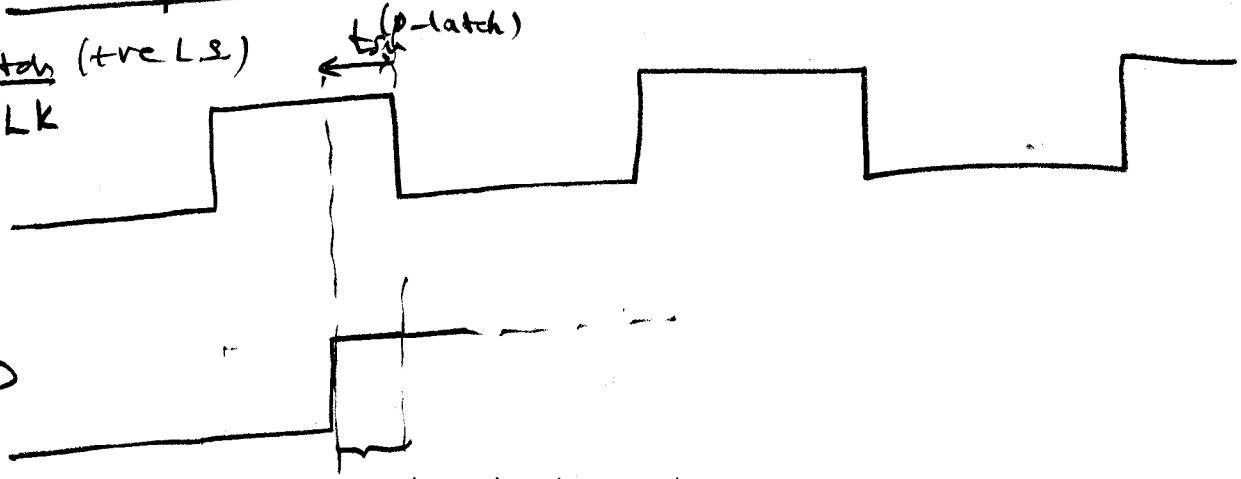
Lecture: Timing

Basic Concepts:

1 Maximum propagation delay

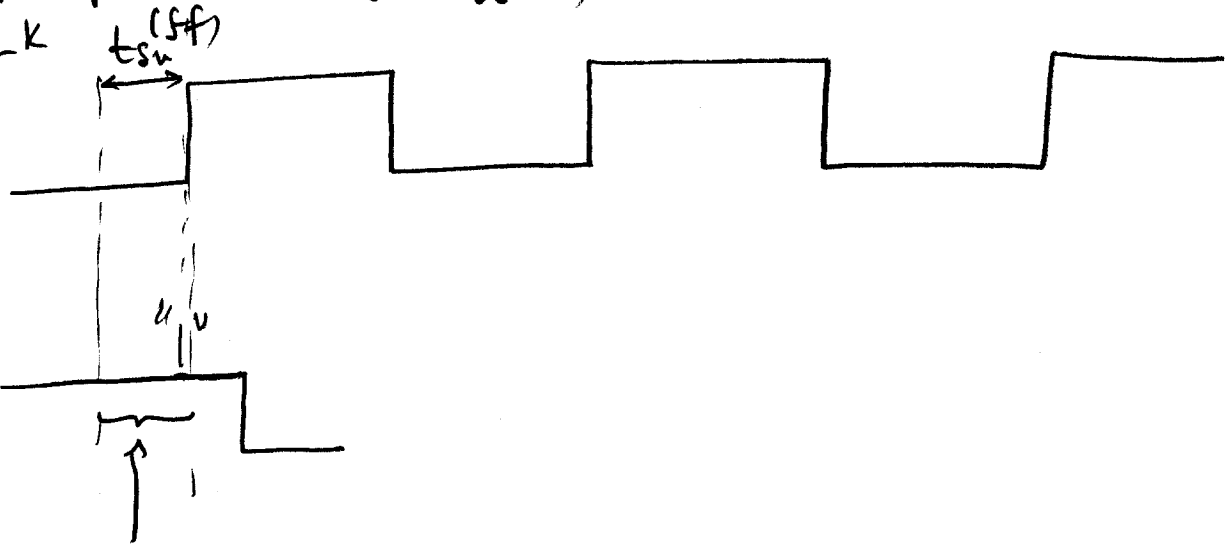
2 Set-up time:

(A) Latch (+ve LS)



D has to be stable at least t_{su} before the falling edge of the clock

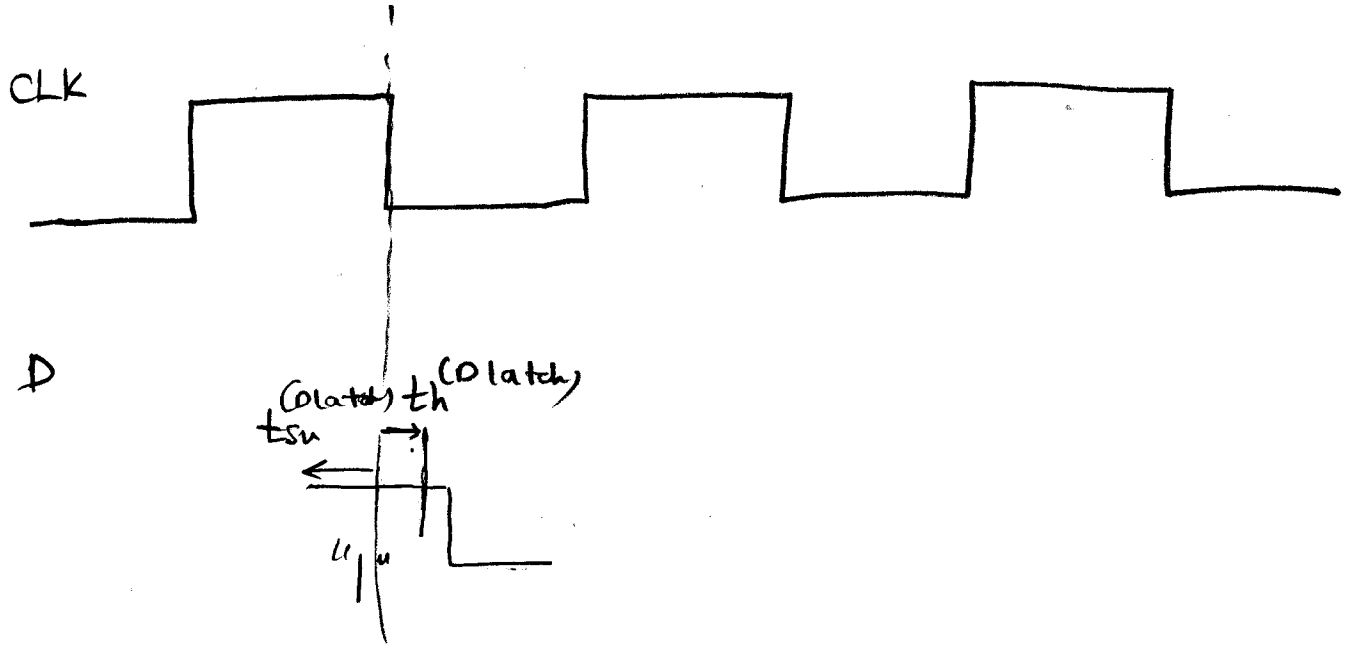
(B) Flip-flop: (+ve edge trigger)



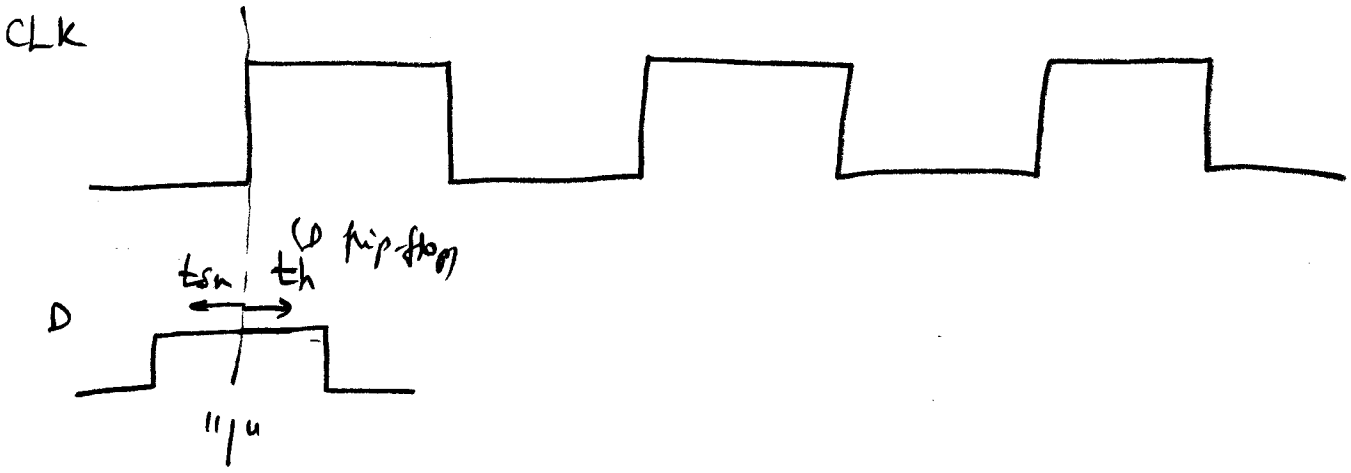
D has to be stable at least this long before the active rising edge of Clk

3 Hold time = minimum duration for which the data input D has to remain stable after the clock edge (that matters)

Ⓐ Positive Level-sensitive D latch:



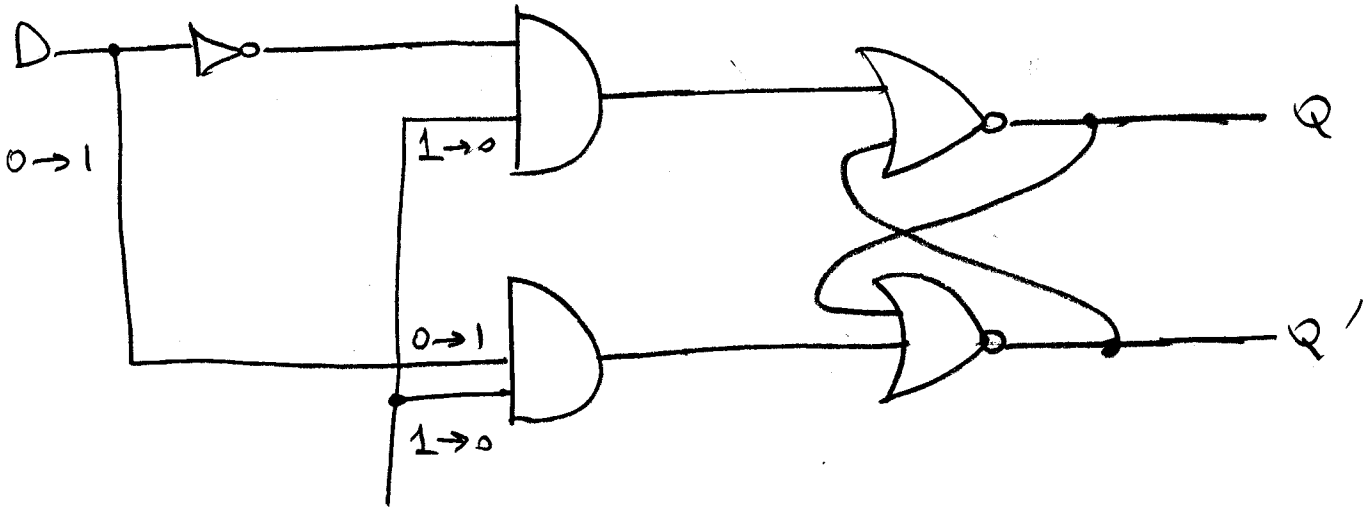
Ⓑ D flip-flop (positive edge-triggered)

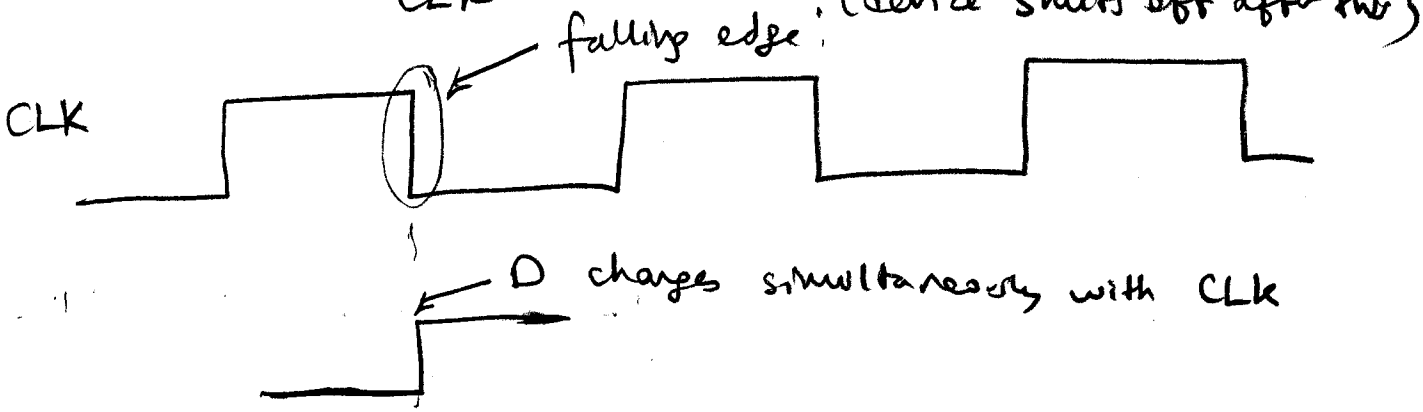


Hold time:

- Why does it arise?

Example 1

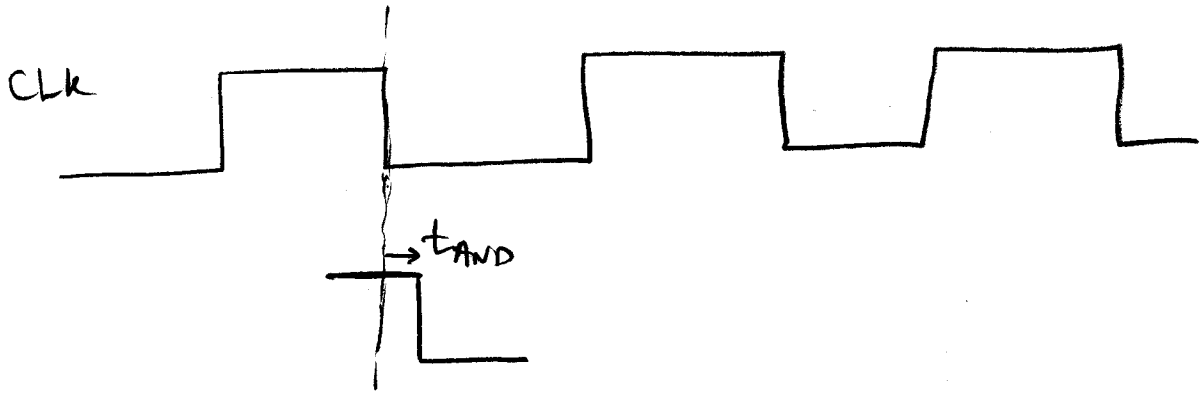




If D: 0→1 and CLK: 1→0, then it is possible that the output of the lower AND gate is 1 for a moment (at clock edge) which would lead to

$\begin{bmatrix} R \\ S \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ forbidden → $\begin{bmatrix} Q \\ Q' \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$.

If D is held stable for at least t_{AND} ns, this can be prevented.

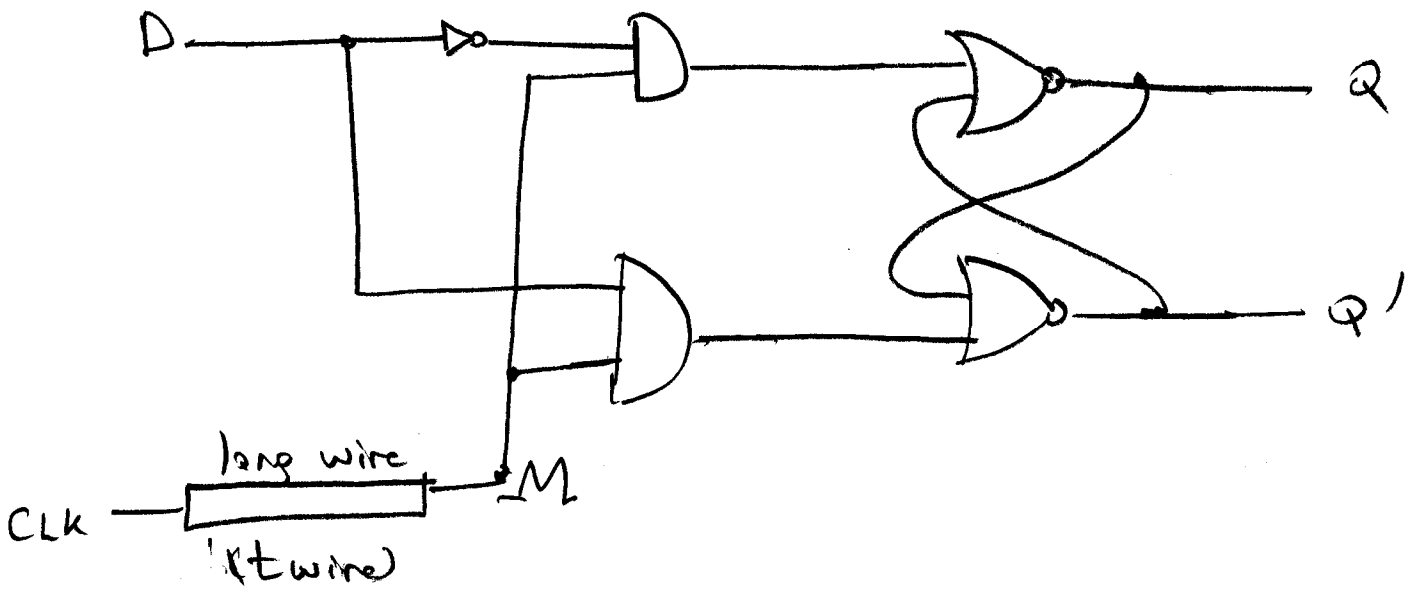


$\therefore$ $\boxed{t_h = t_{AND}}$ for D-latch.

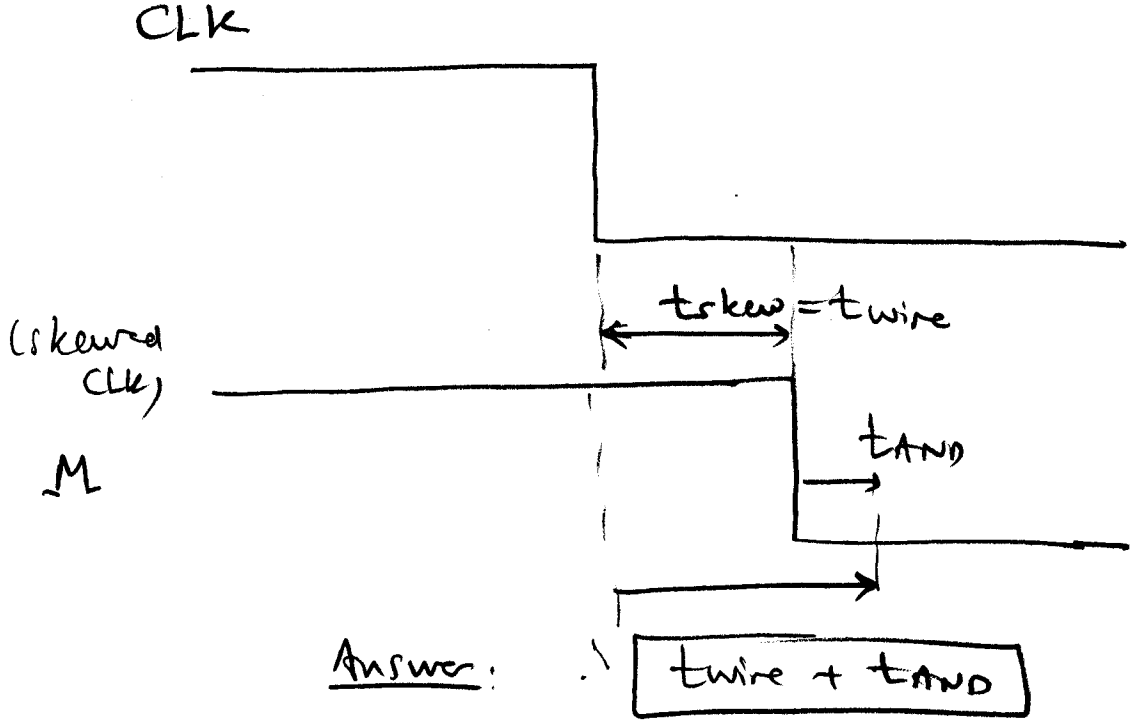
(Note: A conservative estimate, lower values of t_h than t_{AND} ns might be possible if we knew the implementation of the AND gate at transistor level.)

Example 2 : D latch with CLOCK SKEW.

Clock skew: the clock arrives late at the D latch.



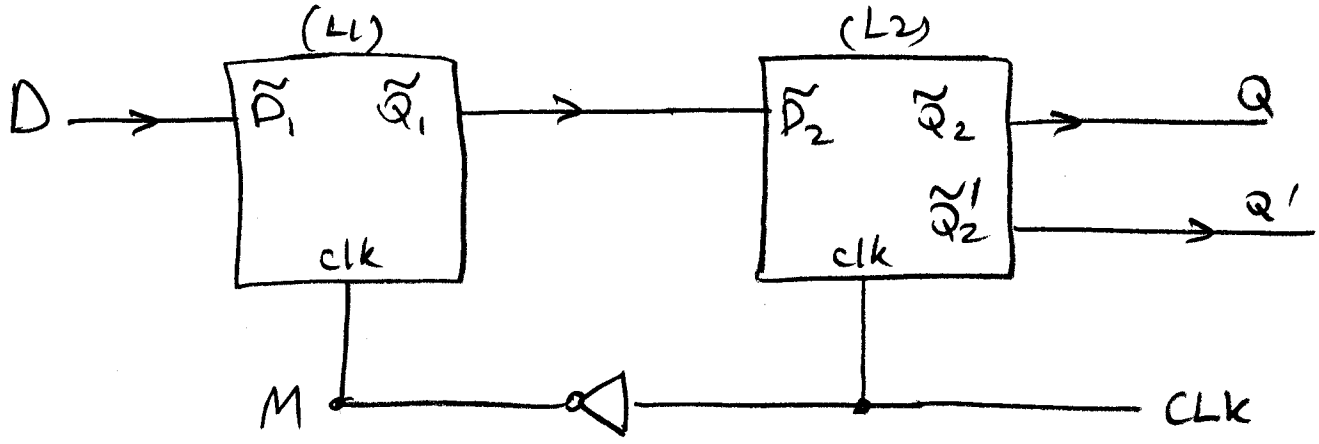
– What is the minimum duration for which D has to be stable after the falling edge of the CLK?



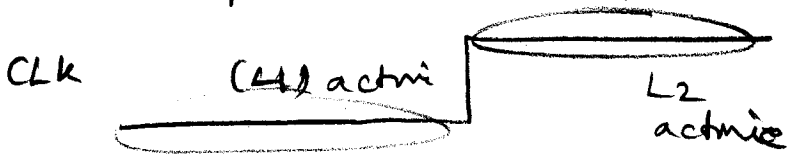
(this assumes that device satisfies a tsu wrt. Global clock

Answer: twire + tAND

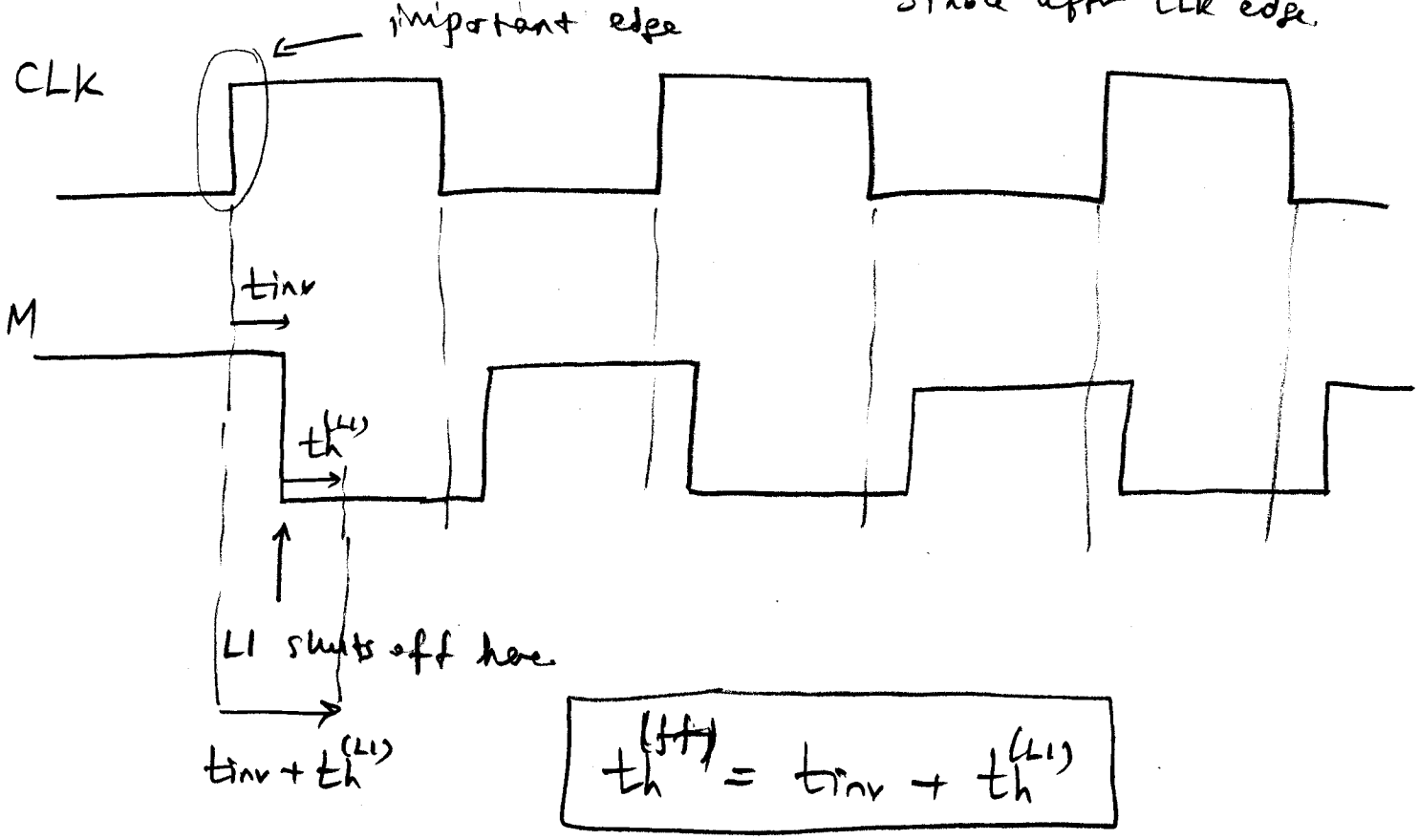
Example 3: Master-slave D flip-flop:



- This is a positive edge-triggered ff.

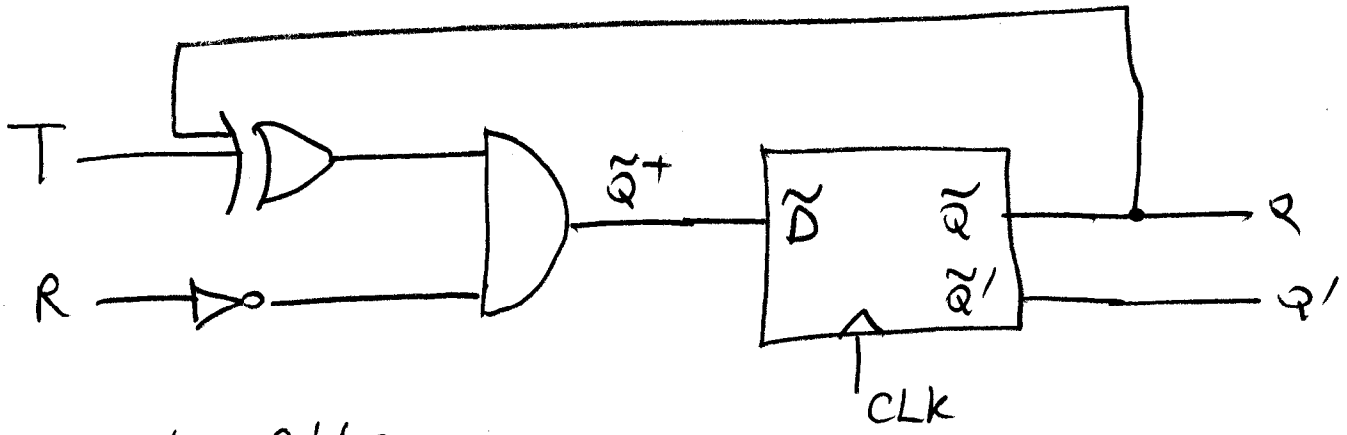


- Hold time ≡ minimum duration for which D has to be stable after CLK edge.



$$t_h^{(ff)} = t_{inv} + t_h^{(L1)}$$

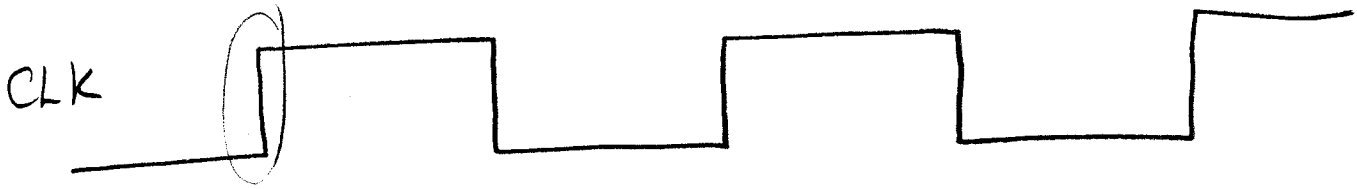
Example 4: T flip-flop



$$Q^+ = R'(Q \oplus T)$$

Evaluate:

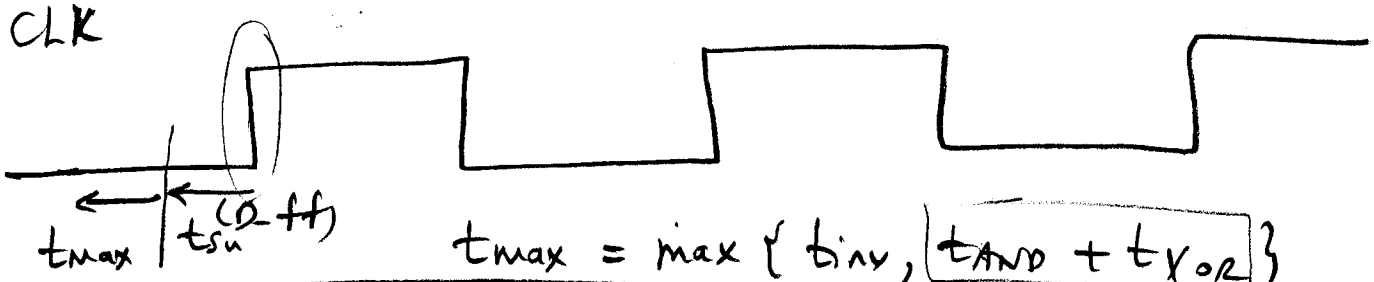
(a) CLK-to-Q delay (of T flip-flop):



$t_{CLK-Q}^{(T\text{-}ff)}$ = max. prop. delay from active edge of CLK to stable $\{Q, Q'\}$ (assuming T, R constant.)

$= t_{CLK-Q}^{(D\text{-}ff)}$ (since $\tilde{Q}^+$ will be stable if T, R stable)

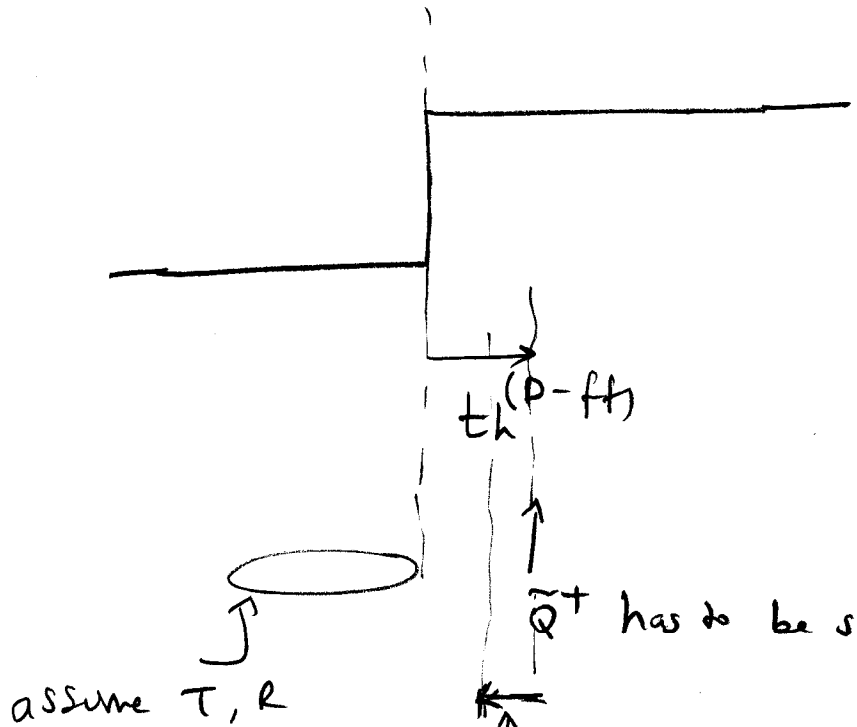
(b) $t_{su}^{(T\text{-}ff)}$ = minimum duration for which T and R have to be stable before CLK edge



$t_{max} = \max\{t_{INV}, t_{AND} + t_{YOR}\}$

$t_{su}^{(T\text{-}ff)} = t_{su}^{(D\text{-}ff)} + t_{max} - t_{min}$

(c) Hold time: = minimum duration for which T and R have to be stable after the CLK edge

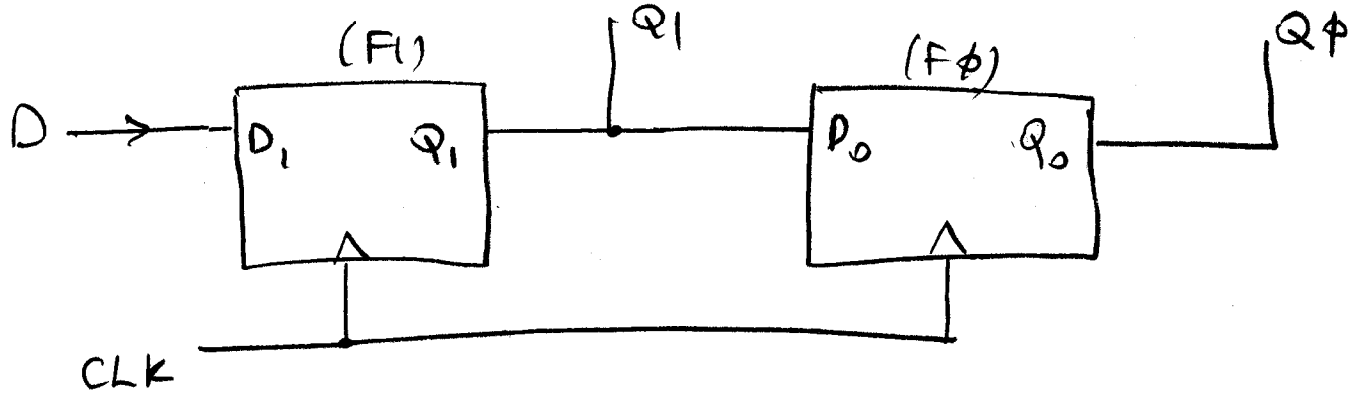


$$\boxed{t_h^{(T-ff)}} = t_h^{(D-ff)} - \min\{t_{AND} + t_{inv}, t_{AND} + t_{XOR}\}$$

$$= \underline{\boxed{t_h^{(D-ff)} - t_{AND} - \min\{t_{inv}, t_{XOR}\}}}$$

∴ Combinational logic in next state computation buys you some hold time

Example 5: 2-bit Shift Register



(a) CLK-to-Q delay of 2-bit SR?

$t_{CLK-Q}^{(SR)}$ ≡ maximum propagation delay from active edge of CLK to stable output vector (assuming D is stable.)

= [Q1 Q0] here.

$$= \max\{ \underbrace{t_{CLK-Q}^{(F1)}}_{\text{(for Q1)}}, \underbrace{t_{CLK-Q}^{(F0)}}_{\text{(for Q0)}} \}$$

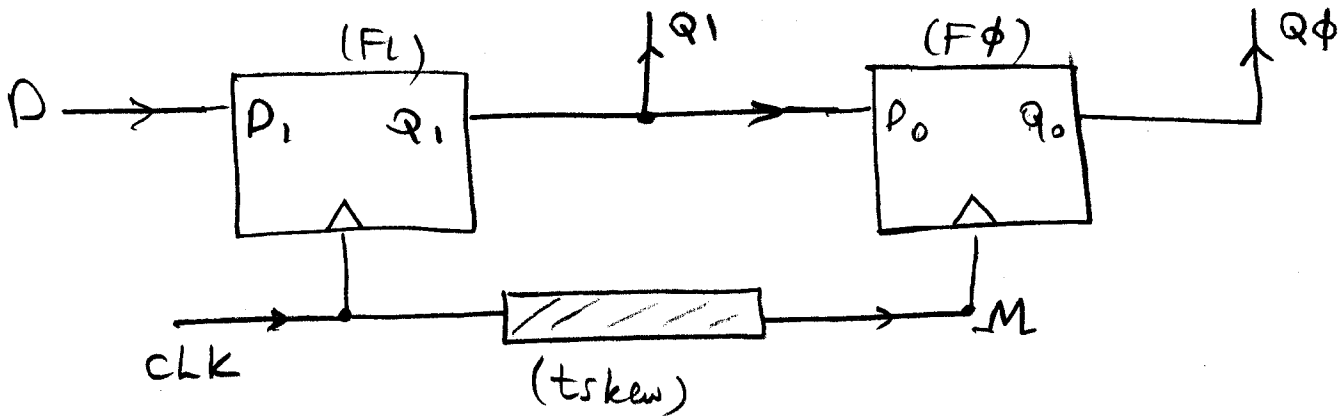
(b) Setup time of SR ≡ minimum duration for which the D input has to be stable before the CLK edge

$$t_{su}^{(SR)} = t_{su}^{(F1)}.$$

(c) Hold time of SR ≡ min. duration for which D has to remain stable after the CLK edge

$$t_h^{(SR)} = t_h^{(F1)}$$

Example 6: Shift Register with Clock skew:

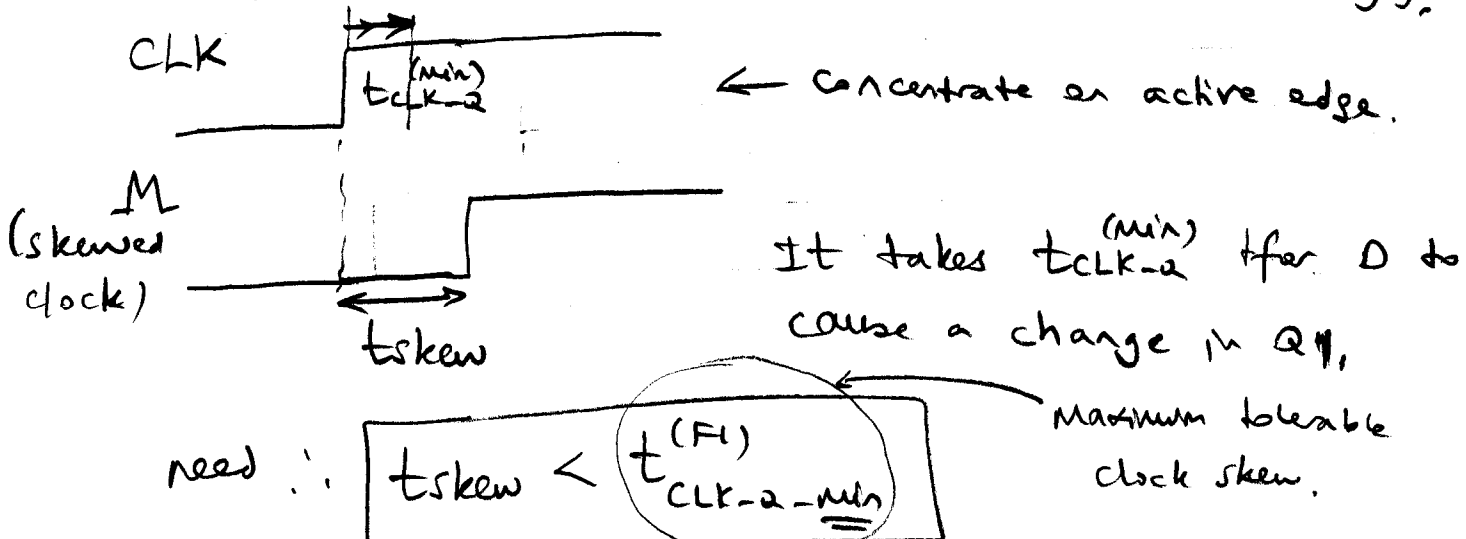


(a) What may go wrong in this situation due to clock skew?

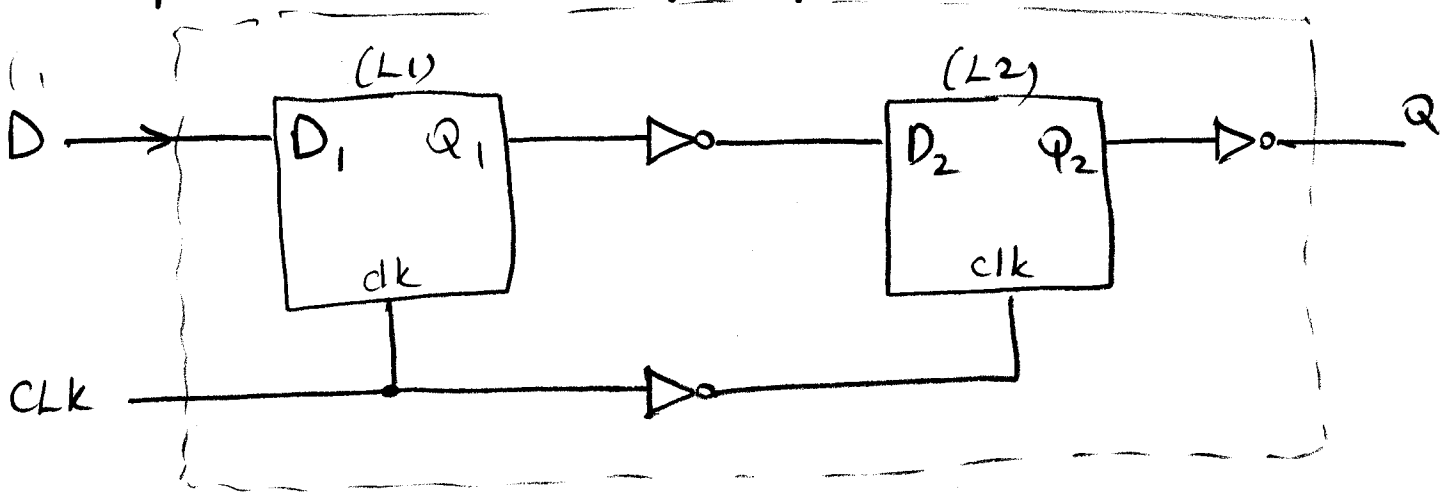
- Visualize the active edge of CLK arriving at F1. What happens at that instant?
- Now, visualize "same" CLK edge arriving at Fϕ, a moment (tskew ns) later. What happens?

– Value of D shoots through to Qϕ.

(b) What is the <u>Maximum clock skew</u> that this shift register can <u>tolerate</u> (i.e still work correctly)?

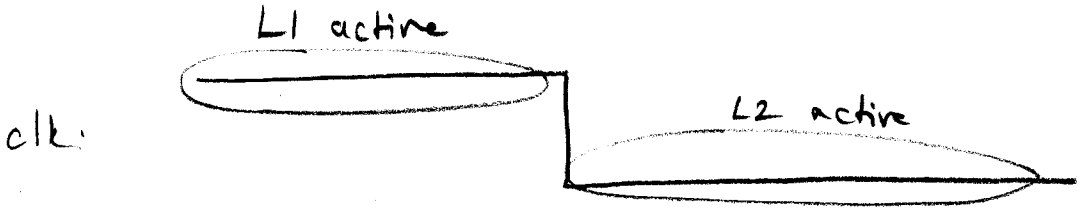


Example 7: A new flip-flop:



(a) What does this device do?

1 Concentrate on the clock first; when is each latch active?



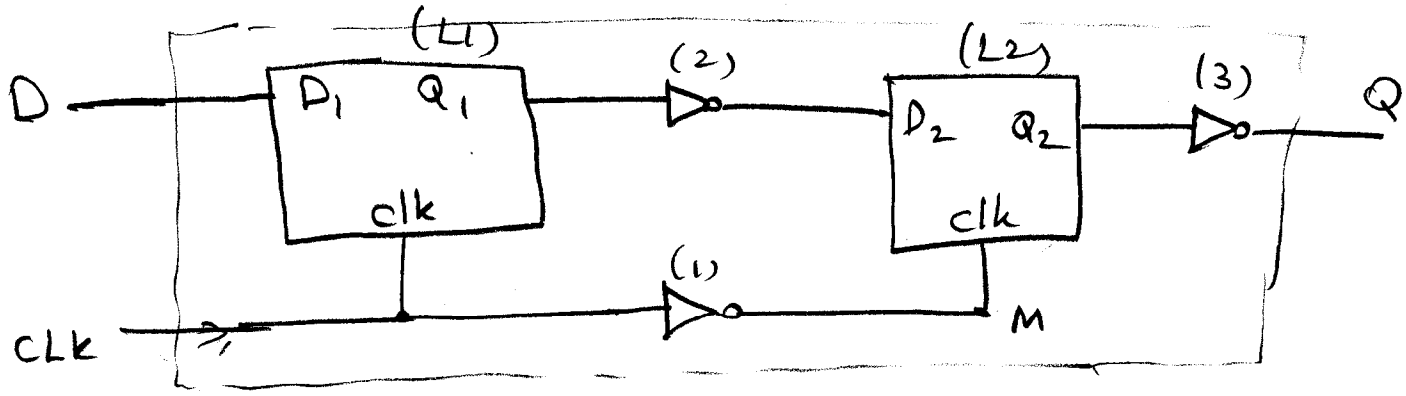
⇒ will be negative edge-triggered.

2 They examine input (D) → output (Q) relationship.

When CLK high D ~~read~~ written in, inverted, clock low → D read out, inverted. ∴ Q = D.

∴ a negative edge-triggered D flip-flop.

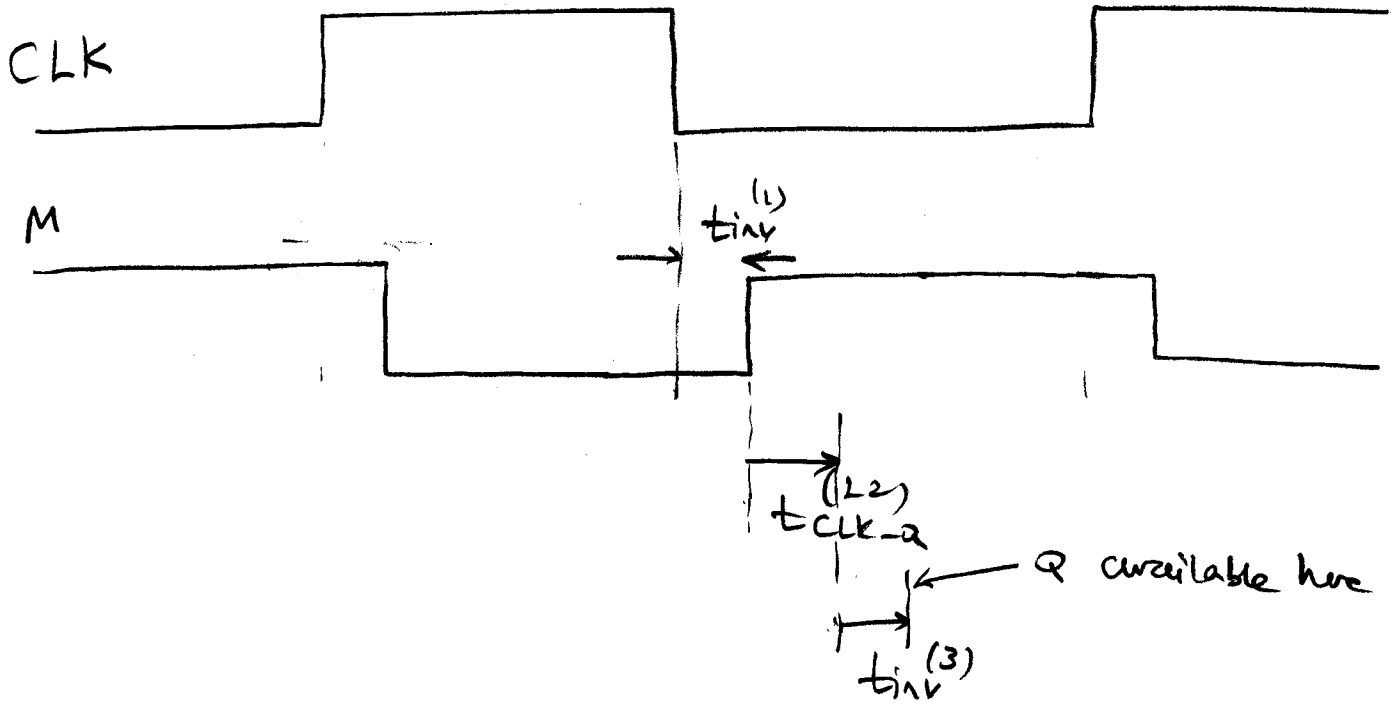
(Example 7: (continued):)



(b) CLK-Q delay:

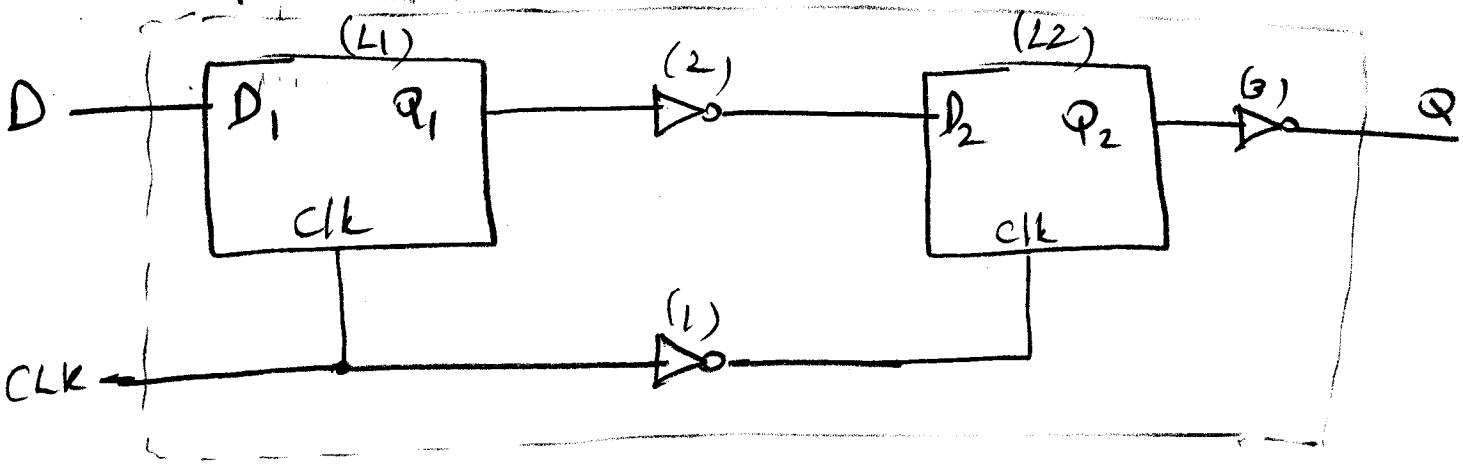
≡ max delay from (critical) CLK edge to stable Q (assuming D stable)

[Since this is a negative edge-triggered ff, critical CLK edge is falling edge; Q available after falling edge.)



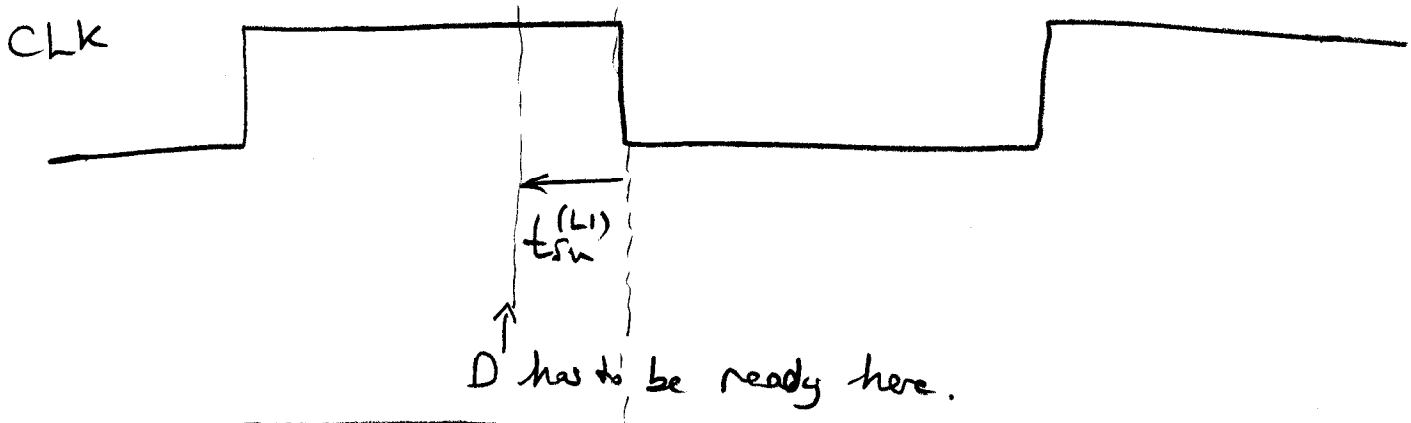
$$t_{CLK-Q} = t_{inv}^{(1)} + t_{CLK-Q}^{(L2)} + t_{inv}^{(3)}$$

(Example 7: continued):



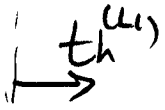
(c) Setup time

≡ Minimum duration for which D has to be stable before the active clock edge.
(falling edge here)



$$t_{su}^{(ff)} = t_{su}^{(L1)}$$

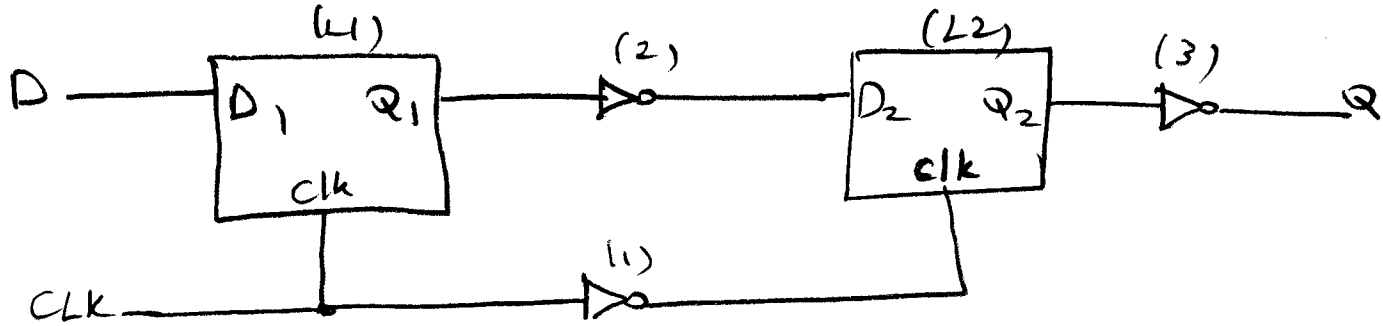
(d) Hold time: ≡ Min. duration D stable after falling clock edge



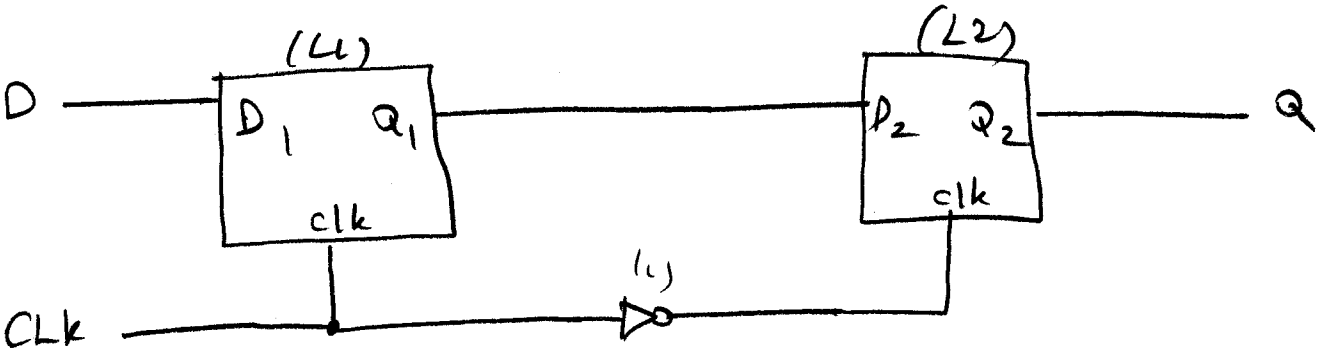
$$t_h = t_h^{(L1)}$$

(Example 7: continued):

(e) What is the advantage of this design



over the standard M-S implementation:



- Visualize what inverters (2) and (3) do to the signal.

– When does CLK arrive at the second latch?

- an inverter delay (1) afterwards

– Due to clock skew (due to $t_{inv}^{(1)}$, if $t_{clk-Q}^{(L1)}{}_{(min)} < t_{inv}$, then the value D may shoot through.)

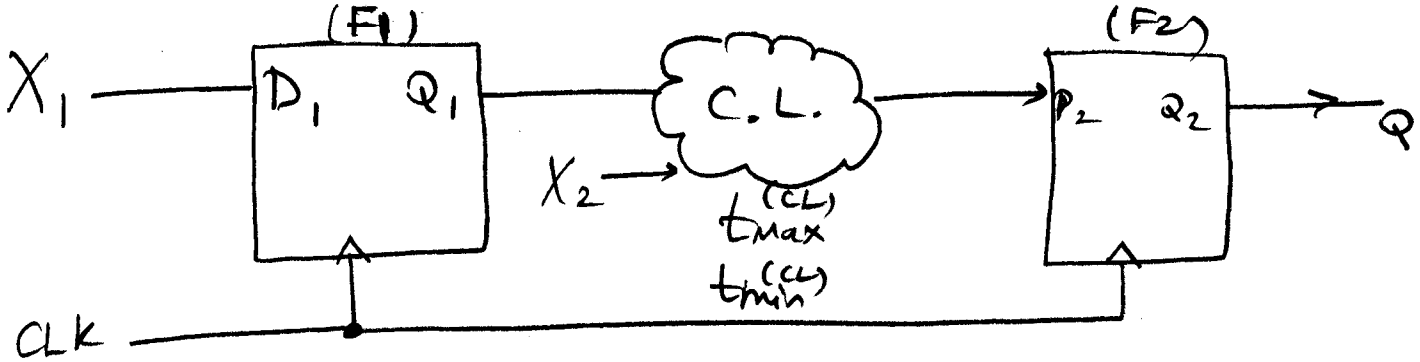
– By putting in (2), we "equalize" the delays along CLK and data routes.

$$t_{skew} < t_{clk-Q}^{(min-L1)} + t_{inv}^{(2)}$$

← increases the max. tolerable clock skew

(f) Disadvantage? – output Q available $t_{inv}^{(3)}$ ns. later

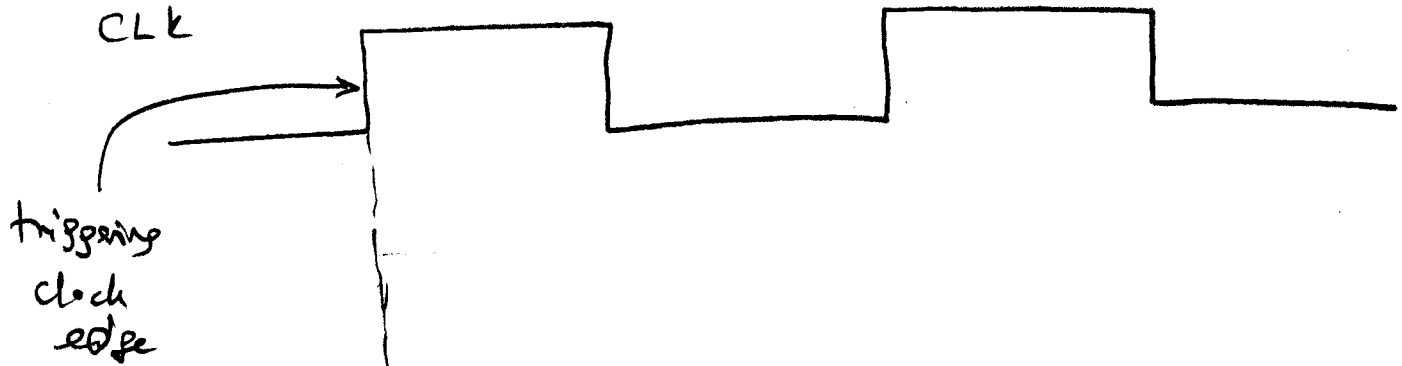
Example 8 : Pipeline Stages



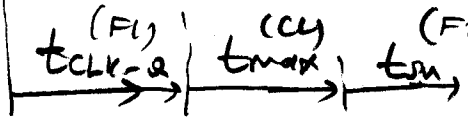
X_1, X_2 data inputs, Q output.

(a) Given this pipeline fragment, what is the <u>minimum clock period</u> for which this fragment of the pipeline will work?

– First, visualize signals to understand why it might not work if the clock period is very low:



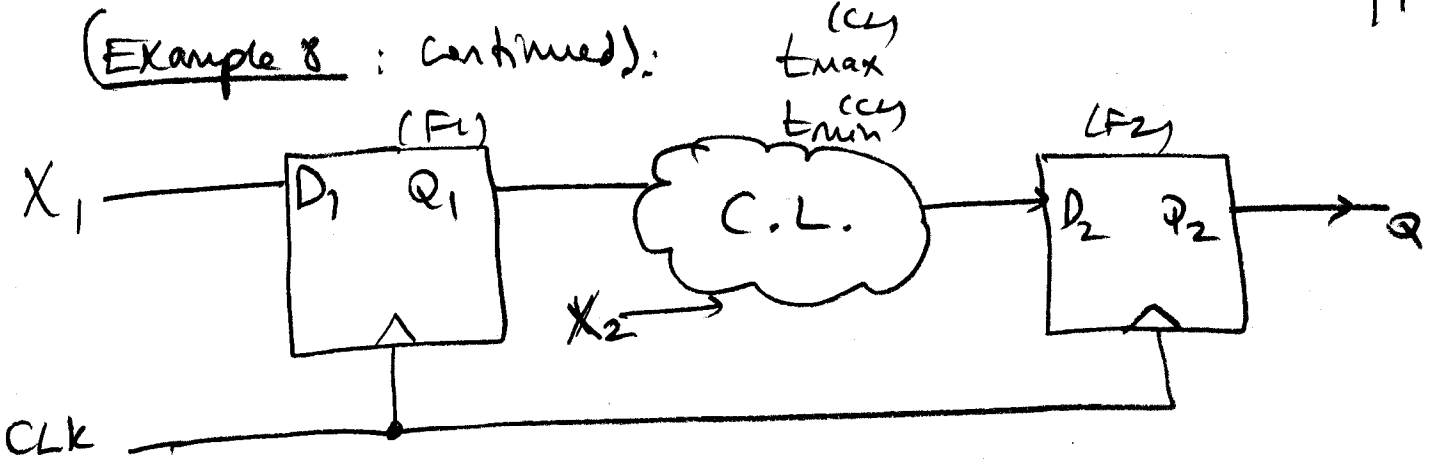
* We want the data X_1 to have the chance to propagate to D_2 and be ready at next clock edge.



$$\therefore \quad T \geq t_{CLK-Q}^{(F1)} + t_{max}^{(CL)} + t_{su}^{(F2)}$$

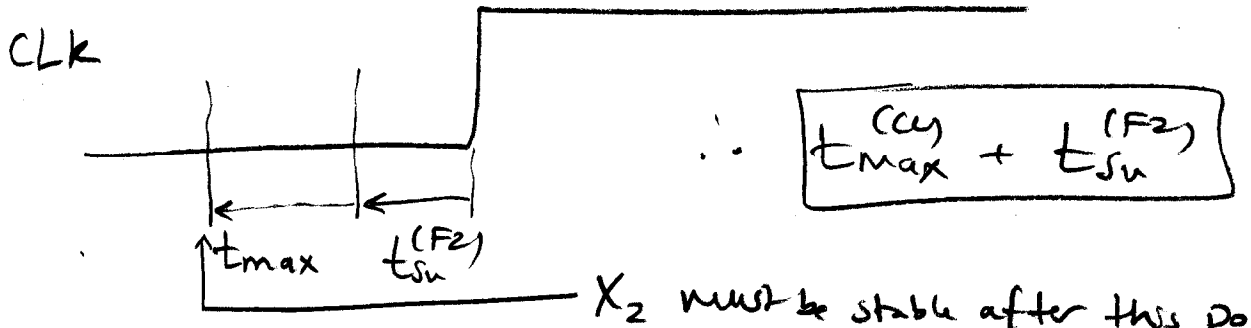
min clock period

(Example 8 : continued):



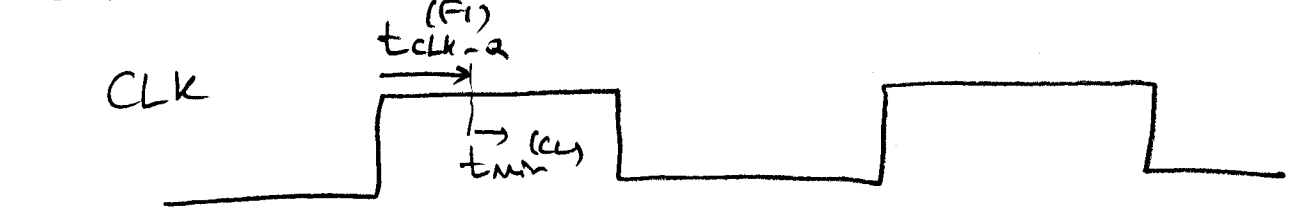
(b) What is the minimum duration for which X_2 has to be stable before the rising edge of CLK?

- Think about why things would go wrong if X_2 changed too close to CLK edge.
- We want X_2's value to be propagated & ready when clock edge arrives at F2.

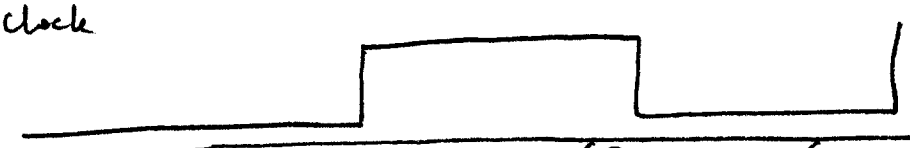


$\therefore$ $t_{max}^{(CL)} + t_{su}^{(F2)}$

(c) What is the maximum tolerable CLK skew to F2?



F2's clock



$t_{skew} < \min\{ t_{CLK-a}^{(F1)} + t_{min}^{(CL)} \}$

So that X_1 won't shoot thru