

I22 — Architecture des ordinateurs et système d'exploitation

J. Razik

Dpt. Informatique — UTLN

2020–2021

Logique Séquencielle

Diagramme d'états

Automate tel que :

- Chaque noeud représente un état du système ;
- La valeur de la sortie externe du système est écrite à côté ou dans le noeud ;
- Les arcs entre les noeuds sont orientés et possèdent comme étiquette la valeur des entrées du système source de cette transition.

Selon les transitions de sortie d'un état, celui-ci peut être qualifié de :

- **stable**, si pour une entrée de transition entrante, le système reste dans le même état pour la même entrée ;
- **transitoire**, si pour une entrée de transition entrante, le système change aussitôt d'état ;
- **terminal**, s'il ne possède pas de transition de sortie.

Diagramme d'états

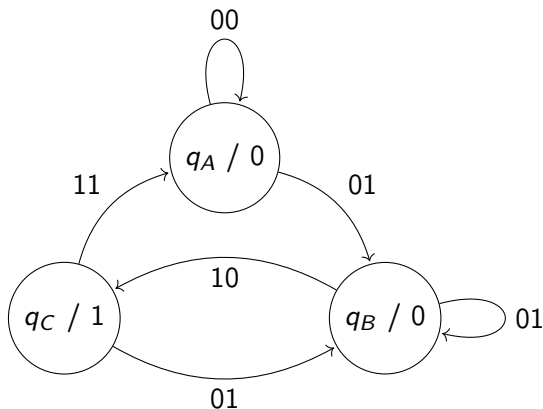


Table de transition

Construite à partir du diagramme d'états

- Chaque ligne correspond aux états et ses états destination ;
- Chaque colonne correspond à une combinaison possible des entrées ;
- La dernière colonne indique la valeur de sortie du système pour l'état considéré ;
- Intersection ligne-colonne : état destination à partir de l'état considéré (ligne) et avec la combinaison d'entrée lue (colonne). Si aucune transition spécifiée alors l'état est non spécifié, noté « - ».

État Entrée	00	01	11	10	Sortie
A	A	B	-	-	0
B	-	B	-	C	0
C	-	B	A	-	1

Réduction de l'automate

Recherche des états compatibles :

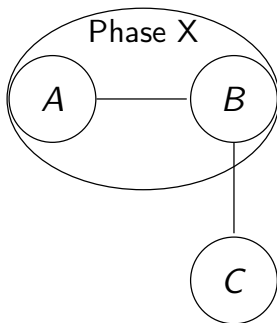
- La sortie des deux états la même (ils émettent la même réponse) ;
- Les transitions des deux états pour une entrée doivent être « identiques », c'est-à-dire soient :
 - ▶ Vers le même état ;
 - ▶ Vers l'un ou l'autre des deux états considérés ;
 - ▶ Une des deux transitions au moins est vers un état indéterminé.

Exemple :

États	0	1	2	3	4	5	Sortie
A	A	B	E	B	F	-	42
B	A	-	-	B	F	G	42
C	-	B	K	-	F	G	42

Diagramme de phases

Diagramme de compatibilité entre états

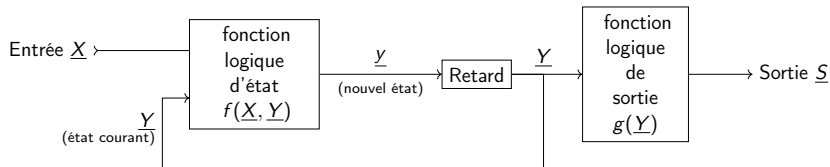


Réduction de l'automate

Fusion des états compatibles d'une même phase

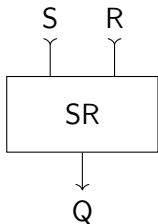
États	0	1	2	3	4	5	Sortie
X	X	X	E	X	F	G	42
C	-	X	K	-	F	G	42

Machine de Moore

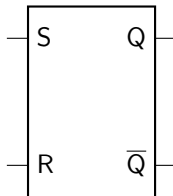


$$\begin{aligned}\underline{y} &= f(\underline{X}, \underline{Y}) \\ \underline{Y} &= \underline{y} \\ \underline{S} &= g(\underline{Y})\end{aligned}$$

Bascule SR — *Set-Reset*

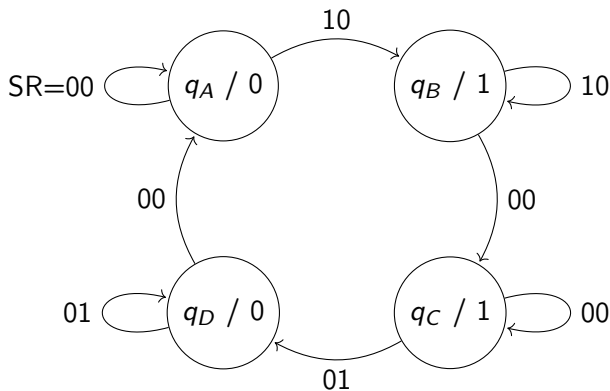


(a) Bascule SR.



(b) Bascule SR complète.

Bascule SR



États-Entrées / <i>SR</i>	00	01	11	10	Sortie <i>Q</i>
<i>A</i>	<i>A</i>	-	-	<i>B</i>	0
<i>B</i>	<i>C</i>	-	-	<i>B</i>	1
<i>C</i>	<i>C</i>	<i>D</i>	-	-	1
<i>D</i>	<i>A</i>	<i>D</i>	-	-	0

Bascule SR

États / Entrées SR	00	01	11	10	Sortie Q
U	U	U	-	V	0
V	V	U	-	V	1

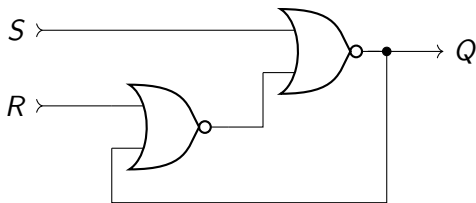
$$y = S + Y.\overline{R}$$

		SR			
		00	01	11	10
Y	0	0	0	-	1
	1	1	0	-	1

Bascule SR

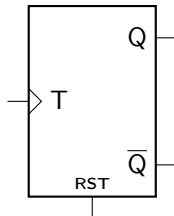
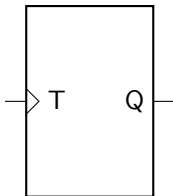
Autre réalisation avec que des portes *NOR*

$$y = \text{NOR}(\text{NOR}(Y, S), R)$$

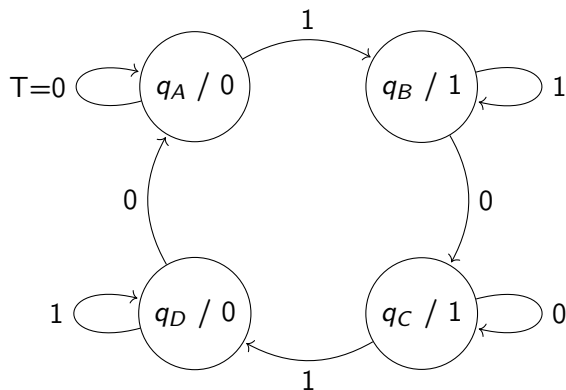


Bascule T — Toggle

Rôle : inverser la valeur de sa sortie à chaque fois que l'entrée T passe à 1.



Bascule T

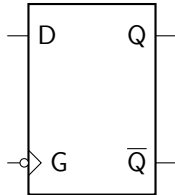


Bascule L — Latch

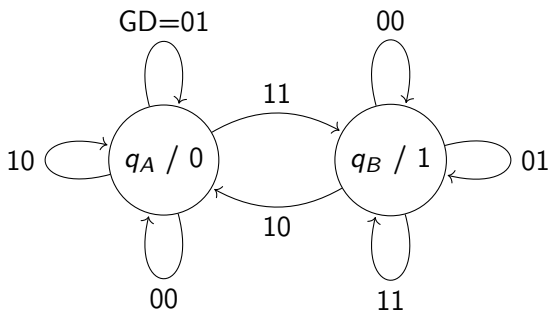
2 entrées : D et G

- Si G vaut 1, alors Q vaut l'entrée D ;
- Si G **pass**e à 0, la sortie conserve la dernière valeur de D avant que G ne passe à 0.

Mémorisation de la valeur de D sur le front descendant de G .



Bascule L



États / Entrées GD	00	01	11	10	Sortie Q
0 (A)	0	0	1	0	0
1 (B)	1	1	1	0	1

D'où l'équation avec aléa :

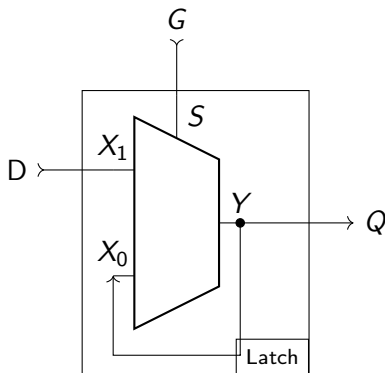
$$y = G.D + Y.\overline{G}$$

et sans aléa :

$$y = G.D + Y.\overline{G} + Y.D$$

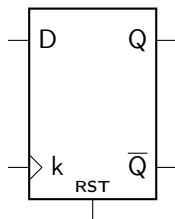
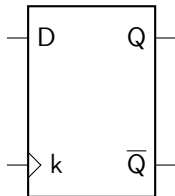
Bascule L

Réalisation possible à l'aide d'un multiplexeur

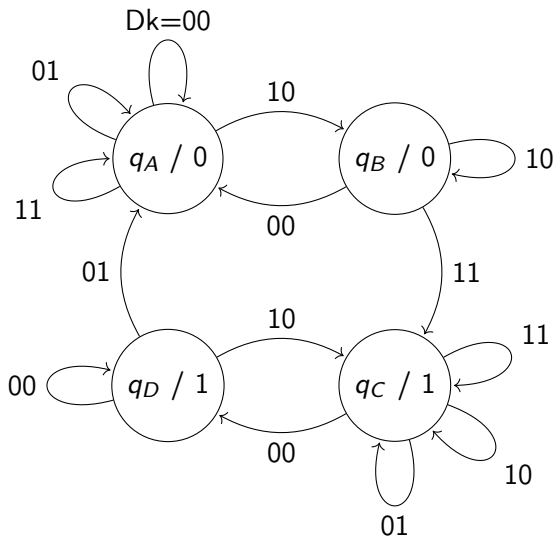


Bascule D — Data

2 entrées : D et k (horloge - *clock*). Mémorisation de D sur front montant.



Bascule D — Data

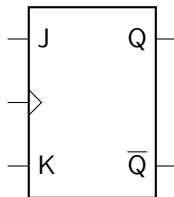


Bascule JK

3 entrées et 2 sorties : J , K et H (horloge) avec mémorisation sur front montant.

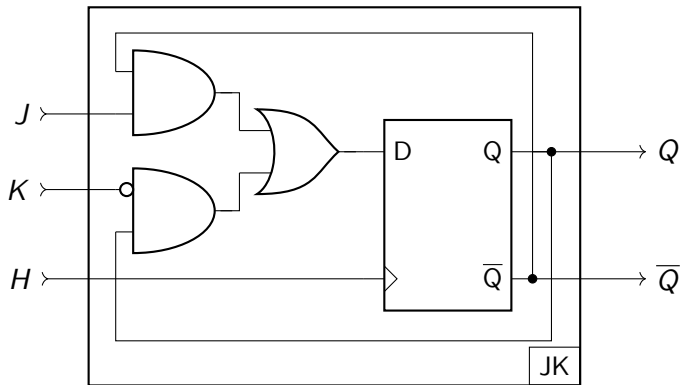
Mélange entre bascule SR et bascule T.

H	J	K	Q
0	-	-	Q
1	-	-	Q
↑	0	0	Q
↑	0	1	0
↑	1	0	1
↑	1	1	\overline{Q}

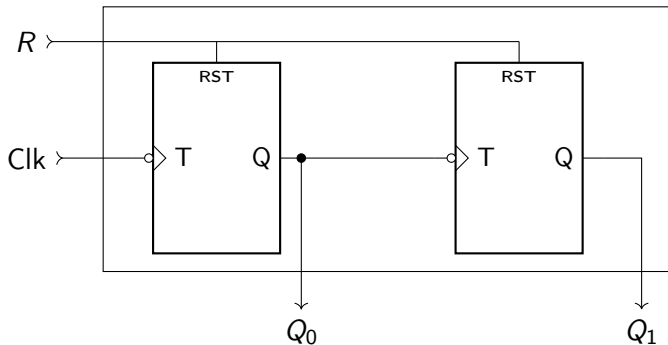


Bascule JK

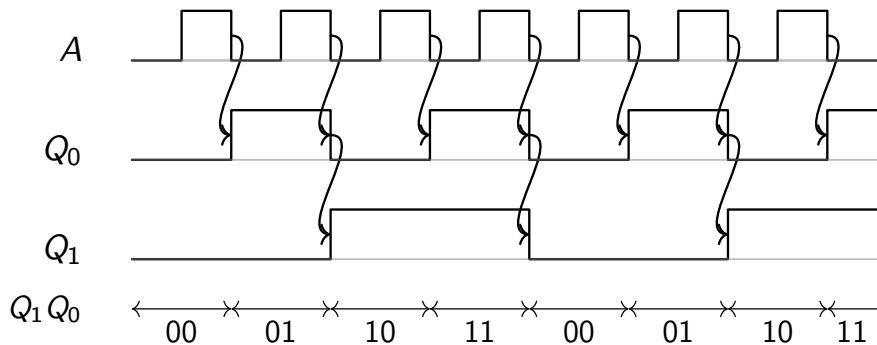
Réalisation à partir d'une bascule D



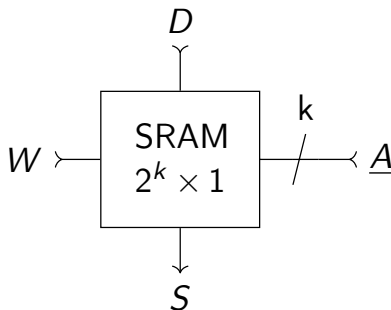
Compteur Asynchrone



Compteur asynchrone



Mémoire statique (SRAM)



Fonctionnement :

- Si $W = 1$ et $\underline{A} = n$, alors $Q_n = D$;
- Si $W = 0$ et $\underline{A} = n$, alors $S = Q_n$.

Mémoire statique

