

INTRODUCTION AUX SYSTÈMES SÉQUENTIELS

Table des matières

1	Les machines à état	2
1.1	Quelques rappels	2
1.2	L'auto maintien	2
1.3	Télérupteur	2
2	Traduction d'une machine à état en LADDER	3
2.1	Les transitions	3
2.2	Activation des Etats	4
2.3	Associer les actions aux différents états	5
2.4	Résumé de la méthode	6
3	Machine à état avec des bascules RS	6
3.1	Représenter un état	6
3.2	Les conditions d'activation et de désactivation	6
3.3	Association des sorties	6
4	Application : Un tableau de classe	8

1 Les machines à état

1.1 Quelques rappels



Définition

Une machine d'état est une abstraction mathématique utilisée pour concevoir des algorithmes. Une machine d'état lit un ensemble d'entrées et passe à un état différent en fonction de ces entrées.

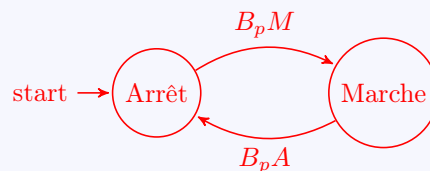
Nous nous intéresserons aux machines déterministes à états finis. Cela signifie que lorsque la machine se trouve dans un état, il est possible de savoir vers quel état la machine va évoluer lorsque l'on connaît les entrées.

1.2 L'auto maintien

Dans le cours précédent, nous avons implémenté un auto-maintien à l'aide de bascule RS puis sous la forme d'un réseau LADDER. Nous pouvons aussi décrire son comportement à l'aide d'une machine à état.

Activité 1

Question 1 Dessiner la machine à Etat d'un auto-maintien commandé par les boutons B_pM (Marche) et B_pA (Arrêt).



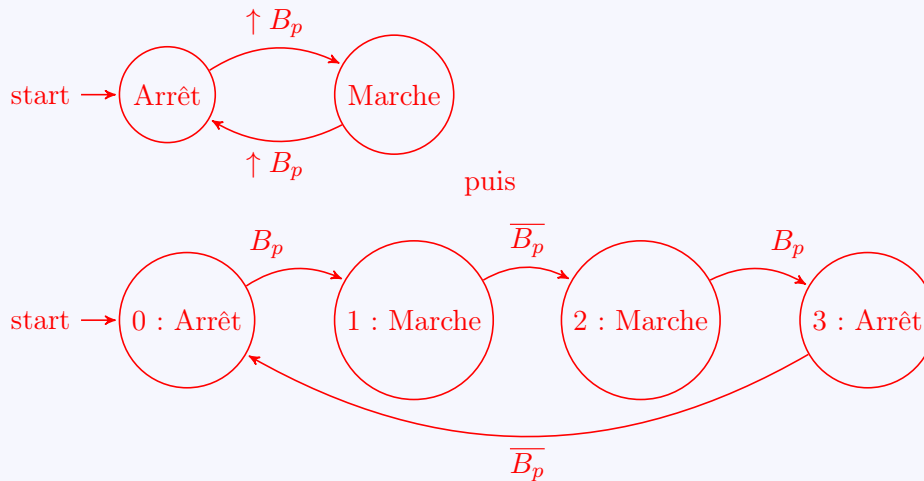
1.3 Télérupteur

Reprenons l'exemple du télérupteur. Un télérupteur a le comportement suivant :

- Au départ la lumière est éteinte
- Un appui sur le bouton BP allume la lumière
- Un nouvel appui sur le bouton BP éteint la lumière et on reprend à l'état initial

Activité 2

Question 2 Dessiner la machine à état d'un télérupteur avec puis sans utilisation de front montant



2 Traduction d'une machine à état en LADDER

Au premier semestre, nous avons vu comment implémenter une machine à état à l'aide du langage C (avec l'utilisation de la structure *switch ... case*). Ici, nous nous proposons d'implémenter une machine à état simple à l'aide du langage LADDER afin de la faire fonctionner sur un automate **LOGO**.



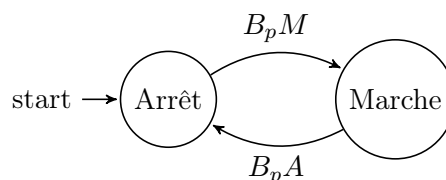
À retenir

Une démarche pour traduire une machine à état en LADDER est la suivante :

1. **Dessiner** la machine à état
2. Implémenter les transitions
3. Implémenter les changements d'état
4. Associer les actions aux différents états

Par convention, les noms des transitions commenceront par un *t* et les noms des variables par un *X*.

Prenons, encore une fois, l'exemple de l'auto-maintien sous forme de machine à état que nous désirons mettre sous la forme d'un LADDER.



2.1 Les transitions

Dans un premier temps, nous implémentons les transitions. Pour cela, voici les étapes à suivre :

- (a) Déclarer une variable associée à cette transition (exemple : $t_{0 \rightarrow 1}$ pour la transition de 0 vers 1)
- (b) Écrire l'équation régissant cette transition
- (c) Dessiner le réseau LADDER

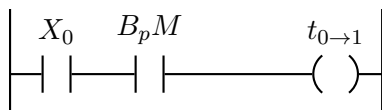
**À retenir**

Une transition n'est franchie que si l'état précédent est actif et que la réceptivité associée est vérifiée.

Exemple:

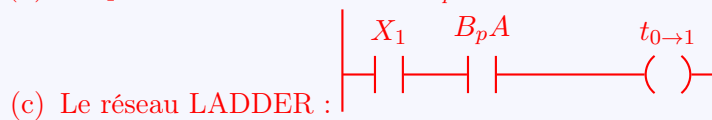
Pour l'auto-maintien, pour la transition depuis l'état 0 vers l'état 1 :

- (a) Utilisons la variable $t_{0 \rightarrow 1}$
- (b) L'équation est : $t_{0 \rightarrow 1} = X_0 \cdot B_p M$
- (c) Le réseau LADDER :

**Activité 3**

Question 3 Dessiner le schéma LADDER correspondant à la transition 1 vers 0

- (a) Utilisons la variable $t_{1 \rightarrow 0}$
- (b) L'équation est : $t_{1 \rightarrow 0} = X_1 \cdot B_p A$

**2.2 Activation des Etats**

Il faut à présent activer les état en à partir des transitions créées précédemment.

**À retenir**

L'écriture d'un état en LADDER se fait en deux étapes :

- Mise à 1 (SET) de l'état à l'aide d'un **OU** sur toutes les transitions **arrivant** sur celui-ci
- Mise à 0 (RESET) à l'aide d'un **OU** sur toutes les transitions **partant** de celui-ci

Ainsi les réseaux associés à l'état 0 de l'auto-maintien sont donnés sur la Figure 1

Le premier réseau active l'état 0 tandis que le deuxième le désactive.

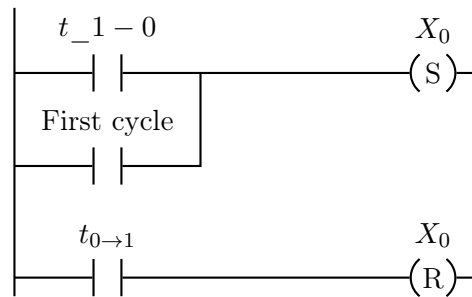


FIGURE 1 – Etat 0 de l'automatien

Activité 4

Question 4 Dessiner les réseaux LADDER implémentant le comportement de l'état 1

2.3 Associer les actions aux différents états

Il s'agit ici d'activer les actionneurs lorsque l'état associé est actif. Dans l'exemple de l'automatien, la sortie $Q1$ devra-t-elle être active lorsque l'état 1 est actif.

Activité 5

Question 5 Dessiner un réseau LADDER permettant l'activation de la sortie lorsque l'état 1 est actif.

2.4 Résumé de la méthode



À retenir *Méthode de traduction d'une machine à état en LADDER*

1. **Dessiner** la machine à état
2. Implémenter les transitions
 - (a) Déclarer une variable associée à cette transition (exemple : $t_{0 \rightarrow 1}$ pour la transition de 0 vers 1)
 - (b) Écrire l'équation régissant cette transition
 - (c) Dessiner le réseau LADDER
3. Implémenter les changements d'état
 - (a) Mise à 1 (SET) de l'état à l'aide d'un *OU* sur toutes les transitions **arrivant** sur celui-ci
 - (b) Mise à 0 (RESET) à l'aide d'un *OU* sur toutes les transitions **partant** de celui-ci
4. Associer les actions aux différents états

3 Machine à état avec des bascules RS

Comme pour le LADDER, il est possible d'implémenter une machine à état sous la forme d'un circuit logique. Pour cela, nous utiliserons des bascules RS afin de gérer les états.

La démarche est alors la suivante :

1. Dessiner la machine à état
2. Associer à chaque état une structure RS - Memento
3. Ecrire et implémenter les équation d'activation et de désactivation de ces états
4. Ajouter les actions associées aux états de la machine à états

3.1 Représenter un état



À retenir

Un état est représenté par une bascule RS en série avec un memento. A chaque instant, il ne peut y avoir qu'un seul état actif.

3.2 Les conditions d'activation et de désactivation

On passera d'un état au suivant si et seulement si la réceptivité de la transition est vérifiée et l'état précédent est actif. Cela signifie qu'une porte ET sera obligatoirement insérée après chaque état pour prendre en compte ces deux paramètres.

3.3 Association des sorties

On ajoute alors à la suite de chaque état les sorties qui sont activés lors de l'activation de chaque état.

Application guidée

En suivant la méthode, traduire la machine à état à l'aide de bascules RS

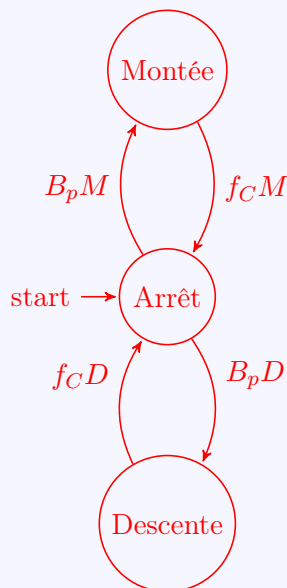
4 Application : Un tableau de classe

Soit un tableau de classe équipé de deux boutons poussoirs et qui suit le comportement suivant :

- Le volet est à l'arrêt
- Un appui sur B_pM met en route le moteur en sens montée jusqu'à ce qu'il arrive en haut (capteur $f_C H$ (fin de course Haut))
- Un appui sur B_pD met en route le moteur en sens de descente jusqu'à ce que le volet arrive au capteur $f_C B$ (Fin de course Bas)

Activité 6

Question 6 Dessiner la machine à état associée à ce comportement



Question 7 Implémenter cette machine à état en LADDER

Question 8 Implémenter cette machine à état à l'aide de bascules RS