

Architecture des ordinateurs

Circuits logiques séquentiels

Pr. Zaynab EL KHATTABI

Plan du cours

- ✓ Introduction
- ✓ Rappels
- ✓ Circuits séquentiels
 - ✓ Définitions
 - ✓ Les bascules
 - ✓ Bi-stable
 - ✓ RS-Latch
 - ✓ RST
 - ✓ D-Latch
 - ✓ D-flip flop
 - ✓ Autres types de bascules
- ✓ Les registres

Introduction

- Tout ordinateur est conçu à partir de **circuits intégrés** qui ont tous une fonction spécialisée (ALU, mémoire, circuit décodant les instructions etc.)
- Ces circuits sont fait à partir de **circuits logiques** dont le but est d'exécuter des opérations sur des variables logiques (binaires).
- Les circuits logiques sont élaborés à partir de composants électroniques – transistors
- Types de circuits logiques:
 - **Combinatoires**
 - **Séquentiels**

Rappels : Circuits combinatoires

- Les fonctions de sortie s'expriment selon **des expressions logiques** des variables d'entrée seulement.
- Un circuit combinatoire est défini par une ou plusieurs **fonctions logiques**
- Le circuit combinatoire est défini lorsque son **nombre d'entrées**, son **nombre de sorties** ainsi que **l'état de chaque sortie** en fonction des entrées ont été précisés.
- Ces informations sont fournies grâce à une **table de vérité**
- La table de vérité d'une fonction de **n variables** à **2^n lignes - états d'entrée**
- Algèbre de Boole et les fonctions logiques sont le support théorique des circuits combinatoires



Rappels : Variables booléennes

- Un **système binaire** est un système qui ne peut exister que dans **deux états autorisés**.
- Diverses notations peuvent être utilisées pour représenter ces deux états :
 - **numérique** : 1 et 0
 - **logique** : vrai et faux
 - **électronique** : ON et OFF, haut et bas
- Une variable logique est une variable qui peut prendre deux états ou valeurs: vrai (V) ou faux (F)
- En faisant correspondre V avec le chiffre binaire 1 et F avec 0, ce type de variable devient une variable booléenne ou binaire.

Rappel : Portes logiques

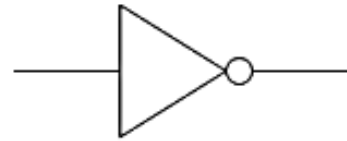
- En électronique les deux états d'une variable booléenne sont associés à deux niveaux de tension: $V(0)$ et $V(1)$ pour les états 0 et 1 respectivement.
- On distingue les logiques **positive** et **négative** selon que $V(1) > V(0)$ ou $V(1) < V(0)$

Niveau	Logique positive	Logique négative
Haut	1	0
Bas	0	1

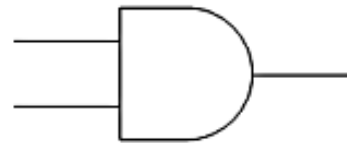
- Toute fonction logique peut être réalisée à l'aide d'un nombre de fonctions logiques de base appelées **portes**.
- Un circuit se représente par un **logigramme**.

Rappel : Réalisation des fonctions booléennes

- Toute fonction logique peut être réalisée à l'aide des portes
- Réalisation d'une fonction booléenne
 1. Écrire l'équation de la fonction à partir de sa table de vérité
 2. Simplifier l'équation
 3. Réaliser l'équation à l'aide des portes disponibles



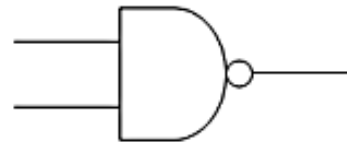
NON (*NOT*)



ET (*AND*)



OU (*OR*)



NON-ET (*NAND*)



NON-OU (*NOR*)

Rappels : Relation d'équivalence des circuits

Soucis majeurs des concepteurs

- Réduire le nombre de portes nécessaires à la réalisation des systèmes
 - Minimiser le coût en nombre de boîtiers
 - La consommation électrique
- Minimiser la complexité
 - Créer un système équivalent avec certains paramètres optimisés
 - Recherche d'équivalence
 - Utiliser les lois et théorèmes de l'algèbre de Boole

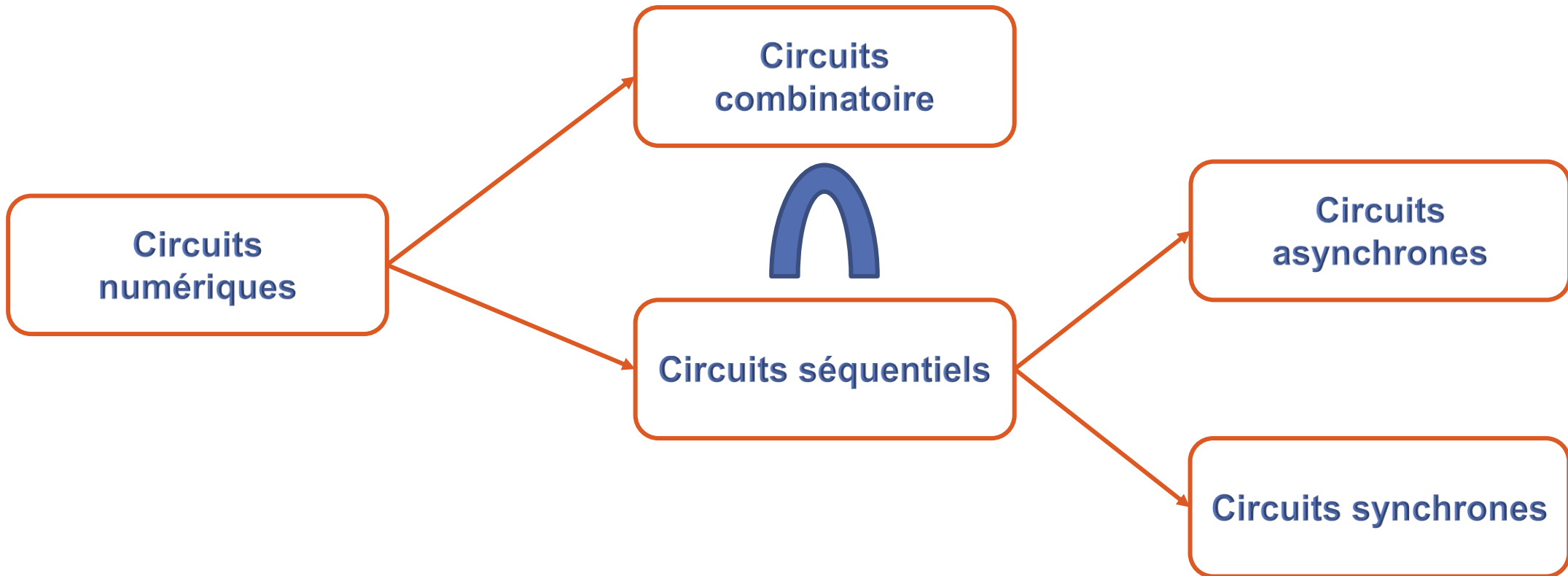
Circuits séquentiels

- Circuits **séquentiels** ou à **mémoire**
- Les fonctions de sortie dépendent non seulement de l'état des variables d'entrée mais également de **l'état antérieur de certaines variables de sortie** (propriétés de **mémorisation**)



Circuits séquentiels

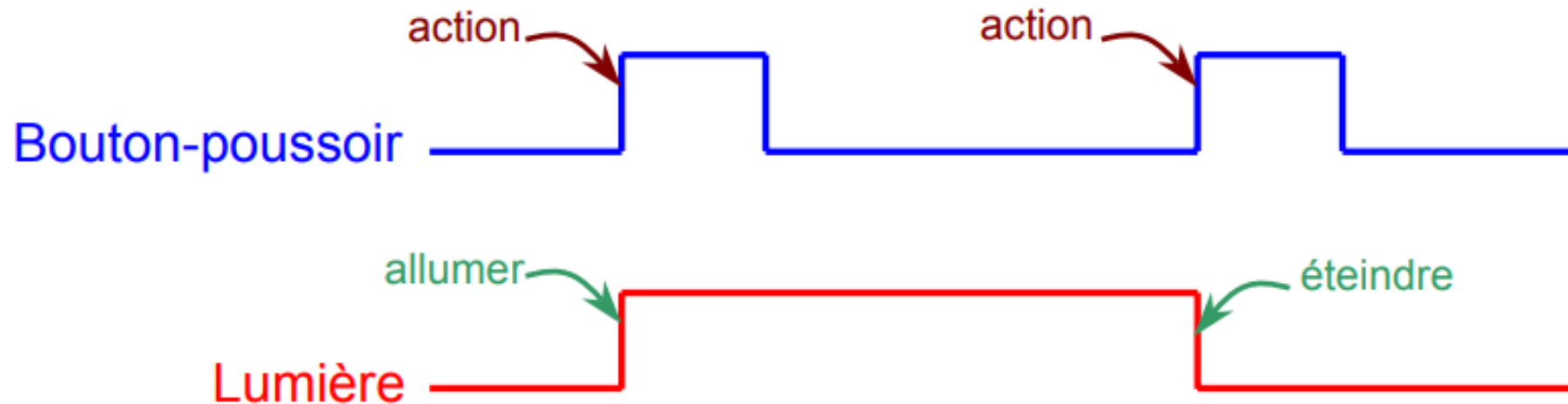
Les circuits séquentiels sont conçus à partir de circuits combinatoires (algèbre de Boole)



Circuits séquentiels

Exemple 1

Lampe électrique commandée par un bouton poussoir



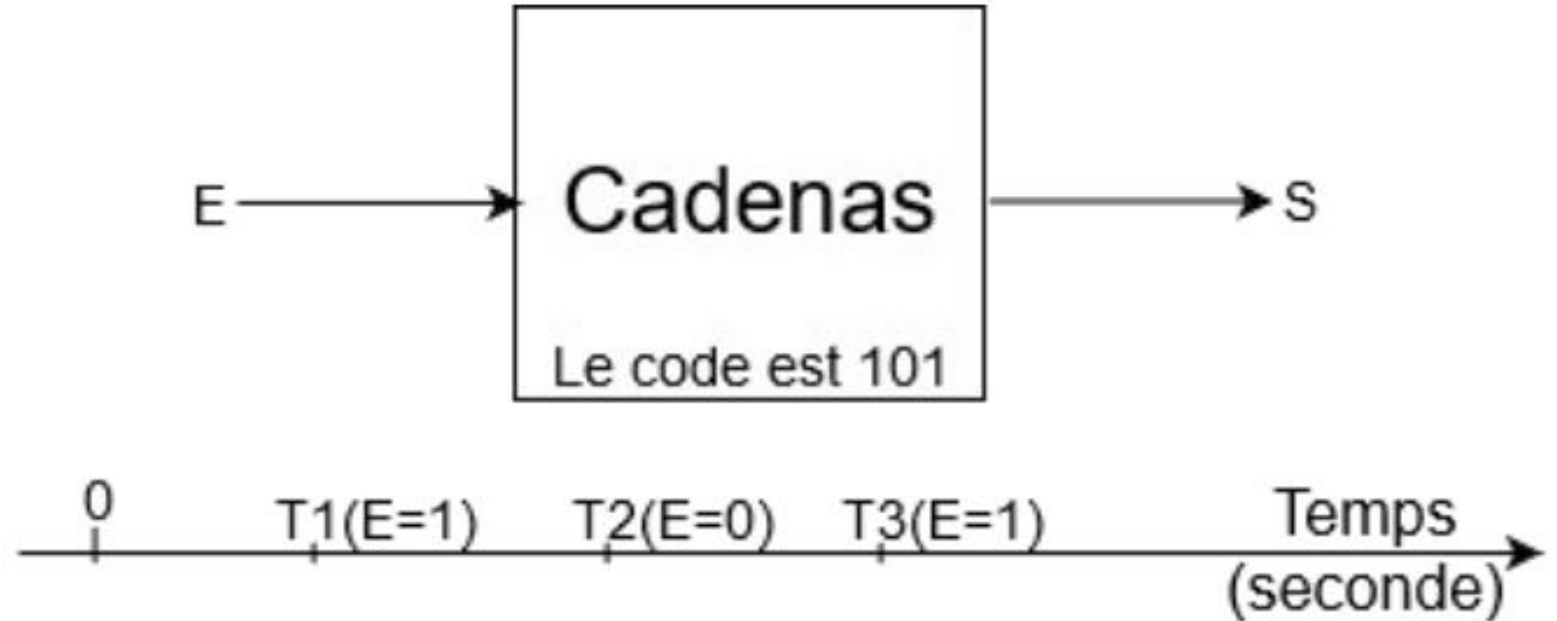
- ➔ L'évolution du système dépend non seulement de la position du bouton poussoir à un instant donné, mais aussi du fait que la lampe soit allumée ou non.
- ➔ Le système dépend donc de l'état précédent car il **conserve la mémoire de l'action précédente** ; c'est la caractéristique essentielle d'un système séquentiel.

Circuits séquentiels

Exemple 2

Cadenas électronique

- Code d'ouverture: 101
 - 1^{ère} entrée: 1
 - 2^{ème} entrée: 0
 - 3^{ème} entrée: 1



- Le Circuit Séquentiel a **une mémoire** pour se rappeler de la séquence précédente.
- Le terme Séquentiel viens en rapport à ce comportement.

Circuits séquentiels

Caractéristiques des Circuits Séquentiels

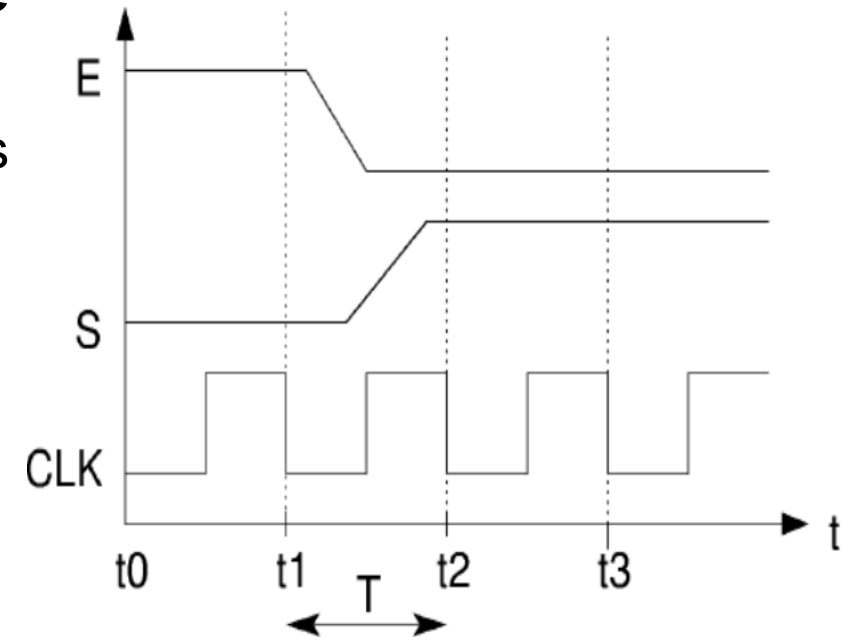
- **Dynamiques:**
 - Evolution dans le temps
- **Boucles:**
 - Le retour d'une partie des sorties vers une partie des entrées
- **Mémoire:**
 - Mémorisation des séquences d'entrées précédentes.
- **Horloge:**
 - généralement une itération (boucle) est effectuée pour chaque impulsion d'horloge.

Circuits séquentiels

Définitions

Horloge: Définition

- un signal numérique périodique à base de 0 et de 1, se caractérisant par une **Période** et une **Fréquence**
- Système logique qui émet régulièrement **une suite d'impulsions** calibrées
- L'intervalle de temps entre 2 impulsions représente le temps de cycle ou **la période** de l'horloge
- L'horloge est notée par **h** ou **ck** (clock).



Circuits séquentiels

Définitions

Circuit synchrone

- Un circuit séquentiel est dit synchrone lorsque la variable horloge est utilisée comme entrée parmi les entrées de ce circuit.
- Tout Circuit Séquentiel doit avoir une entrée d'horloge, souvent nommé Clock(clk).
- Contrairement aux autres entrées, l'entrée du signal d'horloge n'est pas une entrée d'informations ou de données

Circuits séquentiels

Notations

Un circuit séquentiel est un circuit logique dont **la sortie** à l'instant **t+1** (notée **S_{t+1}** ou **S^+**) est une fonction des entrées en même instant t+1 et de **la sortie** précédente à l'instant **t** (notée **S_t** ou **S**) .

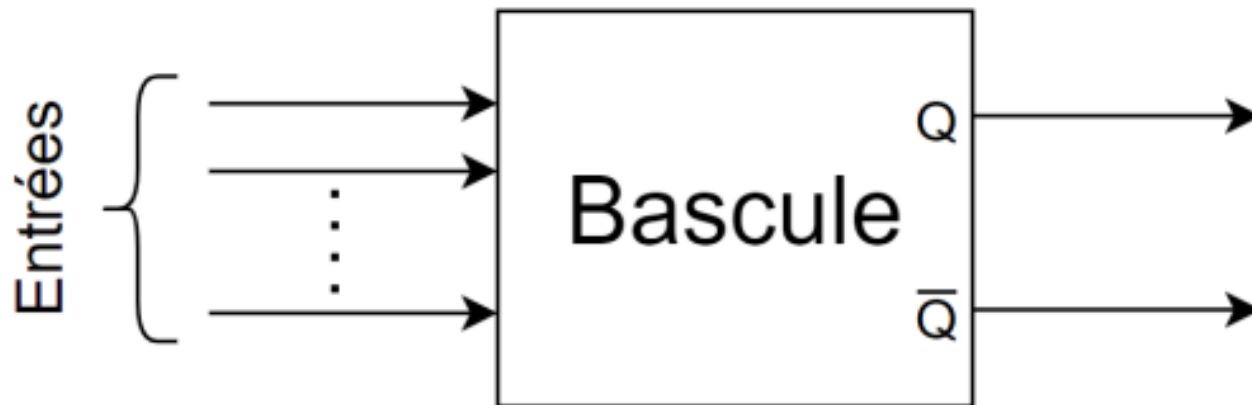
$$S_{t+1} = F(E, S_t)$$

$$S^+ = F(E, S)$$

Circuits séquentiels

Les Bascules

- Technologie de mémoire utilisée dans les Circuits Numériques
- Les bascules sont des circuits à base de la logique séquentielle .
- Une bascule peut posséder une horloge (synchrone) ou non (asynchrone).
- Chaque bascule possède des entrées et deux sorties Q et \overline{Q}
- Une bascule possède la fonction de mémorisation et de basculement.



$$Q^+ = F(E_i, Q)$$

Circuits séquentiels

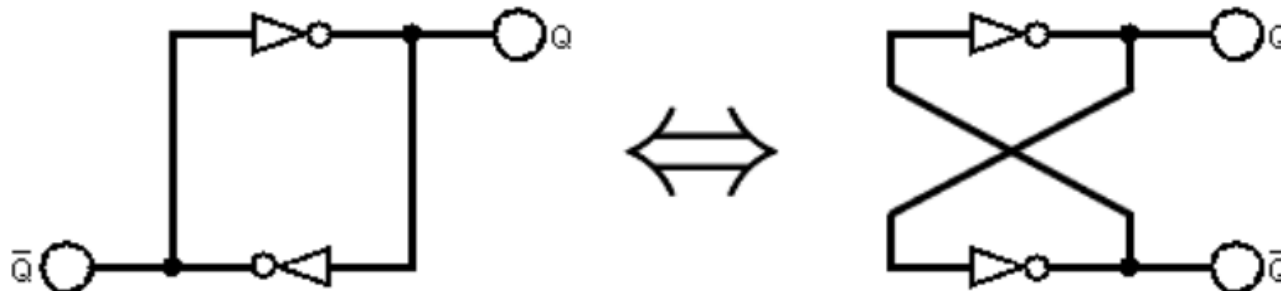
Les Bascules

- Il existe plusieurs types de bascules:
 - Bi-stables
 - RS-Latch
 - D-Latch
 - D-FlipFlop
 - ...
- Ces 4 circuits sont des circuits élémentaires appelés **cellule-mémoire**,
- Ils représentent des circuits pour sauvegarder ou **mémoriser 1 seul bit**, ils peuvent mémoriser la valeur 0 ou 1.
- Les sorties des cellules-mémoires indiquent la valeur sauvegardée à l'intérieur de la cellule, la valeur est dans Q (\bar{Q} est toujours l'inverse de Q).
- Les entrées permettent d'entrer une nouvelle valeur à mémoriser ou de commander la cellule de mémoriser la valeur actuelle.

Circuits séquentiels

Les Bascules : Bi-stable

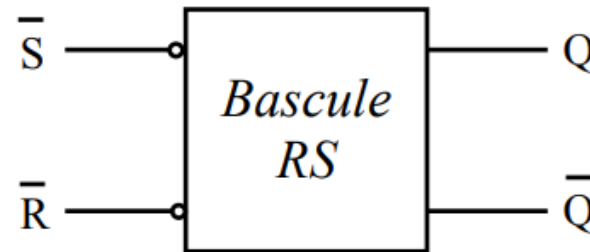
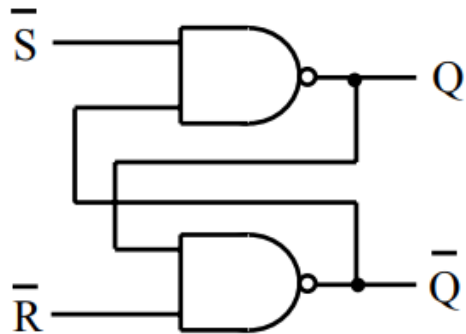
- Le circuit forme un cycle dans lequel la sortie de la 1ère porte NOT arrive à l'entrée de la 2ème porte NOT, et la sortie du 2-ième NOT arrive à l'entrée de la 1ère.
- Le circuit Bi-stable ne contient pas d'entrées, et contient 2 sorties Q et \bar{Q} .
- Le circuit peut suivre 2 scénarios possibles, sur le schéma en bas, le circuit peut se stabiliser sur le Cas 1 : $Q = 0$ et $\bar{Q} = 1$, ou le Cas 2 : $Q = 1$ et $\bar{Q} = 0$, et le signal boucle infiniment d'une façon stable sur l'un des 2 Cas. D'où le nom Bi-stable.
- Ce phénomène de **stabilité permanente** incarne le **comportement de la mémoire**.



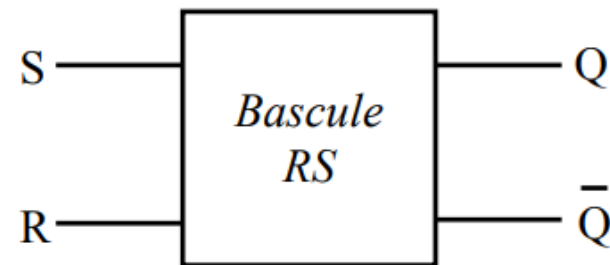
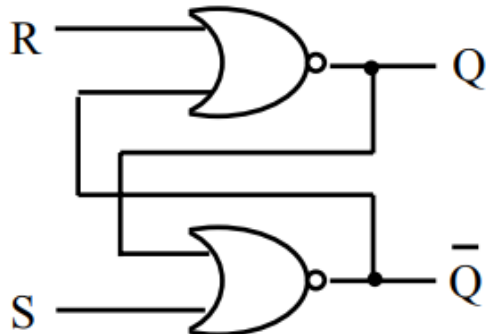
Circuits séquentiels

Les Bascules : RS-Latch

- Le circuit est formé de **2 portes NOR croisées**, tel que la sortie de l'une est l'entrée de l'autre et vice-versa, et former ainsi un cycle.



Bascule RS à portes NAND

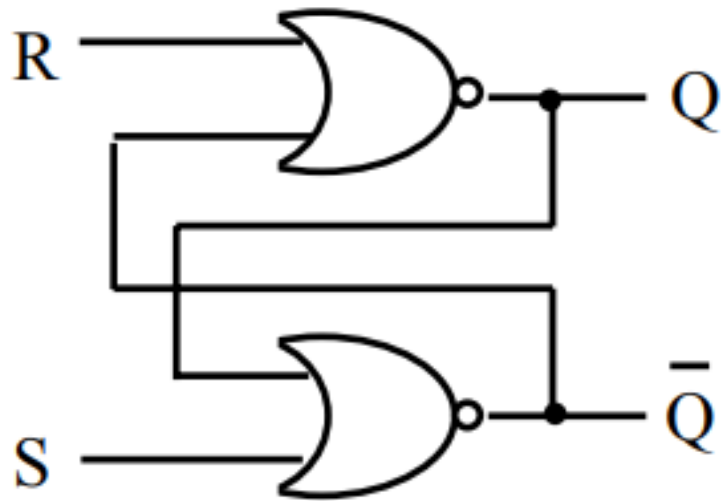


Bascule RS à portes NOR

Circuits séquentiels

Les Bascules : RS-Latch

- La bascule RS dispose de deux entrées R et S et de deux sorties Q et \bar{Q}
- Table de vérité : (cas de **Q=0 initialement**)

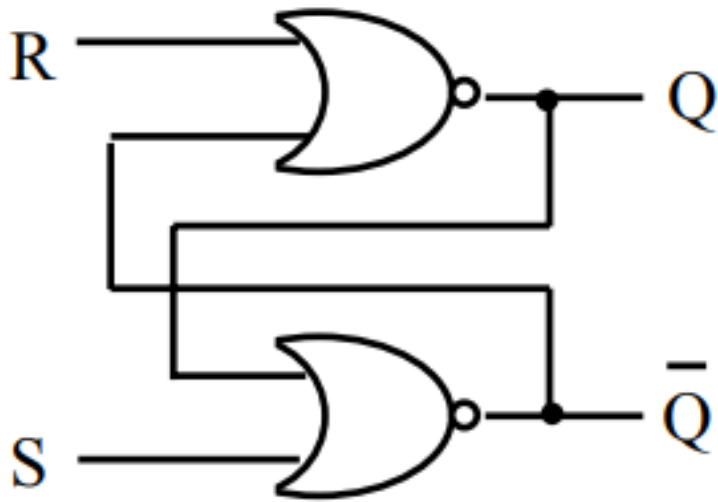


R	S	Q	Q ⁺
0	0	0	0
0	1	0	1
1	0	0	0
1	1	0	X

Circuits séquentiels

Les Bascules : RS-Latch

- Table de vérité : (cas de $Q=1$ initialement)



R	S	Q	Q ⁺
0	0	1	1
0	1	1	1
1	0	1	0
1	1	1	X

↑
Etat incohérent!!

Circuits séquentiels

Les Bascules : RS-Latch

R	S	Q	Q ⁺
0	0	1	1
0	1	1	1
1	0	1	0
1	1	1	X

R	S	Q	Q ⁺
0	0	0	0
0	1	0	1
1	0	0	0
1	1	0	X

R	S	Q ⁺	\bar{Q}
0	0	Q	\bar{Q}
0	1	1	0
1	0	0	1
1	1	X	X

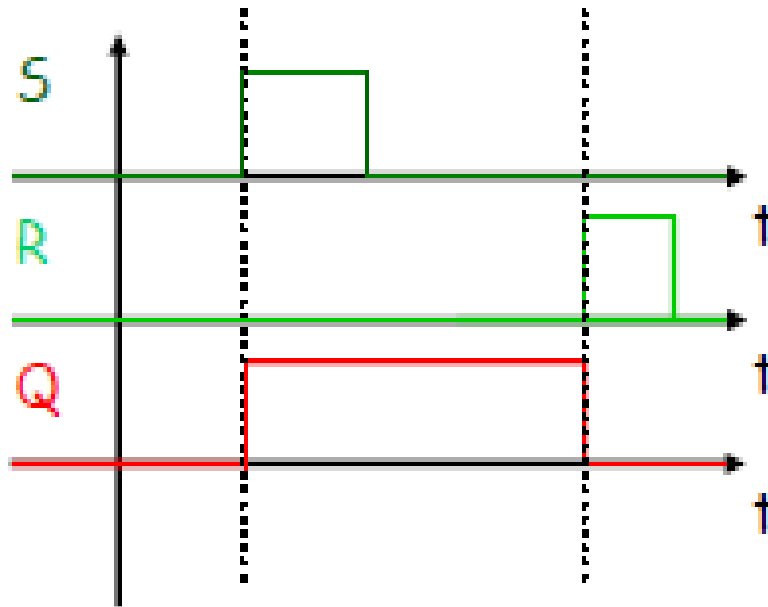
- L'entrée **Mémoriser (00)** conserve **Q**
- L'entrée **mode Set** remet Q à **1**
- L'entrée **mode Reset** remet Q à **0**
- L'entrée **(11)** est interdite

Circuits séquentiels

Les Bascules : RS-Latch

Chronogramme:

- Représente la succession des états logiques en fonction du temps.

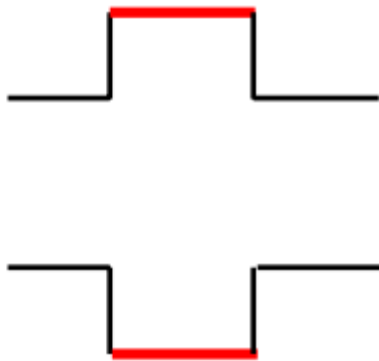


- L'entrée R (**Reset**) est le signal d'effacement de la mémoire ($Q = 0$).
- L'entrée S (**Set**) reçoit l'information à mémoriser ($Q = 1$).

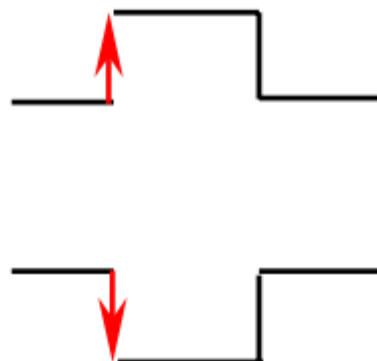
Circuits séquentiels

Les Bascules : RST

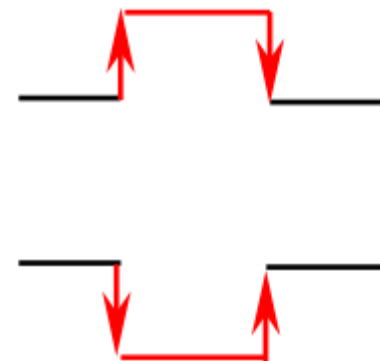
- Dans le cas de la bascule asynchrone RS, le changement d'état de la sortie se produit immédiatement quand les valeurs d'entrée sont changées.
- Il existe d'autres bascules, dites **synchrones**, pour lesquelles le changement d'état de la sortie est cadencé au rythme d'un signal appelé horloge "clock".
- Les bascules synchrones fonctionnent selon l'un des trois modes de synchronisation:



Synchronisation
sur un niveau



Synchronisation
sur un front



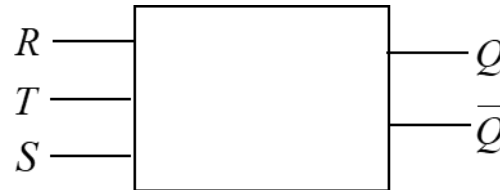
Synchronisation
sur une impulsion

Circuits séquentiels

Les Bascules : RST

- La bascule RST est la synchronisation de la bascule RS par l'adjonction d'un signal **T** (ou H ou **Clk**).
- La bascule RST dispose de trois entrées R, S et T de deux sorties Q et \bar{Q}
 - Si **T=0** l'état de la mémoire est conservée (mémorisée).
 - Si **T=1** la bascule RST est équivalente à la bascule RS

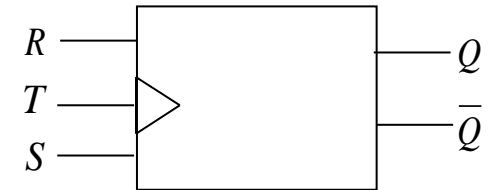
T	R	S	Q ⁺
0	x	x	Q
1	0	0	Q
1	0	1	1
1	1	0	0
1	1	1	X



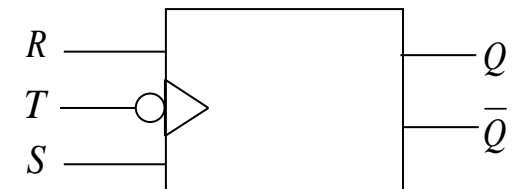
Synchronisation sur niveau haut



Synchronisation sur niveau bas



Synchronisation sur front montant

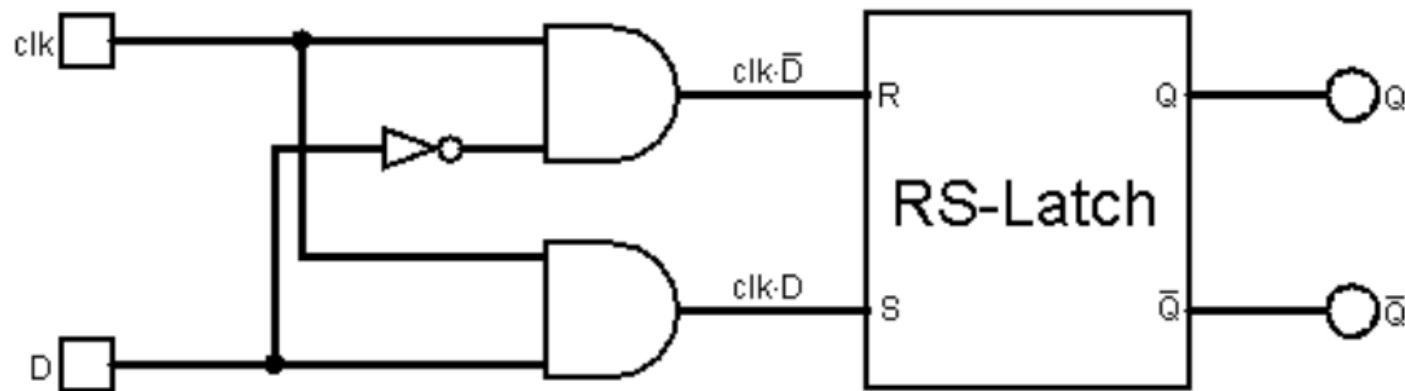


Synchronisation sur front descendant

Circuits séquentiels

Les Bascules : D-Latch

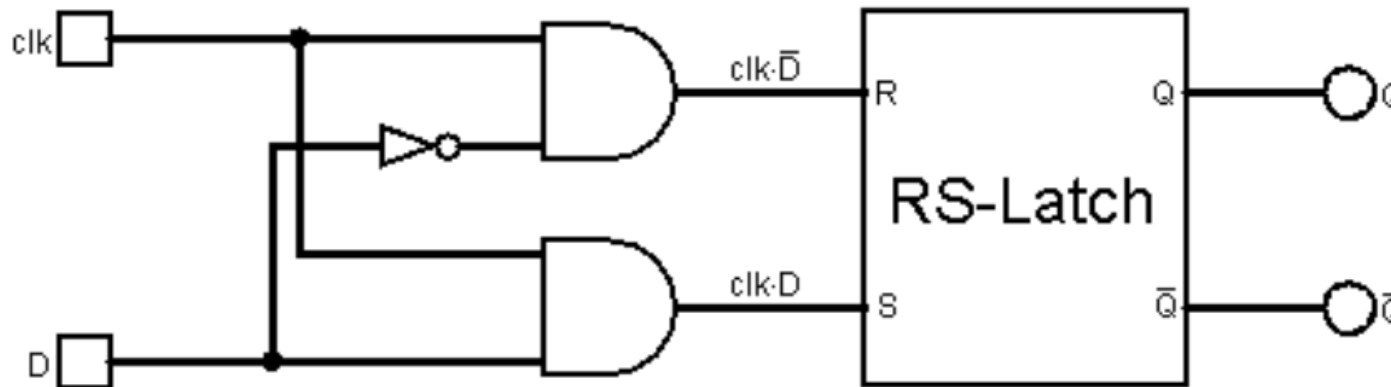
- La D-Latch est **une amélioration de la RS-Latch**, dans le sens où une seule entrée est dédiée pour le changement de valeur de la cellule-mémoire, au-lieu de 2 pour la RS-Latch.
- La 2ème entrée est dédiée pour l'horloge (clk : Clock), entrée indispensable pour tout Circuit Séquentiel Synchrone.
- D-Latch utilise une RS-Latch pour son fonctionnement.



Circuits séquentiels

Les Bascules : D-Latch

Table de vérité:

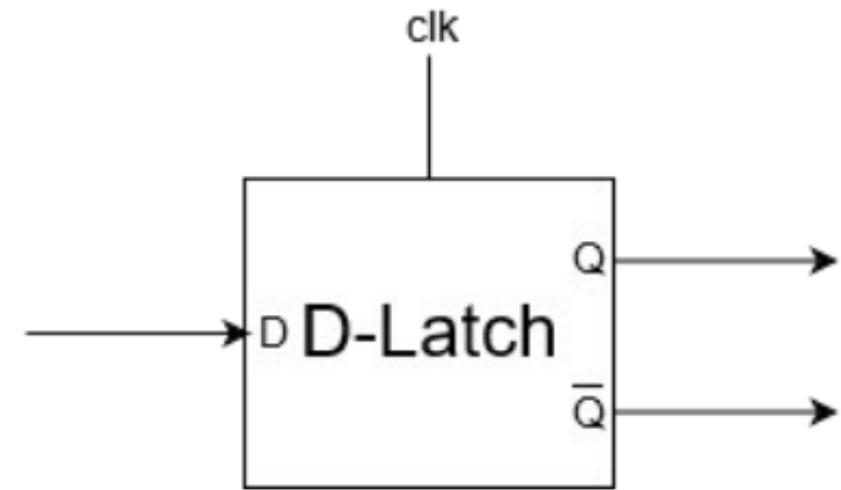


clk	D	Q^+
0	0	Q
0	1	Q
1	0	0
1	1	1

- Si $T=0$ l'état de la sortie est conservée ($Q^+ = Q$).
- Si $T=1$ la valeur de D est copiée sur la sortie ($Q^+ = D$).

Circuits séquentiels

Les Bascules : D-Latch



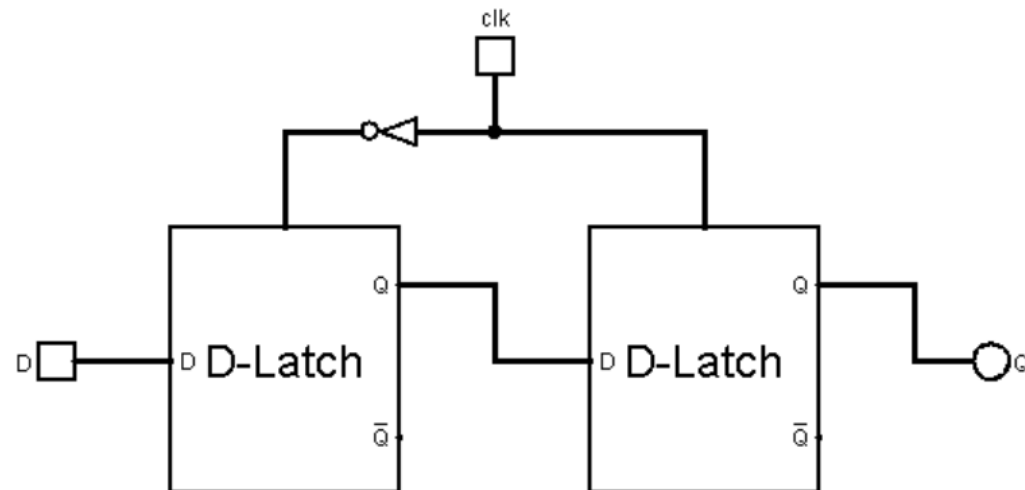
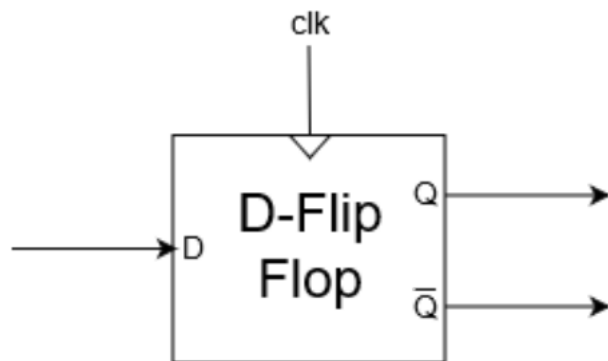
- L'horloge permet de contrôler les **2 modes**:
 - **Mémoriser** si l'horloge est à 0
 - **Changement** de la valeur de la cellule si l'horloge est à 1, dans ce dernier cas c'est l'entrée D qui change directement la valeur de la cellule.
- Lors du mode Mémoriser, la D-Latch est parfois dite **fermée** (ou **opaque**) en raison que la valeur D ne peut pas changer l'intérieur de la cellule.
- Lors du mode Changement, la bascule est dite **ouverte** (ou **transparente**), la valeur de D peut entrer et modifier l'intérieur de la cellule.
- L'avantage de la D-Latch par rapport à la RS-Latch c'est qu'il n'y a pas de risque de tomber dans le cas interdit 11 comme dans la bascule RS-Latch.

Circuits séquentiels

Les Bascules : D-Flip-flop

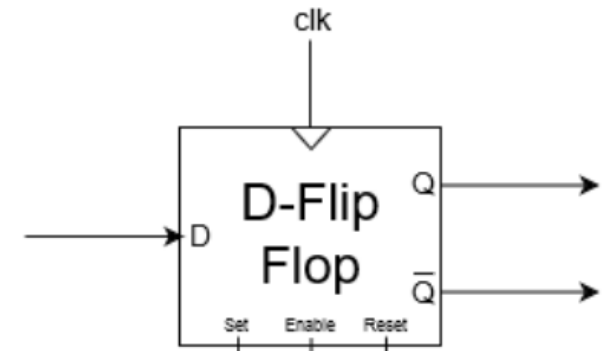
Bascules D sur front:

- Une D-FlipFlop est une Bascule constituée internement par 2 D-Latch mises en série l'une après l'autre, l'une des 2 est rythmée par l'horloge inversée.
- La D-FlipFlop fonctionne d'une manière similaire à la D-Latch, la seule différence c'est que le temps de transparence, de modification de la valeur de la cellule est très réduit, c'est lors du front montant de l'horloge.
- **Sur chaque front montant de T**, la bascule mémorise l'état qu'elle voit sur l'entrée D à cet instant (juste avant le front montant).



Circuits séquentiels

Les Bascules : D-Flip-flop



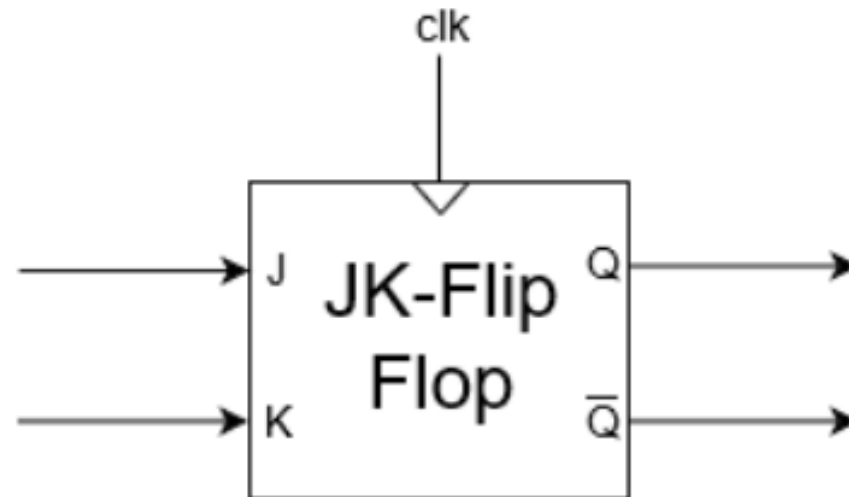
- Les D-FlipFlop peuvent disposer des entrées de commandes additionnelles:
 - Set
 - Reset
 - Enable
- **Enable:** permet avec 1 d'**activer** ou avec 0 de **désactiver** une FlipFlop.
 - Si la FlipFlop est activée elle fonctionne normalement, dans le cas contraire, la désactiver implique la conservation de la donnée.
- **Set** et **Reset** s'ils sont mis à 1, permettent respectivement de modifier la valeur interne de la cellule à 1 et à 0 (Set à 1 et Reset à 0).

Circuits séquentiels

Les Bascules : Autres types de FlipFlop

JK-FlipFlop :

- Semblable à l'RS-FlipFlop dans sa logique d'utilisation, mais en plus elle offre une correction du cas interdit (11) en inversion le contenu sauvegardé de la cellule.

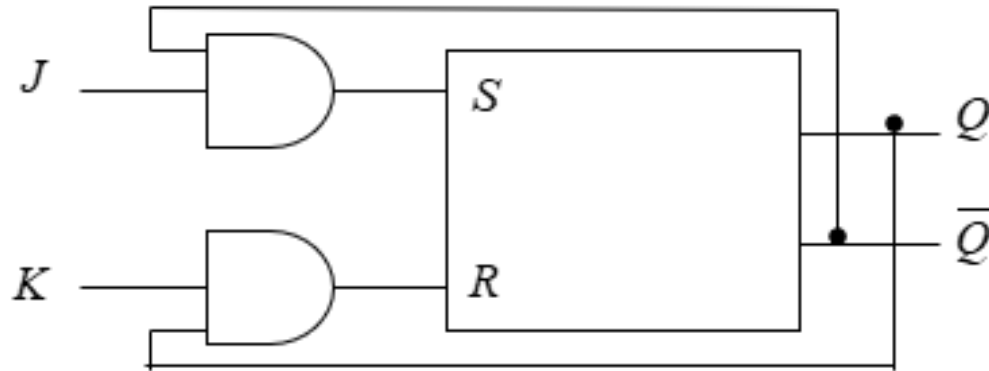


Circuits séquentiels

Les Bascules : Autres types de FlipFlop

JK-FlipFlop :

- La bascule JK peut être obtenu à partir de la bascule RS tel que $S = J.Q$ et $R = K.Q$.
Donc on a éliminé le cas $S=R=1$.



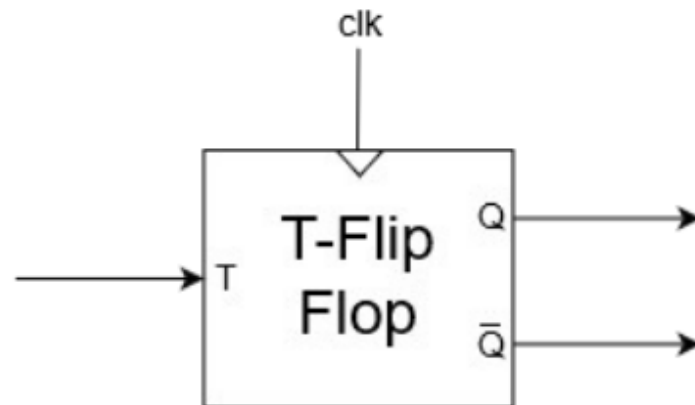
J	K	Q^+
0	0	Q
0	1	0
1	0	1
1	1	\bar{Q}

Circuits séquentiels

Les Bascules : Autres types de FlipFlop

T-FlipFlop :

- T pour Trigger (déclencheur).
- Elle a une seule entrée, si elle est mise à 1 elle va inverser la valeur sauvegardée.
- Pour contrôler le contenu initial de la T-FlipFlop, les entrées Set et Reset sont généralement utilisées.

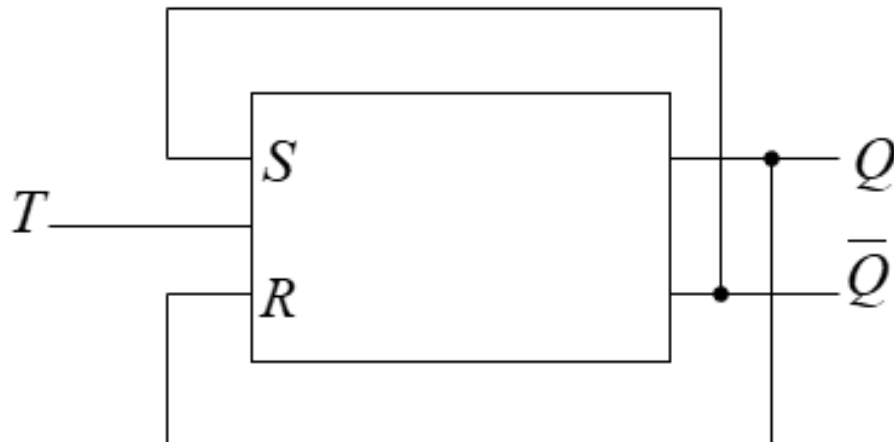


Circuits séquentiels

Les Bascules : Autres types de FlipFlop

T-FlipFlop :

- Si **T=0** l'état de la mémoire est conservée (mémorisée).
- Si **T=1** le complément de l'état de la mémoire est conservée.
- La table de vérité n'est applicable que lors du **front montant de l'horloge**, le reste de la période la FlipFlop sauvegarde sa valeur.



T	Q^+
0	Q
1	\bar{Q}

Circuits séquentiels

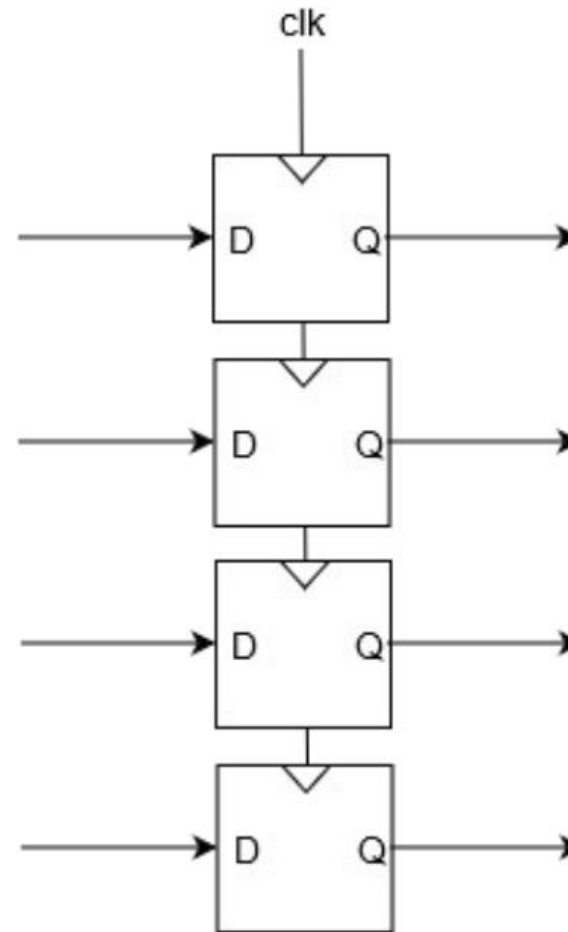
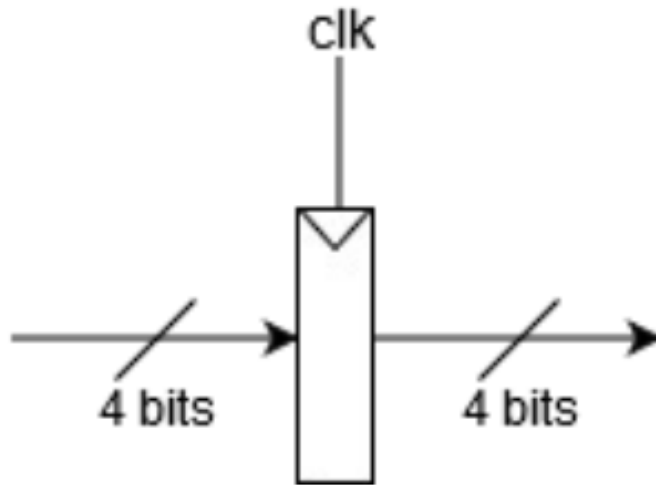
Les registres

- Les Registres représentent un moyen de mémoriser **un mot d'information** (ou mot mémoire) sur n bits; 4 bits, 8 bits, 16 bits, 32 bits...etc, en utilisant un tableau de plusieurs D-FlipFlop synchronisées avec le même signal d'horloge.
- Les registres sont situés dans le processeur (sont regroupés sur un banc de registres) et constituent donc une mémoire locale au processeur.
- Les registres permettent à mémoriser les informations manipulées directement par l'UAL.

Circuits séquentiels

Les registres

Exemple : un Registre de 4 bits

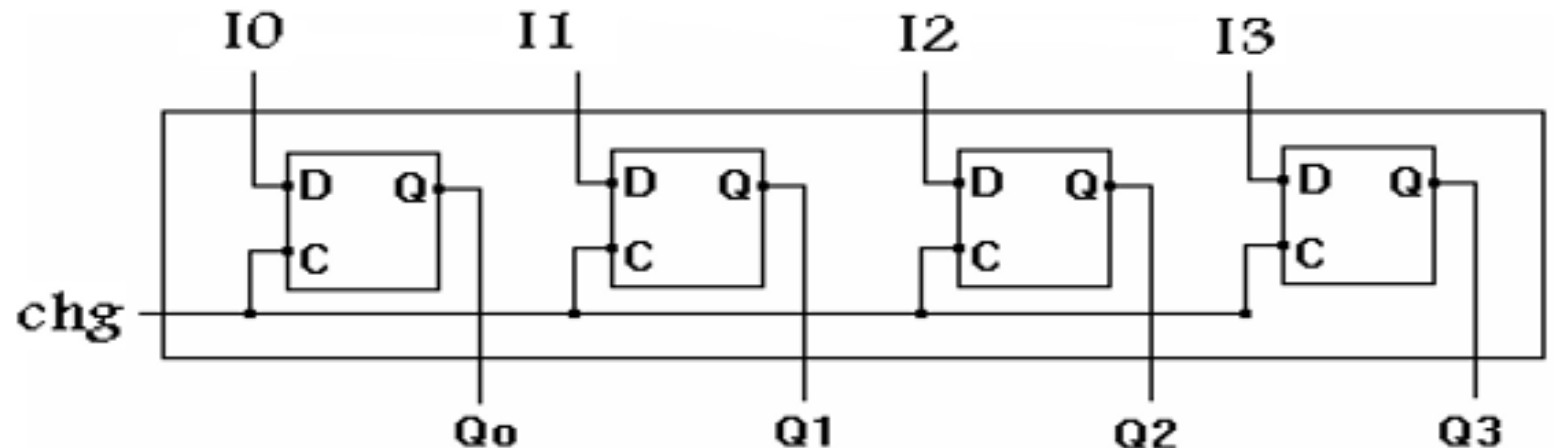


Circuits séquentiels

Les registres

Registre à entrées parallèles et sorties parallèles

- Permet de charger une information sur n bits en même temps.
- Les n bascules changent d'état en même temps.
- Chaque **bascule** B_i prend la valeur de **l'information** I_i .
- chg est l'entrée de chargement:
 - chg=0 → état mémoire
 - chg=1 → chargement

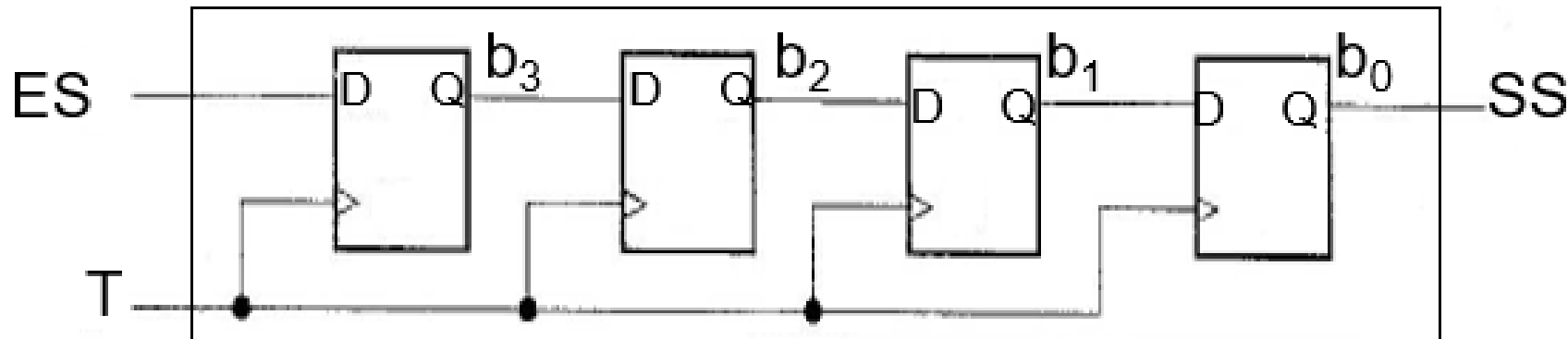


Circuits séquentiels

Les registres

Registre à décalage: (entrée série à gauche)

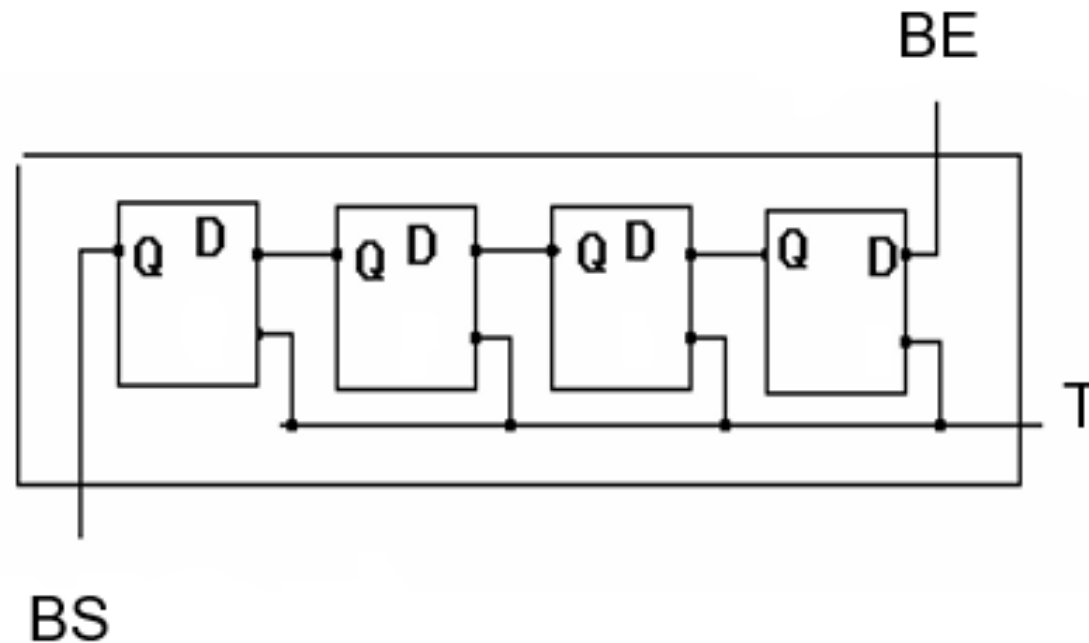
- L'information est introduite bit par bit (**en série**).
- Après 4 activations d'horloge, les 4 bits sont entrées dans le registre (mémorisation).
- À chaque nouvelle entrée **ES**, les bits mémorisés dans le registre sont décalés d'une **position b_i à b_{i-1}** et le bit b_0 sera décalé vers la sortie **SS** (décalage à droite)



Circuits séquentiels

Les registres

Registre à décalage: (entrée série à droite)



Circuits séquentiels

Les registres

Registre à décalage sorties parallèles:

Lorsque l'entrée est stockée, chaque **bit apparaît simultanément** sur les lignes de sortie.

Le registre à décalage est utilisé comme convertisseur série/parallèle. Il est nécessaire à la réception lors d'une transmission série.

