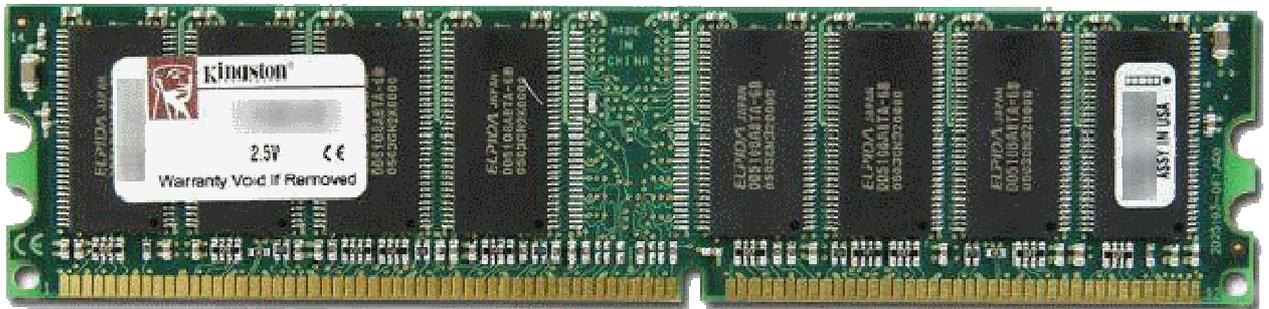


Les structures séquentielles



Baccalauréat STI2D Centre d'intérêt :

- Traitement d'une information numérique (SIN 2.1)
- Traitement de l'information : logique séquentielle (ET 3.1.4)



Objectifs

A la fin de la séquence, l'élève doit être capable

- de distinguer une structure séquentielle d'une structure combinatoire
- de citer les structures séquentielles intégrés principales
- d'exploiter un document constructeur d'une structure séquentielle

Traitement séquentiel

Une structure séquentielle restitue un résultat en fonction d'une séquence d'opérations.

Exemple :

SAISIE DE CODE SECURISE

A= SAISIE DE LA TOUCHE 1

B=SAISIE DE LA TOUCHE 2

C=SAISIE DE LA TOUCHE 3

D=SAISIE DE LA TOUCHE 4

SI ABCD=CODE_CORRECT

ALORS

AUTORISER_OPERATION

SINON

Structures logiques séquentielles



NB 12/09/2012

AFFICHER('CODE ERRONE')

FINSI

FIN

Si les codes ABCD sont saisis dans un ordre différent le code sera refusé.

Structure séquentielle

On appelle structures logiques séquentielles, des structures composées d'éléments logiques faisant intervenir la notion de mémoire et d'états précédents pour provoquer le changement d'état de sortie de la structure.

Ces structures reposent sur une notion de mémorisation.

Structures séquentielles intégrées

Les structures logiques séquentielles sont nombreuses.

Le cours se limitera aux structures élémentaires.

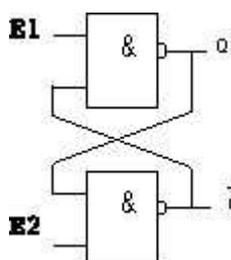
- les mémoires unitaires
- les registres
- les compteurs

Mémoires unitaires

Bascule RS

Une mémoire unitaire permet de stocker un état logique - un bit - "0" ou "1".

La mémoire logique de base repose sur deux portes logiques NAND ou NOR.



Un état logique 0 en entrée d'une porte NAND impose un état "1" en sortie.

En conséquence de quoi, lorsque les deux entrées E1 et E2 sont au niveau logique 1, l'état précédent est mémorisé.

La table de vérité de cette structure est :

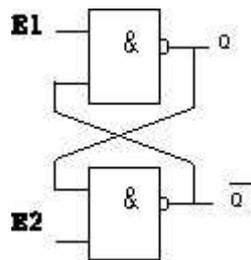
E1	E2	Q _n	/Q _n
0	0	1	1
0	1	1	0
1	0	0	1
1	1	Q _{n-1}	/Q _{n-1}

On remarque que l'entrée E1 permet la mise à 1 de la sortie Q lorsqu'elle est au NLO. Il s'agit de l'entrée Set (ou /S)

E2, si elle est au NLO permet la mise à 0 de Q. C'est l'entrée Reset (ou /R)

Lorsque /R=/S=1 la structure mémorise l'état précédent.

Analyse d'une bascule RS NAND



La recherche du fonctionnement se fait en trois étapes:

E1	E2	Q _n		/Q _n	
0	0				
0	1				
1	0				
1	1	Q _{n-1} =0		/Q _{n-1} =1	
		Q _{n-1} =1		/Q _{n-1} =0	

la moitié de la table de vérité pour laquelle les entrées sont à "0"

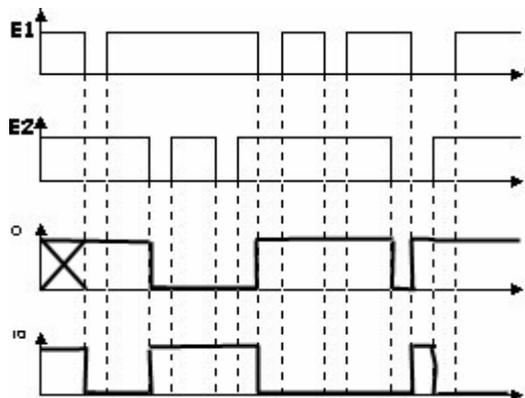
- Compléter les lignes partiellement remplies en positionnant sur le schéma les états connus et en recherchant les états inconnus
- Compléter la ligne restante en tenant compte du fait que l'état précédent (Q_{n-1} ou /Q_{n-1}) à son importance.

- Remplir les 4 cases grises en procédant comme dans l'étape 2

- Remplir les 2 cases blanches en comparant Q_n et Q_{n-1} d'une part puis $/Q_n$ et $/Q_{n-1}$ d'autre part

E1	E2	Q_n			$/Q_n$		
/S	/R						
0	0	1			1		
0	1	1			0		
1	0	0			1		
1	1	$Q_{n-1}=0$	0	Q_{n-1}	$/Q_{n-1}=1$	1	$/Q_{n-1}$
		$Q_{n-1}=1$	1		$/Q_{n-1}=0$	0	

Chronogramme de fonctionnement



Bascule D

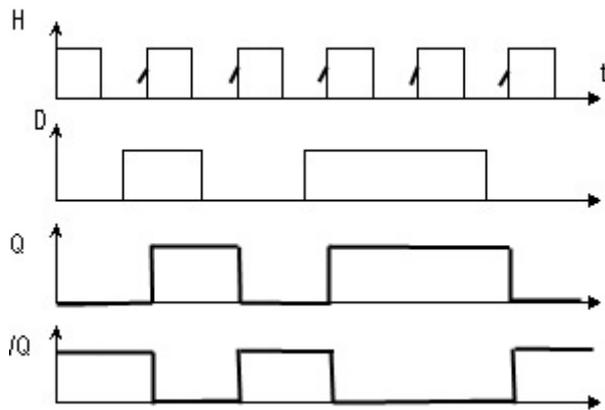
Cette mémoire unitaire est une mémoire synchrone car le changement de sa sortie ne s'opère que lors de changement de l'état de l'entrée d'horloge (appelée C ou H ou CK).

On dit que "le changement se fait au front". Le front pouvant être selon les circuits "montant" (de NL0 à NL1) ou "descendant" (de NL1 à NL0).

En absence de front actif, l'information est mémorisée.

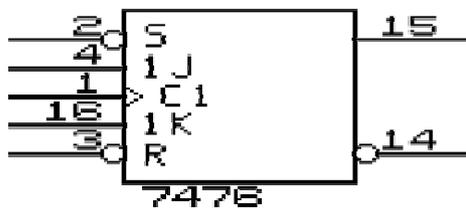
C	D	Q_n	$/Q_n$	Effet
Front actif	0	0	1	Mise à zéro
Front actif	1	1	0	Mise à un
Pas de front	X	Q_{n-1}	$/Q_{n-1}$	Mémorisation

Exemple : comportement d'une bascule D sensible sur front montant.



Bascule JK

Cette bascule possède deux entrées, J et K, qui déterminent l'état de sortie. Il s'agit également d'une bascule synchrone.



Remarque :

On remarque sur le symbole ci-dessus les entrées R et S de la bascule RS interne. Ces entrées sont prioritaires sur les entrées J et K et ne dépendent pas de l'entrée C.

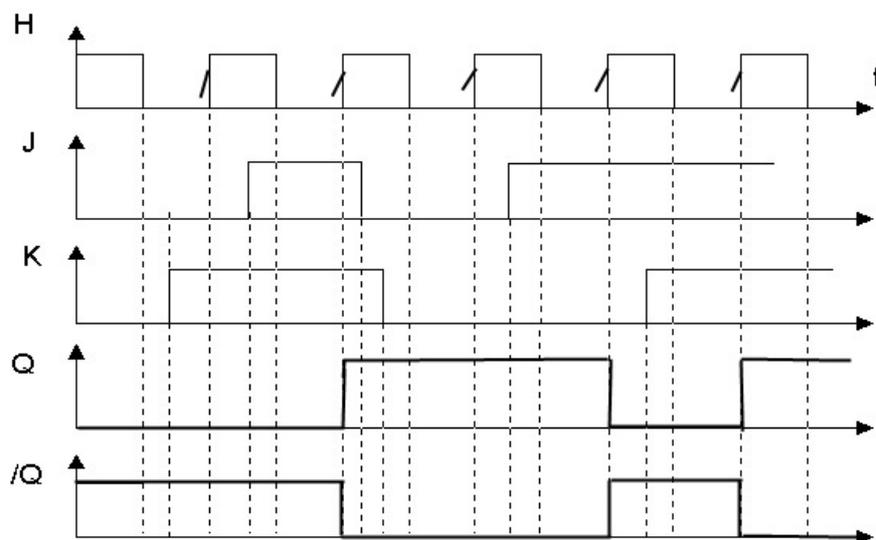


Table de vérité d'un circuit 7476

/S	/R	C	J	K	Qn	/Qn	Effet
0	1	X	X	X	1	0	Forçage à un
1	0	X	X	X	0	1	Forçage à zéro
0	0	X	X	X	1	1	Etat indésirable
1	1	↓	0	0	Qn-1	/Qn-1	Mémorisation
1	1	↓	1	0	1	0	Mise à un
1	1	↓	0	1	0	1	Mise à zéro
1	1	↓	1	1	/Qn-1	Qn-1	Bascule
1	1	0	X	X	Qn-1	/Qn-1	Mémorisation
1	1	1	X	X	Qn-1	/Qn-1	Mémorisation

Registres

Rappels

Mot binaire :

il s'agit de l'association de plusieurs bits. Chaque bit a un poids binaire et véhicule une information. Le bit situé le plus à droite dans le mot, est appelé LSB (Last significant bit) et possède le poids binaire le plus faible (2^0). Le bit situé le plus à gauche dans le mot, est appelé MSB (Most significant bit) et possède le poids binaire le plus fort (2^n).

Octet :

on appelle octet un mot formé de 8 bits ($D^7D^6D^5D^4D^3D^2D^1D^0$). L'octet est souvent codé en hexadécimal pour faciliter la lecture. Il est appelé BYTE en anglais et il ne faut pas le confondre avec le BIT.

Registre

Un registre est une microstructure logique séquentielle qui permet d'enregistrer, de conserver, puis de restituer une information binaire.

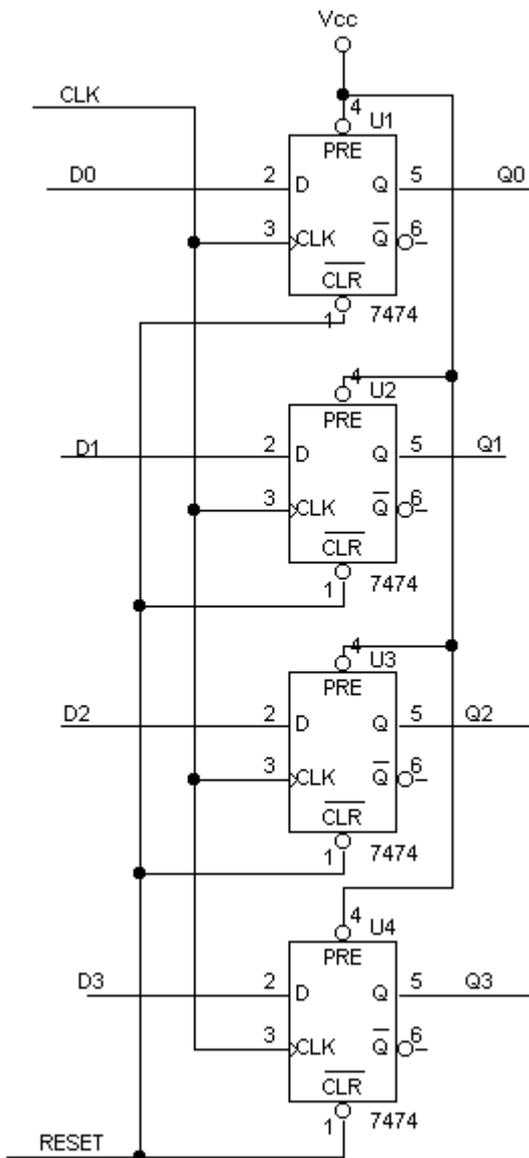
On l'utilise le plus souvent pour stocker temporairement des informations.

Il est défini par :

- sa capacité en bits qui dépend du nombre de bascules qu'il intègre (nombre de bits stockés à un instant donné)
- son mode de travail :

- écriture à décalage
- entrée en parallèle ou série
- sortie en parallèle ou série

D'un point de vue structurel, on obtient un registre de n bits en associant n structures de mémorisation unitaire (bascules). Selon le type de registre, ces bascules sont associées en cascade ou en parallèle.



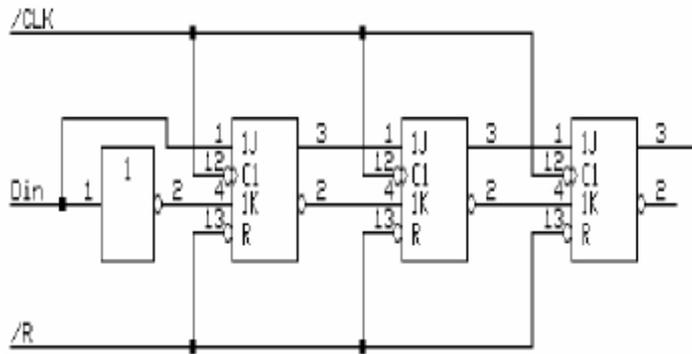
Dans l'exemple ci-contre, la 4 bits D0-D3 sont mémorisés lors du front montant de CLK et directement disponibles en Q0-Q3.

RESET permet de remettre à zéro le contenu du registre.

Registre à décalage

Dans un registre à décalage, l'information est mémorisée ou restituée de manière sérielle (bit par bit).

Exemple d'un registre série-série de 3 bits



Compteurs

Rappels

Système de numération

Tout système de numération repose sur une équation générale :

$$N = a_n \cdot B^n + a_{n-1} \cdot B^{n-1} + \dots + a_2 \cdot B^2 + a_1 \cdot B^1 + a_0 \cdot B^0$$

ainsi :

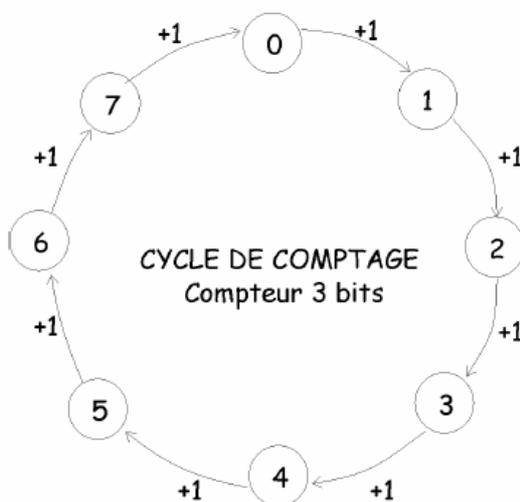
1011 en binaire correspond à $1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 11$ en décimal.

Progression binaire

De la relation précédente, on peut élaborer la progression du code binaire (voir tableau ci-contre).

Un mot binaire est une association de bits qui forme une valeur numérique.

Poids	3	2	1	0
Valeur	$2^3=8$	$2^2=4$	$2^1=2$	$2^0=1$
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1



Les compteurs

Le compteur matérialisé sous forme de circuit intégré à une capacité de comptage qui dépend du nombre de ses sorties.

Si le compteur possède n sorties, le dispositif est capable de dénombrer 2^n événements; soit de 0 à $2^n - 1$. L'évènement qui suit la valeur $2^n - 1$ fait repasser le compteur à 0.

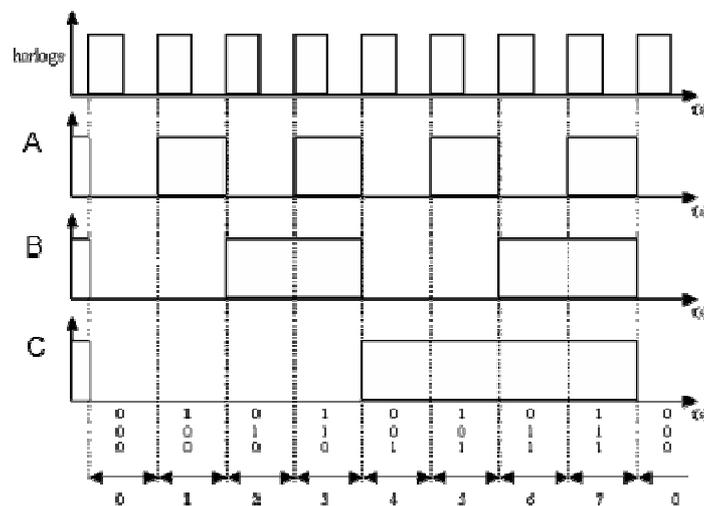
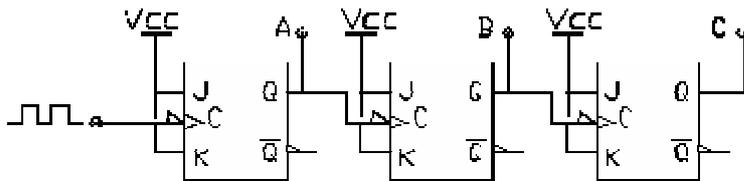
En observant l'évolution du code binaire naturel, on constate un rapport de 2 entre le changement d'un poids binaire et du suivant. On peut en conclure qu'une structure de comptage binaire permet également de réaliser une division de fréquence.

Les compteurs intégrés

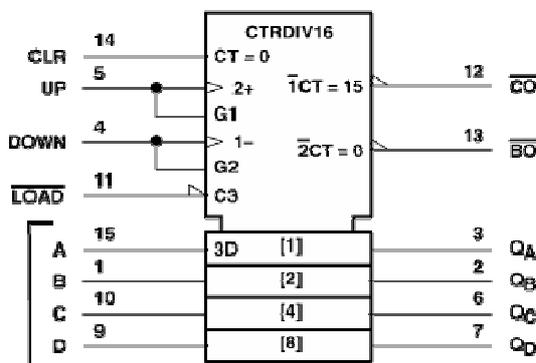
Les compteurs électroniques intégrés reposent sur des bascules placées en cascade.

Chaque bascule réalise une division par deux de son signal d'horloge.

Et la sortie de chaque bascule constitue une sortie du compteur.



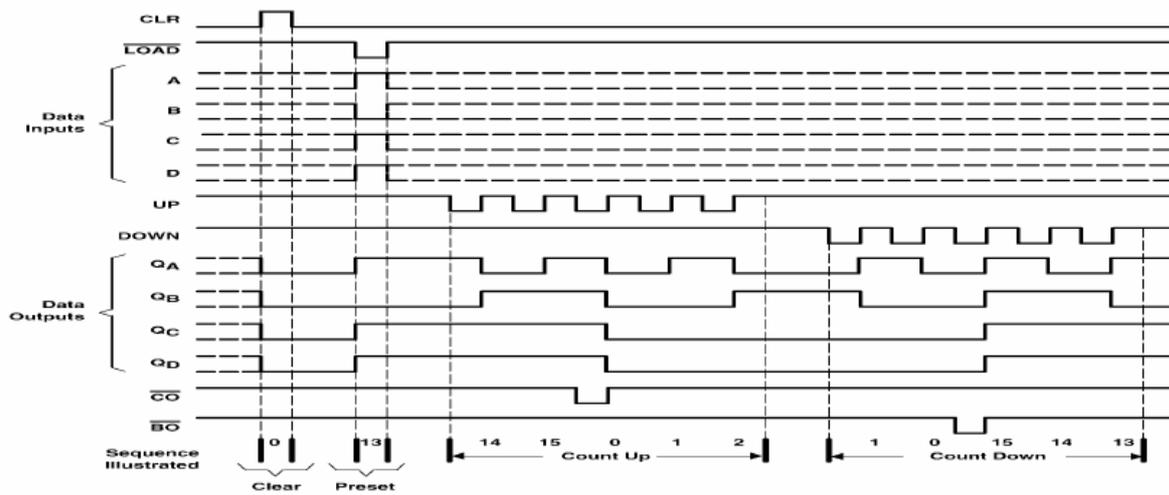
Exemple de compteur logique intégré



Ce compteur intégré peut être préchargé par une valeur logique appliquée en ABCD et validée par le signal /LOAD.

UP et DOWN permet de spécifier si le compteur doit incrémenter sa sortie (Up) ou décrémenter (Down) lors du front actif.

CLR permet d'initialiser le compteur.



Lycée Théodore DECK - NB2012