



ELECINF102 : Introduction à la logique séquentielle

... ou comment organiser les choses dans le temps.

Licence de droits d'usage



Contexte académique } sans modification

Par le téléchargement ou la consultation de ce document, l'utilisateur accepte la licence d'utilisation qui y est attachée, telle que détaillée dans les dispositions suivantes, et s'engage à la respecter intégralement.

La licence confère à l'utilisateur un droit d'usage sur le document consulté ou téléchargé, totalement ou en partie, dans les conditions définies ci-après, et à l'exclusion de toute utilisation commerciale.

Le droit d'usage défini par la licence autorise un usage dans un cadre académique, par un utilisateur donnant des cours dans un établissement d'enseignement secondaire ou supérieur et à l'exclusion expresse des formations commerciales et notamment de formation continue. Ce droit comprend :

- le droit de reproduire tout ou partie du document sur support informatique ou papier,
- le droit de diffuser tout ou partie du document à destination des élèves ou étudiants.

Aucune modification du document dans son contenu, sa forme ou sa présentation n'est autorisé.

Les mentions relatives à la source du document et/ou à son auteur doivent être conservées dans leur intégralité.

Le droit d'usage défini par la licence est personnel, non exclusif et non transmissible.

Tout autre usage que ceux prévus par la licence est soumis à autorisation préalable et expresse de l'auteur :

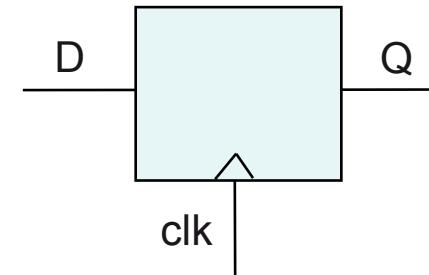
alexis.polti@telecom-paristech.fr

- **Bascules D**
 - fonctionnement
 - registre à décalage
- **Logique séquentielle synchrone**
 - rappels de logique combinatoire
 - principes et lois
 - fréquence de fonctionnement, chemin critique
 - pipe-line
- **Utilisation des bascules D**
 - avec les mains...
 - exemples importants

La bascule D

- **Bascule D :**

- logique séquentielle
- DFF, bascule D, flip-flop, registre



- **Fonctionnement :**

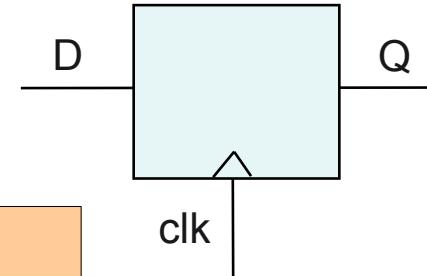
- à chaque front montant de CLK, l'entrée D est photographiée (échantillonnée)
- puis recopiée sur Q et maintenue jusqu'au prochain front montant de CLK.
- En dehors des fronts montants de CLK, Q ne change pas.

- **Remarques**

- la bascule D mémorise un état. La sortie dépend d'un état passé (celui au dernier front montant de l'horloge). Ce n'est donc pas une fonction combinatoire.
- on peut l'implémenter avec des fonction combinatoires rebouclées...

La bascule D

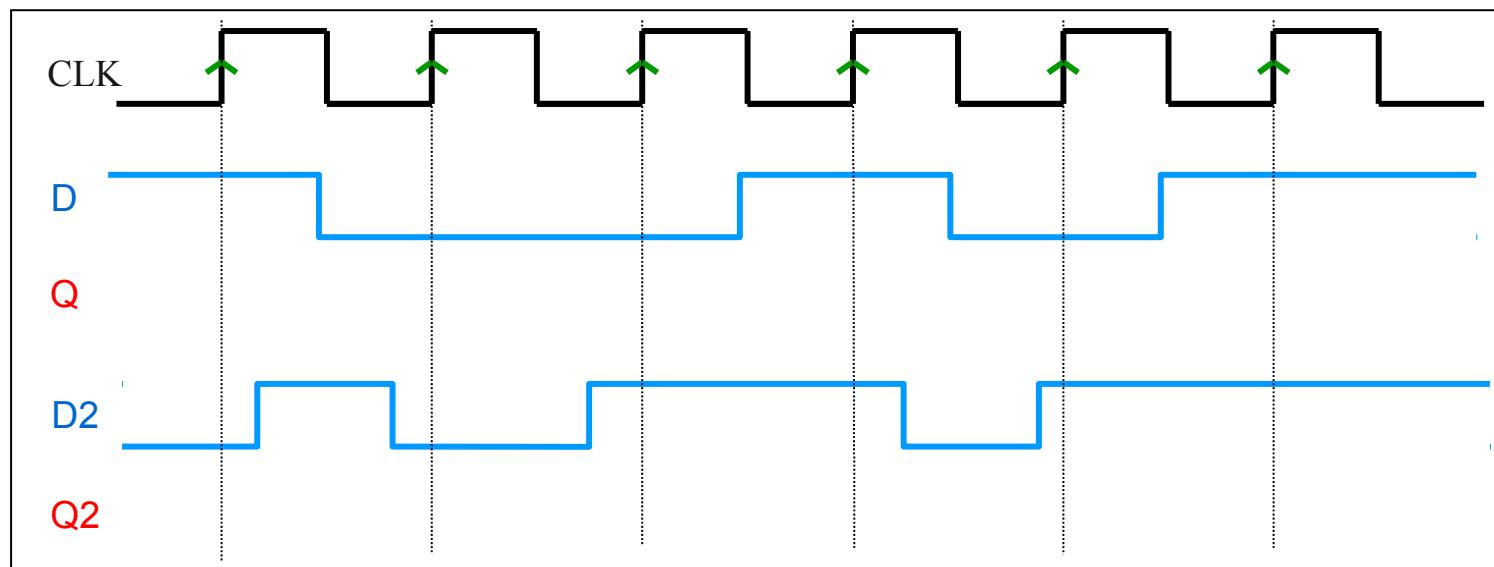
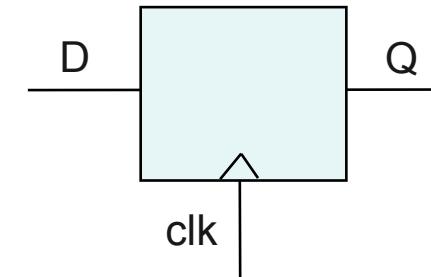
- Table de vérité :



D	CLK	Q	Remarques
0	↑	0	Q prend la valeur de D
1	↑	1	Q prend la valeur de D
X	0	Q	Q ne change pas de valeur
X	1	Q	Q ne change pas de valeur
X	↓	Q	Q ne change pas de valeur

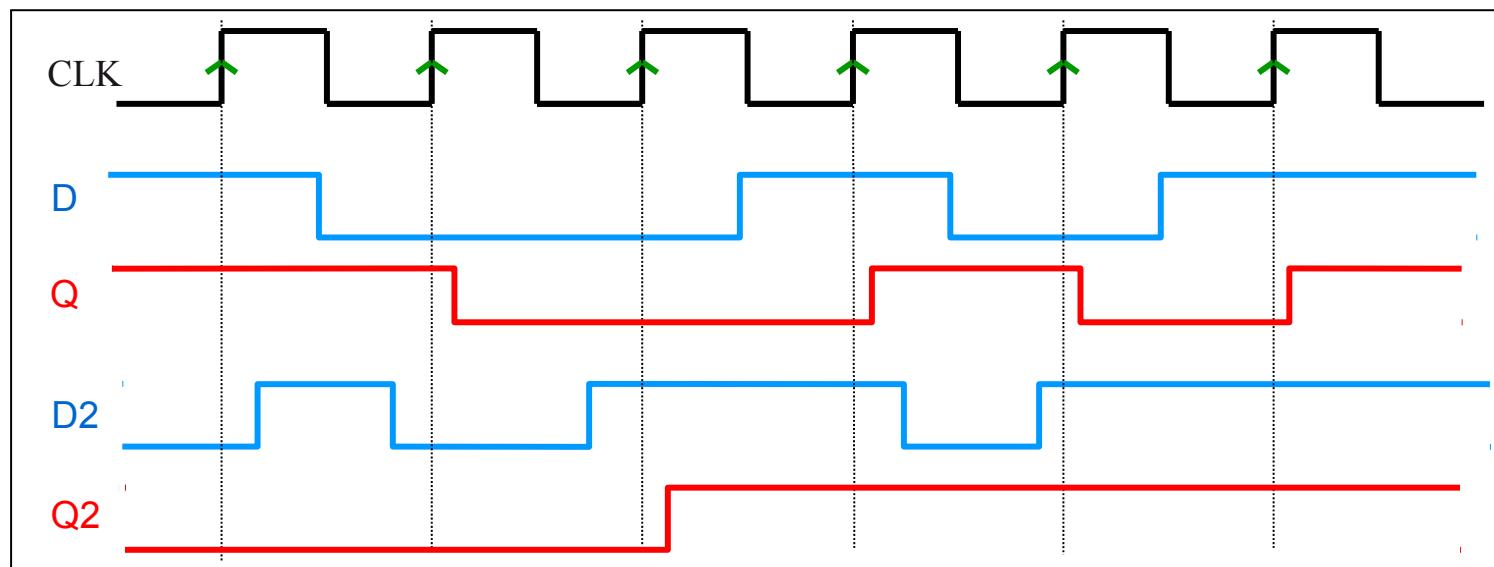
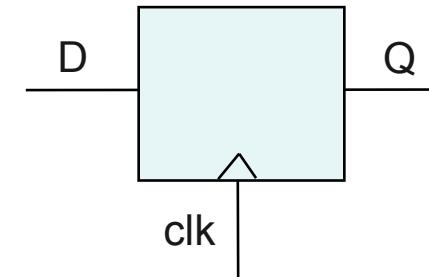
La bascule D

- Exemple :



La bascule D

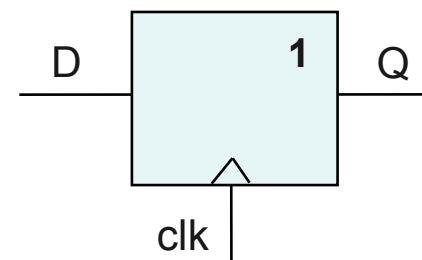
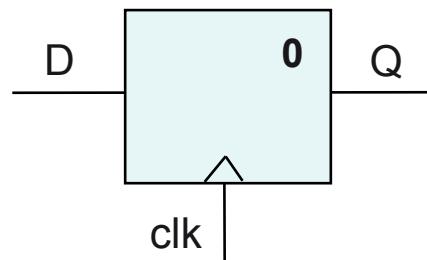
- Exemple :



La bascule D

- Initialisation des bascules D

- initialisation soit à 0 (entrée reset), soit à 1 (entrée preset)
- asynchrone
- un circuit dédié externe s'occupe de générer le signal d'initialisation de toutes les bascules D d'un circuit intégré.



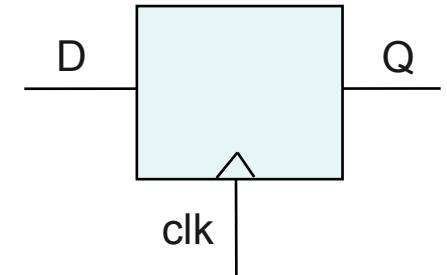
- Précision : on verra dans la suite que les signaux d'horloge et d'initialisation asynchrones ne seront pas représentés sur les schémas.

La bascule D

- **Description fonctionnelle (System Verilog) :**

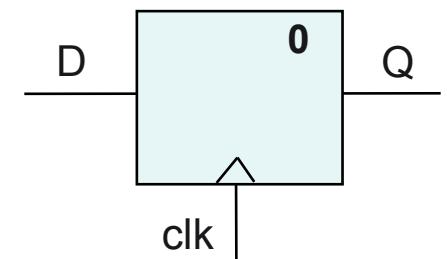
- bascule D :

```
logic q, d;  
always @ (posedge clk)  
  q <= d;
```



- bascule D avec initialisation à 0 (signal d'initialisation actif bas):

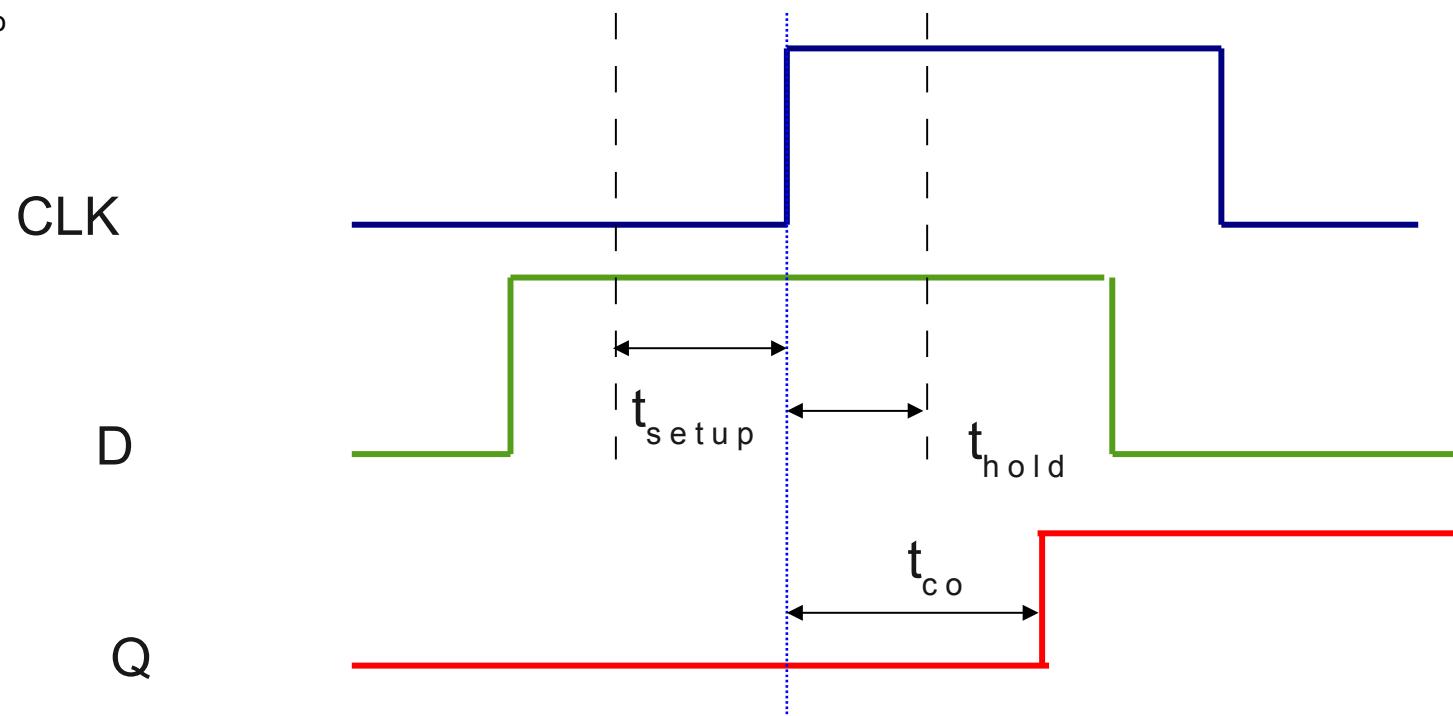
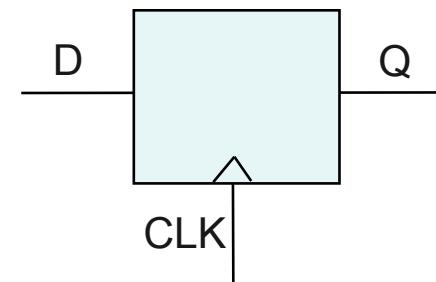
```
logic q, d;  
always @ (posedge clk or negedge init)  
  if (~init)  
    q <= 0;  
  else  
    q <= d;
```



La bascule D

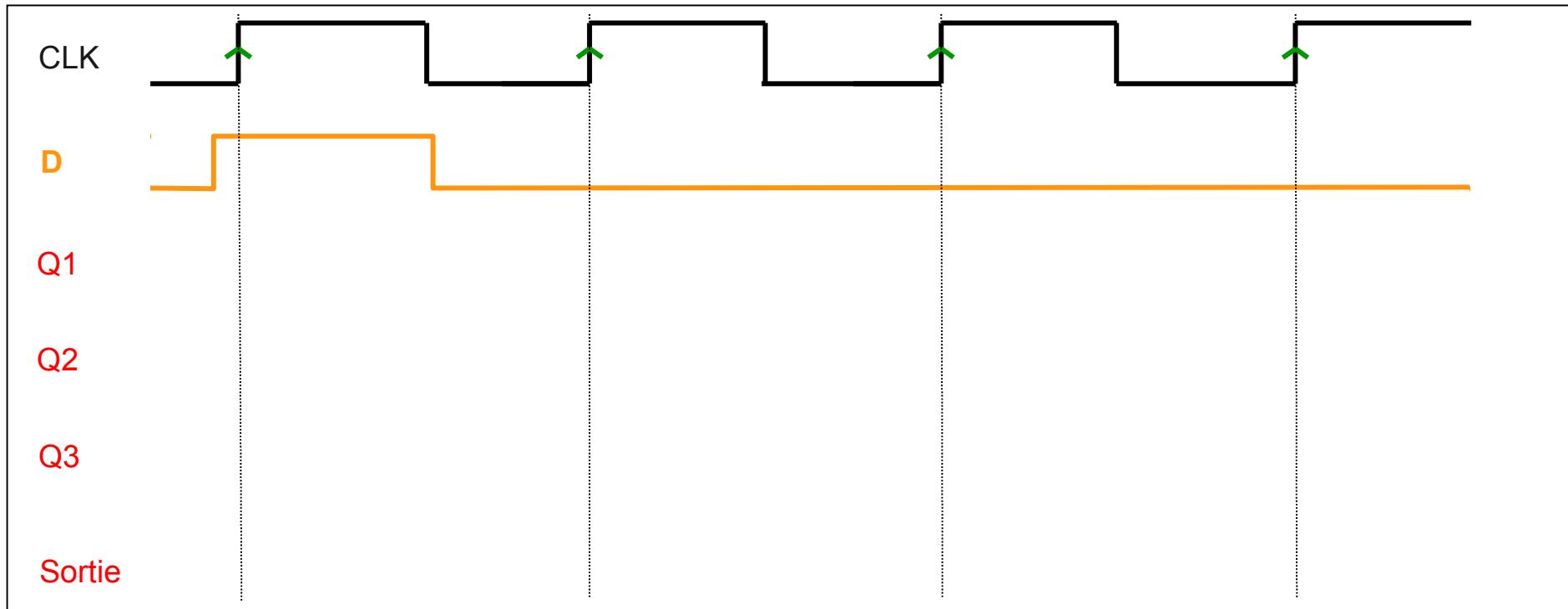
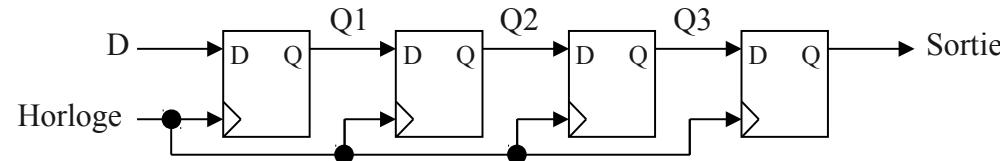
- Bascules réelles :

- photo : respecter un temps de pause pour prendre une photo nette !
 - t_{setup}
 - t_{hold}
- temps de propagation
 - t_{co}



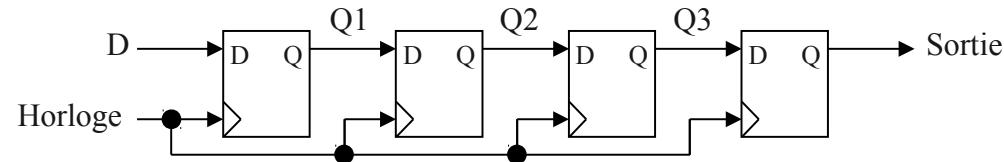
La bascule D

- **Registre à décalage :**



La bascule D

- **Registre à décalage :**



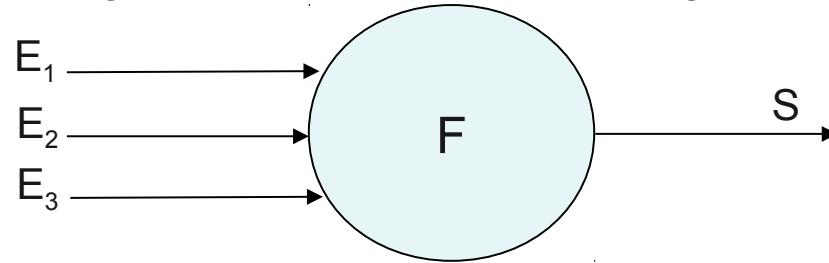
- pour que ce genre de montage fonctionne, il faut que $t_{co} > t_{hold}$
- par construction, c'est le cas
- dans la suite de ce cours, on considère que
 - $t_{setup} = 0$
 - $t_{hold} = 0$

- **Bascules D**
 - fonctionnement
 - registre à décalage
- **Logique séquentielle synchrone**
 - rappels de logique combinatoire
 - principes et lois
 - fréquence de fonctionnement, chemin critique
 - pipe-line
- **Utilisation des bascules D**
 - avec les mains...
 - exemples importants

Combinatoire

- **Principe**

- la sortie ne dépend que des entrées
- à base des portes logiques vues, sans rebouclage

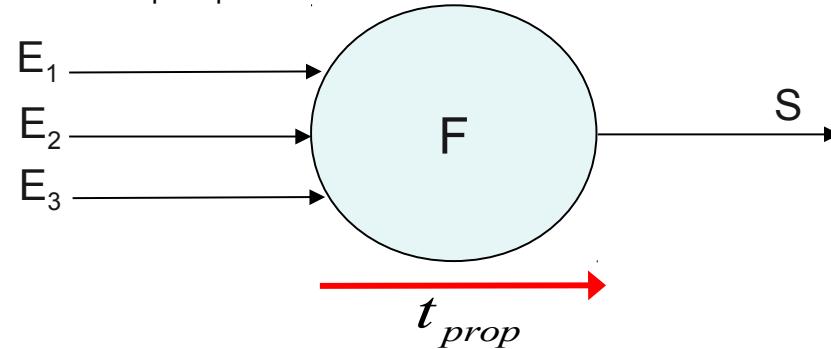


$$S = F(E_1, E_2, E_3)$$

Combinatoire

- En réalité :

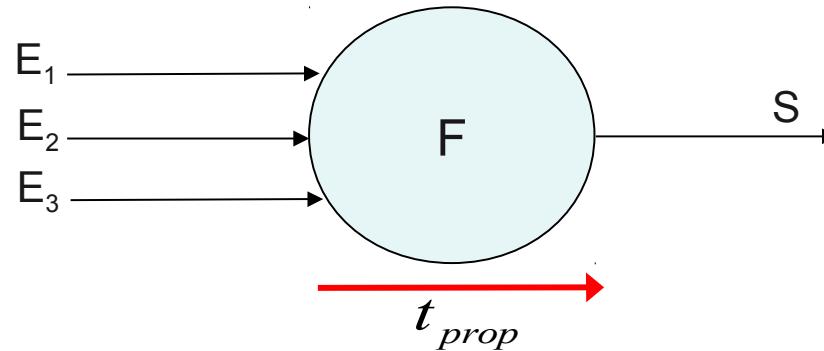
- temp de propagation (t_{prop}) non nul



Combinatoire

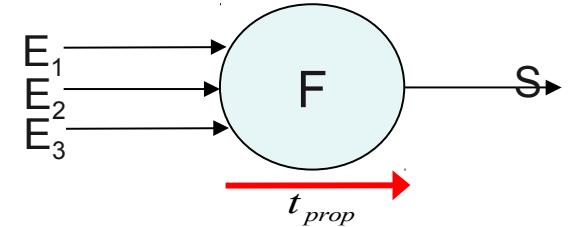
- En réalité :

- temps de propagation (t_{prop}) non nul



- à $t = t_0$: on présente les entrées, et on les garde stables
 - de t_0 à $t_0 + t_{prop}$: la sortie S fait n'importe quoi
 - à partir de $t_0 + t_{prop}$: la sortie S est stable, valide, et vaut $S = F(E_1, E_2, E_3)$

Combinatoire



- **Conséquence :**

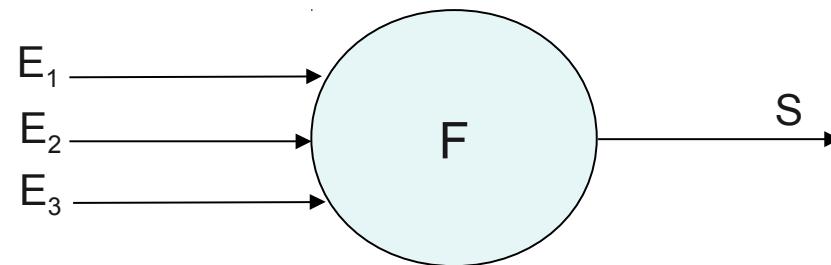
- pour utiliser une fonction combinatoire, il faut :
 - maintenir ses entrées stables pendant au moins t_{prop}
 - n'utiliser sa sortie que lorsqu'elle est valide : après t_{prop}

Logique synchrone

Principe :

on a une fonction combinatoire F qu'on veut utiliser :

- avec différents vecteurs d'entrée, présentés les uns à la suite des autres,
- le tout à une fréquence maximale



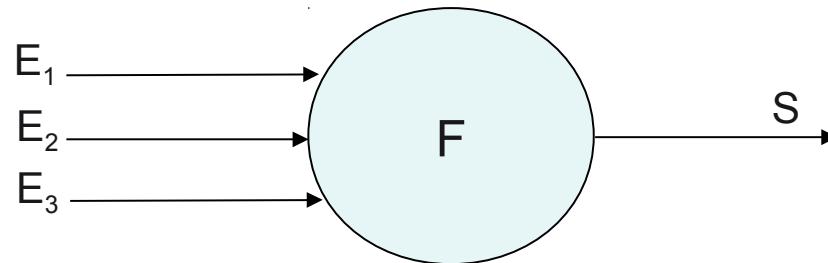
Comment faire ?

Logique synchrone

Principe :

on a une fonction combinatoire F qu'on veut utiliser :

- avec différents vecteurs d'entrée, présentés les uns à la suite des autres,
- le tout à une fréquence maximale



Comment faire ?

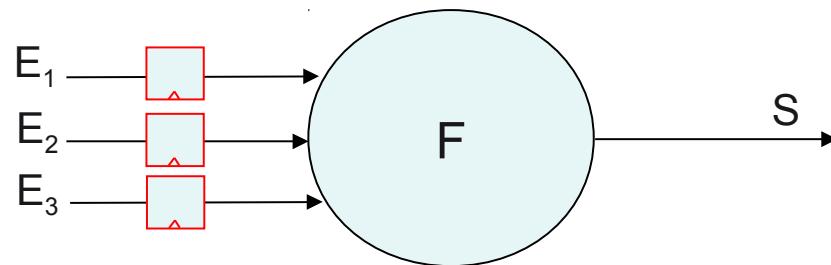
- on maintient les entrées stables pendant au moins t_{prop}
- on échantillonne (photographie) la sortie dès qu'elle est valide
- au même moment, on présente le vecteur d'entrée suivant

Logique synchrone

Principe :

on a une fonction combinatoire **F** qu'on veut utiliser :

- avec différents vecteurs d'entrée, présentés les uns à la suite des autres,
- le tout à une fréquence maximale



Comment faire ?

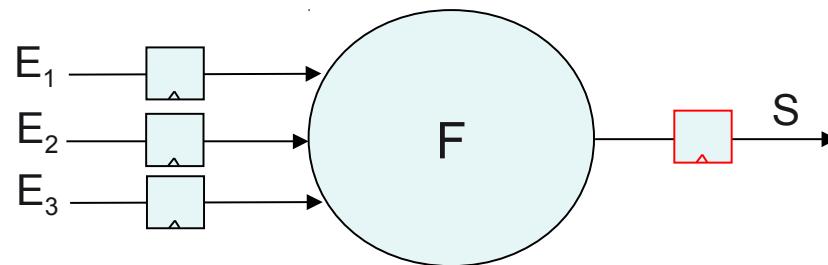
- on maintient les entrées stables pendant au moins t_{prop}
- on échantillonne la sortie dès qu'elle est valide
- au même moment, on présente le vecteur d'entrée suivant

Logique synchrone

Principe :

on a une fonction combinatoire **F** qu'on veut utiliser :

- avec différents vecteurs d'entrée, présentés les uns à la suite des autres,
- le tout à une fréquence maximale



Comment faire ?

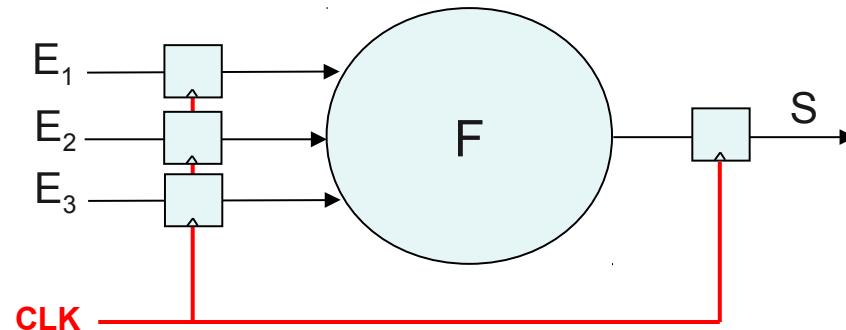
- on maintient les entrées stables pendant au moins t_{prop}
- on échantillonne la sortie dès qu'elle est valide
- au même moment, on présente le vecteur d'entrée suivant

Logique synchrone

Principe :

on a une fonction combinatoire **F** qu'on veut utiliser :

- avec différents vecteurs d'entrée, présentés les uns à la suite des autres,
- le tout à une fréquence maximale



Comment faire ?

- on maintient les entrées stables pendant au moins t_{prop}
- on échantillonne la sortie dès qu'elle est valide
- au même moment, on présente le vecteur d'entrée suivant

Logique synchrone

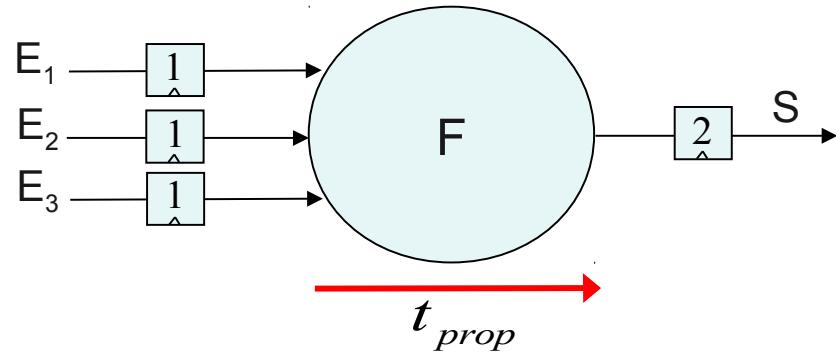
Lois de la logique séquentielle synchrone :

- les fonctions combinatoires sont entourées de registres (bascules D)
- les registres sont commandés par une même référence temporelle pour tout le circuit, appelée « horloge »
- remarques :
 - "même référence" : donc pas de portes logiques sur l'horloge
 - généralement, on ne la représente même pas (implicite)
 - de même, le signal d'initialisation asynchrone de toutes les bascules est global et ne sera pas non plus représenté. On indiquera juste dans un coin l'état de départ de la bascule.

- **Bascules D**
 - fonctionnement
 - registre à décalage
- **Logique séquentielle synchrone**
 - rappels de logique combinatoire
 - principes et lois
 - fréquence de fonctionnement, chemin critique
 - pipe-line
- **Utilisation des bascules D**
 - avec les mains...
 - exemples importants

Logique synchrone

Fréquence de fonctionnement :

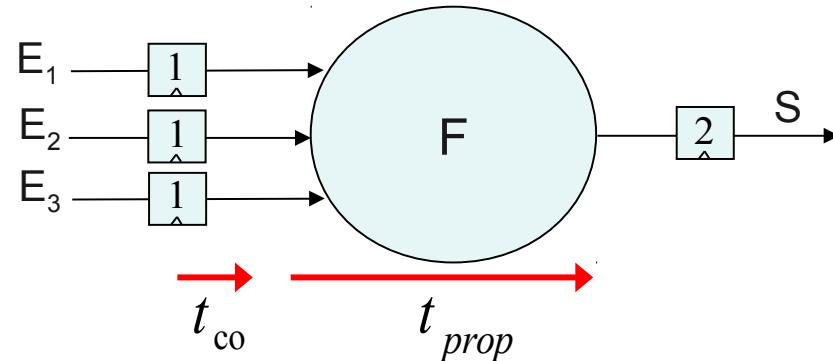


Soit t_0 le moment du premier front d'horloge

- à t_0 , les bascules D1 échantillonnent les entrées
- à $t_0 + t_{c_0}$, les bascules D1 sortent les échantillons
- à $t_0 + t_{c_0} + t_{prop}$, la sortie de F est valide, et prête à être échantillonnée par la bascule D2

Logique synchrone

Fréquence de fonctionnement :

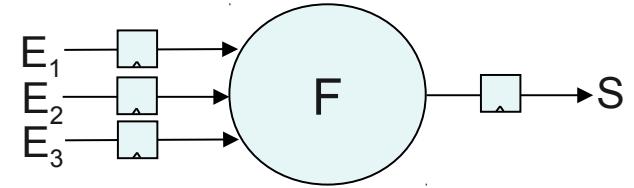


Soit t_0 le moment du premier front d'horloge

- à t_0 , les bascules D1 échantillonnent les entrées
- à $t_0 + t_{co}$, les bascules D1 sortent les échantillons
- à $t_0 + t_{co} + t_{prop}$, la sortie de F est valide, et prête à être échantillonnée par la bascule D2

$$\Rightarrow F_{max} = \frac{1}{t_{co} + t_{prop}}$$

Logique synchrone



Chemin critique :

- la relation $F_{max} = \frac{1}{t_{co} + t_{prop}}$ doit être vérifiée pour toute fonction combinatoire encadrée par des bascules D

Définition :

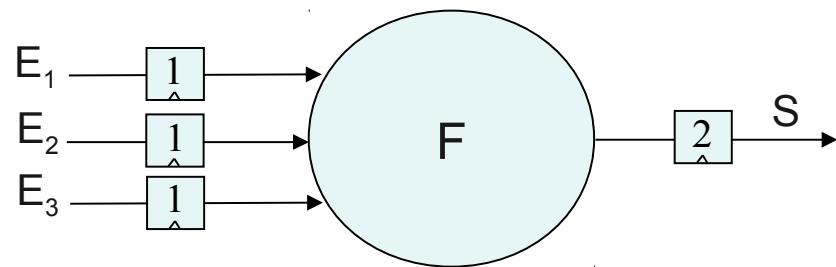
- chemin critique : le temps de propagation le plus grand entre deux bascules D consécutives

Conséquence :

- pour un circuit : $F_{max} = \frac{1}{t_{crit}}$, avec $t_{crit} = t_{co} + \max(t_{prop})$

Pipe-line

Pipe-line :

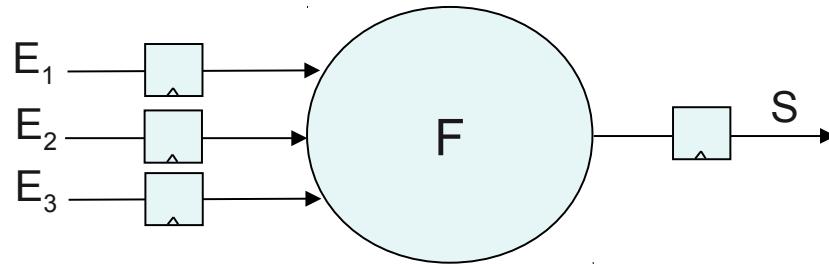


$$F_{max} = \frac{1}{t_{co} + t_{prop}}$$

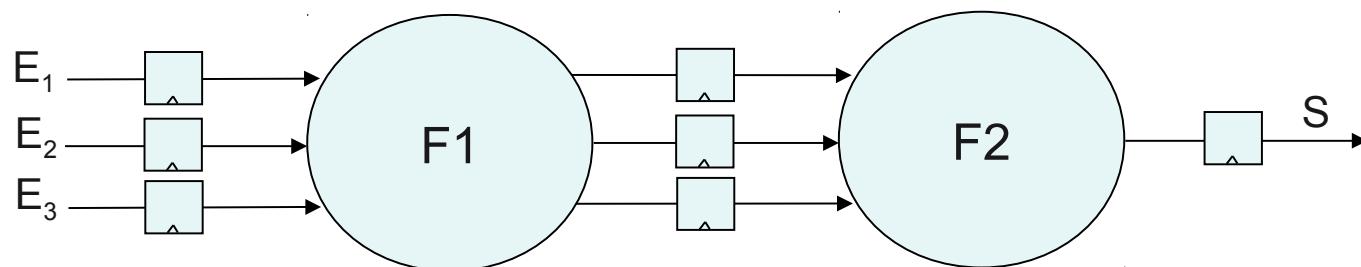
- comment faire pour fonctionner à une fréquence supérieure à F_{max} ?

Pipe-line

Pipe-line :



- on décompose la fonction F en deux fonctions plus rapides F_1 et F_2 telles que $F = F_2 \circ F_1$
- on insère une barrière de registres entre F_1 et F_2
- on est ramené au cas précédent, mais avec des fonctions plus petites (rapides)



$$\Rightarrow F_{max} = \frac{1}{t_{co} + \max(t_{prop1}, t_{prop2})}$$

Pipe-line

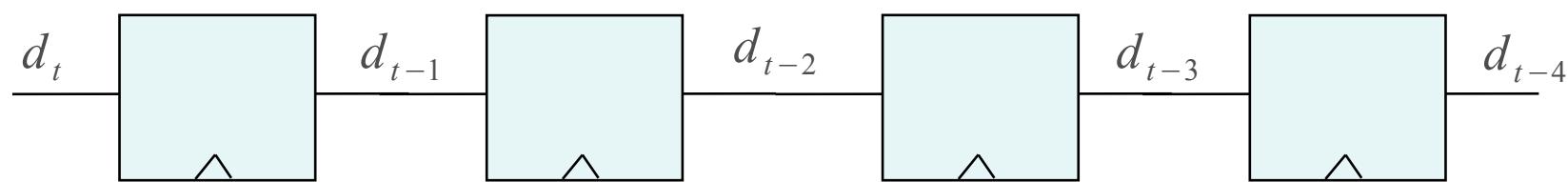
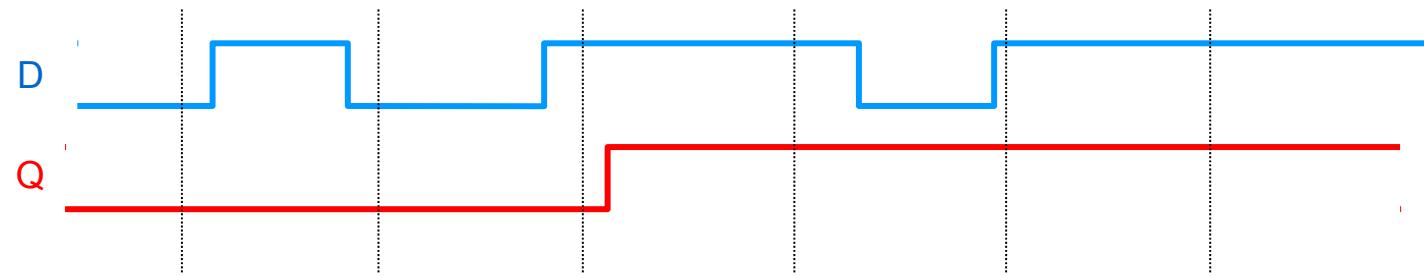
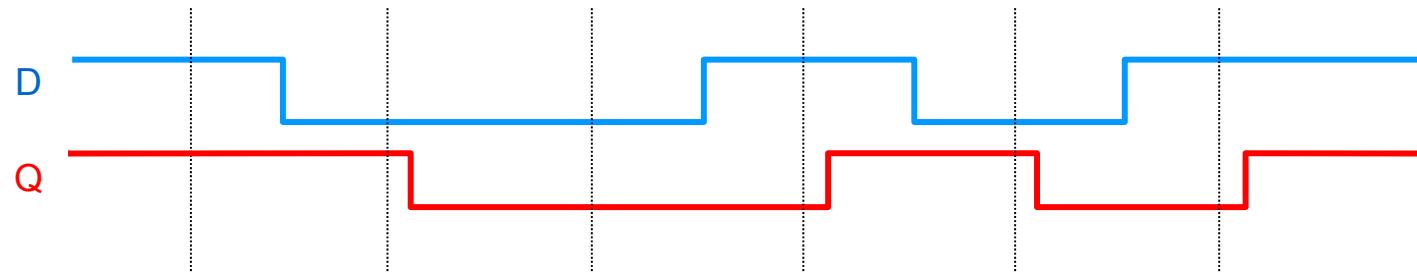
Pipe-line :

- avantage :
 - augmente la fréquence de fonctionnement (débit)
- inconvénients :
 - augmente la latence
 - augmente la taille du circuit

- **Bascules D**
 - fonctionnement
 - registre à décalage
- **Logique séquentielle synchrone**
 - rappels de logique combinatoire
 - principes et lois
 - fréquence de fonctionnement, chemin critique
 - pipe-line
- • **Utilisation des bascules D**
 - avec les mains...
 - exemples importants

Utilisation des bascules D

- exemples :



Utilisation des bascules D

Avec les mains

Les bascules D servent à :

- mémoriser des informations
- recaler des signaux sur le prochain front d'horloge
- pour des signaux déjà recalés, les retarder d'une période d'horloge



Exemples

Bascule D avec enable :

- On veut une bascule D avec "enable"
 - si enable vaut 1, la bascule fonctionne normalement
 - si enable vaut 0, la bascule est gelée, Q garde sa valeur même après un front montant de l'horloge
- Comment l'implémenter ?



Exemples

Compteur : donner à chaque fois un schéma et un code SystemVerilog

- construire un système qui compte (+1) à chaque coup d'horloge (max 255 puis reboucle à 0)
- rajouter la possibilité de le remettre à zéro de façon
 - asynchrone
 - synchrone
- lui rajouter une entrée permettant de le mettre :
 - en mode comptage (+1)
 - ou décomptage (-1)
- rajouter au compteur un système pour qu'il ne compte que jusqu'à 13
 - arrivé à 13, il se bloque
 - arrivé à 13, il repart à 0



Exemples

Exercices :

- Faire un détecteur de front montant
- Faire un système qui sort un signal à 1 tous les n coups d'horloge
 - n fixe
 - n réglable
- On a un flux 8 bits :
 - faire la somme des 4 derniers échantillons
 - faire la moyenne des 4 derniers échantillons
 - même chose avec strictement moins de trois additionneurs
- Sérialisation / parallélisation :
 - que vaut la sortie d'un registre à décalage de n étages ?
 - on a un flux 1 bit ; récupérer ses 4 derniers bits sur un bus 4 bits
 - faire l'opération inverse : un convertisseur parallèle / série



Exemples

Exercices :

- faire une roue codeuse
- faire un digicode
- forcer le copybit SPDIF
- faire un système qui génère les multiples de 2 et 3 dans l'ordre croissant
- faire un générateur PWM
- faire un debouncer
- faire un limiteur audio

Licence de droits d'usage



Contexte académique } sans modification

Par le téléchargement ou la consultation de ce document, l'utilisateur accepte la licence d'utilisation qui y est attachée, telle que détaillée dans les dispositions suivantes, et s'engage à la respecter intégralement.

La licence confère à l'utilisateur un droit d'usage sur le document consulté ou téléchargé, totalement ou en partie, dans les conditions définies ci-après, et à l'exclusion de toute utilisation commerciale.

Le droit d'usage défini par la licence autorise un usage dans un cadre académique, par un utilisateur donnant des cours dans un établissement d'enseignement secondaire ou supérieur et à l'exclusion expresse des formations commerciales et notamment de formation continue. Ce droit comprend :

- le droit de reproduire tout ou partie du document sur support informatique ou papier,
- le droit de diffuser tout ou partie du document à destination des élèves ou étudiants.

Aucune modification du document dans son contenu, sa forme ou sa présentation n'est autorisé.

Les mentions relatives à la source du document et/ou à son auteur doivent être conservées dans leur intégralité.

Le droit d'usage défini par la licence est personnel, non exclusif et non transmissible.

Tout autre usage que ceux prévus par la licence est soumis à autorisation préalable et expresse de l'auteur :

alexis.polti@telecom-paristech.fr