Chapitre 2 : Circuits Séquentiels Et Synthèse D'automates

II. LES AUTOMATES

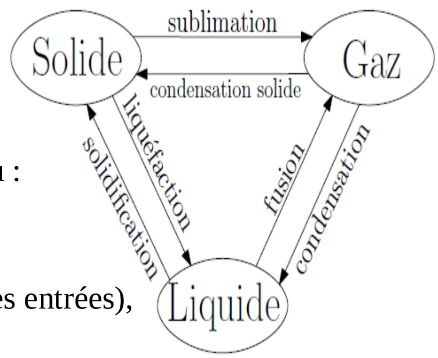
Une machine à état finis (automate) est une machine séquentielle algorithmique caractérisée par un vecteur d’entrées, un vecteur de sorties, et une séquence d’états définissant son comportement.

Tout circuit séquentiel peut être modélisé formellement par un automate. Inversement, "effectuer la synthèse d'un automate" est "réaliser un circuit séquentiel". Après avoir défini un automate,

Exemple Automate états transition :

États : états physiques de l’eau

Transitions : conditions de températures et pression



II.1. Définition Formelle

Un automate A est défini par un quintuplet (Q , i , E ,S , δ) où :

- Q est l'ensemble des états de l'automate,

-I est l’état initial

- E est son vocabulaire d'entrée (ensemble des valeurs possibles des entrées),

- S est son vocabulaire de sortie

- δ est l'ensemble des transitions ⊂ Q × E--> Q× Q

II.2. Représentation d’un automate

5.2. A. GRAPHE D'ETATS

Un automate peut être spécifié par un graphe d'états où :

- chaque sommet représente un état,

- chaque arc représente une transition d'un état à un autre, et est étiqueté par l'élément du vocabulaire d'entrée qui conditionne cette transition,

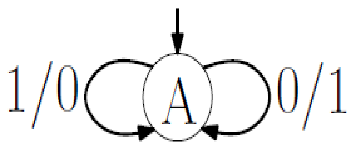
- les sorties sont associées aux arcs (automate de Mealy) ou aux états (automate de Moore).

Exemple

Dans l’exemple il y a deux transitions :

$A \xrightarrow{1/0} A$

$A \xrightarrow{0/1} A$



$A \xrightarrow{x/y} B$ signifie que si on lit x dans l’état A alors on écrit y et on passe à l’état B.

On a dans ce cas □ (A, x) = (B,y).

On met un flèche pour indiquer l’état initial (ici A)

Exemple

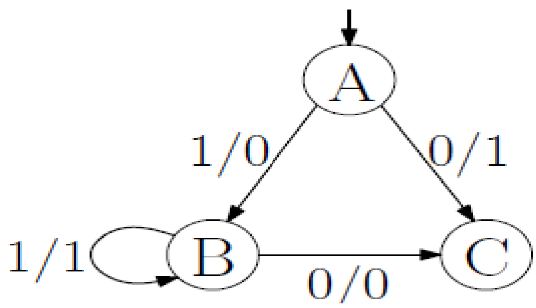
Ensemble d’´etats : Q = {A,B,C}

L’état initial : i = A

Alphabet d’entrée : E = {0, 1}

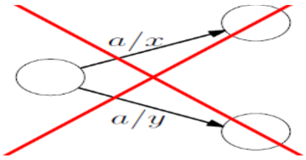
Alphabet de sortie : S = {0, 1}

Fonction de transition δ :

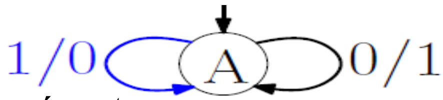


δ (A, 0) = (C, 1)
δ (A, 1) = (B, 0)
δ (B, 0) = (C, 0)
δ (B, 1) = (B, 1)

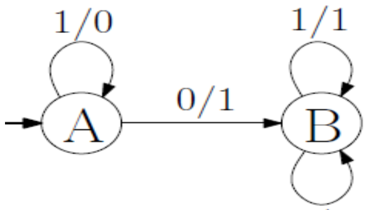
Remarque On ne peut pas avoir deux transitions sortant d'un même état qui correspondent à la même lettre.



Exemple d'automate : inversion de bits
Lecture du mot 01011
Sortie : 10100



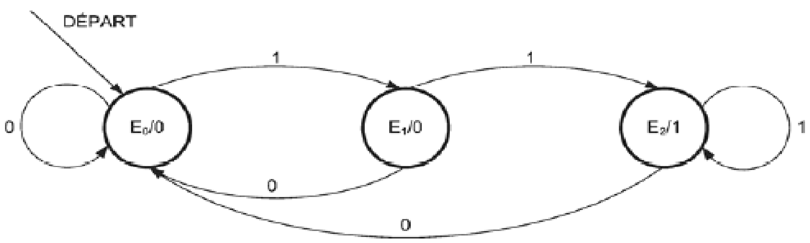
Exemple d'automate : incrémenteur
Lecture du mot 11001
Sortie : 00101



II .3.Machine de Moore et Mealy

Pour représenter ces automates, qu'ils soient matériel ou logiciel, il existe deux architectures différentes : la machine de MOORE et la machine de MEALY.

II.3.1. Machine de Moore - on dit que la machine est une machine **de Moore** lorsque la sortie dépend uniquement de l'état courant(les sorties à l'intérieur de l'état), Le nom de machine de Moore fait référence au professeur Edward F. Moore de l'université de Winsconsin- Madison aux États-unis qui en est l'inventeur.
C'est à dire Les conditions (valeurs du vecteur d'entrée) sont notées sur les arcs et la valeur des sorties est généralement à l'intérieur du cercle (séparée du nom de l'état par un trait oblique : /).
Voici un exemple de machine à états finis où les sorties sont indiquées à l'intérieur des cercles :



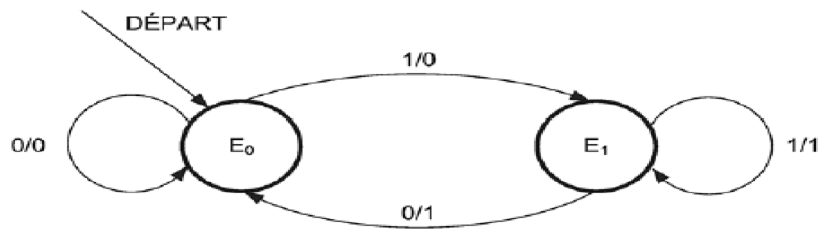
Voici les éléments que l'on y trouve :

- Trois états notés E0, E1 et E2 ;
- un état de départ E0 ;
- une entrée (notée sur les arcs) ;
- une sortie associée à chacun des trois états, respectivement 0, 0 et 1.

Cette machine permet de détecter une séquence de deux 1 consécutifs en entrée. Considérons en effet son fonctionnement :

- au départ, la machine est à E_0 et sa sortie à 0 ;
- tant que la machine ne reçoit pas 1, elle demeure à E_0 et sa sortie à 0 ;
- si la machine reçoit 1, elle change d'état et part vers E_1 : le début de la séquence 11 est détecté. La sortie demeure à 0 ;
- si à cette étape, la machine reçoit en entrée 0, la séquence est brisée. La machine revient à E_0 et tout reprend depuis le début ;
- si la machine reçoit 1 alors qu'elle se trouve en E_1, elle transite vers E_2 et sa sortie est alors à 1 (pour indiquer qu'elle a reçu la séquence 11) ;
- une fois en E_2, si la machine reçoit 1, elle demeure dans le même état car la séquence (entrée précédente - entrée actuelle) est encore 11 : la sortie est à 1 ;
- autrement elle part en E_0 et tout recommence depuis le début.
- au départ, la machine est à E0 et sa sortie à 0 ;
- tant que la machine ne reçoit pas 1, elle demeure à E0 et sa sortie à 0 ;
- si la machine reçoit 1, elle change d'état et part vers E1 : le début de la séquence 11 est détecté. La sortie demeure à 0 …………………etc

II.3.2.Machine de Mealy: on dit que la machine est une machine de Mealy lorsque - la sortie dépend de l'état courant et de l'entrée(les sorties sur les arcs), Le nom de machine de Moore fait référence à G.H. Mealy qui les fit connaître dans un article au Bell System Tech en 1955. C a dire Les conditions (valeurs du vecteur d'entrée) sont notées sur les arcs et la valeur des sorties est généralement indiquée sur l'arc (séparée des entrées par un trait oblique : /)
Voici un exemple de machine à états finis où les sorties sont indiquées sur les arcs :



Voici les éléments que l'on y trouve :

- deux états notés E0 et E1 ;
- un état de départ E0 ;
- une entrée (notée sur les arcs) ;
- des sorties associées aux transitions, 0 pour les transitions de E0 à E0 et de E0 à E1, 1 pour

  les transitions de E1 à E1 et de E1 à E0.

Cette machine (aussi) permet de détecter une séquence de deux 1 consécutifs en entrée. Considérons son fonctionnement :

- Au départ, la machine est à E0;
- Tant que la machine ne reçoit pas 1, elle demeure à E0 et sa sortie à 0 ;
- Si la machine reçoit 1, elle change d'état et part vers E1 : le début de la séquence 11 est détecté. La sortie demeure à 0 mais elle est ici associée à la transition;

- Si à cette étape, la machine reçoit en entrée 0, la séquence est brisée. La machine revient à E0 et tout reprend depuis le début ;
- Si la machine reçoit 1 et qu'elle se trouve en E1, elle demeure en E1 et sa sortie est alors à 1 (pour indiquer qu'elle a reçu la séquence 11). Elle reste à cet état tant et aussi longtemps que l'entrée est 1;
- Autrement elle part en E0 et tout recommence depuis le début.

Comme on le voit, dans sa conception, cette machine est différente de la première, mais dans son fonctionnement global, elle lui est identique.

II.3.3.Convertir une Machine de Moore en Mealy

Les machines de Mealy sont donc plus économiques que les machines de Moore, tandis que les machines de Moore sont plus simples à concevoir.
Il est cependant facile de passer d'une machine de Moore à une machine de Mealy en respectant quelques règles simples :
1. Reporter les sorties sur les arcs arrivant vers cet état
2. Simplifier le circuit si la simplification est possible

II.4. Conception d'une machine de Moore.

Les étapes à suivre pour concevoir une machine à états finis selon le modèle de la machine de **Moore.**à partir d'un cahier des charges sont :
1. Dessiner le diagramme des états
2. Poser la table des états
3. Définir la table de transition
4. Déterminer les expressions des entrées des bascules
5. Déterminer les expressions des sorties
6. Faire le schéma

Exemple

Nous considérerons un séquenceur délivrant une séquence de 8 états successifs correspondant à ceux d'un compteurs binaire par 8. Supposons également que ce séquenceur dispose d'une entrée C permettant à tout moment de revenir de manière synchrone dans l'état initial (000). Ce système peut se mettre sous la forme générale d'un système séquentiel synchrone pour lequel les sorties sont directement les sorties Si sont directement les variables d'état Qi (sorties des bascules)

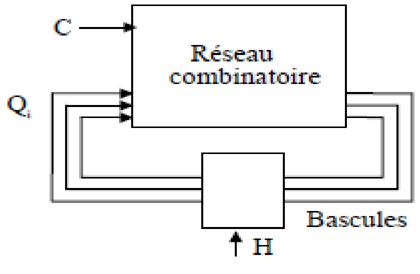


Figure *Schéma général d'un séquenceur*

1. *Dessiner le diagramme des états (graphe d'état)*

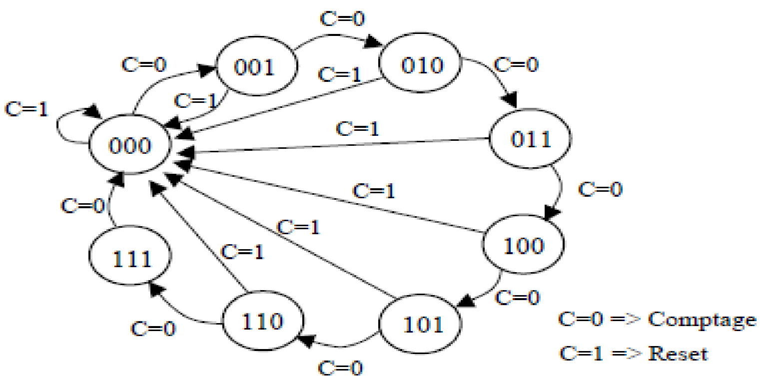


Figure . *Graphe de d'état*

2. *Poser la table des états*

Ce graphe peut également s'exprimer sous une forme tabulaire. Cette représentation, appelée table d'état

Etats	Etats Suivants	
	C = 0	C = 1
000	001	000
001	010	000
010	011	000
011	100	000
100	101	000
101	110	000
110	111	000
111	000	000

Figure *Table d'état*

3. *Définir la table de transition*

Supposons maintenant que pour réaliser ce système nous prenions des bascules D. La table de transition d'une telle bascule est la suivante

D(n)	Q(n+1)
0	0
1	1

Tables de transition des bascules D

Pour chaque bascule i nous connaissons l'état suivant Qi(n+1) (après le coup d'horloge) en fonction de l'état présent Qi(n) et de l'entrée C. Il reste donc à déterminer les entrées Di de chaque bascule à partir des tables de transition correspondantes

Etats $Q_2Q_1Q_0(n)$	Etats Suivants $Q_2Q_1Q_0(n+1)$ C = 0	C = 1	C = 0 $D_2(n)$	$D_1(n)$	$D_0(n)$	C = 1 $D_2(n)$ $D_0(n)$		$D_1(n)$
000	001	000	0	0	1	0	0	0
001	010	000	0	1	0	0	0	0
010	011	000	0	1	1	0	0	0
011	100	000	1	0	0	0	0	0
100	101	000	1	0	1	0	0	0
101	110	000	1	1	0	0	0	0
110	111	000	1	1	1	0	0	0
111	000	000	0	0	0	0	0	0

Figure *Entrées des bascules*

4. *Déterminer les expressions des sorties*

Q_2Q_1 \ Q_0C	00	01	11	10
00				
01				1
11	1			
10	1			1

$D_2 = C'.(Q_2' \oplus Q_1.Q_0)$

Q_2Q_1 \ Q_0C	00	01	11	10
00				1
01	1			
11	1			
10				1

$D_1 = C'.(Q_1 \oplus Q_0)$

Q_2Q_1 \ Q_0C	00	01	11	10
00	1			
01	1			
11	1			
10	1			

$D_0 = C'.Q_0'$

Figure *Synthèse des entrées de bascules*

5. *Faire le schéma*

Ayant les équations des entrées des bascules le circuit peut être réalisé

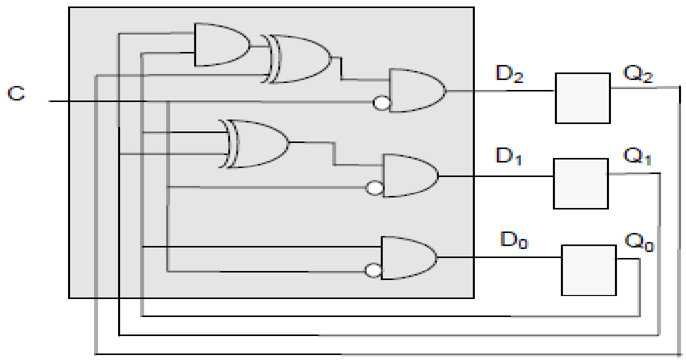


Figure . *Schéma logique*

II.5. Conception d'une machine de Mealy

La méthode d'Huffman-Mealy se décompose en plusieurs étapes.

Ces étapes sont les suivantes :

1 : Modélisation du cahier des charges

- Graphe d'état
- Table d'état

2 : Minimisation du nombre d'états

- Règles de minimisation
- Détermination du nombre de bascules minimum

3 : Codage

- Codage des états
- Codage des entrées de bascules

4 : Synthèse

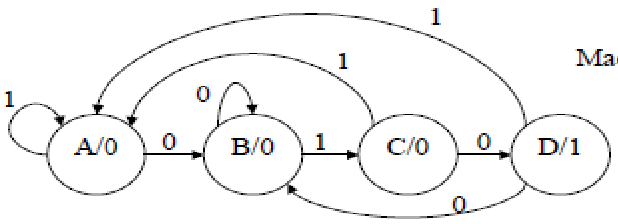
- Synthèse des entrées de bascules et des sorties de la machine
- Implantation technologique (mapping)

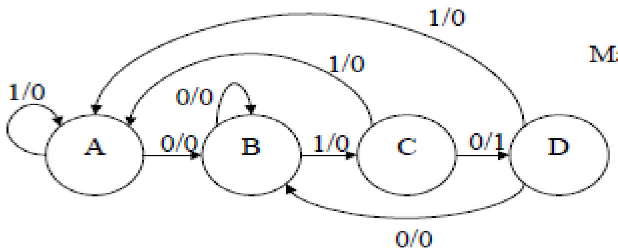
II.5.1. Modélisation du cahier des charges

Le cahier des charges d'un système est généralement donné en langage courant.
Exemple : Le système considéré a une entrée (E) et une sortie (S). Il reçoit sur son entrée des bits arrivant en série. La sortie (S) doit passer à 1 chaque fois qu'une séquence 010 apparaît sur l'entrée (E) puis repasser à 0 sur le bit suivant quel que soit sa valeur.

II.5.1.a. Graphe d'état

Selon que le système sera réalisé sous forme de machine de Moore ou sous forme de Machine de Mealy, le graphe d'état représentant le cahier des charges précédent est représenté sur la Figure





Remarque : La structure des deux graphes paraît identique. En fait, il est toujours possible de passer d'un graphe de Moore à un graphe de Mealy. Pour cela il, suffit de reporter les sorties associées à chaque état (Moore) sur les arcs arrivant à chacun de ces états. Nous verrons par la suite que l'inverse, c'est à dire passer d'un graphe de Mealy à un graphe de Moore n'est pas toujours possible.

II.5.1.b. Table d'état

Etat s	Etats Suivants E=0	Etats Suivants E=1	Sortie
A	B	A	0
B	B	C	0
C	D	A	0
D	B	A	1

Machine de Moore

Etat s	Etats Suivants E=0	Etats Suivants E=1	Sortie E=0	Sortie E=1
A	B	A	0	0
B	B	C	0	0
C	D	A	1	0
D	B	A	0	0

Machine de Mealy

II.5.2. Minimisation du nombre d'états

Le nombre d'états de la machine influe directement sur le nombre de bascules nécessaires pour réaliser ce système. Or, le nombre d'états utilisés pour représenter le cahier des charges, que ce soit sur le graphe d'état ou sur la table d'état, n'est pas nécessairement minimum.

II.5.2.a. Règles de minimisation

- **Règle R1 :** Deux états sont équivalents si pour chaque combinaison d'entrée, ils ont mêmes sorties et mêmes états suivants.
- **Règle R2 :** Les états sont regroupés en différentes classes selon les valeurs de sorties associées. Ainsi, deux états ayant mêmes sorties (pour chaque combinaison d'entrée) sont dans la même classe. Les états appartenant à une même classe sont équivalents s'il ne peuvent être séparés. Or les états appartenant à une même classe doivent être séparés si les états suivants associés à chacun d'eux sont dans des classes différentes.

Lorsque plusieurs états sont équivalents, il suffit de garder qu'un seul représentant par classe d'équivalence et de renommer les états suivants en conséquence.

Exemple 1: Les règles de minimisation appliquées à la table d'état de la machine de Mealy précédente donnent :

R1 : A et D on mêmes sorties et mêmes états suivants, ils sont donc équivalents. L'état D peut par exemple être éliminé. En renommant les états suivants en conséquence, c'est à dire en remplaçant D par A, la table d'état devient :

Etats	Etats Suivants		Sortie	
	E=0	E=1	E=0	E=1
A	B	A	0	0
B	B	C	0	0
C	A	A	1	0

Table d'état réduite

Au sens de la règle R1 il n'y a pas d'autres états équivalents.

R2 : les états peuvent être regroupés en deux classes (classe 1 et classe 2).

(1)	(2)	Classes
(A , B)	(C)	Etats
BA BC		Etats suivants
11 12		Classes des états suivants

Les états A et B doivent être séparés. Il y a maintenant qu'un seul état par classe. Il n'y a donc plus d'états équivalents. Cette machine peut être réalisée avec 3 états.

Le nombre d'états nécessaire à la réalisation d'une machine de Mealy pouvant être inférieur à celui nécessaire à la réalisation d'une machine de Moore, le nombre de bascules peut l'être également. D'où l'avantage qu'il peut y avoir à réaliser une machine de Mealy plutôt qu'une machine de Moore. Ceci dit, les machines de Mealy peuvent avoir des inconvénients liés au fait que les sorties dépendent directement des entrées. En effet, lors du passage d'un état à un autre, les entrées ne doivent pas varier. Il se produit donc un instant entre le changement d'état

et le changement d'entrée ou le système se trouve dans le nouvel état mais en présence de l'entrée ayant conduit à cet
état, c'est à dire de l'entrée précédente. Puisqu'en machine Mealy, les sorties dépendent directement de l'état et des entrées, elles peuvent donc être soumise à des commutations parasites.

II.5.2.b. Détermination du nombre de bascules minimum

Le nombre minimum d'états "q" étant déterminé, on peut en déduire le nombre minimum "n" de variables
d'état et par conséquent de bascules nécessaires au codage de ces états à partir de la double inéquation suivante :

$2^{n-1} < q < 2^{n}$

Exemple : Pour la machine de Mealy précédente, le nombre minimum d'état étant de 3, le nombre de variables d'état nécessaire au codage de ces états est 2. Deux bascules sont donc nécessaires pour réaliser ces systèmes.

II.5.3. Codage

II.5.3.a. Codage des états

Chaque état peut être codé par une combinaison de ces n variables d'état. Chaque état doit avoir un code différent des autres mais le codage des états peut être quelconque. Notons toutefois que le codage influence la structure de la future machine et peut donc influencer sa complexité

Une fois les états codés, la table d'état peut être exprimée en fonction de ces codes.

Exemple : Nous appellerons les variables d'état (sorties des 2 bascules) de la machine détectant la séquence 010, Q1 et Q2. En considérant un codage donné (celui indiqué dans la colonne "Etats"), la table d'état codée de la machine de Mealy correspondant à la table d'état de la Figure ci dessus est représentée sur la Figure suivant

Etats $Q_1Q_2(n)$	Etats Suivants $Q_1Q_2(n+1)$		Sortie	
	E=0	E=1	E=0	E=1
00 (A)	01	00	0	0
01 (B)	01	11	0	0
11 (C)	00	00	1	0

Tables d'état codées de la machine de Mealy

II.5.3.b. Codage des entrées de bascules

Le code des états étant déterminé, les entrées de bascules peuvent l'être. Pour chaque bascule i nous connaissons l'état suivant Qi(n+1) (après le coup d'horloge) en fonction de l'état présent Qi(n) et des entrées. Pour réaliser ce système il reste à déterminer les entrées de chaque bascule.

Avec des bascules D, les entrées Di peuvent être déterminée directement à partir de la relation :

$D_i(n) = Q_i(n+1)$

Exemple : Reprenons la machine de Mealy précédente. Les entrées D1 et D2 des bascules Q1 et Q2 sont représentées sur le tableau

Etats $Q_1Q_2(n)$	Etats Suivants $Q_1Q_2(n+1)$		Sortie		Entrées Bascules $D_1D_2(n)$	
	E=0	E=1	E=0	E=1	E=0	E=1
00 (A)	01	00	0	0	01	00
01 (B)	01	11	0	0	01	11
11 (C)	00	00	1	0	00	00

Détermination des entrées de bascules

II.5.4. Synthèse

II.5.4.a. Synthèse des entrées de bascules et des sorties de la machine

Sur la table précédente on dispose des sortie et entrées de bascules exprimées en fonction des entrées et des variables d'état (sorties des bascules). Il suffit donc maintenant d'exprimer les fonction logiques relatives aux sorties et entrées de bascules.

Exemple : Reprenons la machine de Mealy précédente et déterminons les équations de la sortie S et des entrée de bascules D1 et D2

Q_1Q_2 \ E	0	1
00	0	0
01	0	1
11	0	0
10	ϕ	ϕ

$D_1=$ $E.Q_1$

'.Q_2

Q_1Q_2 \ E	0	1
00	1	0
01	1	1
11	0	0
10	ϕ	ϕ

$D_2= E'.Q_2' + Q_1'Q_2$

Q_1Q_2 \ E	0	1
00	0	0
01	0	0
11	1	0
10	ϕ	ϕ

$S= E'.Q_1$

Synthèse de D1 et D2

II.5.4.b. Implantation technologique

L'implantation technologique de fonctions logiques est un problème en soit qui sera traité dans un chapitre spécifique. Nous pouvons nous contenter ici, d'une implantation à portes obtenue directement à partir de ces équations déterminée précédemment

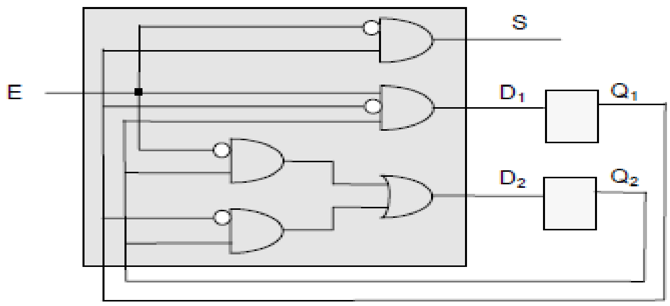


Schéma du détecteur de séquence 010