

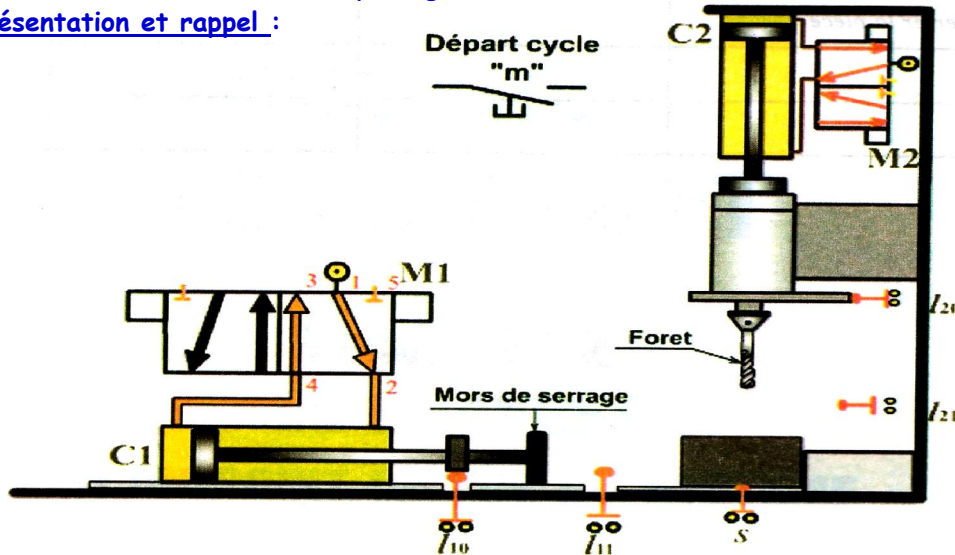
Leçon A4-1 :                   Automate Programmable Industriel

**Objectifs :**

- \* Décrire le fonctionnement d'un système automatisé à l'aide de l'outil GRAFCET.
- \* Identifier les éléments de dialogue d'un système automatisé piloté par API.
- \* Traduire un GRAFCET en langage automate.
- \* Ecrire ou compléter un programme et l'implanter sur un API.

**I-Mise en situation : (Unité de perçage)**

**1-Présentation et rappel :**



**Description :** La poste de perçage est composée essentiellement d'un système de serrage et de perçage. Les vérins C<sub>1</sub> et C<sub>2</sub> sont à double effet, pilotés par des distributeurs 5/2, bistables et à commande électromagnétique. Les pièces à percer arrivent en position a travers une rampe.

**Fonctionnement :** la mise d'une pièce en position de perçage se fait manuellement, un capteur s indiquant la présence d'une pièce déclenche les actions suivantes : **serrage puis perçage**

**Travail demandé :** Décrire le fonctionnement du système par un **GRAFCET** selon les 3 points de vue : **système-PO-PC ?**

**\* GRAFCET point de vue système :**

C'est un graphe qui décrit le **fonctionnement global du système**. Il traduit le **cahier des charges** sans préjuger de la technologie adoptée. Il permet de dialoguer avec des personnes non spécialistes (fournisseurs, décideurs ...) Son écriture, en langage clair, permet donc sa compréhension par tout le monde.

**\* GRAFCET point de vue partie opérative :**

Dans ce type de grafcet on **spécifie la technologie de la partie opérative** ainsi que le type de ses informations reçues (ordres) et envoyées (comptes-rendus). L'observateur de ce point de vue étant un spécialiste de la partie opérative, la partie commande ne l'intéresse que par ses effets.

**\* GRAFCET point de vue partie commande :**

Ce grafcet est établi en **spécifiant la technologie des éléments de dialogue :**

- \*entre PC et PO ;
- \*entre PC et opérateur ;
- \*entre PC et autre système.

C'est un grafcet établi par un spécialiste, c'est la version qui lui permet d'établir les **équations** et éventuellement les **schémas de réalisation** (électrique, pneumatique ...)

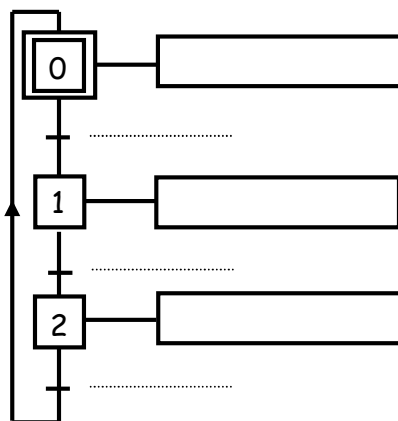
**Régle1** : Les étapes INITIALES sont celles qui sont actives au début du fonctionnement. On les représente en doublant les côtés des symboles. On appelle début du fonctionnement le moment où le système n'a pas besoin de se souvenir de ce qui c'est passé auparavant (allumage du système, bouton "reset",...). Les étapes initiales sont souvent des étapes d'attente pour ne pas effectuer une action dangereuse par exemple à la fin d'une panne de secteur.

**Régle2** : Une TRANSITION est soit validée, soit non validée (et pas à moitié validée). Elle est validée lorsque toutes les étapes immédiatement précédentes sont actives. Elle ne peut être FRANCHIE que lorsqu'elle est validée et que sa réceptivité est vraie. Elle est alors obligatoirement franchie.

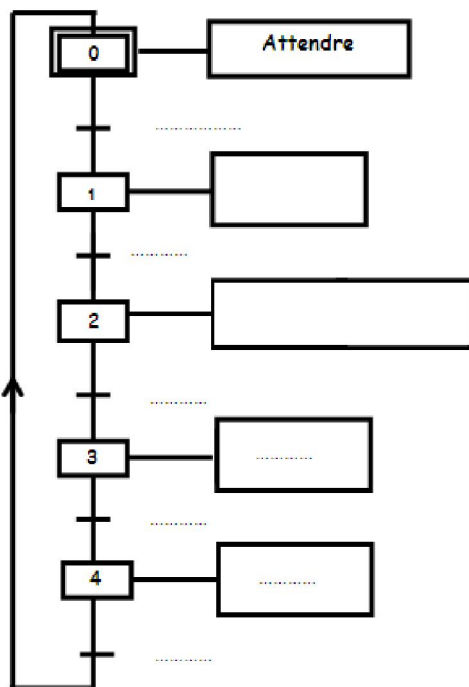
**Régle3** : Le FRANCHISSEMENT d'une transition entraîne l'activation de TOUTES les étapes immédiatement suivante et la désactivation de TOUTES les étapes immédiatement précédentes.

**Régle4** : Plusieurs transitions SIMULTANEMENT franchissables sont simultanément franchies.

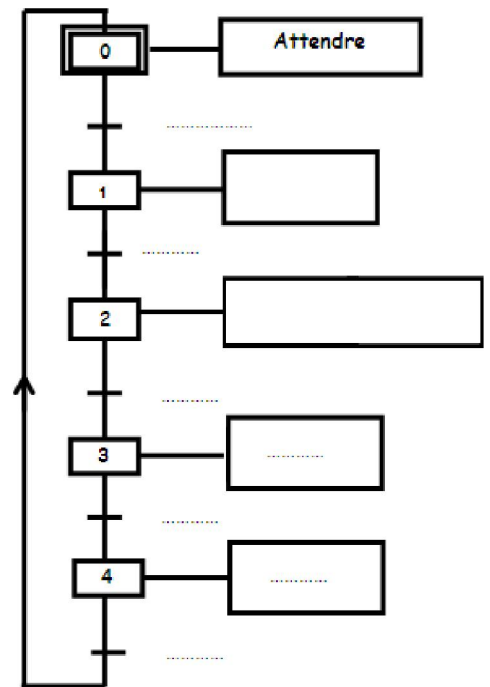
**Régle5** : Si une étape doit être à la fois activée et désactivée, elle RESTE active.



GRAFCET point de vue système



GRAFCET point de vue PO



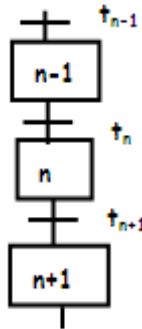
GRAFCET point de vue PC

2-Mise en équation d'une étape d'un GRAFCET :

a-Règle générale :

Pour qu'une étape soit active il faut :

- L'étape immédiatement précédente soit active.
- La réceptivité immédiatement précédente soit vraie.
- L'étape immédiatement suivante soit non active.
- Après activation, l'étape mémorise son état.



Considérons l'étape n :

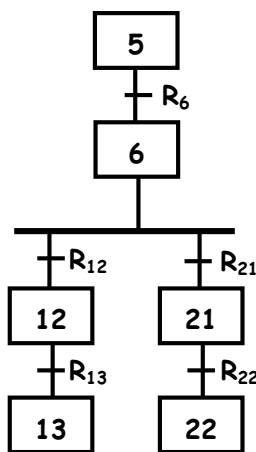
Equation d'activation:  $A_n = \dots\dots\dots$

Equation de désactivation :  $D_n = \dots\dots\dots$

Equation de l'étape:  $X_n = \dots\dots\dots$

b-Différents cas de mise en équations :

**\* Divergence en OU :**



Equation de l'étape 6 :

$X_6 =$

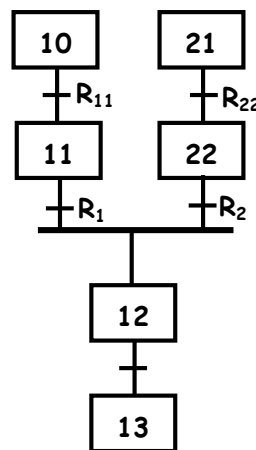
Equation de l'étape 12 :

$X_{12} =$

Equation de l'étape 21 :

$X_{21} =$

**\* Convergence en OU :**



Equation de l'étape 11 :

$X_{11} =$

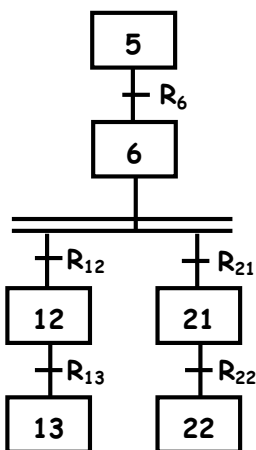
Equation de l'étape 22 :

$X_{22} =$

Equation de l'étape 5 :

$X_{12} =$

**\* Divergence en ET :**



Equation de l'étape 6 :

$X_6 =$

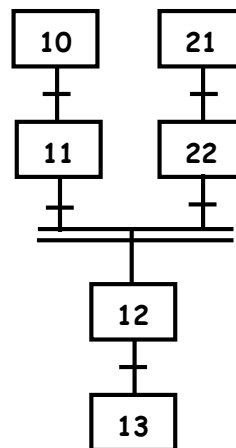
Equation de l'étape 12 :

$X_{12} =$

Equation de l'étape 21 :

$X_{21} =$

**\* Convergence en ET :**



Equation de l'étape 11 :

$X_{11} =$

Equation de l'étape 22 :

$X_{22} =$

Equation de l'étape 5 :

$X_{12} =$

## II- Automate programmable industriel :

### 1-Fonction :

L'automatisation d'un système technique consiste à assurer la commande en adaptant un outil technologique. Le choix de l'outil technologique dépend du procédé et de sa complexité.

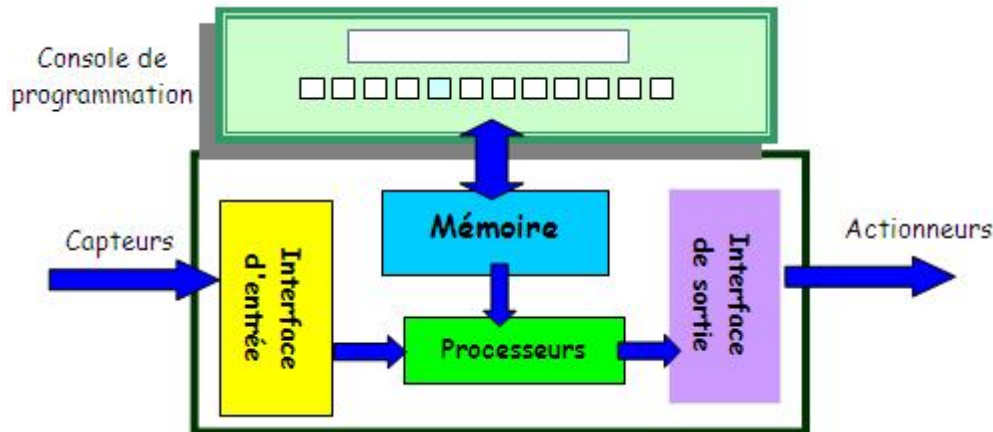
L'automaticien dispose en général deux solutions :

Solution câblée

Solution programmée :-micro-ordinateur

-automate programmable

### 2- Architecture d'un API :



Les différents modules de l'API sont :

- **Module d'entrées** : Ils permettent à l'unité centrale de recevoir les informations logiques des capteurs et des boutons poussoirs du pupitre reliés à l'API.
- **Module de sorties** : ils permettent à l'unité centrale d'émettre des informations ordres de types logiques et électriques vers les pré-actionneurs de l'API.
- **Unité centrale**: l'unité centrale de traitement est l'ensemble électrique qui effectue le travail dans la mémoire programme. La mémoire programme enregistre les instructions du programme de commande ainsi que les données stockées à un endroit précis appelé " PAS" de programme. Les trois parties sont reliées entre elles par des " BUS " (les bus sont des faisceaux de huit fils parallèles, chaque fil véhicule un bit : 0 ou 1).
- **Module d'alimentation** : deux alimentations de 24v continues ou alternatives pour assurer la liaison galvanique entre l'automate et le module d'entrées.

### 3-Choix d'un API :

Le choix d'un API est fonction de la partie commande à programmer. Choisir un API, ça revient à consulter les caractéristiques techniques suivantes :

- Le nombre maximal d'entrées et de sorties.
- La capacité de la mémoire programmé.
- Le nombre de variables internes.
- Le nombre de compteurs.
- Le nombre des temporisateurs.
- La fréquence d'exécution.

### 4-Programmation en langage automate :

#### a-Définition :

Un programme est une suite d'instructions, traduisant le fonctionnement du système automatisé conçu à être traité et exécuter par un automate.

**b-Mode de programmation :**

On peut programmer un automate en mode :

- \*Schéma à contacts
- \*Logigrammes
- \*Lignes d'instructions ou GRAFCET

**c-Types d'automates :**

- \* Automates programmables " Syrelec " :
  - série SLP 3600
  - série SPP 2200
- \* Automates programmables Télémécaniques :
  - série TSX21 ; TSX17 ; TSX17-20 ; TSX27, 47, 67,87
- \* Automates programmables AEG
  - série 924 ; A120
- \* Automates programmables Jeulin
  - série EDUC 1000

**d- Programmation sur AEG 020 :**

**\*Opérandes :**

Entrées	24 entrées: I1 à I24
Sorties	16 sorties : de Q1 à Q16
Variables internes	124 variables internes: de M1 à M124, de M125 à M128 (autres bases de temps)
Temporisateurs	16 temporisateurs: de T1 à T8 (base de temps 100ms), de T8 à T16 (base de temps 25ms)
compteurs	16 compteurs de C1 à C16

**\*Opérateurs :**

Type d'opération	Opérateur	Action	Opérandes utilisables
Opérations logiques	A	Opération logique ET, signal positif	I, T, Q, C
	AN	Opération logique ET, signal négatif	I, T, Q, C
	O	Opération logique OU, signal positif	I, T, Q, C
	ON	Opération logique OU, signal négatif	I, T, Q, C
	A(	Opération logique ET, parenthèse ouverte	
	O (	Opération logique OU, parenthèse ouverte	
	)	Parenthèse fermée, signal positif	
Opérations de sorties	) N	Parenthèse fermée, signal négatif	
	=	Sortie positive	Q, M
	=N	Sortie négative	Q, M
	SL	Activation mémoire	Q, M
Opération comptage	RL	Désactivation mémoire	Q, M
	=T	Entrée temporisateur (sortie temporisations)	
	=Z	Transfert consigne compteur (effacement)	
Opérations d'organisation du programme	=P	Entrée compteur (C : sortie compteur)	
	JI	Saut si "1" (condition positif)	I, Q
	LS	Chargement immédiat (en mémoire de signaux)	
	NO	Sans effet, opération nulle	
	PE	Fin du programme	

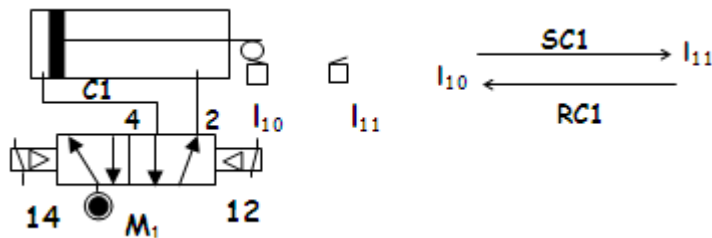
Au début de chaque programme sur AEG A020, il faut activer l'étape initiale, pour cela on peut procéder de différentes manières :

- AM128
- SLM1
- AM128
- RLM128

**5-Applications :**

**Exemple1 : GRAFCET à séquence unique**

Cycle pendulaire : L'action sur un bouton départ cycle **m** déclenche le cycle suivant :

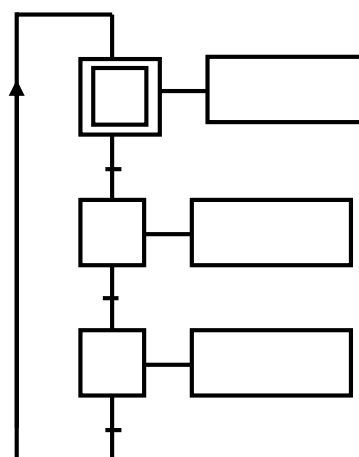
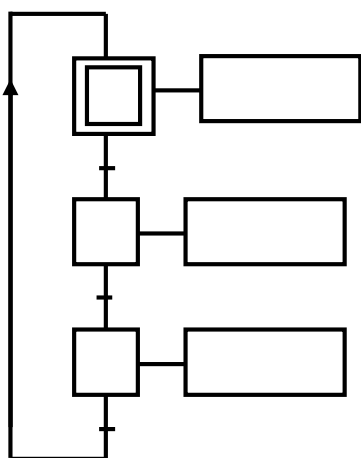


**\*Table des affectations :**

Entrées système	Entrée AEG	Sortie système	Sortie AEG

**GRAFCET P.C**

**GRFCET codé automate**



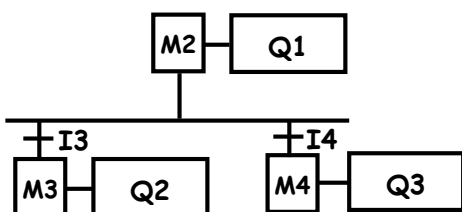
\* Programme en langage automate AEG020 :

Titre	Adr: instr	Commentaire	Information
1:			
2:			
3:			
4:			
5:			
6:			
7:			
8:			
9:			
10:			
11:			

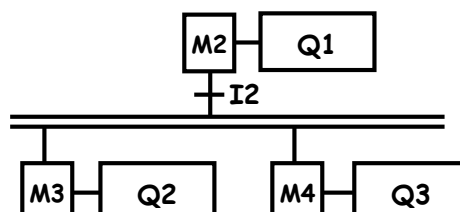
\*Activité 1 : Simuler le programme saisie sur le logiciel AKL

\*Règles Important à appliqués

Divergence en OU



Divergence en ET



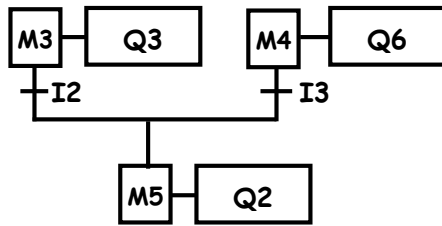
Désactivation de l'étape M2

.....  
 .....  
 .....

Désactivation de l'étape M2

.....  
 .....  
 .....

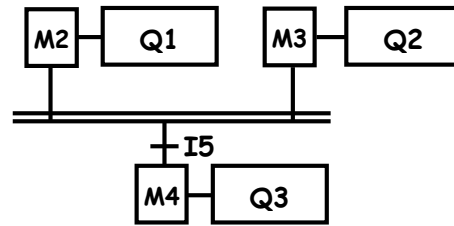
Convergence en OU



Activation de l'étape M5

.....  
 .....  
 .....  
 .....  
 .....  
 .....

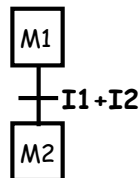
Convergence en ET



Activation de l'étape M4

.....  
 .....  
 .....  
 .....

Remarque importante :



Activation de l'étape M2 :

.....  
 .....  
 .....  
 .....

Exemple2: Divergence en ET : circuit de manutention :

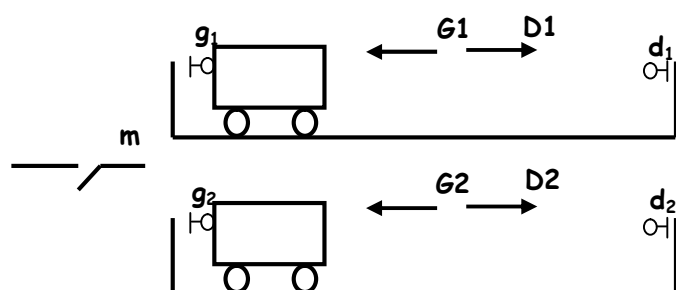
**Fonctionnement :** L'appui sur **m** déclenche le cycle suivant :

- Départ de deux chariots en même temps.
- Le chariot qui arrive le premier au point d'arrivée reste immobile jusqu'à ce que l'autre arrive à son point d'arrivée.
- Les deux chariots marquent un arrêt de **60** secondes depuis l'arrivée du dernier, puis ils reviennent vers le point de départ.
- Un nouveau cycle ne peut se faire que si les deux chariots atteignent leurs positions de départ. Les deux chariots **Ch1** et **Ch2** sont entraînée respectivement par deux moteurs

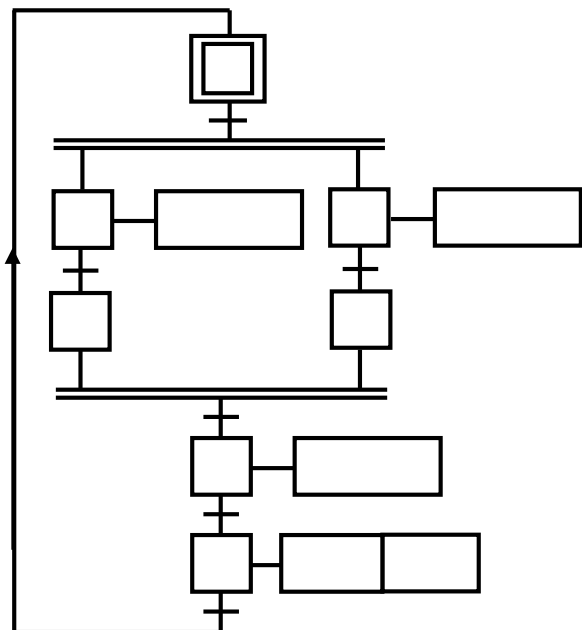
**Mt1** [KM1<sub>(d)</sub>, KM2<sub>(g)</sub>] et **Mt2** [KM3<sub>(d)</sub>, KM4<sub>(g)</sub>].

\*Table des affectations :

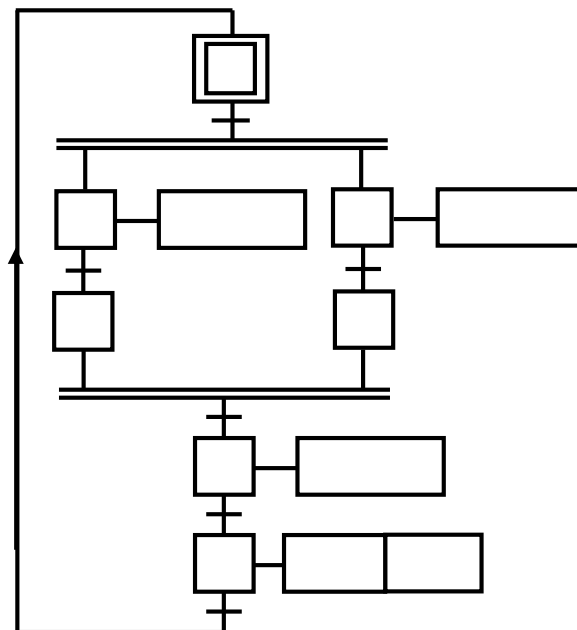
Entrées Système	Entrées AEG	Entrées Système	Entrées AEG







GRAFCET point de vue PC



GRAFCET codé automate

\*Programme en langage automate AEG020 :

Cir	Instr	Comment	Cir	Instr	Comment	Cir	Instr	Comment

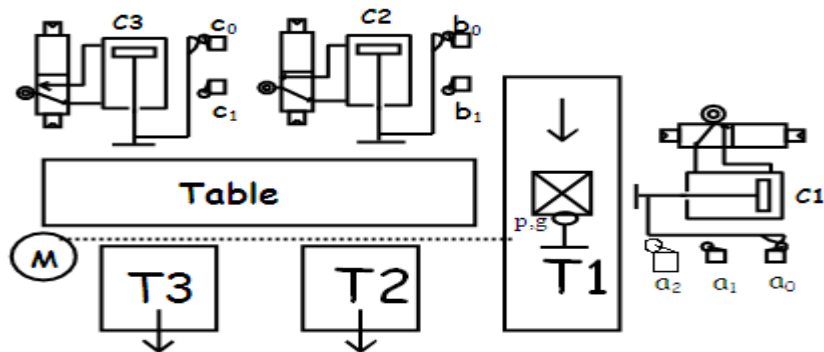
**\*Activité 2 : Simuler le programme saisie sur le logiciel AKL**

**Exemple3 : Divergence en OU: Machine de tri de caisses**

**Description :** La machine de tri de caisses représentée par la figure ci-dessous comprend :

- Trois tapis **T1**, **T2** et **T3** entraînés par un moteur **M** qui ne fait pas parti de l'étude.
- Trois vérin **C1**, **C2** et **C3** à double effet muni chacun d'un distributeur 5/2 à commande pneumatique et dont la commande est asservie à la marche du moteur **M**.

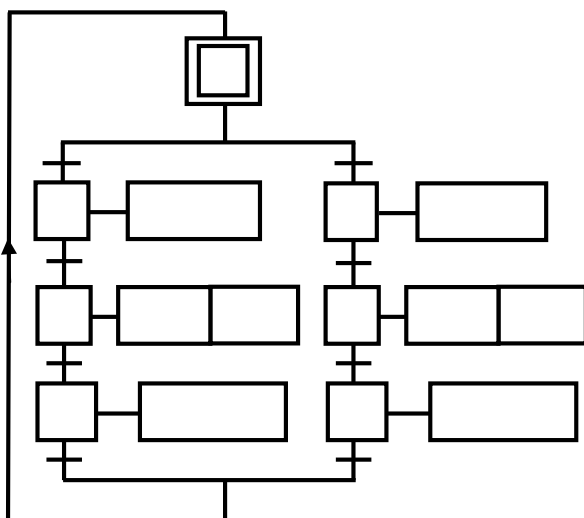
**Fonctionnement :** Les caisses amenées par un tapis **T1** sont de dimensions différentes (petites et grandes). Deux capteurs **p** et **g** détectent respectivement les petites caisses et les grandes caisses. La petite caisse est poussée par le vérin **C1** en face du vérin **C2** qui à son tour l'évacue vers le tapis **T2**. La grande caisse est poussée par le vérin **C1** en face du vérin **C3** qui à son tour l'évacue vers le tapis **T3**.



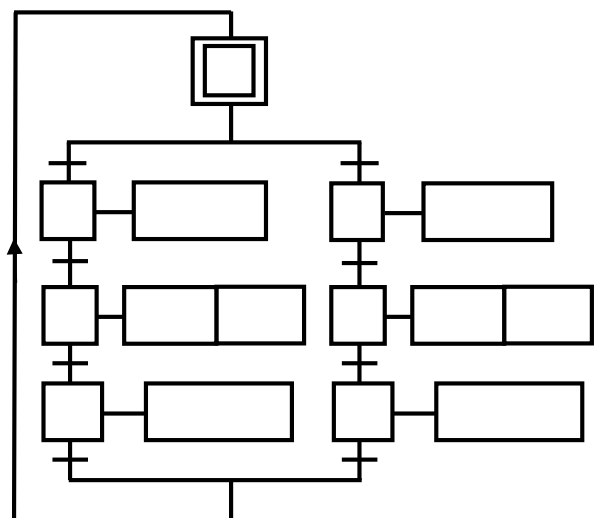
**\*Table des affectations :**

Entrées Système	Entrées AEG	Entrées Système	Entrées AEG

GRAFNET point de vue PC



GRAFNET codé automate

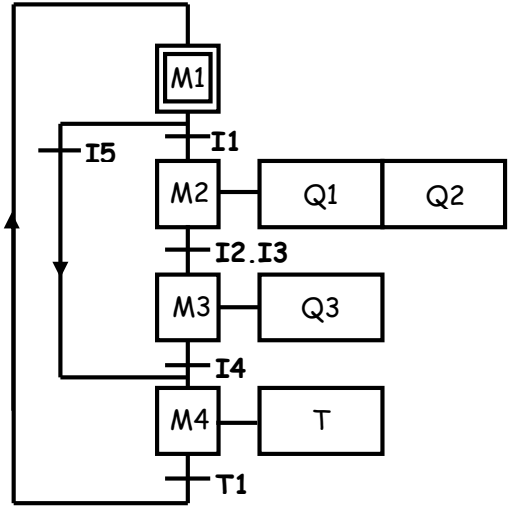


\*Programme en langage automate AEG020 :

Cir	Instr	Comment	Cir	Instr	Comment	Cir	Instr	Comment

**\*Activité 3 :** Simuler le programme saisie sur le logiciel AKL

**Exemple 4 :** On donne le GRAFCET codé automate (saut d'étape) suivante, donner son programme :



\*Programme en langage automate AEG020 :

Cir	Inst	Com	Cir	Inst	Com	Cir	Inst	Com

\*Activité 4 : Simuler le programme saisie sur le logiciel AKL