

# Cours Millenium 3

## Version AC9

### Leçon 9

#### Les FB SFC

#### Les FB SFC (Grafcet)

## SOMMAIRE

<b>1</b>	<b>INTRODUCTION</b>	<b>3</b>
<b>2</b>	<b>ONGLET SFC :</b>	<b>4</b>
<b>3</b>	<b>LES ÉLEMENTS GENERAUX DU LANGAGE</b>	<b>4</b>
3.1	L'étape	8
3.2	L'action	9
3.3	Les liaisons et les transitions	10
3.4	Les réceptivités	10
3.5	Les 5 règles d'évolution	13
3.6	Les branchements	17
3.7	Conversion du GRAFCET au LADDER	20
3.8	Conversion du GRAFCET au SFC & FBD	22
3.9	Mise en équation	23
<b>4</b>	<b>LES FB</b>	<b>25</b>
4.1	FB Étape initiale (INIT STEP)	26
4.2	FB Étape initiale réinitialisable (RESET INIT)	26
4.3	FB Étape (STEP)	26
4.4	FB Divergence OU et convergence OU	27
4.5	FB Divergence ET et convergence ET	29
4.6	FB SFC Attente	30
4.7	FB SFC MOUVEMENT et Multiplexeur moteur	31

Figure 1 : L'étape .....	8
Figure 2 : L'action .....	9
Figure 3 : La transition .....	10
Figure 4 : La réceptivité .....	10
Figure 5 : Résumé .....	12
Figure 6 : Règle 1 .....	13
Figure 7 : Règle 2 .....	14
Figure 8 : Règle 3 .....	15
Figure 9 : Règle 4 .....	16
Figure 10 : Règle 5 .....	16
Figure 11 : Les branchements.....	18
Figure 12 : Conversion Grafcet en SFC .....	22

# 1 Introduction

Extrait de la norme **CEI 60848**

## **Annexe C** (Informative)

### **Les relations entre le GRAFCET selon la CEI 60848 et le SFC selon la CEI 61131-3** **C.1 Introduction**

Les deux normes CEI 60848 et CEI 61131-3 ont chacune un domaine spécifique d'application :

- un langage de spécification de comportement (GRAFCET – GRAPhe Fonctionnel de Commande Etape transition) indépendant de toutes technologies de réalisation pour la CEI 60848, et
- un langage de programmation spécifique (SFC – Sequential Function Chart) pour la CEI 61131-3.

Le GRAFCET de la CEI 60848 est utilisé pour décrire ou spécifier le comportement du système, du point de vue «externe», alors que le langage de SFC de la CEI 61131-3 est employé pour décrire (une partie de) la structure «interne» du logiciel implémenté dans le système.

Si les deux langages étaient employés pour décrire un système de commande, les deux descriptions (deux sortes différentes de document) paraîtraient graphiquement semblables. Cependant, elles n'auraient pas la même signification, pas même si elles étaient graphiquement identiques. Ceci indiquerait juste que la structure du logiciel, décrite en SFC, se comporte d'une telle manière qu'elle peut être décrite graphiquement de façon semblable par un grafcet. Les propriétés des éléments fondamentaux associés aux représentations graphiques d'élément sont néanmoins différentes dans les deux cas.

Bon c'est un extrait, il y a une centaine de pages, ce n'est pas simple à comprendre, aussi je vais essayer de présenter le GRAFCET.

Pourquoi le Grafcet ?

C'est bien pratique quand on veut faire du séquentiel.

Lorsque certaines spécifications sont exprimées en langage courant, il y a un risque permanent d'incompréhension.

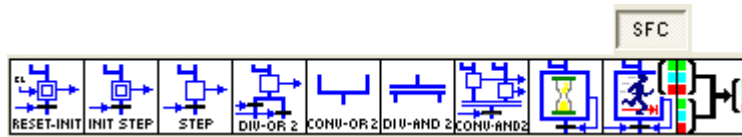
- Certains mots sont peu précis, mal définis ou possèdent plusieurs sens.
- Le langage courant est mal adapté pour décrire précisément les systèmes séquentiels.



Il faut que le vérin remonte avant la fin de la rotation de la perceuse mais seulement si celle-ci est en grande vitesse et que la pièce soit de type A ou C mais avec l'option 4.

## 2 Onglet SFC :

Cet onglet contient les icônes SFC « Sequential Flow Chart », c'est la mise en application du langage « GRAFCET »



## 3 Les Éléments généraux du langage

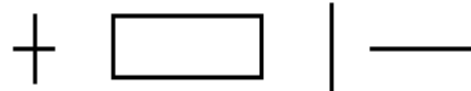
Le Grafcet est un moyen graphique de représenter les phases successives de fonctionnement d'un équipement de contrôle ou d'un automate.

Ce graphe, orienté verticalement, se lit de haut en bas. Il utilise les blocs de base suivant :

Étapes (étape simple, étape initiale)



Transition, action, liaisons (Verticale & Horizontale)



Divergences (en OU et en ET)



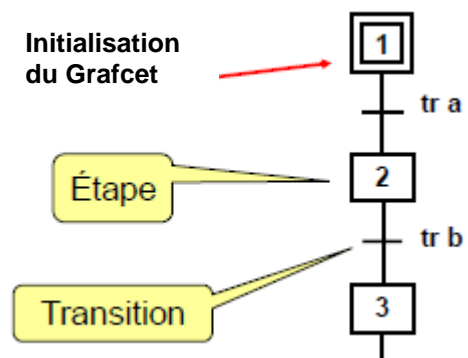
Convergences (en OU et en ET)



Ces blocs de base permettent de :

- Initialiser un enchaînement de phases de fonctionnement au lancement du programme :  
**Étape initiale**

- Représenter et commander des phases consécutives de fonctionnement :  
**Étapes et transitions**

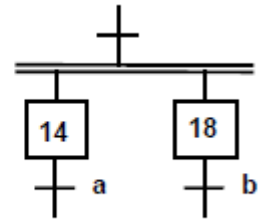


- Représenter et commander des phases simultanées de fonctionnement :

### **Divergence en ET**

SYNCHRONISATION aval

Divergence  
ET

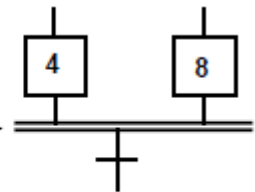


- Enchaîner une phase unique de fonctionnement après des phases simultanées de fonctionnement :

### **Convergence en ET**

SYNCHRONISATION amont

Convergence  
ET

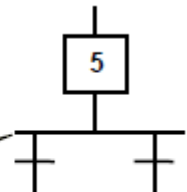


- Enchaîner après une phase de fonctionnement, une phase de fonctionnement parmi un choix de plusieurs phases possibles :

### **Divergence en OU**

SELECTION de séquences

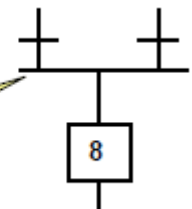
Divergence  
OU



- Enchaîner une même phase de fonctionnement, après l'une ou l'autre phase de fonctionnement :

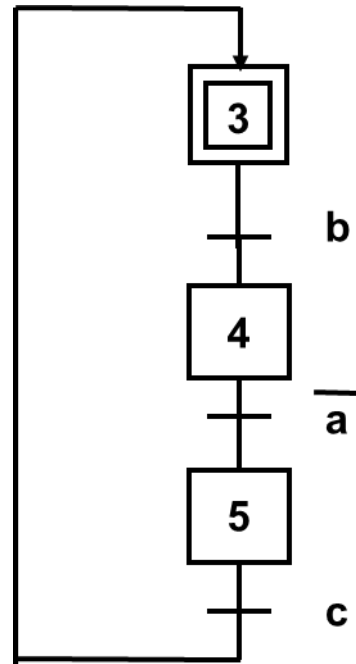
### **Convergence en OU**

Convergence  
OU



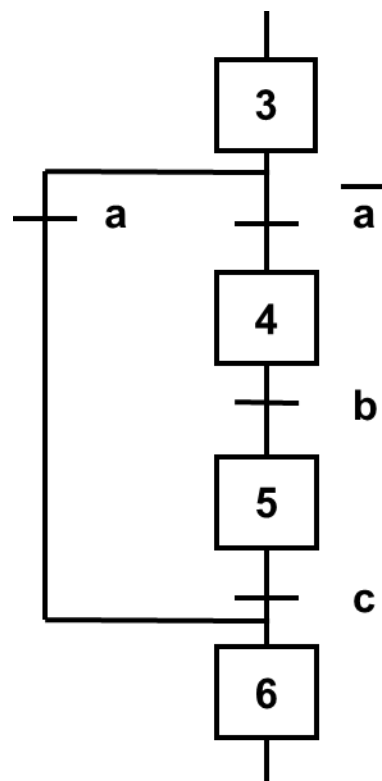
- Construire un saut d'étape saut avec séquence unique

Cycle d'une seule séquence



- Construire un saut d'étape

Saut d'étape  
(Toujours du haut vers le bas)



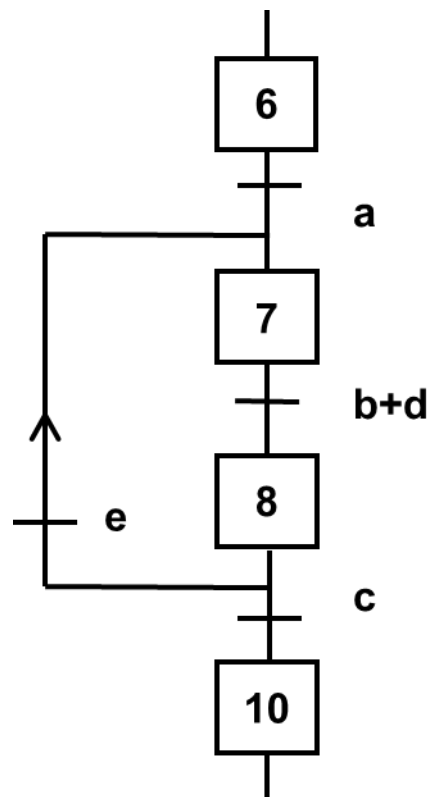
- Construire une suite de phases de fonctionnement sans fin

### Reprise de séquence (boucle)

Du bas vers le haut, mettre une flèche



Permet de recommencer la même séquence jusqu'à ce qu'une condition fixée soit obtenue



### Les éléments de base

Pour comprendre la syntaxe du GRAFCET, il faut connaître les éléments suivants :

Étapes

Transitions

Réceptivités

Actions

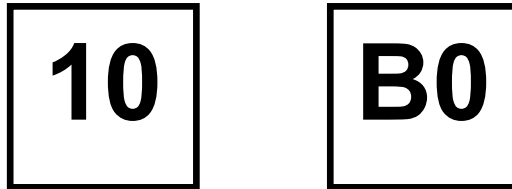
Liaisons

### 3.1 L'étape

Définition : à un instant donné **une étape est soit active, soit inactive**. L'ensemble des étapes actives définit la situation du système à l'instant considéré.

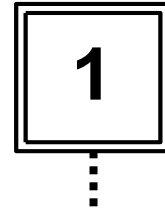
Représentée généralement par un carré, identifiée par un repère alphanumérique.

L'étape



Variable d'étape : l'état d'une étape peut être représenté par les valeurs logiques d'une variable.  
Exemple :  $X_{10} = 1$     $X_{B0} = 0$

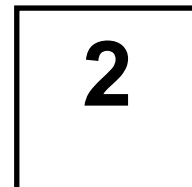
L'étape initiale est représentée par un carré double.



Elle doit être activée lors de la mise sous tension (cas général).

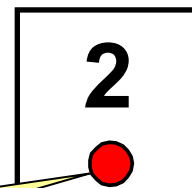
L'état actif ou inactif d'une étape peut être représenté respectivement par les valeurs logiques « 1 » ou « 0 » d'une variable booléenne  $X_i$  ( $i$  = numéro de l'étape)

Si  $X_i = 0$ , étape inactive      ici  $X_2 = 0$



Si  $X_i = 1$ , étape active      ici  $X_2 = 1$

Une étape est donc soit active ou inactive.



L'étape active est symbolisée  
par un point

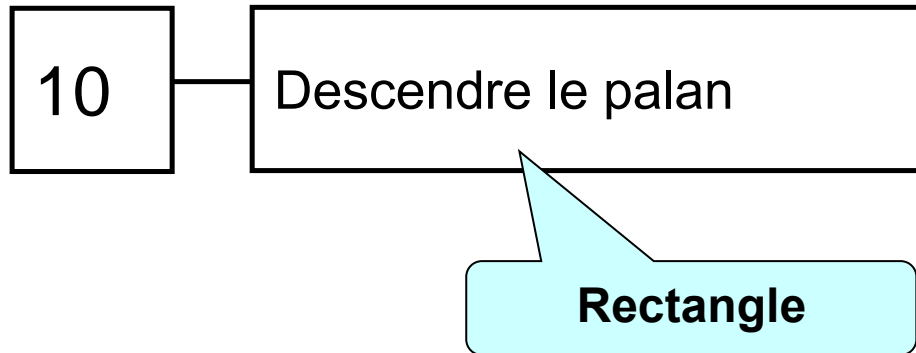
Figure 1 : L'étape



### 3.2 L'action

Libellé d'assignation d'une sortie :

Description des tâches à effectuer lorsqu'une étape est active.



Exemples de représentation :

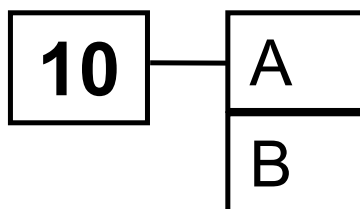
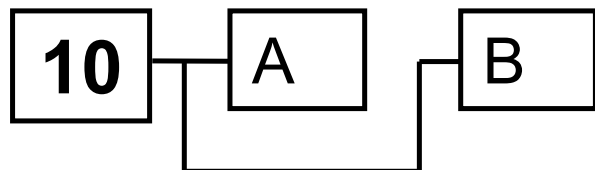
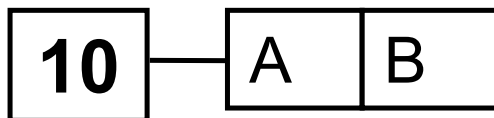


Figure 2 : L'action

### 3.3 Les liaisons et les transitions

#### Les transitions :

Ce sont des barrières entre les étapes qui peuvent être franchies selon certaines conditions.

Liaison orientée de haut en bas, sinon, mettre une flèche...

Relient les étapes entre-elles.

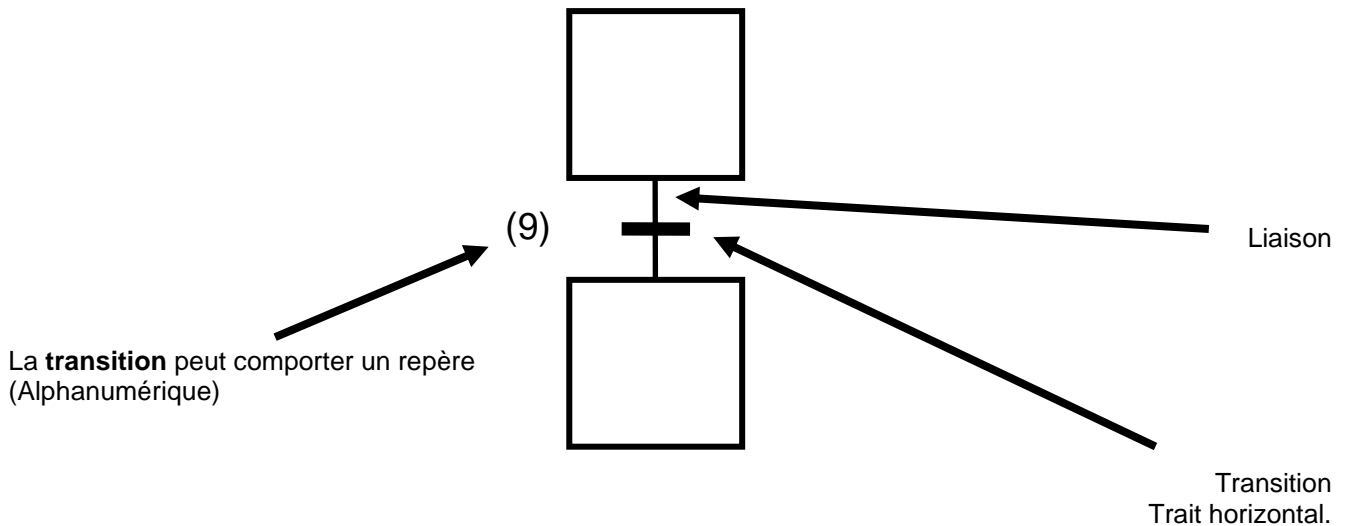


Figure 3 : La transition

### 3.4 Les réceptivités

Ce sont les conditions qui doivent être remplies pour franchir la transition.

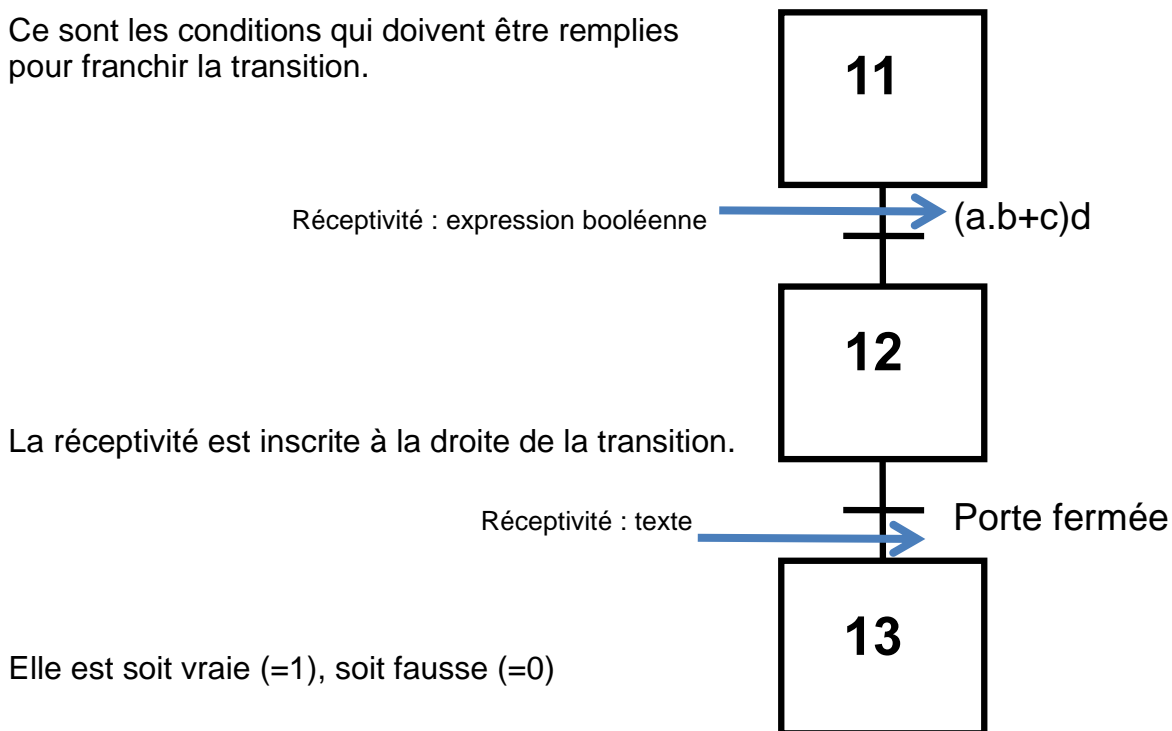
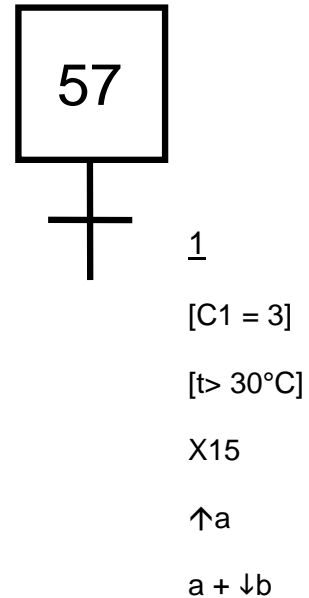


Figure 4 : La réceptivité

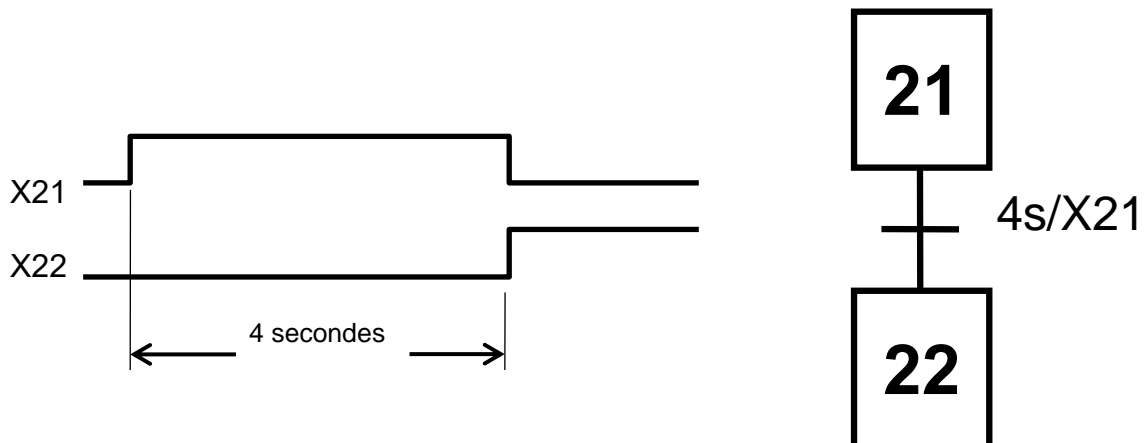
**Remarques :**

Une réceptivité est une proposition logique qui peut renfermer diverses variables booléennes qui peuvent être :

- réceptivité toujours vraie
- des variables auxiliaires (compteurs, temporisations, ...)
- des informations extérieures (capteurs, directives)
- l'état d'autres étapes (attentes, interdictions)
- changement d'état d'autres variables : front montant, front descendant



La notation [xxxx] signifie que la valeur booléenne du prédicat constitue la variable de réceptivité. Ainsi lorsque l'assertion xxxx est vérifiée, le prédicat vaut 1

**La réceptivité dépendante du temps**

La réceptivité associée à la transition sera vraie 4 s après l'activation de l'étape 21, et sera fausse du fait du franchissement de la transition qui désactive l'étape amont.

# Résumé

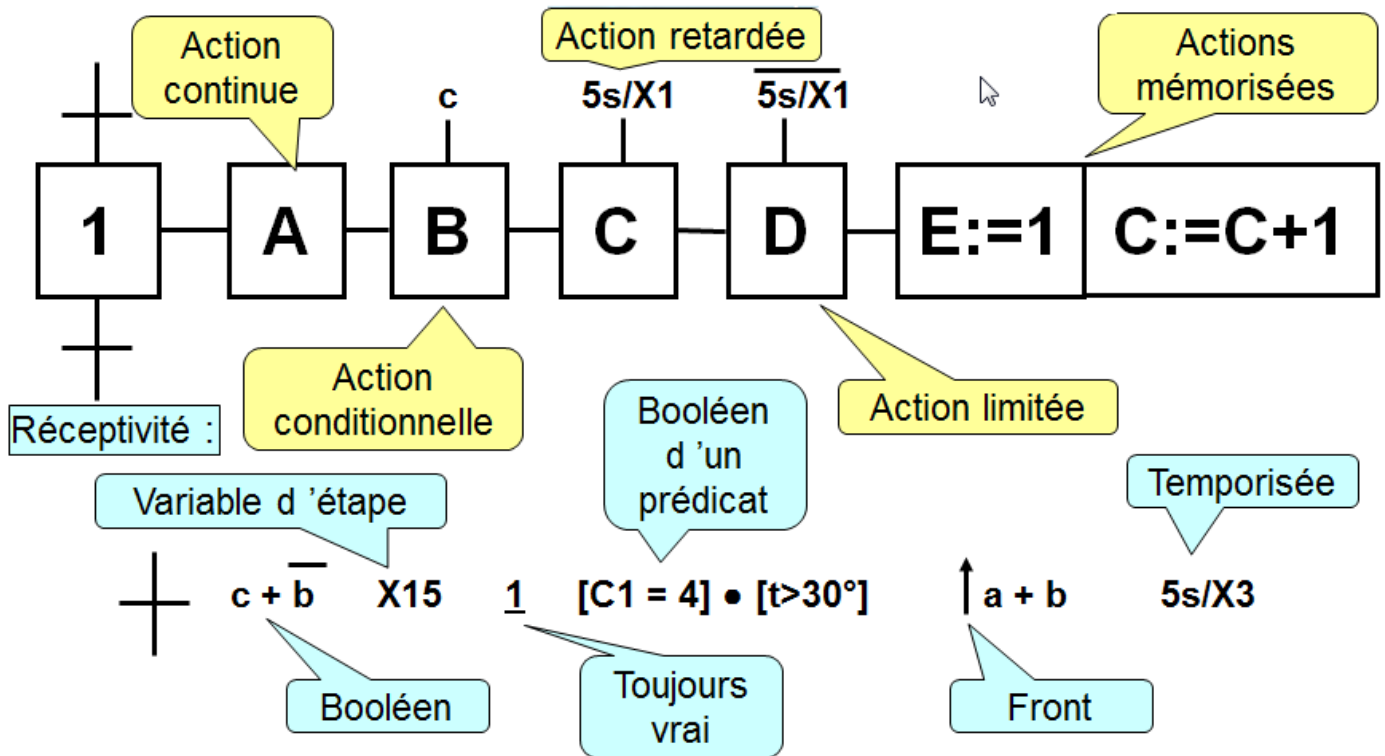


Figure 5 : Résumé

### 3.5 Les 5 règles d'évolution

Pour comprendre comment un GRAFCET fonctionne, il faut connaître les règles suivantes : (extrait)

**Règle 1 : Situation initiale**

L'initialisation précise les étapes activées au début de fonctionnement.

**Règle 2 : Franchissement d'une transition**

Une transition est soit validée soit non validée. Elle est validée lorsque toutes les étapes immédiatement précédentes sont actives.

**Règle 3 : Évolution des étapes actives**

Le franchissement d'une transition entraîne l'activation de toutes les étapes immédiatement suivantes et la désactivation de toutes les étapes immédiatement précédentes.

**Règle 4 : Évolutions simultanées**

Plusieurs transitions simultanément franchissables sont simultanément franchies

**Règle 5 : Activation et désactivation simultanées d'une étape**

Si au cours du fonctionnement, une même étape doit être désactivée et activée simultanément elle reste active

#### 3.5.1 Règle #1 - Situation initiale

Règle 1 : La situation initiale, choisie par le concepteur, est la situation à l'instant initial.

La situation initiale est la situation à l'instant initial, elle est donc décrite par l'ensemble des étapes actives à cet instant. Le choix de la situation à l'instant initial repose sur des considérations méthodologiques et relatives à la nature de la partie séquentielle du système visé.

L'initialisation précise les étapes activées au début de fonctionnement. On repère les étapes initiales du Grafcet en doublant les contours d'une étape

Il existe toujours au moins une étape active lors du lancement de l'automatisme.

Ces étapes activées lors du lancement sont nommées "ÉTAPES INITIALES"

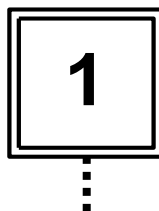


Figure 6 : Règle 1

### 3.5.2 Règle #2 - Franchissement d'une transition

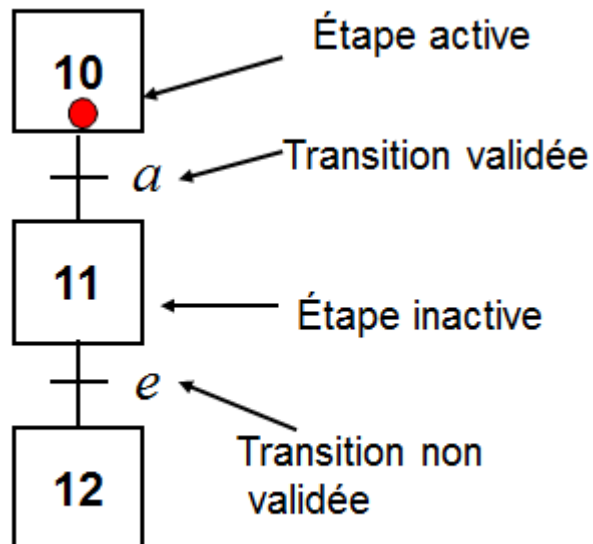
Règle 2 : Une transition est dite validée lorsque toutes les étapes immédiatement précédentes reliées à cette transition sont actives.

Le franchissement d'une transition se produit :

Lorsque la transition est validée

Et que la réceptivité associée à cette transition est VRAIE

#### Exemple 1



#### Exemple 2

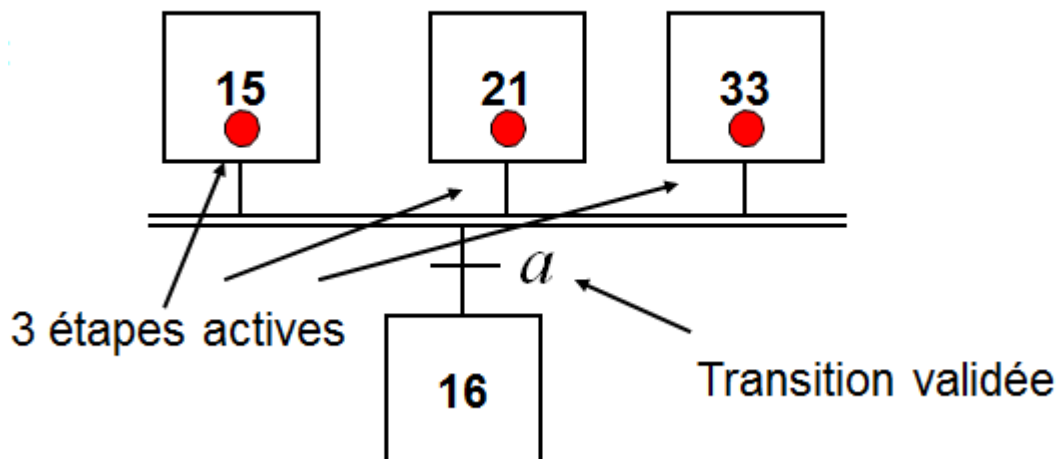


Figure 7 : Règle 2

### 3.5.3 Règle #3 -Évolution des étapes actives

Le franchissement

Règle 3 : Le franchissement d'une transition entraîne simultanément l'activation de TOUTES les étapes immédiatement suivantes et la désactivation de TOUTES les étapes immédiatement précédentes.

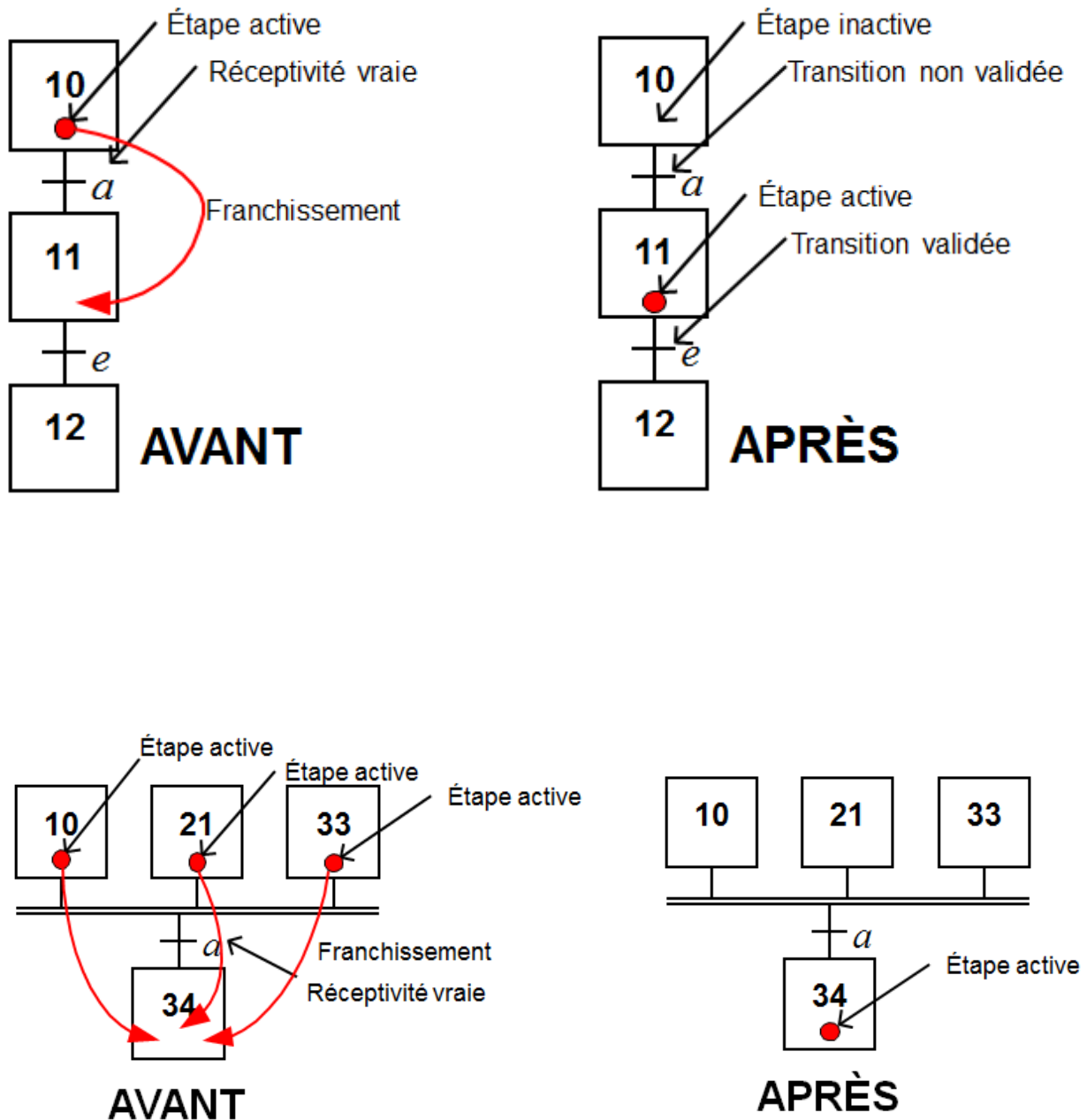


Figure 8 : Règle 3

### 3.5.4 Règle #4 - Évolutions simultanées

Règle 4 :

Plusieurs transitions simultanément franchissables sont simultanément franchies

L'évolution entre deux situations actives implique qu'aucune situation intermédiaire ne soit possible, on passe donc instantanément d'une représentation de la situation par un ensemble d'étapes à une autre représentation.

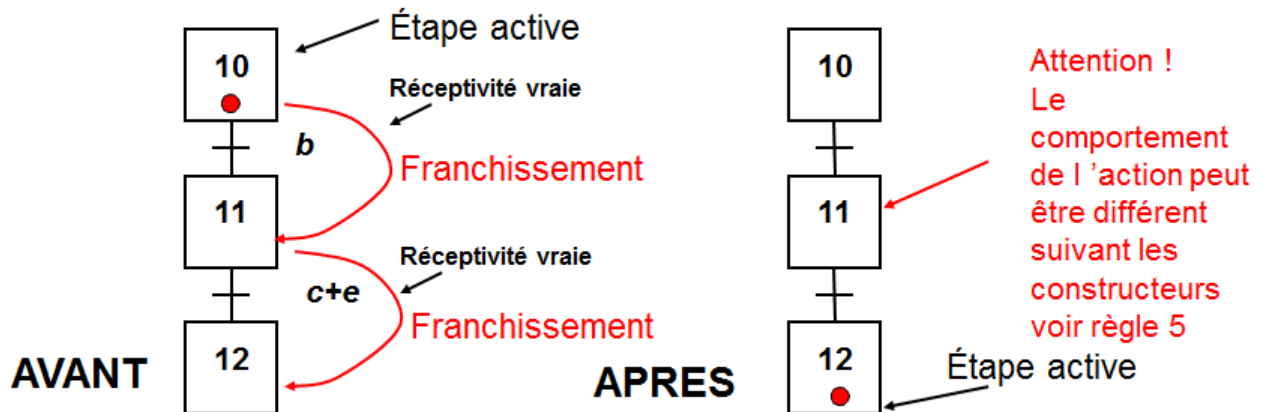


Figure 9 : Règle 4

### 3.5.5 Règle #5- Activation et désactivation simultanées d'une étape

La cohérence

Règle 5 : Si, au cours du fonctionnement, une étape active est simultanément activée et désactivée, alors elle reste active.

Si une même étape participe à la description de la situation précédente et à celle de la situation suivante, elle ne peut, en conséquence, que rester active.

La durée du franchissement d'une transition ne peut être jamais rigoureusement nulle, il en est de même pour la désactivation d'une étape. En outre la règle 5 se rencontre rarement dans la pratique.

Ces règles ont été formulées pour des raisons de cohérence théorique interne au GRAFCET.

Figure 10 : Règle 5



### 3.6 Les branchements

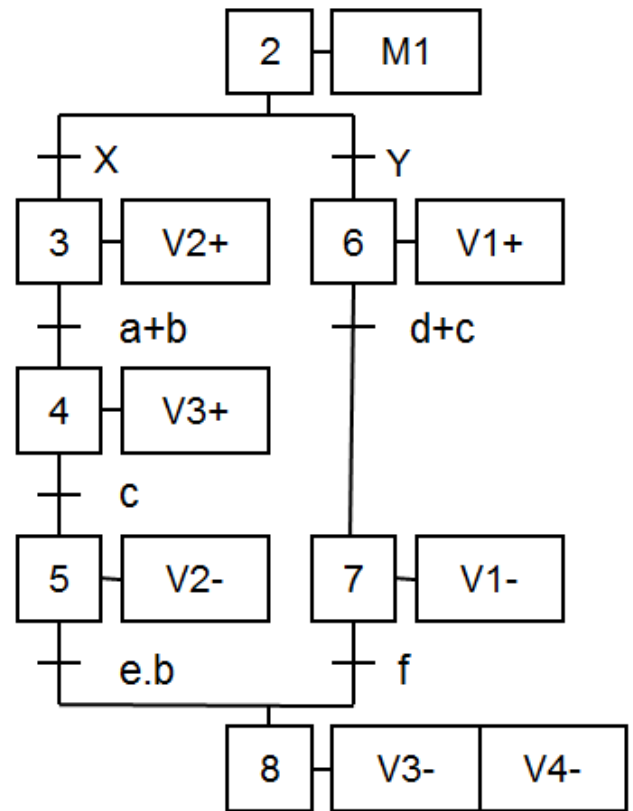
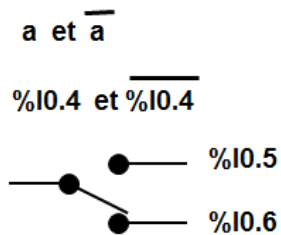
#### 3.6.1 Sélection de séquences

Les séquences exclusives

Appelés aussi « aiguillages »

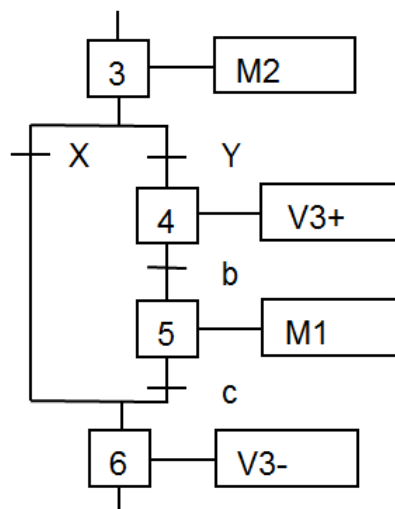
X et Y sont mutuellement exclusifs.

Exemples :



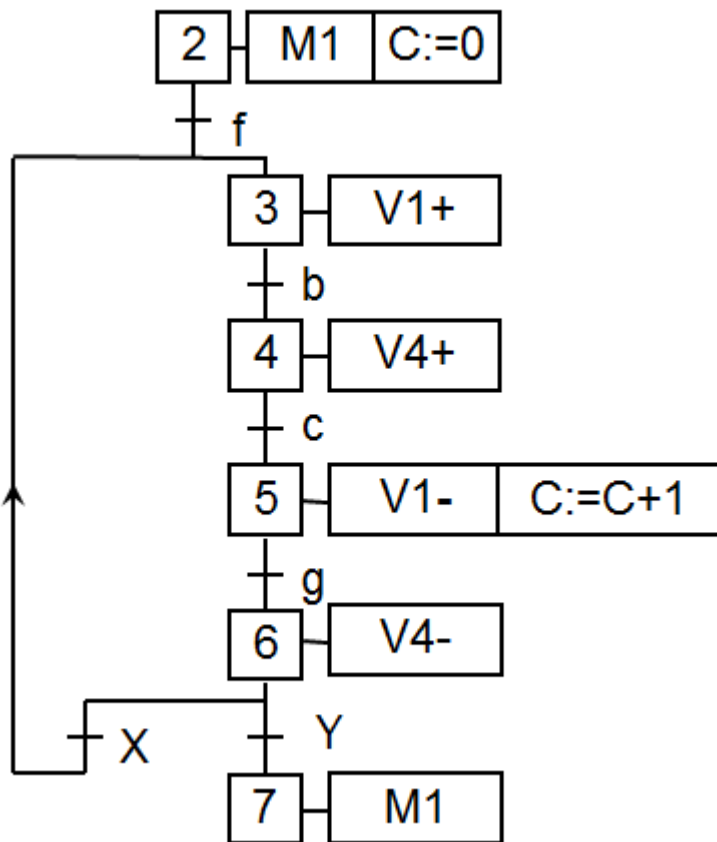
**Variante d'un « aiguillage »**

Le saut d'étapes



### Variante d'un « aiguillage »

La reprise de séquence



Exemples pour X et Y

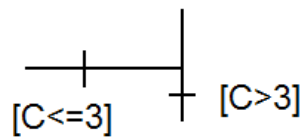
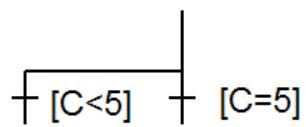
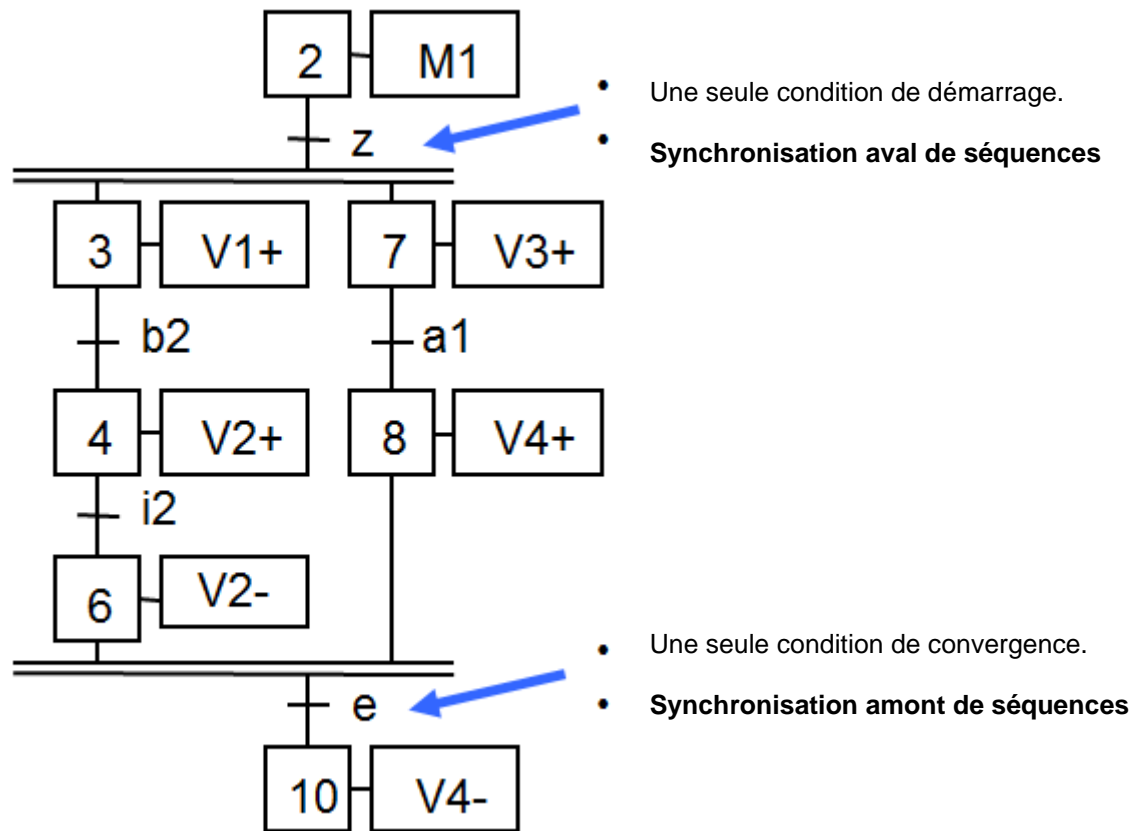


Figure 11 : Les branchements

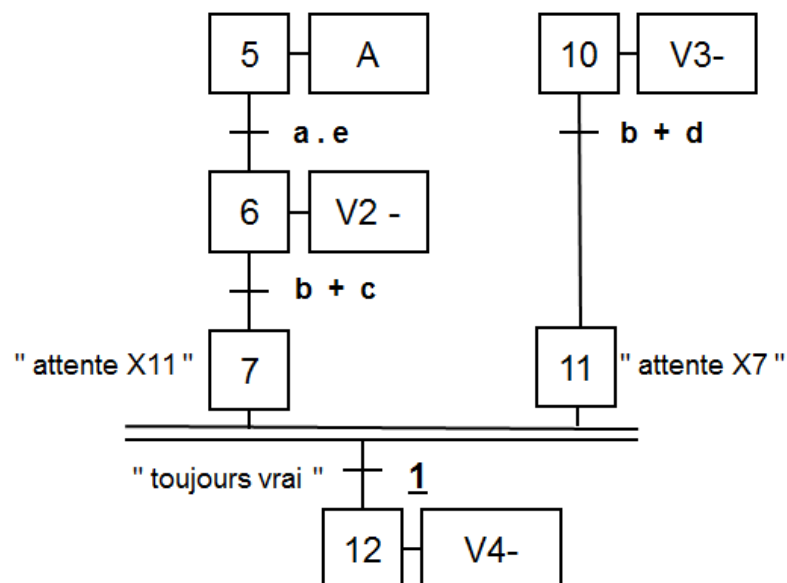
### 3.6.2 Les séquences simultanées



Attention :



Dans certaines applications, il est très fortement recommandé de terminer chaque séquence avec une étape d'attente. En effet on ne saura pas toujours quelle branche se terminera en premier.

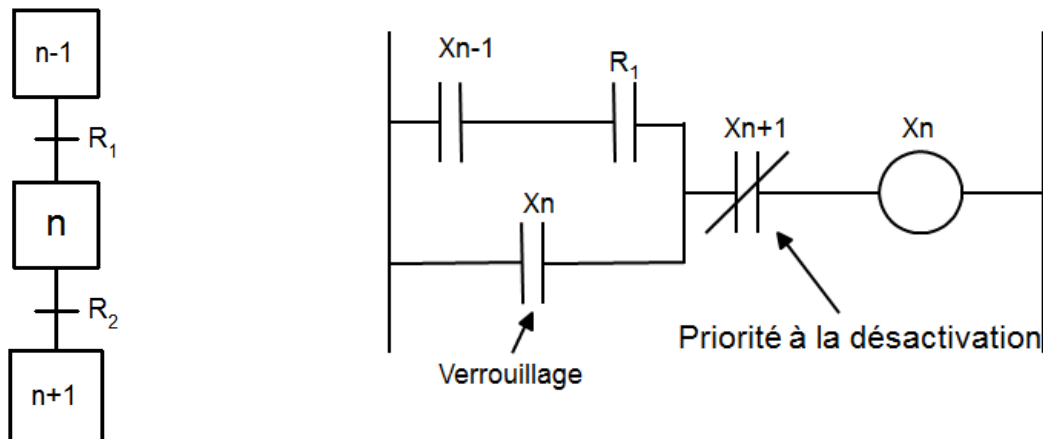


### 3.7 Conversion du GRAFCET au LADDER

#### 3.7.1 Ladder avec priorité à la désactivation

Chaque étape du GRAFCET peut être représenté par l'équation suivante :

$$X_n = (X_{n-1} \cdot R_1 + X_n) \cdot \overline{X_{n+1}}$$



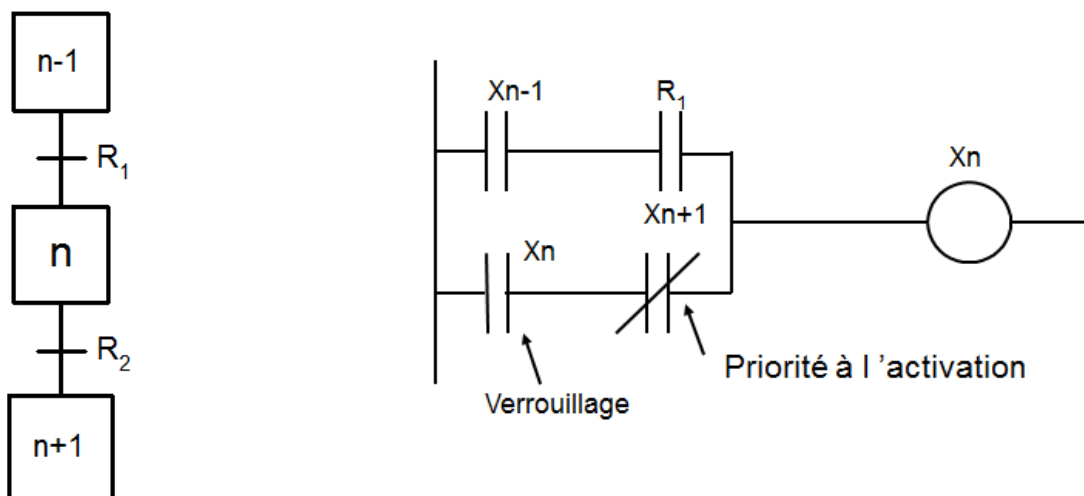
**Attention :** la règle 5 n'est pas respectée

« Si, au cours du fonctionnement, une étape active est simultanément activée et désactivée, alors elle reste active. »

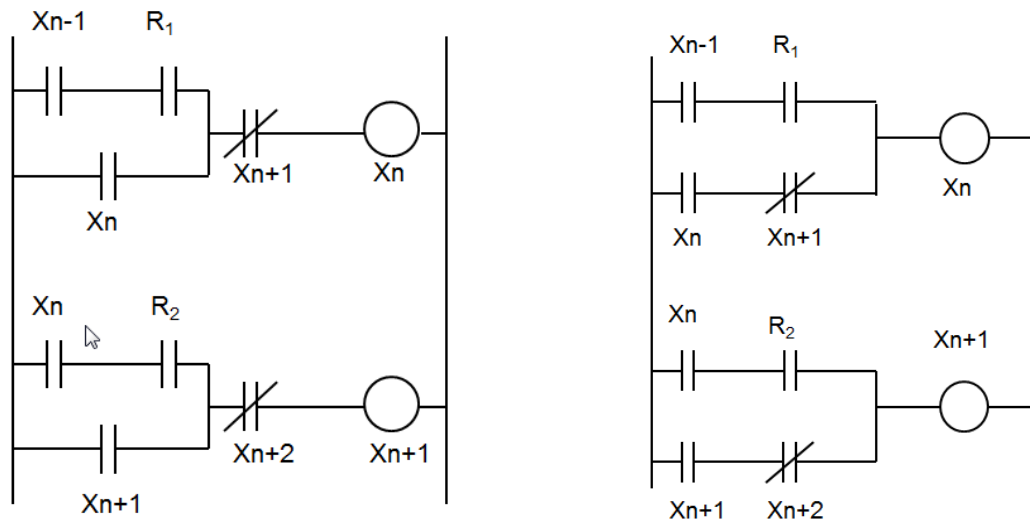
#### 3.7.2 Ladder avec priorité à l'activation

Chaque étape du GRAFCET peut être représenté par l'équation suivante :

$$X_n = X_{n-1} \cdot R_1 + X_n \cdot \overline{X_{n+1}}$$



### 3.7.3 Bug majeur de ces 2 approches



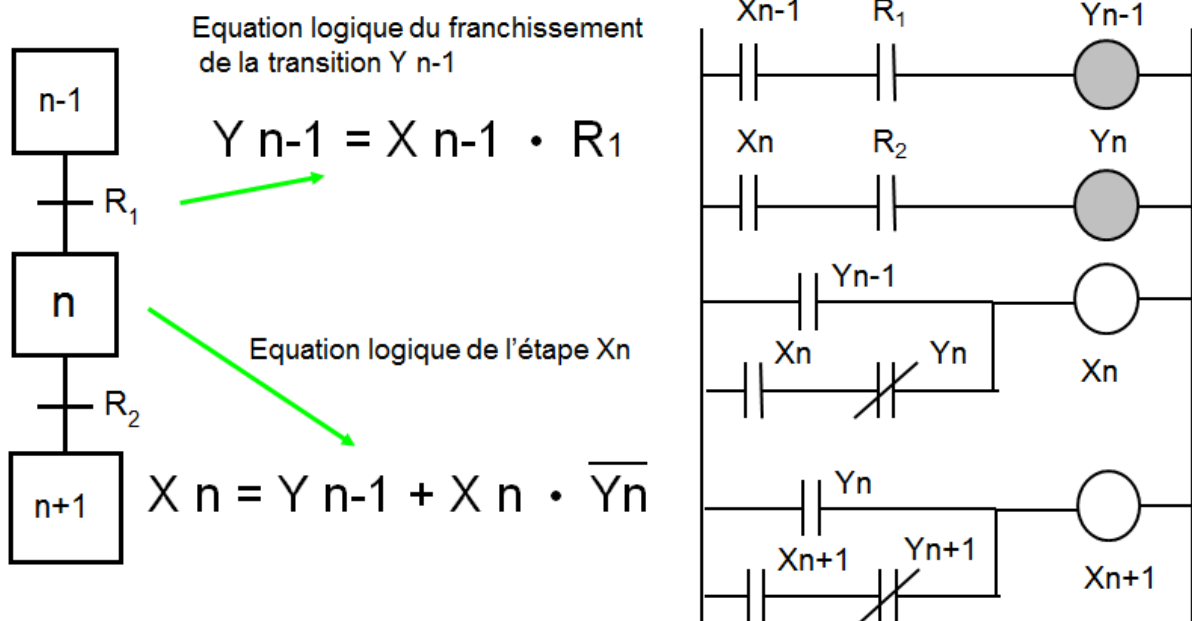
DEUX ÉTAPES SUCCESSIVES À 1 EN MÊME TEMPS !!!

- 1)  $X_n$  est à 1
- 2)  $R_2$  passe à 1
- 3) Pendant un cycle automate  $X_n$  et  $X_{n+1}$  sont à 1 en même temps

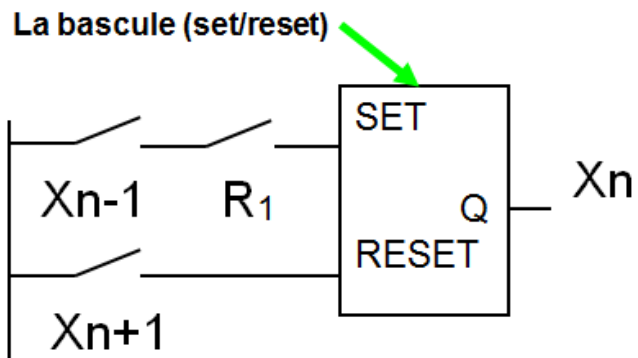
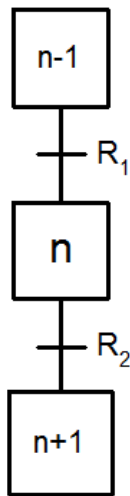
#### Solution

Programmation des transitions séparément

Prendre de préférence cette solution, car elle élimine les problèmes précédents.



### 3.7.4 Utilisation de la bascule (set/reset)



Si « SET » = 1,  $Q = 1$

Si « RESET » = 1,  $Q = 0$

Si les deux = 1 : 2 cas :

Priorité à l'activation ->  $Q = 1$

Priorité à la désactivation ->  $Q = 0$

### 3.8 Conversion du GRAFCET au SFC & FBD

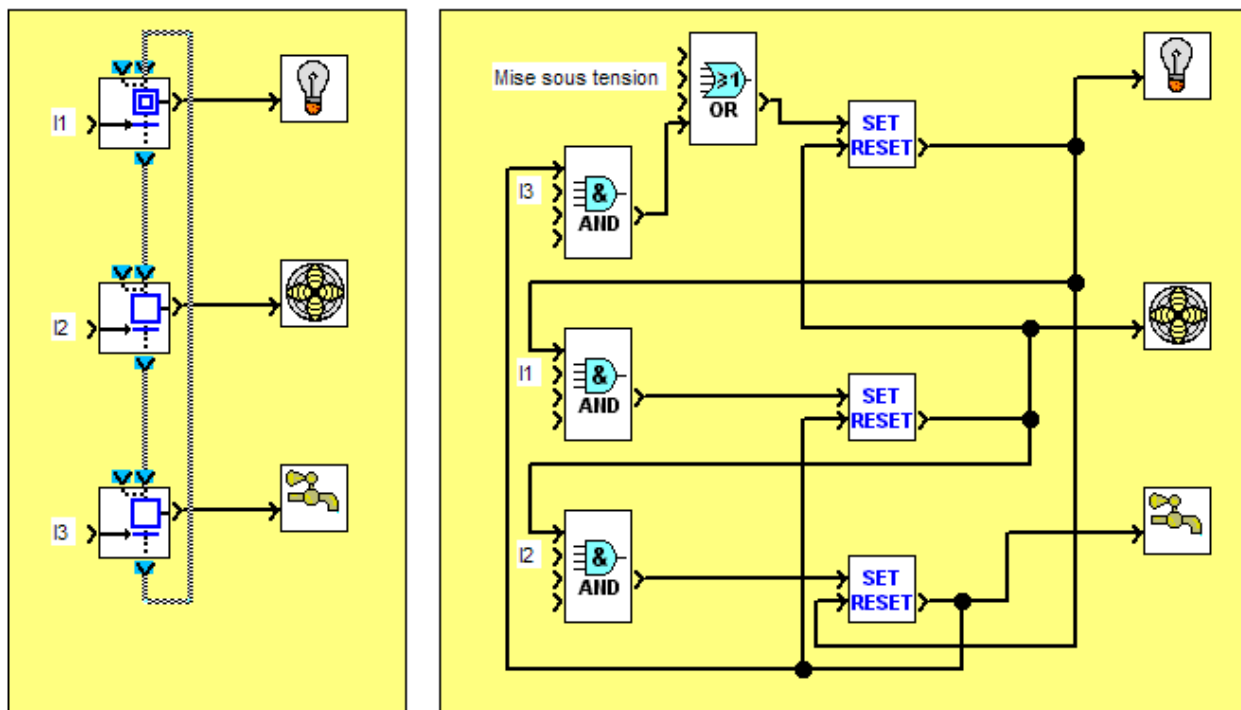


Figure 12 : Conversion Grafcet en SFC

### 3.9 Mise en équation

Équation logique des transitions

$$Y1 = X1 \cdot m$$

$$Y2 = X2 \cdot a$$

$$Y3 = X3 \cdot b$$

Équation logique des étapes

$$X1 = (Y3 + X1 \cdot Y1) + \text{Init}$$

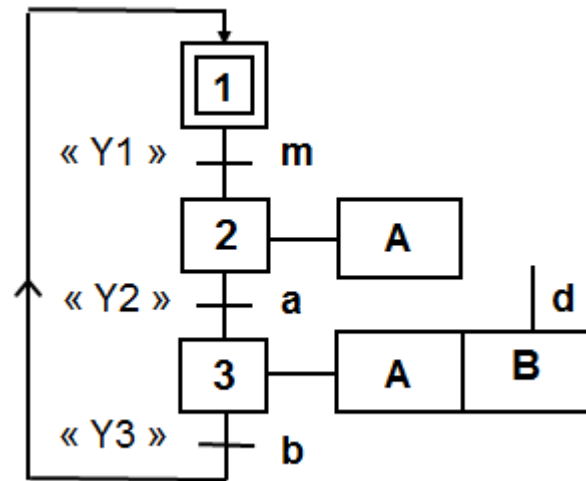
$$X2 = (Y1 + X2 \cdot Y2)$$

$$X3 = (Y2 + X3 \cdot Y3)$$

Équation logique des actions

$$A = X2 + X3$$

$$B = X3 \cdot d$$



#### 3.9.1 Mise en équation divergence OU

Mise en équation de la sélection de séquence

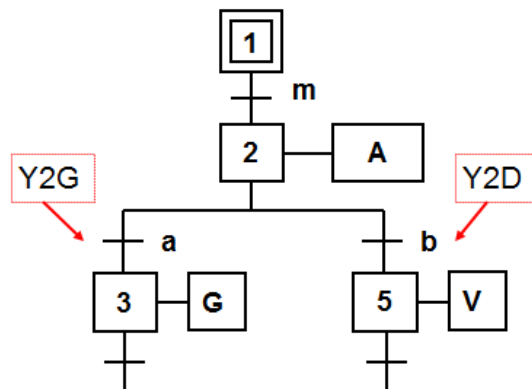
$$Y1 = X1 \cdot m$$

$$Y2G = X2 \cdot a$$

$$Y2D = X2 \cdot b$$

Équation de l'étape 2

$$X2 = Y1 + X2 \cdot Y2G / Y2D$$



#### 3.9.2 Mise en équation convergence OU

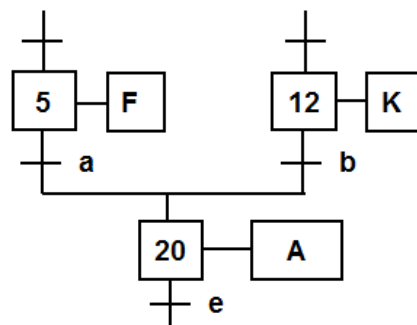
$$Y5 = X5 \cdot a$$

$$Y12 = X12 \cdot b$$

$$Y20 = X20 \cdot e$$

Équation de l'étape 20

$$X20 = (Y5 + Y12 + X20 \cdot Y20)$$



### 3.9.3 Mise en équation divergence ET

Synchronisation aval de séquences

$$Y1 = X1 \cdot m$$

$$Y2 = X2 \cdot e$$

Équation de l'étape 2

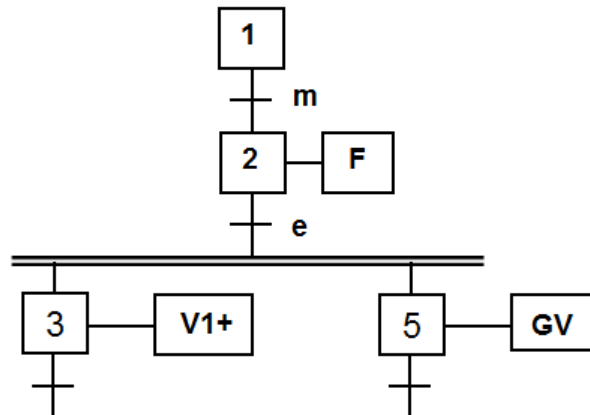
$$X2 = Y1 + X2 \cdot Y2 /$$

Équation de l'étape 3

$$X3 = Y2 + X3 \cdot Y3 /$$

Équation de l'étape 5

$$X5 = Y2 + X5 \cdot Y5 /$$



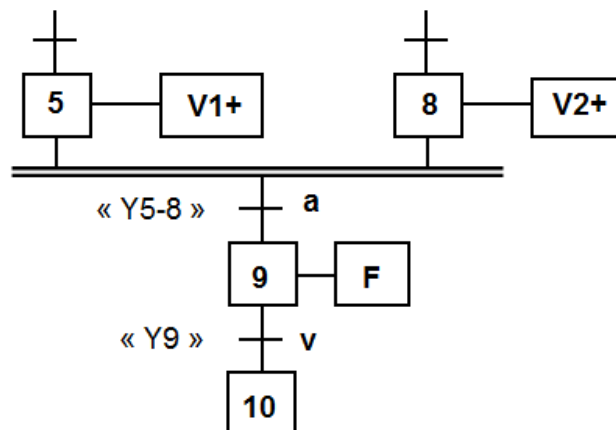
### 3.9.4 Mise en équation convergence ET

$$Y5-8 = X5 \cdot X8 \cdot a$$

$$Y9 = X9 \cdot v$$

Équation de l'étape 9

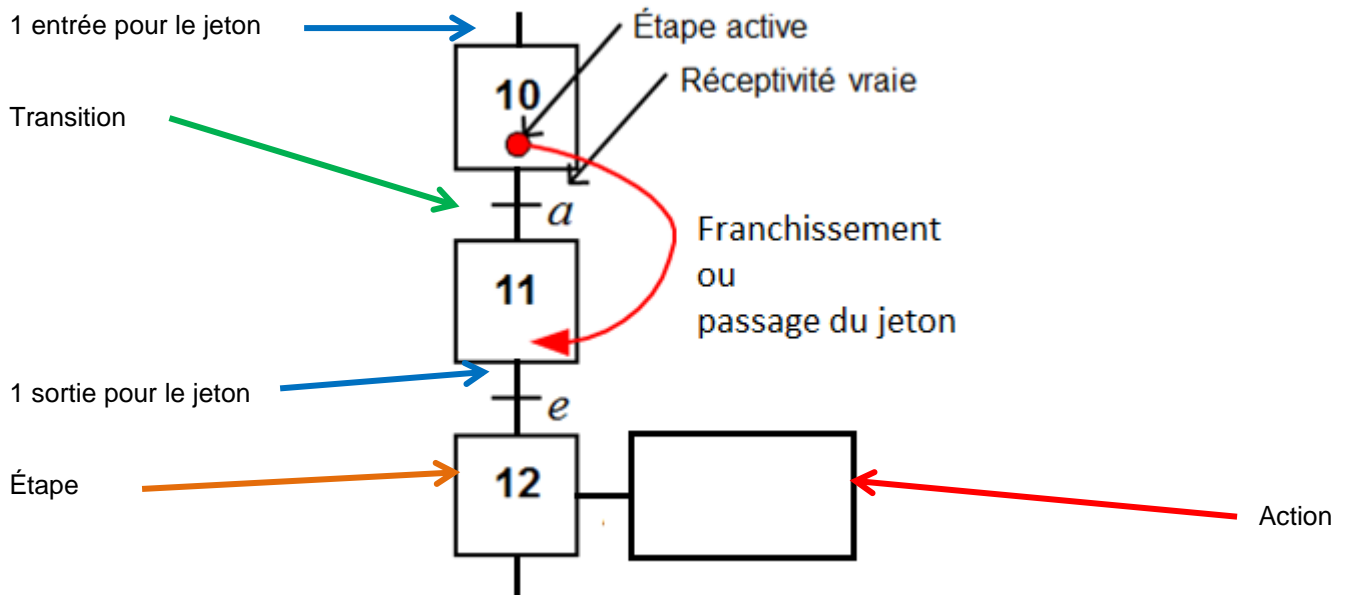
$$X9 = Y5-8 + X9 \cdot Y9 /$$



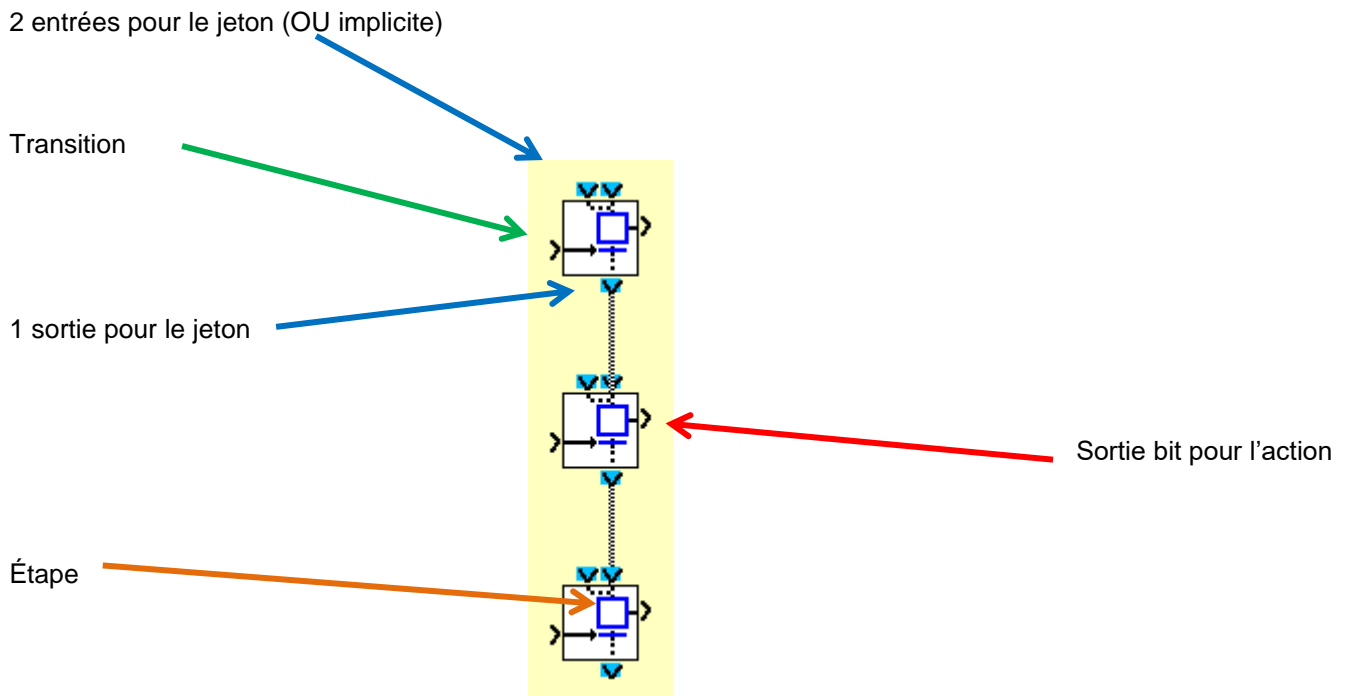


## 4 Les FB

### Représentation « Grafcet »



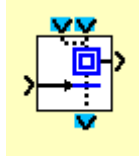
### Représentation « SFC » du Millenium



Le choix à été fait de mettre une étape et une transition dans le même FB.

## 4.1 FB Étape initiale (INIT STEP)

L'étape initiale est représentée par un double carré.



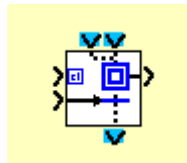
La fonction INIT STEP est initialisée automatiquement dans les cas suivants :

Passage en mode Simulation, en mode marche ou après un retour secteur.

Note : un graphe SFC doit contenir au moins un FB avec une étape initiale.

## 4.2 FB Étape initiale réinitialisable (RESET INIT)

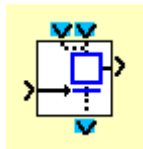
Il existe un FB particulier qui possède une entrée supplémentaire afin de se réinitialiser, l'étape concernée passe à 1 et les autres à 0.



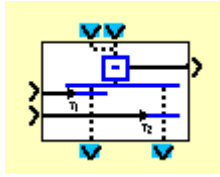
Note : un graphe SFC ne peut contenir qu'une seule fonction RESET INIT

## 4.3 FB Étape (STEP)

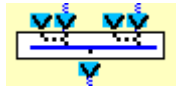
Si l'une des 2 Entrées (en bleu) est active alors la SORTIE D'ÉTAPE est activée et le reste même après la disparition des entrées.



#### 4.4 FB Divergence OU et convergence OU

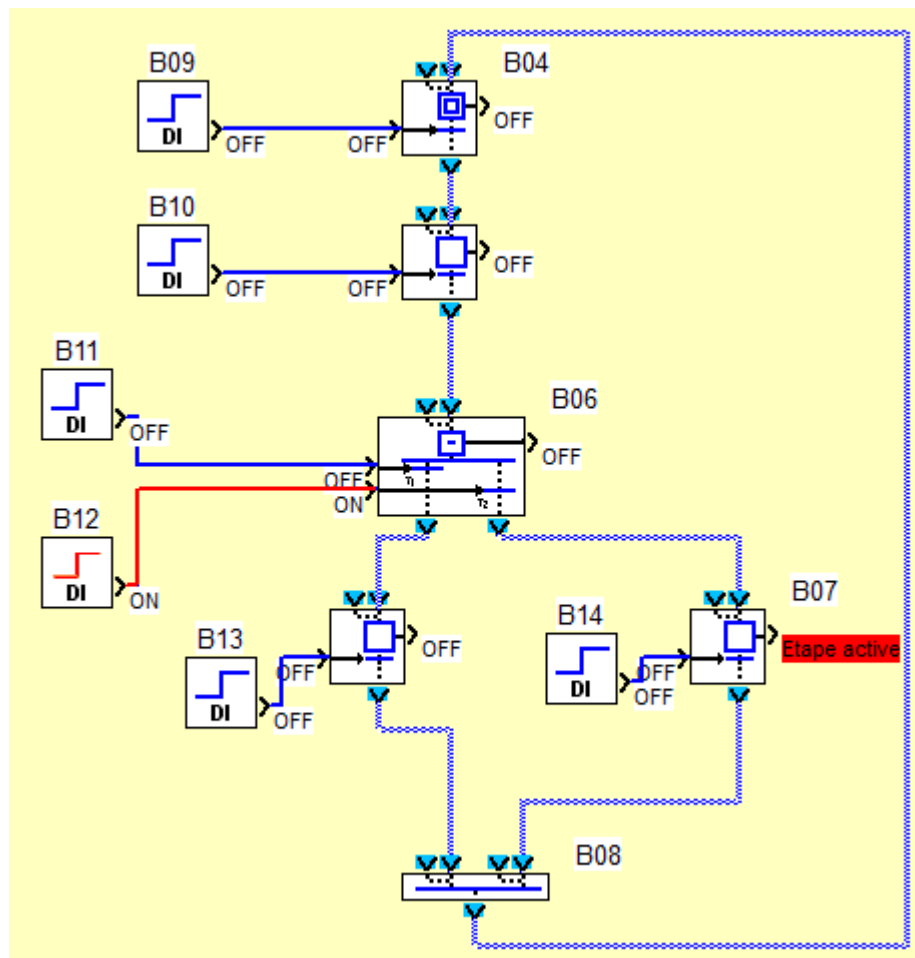


La **Divergence OU** permet de faire circuler le jeton dans l'une ou l'autre des 2 branches suivant l'état de la transition.

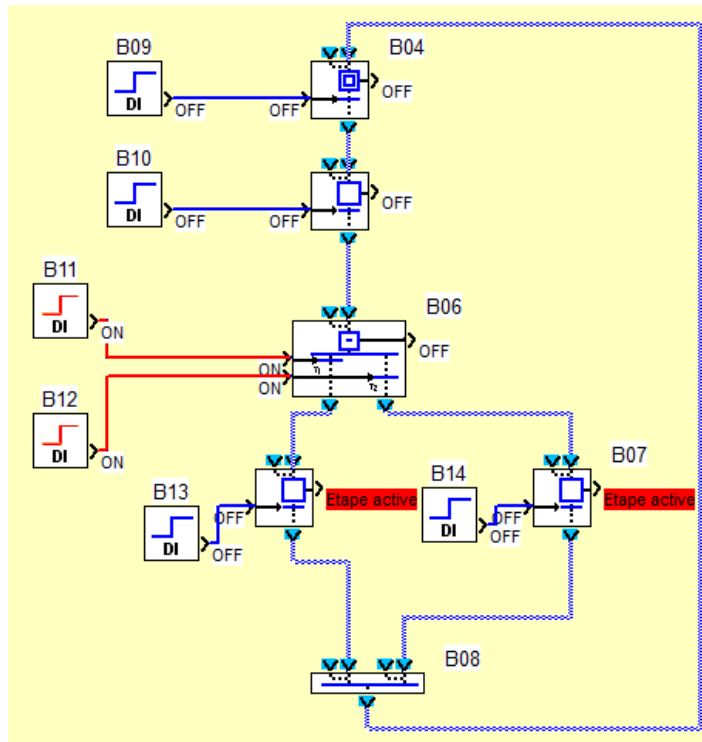


La **Convergence OU** permet de réunir jusqu'à 4 branches.

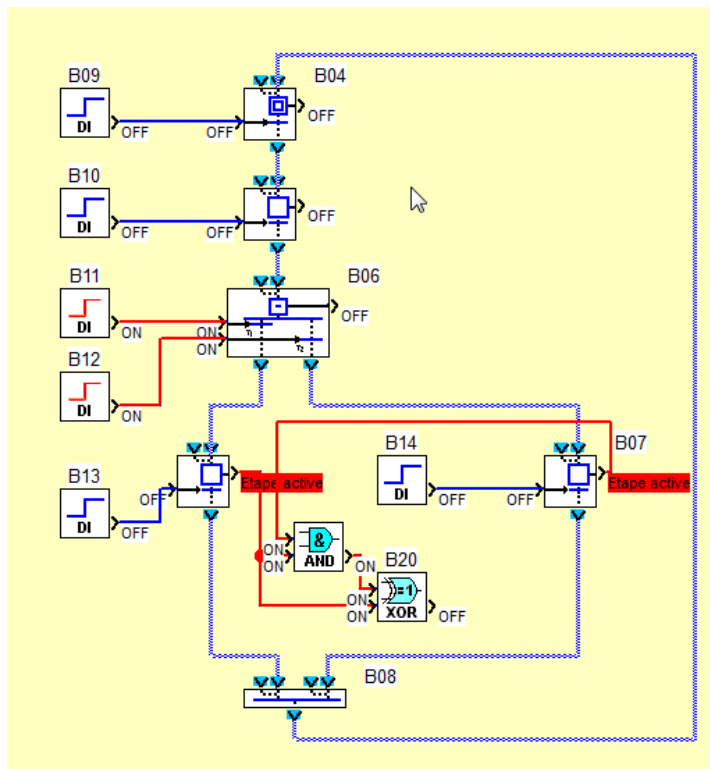
Exemple :



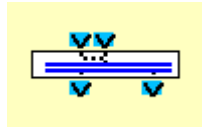
Attention, si les 2 transitions d'une divergence sont à 1 au moment du franchissement (ce n'est normalement pas possible car en pratique c'est généralement un choix exclusif), les 2 branches suivantes sont validées.



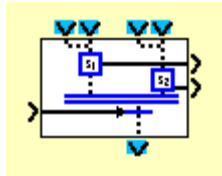
Si le risque existe, il faut ajouter un ET entre les 2 Etapes des 2 branches, et un OU EXclusif entre ce ET et l'Etape non prioritaire. Le nouveau résultat de la première branche est en sortie du OU EX (XOR).



## 4.5 FB Divergence ET et convergence ET

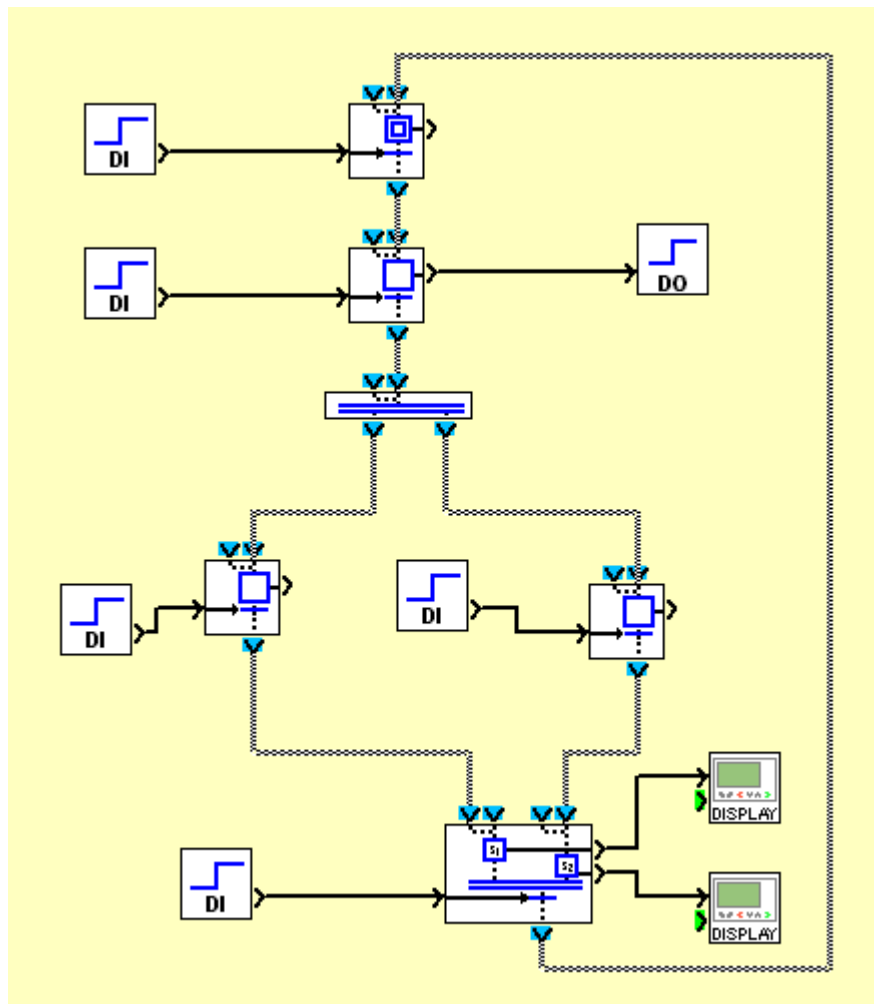


La **Divergence ET** permet de faire circuler le jeton dans les 2 branches.

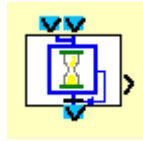


La **Convergence ET** permet de réunir des branches avec des étapes d'attente.

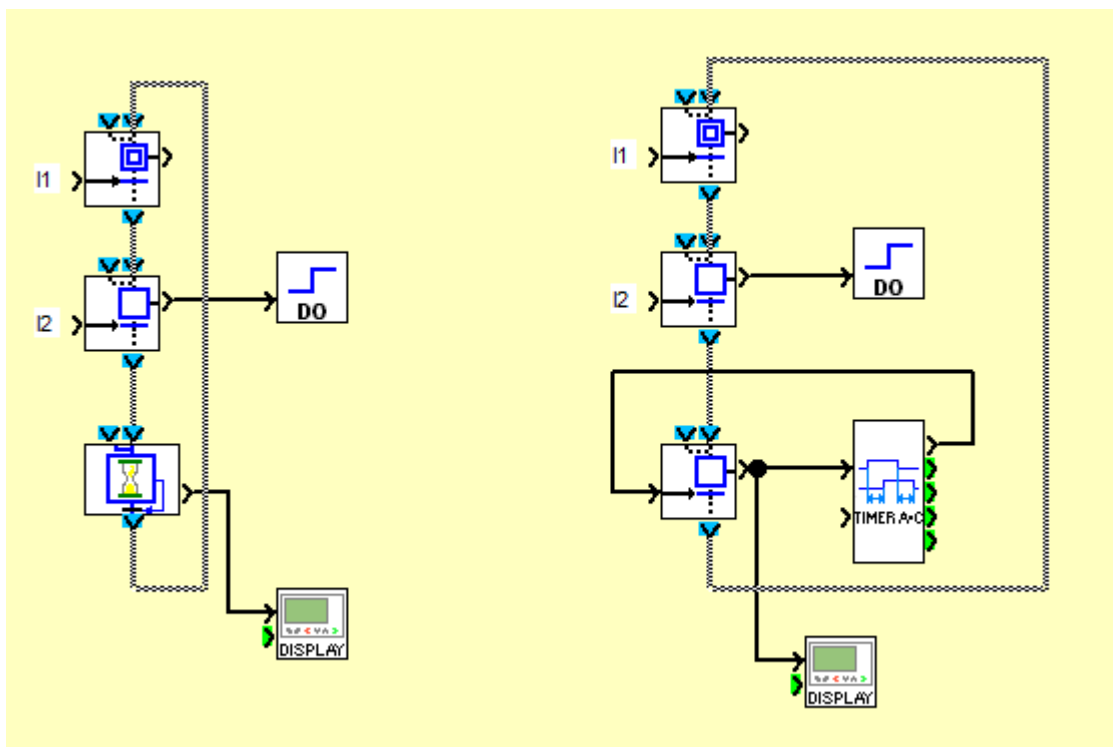
Exemple :



## 4.6 FB SFC Attente

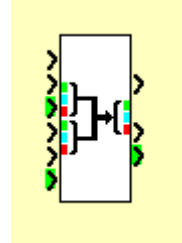
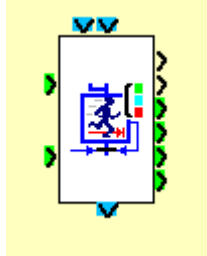


Ce FB n'a pas de transition, elle est implicite, c'est la fin de la temporisation. Celle-ci est paramétrable par double-clic sur le FB.



Ces 2 graphes sont équivalents.

## 4.7 FB SFC MOUVEMENT et Multiplexeur moteur



Ces 2 FB permettent de piloter un moteur.

