Chapitre 4

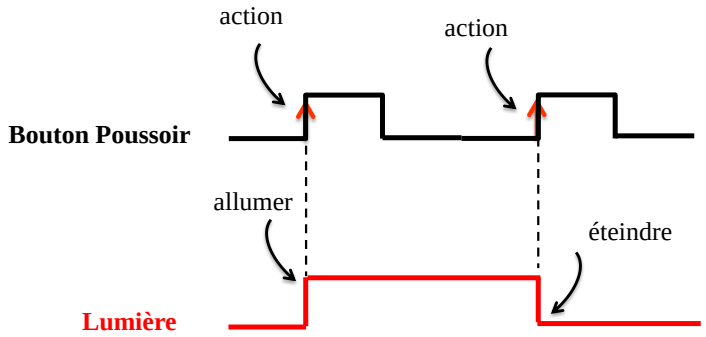
Les Circuits Séquentiels

I. Introduction 36
II. Systèmes asynchrones / synchrones 36
III. Les bascules 37
IV. Les Registres 41
V. Les compteurs / décompteurs 43

I. Introduction

Un circuit combinatoire est un circuit numérique dont les sorties dépendent uniquement des entrées. La différence essentielle entre les systèmes combinatoires que nous avons étudié dans le chapitre précédent et les systèmes séquentiels que nous allons aborder dans ce chapitre, réside dans le fait que la fonction de sortie de ces derniers systèmes dépend à la fois des variables d'entrée et de l'état antérieur des sorties. Or ce dernier dépendait de l'état des entrées précédentes, par conséquence de l'ordre dans lequel se succèdent les états d'entrée, d'où la terminologie « séquentielle » attribuée à ces circuits.

A titre d'illustration, prenons l'exemple très simple d'une lampe électrique commandée par un **bouton poussoir** de telle façon qu'une action sur le bouton allume la lampe, et qu'une action successive l'éteigne, selon l'exemple du chronogramme suivant :



On note bien que l'évolution de ce 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 ; un dispositif de mémoire à maintien la lampe allumée. C'est la caractéristique essentielle d'un système séquentiel.

Les fonctions séquentielles de base sont :

La mémorisation ; le comptage ; le décalage.

Les circuits séquentiels fondamentaux sont :

Les bascules (3 types) ; les compteurs ; les registres ; les RAM (Random AccessMemory).

Ces circuits peuvent travailler soit en mode synchrone, soit en mode asynchrone.

II. Système asynchrones / synchrones

II.1. Systèmes asynchrones

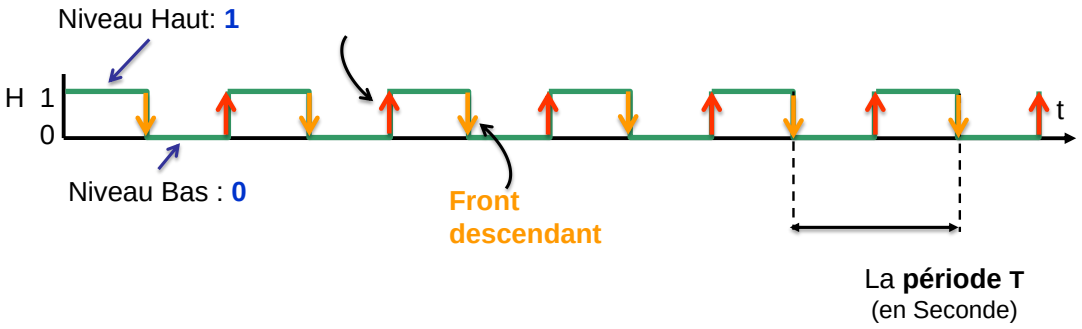
Les signaux d'entrée de ces systèmes peuvent provoquer à tout moment le changement d'état des sorties (après un certain retard qu'on appelle « temps de réponse ». Ces systèmes sont difficiles à concevoir et à dépanner.

II.2. Systèmes synchrones

Le changement d'état sur les sorties se produit pendant une transition appelé « front » (montant ou descendant) d'un signal maître, l'horloge. Les entrées servent à préparer le changement d'état, mais ne provoquent pas de changement des sorties. Tout changement d'état interne du montage est synchronisé sur le front actif de l'horloge.

La majorité des systèmes numériques séquentiels sont synchrones même si certaines parties peuvent être asynchrone (**ex.** : reset).

Une **horloge** est une variable logique qui passe successivement de 0 à 1 et de 1 à 0 d'une façon périodique. Cette variable est utilisée souvent comme une entrée des circuits séquentiels synchrone. L'horloge est notée par **h** ou **ck** (**clock**).



III. Les Bascules

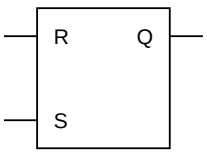
III.1. Bascule RS

La bascule RS **asynchrone** possède une entrée **R** (**Reset**) de mise à zéro, une entrée **S** (**Set**) de mise à 1 et une sortie **Q**.

Table de fonctionnement :

R	S	Q^+	
0	0	Q	Mémoire
0	1	1	Mise à 1
1	0	0	Mise à 0
1	1	Φ	Interdit

Symbole :



L'état $\boldsymbol{R = S = 0}$ maintient l'état de la sortie (mode **mémoire**).

L'état $\boldsymbol{R = S = 1}$ est interdit (mode **interdit**), car il conduit à mettre simultanément la sortie à **1** et à **0**.

III.1.1 Réalisation à base des portes NAND

Table de vérité :

R	S	Q	Q^+	
0	0	0	0	Mémoire
0	0	1	1	
0	1	0	1	Mise à 1
0	1	1	1	
1	0	0	0	Mise à 0
1	0	1	0	
1	1	0	Φ	Interdit
1	1	1	Φ	

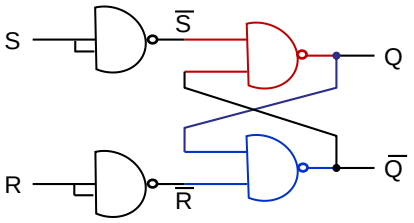
Tableau de Karnaugh :

RS \ Q	0	1
00	0	1
01	1	1
11	Φ	Φ
10	0	0

Equation logique de sortie :

$$Q^+ = S + Q \cdot \bar{R}$$

Logigramme :



Cette bascule **RS** est **prioritaire au 1** car, pour la combinaison $\boldsymbol{R = S = 1}$, la sortie Q est mise à 1 (les Φ ayant été fixés à 1 pour la simplification de Q).

Remarque 4.1 :

Le logigramme fait apparaître une sortie supplémentaire égale au **complément** de la sortie Q uniquement si la combinaison $\boldsymbol{R = S = 1}$ n'apparaît pas.

III.1.2 Réalisation à base des portes NOR

Table de vérité :

R	S	Q	Q^+	
0	0	0	0	Mémoire
0	0	1	1	
0	1	0	1	Mise à 1
0	1	1	1	
1	0	0	0	Mise à 0
1	0	1	0	
1	1	0	Φ	Interdit
1	1	1	Φ	

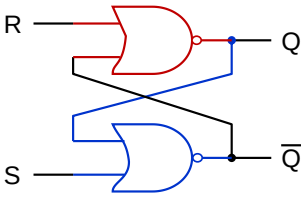
Tableau de Karnaugh :

RS \ Q	0	1
00	0	1
01	1	1
11	Φ	Φ
10	0	0

Equation logique de sortie :

$$Q^+ = \bar{R} \cdot (Q + S)$$

Logigramme :



Cette bascule **RS** est **prioritaire au 0** car, pour la combinaison $\boldsymbol{R = S = 1}$, la sortie Q est mise à 0 (les Φ ayant été fixés à 0 pour la simplification de Q).

Remarque 4.2 :

Le logigramme fait apparaître une sortie supplémentaire égale au **complément** de la sortie Q uniquement si la combinaison $\boldsymbol{R = S = 1}$ n'apparaît pas.

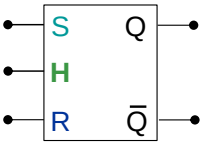
III.2. Bascule RSH

La bascule **RSH** est une bascule **synchronisée** par un **signal d'horloge H.** Les entées ***R*** et ***S*** de la bascule ne sont prises en compte que lorsque l'horloge ***H*** est **active** :

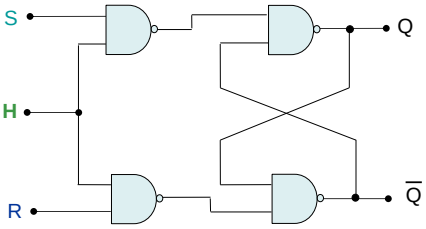
Table de vérité :

R	S	H	Q^+	
0	0	**0**	Q	Mémorisation
0	1	**0**	Q	
1	0	**0**	Q	
1	1	**0**	Q	
0	0	**1**	Q	Valeurs prises par Q^+ quand H passe de 0 à 1.
0	1	**1**	1	
1	0	**1**	0	
1	1	**1**	X	

Symbole :



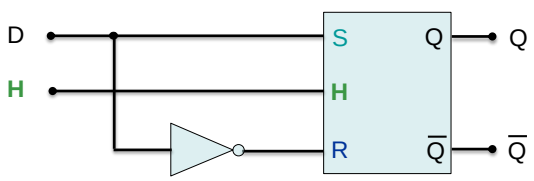
Logigramme avec des NAND :



III.3. Bascule à verrouillage (D-latch)

La **D-latch** est une **bascule RSH** pour laquelle on n'a conservé que les deux combinaisons $RS = (0,1)$ et $RS = (1,0)$. La D-latch a une seule entrée, nommée **D**.

La donnée **D** est **copiée** dans la bascule lorsque l'horloge passe de **1** à **0**. Elle y est **mémorisée** jusqu'à ce que l'horloge redevienne active.



Symbole :

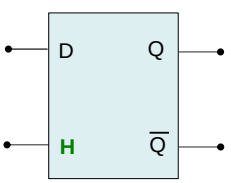


Table de vérité :

D	H	Q_{n+1}	Mode
X	0	Q_n	Verrouillée
0	1	0	Transparent
1	1	1	Transparent

III.4. Bascules J-K

Les bascules JK sont plus polyvalentes que les basculent RS, puisque l'état indéterminé est remplacé par un état complémenté, elles n'ont pas donc d'état ambigu $R = S = 1 \rightarrow Q_{n+1} = \overline{Q_n}$.

La bascule J K synchrone est obtenue à partir d'une bascule RSH dont les sorties sont rebouclées sur les entrées. Ceci permet d'éliminer l'état indéterminé.

Logigramme avec des NAND :

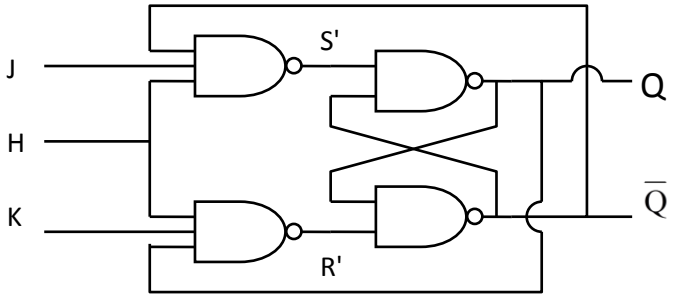


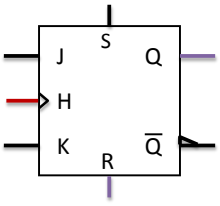
Table de vérité :

J	K	Q_{n+1}	fonctionnement
0	0	Q_n	état mémoire
0	1	0	mise à 0
1	0	1	mise à 1
1	**1**	$\overline{Q_n}$	**basculement**

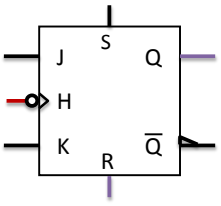
D'après cette table de vérité, nous remarquons que les 3 premières lignes correspondent bien à une bascule **RS** mais la dernière ligne introduit un mode **supplémentaire** qui sera très utilisé pour réaliser les compteurs synchrones que nous verrons plus loin.

Symbole :

bascule JK synchronisée sur front montant



bascule JK synchronisée sur front descend



IV. Les registres

Un registre est un groupe de bascules qui partagent une horloge commune et qui peut stocker un bit. Un registre à ***n bits*** est un groupe de ***n bascules*** qui peuvent stocker n bits. Un registre peut aussi avoir des portes combinatoires qui permettent de mieux traiter les bits. Les bascules dont le stockage de l'information, et les portes déterminent comment l'information est transférée dans le registre.

Les informations peuvent être écrites/lues en même temps (parallèle) ou une après l'autre (série).

Le nombre de bits du registre correspond au nombre de cellules mémoire (nombre de bascule D ou JK) du registre. On note que toutes les entrées d'horloge (H) des cellules sont reliées (ligne d'écriture).

Les registres sont classées par Les registres de mémorisation peuvent être classés selon la méthode d'écriture de données ou de lecture :

- Des registres à entrées parallèles et sorties parallèles : PIPO (**P**arallel **I**N-**P**arallel **O**UT).
- Des registres à entrées parallèles et sorties séries : PISO (**P**arallel **I**N-Serial **O**UT).
- Des registres à entrées séries et sorties parallèles : SIPO (**S**erial **I**N- **P**arallel **O**UT).
- Des registres à entrées séries et sorties séries : SISO (**S**erial **I**N- **S**erial **O**UT).

IV.1. Registres de mémorisation (Registre parallèle)

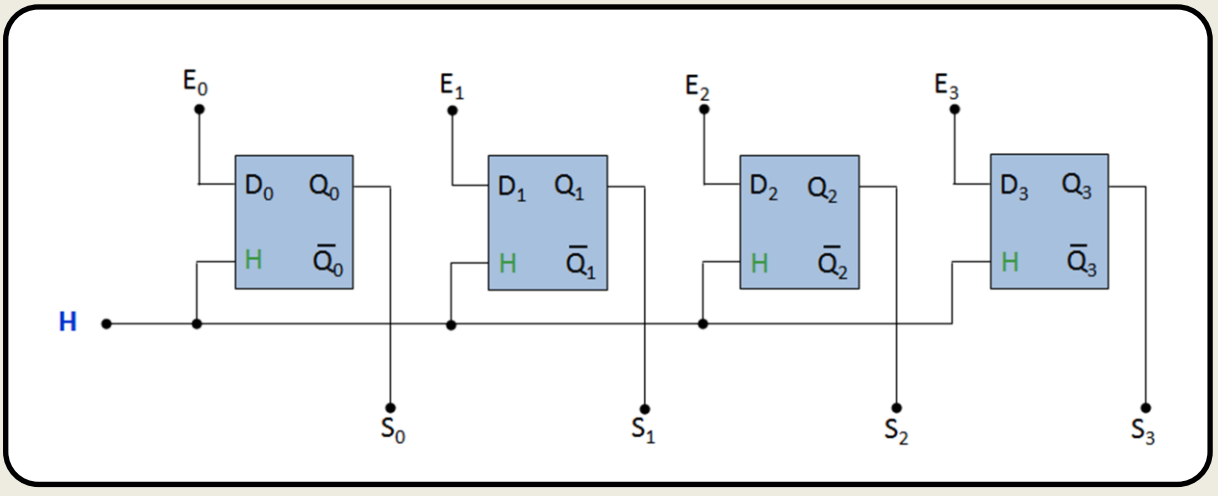
Un registre de mémorisation (ou registre de données) est un registre dans lequel les différents étages sont indépendants les uns des autres, cependant certains signaux agissent sur l'ensemble des étages ; tel que remise à **0** et remise à **1**.

IV.1.1 Registre de mémorisation 4 bits

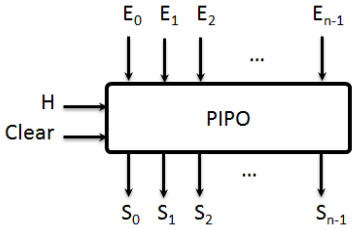
Dans l'exemple ci-dessous, les 4 bascules sont chargées en parallèle et lues en parallèle en synchrone avec le signal d'écriture H. Ce type de registre est appelé aussi registre **PIPO**.

EXEMPLE 4. 1 :

Registre de mémorisation 4 bits



IV.1.2 Schéma fonctionnel d'un registre PIPO.

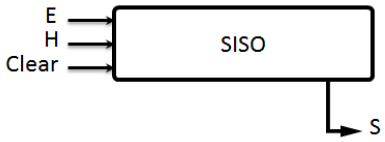


IV.2. Registres à décalage (Registre série)

Ce type de registre est principalement utilisé comme mémoire d'information dynamique ; la fonction de décalage consiste de faire glisser l'information de chaque cellule élémentaire dans une autre cellule élémentaire adjacente.

Ce type de registre est appelé aussi registre **SISO**.

IV.2.1 Schéma fonctionnel



IV.2.2 Décalage à droite

La bascule du rang i doit recopier la sortie de la bascule du rang (i-1) ainsi son entrée doit être connectée à la sortie (i-1).

IV.2.3 Décalage à gauche

L'entrée de la bascule du rang i doit recopier la sortie de la bascule du rang (i+1).

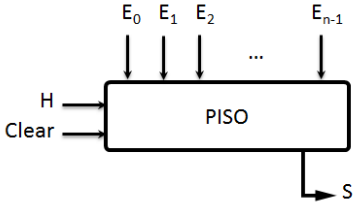
IV.2.4 Décalage réversible

Il existe des registres à décalage réversibles, c'est à dire des registres ou le décalage s'effectue vers la droite et vers la gauche en fonction du niveau logique applique à l'entrée S : « sens de décalage ».

IV.3. Registre mixte

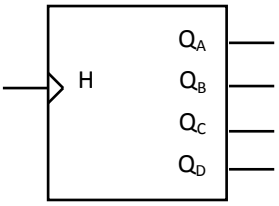
On peut trouver des registres mixtes, donc on peut écrire en parallèle et lire en série (PISO), ou vice versa (PISO), ou qui offrent les deux possibilités « au choix ».

Schéma fonctionnel d'un PISO :



V. Les compteurs

Un compteur est un circuit qui compte des impulsions et qui affiche sur ses sorties le nombre d'impulsions qu'il a reçues depuis le début du comptage (en binaire évidemment).
On le représente par le schéma suivant (exemple d'un compteur sur 4 bits synchronisé sur front montant) :



Q_A est bien sur la sortie de poids faible et Q_D celle de poids fort.

Il existe 2 types de compteurs : les compteurs asynchrones et les compteurs synchrones (les termes "synchrone" ou "asynchrone" n'ont pas la même signification que pour les bascules et les 2 types de compteurs sont réalisés avec des bascules synchrones).

La capacité d'un compteur encore appelée MODULO est le nombre maximum d'états différents que peuvent prendre l'ensemble de ses sorties.

V.1. Les compteurs asynchrones

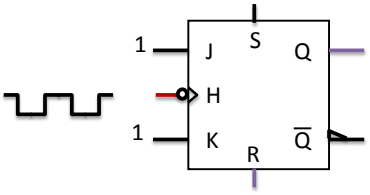
Rappel sur les entrées de forçage des bascules :

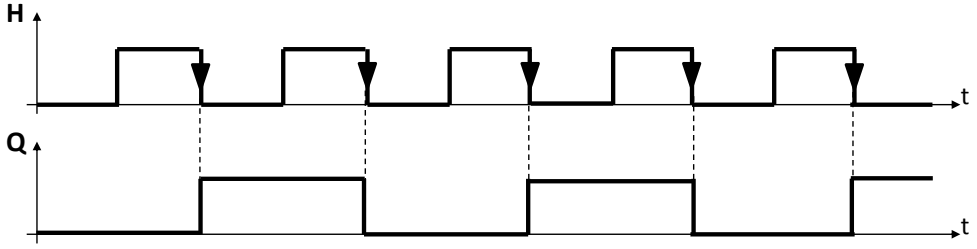
En plus des entrées 'normales', **RS**, **D** ou **JK** et de l'entrée d'horloge **H**, les bascules possèdent des entrées de mise à 0 (**Reset**) ou de mise à 1 (**Set** ou **Preset**) en général actives à l'état bas et qui sont prioritaires par rapport à toutes les autres entrées.

Dès que l'entrée **Reset = 0**, **Q = 0** ; dès que l'entrée **Preset = 0**, **Q = 1**. Evidemment, il faut éviter de les mettre toutes les **2** ensemble à **0** !

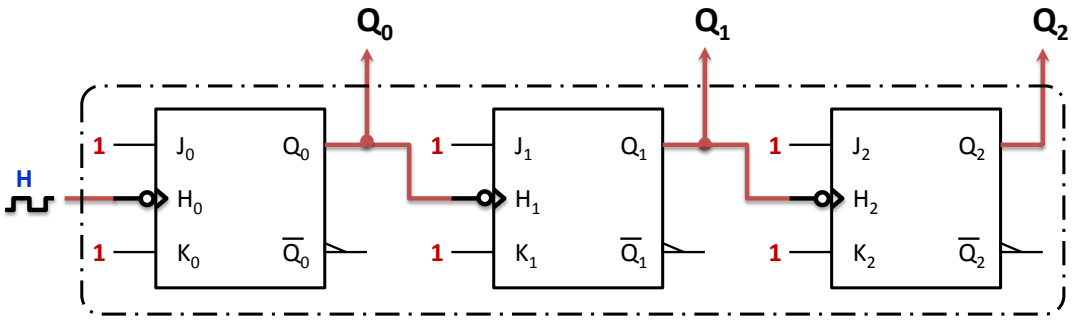
Les compteurs asynchrones sont réalisés avec des bascules montées en **diviseurs de fréquence** par 2 (avec des bascules **JK**, il suffit de mettre **J** et **K** à **1**).

Lorsque les entrées **J** et **K** de la bascule **J-K** sont à **1,** la sortie **Q** au front d'horloge suivant est complémentée. La sortie change d'état sur un *front descendant d'horloge*.





V.1.1 Compteurs modulo 2^n (exemple de 3 bits)



Pour compter en binaire naturel il faut que le bit de rang i + 1 change d'état quand le bit de rang i repasse à 0 : si l'on utilise des bascules synchronisées sur front descendant, il suffit de relier les sorties Q_i aux entrées d'horloge des bascules de rang i + 1, ce qui est réalisé sur le schéma ci-dessus.

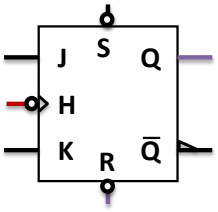
Le compteur représenté ici comptera donc de ***0*** (***000***) à ***7*** (***111***), le retour à 0 se faisant automatiquement à la $8^{ème}$ impulsion d'horloge.

En généralisant, on voit qu'avec n bascules, on obtient des compteurs modulo $\mathbf{2^n}$.

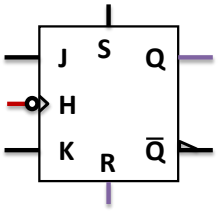
V.1.2 Compteurs modulo $\neq 2^n$ (cycle incomplet)

Pour revenir à 0 sur une autre valeur qu'une puissance de 2, il faut détecter cette valeur et remettre immédiatement le compteur à 0 : On utilisera l'entrée de remise à 0 asynchrone que possèdent toutes les bascules. Elle fonctionne de la façon suivante : dès que l'on applique l'état actif sur cette entrée, la sortie Q de la bascule passe à 0, indépendamment de l'état de l'horloge et des autres entrées de commande (D ou J, K). Elle est notée **R** (Reset) ou **Cl** (Clear) et elle est soit active à l'état bas ou à l'état haut.

Ci-après les schémas de deux bascules JK, synchronisées sur front descendant. La première est munie d'une entrée de remise à 0 active à l'état bas alors que la deuxième est munie d'une entrée de remise à 0 active à l'état haut.

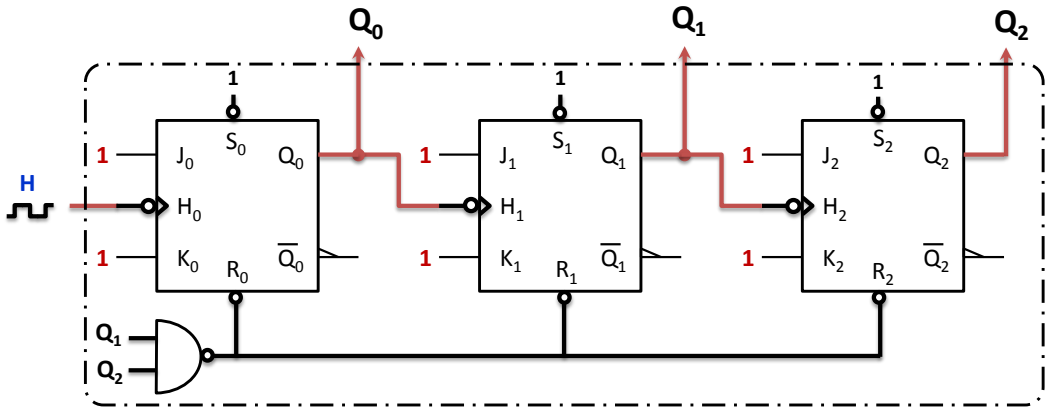


Reset active à l'état ***bas***



Reset active à l'état ***haut***

Par exemple pour réaliser un compteur asynchrone qui compte de **0** à **5** et qui revient à **0** à la $6^{ème}$ impulsion, il faut détecter la valeur $\mathbf{(6)_{10}}$ soit $\mathbf{(110)_2}$ et activer les entrées Clear (**R**) des bascules. La remise à 0 doit donc être activée dès que $\mathbf{Q_1 = Q_2 = 1}$ et $\mathbf{Q_0 = 0}$ mais comme on compte par ordre croissant, c'est la $1^{ère}$ fois que $\mathbf{Q_1}$ et $\mathbf{Q_2}$ sont tous les deux à **1** en même temps : il suffit donc de détecter le passage à **1** de ces **2** sorties et d'envoyer un **0** sur les entrées Clear (**R**) des 3 bascules actives à l'état bas, ce qui se fait avec une simple porte NAND selon le schéma suivant :



Par mesure de précaution, on remet à **0** toutes les bascules, même la bascule **0** qui est déjà.

V.2. Les décompteurs asynchrones

V.2.1 Décompteurs modulo 2^n (exemple de 3 bits)

Pour décompter en binaire naturel il faut que le bit de rang i + 1 change d'état quand le bit de rang i passe à **1** : si l'on utilise des bascules synchronisées sur *front descendant*, il suffit de relier les sorties $\overline{Q_i}$ aux entrées d'horloge des bascules de rang i + 1. Le schéma est donc analogue à celui des compteurs modulo 2^n en tenant compte de la remarque ci-dessus.

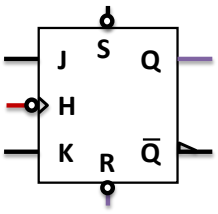
V.2.1 Décompteurs modulo $\neq 2^n$

Le principe est analogue à celui des compteurs à quelques détails près : pour réaliser un décompteur modulo **n**, on doit obtenir la séquence suivante : n-1, n-2, …, 3, 2, 1, 0, n-1, …
La valeur 0 doit bien exister et rester stable tant que la nième impulsion n'est pas arrivée : si le modulo n était une puissance de 2, la valeur suivant 0 serait n-1 c'est à dire que toutes les sorties des bascules seraient à 1.

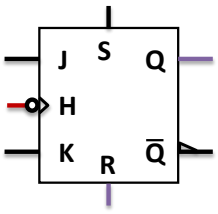
Par exemple pour réaliser un décompteur asynchrone qui décompte de 5 à 0 et qui revient à 5 à la $6^{ème}$ impulsion, il faut détecter la valeur $(\mathbf{7})_{10}$ soit $(\mathbf{111})_2$ et la remplacer par $(\mathbf{5})_{10}$ soit $(\mathbf{101})_2$.

La méthode la plus simple serait de détecter la valeur $(\mathbf{7})_{10}$ et de remettre la sortie $\mathbf{Q_1}$ à **0** en utilisant un simple NAND à **3** entrées.

Pour plus de sécurité, on préfère repositionner toutes les sorties dans l'état correct c'est à dire **101**.On doit donc utiliser des bascules disposant, en plus de l'entrée de remise à **0** (R) d'une entrée de remise à **1** fonctionnant de la même façon que l'entrée Clear : dès que l'on applique ***l'état actif*** sur cette entrée, la sortie Q de la bascule passe à **1**, indépendamment de l'état de l'horloge et des autres entrées de commande (D ou J, K). Elle est notée **Pr** (Preset) ou **S** (Set) et elle est soit active à l'état bas ou à l'état haut.

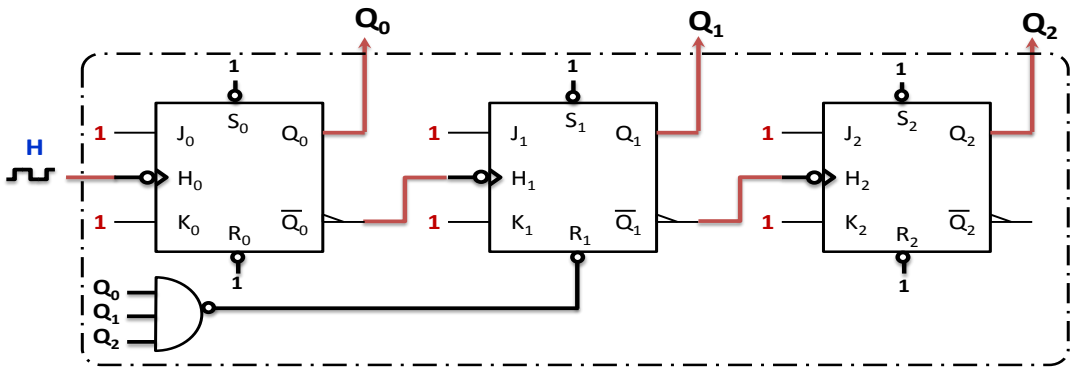


Set active à l'état ***bas***



Set active à l'état ***haut***

Le schéma de ce décompteur asynchrone est le suivant :



V.3. Les compteurs synchrones

Les compteurs asynchrones ont un certain nombre d'inconvénients :

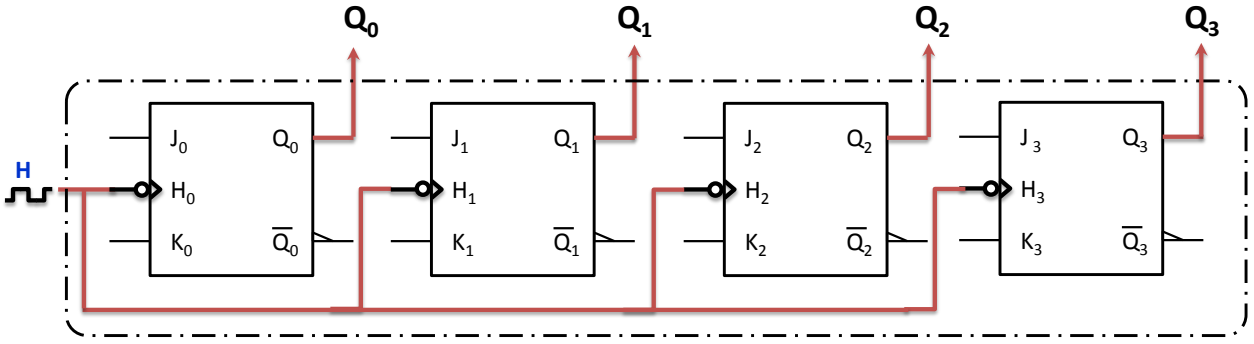
- On ne peut compter ou décompter que dans l'ordre binaire pur : impossible par exemple de construire un compteur en *code Gray*.
- La fréquence maximum de comptage est **limitée** par les temps de retard des bascules qui **s'ajoutent** : si chaque bascule a un **temps de retard** de **50 ns**, le passage de **0111** à **1000** ne sera effectif qu'au bout de **200** ns, le temps que la dernière bascule ait réagi.

On préfère donc utiliser des compteurs dits **synchrones** : c'est quand les entrées d'horloge de toutes les bascules du compteur sont reliées au **même signal d'horloge**, contrairement aux compteurs asynchrones dans lesquels la sortie d'une bascule commandait l'entrée d'horloge de la suivante.

V.3.1 Définition

Un compteur synchrone est constitué de ***n bascules*** qui reçoivent ***en parallèle*** le ***même signal d'horloge***. L'entrée horloge est donc commune à toutes les bascules dont les sorties changent d'états simultanément ensemble contrairement aux compteurs asynchrones dans lesquels la sortie d'une bascule commandait l'entrée d'horloge de la suivante.

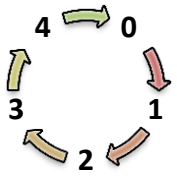
Pour un compteur de modulo ≤ **16**, cela donne le schéma suivant :



Le problème va donc consister à relier correctement les entrées ***J*** et ***K*** de chaque bascule aux sorties ***Q*** des autres bascules pour obtenir le **cycle voulu**.

On va utiliser l'équation d'une bascule **JK** qui donne la valeur de la sortie Q_{n+1} en fonction de son état antérieur noté Q_n et des états de ***J*** et de ***K*** au moment du front d'horloge, soit :

Le plus simple est de partir d'un exemple concret, celui d'un compteur ***modulo 5*** qui nécessitera ***3 bascules : 2***, ***1*** et ***0***. On doit donc obtenir le cycle ***0***, ***1***, ***2***, ***3***, ***4***, ***0***, ***1*** … dont le diagramme des états est le suivant :



V.3.2 Table et fonctions de transition

Les fonctions de commutation ou de transition φ_i sont définies de telle sorte à ce que $\varphi_i = 1$ lorsque la **bascule *i*** change d'état d'une combinaison à une autre (de ***l'état présent*** à ***l'état future***), soit la fonction de transition :

$$\varphi_n = J_n.\overline{Q_n} + K_n.Q_n$$

On commence donc par remplir la ***table de transition*** correspondant au cycle voulu.

	N_{10}	Q_2	Q_1	Q_0	φ_2	φ_1	φ_0
état initial	**0**	**0**	**0**	**0**	0	0	1
après la 1ère impulsion	1	0	0	1	0	1	1
après la 2ème impulsion	2	0	1	0	0	0	1
après la 3ème impulsion	3	0	1	1	1	1	1
après la 4ème impulsion	4	1	0	0	1	0	0
après la 5ème impulsion	**0**	**0**	**0**	**0**			

Puis il faut remplir les tableaux de Karnaugh de chaque fonction φ_i en fonction de Q_i.

φ_0	$Q_1\,Q_0$			
Q_2	00	01	11	10
0	1	1	1	1
1	0	X	X	X

φ_1	$Q_1\,Q_0$			
Q_2	00	01	11	10
0	0	1	1	0
1	0	X	X	X

φ_2	$Q_1\,Q_0$			
Q_2	00	01	11	10
0	0	0	1	0
1	1	X	X	X

On en tire les équations des sorties φ_i en utilisant **éventuellement** les valeurs indéterminées ***X*** :

$$\varphi_0 = \overline{Q_2} \qquad \varphi_1 = Q_0 \qquad \varphi_2 = Q_0.Q_1 + Q_2$$

On identifie chaque équation à l'équation type d'une bascule JK :

$$\varphi_i = J_i.\overline{Q_i} + K_i.Q_i$$

le coefficient de $\overline{Q_i}$ doit être égal à J_i et celui de Q_i doit être égal à K_i.

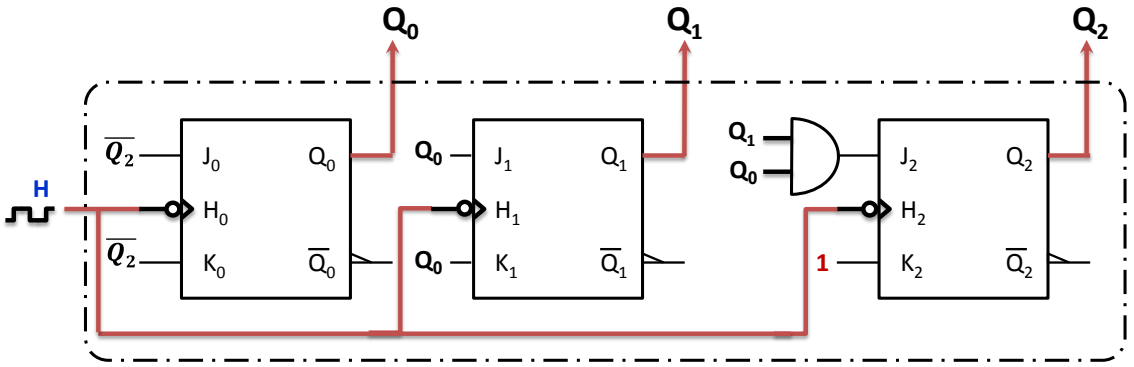
Donc on déduit les égalités suivantes :

$$J_0 = K_0 = \overline{Q_2}\ ;$$

$$J_1 = k_1 = Q_0\ ;$$

$$J_2 = Q_0.Q_1 \text{ et } K_2 = 1.$$

ce qui donne le schéma suivant :



V.4. Compteurs spéciaux

V.4.1 Compteurs réversibles

Ce sont des compteurs/décompteurs qui peuvent compter comme ils peuvent décompter. Le choix de la fonction comptage ou de la fonction décomptage s'effectue à l'aide d'une entrée de sélection.

V.4.2 Compteurs programmables ou pré-positionnés (presettables)

Ce sont des compteurs possédant des entrées de pré-chargement qui contiennent la valeur de départ du comptage ou de décomptage. Le chargement du compteur se fait grâce à une entrée de commande.

V.5. Compteurs synchrones à circuits intégrés

On cite à titre d'exemple :

- Les compteurs synchrones pré-positionnés modulo 10 : ***74LS160*** et ***74LS162.***
- Les compteurs synchrones pré-positionnés modulo 16 : ***74LS161*** et ***74LS163***.
- Les compteurs/décompteurs synchrones pré-positionnés modulo 10 : ***74LS190*** et ***74LS192***.
- Les compteurs/décompteurs synchrones pré-positionnés modulo 16 : ***74LS191*** et ***74LS193***.