

Rappel de la logique combinatoire

Circuits combinatoires

- ✚ Les sorties ne dépendent que de l'état des entrées.
- ✚ Elles ne dépendent pas de leur propre état antérieur.
- ✚ Il n'existe pas de mémoire du passé (de l'état antérieur du système).
- ✚ Dans la conception des circuits combinatoires : pas de re-bouclage des entrées vers les sorties.

Circuits séquentiels, propriétés générales

- ✚ Les sorties ne dépendent pas que de l'état des entrées.
- ✚ Les sorties dépendent aussi de leur propre état antérieur, c'est-à-dire de l'état du système avant le changement de l'état des entrées.
- ✚ Il existe une mémoire du passé (de l'état antérieur du système).
- ✚ Dans la conception de circuits séquentiels : re-bouclage des sorties vers les entrées.

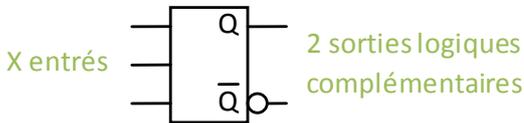


Allez Tous vous faire foutre

Les bascules

Définition générale

Une bascule (flip flop en anglais) a pour rôle de mémoriser une information élémentaire.



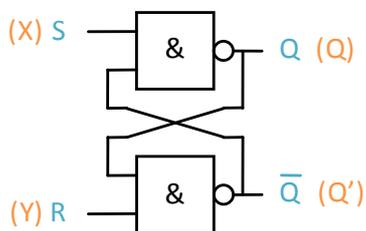
Elle possède :

- ✚ Deux sorties complémentaires : Q et \bar{Q} (les deux sorties ne doivent pas être dans le même état)
- ✚ Le système interne de « verrouillage » (basé sur un circuit de rétroaction des sorties vers les entrées).
- ✚ Une ou plusieurs entrées qui permettent de choisir :
 - Soit de charger la sortie à la valeur souhaitée (0 : mise à zéro ou 1 : mise à un).
 - Soit de conserver son état antérieur (mémorisation de l'état de Q).
 - Soit de faire basculer la sortie (toggle).

Les bascules RS

Bascule RS à NAND :

Elle est constituée de 2 portes NAND dont les deux sorties Q et \bar{Q} sont rebouclées vers les entrées :



Remarque : pour la bascule RS à NAND, l'entrée nommée S correspond à la porte NAND dont la sortie est Q .

Rappel porte NAND : Un 0 en entrée « ferme la porte à 1 ». Un 1 sur l'une des entrées rend la porte à 0.

Mise à zéro d'une seule des entrées à la fois. (RA1 = remise à 1)

| Action | S | R | Q | Q' = \bar{Q} | Fonction |
|--------------------|---|---|---|----------------|----------------|
| Hypothèse initiale | 1 | 1 | 0 | 1 | |
| S passe à 0 | 0 | 1 | 1 | 0 | RA1 ou bascule |
| S revient à 1 | 1 | 1 | 1 | 0 | Rien ou RA1 |
| S passe à 0 | 0 | 1 | 1 | 0 | RA1 ou Rien |
| S revient à 1 | 1 | 1 | 1 | 0 | Rien ou RA1 |
| R passe à 0 | 1 | 0 | 0 | 1 | RAZ ou bascule |
| R revient à 1 | 1 | 1 | 0 | 1 | Rien ou RAZ |
| R passe à 0 | 1 | 0 | 0 | 1 | RAZ ou Rien |

| R | S | Q_n | Fonction |
|---|---|-----------|----------|
| 0 | 0 | | |
| 0 | 1 | 0 | RAZ |
| 1 | 0 | 1 | RA1 |
| 1 | 1 | Q_{n-1} | Rien |

Mise à 0 des deux entrées à la fois :

| Action | S | R | Q | Q' | Remarque |
|--------------------|---|---|---|----|-----------|
| Hypothèse initiale | 0 | 1 | 1 | 0 | |
| R passe aussi à 0 | 0 | 0 | 1 | 1 | Q = Q' !! |

Puis retour à la combinaison (1; 1) en entrée : (simultanéité impossible).

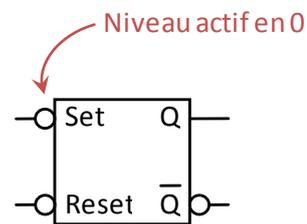
1er chemin :

| Action | S | R | Q | Q' | Remarque |
|--------------------|---|---|---|----|----------|
| Hypothèse initiale | 0 | 0 | 1 | 1 | Q = Q' |
| S revient à 1 | 1 | 0 | 0 | 1 | |
| R revient à 1 | 1 | 1 | 0 | 1 | |

2ème chemin :

| Action | S | R | Q | Q' | Remarques |
|--------------------|---|---|---|----|-----------|
| Hypothèse initiale | 0 | 0 | 1 | 1 | Q = Q' |
| R revient à 1 | 0 | 1 | 1 | 0 | |
| S revient à 1 | 1 | 1 | 1 | 0 | |

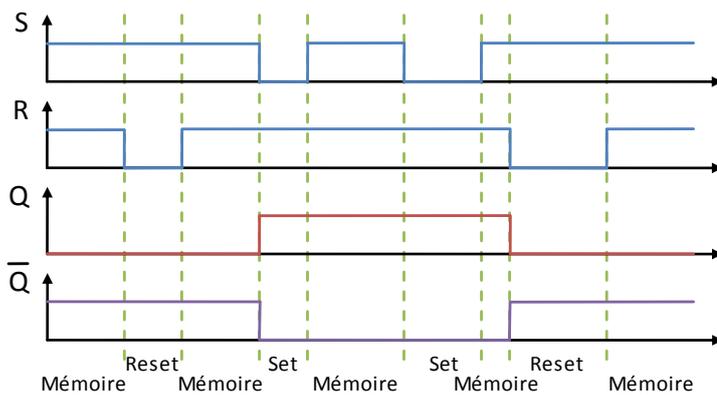
En résumé :



Symbole de la bascule RS à NAND

- ✚ Niveau actif des entrées : niveau bas.
- ✚ « S » = Set = mise à 1 de la bascule.
- ✚ « R » = Reset = mise à zéro de la bascule.

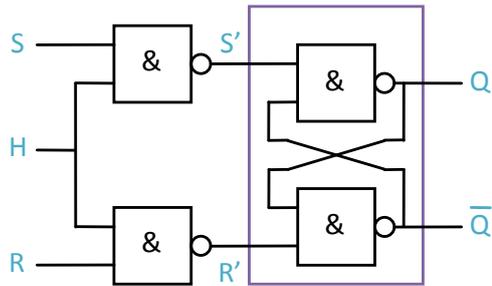
Exemple de chronogrammes avec la bascule RS à NAND :



Bascule RS à NAND avec horloge :

On ajoute une entrée d'horloge H au circuit de base de la mémoire RS. En jouant sur cette entrée d'horloge, on pourra :

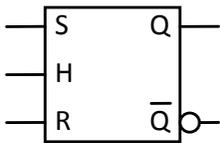
- ✚ Soit imposer la fonction mémoire.
- ✚ Soit avoir accès au fonctionnement de type RS.



Si $H = 0$, alors $S' = R' = 1$: la mémoire RS est bloquée dans sa fonction mémoire.

Si $H = 1$, alors $S' = \bar{S}$ et $R' = \bar{R}$.

On retrouve le fonctionnement de type RS avec cette fois-ci :



- ✚ Un niveau actif à 1.
- ✚ Une combinaison interdite (1; 1).
- ✚ Une combinaison mémoire (0; 0).

| H | R | S | Q_{n+1} | Fonction |
|----------------|---|---|-----------|----------|
| Niveau actif 1 | 0 | 0 | Q_n | Mémoire |
| | 0 | 1 | 1 | RA1 |
| | 1 | 0 | 0 | RAZ |
| | 1 | 1 | ? | Interdit |
| Niveau 0 | X | X | Q_n | Mémoire |

Notations utilisées :

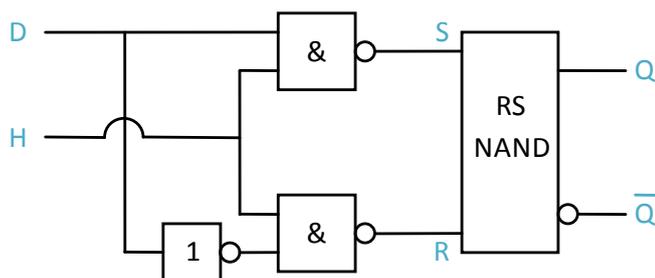
- ✚ Q_n = état de la sortie avant le passage au niveau active de l'horloge.
- ✚ Q_{n+1} = état de la sortie après le passage niveau actif de l'horloge.
- ✚ R et S ne sont prises en considération que lorsque $H = 1$ (activé).
- ✚ Entrées synchrones (à l'horloge).

Bascule D à verrouillage « data latch » (active sur niveau d'horloge)

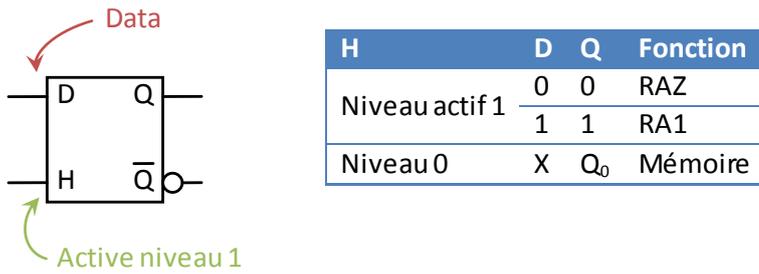
Pour la mémoire RS avec horloge précédente on peut remarquer que :

- ✚ La combinaison $S = R = 1$ est « interdite », « indésirable ».
- ✚ La combinaison $S = R = 0$ est redondante puisqu'on possède déjà la fonction mémoire en imposant $H = 0$.
- ✚ On peut imposer $R = \bar{S}$ et n'utiliser qu'une seule entrée D (S prend le nom de « D » = dopée) tout en continuant de posséder :
 - La fonction mémoire (grâce à l'entrée H)
 - Les fonctions de mise à 0 ou de mise à 1 de la sortie Q

Câblage de la bascule D :



En supprimant les 2 lignes S = R de la table précédente on obtient pour la bascule D :



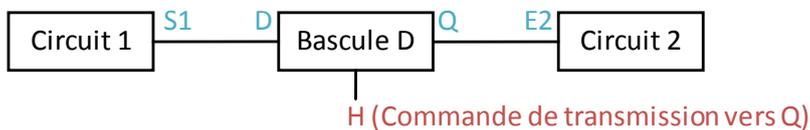
Transmission de la valeur de D vers la sortie Q pendant le niveau d'horloge H = 1.

Sinon : mémoire de la dernière valeur de Q.

Exemple d'utilisation de la bascule D :

On souhaite transmettre, pas en permanence, mais sous l'action d'une commande uniquement, la sortie S1 d'un circuit 1 vers l'entrée E2 d'un circuit 2.

On insère une bascule D entre les deux circuits 1 et 2 :



Bascule JK (active sur niveau d'horloge)



Comme la bascule RS avec horloge, la bascule JK possède :

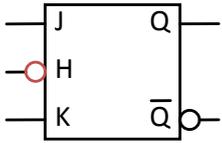
- ✚ 2 entrées (nommées J et K)
- ✚ Les fonctions de mise à 1 et de mise à 0
- ✚ La fonction mémoire

Contrairement à la mémoire RS, la combinaison (1; 1) n'est pas interdite mais correspond à une fonction supplémentaire :

- ✚ Le « Toggle » c'est-à-dire la bascule de l'état de sortie vers son état complémentaire.

Bascules actives sur niveau bas de l'horloge

Toutes les bascules que nous venons de voir existent dans leur version « actives sur niveau bas ». On indique alors sur les schémas un « petit cercle » supplémentaire sur l'entrée d'horloge.

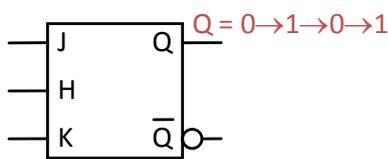


Problèmes liés au déclenchement des bascules sur niveau d'horloge

Le déclenchement des bascules sur niveau d'horloge peut parfois poser quelques problèmes.

✚ Problème d'oscillation des sorties :

Par exemple, prenons une bascule JK en fonction Toggle telle que $Q = 0$.



- Lorsque H passe à 1, la sortie Q doit basculer à 1.
- Si H reste à 1, la sortie Q doit basculer à 0.
- Si H reste encore à 1, la sortie Q doit basculer à 1.

Pour palier à ce problème, on a recours à des câblages plus complexes que celui présenté ici...

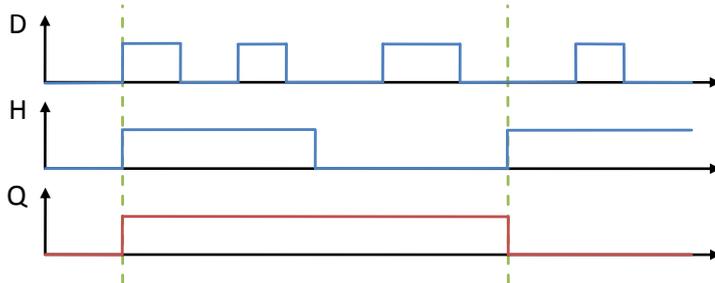
Exemple : Câblage de plusieurs bascules en cascade dit en « maître esclave ».

✚ Problème de sensibilité aux parasites.

Bascules actives sur FRONTS d'horloge

Pour palier aux problèmes rencontrés lors d'un déclenchement sur niveau d'horloge... on utilise des bascules déclenchées par FRONTS d'horloge.

Exemple pour un déclencheur d'une bascule D sur front MONTANT :

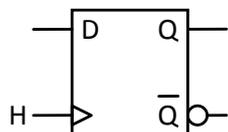


Au moment du passage d'un front montant de H (H passe de 0 à 1) : Transmission de D vers Q.

En dehors des fronts montants de H : La valeur de D n'est plus prise en compte. La bascule est en fonction « mémoire »

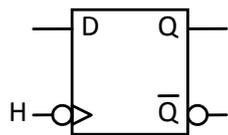
Symboles des bascules D déclenchées par fronts d'horloge

Fronts montant



| H | D | Q | Fonction |
|----------------------|---|---|-----------------|
| | 0 | 0 | RAZ |
| | 1 | 1 | RA1 |
| En dehors des fronts | | | X Q_0 Mémoire |

Fronts descendant



| H | D | Q | Fonction |
|----------------------|---|---|--------------------------|
| | 0 | 0 | RAZ |
| | 1 | 1 | RA1 |
| En dehors des fronts | | | X Q ₀ Mémoire |

Circuit détecteur de fronts descendants

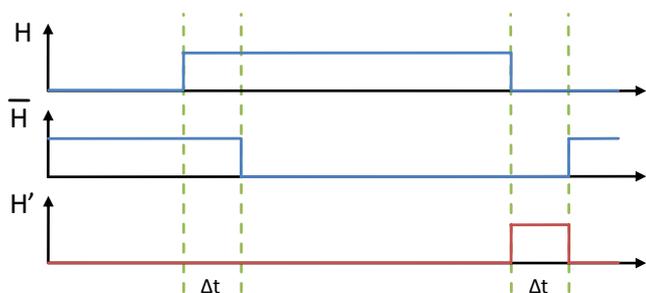
Considérons le circuit suivant :



On tient toujours compte du fait que :

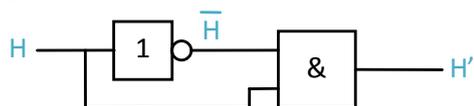
- Un circuit numérique engendre un petit retard du signal de sortie par rapport au signal.
- Δt = quelques nanosecondes.

Rappel : Porte NOR : sortie à 1 si, et seulement si les deux entrées sont à 0.

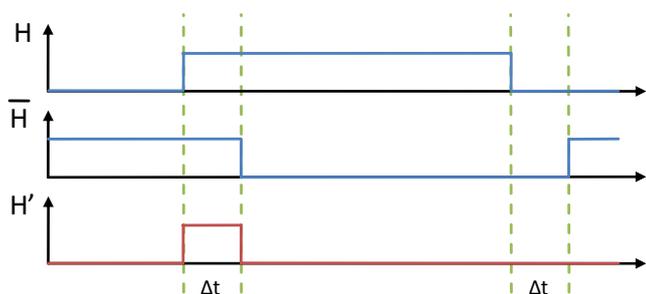


Au moment du front DESCENDANT de H, apparition sur H' du niveau 1 de très courte durée.

Circuit détecteur de fronts montants

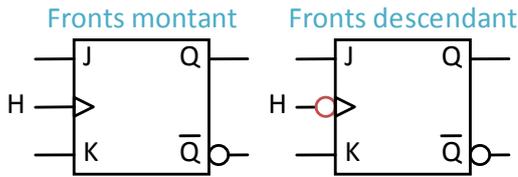


Rappel : Porte AND : sortie à 1 si, et seulement si les deux entrées sont à 1.



Au moment du front montant de H, apparition sur H' de niveau 1 de très courte durée Δt .

Symboles des bascules JK déclenchées par fronts d'horloge



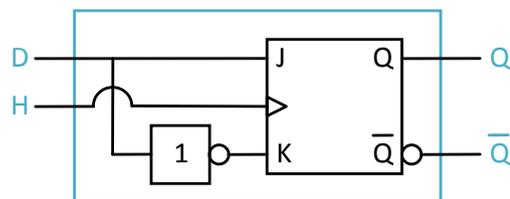
| H | J | K | Q | Fonction |
|--------------|---|---|------------------|----------|
| ↑ ou ↓ | 0 | 0 | Q_0 | Mémoire |
| | 0 | 1 | 0 | RAZ |
| | 1 | 0 | 1 | RA1 |
| | 1 | 1 | $\overline{Q_0}$ | Toggle |

En dehors des fronts d'horloge, les sorties ne sont pas affectées par les valeurs de J et K. La bascule JK est en fonction mémoire. Q conserve la dernière valeur qu'il a prise.

La bascule JK câblée en bascule D

Si on active la bascule JK lorsque $K = \overline{J}$ (voir les deux lignes centrales de la table de fonctionnement) :

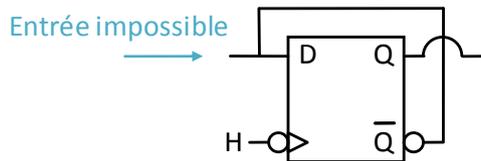
- ✚ La valeur de J est transférée vers la sortie Q.
- ✚ Même type de fonctionnement qu'une bascule D.
- ✚ On peut dire fabriquer une bascule D à partir d'une bascule JK en imposant $K = \overline{J}$.



La bascule D en fonction Toggle

Étudions un câblage particulier de la bascule D

La sortie \overline{Q} est envoyée sur l'entrée D de la bascule :



Si impulsion d'horloge, D est transférée vers Q.

| | Q | \overline{Q} | D |
|-----------------------------------------------|---|----------------|---|
| Hypothèse initiale | 0 | 1 | 1 |
| Après la 1 ^{ère} impulsion d'horloge | 1 | 0 | 0 |
| Après la 2 nd impulsion d'horloge | 0 | 1 | 1 |
| Après la 3 ^{ème} impulsion d'horloge | 1 | 0 | 0 |
| Après la 4 ^{ème} impulsion d'horloge | 0 | 1 | 1 |

En conclusion :

- ✚ À chaque impulsion d'horloge, la sortie Q bascule vers son état complémentaire.
- ✚ On fabrique ainsi la fonction Toggle avec une bascule D.

Attention : L'entrée D de la bascule n'est, bien sur, plus disponible.

Les entrées synchrones des bascules

Une entrée de bascule est dite « synchrone » à l'horloge si son état logique n'est pris en compte en sortie que lorsque la bascule est activée par l'horloge.

- ✚ Pour la bascule D : l'entrée D est une entrée synchrone

Pour la bascule JK : les deux entrées J et K sont des entrées synchrones. Par exemple pour une bascule D activée sur fronts montants d'horloge :

- ✚ La sortie Q de la bascule ne prend la valeur de D qu'au moment du passage d'un front montant d'horloge .
- ✚ En dehors des fronts montants d'horloge, le changement de l'état de D ne provoque aucun changement de la sortie Q.

Les entrées asynchrones des bascules

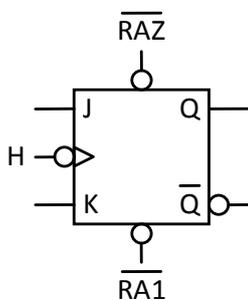
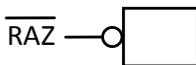
Une entrée de bascule est dite « asynchrone » à l'horloge si, à tout moment, son état logique est pris en compte en sortie, indépendamment de l'état de l'horloge, indépendamment des entrées synchrones.

- ✚ Ces entrées sont donc prioritaires.
- ✚ Elles permettent, à tout moment, de forcer la mise à 0 ou la mise à 1 des sorties des bascules.
- ✚ Elles peuvent :

- Etre actives sur niveau 1



- Etre actives sur niveau 0



| \overline{RAZ} | $\overline{RA1}$ | Q | Fonction |
|------------------|------------------|-----|--------------------|
| 0 | 0 | /// | Interdit |
| 0 | 1 | 0 | RAZ |
| 1 | 0 | 1 | RA1 |
| 1 | 1 | | Fonction synchrone |

Lorsque \overline{RAZ} et $\overline{RA1}$ sont désactivés, fonctionnement classique d'une bascule JK.

Dénomination (non normalisée) :

- ✚ RA1 / RAZ
- ✚ SET / RESET
- ✚ PRESET / CLEAR (CLR)
- ✚ DC SET / DC CLEAR
- ✚ SD (direct Set) / CD (direct Clear)

Les applications les plus courantes des bascules

Stockage de données :

- ✚ Bascule = circuit électronique pouvant enregistrer, conserver, et restituer une information d'un bit.
- ✚ Dans un groupement de bascules (appelé « registre ») on peut stocker une information de plusieurs bits (mot binaire).

Transferts de données :

- ✚ Echange de données d'un registre à l'autre.
- ✚ Ex : stockage (parallèle) d'une information de plusieurs bits puis envoi de cette information sur le port d'une imprimante (transfert série).

Comptage d'impulsion

- ✚ Compteurs = associations de bascules.
- ✚ Ex : comptage du temps qui passe et affichage de l'heure.
- ✚ Ex : commande d'une action mécanique toutes les x impulsions d'une horloge de référence.

Division de la fréquence

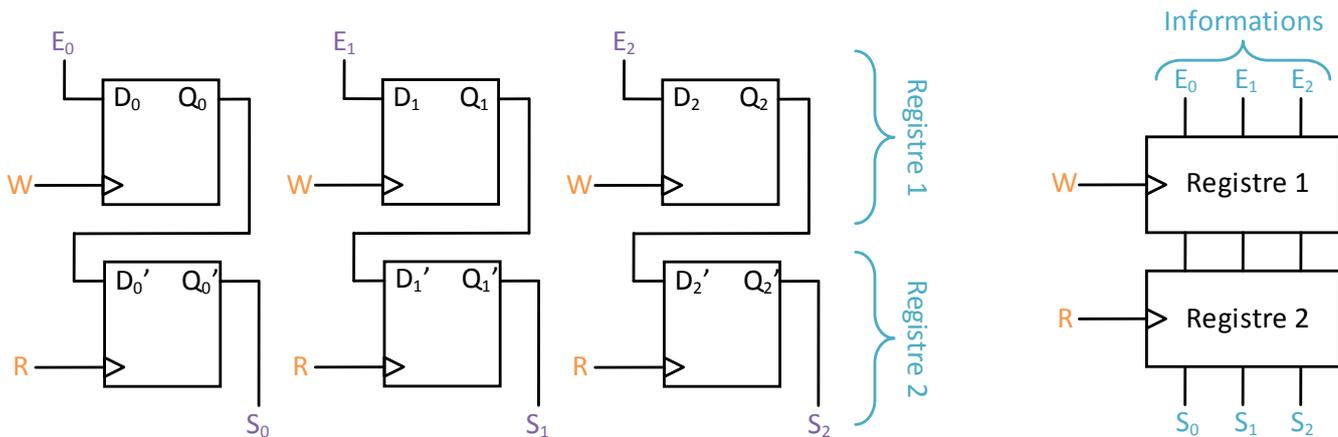
- ✚ À partir d'un signal TTL, obtenir des signaux de fréquence moindre : possible en réalisant un compteur avec des bascules.

Les registres : stockage et transfert de données

Registre = ensemble de n bascules, synchronisées sur la même horloge, qui permet de stocker momentanément une information de n bits.

Stockage et transfert parallèle

Exemple avec deux registres de 3 bits :



Stockage de l'information :

L'horloge W (Write) (envoyée sur toutes les bascules du registre 1) commande l'écriture des données E_0, E_1, E_2 sur les sorties Q_0, Q_1, Q_2 du registre.

Transfert de l'information :

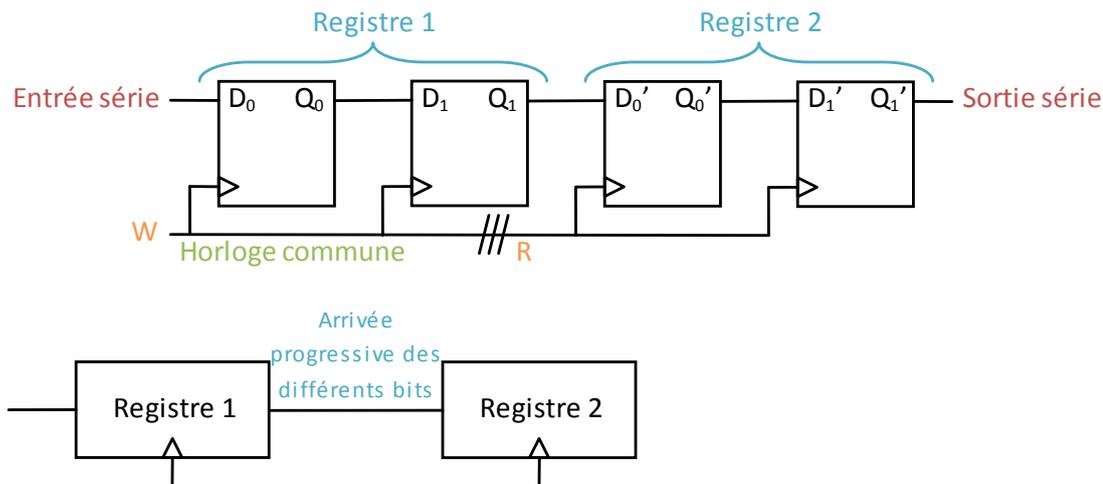
L'horloge R (Read) (envoyée sur toutes les bascules du registre 2) commande la lecture des données Q_0, Q_1, Q_2 stockées en mémoire en les recopiant sur les sorties S_0, S_1, S_2 du circuit.

Caractéristiques du transfert parallèle :

- ✚ Une impulsion d'horloge suffit à transférer les n bits.
- ✚ Transfert rapide (haute fréquence possible).
- ✚ Chaque bascule de départ est reliée à une bascule d'arrivée.
- ✚ Interconnexions nombreuses.
- ✚ Inconvénient majeur si émetteur loin du récepteur.

Stockage et transfert série

Exemple avec deux registres de 2 bits :



Les bascules sont connectées en cascades.

A chaque impulsion d'horloge, l'information d'entrée est transmise à la première bascule, puis à la suivante etc....

Le registre 2 reçoit progressivement les informations du registre 1.

Caractéristiques du transfert série :

- ✚ Il faut n impulsions d'horloge pour récupérer les n bits du registre 1 dans le registre 2.
- ✚ Transferts lents.
- ✚ Seule la dernière bascule du registre 1 est reliée à la première bascule du registre 2.
- ✚ Peu d'interconnexions, peu d'encombrement.

Ce type de registre (montage en cascade – transfert série) s'appelle des registres à décalage.

Possibilité de :

- ✚ Décalage vers la droite.
- ✚ Décalage vers la gauche.
- ✚ Décalage circulaire (rotation droite ou gauche).

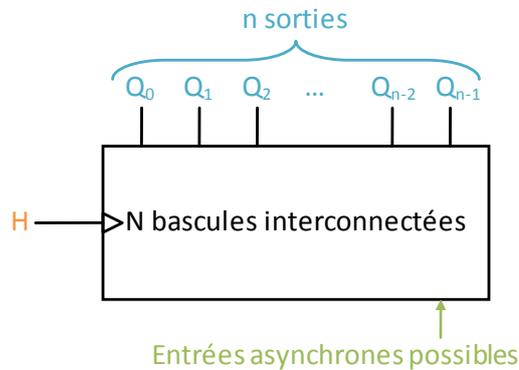
Registres universels

Ils possèdent les différentes fonctions :

- ✚ Entrées séries.
- ✚ Entrées parallèles.
- ✚ Sorties séries.
- ✚ Sorties parallèles.
- ✚ Décalage à droite.
- ✚ Décalage à gauche.
- ✚ Entrées de validation de la fonction voulue.

Les compteurs

Définition d'un compteur



- ✚ Ensemble de n bascules interconnectées (+ éventuelles portes logiques).
- ✚ Une horloge qui commande le changement d'état des n sorties.
- ✚ N combinaisons possibles en sortie ($N \leq 2^n$).

Compteur « modulo N » = compteur « par N »

A chaque impulsion d'horloge H, une nouvelle combinaison apparaît en sortie, l'ordre d'apparition des combinaisons étant prédéfini.

Lorsque le compteur a balayé les N combinaisons possibles, retour à la combinaison de départ, etc...

Un système à n bascules peut au maximum présenter 2^n combinaisons différentes en sortie.

Compteur à « cycle complet »

Si le compteur balaie toutes les combinaisons possibles...

- ✚ $N = 2^n$ (n = nombre de bits en sortie).
- ✚ Le compteur est dit « à cycle complet ».

Exemple : compteur à deux bascules, sorties Q₀ et Q₁.

| Séquence décimale | Ordre | Q ₀ | Q ₁ | Fonction |
|-------------------|------------------|----------------|----------------|------------------------------------|
| 0 | 1 ^{er} | 0 | 0 | Etat initial ou après 4 impulsions |
| 3 | 2 ^{ème} | 1 | 1 | Après 1 impulsion |
| 2 | 3 ^{ème} | 1 | 0 | Après 2 impulsions |
| 1 | 4 ^{ème} | 0 | 1 | Après 3 impulsions |

Compteur à cycle incomplet

Si le compteur ne balaie pas toutes les combinaisons possibles :

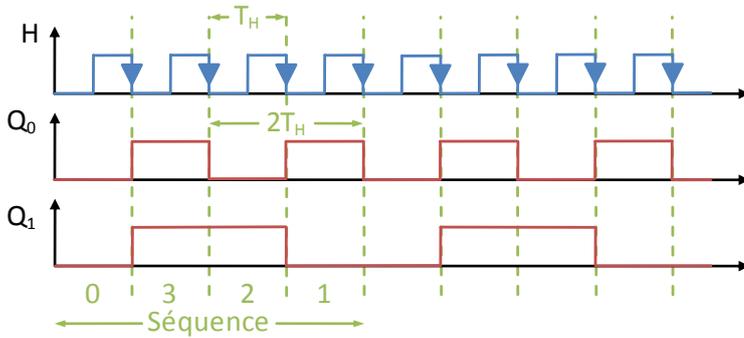
- ✚ $N < 2^n$
- ✚ Le compteur est dit « à cycle incomplet ».

Exemple : compteur à deux bascules, sorties Q_0 et Q_1 .

| Séquence décimale | Ordre | Q_0 | Q_1 | Fonction |
|-------------------|------------------|-------|-------|------------------------------------|
| 2 | 1 ^{er} | 1 | 0 | Etat initial ou après 2 impulsions |
| 1 | 2 ^{ème} | 0 | 1 | Après 1 impulsion |

Chronogrammes associés aux compteurs :

Exemple : Compteur par 4. Deux sorties Q_0 et Q_1 :



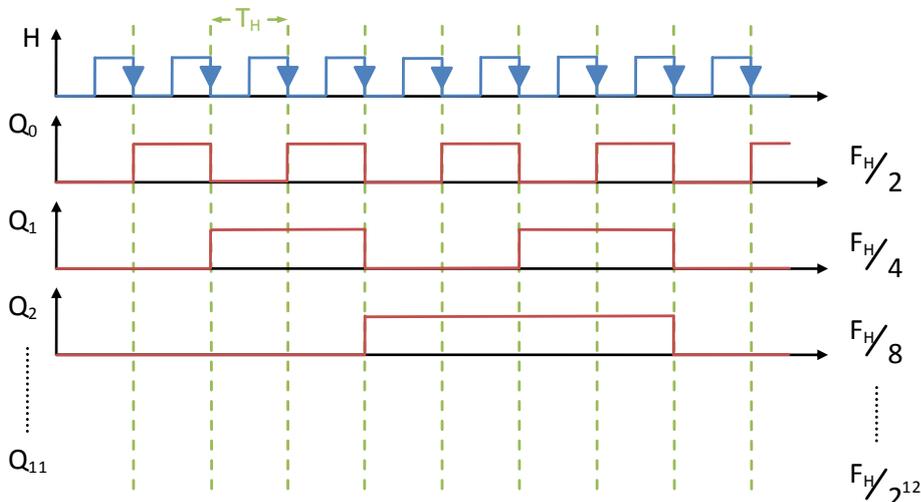
$$T_{Q0} = 2T_H, F_{Q0} = F_H / 2$$

$$T_{Q1} = 4T_H, F_{Q1} = F_H / 4$$

- ✚ Le motif se répète toutes les N impulsions d'horloge.
 - Compteur « modulo N »
- ✚ Si F_H est la fréquence de l'horloge.
 - Il existe toujours au moins une sortie dont la fréquence est F_H / N
 - Compteur par N = diviseur par N

Exemple de compteur commercial : n°4040 (voir TP)

- ✚ Compteur par $4096 = 2^{12}$ → Il contient 12 bascules élémentaires (sorties Q_0, \dots, Q_{11}).
- ✚ Il est déclenché sous fronts d'horloge descendants.
- ✚ Il balaie les 4096 combinaisons dans l'ordre binaire croissant.
- ✚ Chronogramme des 3 premières sorties :



$$F_n = F_H / 2^{n+1}$$

Il peut être utilisé en TP comme un diviseur de fréquence :

Si on envoie en entrée du circuit 4040 un signal TTL de fréquence F_H :

Signaux disponibles en sortie :

- ✚ Sortie Q0 : signal TTL de fréquence $F_H / 2$
- ✚ Sortie Q1 : signal TTL de fréquence $F_H / 4$
- ✚ ...
- ✚ Sortie Q8 : signal TTL de fréquence $F_H / 2^9 = F_H / 512$

D'après le constructeur, le circuit possède également une RAZ active sur niveau 1 :

| RAZ | Fonction |
|-----|----------------------------------------|
| 0 | Fonction synchrone : Compteur par 4096 |
| 1 | Mise à zéro de toutes les sorties |

Il faut donc câbler RAZ à la masse pour avoir accès à la fonction « compteur » = diviseur de fréquence.

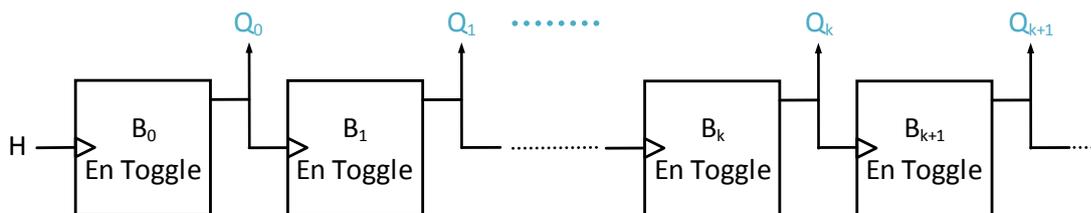
Technologie MOS, les entrées non connectées rendent le composant instable. Il faut donc toujours câbler les entrées.

Technologie TTL, elles sont considérées comme un état 1.

Les compteurs asynchrones

Principe d'un compteur asynchrone (ou compteur à propagation)

- ✚ Les n bascules fonctionnent en mode Toggle.
- ✚ Seule la première bascule est déclenchée par le signal d'horloge H .
- ✚ La bascule de rang k reçoit son signal de déclenchement d'horloge H .
- ✚ La bascule de rang k reçoit son signal de déclenchement via la bascule de l'étage de rang $k-1$ (Soit directement, soit avec des portes logiques intermédiaires).
- ✚ Câblage en cascade des n bascules :



Avantage :

- ✚ Circuiterie assez simple, nombre d'interconnexions limité.

Inconvénients :

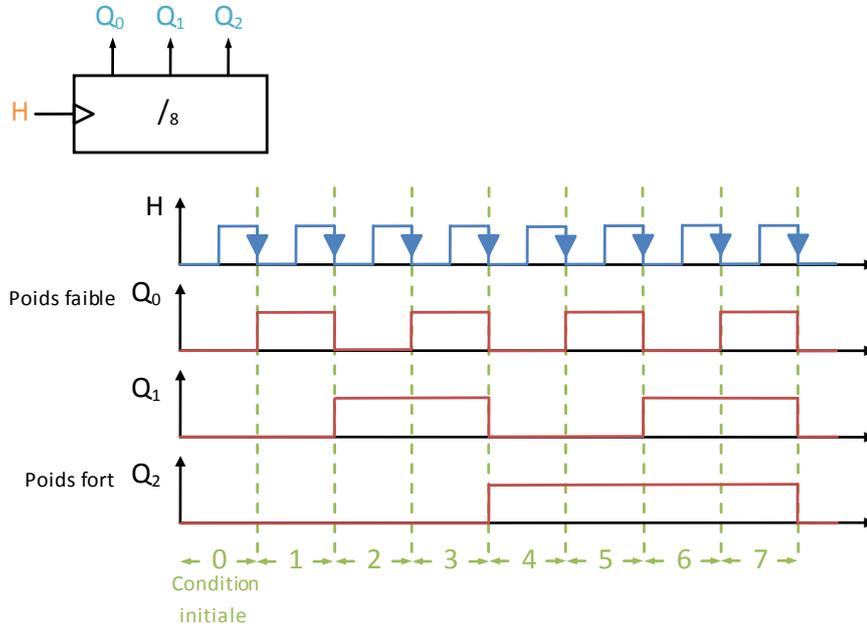
- ✚ La bascule B_1 doit attendre l'arrivée du signal en sortie de la bascule B_0 .
- ✚ La bascule B_2 doit attendre l'arrivée du signal en sortie de la bascule B_1 .
- ✚ Possibilité d'états transitoires indésirables.
- ✚ Les temps de réponse de chaque bascule (quelques 10 ns) s'accumulent.
- ✚ Fonctionnement relativement lent du compteur = limité en fréquence : l'horloge H ne doit pas varier trop rapidement pour laisser le temps à toutes les bascules de commuter.

Que l'on veuille concevoir un compteur ASYNCHRONE à cycle complet ou incomplet, on commence par câbler un compteur à cycle complet.

Compteurs asynchrones à cycle complet

Exemple : conception d'un compteur progressif modulo 8 (binaire naturel) :

Poids faible



| Décimal | Q ₂ | Q ₁ | Q ₀ |
|---------|----------------|----------------|----------------|
| 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 |
| 2 | 0 | 1 | 0 |
| 3 | 0 | 1 | 1 |
| 4 | 1 | 0 | 0 |
| 5 | 1 | 0 | 1 |
| 6 | 1 | 1 | 0 |
| 7 | 1 | 1 | 1 |

Bascule 0 :

- Doit basculer à chaque front descendant de H.
- En Toggle + déclenchée par H.

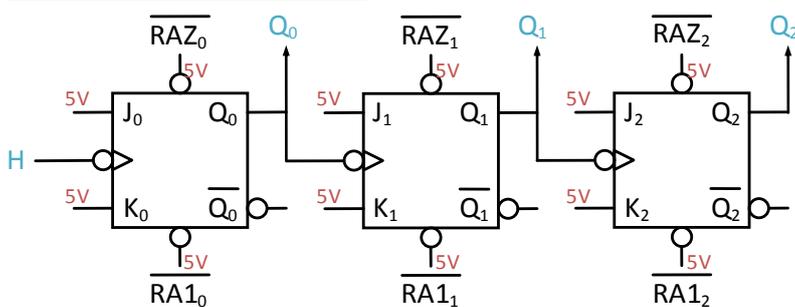
Bascule 1 :

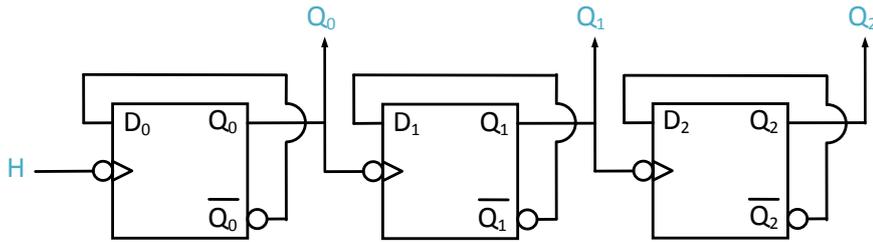
- Doit basculer à chaque front descendant de Q₀.
- En Toggle + déclenchée par Q₀.

Bascule 2 :

- Doit basculer à chaque front descendant de Q₁.
- En Toggle + déclenchée par Q₁.

Câblage avec des bascules JK :



Câblage avec des bascules D :

Compteur régressif modulo 8 (voir TD).

Compteur et décompteur modulo 8 (voir TD).

Compteurs asynchrones à cycle incompletPrincipe :

- ✚ Le compteur doit balayer un nombre de combinaisons $N < 2^n$.

1^{ère} étape :

- ✚ On construit d'abord le compteur asynchrone modulo $2^n > N$ correspondant à un cycle complet.

2^{ème} étape :

- ✚ On laisse le compteur défiler les N combinaisons souhaitées (sans tronquer la $N^{\text{ième}}$).
- ✚ On laisse apparaître (quelques nanosecondes) la $(N+1)^{\text{ème}}$ combinaison (non souhaitée).
- ✚ On utilise cette $(N+1)^{\text{ème}}$ combinaison pour forcer le compteur à se diriger vers la 1^{ère} combinaison souhaitée.
- ✚ On force cette 1^{ère} combinaison en activant les entrées asynchrones (RAZ et RA1) des n bascules.

Exemple 1

Compteur devant parcourir la séquence « décimale » : 1, 2, 3, 4, 5 :

- ✚ C'est un compteur modulo $N = 5$.
- ✚ On construit le compteur à cycle complet correspondant :
Modulo 8 = 2^3 .
3 bascules.
Comptage progressif.
- ✚ L'apparition de la combinaison $Q_2, Q_1, Q_0 = 110 = \ll 6 \gg$ devra diriger le compteur vers la combinaison $(Q_2, Q_1, Q_0) = (001)$ c'est-à-dire activer :
 - La mise à 1 de la bascule B_0 (poids faible).
 - La mise à 0 de la bascule B_1 .
 - La mise à 0 de la bascule B_2 (poids fort).

Exemple 2

Séquence décimale : 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 -> modulo $N = 10$.

Compteur à cycle complet correspondant : modulo 16 = 2^4 -> 4 bascules.

4 bascules JK déclenchées sous fronts descendants, possédant deux entrées asynchrones actives sur niveau 0 :

$\overline{\text{RAZ}}$ et $\overline{\text{RA1}}$.

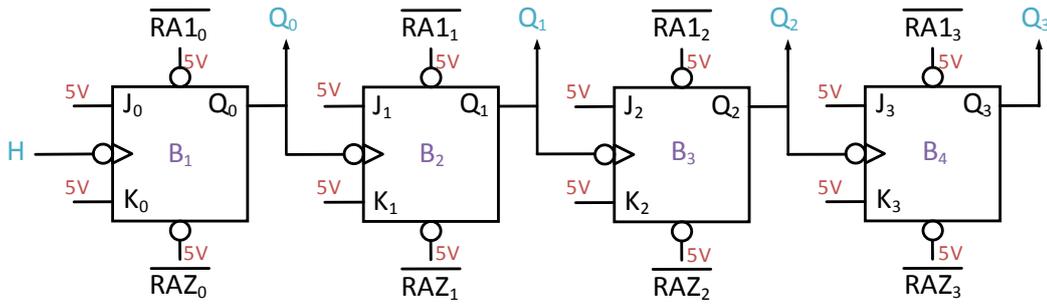


Tableau de fonctionnement du compteur par 10 :

| Décimal | D Q ₃ | C Q ₂ | B Q ₁ | A Q ₀ | \overline{RAZ}_i | \overline{RAZ}_i |
|---------|---------------------|---------------------|---------------------|---------------------|--------------------|--------------------|
| 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 | 0 |
| 2 | 0 | 0 | 1 | 0 | 1 | 0 |
| 3 | 0 | 0 | 1 | 1 | 1 | 0 |
| 4 | 0 | 1 | 0 | 0 | 1 | 0 |
| 5 | 0 | 1 | 0 | 1 | 1 | 0 |
| 6 | 0 | 1 | 1 | 0 | 1 | 0 |
| 7 | 0 | 1 | 1 | 1 | 1 | 0 |
| 8 | 1 | 0 | 0 | 0 | 1 | 0 |
| 9 | 1 | 0 | 0 | 1 | 1 | 0 |
| 10 | 1 | 0 | 1 | 0 | 0 | 1 |

Tous les \overline{RAZ}_i sont mis à 0 pour obtenir $(Q_3, Q_2, Q_1, Q_0) = (0,0,0,0)$.

Lorsque la combinaison (1, 0, 1, 0) apparaît, les \overline{RAZ} des 4 bascules doivent être activées pour diriger le compteur vers la combinaison (0, 0, 0, 0).

\overline{RAZ}_i = fonction combinatoire des (Q_3, Q_2, Q_1, Q_0) sorties du compteur.

$$\overline{RAZ} = \overline{A} \cdot B \cdot \overline{C} \cdot D$$

$$\overline{RAZ} = \overline{A} \cdot B \cdot \overline{C} \cdot D$$

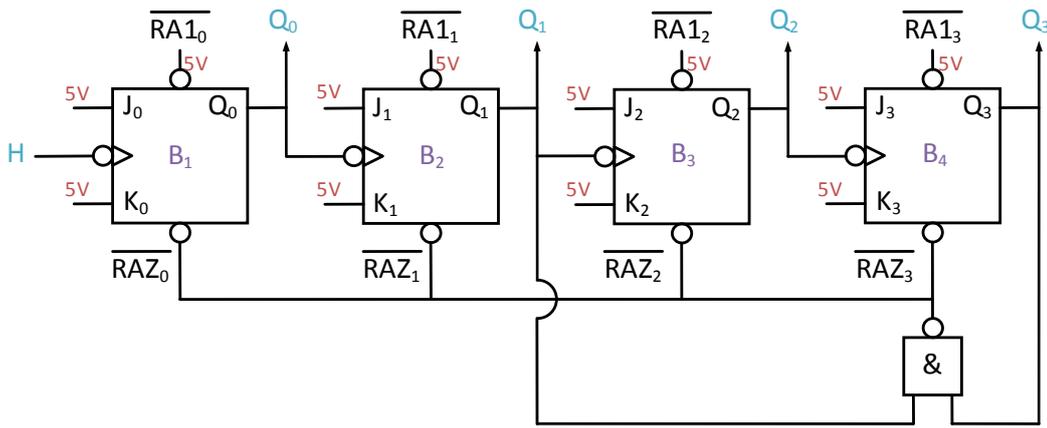
Remarque :

On peut se limiter à n'activer que \overline{RAZ}_1 et \overline{RAZ}_3 puisque Q_0 et Q_2 sont déjà à 0. Mais dans ce cas, il faut tout de même câbler \overline{RAZ}_0 et \overline{RAZ}_2 (à 5V) afin de ne pas laisser de broches non câblées et ainsi éviter les problèmes de parasites.

| | | Q_1 | | | |
|-------|-------|----------------|----------------|----------------|----------------|
| | | $Q_1 Q_0$ | Q_0 | | |
| Q_3 | Q_2 | 00 | 01 | 11 | 10 |
| | 00 | 0 | 0 | 0 | 0 |
| | 01 | 0 | 0 | 0 | 0 |
| | 11 | X ₀ | X ₀ | X ₁ | X ₁ |
| 10 | 0 | 0 | X ₁ | 1 | |

$$\overline{RAZ} = Q_1 \cdot Q_3 = \overline{RAZ}$$

Finalement, $\overline{RAZ} = \overline{Q_1} \cdot Q_3 \rightarrow$ Porte NAND.



Compteur asynchrone, cas d'une séquence quelconque

Objectif : compteur modulo N mais dont la séquence n'est ni progressive ni régressive.

On construit un compteur asynchrone modulo N progressif (par exemple)

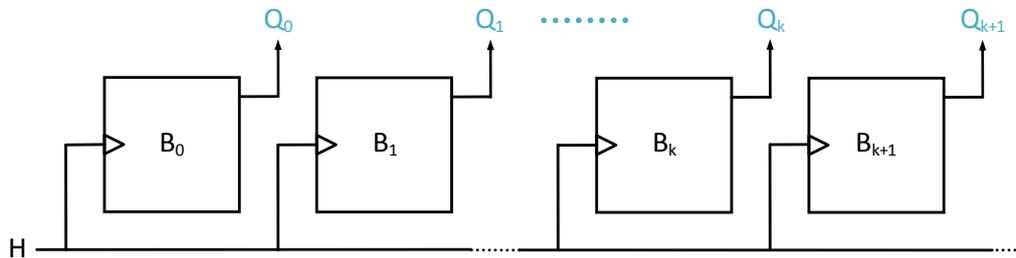
On fabrique les sorties voulues à l'aide de portes logiques.

Exemple en TD : compteur modulo 6 de séquence 7, 0, 5, 4, 6, 2.

Les compteurs synchrones

Principe d'un compteur synchrone

- Toutes les bascules sont déclenchées en même temps par le même signal d'horloge H.
- Câblage en parallèle des n bascules. (Les bascules ne sont pas forcément en toggle)



- Les entrées (J_i , K_i) ou (D_i) de toutes les bascules doivent être « préparées » avant chaque impulsion d'horloge pour provoquer la « bonne » réaction de la sortie Q_i au moment de l'impulsion d'horloge.

| Q_i | Prévoir J_i et K_i telles que |
|-------|-----------------------------------|
| 0 | Transition de Q_i de 0 vers 0 |
| 0 | Transition de Q_i de 0 vers 1 |
| 1 | Transition de Q_i de 1 vers 0 |

Avantages :

- Réponse rapide et quasi simultanée de toutes les sorties du compteur.
- Possibilité de fréquence de fonctionnement importante.

Inconvénient :

- Circuit plus complexe, plus d'interconnexions.

Compteurs synchrones réalisés avec des bascules JK

Rappel table de fonctionnement bascule JK :

| H | J | K | Q | Fonction |
|---|---|---|------------------|----------|
| ↑ | 0 | 0 | Q_0 | Mémoire |
| | 0 | 1 | 0 | RAZ |
| ↓ | 1 | 0 | 1 | RA1 |
| | 1 | 1 | $\overline{Q_0}$ | Toggle |

Table de transition de la bascule JK :

| Avant impulsion | Après impulsion | Fonction possible | A préparer avant impulsion | |
|-----------------|-----------------|-------------------|----------------------------|-------|
| Q_n | Q_{n+1} | | J_n | K_n |
| 0 | 0 | Mémoire ou RAZ | 0 | X |
| 0 | 1 | RA1 ou Toggle | 1 | X |
| 1 | 0 | RAZ ou Toggle | X | 1 |
| 1 | 1 | Mémoire ou RA1 | X | 0 |

Exemple du compteur modulo 10 synchrone, bascules JK

Modulo 10 : cycle incomplet $N = 10 < 2^4 \rightarrow 4$ bascules.

Table de fonctionnement du compteur et préparation des valeurs des J_i et K_j :

| Décimal | Q_3 | Q_2 | Q_1 | Q_0 | J_3 | K_3 | J_2 | K_2 | J_1 | K_1 | J_0 | K_0 |
|---------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 0 | 0 | 0 | 0 | 0 | 0 | X | 0 | X | 0 | X | 1 | X |
| 1 | 0 | 0 | 0 | 1 | 0 | X | 0 | X | 1 | X | X | 1 |
| 2 | 0 | 0 | 1 | 0 | 0 | X | 0 | X | X | 0 | 1 | X |
| 3 | 0 | 0 | 1 | 1 | 0 | X | 1 | X | X | 1 | X | 1 |
| 4 | 0 | 1 | 0 | 0 | 0 | X | X | 0 | 0 | X | 1 | X |
| 5 | 0 | 1 | 0 | 1 | 0 | X | X | 0 | 1 | X | X | 1 |
| 6 | 0 | 1 | 1 | 0 | 0 | X | X | 0 | X | 0 | 1 | X |
| 7 | 0 | 1 | 1 | 1 | 1 | X | X | 1 | X | 1 | X | 1 |
| 8 | 1 | 0 | 0 | 0 | X | 0 | 0 | X | 0 | X | 1 | X |
| 9 | 1 | 0 | 0 | 1 | X | 1 | 0 | X | 1 | X | X | 1 |

Lorsque la combinaison (0000) apparaît en sortie, il faut que : $J_2 = 0$.

Lorsque la combinaison (0001) apparaît en sortie, il faut que : $J_2 = 0$.

Etc... et idem pour toutes les J_i et K_i .

Les entrées J_i et K_i des bascules B_i peuvent alors être considérées comme des fonctions combinatoires des sorties $Q_3 Q_2 Q_1 Q_0$.

Etude des entrées J_2 et K_2 de la bascule B_2 :

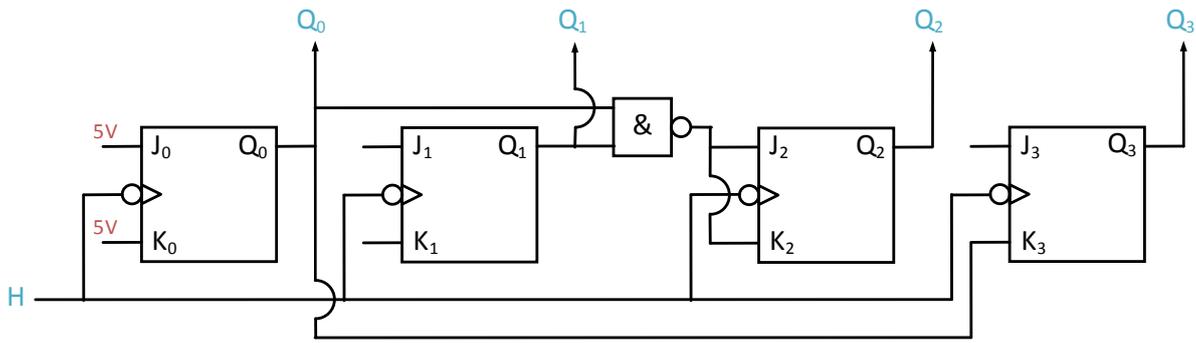
| $Q_3 Q_2 \backslash Q_1 Q_0$ | 00 | 01 | 11 | 10 |
|------------------------------|----|----|----|----|
| 00 | 0 | 0 | 1 | 0 |
| 01 | X | X | X | X |
| 11 | X | X | X | X |
| 10 | 0 | 0 | X | X |

$$J_2 = Q_0 \cdot Q_1$$

| $Q_3 Q_2 \backslash Q_1 Q_0$ | 00 | 01 | 11 | 10 |
|------------------------------|----|----|----|----|
| 00 | X | X | X | X |
| 01 | 0 | 0 | 1 | 0 |
| 11 | X | X | X | X |
| 10 | X | X | X | X |

$$K_2 = Q_0 \cdot Q_1$$

Etc... pour J_1, K_1, J_3 .



Compteurs synchrones réalisés avec des bascules D

Rappel de la table de fonctionnement de la bascule D :

| H | D | Q | Fonction |
|----|---|---|----------|
| | 0 | 0 | RAZ |
| ou | 1 | 1 | RA1 |

Alors il suffit de préparer la valeur de D_n avant l'impulsion à la valeur Q_{n+1} souhaitée après l'impulsion d'horloge.

Compteur modulo 10 synchrone - bascule D

Modulo 10 : cycle incomplet $N = 10 < 2^4 \rightarrow 4$ bascules.

Table de fonctionnement du compteur et préparation des valeurs de D_i :

| Déc | Sortie du compteur | | | | Entrées des bascules | | | |
|-----|--------------------|-------|-------|-------|----------------------|-------|-------|-------|
| | Q_3 | Q_2 | Q_1 | Q_0 | D_3 | D_2 | D_1 | D_0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| 2 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |
| 3 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 4 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 |
| 5 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 |
| 6 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 |
| 7 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| 8 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| 9 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |

Remarque : La colonne de D_i correspond à celle de Q_i décalée d'une ligne vers le haut.

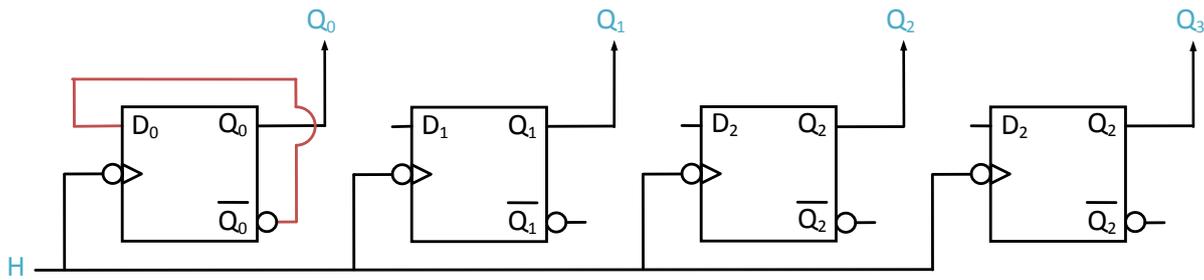
Etude de la fonction D_2 :

| $Q_3 Q_2 \backslash Q_1 Q_0$ | 00 | 01 | 11 | 10 |
|------------------------------|----|----|----|----|
| 00 | 0 | 0 | 1 | 0 |
| 01 | 1 | 1 | 0 | 1 |
| 11 | X | X | X | X |
| 10 | 0 | 0 | X | X |

$$D_2 = \overline{Q_1} Q_2 + Q_0 Q_2 + Q_0 Q_1 Q_2$$

Procédure identique pour D_0 , D_1 et D_2 .

Puis câblage du compteur :

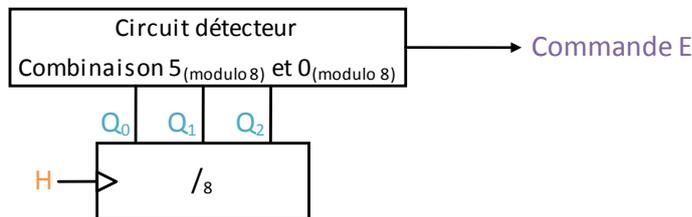
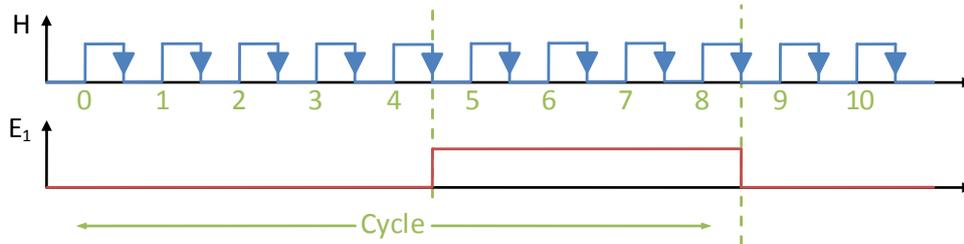


Décodage de l'état d'un compteur

Dans de nombreuses applications, les compteurs servent à commander le cadencement ou à ordonner automatiquement les opérations.

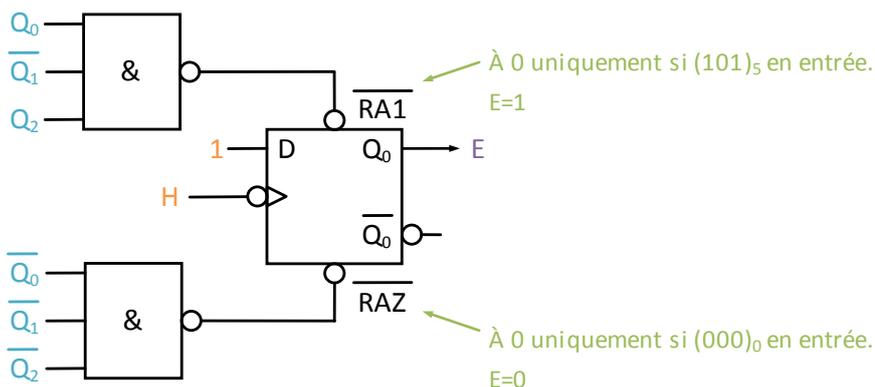
Exemple : Commande cyclique d'une électrovanne.

- De 0s à 5s, électrovanne fermée (état E=0).
- De 5s à 8s, électrovanne ouverte (état E=1).
- Puis reprise du cycle.



- Horloge de période $T=1s$.
- Conception d'un compteur modulo 8 (3 bascules) : sorties Q2, Q1, Q0.
- Détection des états $(101)_5 \text{ modulo } 8$ et $(000)_0 \text{ modulo } 8$: mise à 1 de E si état (101) détecté et mise à 0 de E si état (000) détecté.

Par exemple en jouant sur les entrées asynchrones \overline{RAZ} et $\overline{RA1}$ d'une bascule supplémentaire.

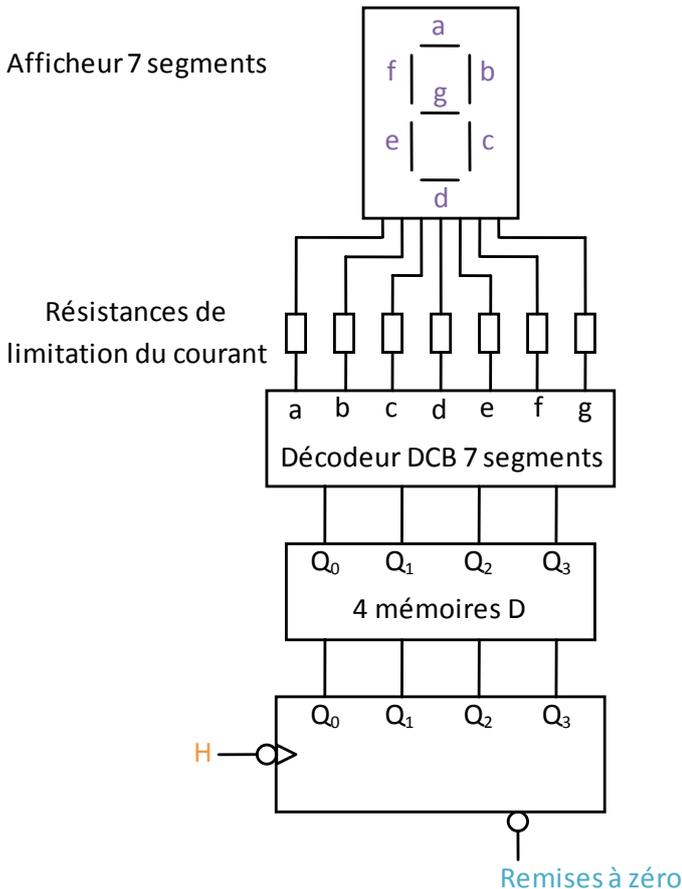


Généralisation :

Si on souhaite décoder tous les états d'un compteur modulo $N=2^n$ alors : N portes NAND à n entrées.
 Si 16 états à détecter : 1 porte NAND à 4 entrées.... Circuiterie de plus en plus lourde.

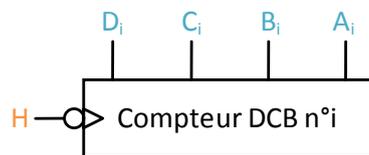
Afficheurs de l'état d'un compteur sur un afficheur 7 segments.

En TP : compteurs modulo 10 commerciaux - circuit 7490 (voir doc)



Compteurs DCB montés en cascade

- ✚ Un compteur par 10 pour les unités.
- ✚ Un compteur par 10 pour les dizaines.
- ✚ Un compteur par 10 pour les centaines...

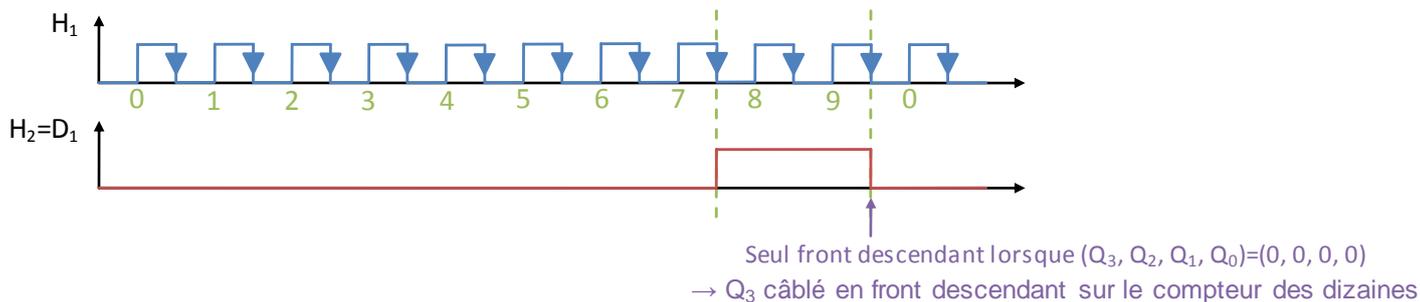


Objectif : incrémenter le compteur des dizaines (2) lorsque le compteur des unités (1) termine un cycle...

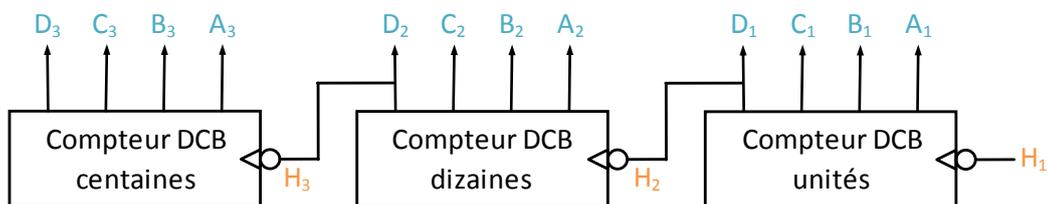
Traduction : envoyer un front d'horloge (descendant) sur l'horloge H_2 , quand (et seulement quand) le compteur des unités passe de la combinaison $(1001)_9$ à la combinaison $(0000)_0$

Inutile de fabriquer cette impulsion H_2 : la sortie de D_1 (poids fort) du compteur 1 présente un front descendant (et un seul) exactement à l'instant souhaité.

Cycle du compteur des unités :



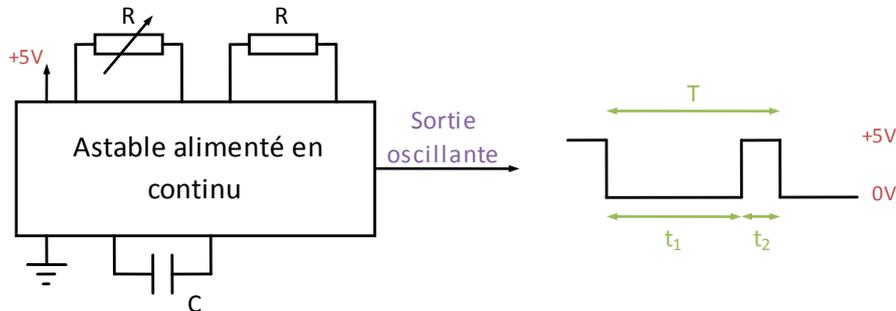
Même raisonnement avec le compteur des centaines qui doit être incrémenté par le compteur des dizaines.



Deux circuits annexes

Minuterie 555 montée astable

Astable = Multivibrateur Astable = Oscillateur



En sortie, le système oscille en permanence entre deux états instables (ici 0V et 5V)

- Multivibrateur astable = Générateur de signaux TTL (Rapport cyclique $\frac{T}{t_1} \neq \frac{1}{2}$)

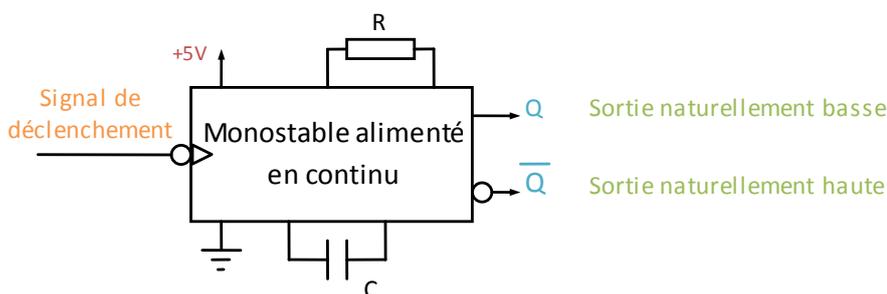
Pour le circuit 555 des maquettes de TP :

- Deux résistances extérieures (dont une variable : potentiomètre).
- Une capacité extérieure.
- Possibilité de régler la fréquence du signal de sortie.
- Possibilité de régler le rapport cyclique $\frac{T}{t_1}$ du signal de sortie.

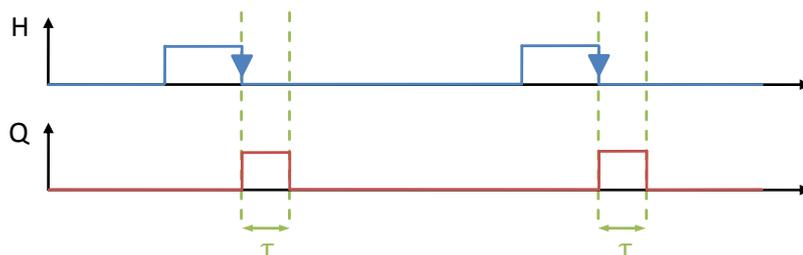
Préréglage des maquettes de TP : $F_{\text{astable}} = 2048\text{Hz}$

Circuit monostable - 74123

Monostable = multivibrateur monostable



Sous l'action d'un signal de déclenchement H, la sortie Q passe de son état (0V) à son état instable (5V) puis, au bout d'un temps τ prédéfini, revient naturellement dans son état stable.



- ✚ τ dépend de R et de C. (ex : $\tau = RC \cdot \ln 2$)
- ✚ τ est variable de quelques nanosecondes à quelques 10s.

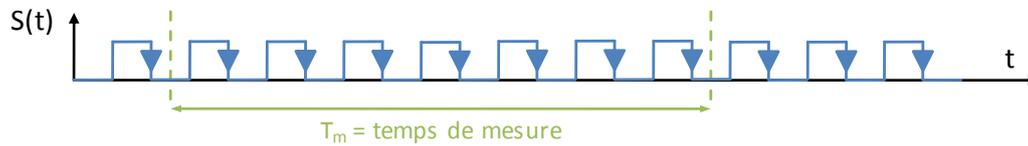
CIRCUIT MONOSTABLE = GENERATEUR D'IMPULSION CALIBREES

En TP – 74123 :

- ✚ Deux types de déclenchements possibles (front montant ou front descendant).
- ✚ Deux monostables par boîtier.

Application pratique des compteurs : Le fréquencesmètre

Soit $S(t)$ un signal de type TTL de fréquence f_s .



$$f_s = \frac{\text{nombre d'événements}}{\text{temps de mesure}} = \frac{\text{nombre de fronts descendants}}{t_m}$$

Si on choisit $t_m = 1\text{s}$:

- ✚ Mesurer f_s revient à compter le nombre de fronts montants ou descendants de $S(t)$ pendant une seconde.
- ✚ Afficher le résultat du comptage (se munir d'afficheurs).
- ✚ Remettre les compteurs à zéro (tout en continuant d'afficher le résultat).
- ✚ Reprendre le comptage pendant une seconde, etc.

1^{ère} étape :

- ✚ Disposer d'un autre signal $E(t)$ qui définit le temps de mesure (ici 1s).
- ✚ À l'aide de $S(t)$ et de $E(t)$ créer un signal « échantillonné » $S_{\text{éch}}(t)$ qui :
 - Possède le même nombre de fronts que $S(t)$ pendant une seconde.
 - Ne possède pas de front pendant la seconde suivante, etc.
- ✚ Envoyer le signal $S_{\text{éch}}(t)$ sur le compteur qui dénumbrera le nombre de fronts. (exemple ici : comptage des fronts descendants).

2^{ème} étape :

Une fois la seconde terminée...

- ✚ Transférer l'information « total du nombre de fronts comptés pendant une seconde » vers les afficheurs.
- ✚ Mémoriser cette dernière information sur les afficheurs.

Pour cette étape, l'information « fin de seconde » fournie par $E(t)$ sera utilisée pour générer une impulsion calibrée qui activera momentanément le transfert vers les afficheurs.

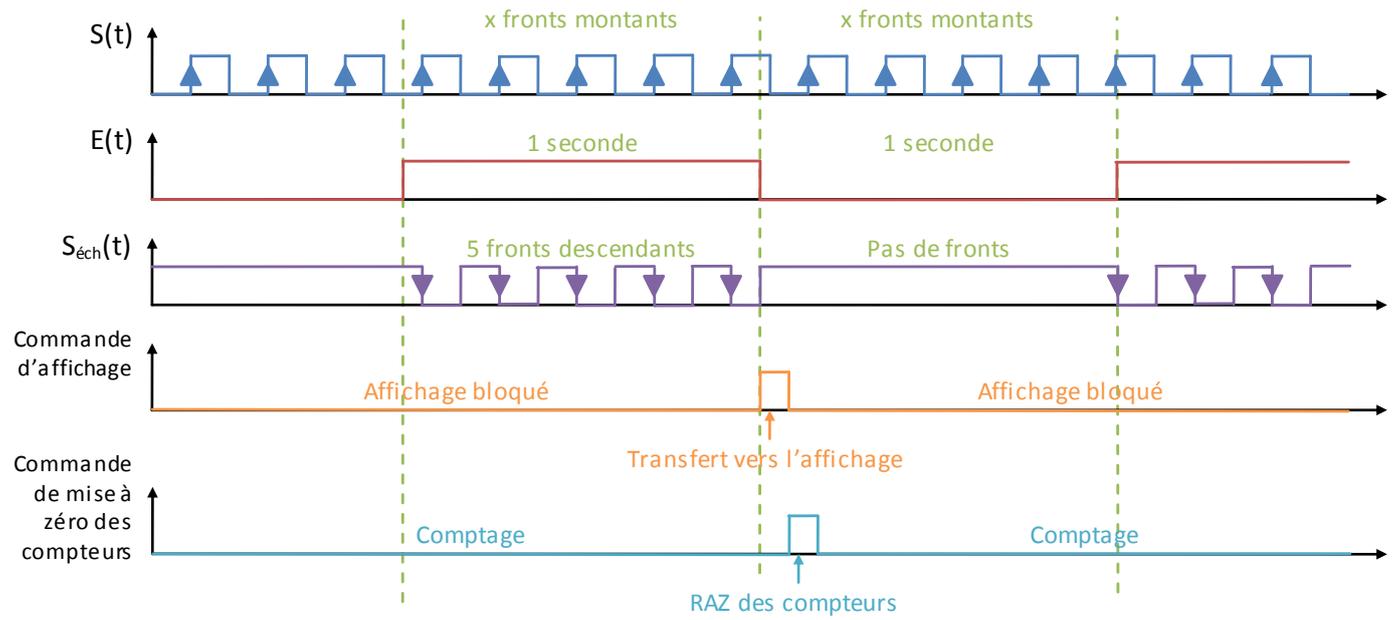
3^{ème} étape :

Une fois le transfert terminé, remettre les compteurs à 0.

Pour cette étape, l'information « fin de transfert vers les afficheurs » sera utilisée pour générer une impulsion calibrée qui activera momentanément la mise à zéro des compteurs.

Le système est alors prêt à reprendre la mesure de f_s .

Exemple de chronogrammes associés au fréquencemètre :



Quelques notions techniques sur les circuits numériques

Puces électroniques :



- ✚ Circuit intégré et encapsulé dans des boîtiers protecteurs (plastique ou céramique).
- ✚ Broches pour connections (14, 16, 20, 28, 40, 64, ...).

Circuits intégrés :

- ✚ Circuits composés de résistances, diodes, transistors, etc...
- ✚ Fabriqués directement dans un substrat semi-conducteur (silicium) auquel on fait subir différents traitements, exemple :
 - Implantations ioniques.
 - Oxydation.
 - Dépôts métalliques.

Avancées technologiques :

- ✚ Diminution des dimensions : micro → nano électronique.
- ✚ Augmentation des taux d'intégration : + de 10^6 portes logiques / puces.
- ✚ Diminution des coûts : production en grande série.
- ✚ Augmentation de la fiabilité (interconnexions intégrées).
- ✚ Diminution de la consommation (miniaturisation → moindre ventilation).

Persistance des composants discrets, notamment pour les composants à fortes puissances ou les transformateurs, bobines, grandes capacités...

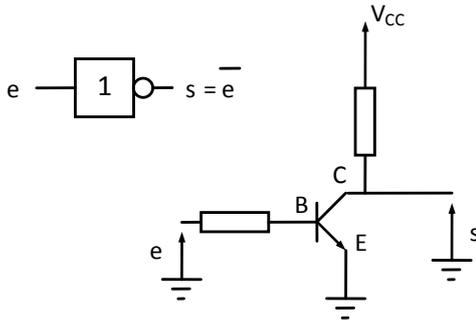
Famille logique TTL : « Transistor – Transistor – Logic »

Basée sur le fonctionnement du transistor bipolaire (NPN ou PNP).

Exemple : une porte logique NON contient :

- ✚ 4 transistors.
- ✚ 2 diodes.
- ✚ 4 résistances.
- ✚ Deux bornes d'alimentation (V_{cc} et la masse).

Schéma interne simplifié :



Série TTL Standard : numéro d'identification 74...

ex : 74000 (4 portes NAND à 2 entrées).

Sous familles :

- ✚ 74H : TTL rapide
- ✚ 74L, ... TTL faible consommation
- ✚ 74LS : Schottky faible consommation

Famille logique CMOS : « Complementary Metal Oxide Semiconductor »

Basée sur le fonctionnement du transistor à effet de champ (MOSFET à canal P ou à canal N), ne nécessite pas d'éléments résistifs : moins encombrant, moins d'interconnexions.

Broches non connectées :

Une broche non connectée peut se comporter comme une **antenne** et capter des signaux parasites susceptibles **d'entraver le bon fonctionnement** des portes logiques.

Famille TTL :

Une broche non connectée se comporte comme un 1 logique.
Il est cependant plus prudent de connecter toutes les broches.

Famille CMOS :

Signaux parasites + charges électrostatiques = polarisation des transistors, états conducteurs, surconsommation, surchauffe...

Une broche non connectée peut entraîner la destruction de la porte logique.

- ✚ Ne jamais laisser de broches non connectées.

Entrance d'une porte logique :

« Puissance » minimum à appliquer en entrée d'une porte logique pour pouvoir la piloter.
Pour une porte standard : entrance = 1.

Sortance d'une porte logique (facteur de charge – fan out) :

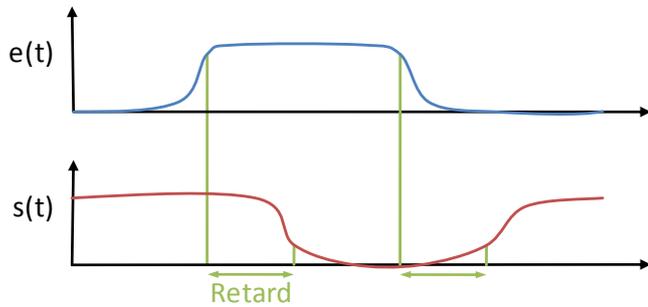
Nombre maximum d'entrées logiques standard (d'entrance 1) que la porte logique peut piloter.
Au-delà de cette valeur, les niveaux logiques ne sont plus assurés.

Exemple : Pour la série TTL 74LS, le constructeur indique : sortance = 20.

La sortie de la porte logique peut piloter jusqu'à 20 autres portes logiques standard.

Retard à la propagation :

Temps de réaction de la sortie de la porte.



Typiquement :

- ✚ Famille TTL : 10ns.
- ✚ Famille CMOS : de 10 à 50ns.

Autres indications des constructeurs :

- ✚ Consommation : typiquement 10mWatt.
- ✚ Marge de sensibilité au bruit : typiquement : 0,4V.
- ✚ Plage de T° admissibles.
- ✚ Niveaux des tensions d'alimentations...