



IP PARIS

Systemes numériques

Base d'architectures

Tarik Graba

Année scolaire 2024/2025

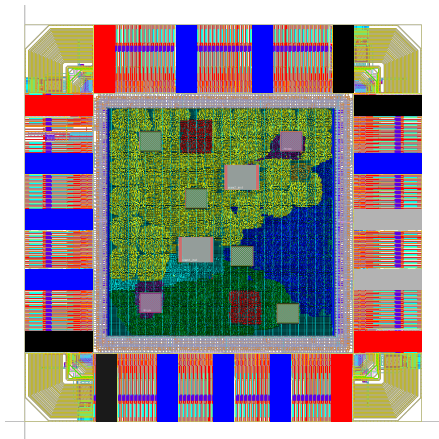


Électronique numérique

Un circuit intégré numérique

Un circuit intégré est un objet physique avec:

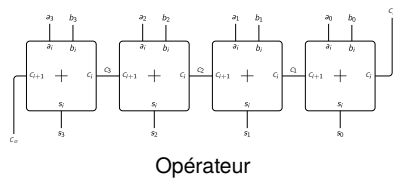
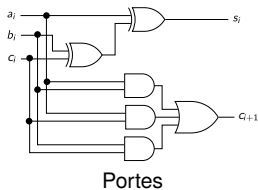
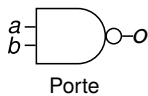
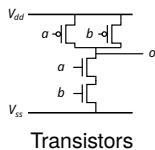
- des entrées sorties
 - pour interagir avec l'extérieur
- des opérateurs de calcul et de décision
 - pour faire quelque chose
- éléments mémorisant
 - pour se souvenir de ce qu'il fait



Électronique numérique

Logique combinatoire

On sait construire des portes logiques et les assembler en opérateurs:

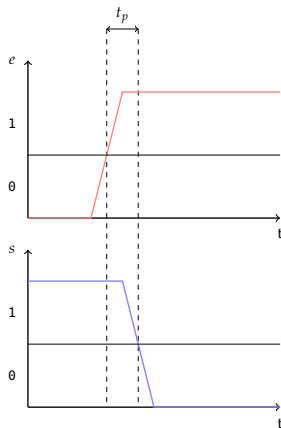
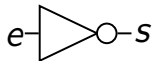


Électronique numérique

Logique combinatoire

On sait que les calculs se propagent.

- les signaux se propagent de porte en porte
- les signaux se propagent dans les connexions
- changer d'état prend du temps
 - en fonction de la physique de la porte
 - en fonction du nombre de portes qui la suivent (sa charge)





Plan

Logique séquentielle synchrone

L'horloge

Remise à zéro

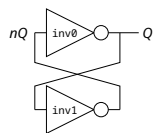
Mémoires

Entrée/sorties

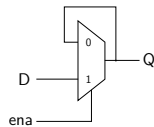
Données et contrôle

L'élément mémorisant synchrone: la bascule D

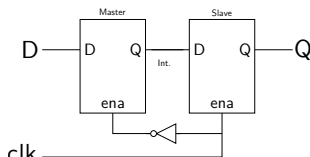
Du point mémoire à la bascule sur front



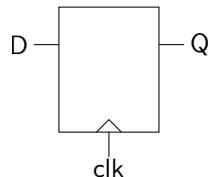
Point mémoire
CMOS



Latch
mémorisation sur
niveau



Bascule D
mémorisation sur front



Bascule D (D-FF) ou
registre

Pourquoi de la logique synchrone?

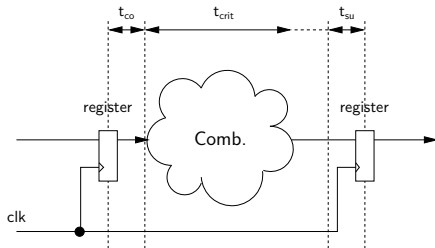
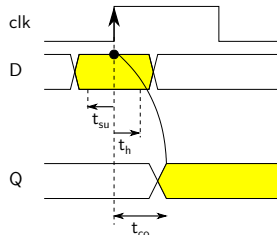
Comportement garanti quand les *timings* sont respectés:

■ Signaux stables au front de l'horloge

- garanti si on sort d'un registre ($t_{co} > t_h$)

■ Chemin critique/période d'horloge

$$T > t_{co} + t_{crit} + t_{su}$$





Plan

Logique séquentielle synchrone

L'horloge

Remise à zéro

Mémoires

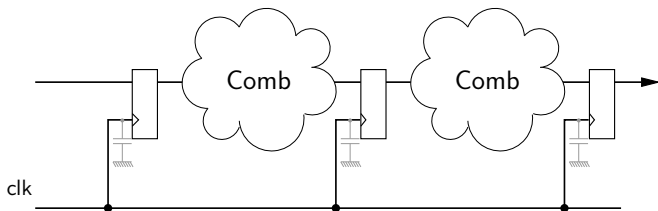
Entrée/sorties

Données et contrôle

Synchronisme

Propagation de l'horloge

L'horloge est supposée arriver en même temps à tous les registres du circuit.

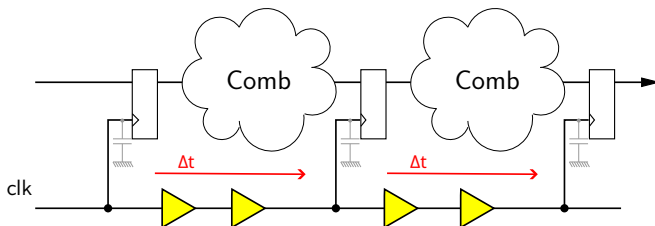


- Le signal d'horloge est chargé
- Il faut l'amplifier

Synchronisme

Propagation de l'horloge

Cette amplification (avec des buffers) ajoute du délai

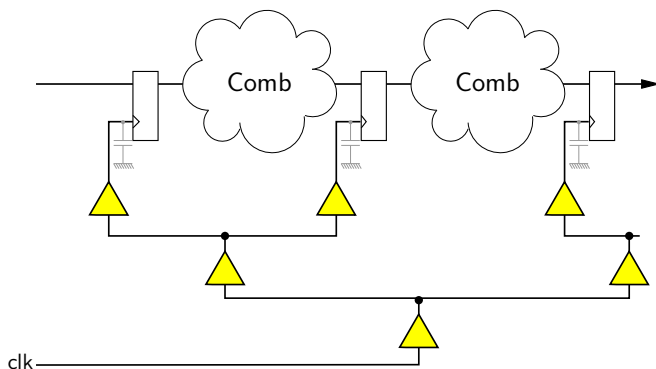


L'horloge n'arrive plus en même temps aux registres!

Synchronisme

L'arbre d'horloge

Pour équilibrer les charges et les temps de propagation des horloges, on construit un arbre (*clock tree*).





Synchronisme

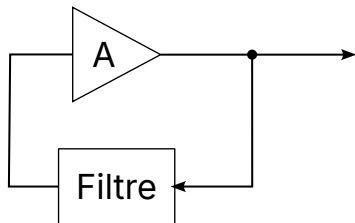
Le signal d'horloge

- Les contraintes de propagation sur les signaux d'horloge en font des signaux spéciaux.
- Il ne faut pas ajouter de logique sur les signaux d'horloge au risque de rendre le synchronisme complexe à mettre en œuvre.

Génération d'un signal d'horloge

Comment générer une horloge stable?

Un oscillateur est un composant analogique



Un oscillateur harmonique est système bouclé :

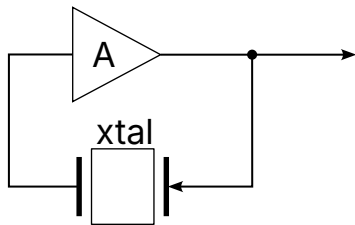
- un amplificateur pour régénérer le signal,
- un filtre sélectif pour conserver une seule fréquence,
 - filtre analogique *LC* par exemple.

avec une contre-réaction en opposition de phase et un gain de 1.

Génération d'un signal d'horloge

Comment générer une horloge stable?

Les plus stables utilisent un cristal de quartz comme filtre.

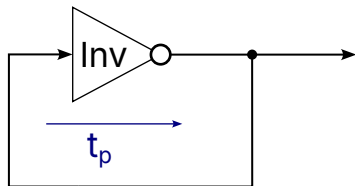


- Filtre très sélectif et dérive très faible (quelques ppm).
- Le Quartz est forcément à l'extérieur du circuit.
 - l'électronique de l'oscillateur peut être en partie intégré au circuit, mais il existe des oscillateurs complets.

Génération d'un signal d'horloge

Peut-on faire des oscillateurs numériques?

Oui, mais ils ne sont pas bons.



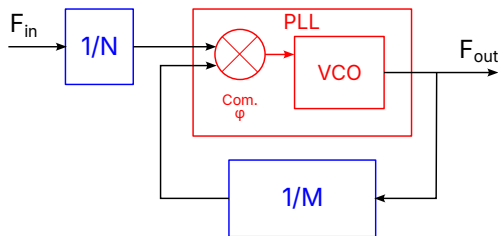
- en rebouclant un nombre impair d'inverseurs.
- $F = 1/(2 \cdot t_p)$
 - Cette fréquence n'est pas très stable ni vraiment maîtrisée.

Génération d'un signal d'horloge

Comment changer la fréquence dans le circuit?

En utilisant une PLL (*Phase-Locked Loop*)

- Génère des horloges de fréquences différentes à partir d'une **horloge stable** en entrée.
 - la fréquence de l'horloge externe est souvent limitée par la bande passante des IOs
- Composant souvent analogique,
 - comparateur de phase et oscillateur contrôlé
 - configurable numériquement (diviseurs de fréquence)



$$F_{out} = F_{in} \frac{M}{N}$$



Plan

Logique séquentielle synchrone

L'horloge

Remise à zéro

Mémoires

Entrée/sorties

Données et contrôle

État *initial* d'une bascule D

L'état à la **mise sous tension** d'une bascule D/registre/point mémoire n'est pas maîtrisé.

- symétrie des éléments mémorisant,
- probabilité d'avoir 0 ou 1 au démarrage

S'il faut maîtriser l'état initial, il faut utiliser un signal externe pour forcer l'état initial (le reset)

- asynchrone, qui agit immédiatement
 - généralement utilisé pour l'initialisation globale
- synchrone, qui agit au front d'horloge
 - généralement utilisé pour de l'initialisation fonctionnelle

Synchronisme

Et par rapport à l'horloge?

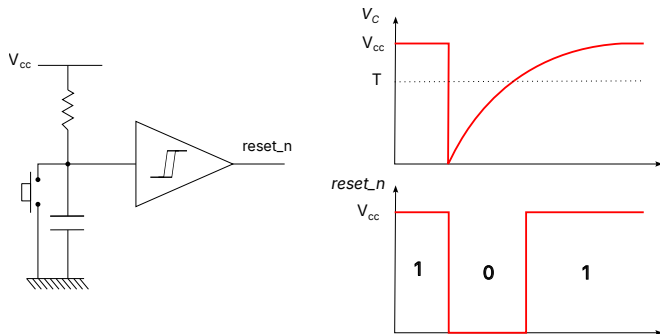
Comme pour l'horloge, doit être traité de façon particulière.

- Un signal de remise à zéro global doit arriver à tous les registres qui doivent être initialisés.
- Il suffit de le maintenir actif suffisamment longtemps.
- pas besoin d'arbre ici.
- La désactivation du reset doit être vue en même temps par tous les registres.
- Ce passage doit être cohérent par rapport à l'horloge.
- Vous verrez des techniques pour garantir un comportement synchrone cohérent.

Signal de remise à 0

Comment le générer

Exemple de circuit pour générer le signal reset.



Ce circuit assez simple permet de maîtriser le temps durant lequel le reset est actif.



Plan

Logique séquentielle synchrone

L'horloge

Remise à zéro

Mémoires

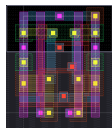
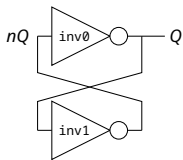
Entrée/sorties

Données et contrôle

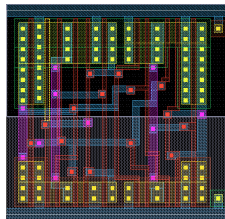
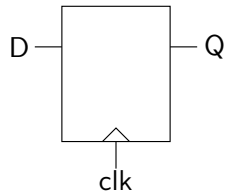
Blocs mémoire

Point SRAM vs D-FF

Pour augmenter la densité.



Point mémoire CMOS

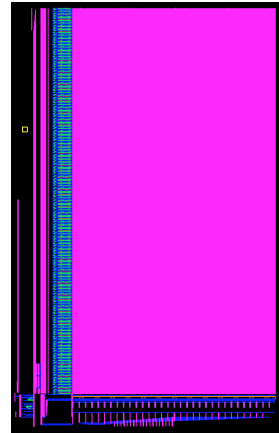
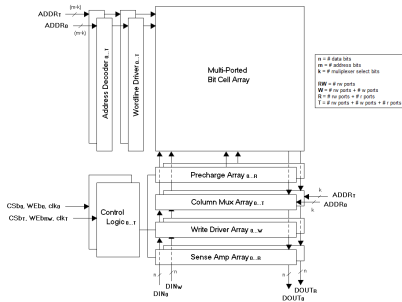
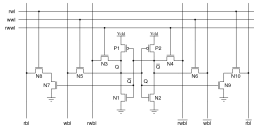


Bascule D (D-FF)

Source OpenRAM : <https://github.com/VLSIDA/OpenRAM/blob/stable/docs>

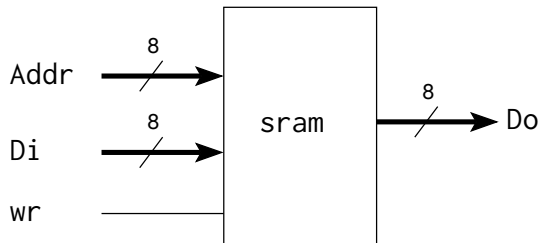
Blocs mémoire

Mettre en commun la logique de sélection (décodage d'adresse) pour construire des matrices denses.



Source OpenRAM : <https://github.com/VLSIDA/OpenRAM/blob/stable/docs>

Une table unique et dense, mais:

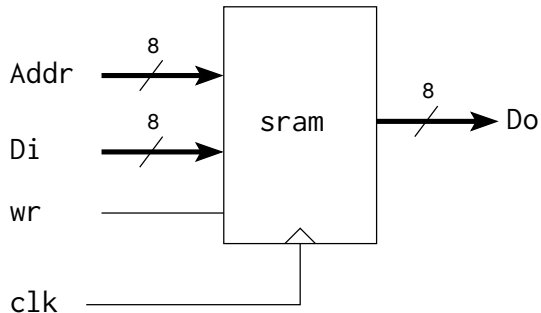


Il existe des mémoires dual-ports permettant deux accès concurrents.

■ Un seul élément à la fois:

- l'élément est sélectionné par son adresse
- seul cet élément peut être lu
- seul cet élément peut être modifié

Blocs mémoire synchrones



- Des blocs de mémoire synchrones.
- Pour être utilisée conjointement avec de la logique synchrone:
 - ajout de registres en sortie
 - ces registres sont intégrés au bloc
- Simplifie l'analyse des temps de propagation



Plan

Logique séquentielle synchrone

L'horloge

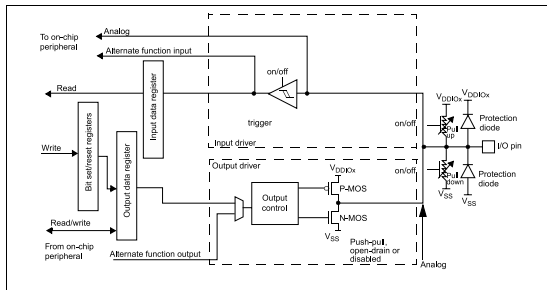
Remise à zéro

Mémoires

Entrée/sorties

Données et contrôle

Cellule d'entrée/sortie



Cellule d'entrées/sortie d'un stm32l475

Le rôles des cellules d'I/O est

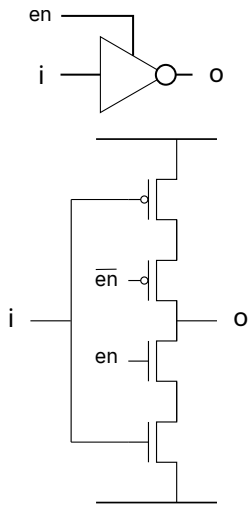
- d'adapter les niveaux de tension
- pour les horloges, les temps de montée et de descente (trigger)

Elles peuvent être configurables:

- Entrée ou sortie ou les deux
- Le niveau de tension externe
- Le drive (limite de courant)

Ce sont des cellules avec des fonctions analogiques.

Tri-state Buffer 3-états



Cellule CMOS qui permet de "déconnecter" un signal.

- si désactivé, la sortie est flottante.
- sinon, un signal numérique standard.

Cet état est symbolisé par la lettre Z .



Plan

Logique séquentielle synchrone

L'horloge

Remise à zéro

Mémoires

Entrée/sorties

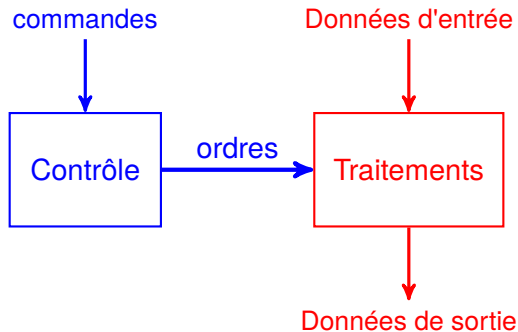
Données et contrôle

Séparation donnée contrôle

Comment s'organiser?

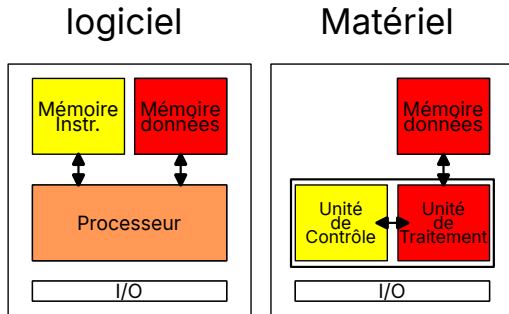
Dans une architecture, on distingue traditionnellement:

- le **traitement** des données,
- le **contrôle** de ce traitement.



Séparation donnée contrôle

Comment s'organiser?



Dans une implémentation logicielle:

- le processeur fait les deux
- le contrôle vient du programme

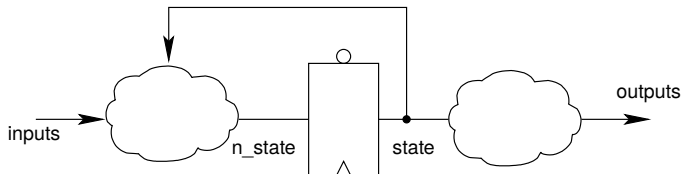
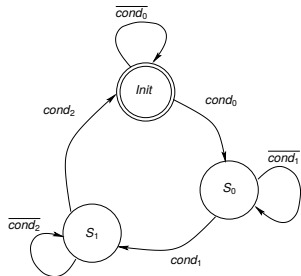
Dans une implémentation matérielle:

- les deux sont séparées
- l'unité de contrôle est matérielle

Automates/Machine à état

Comment les implémenter de façon matérielle

Comment passer d'une séquence/automate à une implémentation séquentielle synchrone.



Faire évoluer des sorties en fonction d'un état et des évolutions des entrées.