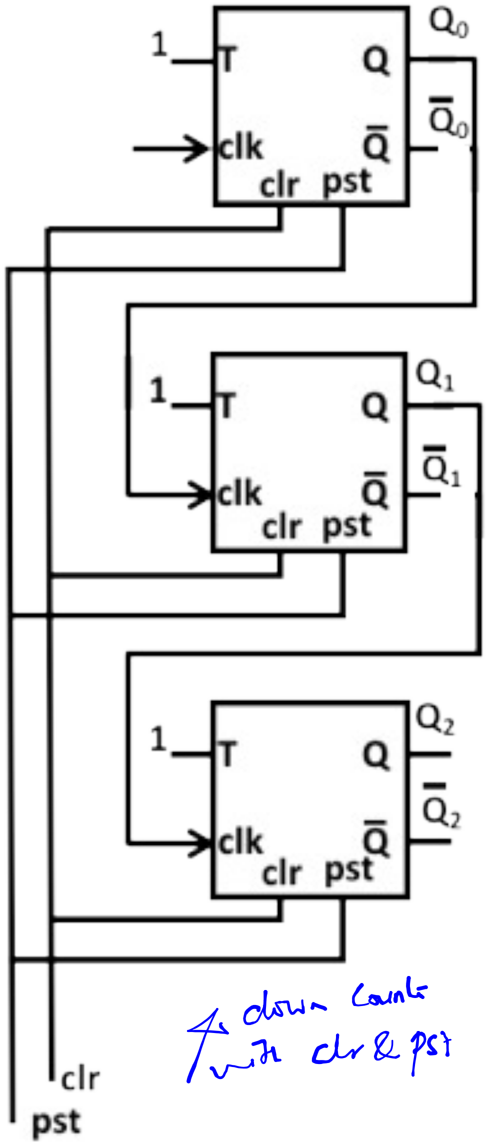
- Assuming the clock edge arrives at 10 ns, minimum and maximum propagation delay of the combinational circuit is 3ns and 1ns, setup time of FF is 2ns and hold time of the FF is 1ns, find out the time during which input x should be stable for stable output Q corresponding to current value of X?
- What if t_{skew} is 1ns?

Tomorrow's class

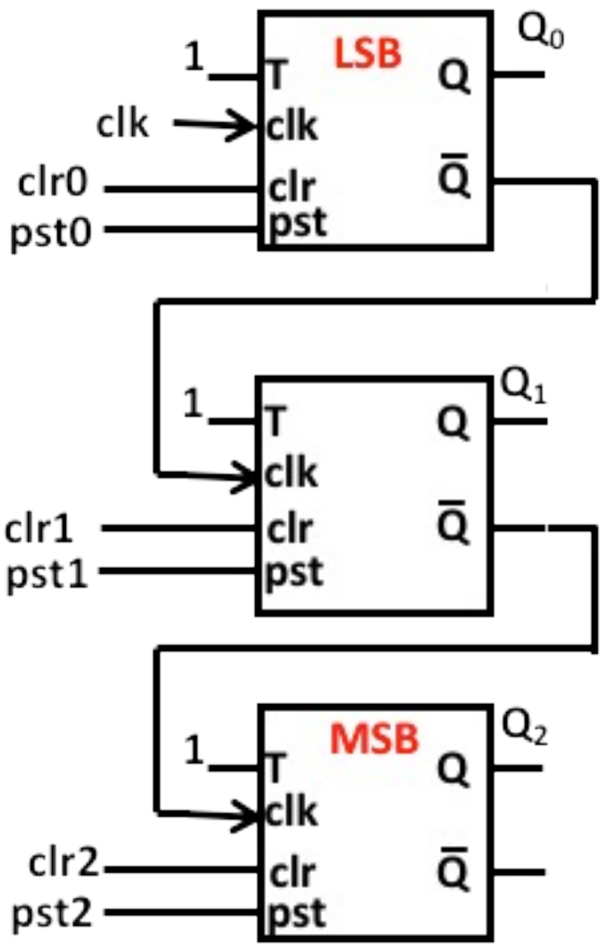
Ripple Counters:



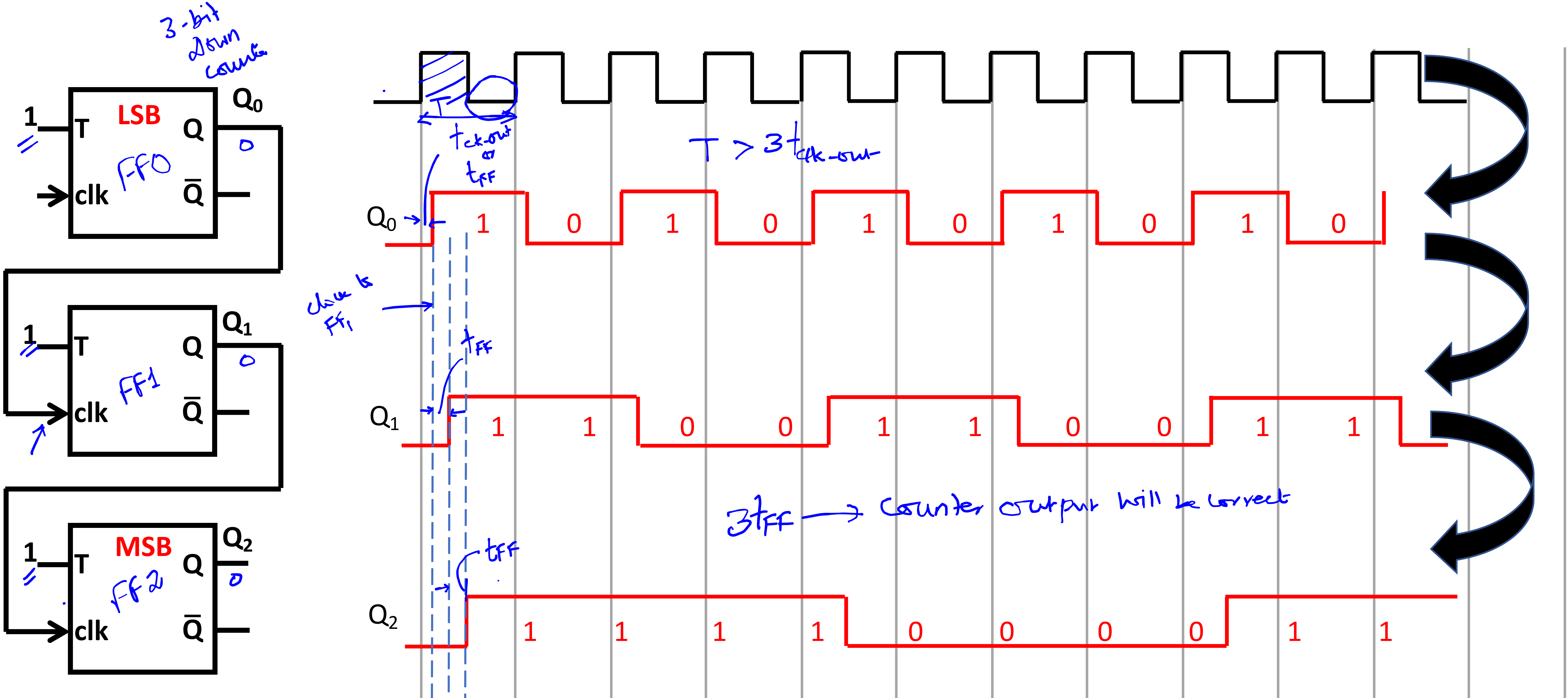




4 down counter with clr & pst

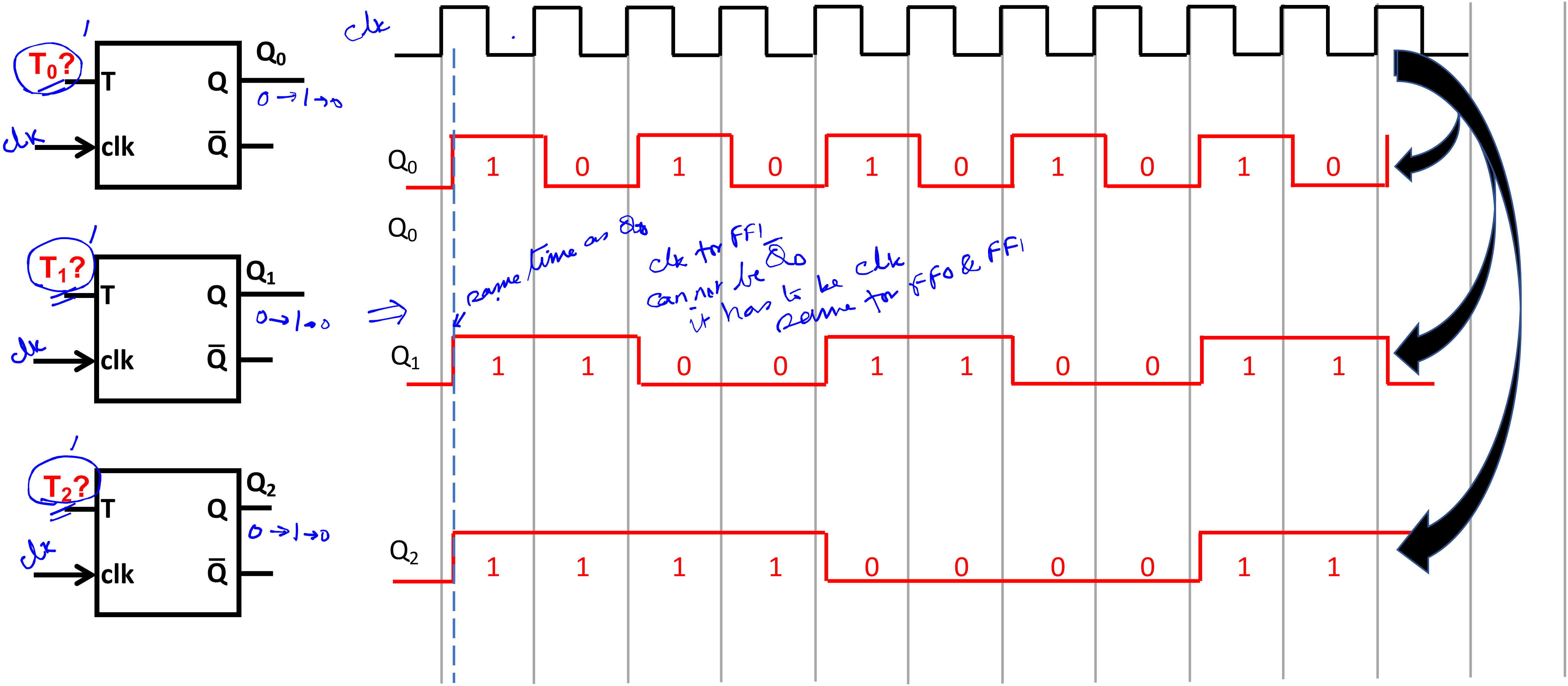


FF Delay ⟶ Ripple or Asynchronous Binary DOWN Counters:



$$T_{clock} \geq 2 \times 3 \times t_{clk-out}$$

Synchronous Binary DOWN Counter:



same time as Q₀

clk for FF1 can not be $\bar{Q}_0$ it has to be clk same for FF0 & FF1

We require to count: 111 – 110 – 101 – 100 – 011 – 010 – 001– 000 -- 111

State Transition Table:

PS -- Present State (Q_{n-1}); NS – Next State (Q_n)

State Transition

Excitation Table

Excitation

PS	NS	Q_2 PS	Q_1 PS	Q_0 PS	Q_2 NS	Q_1 NS	Q_0 NS	T_2	T_1	T_0
111	110	1	1	1	1	1	0	0	0	1
110	101	1	1	0	1	0	1	0	1	1
101	100	1	0	1	1	0	0	0	0	1
100	011	1	0	0	0	1	1	1	1	1
011	010	0	1	1	0	1	0	0	0	1
010	001	0	1	0	0	0	1	0	1	1
001	000	0	0	1	0	0	0	0	0	1
000	111	0	0	0	1	1	1	1	1	1

Transition Table

included

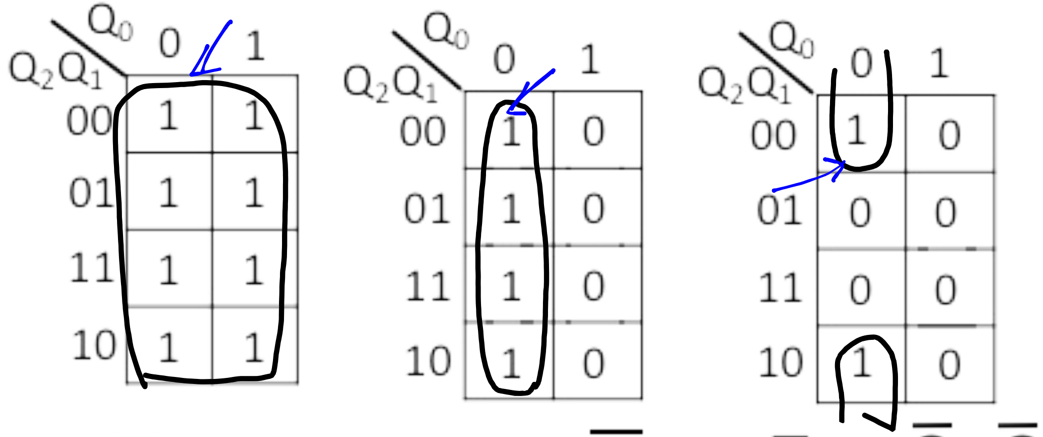
		Q_2			Q_1			Q_0		
PS	NS	PS	NS	T_2	PS	NS	T_1	PS	NS	T_0
111	110	1	1	0	1	1	0	1	0	1
110	101	1	1	0	1	0	1	0	1	1
101	100	1	1	0	0	0	0	1	0	1
100	011	1	0	1	0	1	1	0	1	1
011	010	0	0	0	1	1	0	1	0	1
010	001	0	0	0	1	0	1	0	1	1
001	000	0	0	0	0	0	0	1	0	1
000	111	0	1	1	0	1	1	0	1	1

We require to count: 111 – 110 – 101 – 100 – 011 – 010 – 001– 000 -- 111

<u>State Transition Table</u>: PS -- Present State (Q_{n-1}); NS – Next State (Q_n)

Kernaugh Map Q_2, Q_1, Q_0 [illegible] T_2, T_1, T_0

PS	NS	Q_2 PS	Q_1 PS	Q_0 PS	Q_2 NS	Q_1 NS	Q_0 NS	T_2	T_1	T_0
111	110	1	1	1	1	1	0	0	0	1
110	101	1	1	0	1	0	1	0	1	1
101	100	1	0	1	1	0	0	0	0	1
100	011	1	0	0	0	1	1	1	1	1
011	010	0	1	1	0	1	0	0	0	1
010	001	0	1	0	0	0	1	0	1	1
001	000	0	0	1	0	0	0	0	0	1
000	111	0	0	0	1	1	1	1	1	1



Q_2Q_1 \ Q_0	0	1
00	1	1
01	1	1
11	1	1
10	1	1

$T_0 = 1$

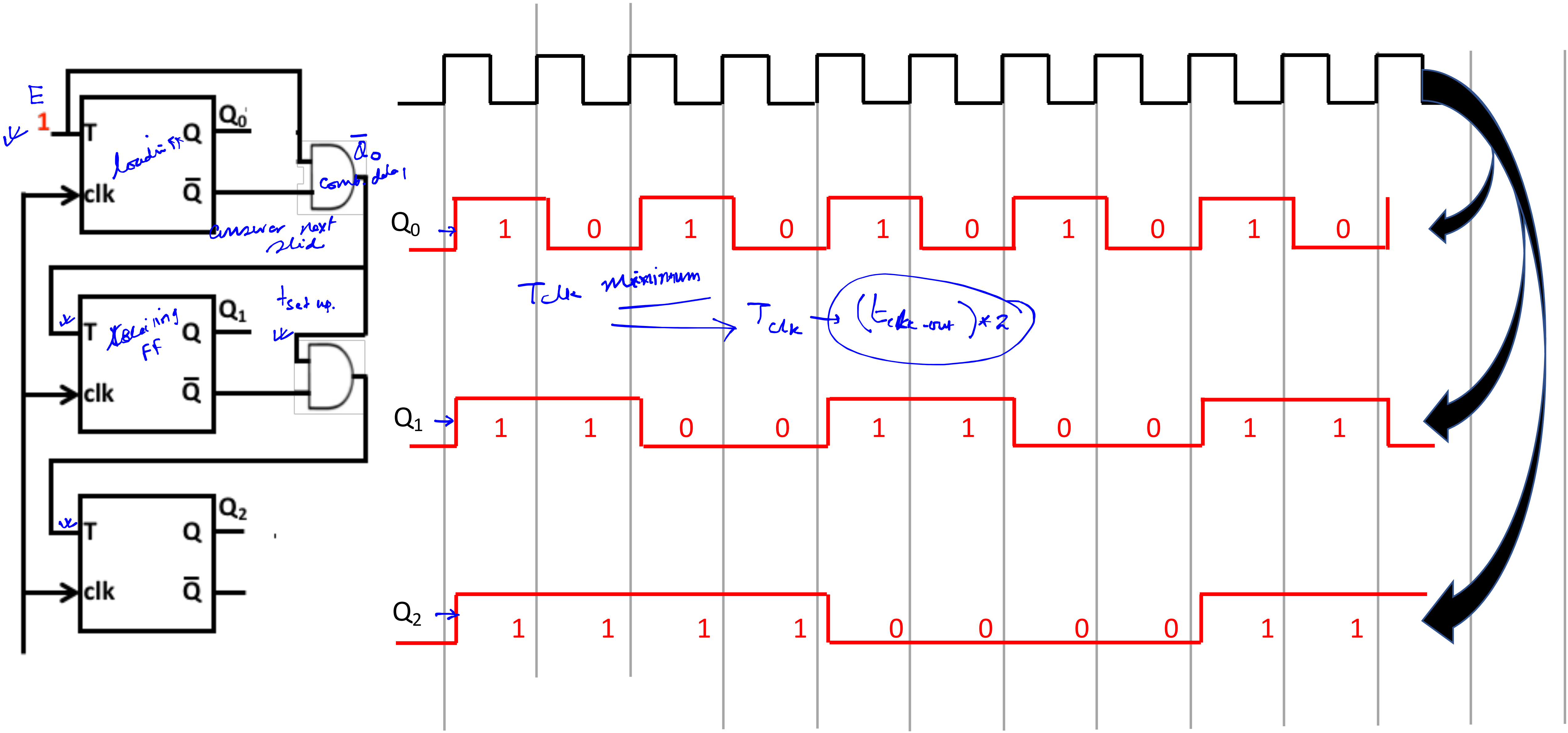
Q_2Q_1 \ Q_0	0	1
00	1	0
01	1	0
11	1	0
10	1	0

$T_1 = \overline{Q}_0$

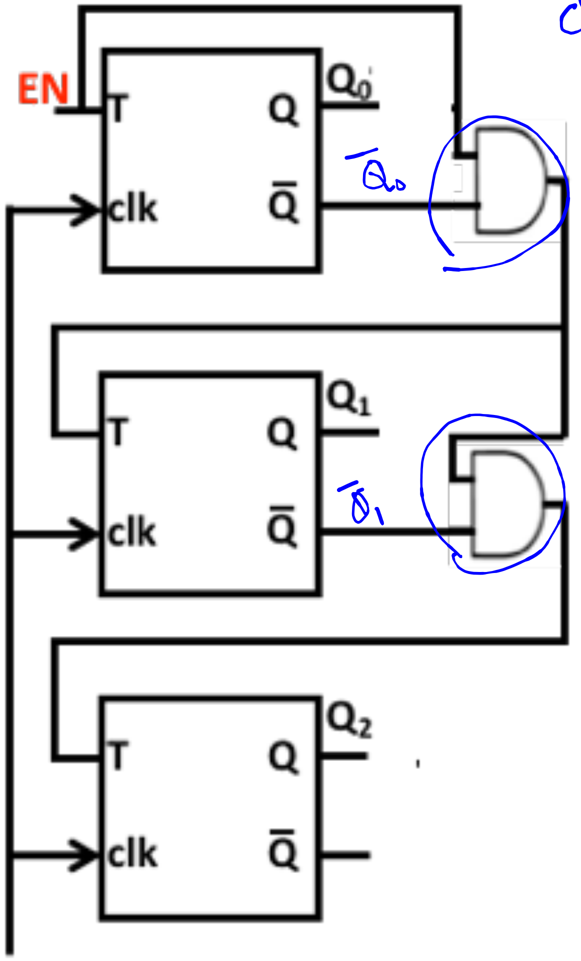
Q_2Q_1 \ Q_0	0	1
00	1	0
01	0	0
11	0	0
10	1	0

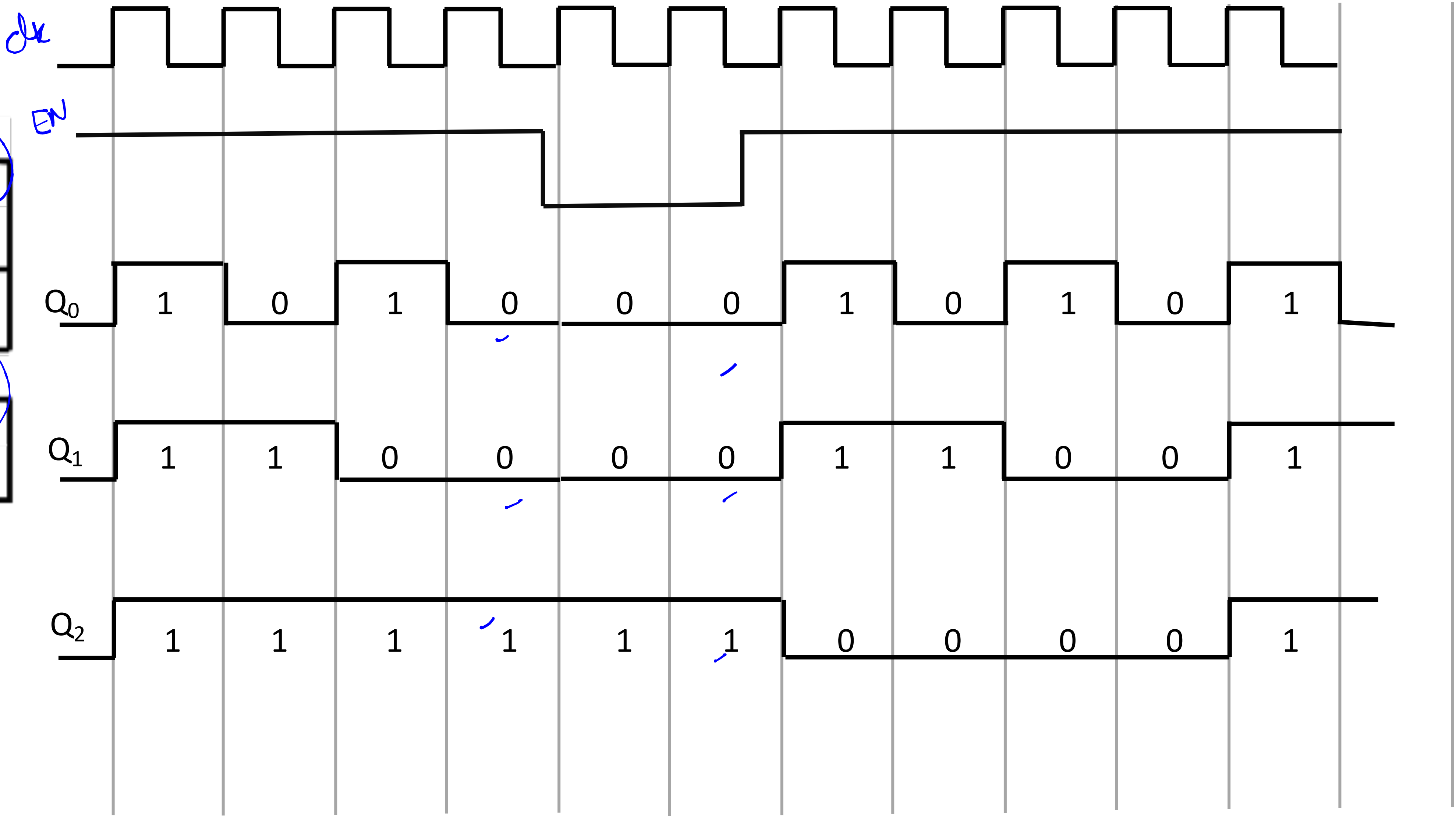
$T_2 = \overline{Q}_0\, \overline{Q}_1$

Synchronous Binary DOWN Counter:



Synchronous Binary DOWN Counter with Memory:





HW

- Design 3-bit synchronous binary up counter
- Design a 3-bit synchronous up/down counter

by choice

M

add or subtract

adder

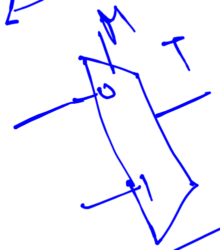
Thursday's class

0 2 4 3 5

Up counter 0 to 6

Down counter 5 to 0

T_i up down ← M



Up → Mod 7

Down → Mod 5