

Electronique numérique

Automates (1)

Tous les exercices ne seront pas forcément résolus en TE.

Ce TD aborde plusieurs points relatifs aux *automates*, aussi appelés *machines d'états finis*¹ *finite state machines* (FSM).

- Diagrammes états transitions : machines de Mealy et de Moore.
- Causalité des machines d'états finis.
- Conception d'un automate matériel à partir d'un diagramme états-transitions.

1 Diagramme états-transitions. Moore et Mealy

On rappelle que le diagramme à bulle (ou “state transition diagram” ou diagramme “états-transitions”) est une représentation graphique pratique des automates. Elle consiste à décrire l'enchaînement des transitions d'un état à l'autre par une flèche, les états étant représentés par des cercles ou “bulles”.

On distingue deux grands types de FSM (voir figure 1)². Dans le cas des automates de Moore, les sorties ne dépendent que des états, alors que dans le cas des automates de Mealy, les sorties dépendent à la fois des états et des entrées. Sur les transitions, on ajoute les *conditions* booléennes $c_i(s)$ qui autorisent à passer d'un état à un autre, ainsi que d'éventuelles sorties (généralement après une ou deux barres verticales) dans le cas des automates de Mealy. Lorsqu'on a affaire à un automate de Moore, les sorties sont annotées en conséquence dans les bulles (ou juste à côté!).

1. Ce nom quelque peu étrange signifie que le *nombre* d'états est *fini*. Il existe également des automates dont le nombre d'états est infini...

2. Attention! Les deux automates représentés l'un à côté de l'autre ne sont pas équivalents en terme de fonctionnement. Il existe des algorithmes pour transformer Mealy en Moore (et réciproquement), de manière à obtenir des comportements identiques.... Voir par exemple https://fr.wikipedia.org/wiki/Machine_de_Mealy)



FIGURE 1 – FSM de Moore et Mealy

2 Causalité d'une FSM

Pour que qu'une FSM soit considérée comme *causale*, correcte ou *consistante*, elle doit vérifier les deux propriétés suivantes :

- **Réactivité** (Exhaustivité des conditions) : il doit exister une condition vraie sur au moins une des transitions sortant d'un état. .

$$\forall s \in S : \sum_i c_i(s) = 1$$

- **Déterminisme** (Exclusivité) : deux conditions sur les transitions sortant d'un état ne peuvent être vraies en même temps.

$$\forall s \in S, \forall (i, j) \text{ avec } i \neq j, c_i(s).c_j(s) = 0$$

Une autre manière de résumer ces deux propriété est qu'il existe – à tout instant discret – *un, et un seul, état suivant*.

Notons qu'un état peut avoir comme état suivant lui-même : dans ce cas, le système décrit ne change pas d'état.

Observez la figure 2.

- Vérifier la consistance de la machine d'état suivante.
- Est-ce un automate de Moore ou de Mealy ?

2.1 Equations logiques de l'automate

On cherche les équations logiques de la fonction "état suivant", ainsi que de la fonction de sortie de l'automate précédent. Ces deux ensembles nous permettrons de réaliser concrètement le circuit. On suggère ici de procéder en 2 étapes. La première consiste à conserver le nom *symbolique* des états tandis que la seconde fait apparaître l'*encodage* de ces états. Dans la FSM à réaliser, en choisissant un encodage *one-hot*, trois bits seraient nécessaires pour encoder les états. On va se limiter toutefois ici à un encodage *dense* où 2 bits seuls sont nécessaires. On s'appuie sur les tableaux génériques suivant 1 et 2, qui permettent de procéder de manière systématique.

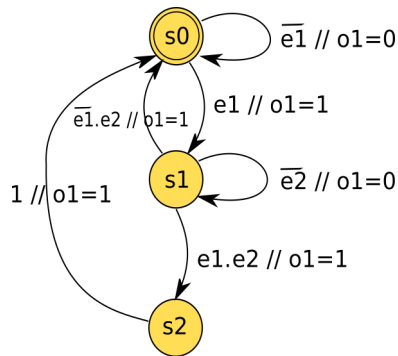


FIGURE 2 – FSM de l'exercice 1 et 2

TABLE 1 – Encodage symbolique (nom des états préservés)

état courant	entrée 1	...	entrée n	état suivant	sortie 1	...	sortie n

TABLE 2 – Encodage binaire (états encodés)

Q1	Q0	entrée 1	...	entrée n	D1	D0	sortie 1	...	sortie n

3 Application

Soit le diagramme états-transitions de la figure 3. On suppose que les entrées A et B proviennent de deux capteurs. Les sorties sont S1,S2 et S3.

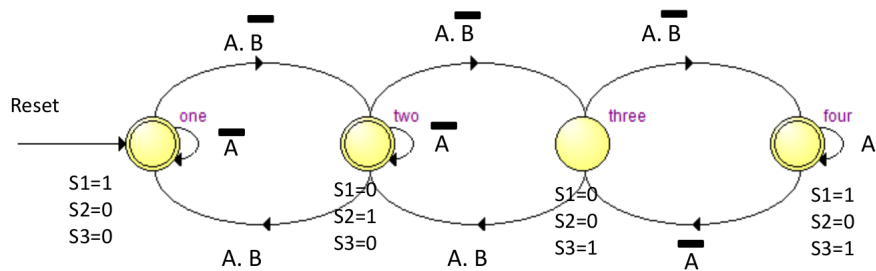
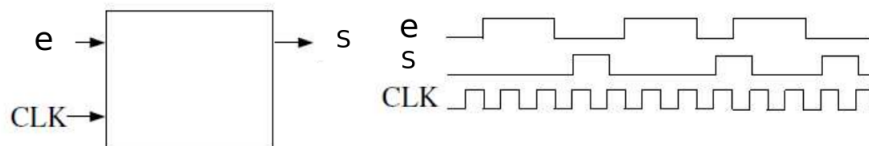


FIGURE 3 – Diagramme états-transitions (à modifier).

1. Cette machine est-elle causale? Si non, proposer une modification afin de la rendre causale.

- ## 4 Automate "détecteur de séquence"

- L'analyse de paquets réseaux, détection de paquets interdits (entrants ou sortants) dont on connaît une certaine signature numérique. C'est le travail de filtrage d'un "proxy", etc
- L'analyse de flux multimedia : par exemple "start codes" de séquences, dans un film (stocké en numérique, dans un format de compression).
- L'analyse génomique : il existe des accélérateurs FPGA dédiés à la reconnaissance de séquence du génôme, etc.
- etc.



On considère le détecteur de séquence de la figure 5. Son rôle est de détecter les séquences "0-1-1-0" sur son unique entrée x . Lorsque cette séquence est détectée, il émet un '1' sur sa sortie y . Un chronogramme est fourni à titre d'exemple. Notez que dans le flux "0-1-1-0-1-1-0", la séquence apparaît deux fois!

- 4

5 Automate "serrure numérique"

L'accès à un local est protégé par une serrure codée associée à un automate commandant la gâche électrique de la porte. La combinaison secrète est : A,D,C.

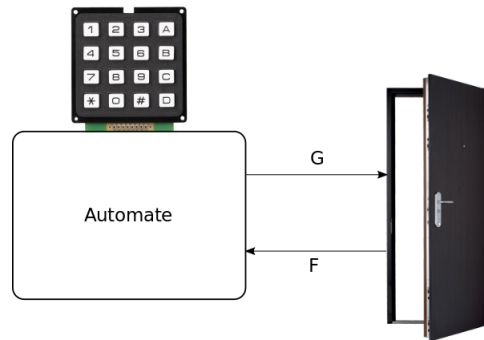


FIGURE 5 – Schéma de principe de la serrure numérique.

Un capteur permet de connaître l'état de la porte : F indique son état. F vaut à 1 si la porte est fermée à 0 si elle est ouverte. G commande l'ouverture de la porte : si $G=1$ la porte peut être ouverte. Lorsque la porte se referme, on a $G=0$. A l'initialisation la porte est fermée, le contact de porte $F=1$. On précise qu'on ne peut appuyer que sur un seul bouton à la fois.

1. Proposer un automate de contrôle du dispositif.
2. Discuter de la robustesse de la solution, et de son réalisme. Proposer des alternatives.
3. Optionel : construire le circuit complet.