SYNTHESE
D'AUTOMATES D'ETATS FINIS

- Automate d'états finis (vu dans d'autre enseignement ?)
- Spécification très utilisée en Informatique (Circuits, Réseaux, Preuve...)
- Réalisation matérielle d'un automate à l'aide de circuits
- Méthode automatique
- Logiciel de CAO effectue ce travail

Définition

- Un automate d'états finis A est un quintuplet (Q,E,S,t,o)
  - Q : ensemble des états
  - E : vocabulaire d'entrée (ensemble des valeurs possibles des entrées)
  - S : vocabulaire de sortie (ensemble des valeurs possibles des sorties)
  - t : fonction de transition : Q * E -> Q
  - o : fonction de sortie :
    - 2 types d'automate
      - **Mealy : Q * E -> S**
      - **Moore : Q -> S**

Equivalence d'automates

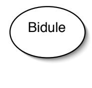
- Deux automates sont équivalents si pour toute séquence d'entrées ils donnent la même séquence de sorties

- **Automate minimal**: automate tel qu'il n'existe pas d'automate équivalent comportant un nombre d'état plus petit

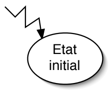
Représentation

- **Graphe d'états**

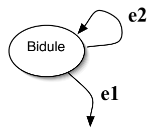
- Un état : un rond et un nom à l'intérieur



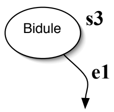
- Etat initial : une flèche tordue



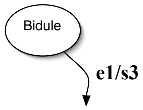
- Une transition : une flèche portant la valeur d'entrée associée à cette transition



- Une sortie :
  - soit associée à l'état (Moore)



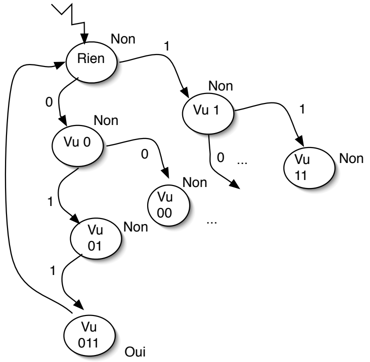
  - soit associée à une transition (noté e/s) (Mealy)

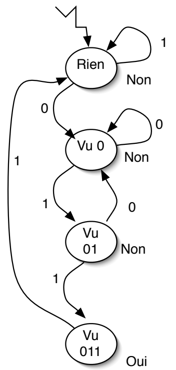


Exemple de graphe d'automate

Reconnaisseur de séquence

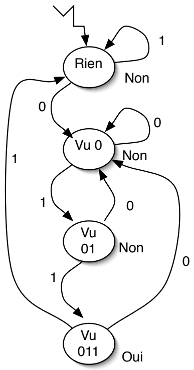
• E={0,1}, S={oui, non}, On veut reconnaître la séquence 011



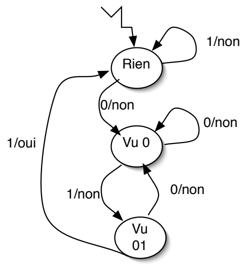


Exemple de graphe d'automate

Automates minimaux du reconnaisseur de la séquence 011



Moore



Mealy

Réalisation d'un automate à l'aide de circuits digitaux

• Codage binaire

• Les entrées et les sorties

- Exemple : S={a,b,c}
- Codage avec le nombre minimum de bits : compact
- Exemple :

	c1	c2
a	0	0
b	0	1
c	1	0

Codage des états

• Les états

Les bits de code sont appelés les variables d'états

Deux types de codage utilisés

- Codage compact (minimum de bits)
- Codage 1 parmi n

Une variable par état. Pour chaque état une seule variable d 'états à 1

•Exemple : pour 3 états E1, E2, E3

	q1,	q2,	q3
E1	0	0	1
E2	0	1	0
E3	1	0	0

Influence des codages binaires sur le réalisation

- Le choix de ces codages ne changent pas le comportement du circuit résultant
- Mais les fonctions booléennes ne seront pas les mêmes
- Ces codages ont donc une importance sur les qualités du circuit résultant: taille, vitesse...
- Les outils de CAO prennent en compte ces critères lors de la réalisation automatique d'automate

Mémorisation de l'état

- Utilisation d'une bascule pour mémoriser chaque variable d'état
- L'état suivant va dépendre de l'état courant

➡ Bascules sensibles au front

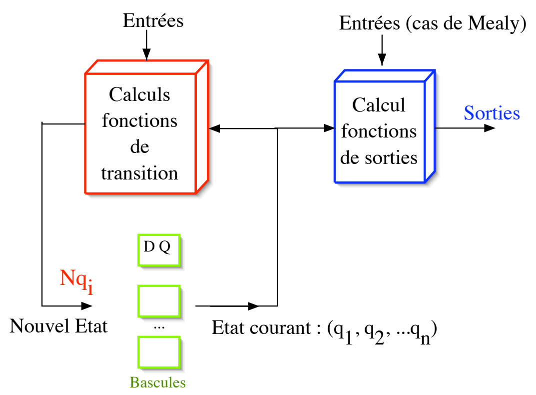
- Deux types de réalisation d'automate :
  - **Asynchrone** : c'est le changement des entrées qui provoque le changement d'état

    Délicat à mettre en œuvre, très peu utilisé
  - **Synchrone** : c'est une entrée particulière (appelée ***horloge***) qui provoque le changement d 'état. Les entrées ne doivent pas changer au moment des fronts montants de l'horloge.

On ne réalisera que des automates synchrones

L'horloge est donc envoyée sur la fonction d'activation de chaque bascule

Architecture générale du circuit réalisant un automate



Initialisation Reset/Set Bascule suivant le codage de l'état initial

Fonctions booléennes réalisées

• La fonction de transition Q*E ->Q

Une fonction booléenne par variable d'état

Nqi est la valeur de la variable d'état à "l'instant suivant" (nouvel état au prochain front de d'horloge)

➡ **Nq_i = fonction(q_j, e_j)**

Ces fonctions peuvent être phi-booléennes si tous les codes ne sont pas utilisés pour les entrées et/ou les états

• La fonction de sortie Q*E ->S ou Q-> S

Une fonction booléenne de sortie par variable de sortie

➡ **s_j=fonction(q_i, e_i) ou s_j= fonction(q_i)**

Exemple de réalisation

- Reconnaisseur de séquence (Mealy)
- **Codage des sorties**

Arbitraire :

Sortie	**s**
Oui	1
Non	0

Codage des entrées

Arbitraire :

Entrée	**e**
0	0
1	1

- **Codage des états**

Compact arbitraire :

Etat	**q1**	**q2**
rien	0	0
Vu 0	0	1
Vu 01	1	0

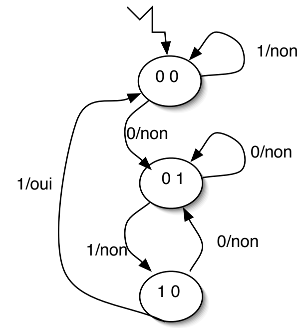
Initialisation par ***reset*** sur les 2 bascules (état initial= 0 0)

Exemple de réalisation

- **Fonctions de transitions et de sorties**

Nouvel état = **fonction booléenne** (Etat courant, entrée)
Sortie= **fonction booléenne** (Etat courant, entrée)

q_1	q_2	e	Nq_1	Nq_2	S
0	0	0	0	1	0
0	0	1	0	0	0
0	1	0	0	1	0
0	1	1	1	0	0
1	0	0	0	1	0
1	0	1	0	0	1



Phi booléen pour les cas où $(q_1, q_2)= (1, 1)$

Exemple de réalisation

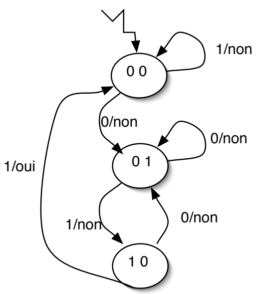
- **Fonctions de transitions**

e \ q_1/q_2	00	01	11	10
0	0	0	phi	0
1	0	1	phi	0

$Nq_1 = q_2 . e$

e \ q_1/q_2	00	01	11	10
0	1	1	phi	1
1	0	0	phi	0

$Nq_2 = \overline{e}$

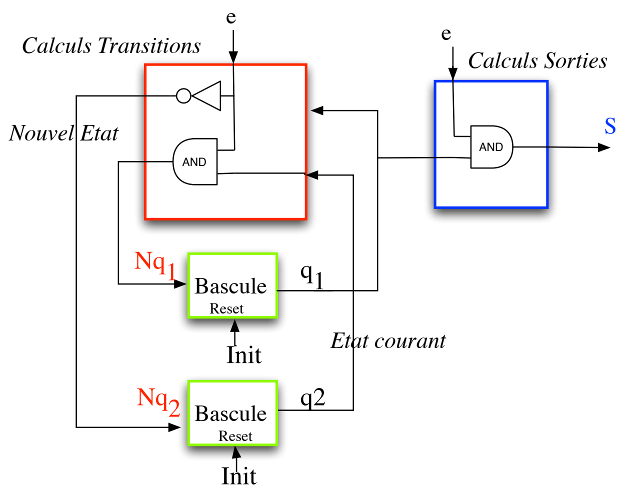


- **Fonctions de sortie**

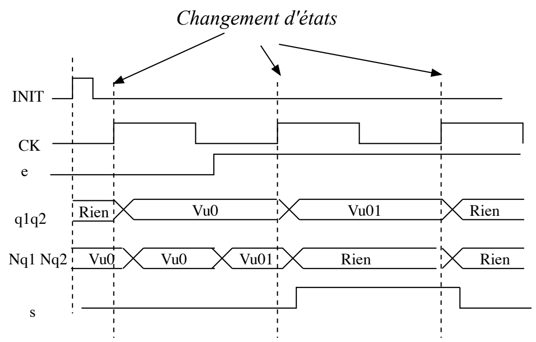
e \ q_1/q_2	00	01	11	10
0	0	0	phi	0
1	0	0	phi	1

$S = q_1 . e$

Exemple de réalisation



Evolution des états et sorties



• **Attention** certains changements des entrées peuvent ne pas être pris en compte, si la fréquence de l'horloge n'est pas assez élevée

Le codage 1 parmi n des états

• Une variable d'état par état, donc une bascule par état

• Plus de bascules, circuit “meilleur” (en surface par exemple) ou pas suivant les cas

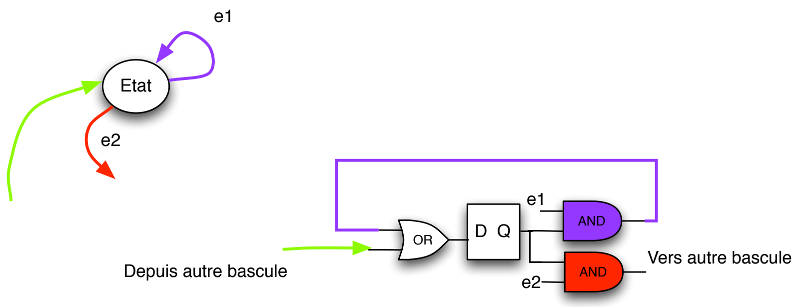
• Intérêt : pas de calcul par le concepteur

• Plutôt de donner les tables de vérité des variables d'état et des sorties

• On peut dessiner le circuit à partir de l'automate sans calcul

Le codage 1 parmi n des états

- Un état : une bascule
- une transition : un AND entre la sortie de la bascule et l'entrée
- En entrée d'une bascule : un OR des différentes transitions arrivant à cet état (sortie des AND)
- Exemple:

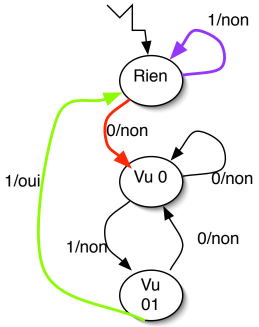


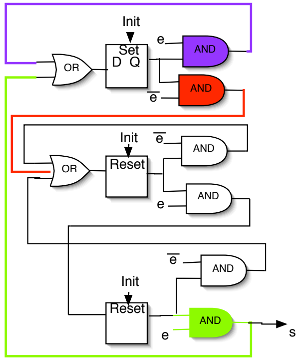
Le codage 1 parmi n des états

• Pour les sorties :
- Moore : un OR des sorties des états pour lesquels la sortie vaut 1
- Mealy : un OR des transitions pour lesquelles la sortie vaut 1

• Initialisation :
  - o Init : SET sur la bascule contenant l'état initial
    RESET sur toutes les autres bascules
  - o Si bascule sans SET et RESET
    - Entrée état initial= OR (transitions, Init)
    - Entrée Autre état = AND (transitions, Initbarre)

Exemple codage 1 parmi N





EXERCICE

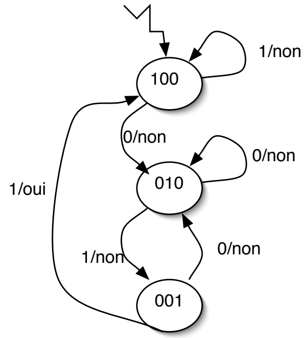
- Retrouvez le circuit précédent en faisant les calculs des fonctions booléennes

	q1, q2, q3
rien	1 0 0
Vu 0	0 1 0
Vu 01	0 0 1

- Beaucoup de phi-booléen
- Fonctions à 4 entrées
- Exemple:

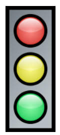
e	q1	q2	q3	Nq1
0	1	0	0	0
1	1	0	0	1
0	0	1	0	0
1	0	1	0	0
0	0	0	1	0
1	0	0	1	1

Autres valeur à Phi

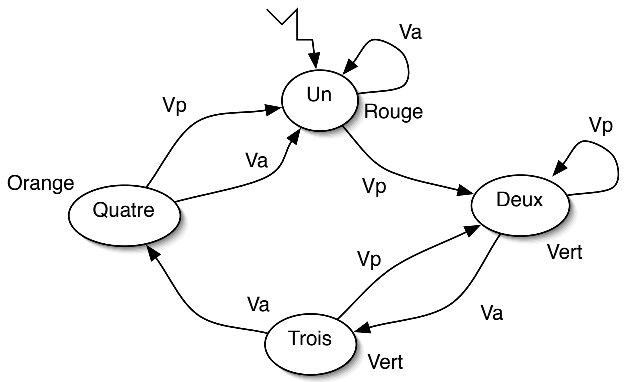


Exercice: commande de feu tricolore

- Les informations d'entrées sont : voiture-présente (vp), voiture-absente (va).
- La sortie est la couleur du feu : Vert (V), Orange(O) ou Rouge (R).
- Le comportement du système est le suivant:
  - ✓Au départ le feu est rouge.
  - ✓Si le feu est rouge :
    - ‣si une voiture est présente, le feu passe au vert
    - ‣sinon le feu reste rouge.
  - ✓Si le feu est orange le feu passe au rouge.
  - ✓Si le feu est vert :
    - ‣si une voiture est présente, le feu reste au vert ;
    - ‣si une voiture est absente deux fois de suite, le feu passe au orange.

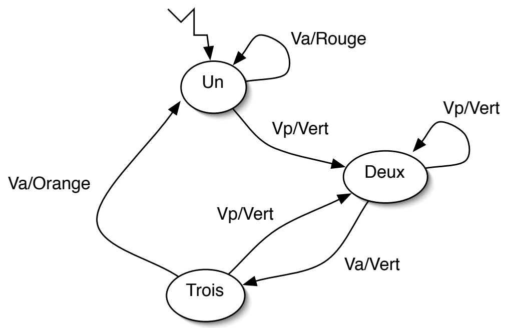


Commande de feu tricolore
Automate de Moore



Graphe de l'automate commande de feu (en Moore)

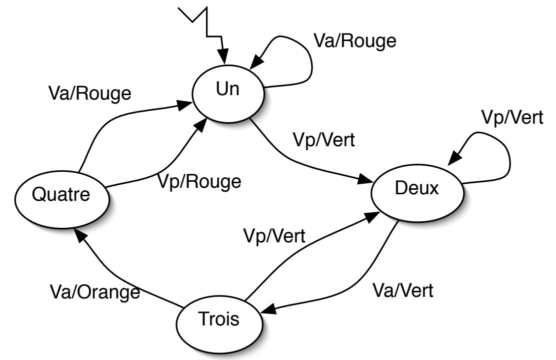
Commande de feu tricolore
Mealy 1



Graphe de l'automate commande de feu (en Mealy)

- Attention le feu ne passe pas forcément au rouge après le orange

Commande de feu tricolore
Automate Mealy 2



Graphe de l'automate commande de feu (en Mealy)

Cette fois on passe forcément par le rouge après le orange

Synthèse commande de feu tricolore

- **Codage compact des états**

Etat	q1, q2
Un	0 0
Deux	0 1
Trois	1 0
Quatre	1 1

- **Codage des sorties**

Sorties	S1 S2
Vert	0 0
Orange	0 1
Rouge	1 0

- **Codage des entrées**

Entrées	E
Va	0
Vp	1

Synthèse commande de feu tricolore

- **Fonctions de transition et de sortie (Moore)**

e	q1	q2	Nq1	Nq2	S1	S2
0	0	0	0	0	1	0
1	0	0	0	1	1	0
0	0	1	1	0	0	0
1	0	1	0	1	0	0
0	1	0	1	1	0	0
1	1	0	0	1	0	0
0	1	1	0	0	0	1
1	1	1	0	0	0	1