

Note de cours

INFORMATIQUE INDUSTRIELLE

Niveau : 4^{ème} Année

Département : AEEE

Préparé par : Pr SABOR Jalal, Pr HAFIANE Morad

Chapitre I : Le GRAFCET

(GRaphe Fonctionnel de Commande, Etape et Transition)

I. Définition

En 1977, le GRAFCET a été développé en France pour être un outil de conception et de programmation des systèmes automatisés de production. Il a été normalisé en 1987.

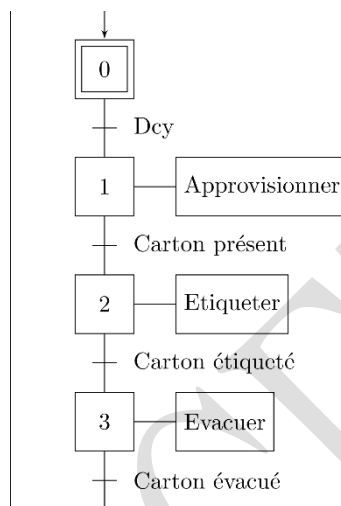


Figure 1 : Exemple d'un GRAFCET

Il permet de traduire et de normaliser de manière graphique un cahier des charges selon trois niveaux :

1. GRAFCET niveau 1 :

Description fonctionnelle **sans tenir compte** de l'aspect technologique des capteurs et actionneurs

2. GRAFCET niveau 2 :

Description de l'automatisme en **tenant compte** de l'aspect technologique des capteurs et actionneurs.

3. GRAFCET niveau 3 :

Opérationnel en tenant compte de la fiabilité, maintenabilité et mode marche/arrêt du processus.

Aussi la représentation d'un système automatisé par un GRAFCET prend en compte le "point de vue" selon lequel l'observateur s'implique au fonctionnement de ce système. On distingue trois "points de vue" :

- GRAFCET du point de vue système ;
- GRAFCET du point de vue partie opérative ;
- GRAFCET du point de vue partie commande.

Le fonctionnement d'un automatisme logique peut être représenté par un ensemble d'**ETAPES** auxquelles sont associées des **ACTIONS**, de **TRANSITIONS** auxquelles sont associées des **RECEPTIVITES** et des **LIAISONS** orientées.

II. Les éléments du GRAFCET

1. Etape

Une situation stable où il se trouve l'automatisme durant son évolution., elle est représentée par un carré ou rectangle avec un numéro et un nom en option.



Figure 2 : Etape initiale (0), Etape normale (1)

Une étape est soit active, soit inactive. Elle est dite active lorsque le système ou une partie du système se trouve dans la situation qui correspond à cette étape.

La situation d'un automatisme est définie par les étapes actives.

2. Action

On associe pour chaque étape une ou les actions à effectuer. Ces dernières ne sont exécutées que lorsque l'étape est active. Elles sont listées à gauche de l'étape auxquelles sont associées.

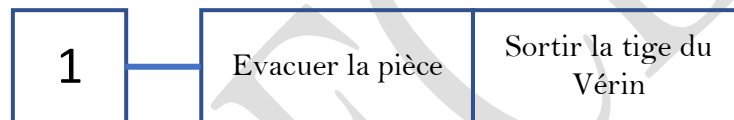


Figure 3 : Actions associées à l'étape 1

3. Transition :

C'est l'évolution ou passage entre les différentes étapes. On associe à chaque transition une condition logique appelée « Réceptivité ». Elle est représentée par une barre horizontale.

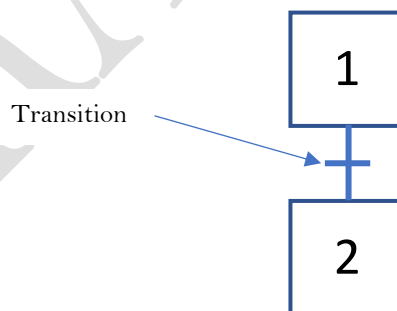


Figure 4 : Transition

4. Réceptivité :

C'est une condition logique formée par les états de capteurs. Elle est placée à **droite** de la transition.

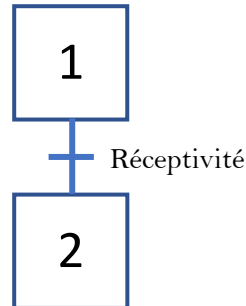


Figure 5 : Réceptivité associée à une transition

5. Liaisons orientées :

Elles indiquent les voies d'évolution de la situation du GRAFCET. Elles relient les étapes et transitions, elles sont représentées par une barre verticale sans ou avec une flèche.

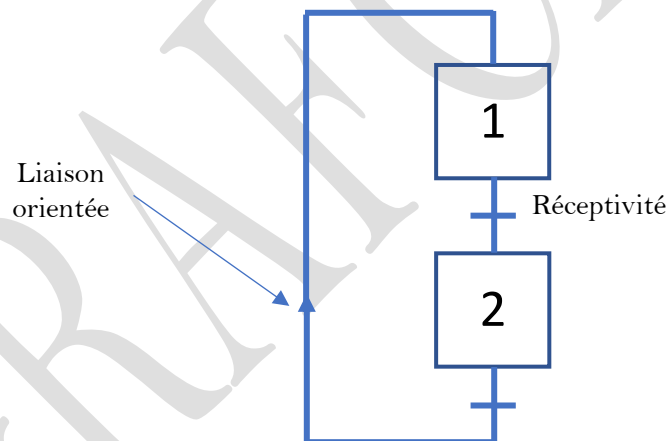


Figure 6 : Liaison orientée

Remarque

- ✓ Une étape est dite active lorsque l'automatisme se retrouve dans cette étape.
- ✓ Les actions qui lui sont associées ne sont exécutées que lorsque l'étape est active.
- ✓ Le GRAFCET est parcouru par défaut du haut vers le bas.

Exemple : Poinçonneuse

A l'état initial le poinçon est en position haute, l'opérateur installe la pièce, une action sur marche fait descendre le poinçon jusqu'à la position basse, puis il retourne en position initiale

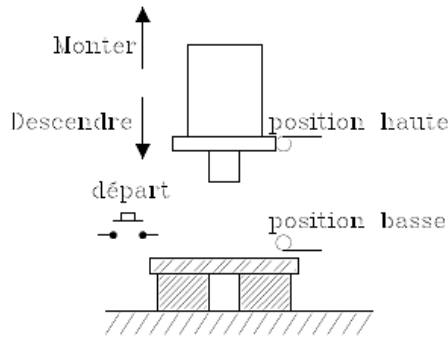


Figure 7 : Poinçonneuse à vérin

GRAFCET fonctionnel

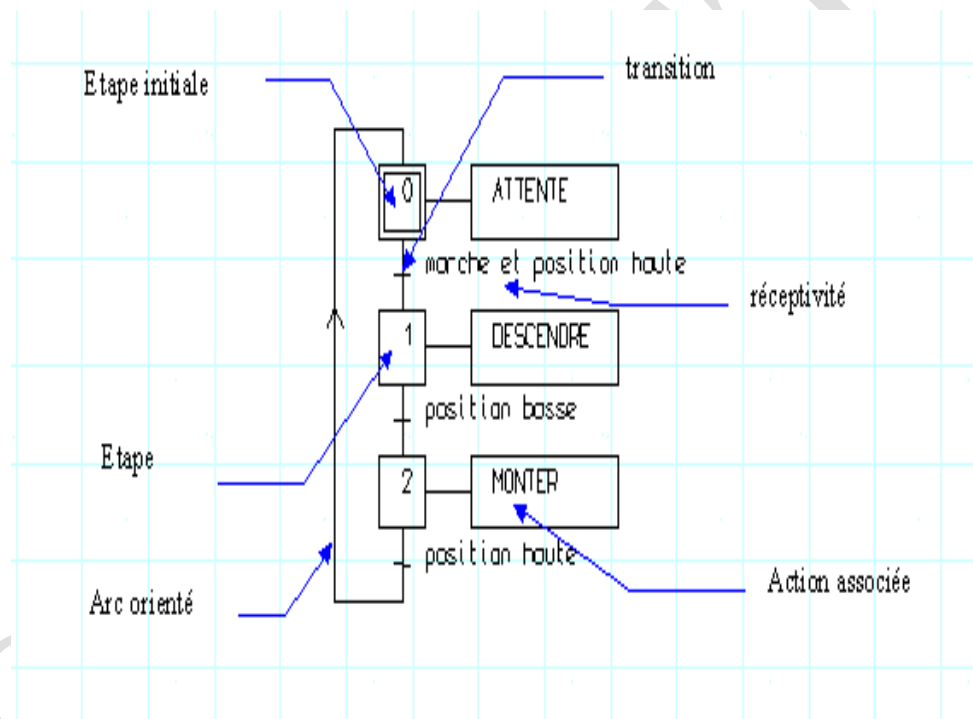


Figure 8 : GRAFCET au point de vue fonctionnel de la poinçonneuse

Cycle de fonctionnement d'une perceuse

Une fois la broche est en position haute **h**, et elle est détectée en rotation permanente, l'opérateur donne l'information départ cycle. La broche descend avec une grande vitesse jusqu'au position **b1**, puis elle effectue le perçage en petite vitesse jusqu'au position **b2**. Dès le perçage est terminé la broche remonte en grande vitesse jusqu'au position haute.

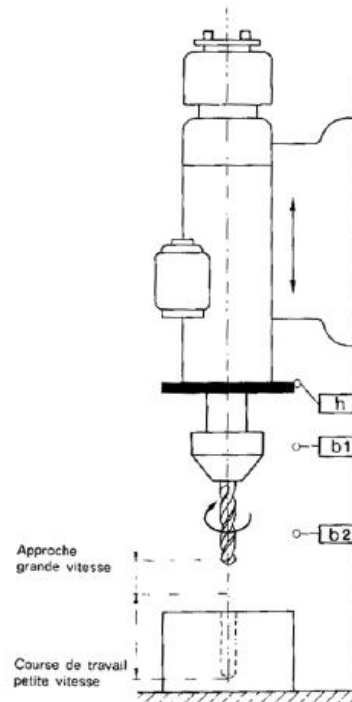


Figure 9 : schéma de la perceuse

GRAFCET de point de vue système

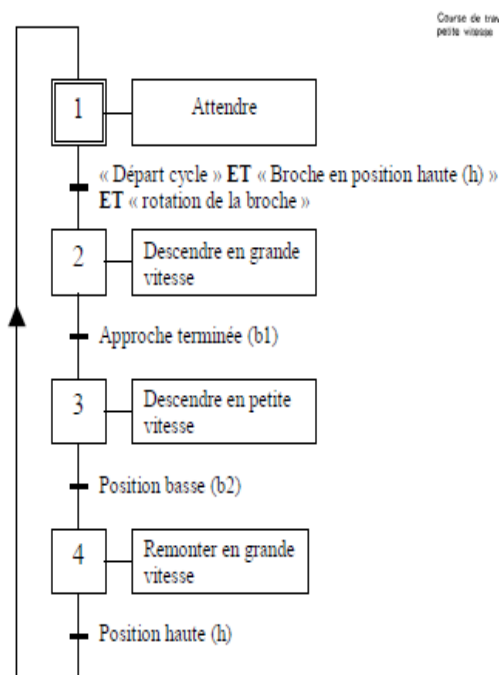


Figure 10 : GRAFCET au point de vue système de la perceuse

III. Les règles d'évolution

1. Règle N°1 : Situation initiale

La situation initiale d'un GRAFCET caractérise le comportement initial de la partie commande vis-à-vis de la partie opérative, elle correspond aux étapes actives au début du fonctionnement : ces étapes sont les étapes initiales.

L'étape initiale est représentée par un double carré ou double rectangle

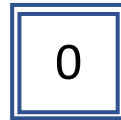


Figure 11 : Etape initiale

2. Règle 2 : Franchissement d'une transition

Une transition est dite validée lorsque toutes les étapes amonts (immédiatement précédentes reliées à cette transition) sont actives.

Le franchissement d'une transition se produit :

- Lorsque la transition est validée.
- La réceptivité associée à cette transition est vraie

Remarque

Lorsque les deux conditions sont vérifiées, la transition est obligatoirement franchie.

3. Règle 3 : Evolution des étapes actives

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

4. Règle 4 : Evolution simultanée

Plusieurs transitions simultanément franchissables sont simultanément franchies.

5. Règle 5 : Activation et désactivation simultanée d'une étape

Si au cours du fonctionnement la même étape est simultanément activée et désactivée elle reste active.

IV. Compléments sur GRAFCET

1. Divergence en OU

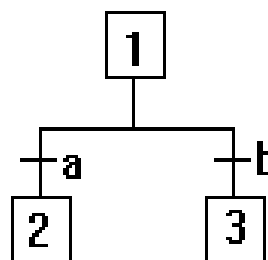


Figure 12 : Divergence en OU

Si étape 1 active :

- Si $a=1$ seul alors désactivation de 1 activation étape 2 et 3 inchangée.
- Si $a=1$ et $b=1$, lorsque 1 devient active, alors désactivation 1, activation 2 et 3.

2. Convergence en OU

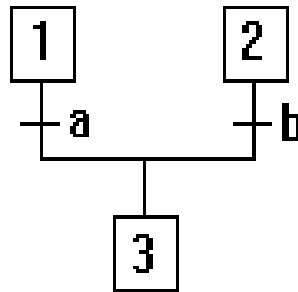


Figure 13 : Convergence en OU

Si étape 1 active :

- Et $a=1$, $b=0$ alors activation de 3 et désactivation de 1. 2 reste inchangée
- Si 1 et 2 active et $a=1$, $b=1$ alors désactivation 1 et 2, activation 3

3. Sélection exclusive

Pour obtenir une sélection exclusive, il est nécessaire de s'assurer que toutes les réceptivités associées aux transitions sont exclusives au point de vue physique, c'est à dire qu'elles ne peuvent pas être vraies simultanément.

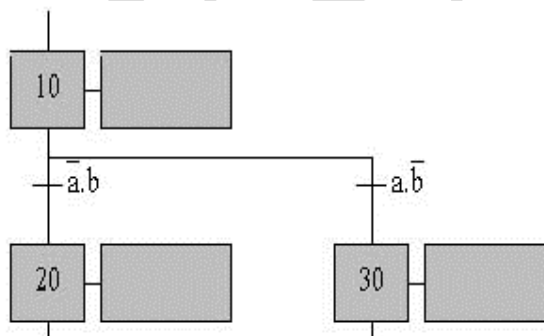


Figure 14 : Sélection sélective

4. Divergence en ET

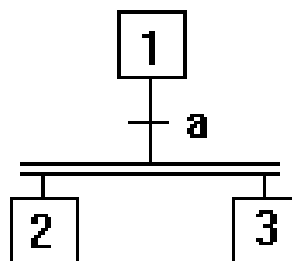


Figure 15 : Divergence en ET

Si 1 active et si $a=1$, alors désactivation de 1 et activation de 2 et 3.

5. Convergence en ET

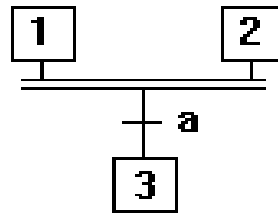


Figure 16 : Convergence en ET

Si 1 active seule et a alors aucun changement.

Si 1 et 2 et a , alors activation de 3 et désactivation de 1 et 2.

6. Saut d'étapes

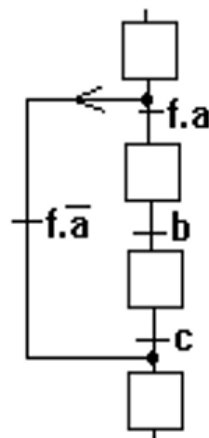


Figure 17 : Saut d'étape

7. Boucle répéter jusqu'à

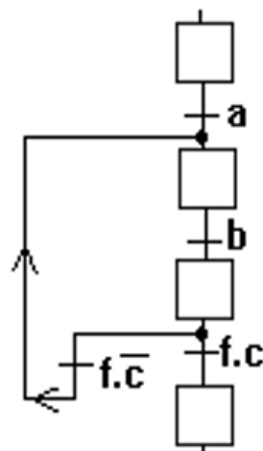


Figure 18 : Boucle répéter jusqu'à

Répéter des étapes jusqu'à ce que la condition $f.c$ soit vraie

V. Différents types d'actions associées aux étapes

On notera X_i la variable binaire associée à l'étape n°i, Donc nous avons $X_i=1$ si étape i est active.

1. Action continue

On effectue A pendant toute la durée de l'activité de l'étape 1

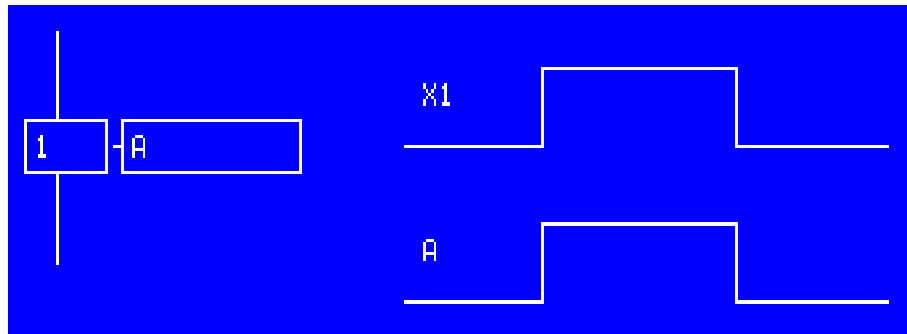


Figure 19 : Action continue

2. Action conditionnelle

On effectue A si l'étape 1 est active ET si $b=1$

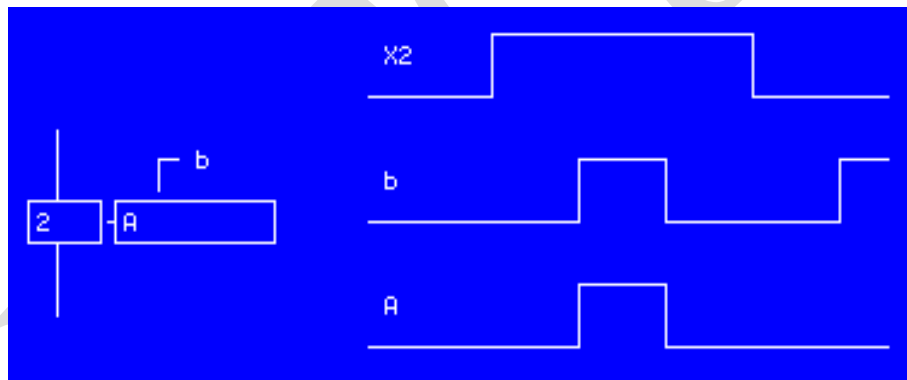


Figure 20 : Action conditionnelle



Figure 21 : Action conditionnée (autre écriture)

3. Action à durée limitée

L'action est exécutée tant que la temporisation n'est pas terminée. L'action M ne dure que 3 secondes à partir du début de l'étape X7



Figure 22 : Action à durée limitée (2 représentation normalisées)

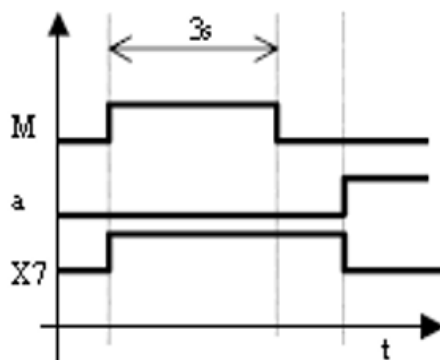


Figure 23 : Chronogramme d'une action à durée limitée (a : réceptivité suivante)

4. Action à durée retardée

L'action n'est exécutée que si le délai est écoulé. L'action M ne débute que 3 secondes à partir du début de l'étape X7



Figure 24 : Action à durée retardée (2 représentations normalisées)

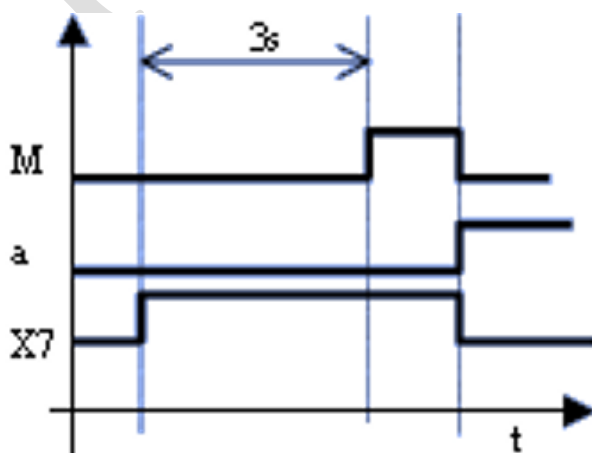


Figure 25 : Logigramme d'une action à durée retardée (a : réceptivité suivante)

5. Action mémorisée

Elle commence avec le début de l'étape 4 et se termine avec le début de l'étape 7

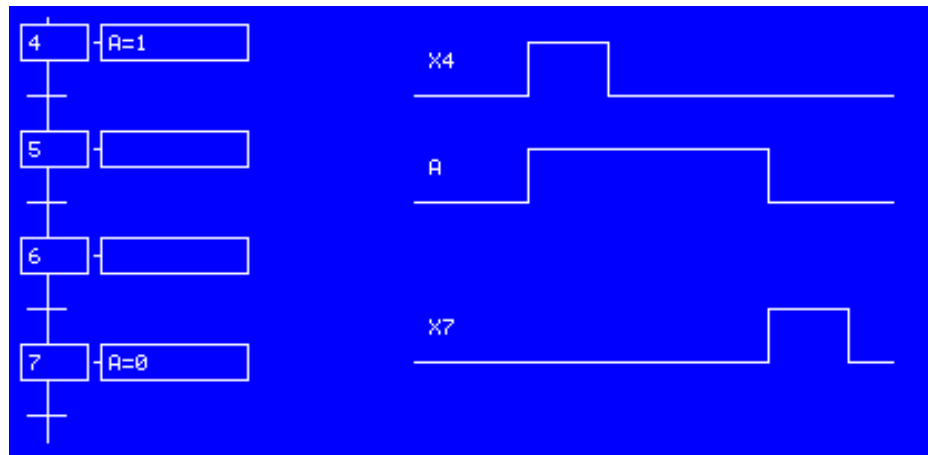


Figure 26 : Grafcet avec actions mémorisées et chronogramme

VI. Différents types de réceptivités associées aux transitions

1. Franchissement sur niveau

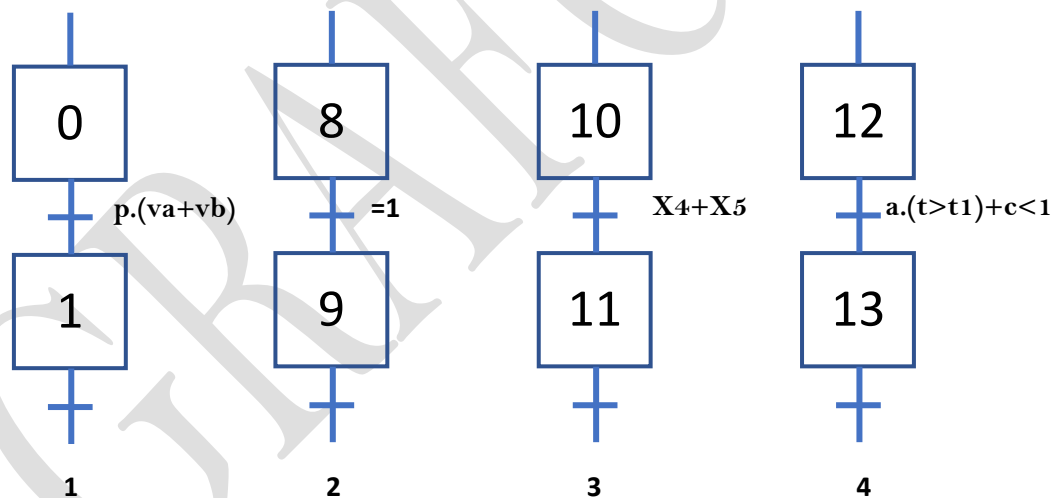


Figure 27 : **1-** Transition franchie si l'étape 0 est active ET $p.(va+vb)=1$; **2-** Transition toujours franchie si l'étape 8 est active ; **3-** Transition franchie si étape 10 ET (l'étape 4 OU étape 5) sont actives ; **4-** Transition franchie Si l'étape 12 est active ET $[a.(t>t1)+C<10] = 1$

2. Franchissement sur front

On appelle front montant de la variable binaire **a**, la variable, notée $\uparrow a$, qui prend la valeur 1 à l'instant du passage de 0 à 1 de la variable **a**.

On appelle front descendant de la variable binaire **a**, la variable, notée $\downarrow a$, qui prend la valeur 1 à l'instant du passage de 1 à 0 de la variable **a**.

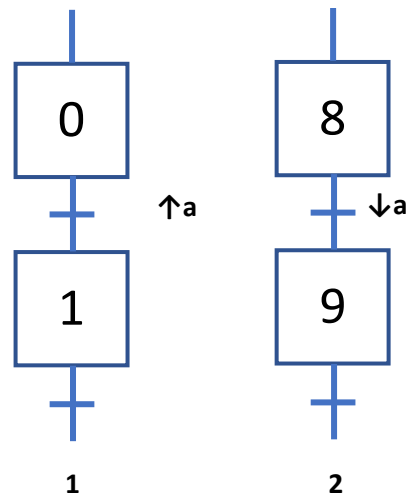


Figure 28 : 1- Transition franchie si l'étape 0 est active ET la détection du front montant de **a**; 2- Transition franchie si l'étape 8 est active ET détection du front descendant de **a** ;

VII. Comptage

Il est souvent nécessaire de compter un nombre de cycles ou de pièces, d'évènements dans un GRAFCET.

Un cycle de comptage comprend en général :

- Une initialisation de la variable de comptage.
- Une incrémentation (ou décrémentation) de cette variable
- Des réceptivités qui testent la valeur de la variable de comptage.

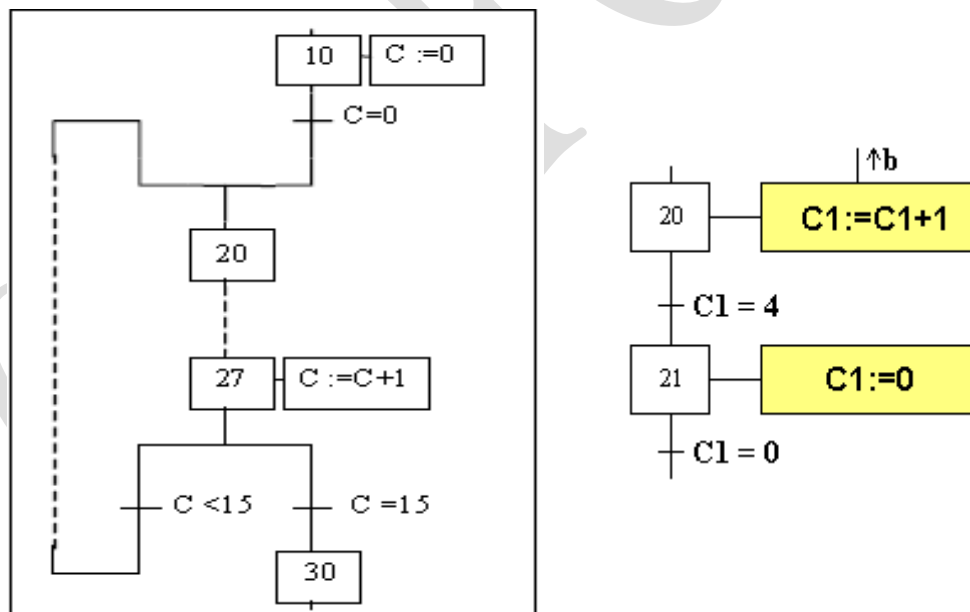


Figure 29 : Cycle de comptage par deux méthodes différentes

VIII. Temporisation

La prise en compte du temps peut aussi être réalisée dans la réceptivité. La temporisation est lancée dès l'activation de l'étape X7, elle n'est effective qu'au bout du temps $T=3$ secondes. La réceptivité étant vraie, la transition est franchie.

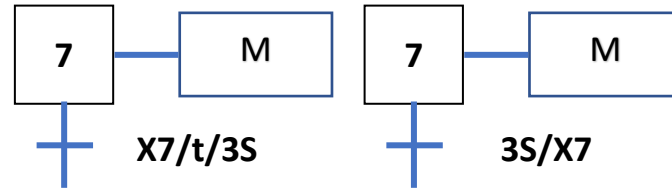


Figure 30 : Temporisateur dans la réceptivité (deux méthodes différentes)

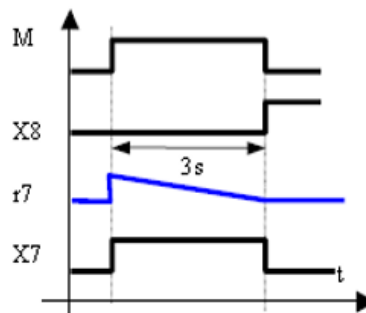


Figure 31 : Chronogramme d'un temporisateur

IX. Exemple : poinçonneuse avec GRAFCET niveau 2

Dans ce niveau on doit tenir compte des spécifications technologiques des capteurs et actionneurs. Pour cet exemple, supposons que les mouvements de descente et de montée sont obtenus soit par **un vérin pneumatique double effet**, soit par un vérin pneumatique simple effet et que les informations « haute » et « bas » sont obtenues au moyen de fins de course pneumatiques, et que l'information **départ** est donnée par un bouton poussoir pneumatique

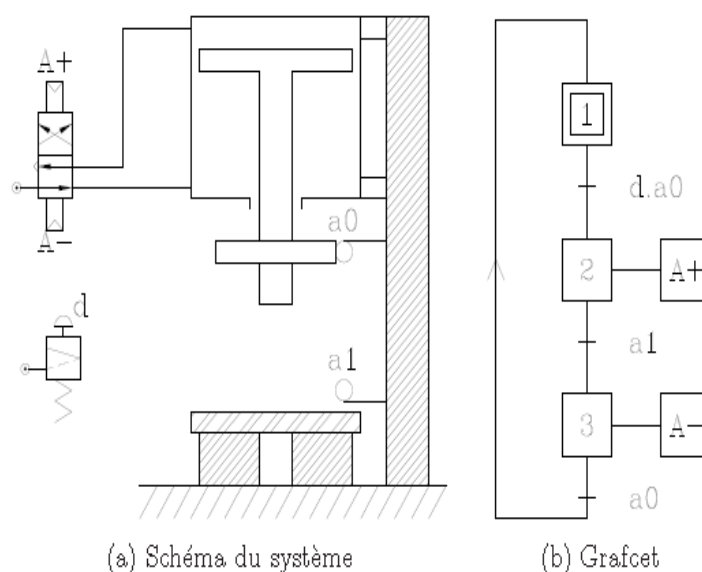


Figure 32 : GRAFCET poinçonneuse avec vérin double effet

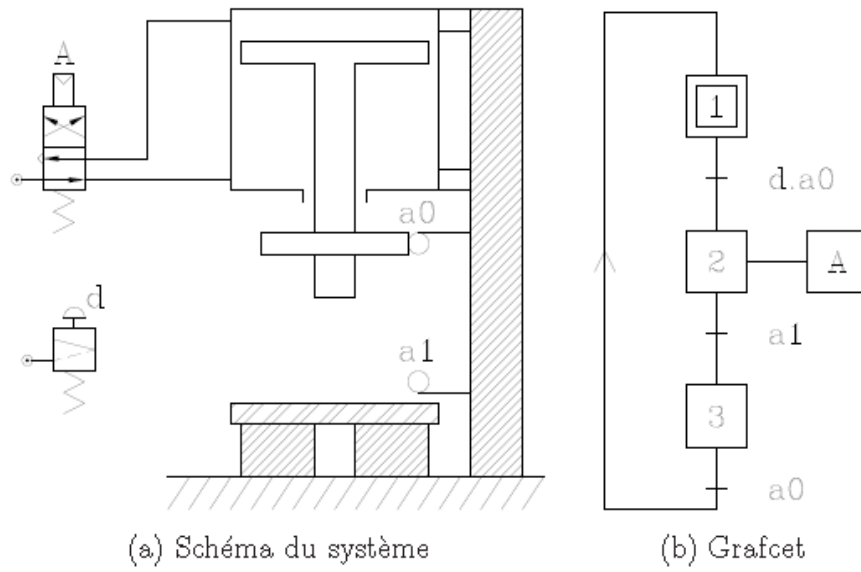


Figure 33 : GRAFCET poinçonneuse avec vérin simple effet

CHAPITRE II : GRAFCET avancé

I. Introduction

1. Enoncé

Le chariot fait des navettes entre un point A et un point B. On veut le commander grâce à deux boutons : Marche et Arrêt. C'est à dire que le chariot commence à effectuer ses cycles une fois le bouton Marche aura été appuyé, il s'arrêtera une fois son cycle est terminé et si l'on a appuyé sur le bouton Arrêt.

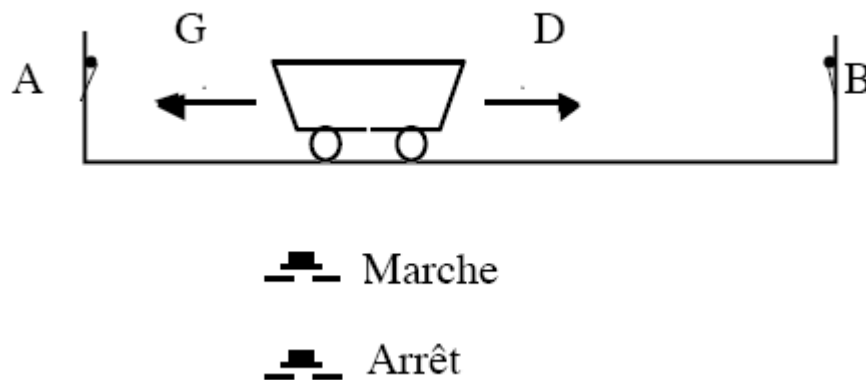


Figure 34 : Chariot en mouvement Gauche-Droit

2. Solution proposée

Cette solution a un inconvénient : pour arrêter le chariot il faut appuyer sur Arrêt au moment où le chariot arrive en A. Ce n'est pas satisfaisant. L'idée est d'avoir un GRAFCET qui lit les boutons et qui commande un autre GRAFCET. On parle d'un GRAFCET superviseur

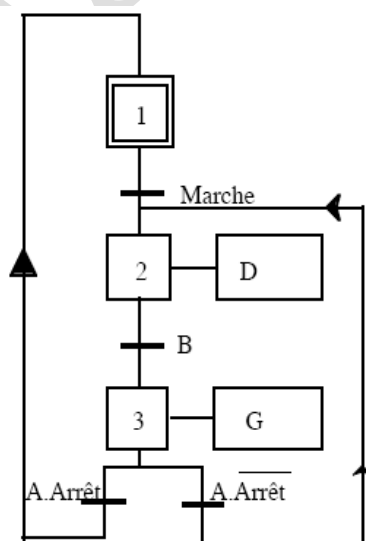


Figure 35 : GRAFCET proposé

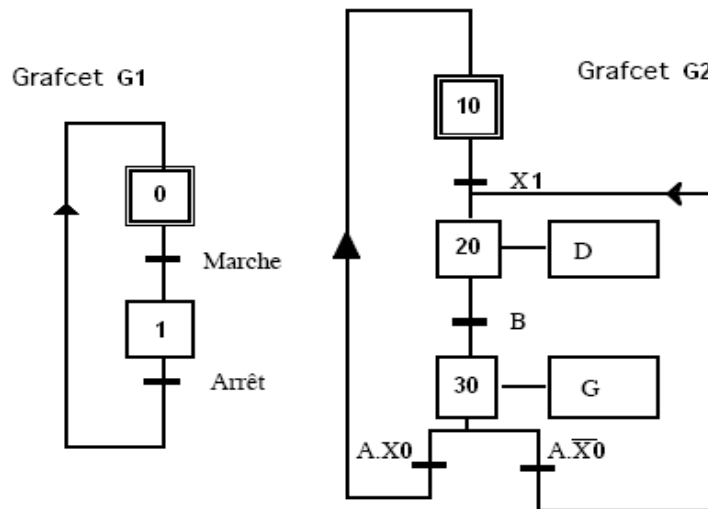


Figure 36 : GRAFCET de conduite (maître) ; GRAFCET de fonctionnement normal (Esclave)

Dans ce cas le GRAFCET maître (superviseur) s'occupe du démarrage, de la séquence, de la synchronisation et de l'arrêt de différentes tâches. Chaque tâche est décrite par un GRAFCET commandé par le superviseur, on parle ici de la notion de structures hiérarchisées.

Une analyse structurée d'un système permet de décrire celui-ci par un ensemble de GRAFCETs qui correspondent aux différentes fonctions du système.

Parmi ces principaux GRAFCETs on trouve :

- **GRAFCET de Surveillance (GS)** : (sécurité) : Il décrit l'ensemble des procédures de sécurité (arrêt d'urgence, mise en route).
- **GRAFCET de Conduite (GC)** : Il décrit la procédure du mode Marche (cycle, auto, manuel) / Arrêt.
- **GRAFCET de Maintenance (GM)** : Il décrit la procédure de réglage de la partie PO.
- **GRAFCET de Production Normale (GPN ou GFN)** : Il décrit le fonctionnement normal du système.

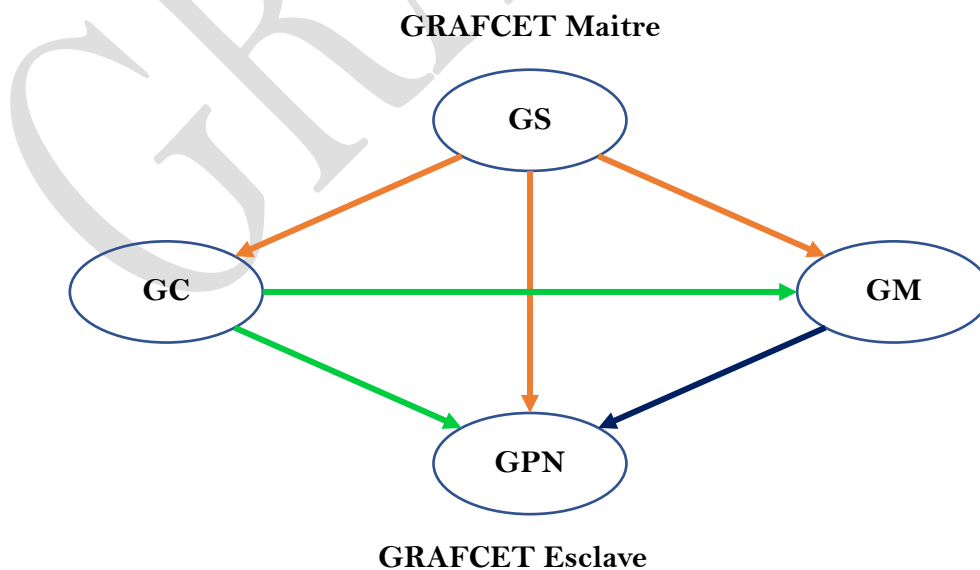


Figure 37 : Structure Hiérarchique des GRAFCETs

II. Macro - représentations.

Une macro-représentation consiste à remplacer une partie d'un GRAFCET par une représentation beaucoup plus condensée sous la forme :

- D'une macro-étape ;
- D'une tâche ;

3. Macro-étape

Une macro-étape est une représentation unique d'un ensemble d'étapes et de transitions appelée **expansion de la macro-étape**.

Remarque : Aucune action ne doit être associée à une macro-étape.

- L'expansion a toujours une étape d'entrée (E) et une étape de sortie (S) ;
- L'étape d'entrée est active dès le franchissement de la transition amont de la macro-étape ;
- L'activation de l'étape de sortie de l'expansion déclenche le franchissement de la transition aval de la macro-étape ;

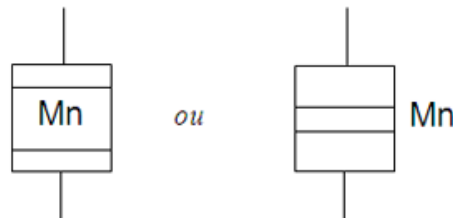


Figure 38 : Représentation normalisée d'une macro-étape

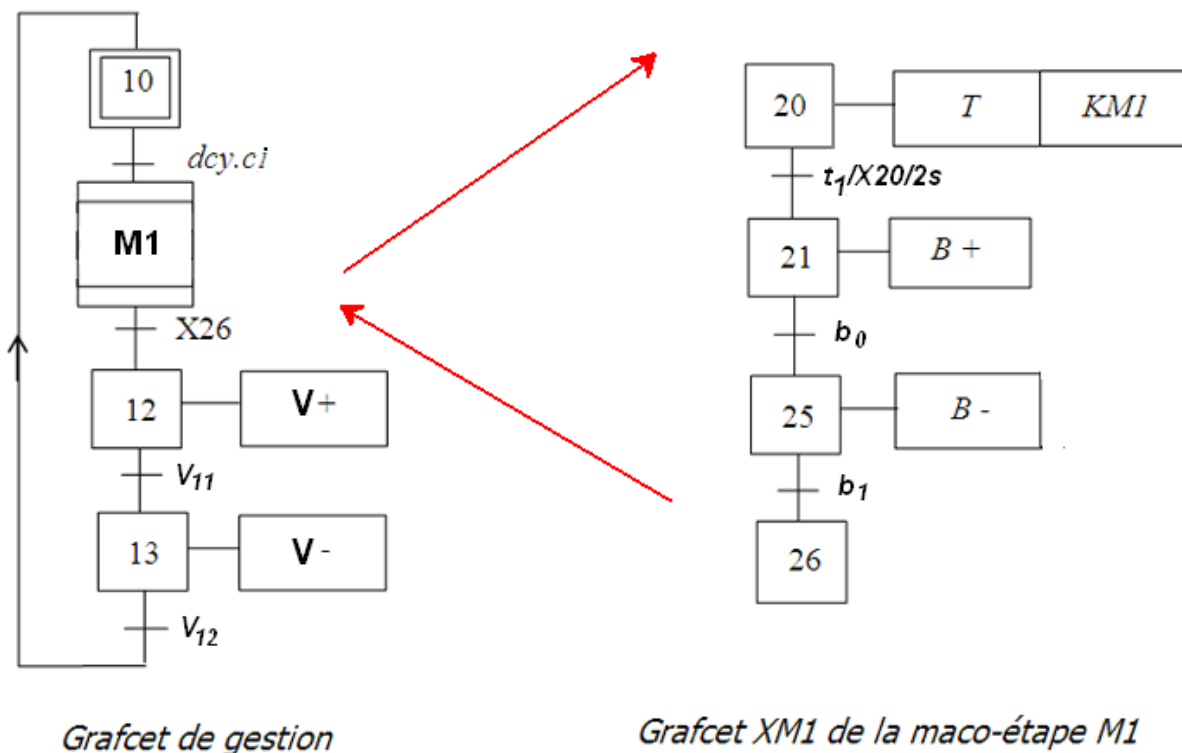


Figure 39 : 1- Expansion d'une Macro-étape

4. Structure d'un GRAFCET de tâche

Les tâches ont pour but de simplifier et de faciliter la description de systèmes complexes en allégeant le graphisme d'un GRAFCET et en détaillant séparément certaines parties.

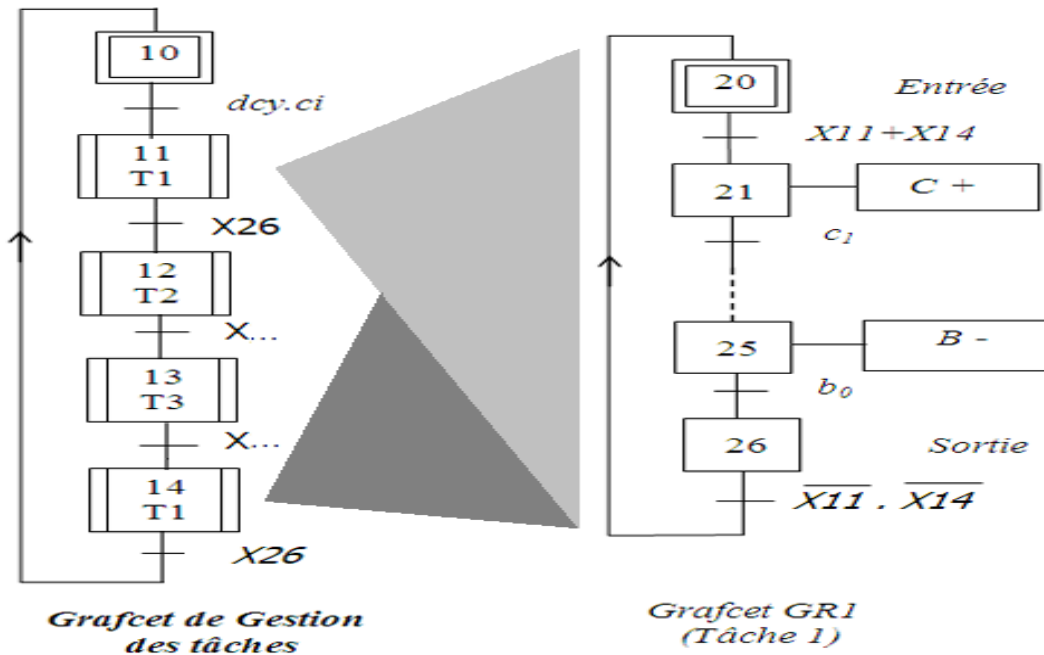


Figure 40 : GRAFCET des tâches

III. Structure d'un Sous-GRAFCET

Dans les automatismes séquentiels, il est fréquent de rencontrer des séquences répétitives dans le même cycle. Une séquence répétitive peut être représentée par un sous-GRAFCET ou un GRAFCET sous-programme. Cette notion s'est empruntée au langage informatique.

Un GRAFCET sous-programme est écrit sous la forme d'un GRAFCET indépendant, connecté au GRAFCET principal.

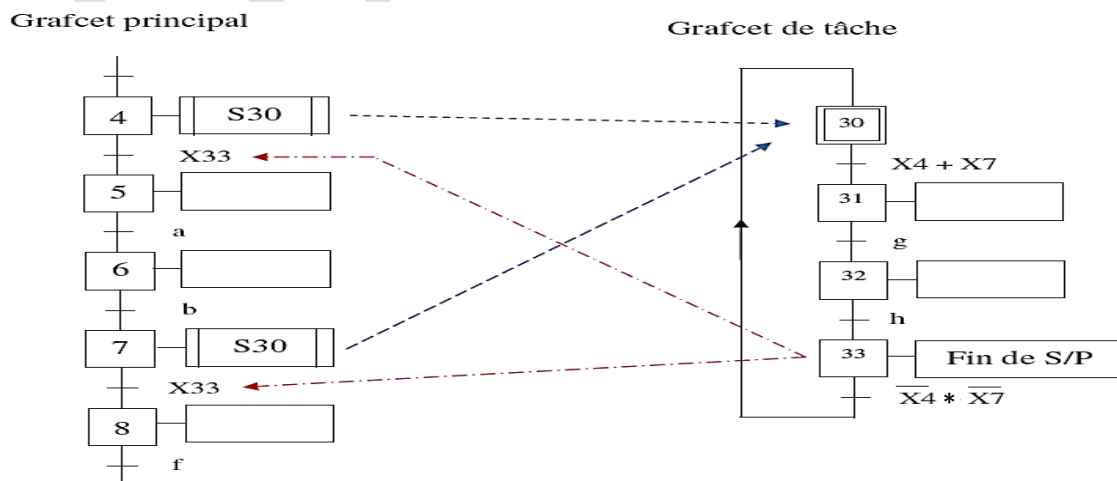


Figure 41 : GRAFCET sous-programme

IV. Etape encapsulante :

- Une étape encapsulante contient un ensemble d'étapes encapsulées sous forme de graphes partiels.
- Un graphe encapsulé peut lui-même contenir une étape encapsulante.
- Une étape encapsulante peut être initiale
- La désactivation d'une étape encapsulante provoque la désactivation de toutes les étapes de ses graphes encapsulés.

L'astérisque placé à côté de l'étape 11 de l'encapsulation indique l'étape qui sera activée au même instant que l'étape encapsulante 1

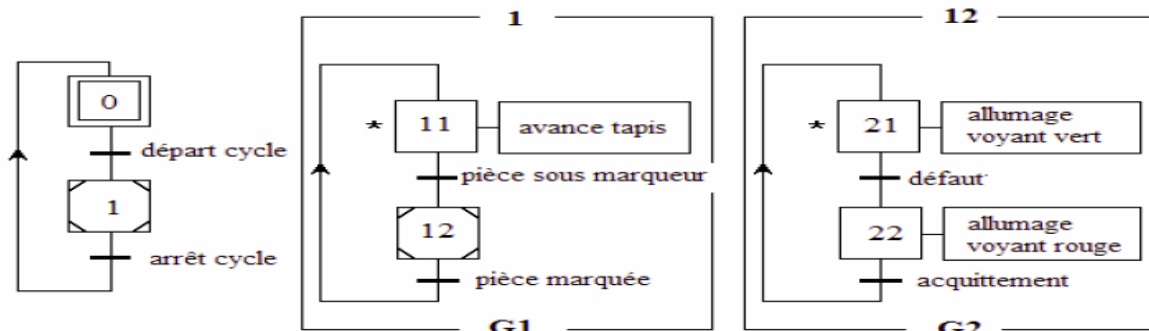


Figure 42 : Encapsulation des étapes

V. Forçage

- Le forçage est le passage imposé de la situation courante du GRAFCET à une situation déterminée et différente de celui-ci.
- Le GRAFCET forcé évolue alors sans franchissement de transition.
- Les GRAFCETs forcés sont maintenus dans la situation de forçage tant que l'ordre de forçage est valide.
- A tout instant le GRAFCET ne peut être forcé que dans une situation et une seule à partir d'un et d'un seul GRAFCET supérieur.
- La situation de forçage est prioritaire à toute activité du GRAFCET.

Le forçage est une action continue, notée dans un double cadre, qui agit sur le graphe Gi noté



On peut également forcer un GRAFCET :

- Soit en situation non vide
- Soit en situation vide

1. Forçage en situation non vide :

L'ordre de forçage provoque simultanément l'activation des étapes correspondantes à la situation imposée et la désactivation des autres étapes du GRAFCET forcé.

Exemple 1 :

- L'activation de l'étape 10 du GRAFCET G1 force le GRAFCET G2 .
- L'étape 21 est activée (forcée à 1), les autres étapes du GRAFCET G2 sont désactivées (forcées à 0).

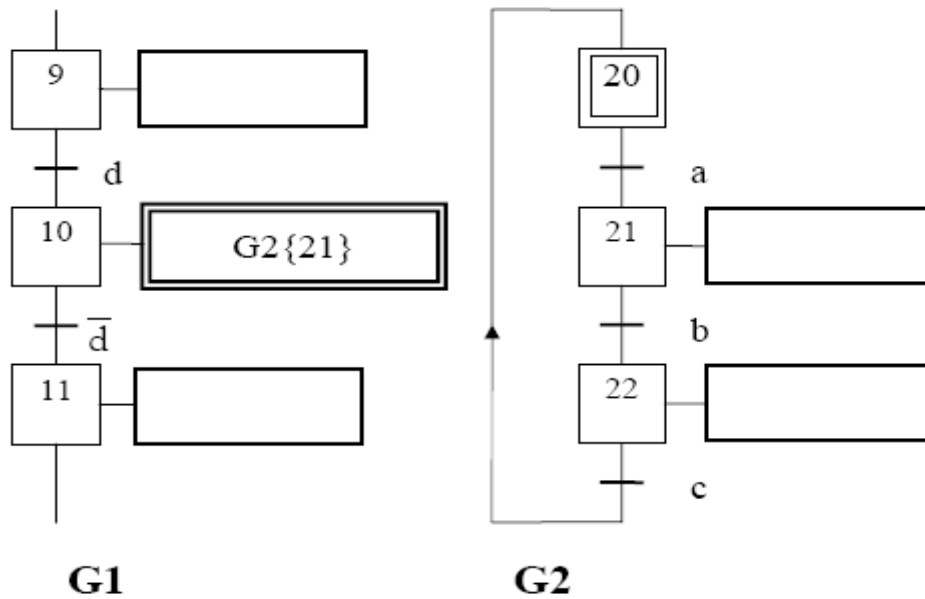


Figure 43 : Forçage à l'étape 21 du GRAFCET G2 par G1

Exemple 2 :

A l'activation de l'étape 5, toutes les étapes 20, 23, 24, du GRAFCET 2 sont désactivées et les étapes 21, 22 sont activées. A la désactivation de 5, il reprend son évolution normale.

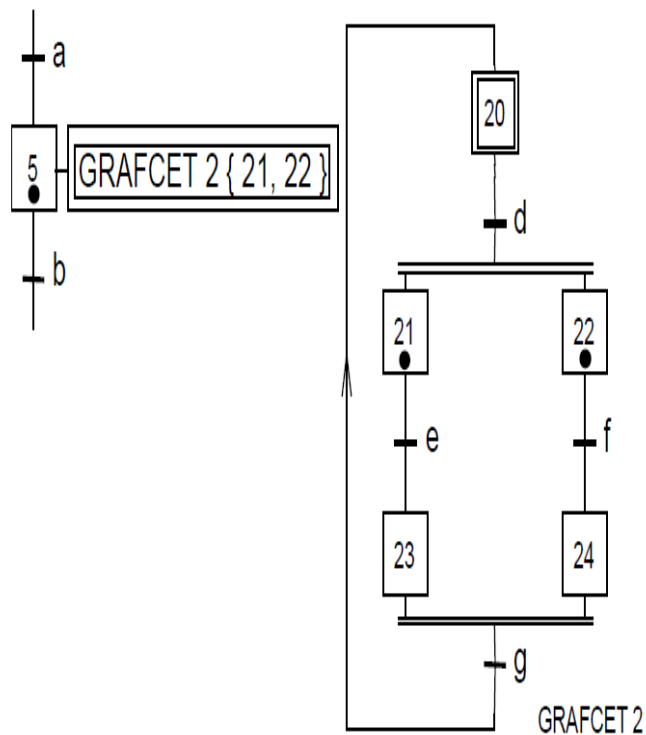


Figure 44 : Forçage à l'étape 21 et 22 du GRAFCET 2

Exemple 3 :

A l'activation de l'étape 5, l'étape initiale 20 du GRAFCET 2 est activée et toutes les autres étapes sont désactivées. A la désactivation de 5, il reprend son évolution normale.

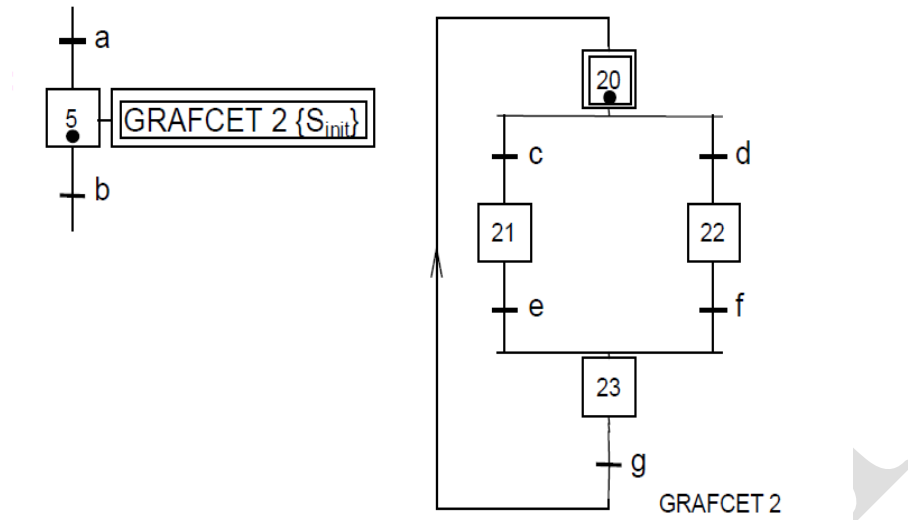


Figure 45 : Forçage à l'étape initiale du GRAFCET 2

2. Forçage en situation vide

Provoque simultanément la désactivation de toutes les étapes du GRAFCET forcé y compris les étapes initiales.

Le GRAFCET figé voit son évolution stoppée dès l'apparition de l'ordre de figeage.

Exemple :

A l'activation de l'étape 5, toutes les étapes du GRAFCET 2 sont désactivées. Le forçage à vide nécessite par la suite le forçage dans une situation non vide pour pouvoir redémarrer.

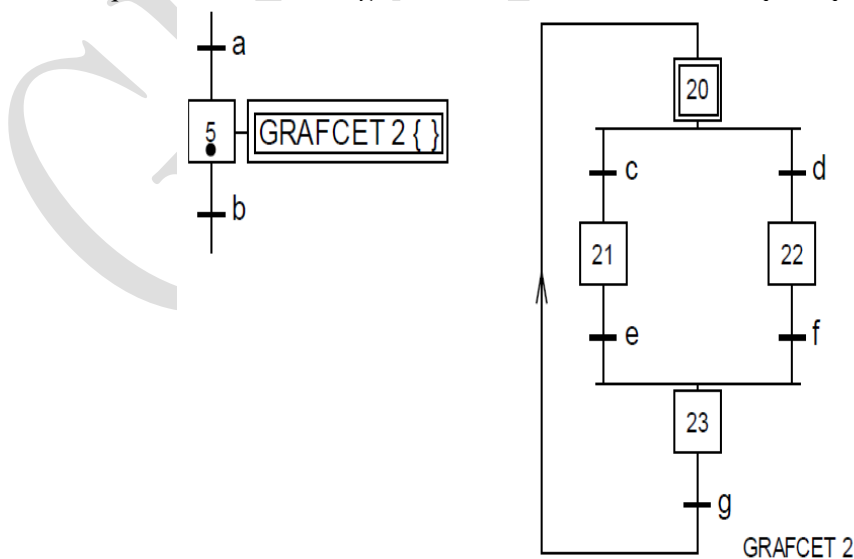


Figure 46 : Forçage en situation vide du GRAFCET 2

3. Figeage dans une situation courante

C'est un cas particulier du forçage, Il s'agit de maintenir le forçage dans la situation courante c'est à dire bloquer (gel) l'évolution du GRAFCET.

Le GRAFCET figé voit son évolution stoppée dès l'apparition de l'ordre de figeage. L'ordre de figeage est toujours vrai indépendamment des réceptivités. Il est souvent nécessaire de bloquer simultanément les sorties.

Exemple :

A l'activation de l'étape 5, le GRAFCET est figé dans la situation de cet instant : si 22 et 23 étaient actives au moment du forçage, elles le restent, jusqu'à désactivation de l'étape 5.

Le GRAFCET reprend alors son évolution normale.

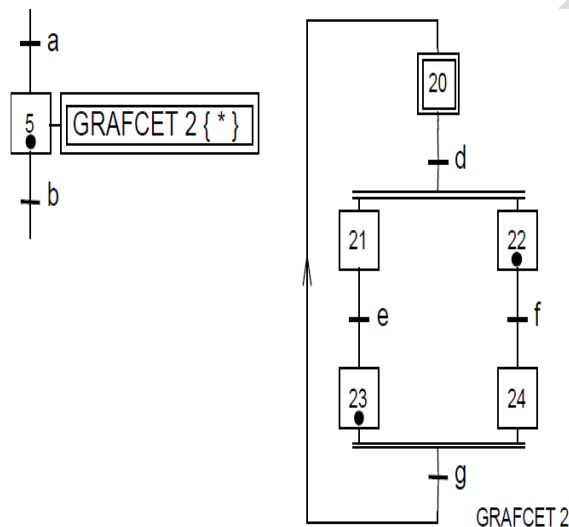


Figure 47 : Forçage en situation courante du GRAFCET 2

Exemple 1 : Chariot

Soit un petit wagonnet mis en mouvement par un moteur électrique commandé par un contacteur KM1 (à droite) et KM2 (à gauche).

- Le départ vers la droite se fait par un bouton poussoir S3.
- Le retour vers la gauche se fait automatiquement par S1.
- L'arrêt à gauche se fait automatiquement par S2.
- L'arrêt d'urgence se fait le bouton S4.

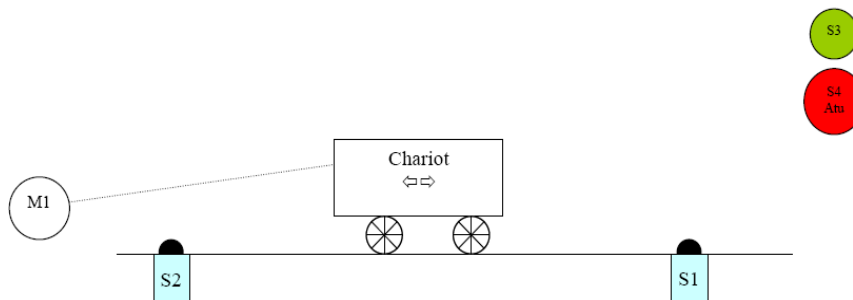


Figure 48 : Chariot

a. Gestion de Sécurité

Les arrêts d'urgence ou de dysfonctionnement qui peuvent survenir à tout instant, quelle que soit l'étape active, l'installation doit être stoppée et remis à l'étape initiale.

Pour cela on rajoute un autre GRAFCET, avec l'un prioritaire sur l'autre. C'est un GRAFCET hiérarchisé.

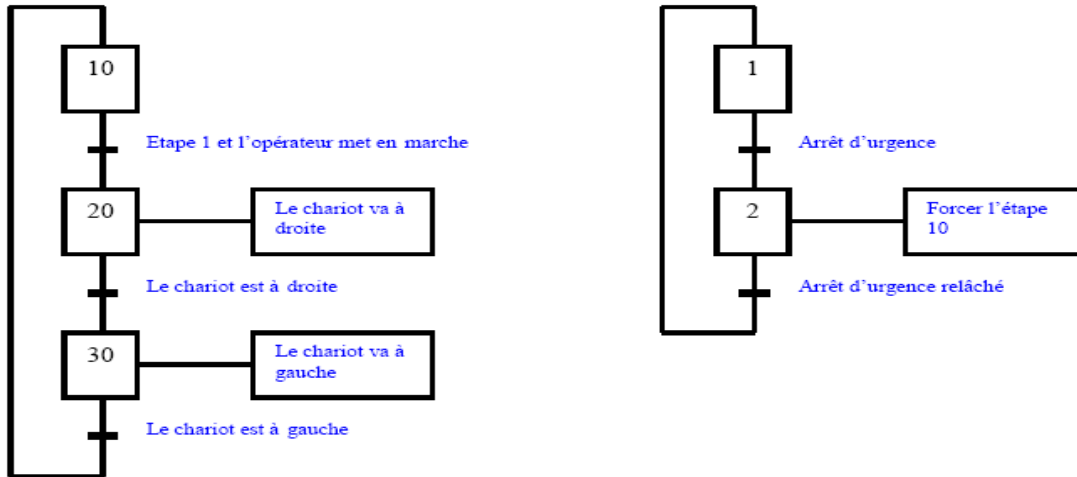


Figure 49 : GRAFCET hiérarchique : Niveau 1

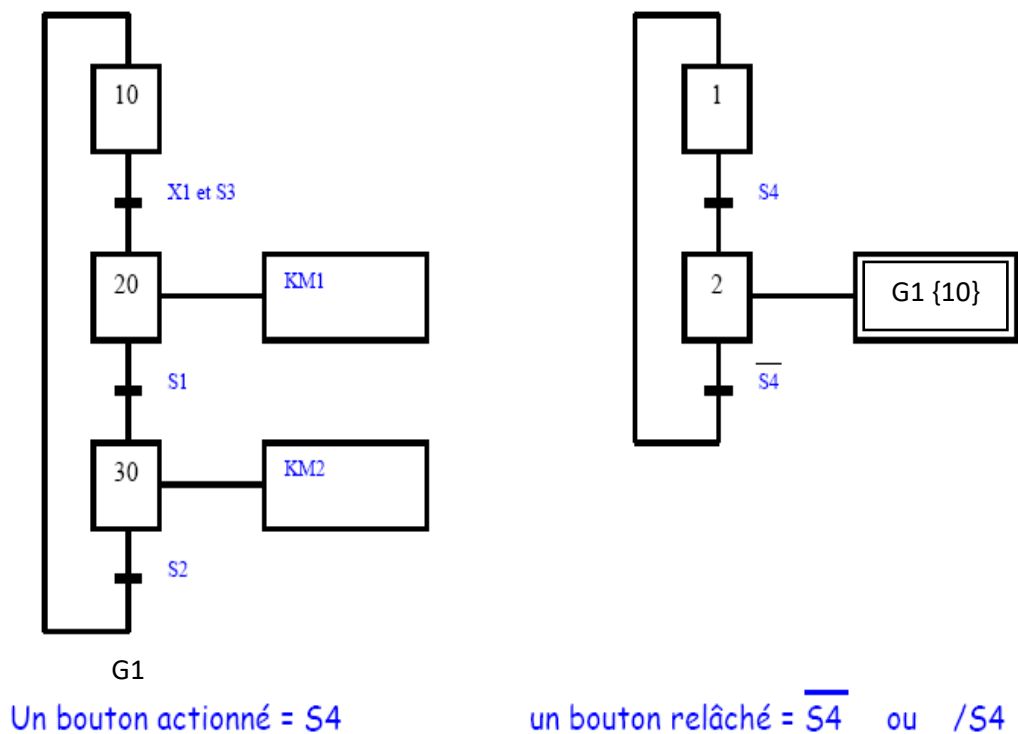


Figure 50 : GRAFCET hiérarchique : Niveau 2

Exemple 2 :

Le pupitre de commande est équipé de :

- D'un sélecteur à trois positions permettant de choisir le mode de marche : Auto. Semi (Semi-auto) / Manu (Pas à pas = Etape par étape).
- D'un bouton de mise en marche M et d'un autre d'arrêt A connectés aux entrées d'une mémoire délivrant une information V : (V=1 : marche, V=0 : arrêt).
- D'un bouton de départ de cycle Dcy.
- D'un bouton arrêt d'urgence AU et d'un bouton de réarmement REA.

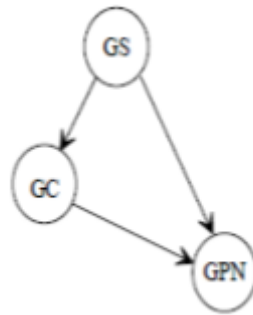


Figure 51 : GRAFCET hiérarchique :

a. Conditions initiales :

A la mise sous tension :

GS force GC à l'étape 100 et GPN à l'étape 1. Si AU n'est pas enclenché, et si l'on appuie sur REA, GS passe à l'étape 11. Donc GC et GPN peuvent évoluer.

L'opérateur choisit son mode de marche et appuie sur M=> V=1

Si AUTO :

101 est activée (si CI est vraie) donc le GPN déroule son cycle indéfiniment jusqu'à la disparition de V(A) ou l'apparition d'un autre mode de marche.

Si Semi :

102 est activée. Le départ du cycle GPN est conditionnée à l'appui sur DCY. Un seul cycle s'effectue. Il faut réappuyer sur Dcy pour un nouveau départ cycle.

Si Manu :

Le GPN est forcé à l'étape 1 par le GC. Les commandes manuelles sont validées. L'opérateur peut faire ses réglages.

A n'importe quel moment, un appui sur AU (Arrêt d'urgence) fait passer GS à l'étape 10. Donc GC est forcé à l'étape 100 et GPN à l'étape 1. Le redémarrage ne peut avoir lieu que si AU est désenclenché et si l'on appuie sur REA

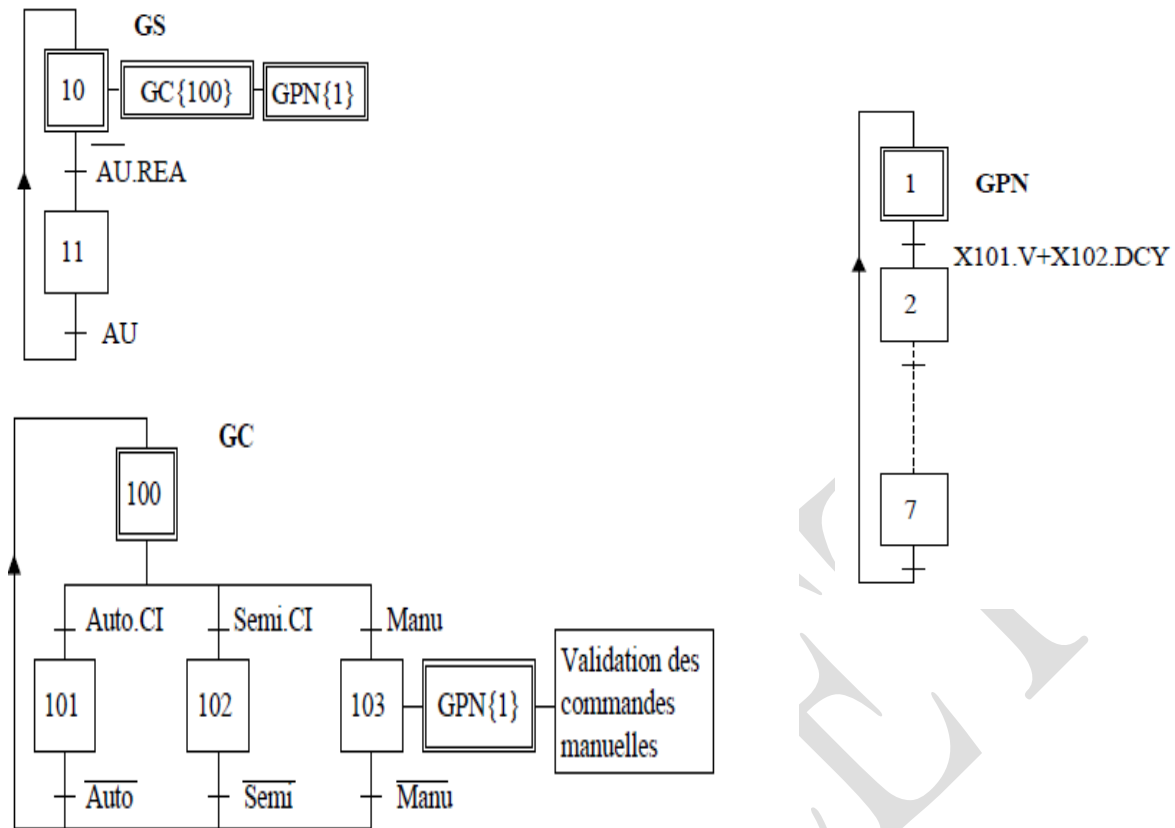


Figure 52 : GRAFCET hiérarchique :

Exemple 3 : Commande d'une perceuse

b. Fonctionnement normal :

- En position repos, le foret est en position haute A.
- Le démarrage s'effectue par appui et relâchement d'un bouton noté MA.

Dès le relâchement de MA, le foret doit entrer en rotation et simultanément le cycle de perçage doit commencer comme suit :

- Descente rapide de A en B.
- Descente lente de B en C, (perçage).
- Montée du foret de C en B, (débourrage).
- Descente rapide de B en C.
- Descente lente de C en D, (perçage).
- Montée du foret de D en A.

Arrivé en A, arrêt de la montée et de la rotation ; un nouveau cycle ne pourra redémarrer que sur commande de MA.

c. Gestion de l'arrêt d'urgence :

A l'apparition d'un signal AU, (arrêt d'urgence), arrêt de toutes les actions en cours.

Tant que AU = 1, aucune évolution n'est possible.

L'appui sur un bouton poussoir REP, (reprise), provoque le retour du foret en conditions initiales, (montée jusqu'en A).

Le préparant ainsi à recevoir une nouvelle commande de démarrage MA.

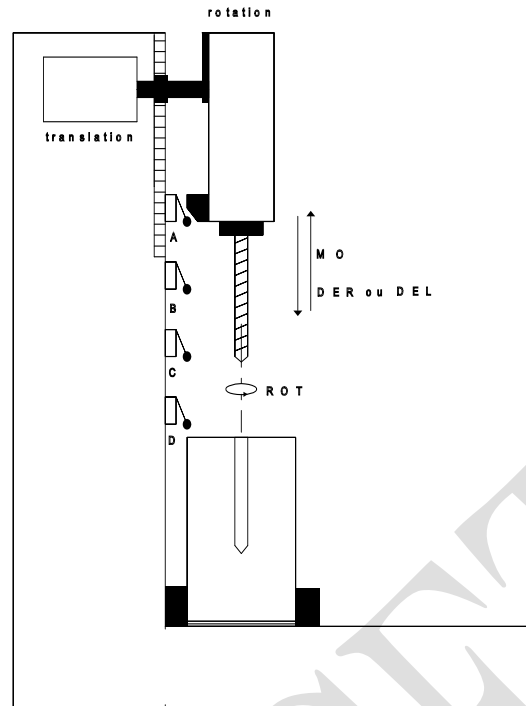


Figure 53 : Schéma d'une perceuse :

Chapitre II : Automate Programmable Industriel

I. Architecture interne d'un automate programmable industriel (API)

Un API est dispositif électronique capable d'exécuter des programmes en vue de la commande dans un environnement industriel. Son architecture interne est donnée par :

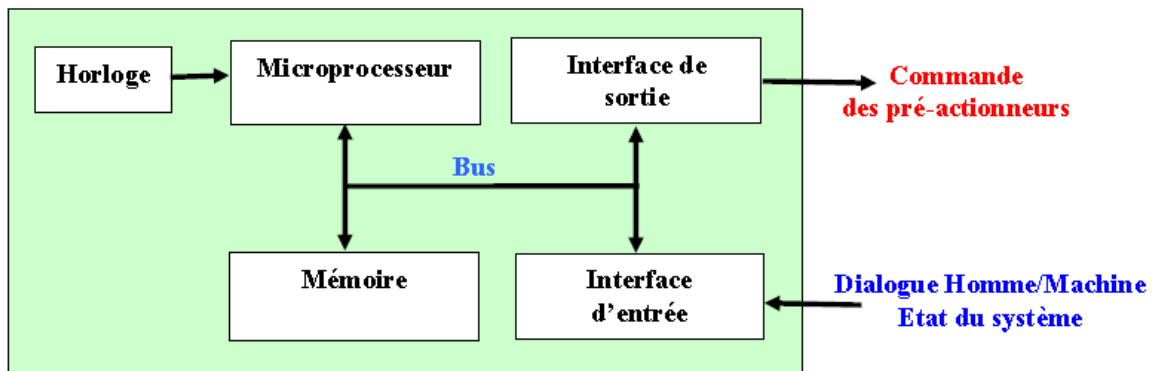


Figure 54 : 1- Architecture interne d'un API :

L'automate programmable reçoit les informations relatives à l'état du système par les capteurs et puis commande les pré-actionneurs suivant le programme inscrit dans sa mémoire.

Un API se compose donc de trois grandes parties :

- Le processeur ;
- La zone mémoire ;
- Les interfaces Entrées/Sorties

1. Processeur :

Le microprocesseur réalise toutes les fonctions logiques ET, OU, les fonctions de temporisation, de comptage, de calcul arithmétique à partir d'un programme contenu dans sa mémoire.

2. Zone Mémoire :

Elle est composée de deux zones :

- Zone mémoire de données : pour stocker les données temporaires
- Zone mémoire utilisateur pour stocker le programme

Exemple : Mémoire vive RAM, mémoire morte effaçable EEPROM

3. Les interfaces d'entrées/sorties

Les entrées reçoivent des informations en provenance des éléments de détection (capteurs) et du pupitre opérateur (BP). (Soit de type Tout ou Rien ou analogique)

Les sorties transmettent des informations aux pré-actionneurs (relais, électrovannes ...) et aux éléments de signalisation (voyants) du pupitre. (Soit de type Tout Ou Rien ou analogique)

II. Automate TSX Micro de chez Télémécanique

Les versions de Télémécanique :

- Automates Modicon Micro. TSX 3705/ 3708/ 3710/ 3720.
- Automates Premium : TSX P57 2623/2823M

Son synoptique est donné par le schéma suivant :

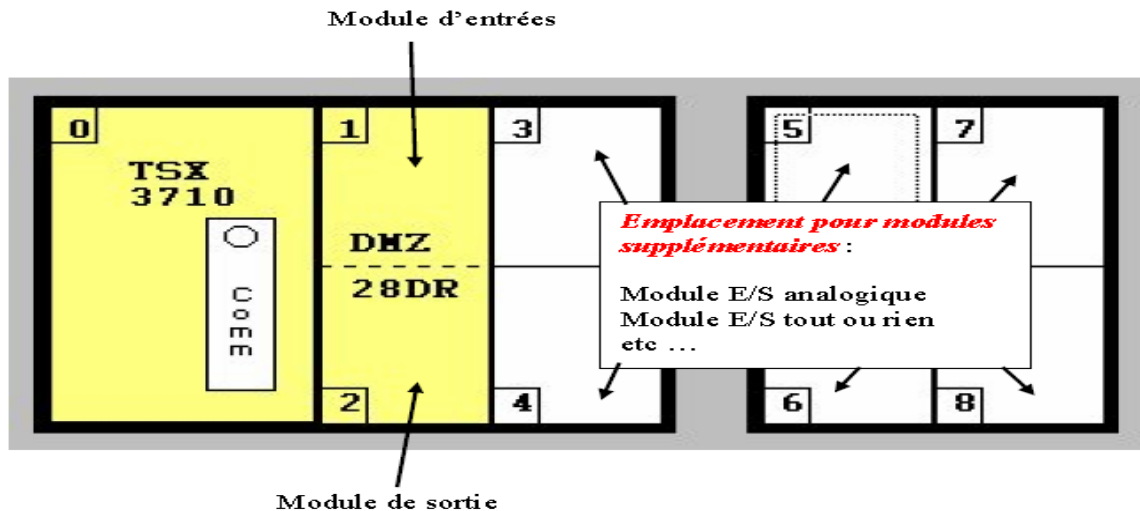


Figure 55 : Synoptique d'un API TSX Micro 3710 :

Il est composé par une unité centrale et des modules d'entrée/sortie.

1. Le Module d'entrée type TOR :

Chaque module est noté par %I i.j avec i : numéro de module d'entrée et j : numéro d'entrée au module i.

Exemple : %I1.2 présente l'entrée n°2 du module d'entrée n°1

%I3.77.....3

2. Module de sortie de type TOR :

Chaque module de sortie est noté par : %Q i.j avec i numéro de module de sortie et j : numéro de sortie sur le module de sortie i

Exemple : %Q2.8 : sortie n°8 du module de sortie n°2

III. Langage de programmation du logiciel PL7

Le logiciel PL7 intègre plusieurs langages de programmation permettant d'implanter le GRAFCET sur les automates Télémécanique.

Parmi ces langages on cite :

- Langage à contacts ou à relais (LADDER : LD)
- Langage Liste d'Instructions ou langage Booléen « machine » (IL)
- Langage littéral Structuré (ST) type « informatique ».
- Langage « GRAFCET » ou Interface GRAFCET.

Dans ce cours on va s'intéresser au langage à contacts et langage « GRAFCET »

C'est une expression graphique constituée par un ensemble de contacts (entrées) et de bobines fictifs (sorties) qui sont reliés entre eux par des connexions horizontales et verticales.

- Il se compose d'une suite de « réseaux » exécutés en séquence par l'automate. C'est l'ordre de saisie des réseaux qui est pris en compte par le système lors de la scrutation.
- Chaque réseau porte une étiquette unique (en option) qui à la syntaxe suivante : %Li (i,0 à 999).
- L'ordre des étiquettes est quelconque.
- Chaque réseau est composé par une zone de test et une zone d'action
- Chaque réseau est composé de 16 lignes maximums et 11 colonnes (pour automates Premium), 11 colonnes et 8 lignes pour Automate Micro.
- Les éléments de base de ce langage sont les contacts, les bobines, les variables internes, les blocs de fonctions, etc.
- La zone test, dans laquelle figurent les conditions nécessaires à une action (contact d'entrée, bit interne, etc.)
- La zone action, qui applique le résultat consécutif à un enchaînement de test (bobines de sortie, blocs de fonction, etc.)
- Il scrute la zone de test du haut vers le bas et du gauche vers la droite.
- Puis il exécute la zone d'action simultanée.

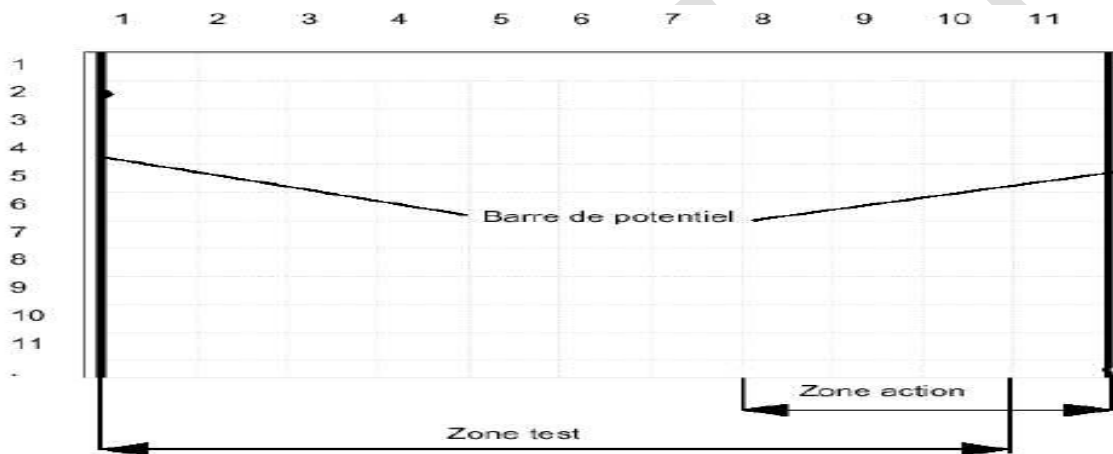


Figure 56 : Zone de tests et zone d'actions

1. Zone de test (variable binaire) : Les contacts

Désignation	Graphisme	Fonctions
Contact à fermeture		Contact passant quand l'objet bit qui le pilote est à l'état 1.
Contact à ouverture		Contact passant quand l'objet bit qui le pilote est à l'état 0.
Contact à détection de front montant		Front montant : détection du passage de 0 à 1 de l'objet bit qui le pilote.
Contact à détection de front descendant		Front descendant : détection du passage de 1 à 0 de l'objet bit qui le pilote.

Figure 57 : Eléments de test sur les variables binaires

2. Zone d'action : Les bobines

Langage à contacts	Liste d'instructions	Littéral structuré	Description
	ST	:=	aux bobines directes: l'objet bit associé prend la valeur du résultat de l'équation.
	STN	:=NOT	aux bobines inverses: l'objet bit associé prend la valeur de l'inverse du résultat de l'équation.
	S	SET	aux bobines à enclenchement: l'objet bit associé est mis à 1 lorsque le résultat de l'équation est à 1.
	R	RESET	aux bobines à déclenchement: l'objet bit associé est mis à 0 lorsque le résultat de l'équation est à 1.

Tableau 58 Elément d'action sur les variables binaires

3. Eléments de liaison

Les éléments graphiques de liaison permettent de relier les éléments graphiques de test et d'action.




Désignation	Graphisme	Fonctions
Connexion horizontale		Permet de relier en série les éléments graphiques de test et d'action entre les deux barres de potentiel.
Connexion verticale de potentiel		Permet de relier en parallèle les éléments graphiques de test et d'action.
Dérivation court-circuit		Permet de relier 2 objets au travers de plusieurs connexions.

Tableau 59 : Eléments de liaison

4. Exemple de réseau en CHART

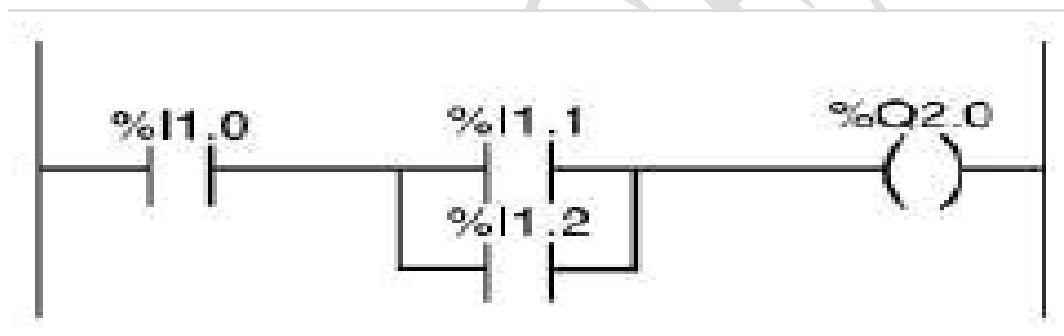


Figure 60 : Exemple de réseau CHART

5. Variables internes

On distingue les variables binaires, entières et en format flottant

a. Les variables binaires :

- Bits internes : %Mi (i, 0 à 255)
- Bits système : %Si (i, 0 à 127)
- Bits images d'activité d'étapes : %Xi (i, 0 à 1023)

Remarque : En général le nombre des variables dépend de type d'automate utilisé (TSX Micro, Premium etc.).

b. Les variables entières :

- Variable entière signée codée sur 16 bits : %MWi (i, 0 à 511)
- Variable entière signée codée sur 32 bits : %MDi (i, 0 à 511)
- Variable octet codée sur 8 bits (pour caractère) : %MBi (i, 0 à 511)

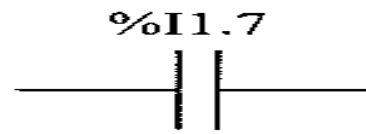
c. Autres

Les variables en format flottant en 32 bit: %MFi (i, 0 à 511).

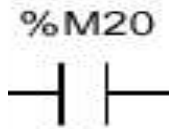
Les constantes codées sur 16 bits : %KMi (i, 0 à 512)



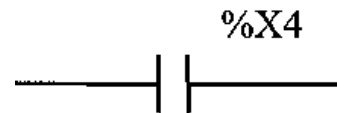
Bloc de comparaison
horizontal



Test d'une entrée %I1.7



Test d'un bit %M20



Test si l'étape X4 est
active



Bloc de comparaison
Temporisation de 2 secondes
Par défaut on multiplie la valeur
choisie par 0.1s

Figure 61 : Eléments de test des variables

IV. Bloc de fonction : Bloc d'opération

Il permet d'effectuer les opérations arithmétiques et logiques.

1. Opérations arithmétiques :

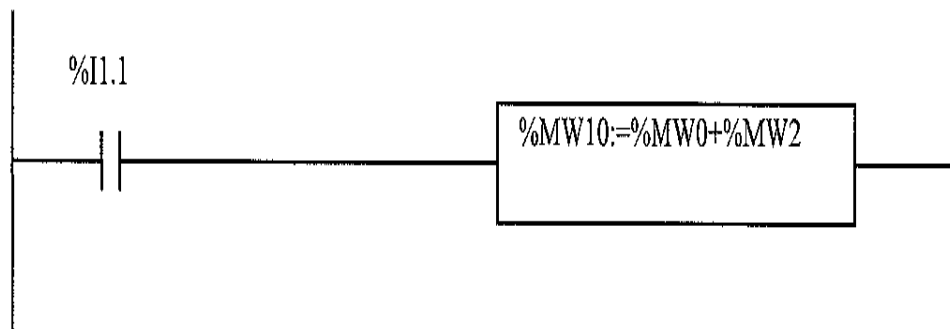


Figure 62 : Si % I1.1 = '1' alors, % MW10 vaudra %MW0+%MW2

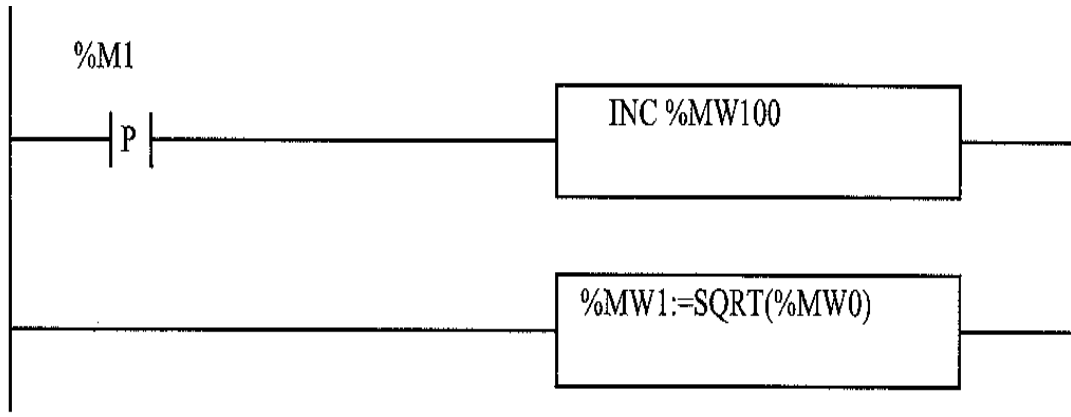


Figure 63 : Incrémenter %MW100 sur un front montant de %M1. 2-%MW1 est la racine carrée de %MW0

2. Opérations logiques

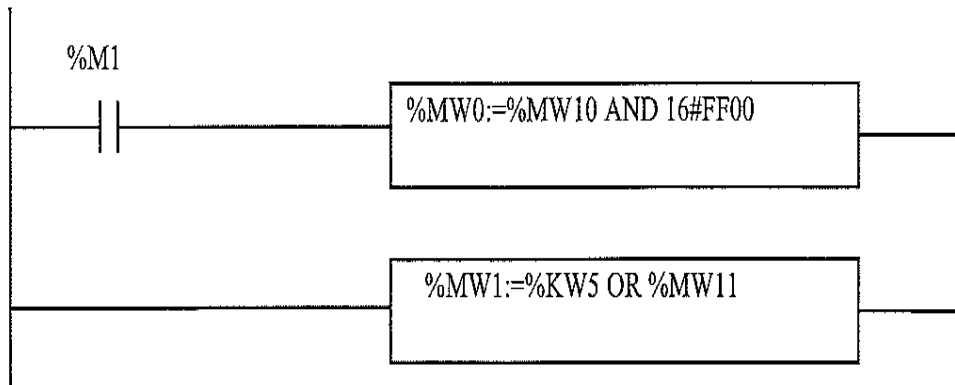


Figure 64 : Instructions utilisables : AND (ET bit à bit), OR (OU bit à bit), XOR (OU EXCLUSIF bit à bit), NOT (complément logique bit à bit)

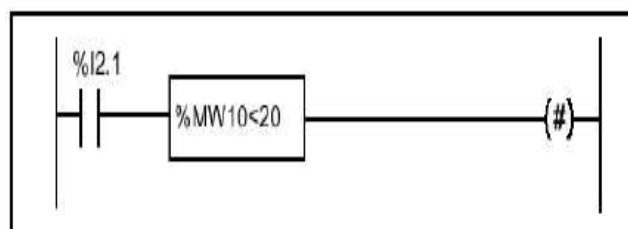


Figure 65 : Programmation d'une réceptivité en langage à contact

3. Réseau de contacts

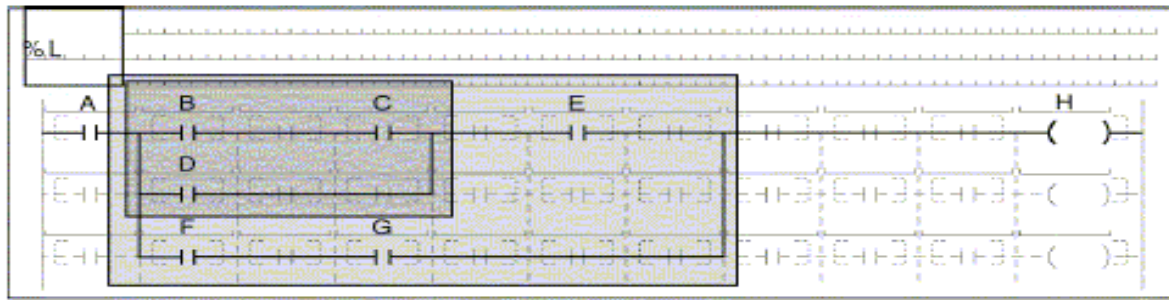


Figure 66 : Réseau de contact

Phase	Description
1	Evaluation du réseau jusqu'à rencontre de la 1 ^{ère} liaison verticale de convergence : Contact A, B, C
2	Evaluation du premier sous réseau : Contact D.
3	Poursuite de l'évaluation du réseau jusqu'à la rencontre de la deuxième liaison verticale de convergence : Contact E
4	Evaluation du 2 ^{ème} sous réseau : Contact F et G
5	Evaluation de la bobine H.

Tableau 1 : Phase d'évaluation d'un réseau de contact

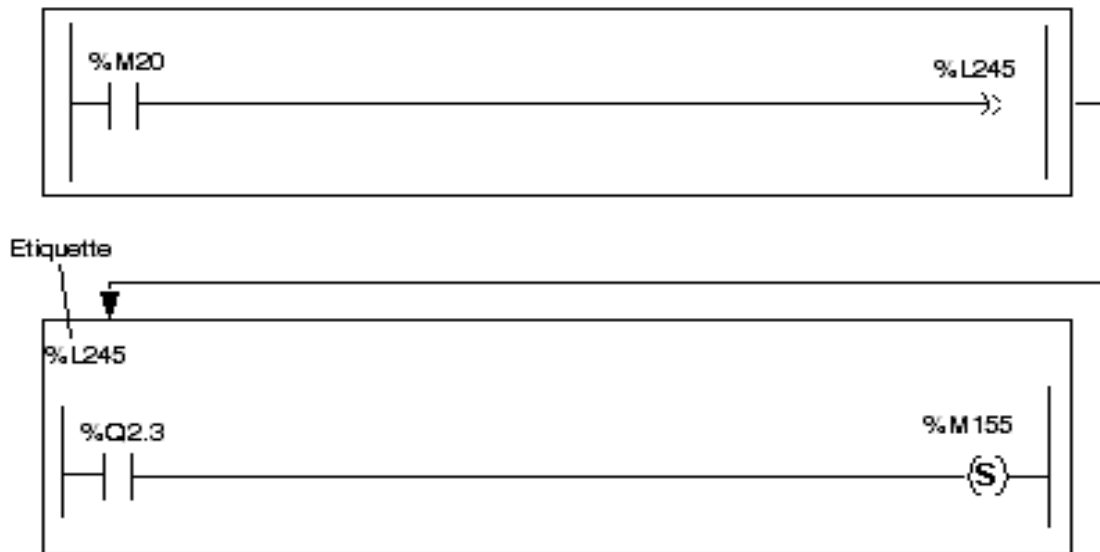


Figure 67 : Saut de programme vers un autre réseau en utilisant les étiquettes %Li du réseau

V. Bloc fonction temporisateur :

Temporisateur a 3 modes de fonctionnement :

- TON : permet de gérer des retards à l'enclenchement.
- TOF : permet de gérer des retards au déclenchement.
- TP : permet d'élaborer une impulsion de durée précise.

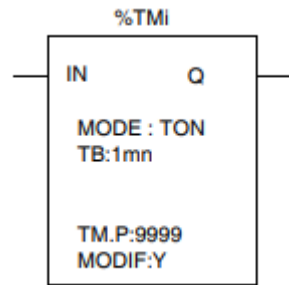


Figure 68 : Représentation graphique du bloc fonction temporisateur

Il est caractérisé par :

Caractéristiques	Repère	Valeur
Nom du temporisateur	%TMI	0 à 63 pour un TSX 37
Mode		TON, TOF, TP
Base de temps	TB	1 mn par défaut, 1s, 100 ms, 10 ms
Valeur courante	%TM.V	%TMI.V mot qui croît de 0 à %TMI.P
Valeur de présélection	%TM.P	%TMI.P mot qui peut être lu, testé et écrit par programme. Il est mis à la valeur 9999 par défaut. La durée ou le retard élaboré est %TMI.P x TB.
Entrée d'activation	IN	Sur front montant (mode TON ou TP) ou front descendant (mode TOF), démarre le temporisateur.
Sortie temporisateur	%TM.Q	Le bit de sortie %TMI.Q passe à l'état logique 1 dès que la valeur courante a atteint %TMI.P, puis reste à 1 tant que IN reste à 1.

Tableau 2 : Caractéristiques d'un temporisateur

1. Mode TON (Retard à l'enclenchement).

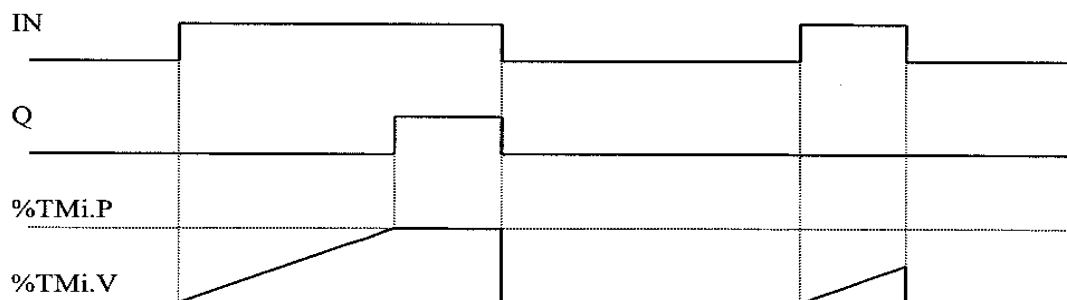


Figure 69 : Chronogramme d'un temporisateur en mode TON

2. Mode TOF (retard au déclenchement)

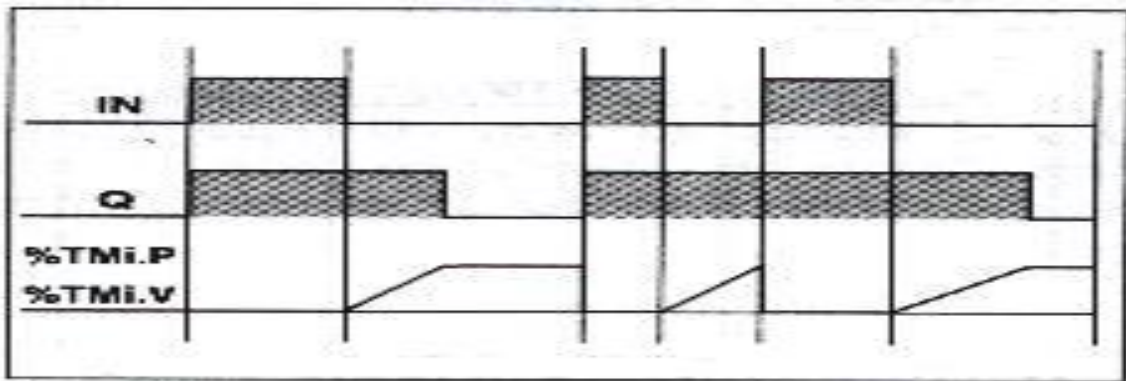


Figure 70 : Chronogramme d'un temporisateur en mode TOF

3. Mode TP (monostable)

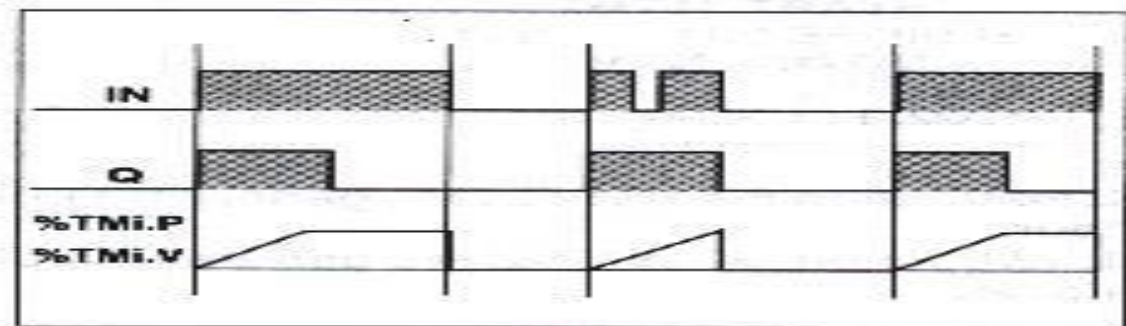


Figure 71 : Logigramme d'un temporisateur en mode TP

VI. Structure du programme

Le programme réalisé peut avoir une structure soit monotâche ou multitâche.

1. La structure monotâche :

Elle comporte seulement une tâche maître (MAST) exécutée cycliquement (mode par défaut) ou périodiquement (période fixé par l'utilisateur). Dans cette tâche, le programme comporte un traitement principal et des sous programmes (SR).

Le traitement principal est subdivisé en trois sections :

- Le traitement préliminaire (PRL) écrit en LD, IL ou ST. Il est exécuté avant la section GRAFCET.
- GRAFCET (CHART) dans laquelle elles sont programmées les réceptivités et les actions.
- Le traitement postérieur (POST) écrit en LD, IL ou ST. Dans lequel elles sont programmées les actions.

Le cycle de traitement exécuté par l'automate est : PRL, Chart et POST.

Les sous-programmes Sri (i de 0 à 254) se programment comme des entités séparées. Les appels aux sous-programmes s'effectuent dans PRL, dans POST, dans les actions associées aux étapes et même depuis d'autres sous-programmes (8 imbrications maximum).

L'automate exécute les opérations de la tâche MAST dans l'ordre suivant d'une façon cyclique :

- Traitement préliminaire (bit internes, test des entrées.)
- Traitement du programme CHART (GRAFCET)
- Traitement en postérieur (POST) (mise à jour des sorties)

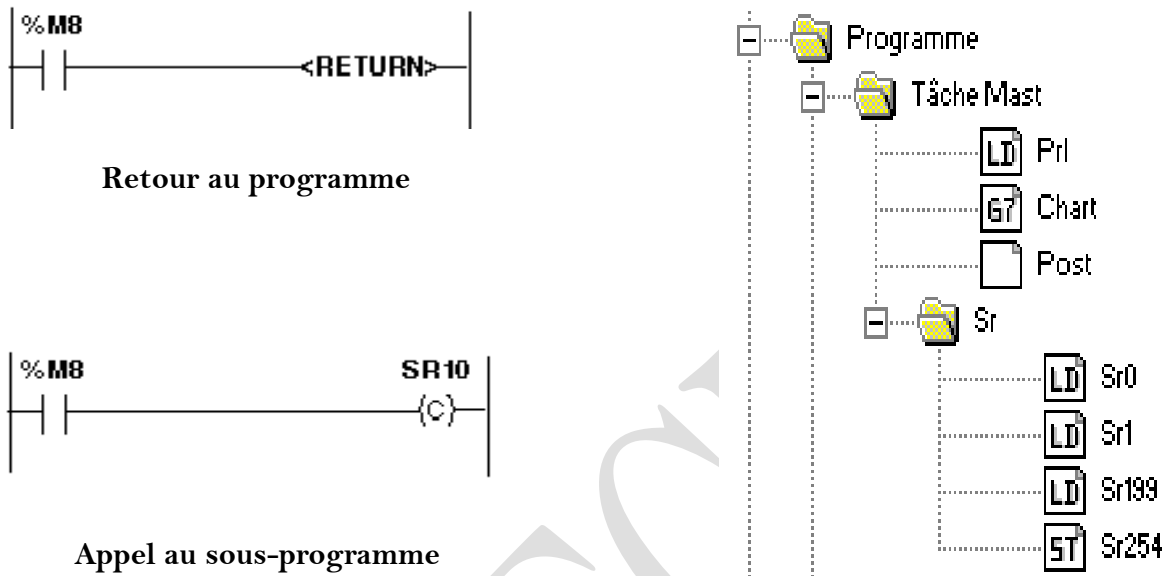


Figure 72 : Structure d'un programme monotâche

2. Structure multitâche :

Elle est utilisée dans un processus qui comporte des priorités d'exécution différentes. Elle comprend la tâche maître, et une ou plusieurs tâches événementielles (EVT)(leur exécution est déclenchée par un appel externe en provenance de certains modules d'entrée).

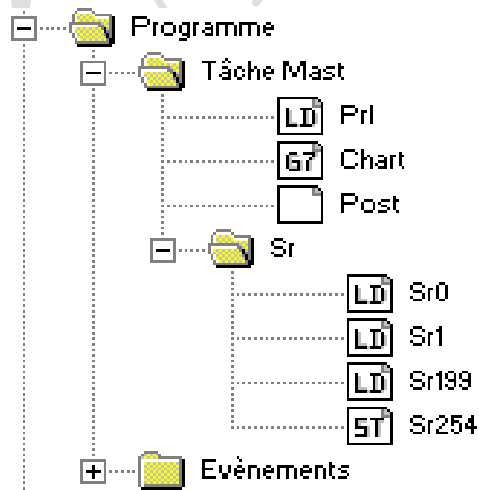


Figure 73 : Structure d'un programme

3. Langage « GRAFCET » ou interface IHM

Le GRAFCET se programme sur 8 pages (0 à 7). Chaque page peut comporter 96 étapes
Le nombre maximum de transitions est de 1024 dont au plus 24 simultanées.

La programmation s'effectue soit par le langage LD, IL ou le ST. Les réceptivités sont associées aux transitions et les actions sont associées aux étapes.

Bits système associées au GRAFCET

VII. Les bits systèmes du GRAFCET

1. Le bit %S21 (Initialisation du GRAFCET)

Ce bit système est normalement à l'état logique « 0 ». La mise à 1 de %S21 provoque la désactivation des étapes actives et l'activation des étapes initiales.

Ce bit est remis à l'état « 0 » automatiquement par le système au début du traitement séquentiel juste avant le traitement préliminaire. Le programme utilisateur utilise %S21 doit être donc obligatoirement placé dans le traitement préliminaire.



Figure 74 : Activation du bit S21

2. Bit %S22 (Reset du GRAFCET)

La mise à l'état 1 de %S22 provoque la désactivation des étapes actives de l'ensemble du traitement séquentiel. Ce bit est remis automatiquement par système à l'état logique « 0 » au début du traitement séquentiel. Il faut utiliser ce bit dans le traitement préliminaire PRL.

Ce bit sert au pré positionnement du GRAFCET lors du passage d'un fonctionnement normale en marche spécifique ou à l'apparition d'un incident

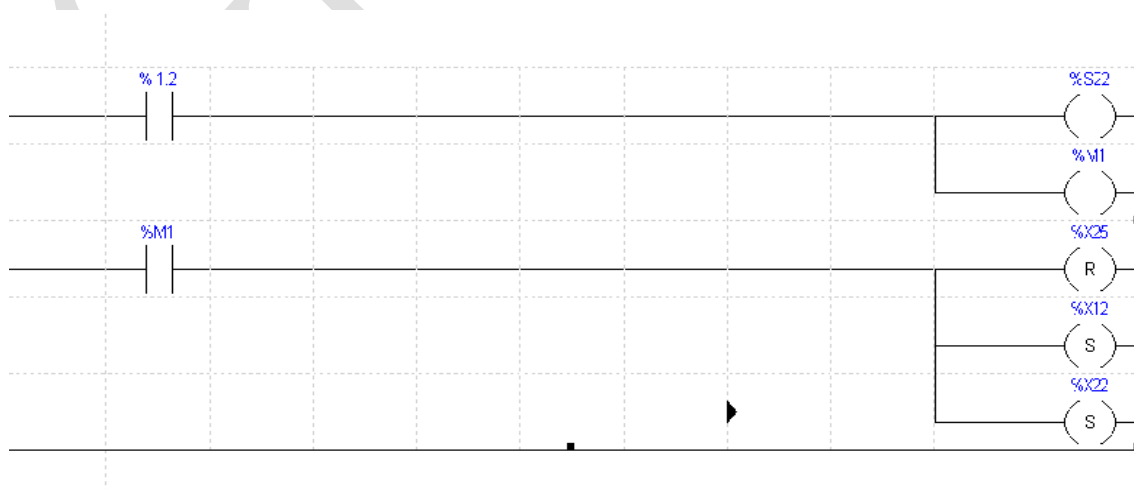


Figure 75 : Activation du bit S22

3. Bit %S23 : (Gel)

Le figeage du GRAFCET s'effectue par le bit système %S23. Normalement à l'état 0, la mise à l'état 1 de %S23 provoque le maintien en l'état des GRAFCET. Quelle que soit la valeur des réceptivités aval aux étapes actives, les GRAFCET n'évoluent pas. Le gel est maintenu tant que le bit %S23 est à 1. Ce bit n'est pas remis automatiquement à l'état « 0 » mais par programme.

Remarque

Ce bit ne peut arrêter l'exécution des actions qui sont associées à une étape figée active.

4. Le bit système %S9

Ce bit est remis à 0 au début du programme dans la partie PRL. Il peut être mis à 1 par programme dans la partie PRL pour provoquer le forçage des sorties de l'automate à l'état 0. Lorsqu'il est remis à 0, les sorties sont mises à jour normalement.