

Tricks de FDS (part II)

1. Cours

- set-reset latch: type de circuit de mémoire basique utilisé en électronique.
deux entrées : S et R. Elles sont asynchrones (pas besoin que la clock passe en front montant ou en front descendant pour qu'elles marchent) ! Elles ne dépendent que des inputs.
- D latch : comme un set-reset latch, mais plus contrôlable (input R devient input C qui nous permet de contrôler quand on veut que l'output du latch se verrouille ou suive la valeur de S).
- D flip-flops : comme un D latch, mais l'output ne prend la valeur de S que quand l'input C change de valeur (devient vrai). Quand il est vrai, même si S change, l'output reste fixe (à la valeur qu'il avait quand C était en front montant).

→ on utilise l'horloge pour décider quand notre state change, pour des soucis de synchronisation ! En effet si les composants commencent à maj leur valeur n'importe quand, et d'autres à un moment précis, on a des pb de stabilité (pendant un court moment, un output peut être invalide).

- FSM (finite state machine) -> if Moore then no input (it only takes the previous state), if Mealy it takes both the previous input **and** other inputs
- clock duty ratio : $\frac{\text{time of high sign}}{\text{duration of low sign}}$

2. Setup and Hold times

Valerio se tourne, va chercher un stylo chez Habib, et se retourne vers moi. Je prends le stylo toutes les 5 secondes → **clock !** et :

- j'ai besoin de voir le stylo 2s avant de commencer à le prendre (le moment où je vais arriver à 0 mod 5s) → **t_{setup} !** Cette condition est violée si je vais trop vite (la clock est trop rapide), que Valerio a pas le temps de me montrer le stylo. Soit Valerio doit aller plus vite (on diminue le **t_{comb}**), soit on doit me ralentir (on diminue le **clock**).
- je mets 1s à prendre le stylo **t_{hold} !** Cette condition est violée si Valerio va chercher un autre stylo chez Habib trop vite.

on veut que la valeur soit bonne avant le t_{setup}

$$\Leftrightarrow t_c Q + t_{\text{comb}} \leq t_{\text{clock}} - t_{\text{setup}}$$

Maintenant on rajoute un t_{skew} , qui décale un peu le t_{clock} (c'est du temps bonus qu'on ajoute) :

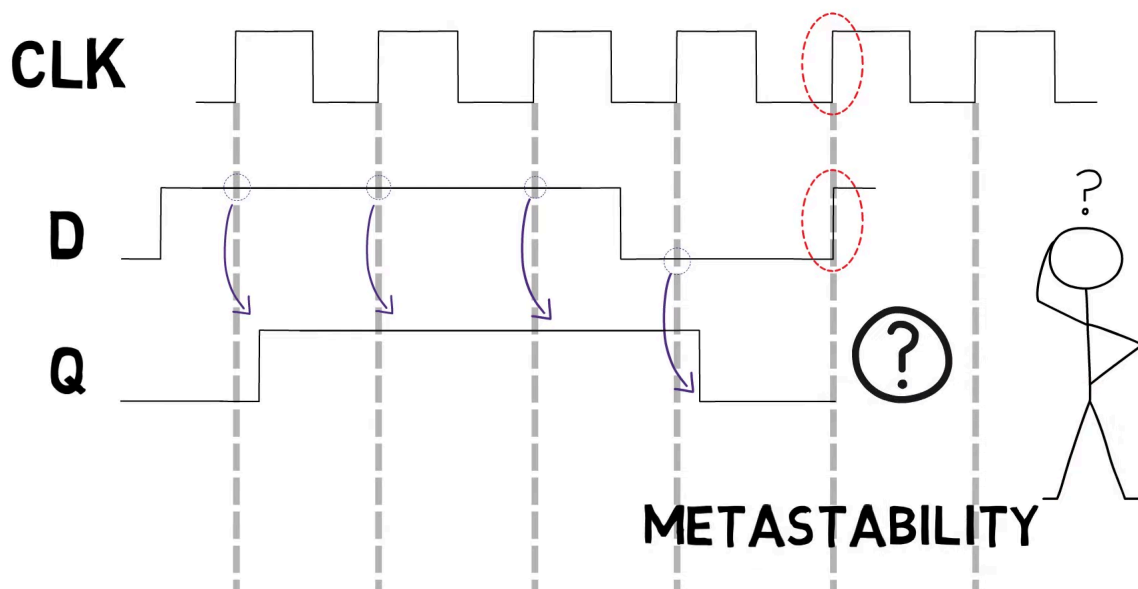
$$\Leftrightarrow t_c Q + t_{\text{comb}} \leq t_{\text{clock}} - t_{\text{setup}} + t_{\text{skew}}$$

Calculer le temps entre FF A vers FF B pour calculer f_{max}

$$t_c Q_{\text{max}} + t_{\text{comb}} + t_{\text{setup}} + t_{\text{skew_du_FF_A}} - t_{\text{skew_du_FF_B}} = \text{delay}$$

Vérifier le hold:

$$t_c Q_{\text{min}} + t_{\text{comb}} + t_{\text{skew_du_FF_A}} - t_{\text{skew_du_FF_B}} = \text{delay}$$



<https://www.youtube.com/watch?v=xCA54Qu4WtQ>

SETUP TIME is defined as the minimum amount of time before the clock's active edge that the data must be stable for it to be latched correctly

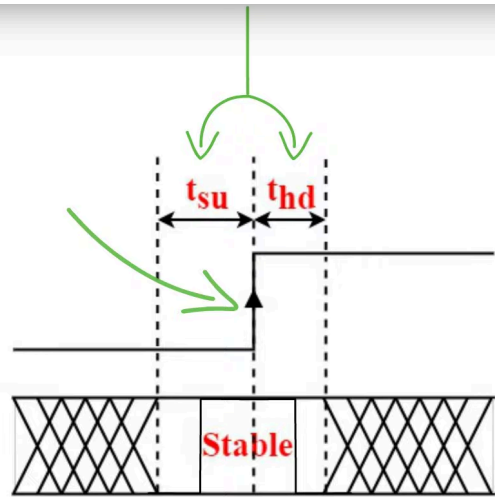
HOLD TIME is defined as the minimum amount of time after the clock's active edge during which data must be stable.



TIMING VIOLATION

CLK

D



SETUP VIOLATION

HOLD VIOLATION