

DIAGRAMME D'ETATS-TRANSITIONS / LANGAGE SFC



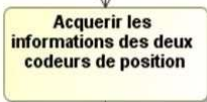

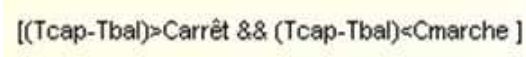
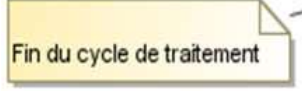

1. Diagramme d'états-transitions

Le diagramme d'états-transitions est un outil de modélisation qui permet de représenter graphiquement le fonctionnement d'un système séquentiel. La description du fonctionnement est représentée par un ensemble :

- *d'états*..... auxquels sont associées des *actions*.....,
- *de transitions*..... auxquelles sont associées des *conditions*.....

1.1. Éléments constitutifs

Sur un diagramme d'états-transitions, on trouve :

	<i>État initial</i>
	<i>État final</i>
	<i>État</i>
	<i>Transition</i>
	<i>Condition</i>
	<i>Commentaire</i>
	<i>Test</i>

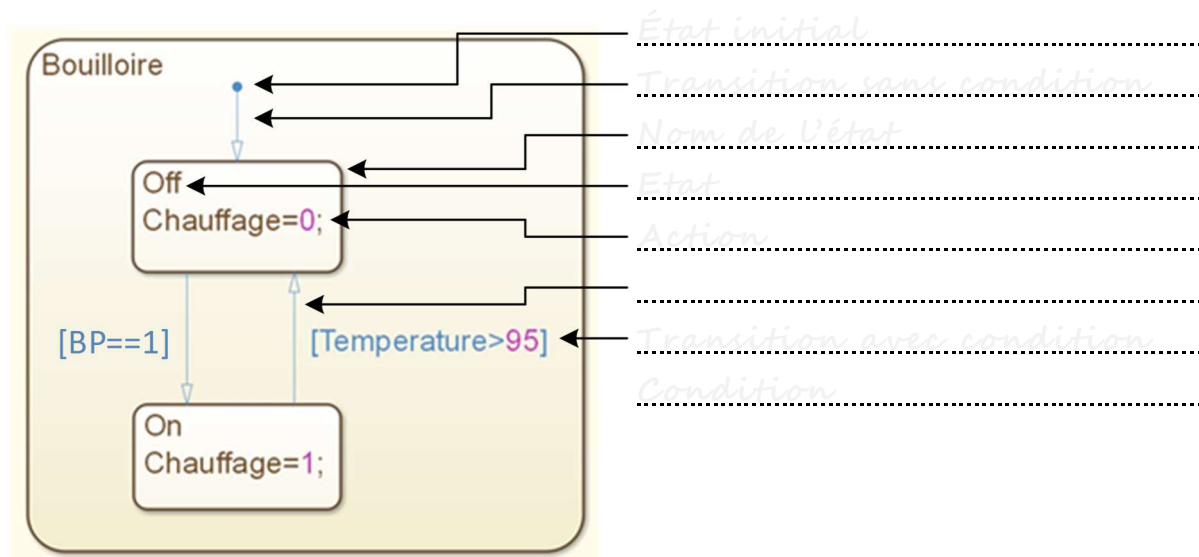
1.2. Tests de condition : les opérateurs logiques

Algorithme	Syntaxe
Vérifier que la variable a est égale à b	[a == b]
Vérifier que la variable a est différente de b	[a != b]
Vérifier que la variable a est supérieure ou égale à b	[a >= b]
Vérifier que la variable a est supérieure à b	[a > b]
Vérifier que la variable a est inférieure ou égale à b	[a <= b]
Vérifier que la variable a est inférieure à b	[a < b]
La transition est validée si la condition1 ET la condition2 sont vérifiées	[Condition1 && Condition2] Ex. : [(a == 1) && (b > 0)]
La transition est validée si la condition1 OU la condition2 sont vérifiées	[Condition1 Condition2]

b peut être un nombre ou une variable

1.3. Exemple de diagramme d'états-transitions

Lors d'un appui sur un bouton poussoir BP une bouilloire doit se mettre à chauffer l'eau jusqu'à ce que la température de l'eau atteigne 95°C :



1.4. Programmation d'un diagramme d'états-transitions

Différents logiciels nous permettent de représenter un diagramme d'état. Dans l'environnement **Matlab**, c'est **Stateflow** que nous utiliserons pour :

- modéliser les comportements séquentiels d'un système à l'aide de diagrammes d'état,
- simuler le fonctionnement du système,
- implanter les diagrammes d'état dans une carte microcontrôleur (Arduino, Controllino...).

1.4.1. États

Un état est représenté par un rectangle aux coins arrondis. Le nom de l'état ou son étiquette (« *state label* ») est écrit en haut à gauche du rectangle. Lorsqu'un état est actif, la ou les actions associées à cet état s'accomplissent. Les actions associées aux états peuvent avoir lieu :

- à l'activation de l'état, *entry: action*,
- à la désactivation de l'état *exit: action*,
- en continu tant que l'état est activé, *during: action*.

Les termes *entry*, *during*, *exit* sont appelés mots-clés (*keyword*). En absence de mot-clé, l'action est réalisée en entrant dans l'état. Ci-dessous, un exemple d'utilisation des mots-clés pour les actions avec la syntaxe *STATEFLOW* :

Exemple
 entry: N=N+1;
 during: moteur=cmd;
 exit: moteur=0;

Chaque action est conclue par un point-virgule.

1.4.2. Transitions

Le passage d'un état à l'autre se fait par une transition qui se matérialise par une liaison orientée entre un état source et un état pointé. La transition est caractérisée par une étiquette qui décrit les circonstances ou les conditions de passage d'un état à un autre.

L'étiquette de transition contient le plus souvent une condition (écrite entre crochets) ou un opérateur temporel. Vous avez vu les conditions précédemment voici les opérateurs temporels spécifiques à Stateflow.

Opération temporelle	Syntaxe STATEFLOW
La transition est franchie après 10 secondes	[after(10,sec)]
La transition est franchie si a vaut 1 avant 3 secondes	[a && before(3,sec)]

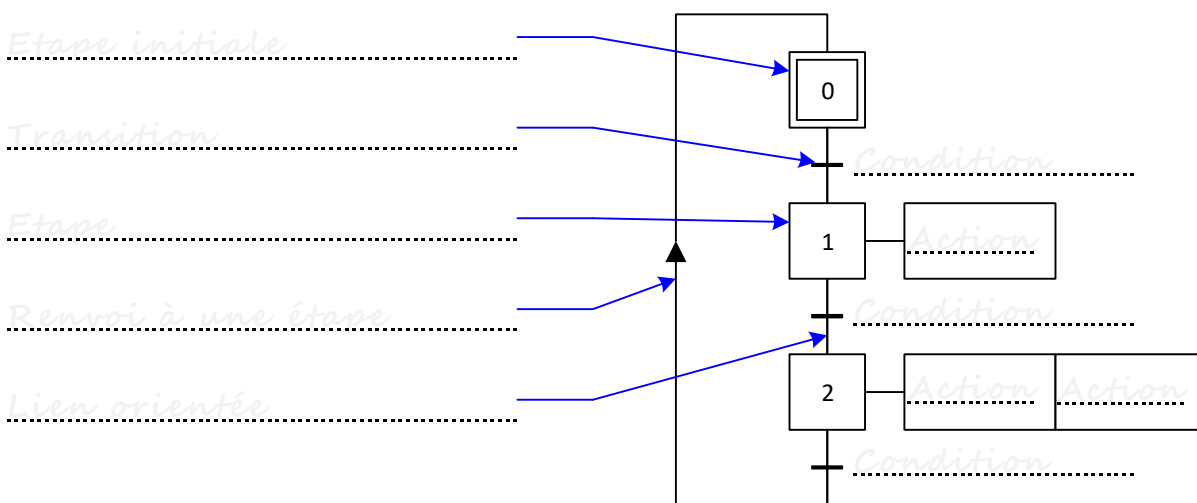
2. Langage SFC

Les logiciels de programmation de chez Schneider Electric (PLC Zelio), Crouzet (PLC EM4) et Siemens (PLC Logo) ne permettent pas la programmation d'un diagramme d'états-transitions. Il est donc nécessaire de transcrire les diagrammes d'états-transitions dans un langage reconnu par les PLC : le langage SFC.

Le langage SFC (Sequential Function Chart) est un langage graphique utilisé pour décrire des opérations séquentielles. Il décrit une suite connue d'étapes, reliées entre elles par des transitions. Une condition logique (booléenne) est attachée à chaque transition. Un ensemble d'actions est attaché à chaque étape.

2.1. Symboles graphiques

Les symboles graphiques du langage SFC sont :



Il est recommandé de numéroter les étapes d'un graphe SFC afin de les différencier.

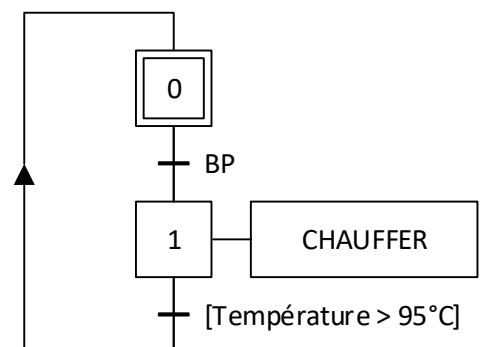
2.2. Conception d'un graphe SFC

Lors de la conception (rédaction) d'un graphe SFC, afin de préparer la programmation, il faut :

- Associer une *condition logique* aux *transitions* (à droite des transitions)
- Associer une ou plusieurs *actions* (ou aucune) aux *étapes* (dans un rectangle à droite)

2.3. Exemple de graphe SFC

Lors d'un appui sur un bouton poussoir BP une bouilloire doit se mettre à chauffer l'eau jusqu'à ce que la température de l'eau atteigne 95°C :



2.4. Programmation en langage SFC

Seule la structure du graphe est saisie en langage SFC. Les conditions et les actions sont décrites avec d'autres langages (LD, FBD...).

Le graphe SFC de la bouilloire donne le programme suivant avec le logiciel ZelioSoft2 :

