

N7_SdN_1A

Architecture des ordinateurs

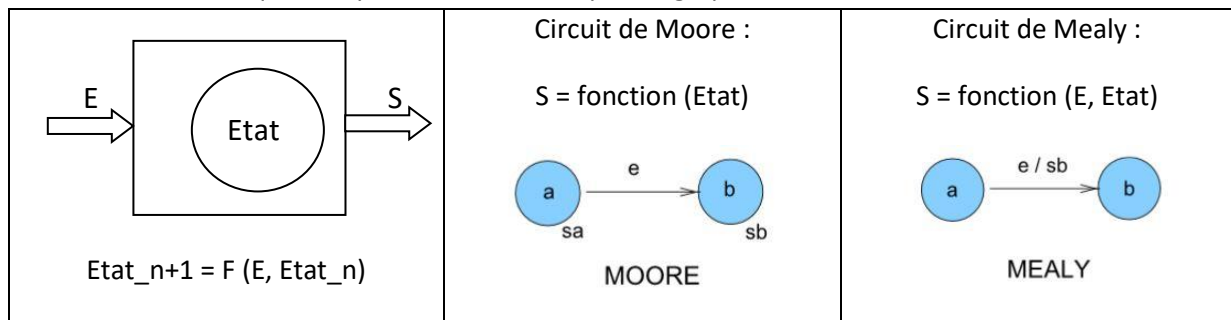
Semestre 2

TD2 : Circuits séquentiels

1- Rappel (lecture, questions, réponses et explications : 20 mn)

Circuit séquentiel = circuit qui conserve un état qui évolue en fonction des entrées et de lui-même.

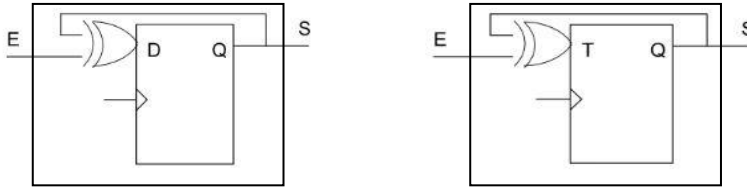
- On notera $Etat_n$ l'état à l'instant n (courant) et $Etat_n+1$ l'état à l'instant $n+1$ (suivant)
- Un circuit séquentiel peut être modélisé par un graphe d'états



- Lorsque les changements d'états sont cadencés par une horloge, le circuit est dit synchrone. On travaillera exclusivement avec des circuits synchrones.
- Pour coder les différents états d'un circuit, on utilisera des bascules, qui sont des mémoires 1 bit. Historiquement, il en existe 3 sortes, dont voici les équations caractéristiques :
 - Bascule D (Delay) : l'état X (=sortie) à l'instant $n+1$ et égal à l'entrée D . On notera :
 $X := D$ // bien noter l'affectation « $:=$ » qui indique une équation séquentielle
 - Bascule T (Trigger) : l'état X_{n+1} (à l'instant $n+1$) et égal à X_n si l'entrée $T=0$, et $= /X_n$ si $T=1$. Ce qui donne l'équation caractéristique : $X := /T * X + T * /X$
 - Une troisième bascule moins utilisée, nommée JK (2 entrées J et K) et dont l'équation est : $X := /K * X + J * /X$
- On utilisera principalement des bascules de type D, simples à décrire, et dont la syntaxe shdl est la suivante : $Q := D$ on clk reset when rst enabled when en
 - « D » pouvant être une expression logique, sous la forme de somme de min-termes
 - « clk » est le signal d'horloge qui fait passer la bascule de l'état n à l'état $n+1$
 - « rst » est le signal qui effectue une remise à zero asynchrone (forçage) de la bascule
 - « en » est le signal qui permet d'activer la bascule. L'expression « enabled when en » est facultative. Par défaut, « en » est à 1.
- Dans certaines situations, le fonctionnement du circuit ressemble à celui d'une bascule T (changements d'états bien identifiés), et il peut s'avérer intéressant de le réaliser en utilisant des bascules T :
 - $X := /T * X + T * /X$ on clk reset when rst ...
 - $T = \text{fonction} (E, X)$
- Pour un circuit séquentiel synchrone ayant N états, on utilisera M bascules, pour représenter ces N états, avec $2^M \geq N$. Dans certains cas, on peut être amené à utiliser une bascule par état pour faciliter la description du circuit.

Exemple : (5 + 5 mn)

Indiquer pour les deux circuits suivants le vecteur des entrées (entrées), le vecteur d'état (états), et le vecteur des sorties (sorties) ; et décrire ce qu'ils réalisent.



Donner leur description en langage shdl.

Un registre est un dispositif (ensemble de cases binaires : 8, 16, 32, ...) qui sert à mémoriser des informations utiles à l'exécution des instructions dans un processeur : opérandes, adresses, etc.

Pour implanter un registre, est-il plus efficace d'utiliser des bascules D ou des bascules T ?

Exercice 1 : (15 + 10 mn)

On souhaite réaliser le circuit qui permet de piloter un robot. Ce dernier peut prendre 4 états (A : arrêt, V1 : vitesse faible, V2 : vitesse moyenne, et V3 : vitesse haute, et est commandé via une seule commande M (Marche) :

- On ne peut transiter que vers un état voisin (A -> V1, V1 -> V2, V2 -> V3, V3 -> V2, V2 -> V1, V1 -> A)
- Lorsque M=0, le robot doit revenir à l'état Arrêt en respectant la règle dessus
- Lorsque M=1, le robot doit aller à l'état V3 en respectant la règle dessus

- A- Dessiner le graphe d'états
- B- Dresser la table de transitions
- C- Déterminer le nombre de bascules nécessaires, et effectuer l'assignation des états
- D- Réaliser le circuit en utilisant des bascules D
- E- Réaliser le circuit en réalisant des bascules T

Exercice 2 : (10 + 10 mn)

On souhaite réaliser un circuit qui compte de façon cyclique : 0, 1, 2, 3, ..., 15, 0, ... lorsque l'entrée en = 1, et n'évolue pas lorsque en = 0.

- A- Quel est le nombre d'états de ce circuit
- B- Quel est le nombre de bascules nécessaires ? on les nommera par la suite, C[0], C[1], ...
- C- Dans quels cas où C[0] change d'état (0 vers 1 et inversement), où S[1] change, etc.
- D- Donner la description de ce circuit en shdl sans passer par la table de transitions

Exercice3 : (10 + 5 mn)

Transformer une copie du circuit précédent en count_init4 (rst, clk, en, init, e[3..0] : s[3..0]) qui compte comme précédemment lorsque en=1 et sclr=0, et s'initialise avec l'entrée e[3..0] lorsque en=1 et sclr=1.

Exercice4 : (10 + 5)

En utilisant le module count_init4, réaliser le circuit count0a9 (rst, clk, en : s[3..0]) qui compte de 0,1, ..., 9, 0 ... lorsque en=1.