

République Algérienne Démocratique et Populaire  
Ministère de L'enseignement Supérieur et de la Recherche Scientifique  
Université 8 Mai 1945 - Guelma  
Faculté des Sciences et de la Technologie  
Département Electronique et Télécommunications

696



**Mémoire de fin d'étude  
Pour l'obtention du diplôme de Master Académique**

Domaine : **Sciences et Techniques**  
Filière : **Electronique**  
Spécialité : **Systemes Electroniques**

---

**Automatisation d'un processus industriel :  
Station de distribution et station de tri**

---

Présenté par :

-Siafa Nawel  
-Hafiane khroufa

Sous la direction de : Mr :Nemissi Mohamed

JUIN 2011



# Remerciement

*Au terme de ce travail nous tenons à remercier tout d'abord notre encadreur Mr. Nemissi mohamed ce qui nous a accordé l'honneur de diriger ce travail, sa compétence et ses conseils pertinents ont été pour nous un solide repère et réconfort dans tous les moments.*

*Nous exprimons également notre reconnaissance à qui ont accepté de participer à ce travail et d'avoir bien voulu contribuer à notre travail.*

*Je tenon à remercier tout les professeurs et les employés de département Génie électrique.*

*Et enfin mes remerciements à tous mes collègues d'études*

11/2702



# Dédicace



*Je dédie ce modeste travail fruits de mes année d'étude à :*

*Mon cher père et ma chère mère A mon époux Tahar qui nous aident et  
encouragé et sacrifié pour moi jusqu'au bous*

*Quoi que j'écrive je ne parviendrais jamais à exprimer ma gratitude et  
ma reconnaissance on vers eux qui ont été toujours derrière les plus  
grands événements de ma vie*

*Ma collègue Nawel*

*A mes frères A mes sœurs*

*A toute la famille.*

*Khroufa*



# Dédicace



*Je dédie ce modeste travail fruits de mes années d'étude à :*

*Mon cher père et ma chère mère A mon époux Mohamed qui nous  
aident et encouragé et sacrifié pour moi jusqu'au bous*

*Quoi que j'écrive je ne parviendrais jamais à exprimer ma gratitude et  
ma reconnaissance on vers eux qui ont été toujours derrières les plus  
grands événements de ma vie*

*Ma collègue Khroufa*

*A mes frères A mes sœurs*

*A toute la famille.*

*Nawel*



# Sommaire

Résumé

Introduction général

## Chapitre I : Automatismes et automate programmable industriel

I-L'Automatisme .....	10
1-Introduction .....	10
2-Objectifs de l'automatisation .....	10
3-Définition d'un automatisme .....	11
5-Les capteurs .....	12
5.1-définition .....	12
5.2-Constitution d'un capteur .....	13
5.3-Mode de fonctionnement des capteurs.....	13
5.4-principales caractéristiques des capteurs .....	13
5.4.1-Etendue de mesure .....	14
5.4.2-Sensibilité .....	14
5.4.3-Précision .....	14
5.4.4-Fidélité et justesse .....	14
5.4.5-rapidité .....	14
5.4.6-Stabilité .....	14
5.4.7-Respectabilité et Reproductibilité .....	14
5.4.8-Bruit de fond .....	15
6-Types des capteurs .....	15
6.1-Capteur capacitif.....	15
6.2 -Capteur inductif .....	15
6.3-Capteur optique .....	15
II-Les automates programmables industriels .....	16
1-Introduction .....	16
2-Définition d'un automate programmable .....	16
3- Principe de fonctionnement .....	16
4- Constitution .....	17
5-Architecture des automates .....	19
5-1-Aspect extérieur .....	19
5-2-Structure interne .....	20

5-3-Fonctions réalisées .....	21
5-4-Principales fonctions .....	21
6-Traitement du programme automate .....	22
7-- Langages d'automates .....	23

## **Chapitre II : Grafcet**

1-Introduction .....	25
2-Définition .....	25
3- Types de représentation du Grafcet .....	25
4-Les éléments de bases .....	25
4.1 - Etape .....	26
4.2 - Actions associées aux étapes.....	26
4.2.1- Classification des actions associées aux étapes .....	26
1- Ordres continus .....	27
2- Ordres conditionnels .....	27
3- Ordre retardé .....	28
4- Ordre de durée limitée.....	28
5- Ordre fugitif .....	29
6- Ordre maintenu sur plusieurs étapes .....	29
7- Action mémorisée par une fonction opérative auxiliaire .....	30
4.3-Les étapes initiales .....	31
4.5 - Liaisons orientées.....	32
5-Règles d'évolution d'un GRAFCET.....	32
5.1 - Règle 1 : Initialisation.....	32
5.2 - Règle 2 : Franchissement d'une transition.....	32
5.3 - Règle 3 : Evolution des étapes actives .....	33
6 - GRAFCET à séquences multiples.....	33
7-Liaison de deux Grafquets.....	34
8-Synchronisation de deux Grafquets.....	36
8.1 - Principe .....	36
8.2 - Méthode .....	36

## **Chapitre III : STEP 7**

1-Introduction.....	39
2-Définition du STEP7.....	39
3-Fonctionnalités.....	40

4-Caractéristiques techniques.....	40
5-Fonctions du logiciel de base.....	41
6-Interaction du logiciel et du matériel.....	41
7-SIMATIC Manager.....	42
8- Utilisation de STEP 7.....	42
9-Les différents blocs.....	43
9.1-Bloc utilisateur.....	43
9.1.1-OB (bloc d'organisation).....	43
9.1.2-FB (bloc fonctionnel).....	43
9.1.3-FC (fonction).....	43
9.1.4-DB (bloc de données).....	43
9-2- Bloc système pour fonctions standard et fonctions système.....	44
9.2.1-SFB (bloc fonctionnel système).....	44
9.2.2-SFC (fonction système).....	44
9.2.3-SDR (données système).....	44
10-Programmation symbolique.....	44
10.1- Adresse absolue.....	45
10.2-Programmation symbolique.....	45
11-Les modes de programmation.....	45

#### **Chapitre IV : Application :station de distribution et station de tri**

1-Le système MPS (Système de Production Modulaire).....	48
2-Station de distribution.....	48
2.1- Fonction.....	48
2.2-Station de distribution contient.....	49
2.3-Objectifs de formation des projets étudiés.....	49
2.4- Les principaux composants.....	50
2.4.1-Le module de magasinage à empilage.....	50
2.4.2-Module de transfert.....	50
2.4.3-Régulateur de pression.....	51
2.4.4- Vacuostat.....	51
2.4.5-Pupitre de commande.....	52
2.5-Description de la séquence.....	52
2.6-Grafcet de systeme.....	53
3-Station de tri.....	56

3.1- Fonction .....	56
3.2- Thème central : Capteurs .....	56
3.3-Capteur inductif .....	56
3.4- Barrières photoélectrique reflex.....	57
3.5-Option : Traitement de valeurs analogiques .....	57
3.6-Objectifs de formation des projets étudiés.....	57
3.7.2-Module de levage.....	58
3.7.3- Module de mesure.....	59
3.7.4-Module de glissière pneumatique .....	60
3.7.5-Comparateur analogique .....	60
3.8-Graficet du système.....	61
Conclusion	



## Résumé

L'objectif de ce travail est l'étude de deux stations : station de distribution et station de tri qui font partie d'un ensemble des stations formant une simulation d'un mini usine d'assemblage des petits vérins. Dans ce travail nous réalisons un programme qui permet un fonctionnement manuel et automatique. Dans le premier chaque station fonctionne indépendamment et cycle par cycle. Dans le deuxième mode les deux stations travaillent ensemble d'une façon continue.

L'importance de l'étude de cette station réside essentiellement dans le fait qu'elle comporte de différents aspects industriels tels que la pneumatique et l'utilisation d'une multitude de capteurs.

# Introduction

# Introduction générale

Depuis les temps les plus anciens, l'homme a cherché un appui matériel pour effectuer ces tâches quotidiennes. En premier lieu, il eut recours à des simples machines, mais actuellement, et vu l'évolution technologique moderne, il rêve de concevoir des machines qui pourraient reproduire son intelligence. Des machines qui peuvent à la fois effectuer des tâches difficiles et pénibles, et en même temps, capables de fonctionner automatiquement. Ainsi, l'automatisation des procédés industriels est introduite pour permettre aux machines de réagir d'une façon autonome selon les excitations externes. L'automatisme est alors devenu un facteur important permettant d'assurer une bonne compétitivité et l'amélioration des automatismes constitue un axe de recherche très important.

Les automates programmables industriels (API) sont des mémoires programmables introduits essentiellement pour assurer les tâches d'automatisation. Ces appareils, largement répandus dans l'industrie, sont apparus aux États-Unis vers 1969 et en France vers 1971. Ils sont couramment utilisés dans des systèmes critiques (avions, ascenseurs, trains,...).

L'objectif de notre travail est l'étude de deux stations (distribution et tri) qui font partie d'un ensemble des stations formant une simulation d'une mini usine d'assemblage des petits véris. L'importance de l'étude de cette station réside essentiellement dans le fait qu'elle comporte de différents aspects industriels tels que la pneumatique et l'utilisation d'une multitude de capteurs

Le présent mémoire consiste en quatre chapitres :

Le premier chapitre est réservé à l'automatisme et les automates programmables industriels. Dans lequel, on présente la structure générale d'un automatisme, les différents éléments qui le constituent et les automates programmables.

Le deuxième chapitre introduit le Grafcet et son principe de fonctionnement.

Dans le troisième chapitre nous étudions deux stations (distribution et tri), nous évoquons leur description, leur fonctionnement avec détail, les Grafcets correspondant et ces programmes (en langages contact).

# Chapitre I :

## Automatisme et automate programmable industriel

## **I-L'Automatisme :**

### **1-Introduction :**

Un système automatique est un ensemble qui après avoir reçu des Instructions fournis par l'opérateur, décide et agit de lui même, se substituant ainsi à l'homme. Cette substitution conduit à une plus grande rapidité d'exécution et une meilleure régularité des Résultats. Tout système automatisé comprend deux parties technologiques et un opérateur humain. L'opérateur humain est la personne chargée, d'agir sur des matériels technologiques divers pour Fournir les instructions permettant d'élaborer la programmation et de rester en relation avec le Système lors de la mise au point au cours du fonctionnement pour intervenir en cas de perturbation.

La partie opérative a pour mission de faire subir à l'objet à traiter diverses transformations, elle Reçoit les commandes et est chargée d'enregistrer les instructions données par l'opérateur et donner Les ordres à la partie opérative afin de la faire fonctionner ; elle transmet à l'opérateur les Informations reçues de la partie opérative à l'aide de signalisation à l'affichage aussi l'opérateur Instruit la partie commande qui ordonne la partie opérative cette dernière rend compte à la partie Commande des opérations effectuées qui à son tour les signale à l'opérateur

### **2--Objectifs de l'automatisation :**

L'automatisation permet d'apporter des éléments supplémentaires à la valeur ajoutée par le Système. Ces éléments sont exprimables en termes d'objectifs par :

- accroître la productivité du système c'est-à-dire augmenté la quantité de produits Elaborés pendant une durée donnée. Cet accroissement de productivité exprime un Gain de valeur ajoutée sous forme :
- D'une meilleure rentabilité,
- D'une meilleure compétitivité,
- Améliorer la flexibilité de production,
- Améliorer la qualité du produit grâce à une meilleure répétitivité de la valeur ajoutée,
- S'adapter à des contextes particuliers,
- Adaptation à des environnements hostiles pour l'homme (milieu salin, spatial, Nucléaire...),
- Adaptation à des tâches physiques ou intellectuelles pénibles pour l'homme (Manipulation de lourdes charges, tâches répétitives parallélisées...),

- augmenter la sécurité, etc.

D'autres objectifs, à caractères sociaux, financiers... peuvent s'ajouter à ceux-ci.

### 3-Définition d'un automatisme :

L'automatisme consiste en l'étude de la commande de systèmes industriels.

Les techniques et méthodes d'automatisation sont en continuelle évolution ; elles font appel à des technologies : électromécaniques, électronique, pneumatique, hydraulique. Les automatismes sont présents dans tous les secteurs d'activité (menuiserie, textile, alimentaire, automobile...). La première amélioration des conditions de travail a été de remplacer l'énergie humaine fournie par l'ouvrier par une machine (Partie Opérative P.O).

### 4-Structure d'un système automatisé :

Un système automatisé est un moyen d'assurer l'objectif primordial d'une entreprise, la compétitivité de ses produits. Il permet d'ajouter une valeur aux produits entrants.

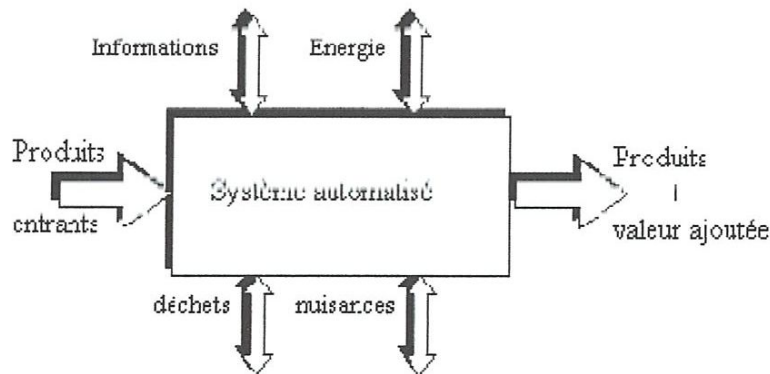


Fig.1.1 : Système automatisé

La notion de système automatisé peut s'appliquer aussi bien à une machine isolée qu'à une unité de production, voire même à une usine ou un groupe d'usines. Il est donc indispensable, avant toute analyse, de définir la frontière permettant d'isoler le système automatisé étudié de son milieu extérieur.

On peut décomposer fonctionnellement un système automatisé de production en deux parties.

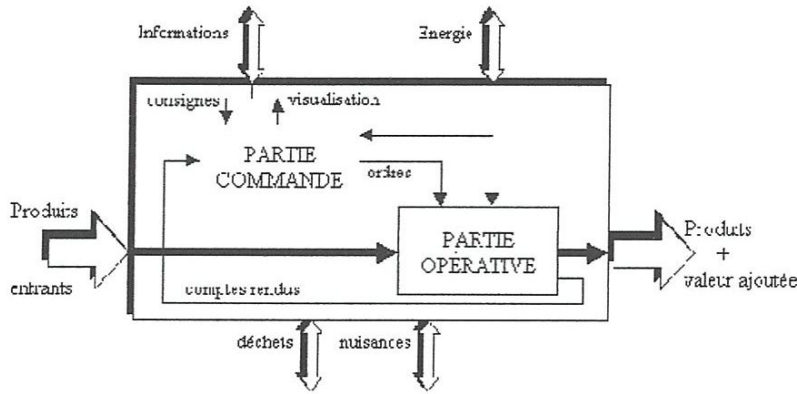


Fig.1.2 : Schéma général d'un automatisme

Chaque système automatisé comporte deux parties :

- une partie opérative (PO) dont les actionneurs (moteur électrique, vérin hydraulique, ...) agissent sur le processus automatisé,
- une partie commande (PC) qui coordonne les différentes actions de la partie opérative.

Les émissions d'ordres ou de signaux de commande vers la partie opérative sont transmises par les pré actionneurs, les comptes rendus sont fournis à la partie commande par les capteurs.

**5-Les capteurs :**

**5.1-définition :**

Un capteur est un organe chargé de prélever une grandeur physique à mesurer et de la transformer en une grandeur exploitable. Le capteur est un organe de saisie d'informations. C'est le premier maillon de toute une chaîne de mesure, acquisition de données, de tout système d'asservissement, régulation, de tout dispositif de contrôle, ...

**5.2-Constitution d'un capteur :**

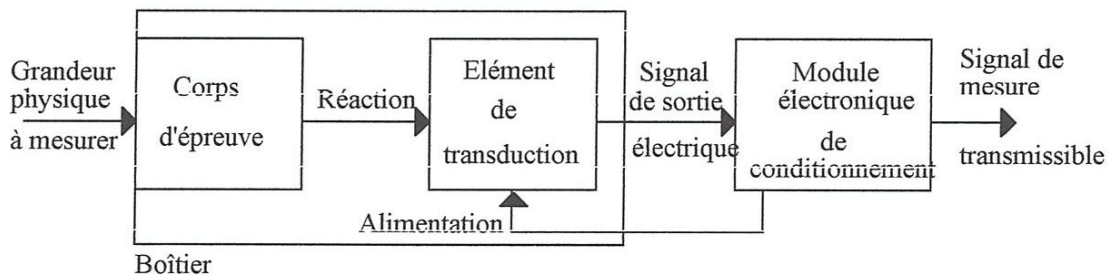


Fig.1.3 : Constitution d'un capteur

- **Corps d'épreuve :** élément mécanique qui réagit sélectivement à la grandeur à mesurer (appelée aussi mesurer).

But : transformer la grandeur à mesurer en une autre grandeur physique dite mesurable.

- **Elément de transduction** : élément sensible lié au corps d'épreuve. Il traduit les réactions du corps d'épreuve en une grandeur électrique constituant le signal de sortie.
- **Boîtier** : élément mécanique de protection, de maintien et de fixation du capteur.
- **Module électronique de fonctionnement** : il a, selon les cas, les fonctions suivantes :
  - alimentation électrique du capteur (si nécessaire)
  - mise en forme et amplification du signal de sortie
  - filtrage, amplification
  - conversion du signal

### 5.3-Mode de fonctionnement des capteurs :

Les capteurs fonctionnent selon deux principes de base suivant l'origine du signal électrique de sortie. On distingue :

- les capteurs actifs fonctionnant en générateur
- les capteurs passifs fonctionnant en modulateur.

Dans les capteurs actifs; une partie de l'énergie physique prélevée sur la mesurer est transformée directement en une énergie électrique qui constitue le signal de sortie. Ce signal est un courant, une tension ou une quantité d'électricité. Les signaux de sortie délivrés par les capteurs actifs sont de faible puissance. Ils sont dits de bas niveau et doivent être amplifiés pour pouvoir être ensuite transmis à distance.

Dans les capteurs passifs, c'est l'impédance du capteur qui est sensible aux variations du mesurer. Ces variations d'impédance ne sont mesurables que par l'intermédiaire d'un circuit électronique de pré conditionnement. Les capteurs passifs doivent être alimentés par une source d'énergie extérieure. Cette source peut être une tension continue ou modulée en fréquence.

Ces modes de fonctionnement correspondent à des principes de base et s'adapte essentiellement aux capteurs à sortie analogique. Le mode de fonctionnement des capteurs à sortie numérique et logique est des cas particuliers.

### 5.4-principales caractéristiques des capteurs :

#### 5.4.1-Etendue de mesure :

Domaine de mesure pour lequel les indications du capteur ne doivent pas être entachées d'une erreur supérieure à l'erreur maximale tolérée. On appelle les valeurs limites du domaine, « portée minimale » et « portée maximale ».



#### 5.4.2-Sensibilité :

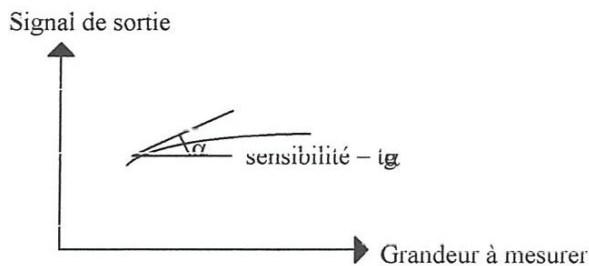
C'est le rapport de la variation du signal de sortie à la variation correspondante de la grandeur à mesurer.

C'est à dire à la pente de la courbe de réponse du capteur pour une valeur donnée :

$$S = DS/Dc$$

DS : variation de sortie

De : variation de l'entrée



#### 5.4.3-Précision :

C'est l'aptitude du capteur à donner des indications proche de la valeur vraie de la grandeur mesurée.

#### 5.4.4-Fidélité et Justesse :

La **justesse** est la qualité d'un capteur à fournir des indications précises.

La **fidélité** est la qualité d'un capteur à fournir des indications identiques pour une même valeur de la grandeur à mesurer.

#### 5.4.5-rapidité :

C'est l'aptitude du capteur à suivre dans le temps les variations de la grandeur à mesurer .

Il faut donc tenir compte du temps de réponse, de la bande passante et la fréquence de coupure du capteur.

#### 5.4.6-Stabilité

La stabilité qualifie la capacité d'un capteur à conserver ses performances pendant une longue durée (problème de dérive du zéro par exemple).

#### 5.4.7-Respectabilité et Reproductibilité :

La **respectabilité** est l'étroitesse de l'accord entre les résultats de mesures successifs d'une même grandeur effectuée avec la même méthode, par le même observateur, avec les mêmes instruments de mesure et à des intervalles de temps assez courts.

La **reproductibilité** est l'étroitesse de l'accord entre les résultats de mesures successifs d'une même grandeur dans le cas où les mesures sont effectuées dans les conditions différentes que pour la respectabilité.

#### **5.4.8-Bruit de fond :**

C'est une variation parasite, souvent aléatoire, du signal de sortie, dont la valeur moyenne est nulle et qui vient se superposer à la valeur à mesurer

### **6-Types des capteurs :**

#### **6.1-Capteur capacitif :**

Les capteurs capacitifs sont des capteurs de proximité qui permettent de détecter des objets métalliques ou isolants. Lorsqu'un objet entre dans le champ de détection des électrodes sensibles du capteur, il provoque des oscillations en modifiant la capacité de couplage du condensateur.

#### **6.2 –Capteur inductif :**

Les capteurs inductifs produisent à l'extrémité leur tête de détection un champ magnétique oscillant. Ce champ est généré par une bobine et une capacité montée en parallèle. Lorsqu'un objet métallique pénètre dans ce champ, il y a perturbation de ce champ puis atténuation du champ oscillant. Cette variation est exploitée par un amplificateur qui délivre un signal de sortie, le capteur commute.

#### **6.3-Capteur optique :**

Un capteur photoélectrique est un capteur de proximité. Il se compose d'un émetteur de lumière associé à un récepteur. La détection d'un objet se fait par coupure ou variation d'un faisceau lumineux. Le signal est amplifié pour être exploité par la partie commande.

## II-Les automates programmables industriels

### 1-Introduction :

Les automates programmables industriels (API) 1969 où ils répondaient aux désirs des industries de l'automobile de développer des chaînes de fabrication automatisées qui pourraient suivre l'évolution des techniques et des modèles fabriqués.

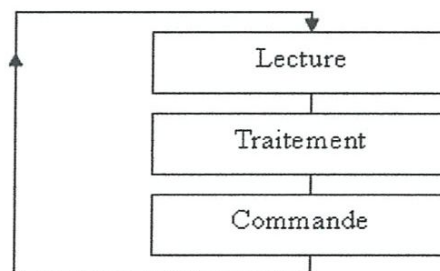
Un API se distingue d'un calculateur par le fait qu'il s'agit d'un système électronique programmable spécialement adapté pour les non informaticiens. Il est en général destiné à être mis entre les mains d'un personnel dont la formation a été surtout orientée vers l'électromécanique. L'API s'est substitué aux armoires à relais en raison de sa souplesse (mise en œuvre, évolution ...), mais aussi parce que dans les automatisations de commande complexe, les coûts de câblage et de mise au point devenaient trop élevés

### 2-Définition d'un automate programmable :

L'automate programmable industriel (API) est une machine électronique programmable destinée à piloter dans une ambiance industrielle et en Temps réel des procédés logiques séquentiels. Autrement dit, un Utilisateur (censé être un automaticien) l'utilise pour le contrôle et essentiellement la commande d'un procédé industriel en assurant l'adaptation nécessaire entre tout ce qui est de grande puissance par rapport à ce qui est de faible puissance côté commande. Son objectif principal est de rendre tout le mécanisme de type "laisser faire seul" : le système contrôle ses sorties, décide et agit sur ses entrées afin de maintenir le fonctionnement comme prévu par l'utilisateur. C'est le principe de l'automatisme.

### 3- Principe de fonctionnement :

L'automate programmable fonctionne par déroulement cyclique du programme. Le cycle comporte trois opérations successives qui se répètent normalement comme suit :



-**La lecture** : la scrutation des entrées binaires pour transférer leurs états dans la zone image d'entrée

-**Le traitement** : le processeur exécute les instructions de la mémoire programme en fonction des informations de la mémoire des données. Cette exécution se traduit par la modification de certaines variables et leur mise à jour dans la zone correspondante.

-**La commande** : les images des sorties dans la mémoire des données sont transférées dans le module de sortie pour être converti en signaux électriques pour la commande des pres-actionneurs et des dispositifs de visualisation. Ces valeurs sont verrouillées jusqu'au cycle prochain.

Ce cycle se répète infiniment tant qu'il n'y a pas d'interruption interne ou externe qui engendre l'arrêt temporaire ou permanent de l'automate. A chaque cycle seul, l'automate fait une mise à jour de ses données en entrée, garde cet état des entrées et passe à la phase de traitement. Cette dernière phase nécessite un temps prédéfini pour qu'elle se termine, dépendant de la fréquence du processeur et de la technologie interne et de la nature du traitement aussi.

Une fois terminée, on est dans la troisième et finale phase de sortie, où l'automate mets à jour ses signaux de sortie dépendant des résultats obtenus lors du traitement des entrées. Ces sorties restent figées jusqu'au prochain cycle.

Chaque fois que l'on minimise le temps d'un cycle, on améliore l'efficacité de notre automate. Malheureusement, le constructeur joue le rôle principal dans ce cas puisqu'il fixe la fréquence interne en se référant au processeur qu'il a utilisé. Mais l'utilisateur peut minimiser ce temps écoulé en améliorant le coût de son algorithme.

#### 4- Constitution :

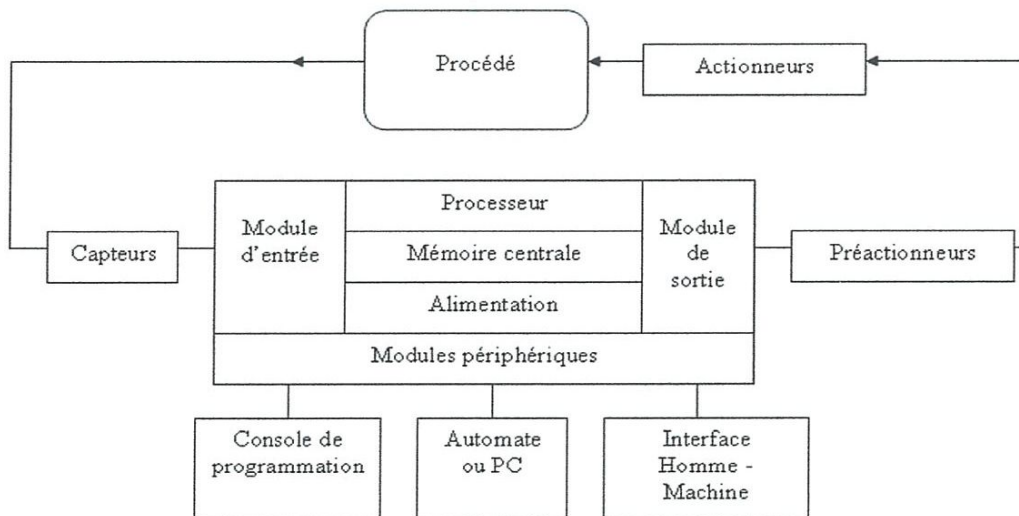


Fig.1.4 : Constitution des automates

- **Unité centrale** : Constituée du processeur et de la mémoire centrale.
- **Mémoire centrale** : Elle contient le programme moniteur, le programme utilisateur et les données. Le programme moniteur est un logiciel de base gérant le fonctionnement de la machine et fourni par le constructeur.
- **Module d'entrée** : Il transforme les signaux provenant des capteurs et des ordres de l'opérateur en signaux compréhensibles par l'automate. Le processeur stocke ensuite ces informations dans la mémoire de données image des entrées afin de les mémoriser.
- **Module de sortie** : Il transmet aux prés actionneurs et aux dispositifs de dialogue les ordres de commande et de signalisation résultants de l'exécution du programme. Le processeur vient chercher ses ordres dans la mémoire de données image des sorties, et le transfert e de sortie qui seront transformés en signaux électriques par la suite.
- **Module de communication** : Sert pour le dialogue entre l'automate et un autre équipement (automate, PC, ... etc.). L'exemple de : ModBus, ModBus Plus, Profibus, Inter Bus, Dévissent, LonWorks, Ethernet, FIPIO, FIPWAY, RS232, RS-485, AS, CANope.
- **Console de programme** : Le console de programmation permet le développement et le test du programme utilisateur ainsi que des opérations sur les éléments internes et l'entrée sortie.
- **Alimentation électrique**: Tous les automates actuels sont équipés d'une alimentation 240 V 50/60 Hz, 24 V DC. Les entrées sont en 24 V DC et une mise à la terre doit également être prévue.
- **Le processeur**: Son rôle consiste d'une part à organiser les différentes relations entre la zone mémoire et les interfaces d'entrées et de sorties et d'autre part à gérer les instructions du programme.
- **Les entrées -sorties**: Il existe deux sortes de signaux E/S
- **Les E/S TOR** : le mot TOR vient de "Tout Ou Rien" qui représente les signaux logiques. Les dispositifs d'entrée binaire réalisent, outre l'acquisition de l'information, les opérations suivantes :
  - Mise en forme du signal (calibrage)
  - Filtrage (élimination des parasites)
  - Isolation (galvanique ou par lumière)
- **Les E/S analogiques** : Les entrées analogiques transforment une grandeur analogique variant d'une façon continue en un code numérique, généralement sur 11 bits plus un bit de signe. Ces entrées disposent d'un seul convertisseur A/N (CAN), elles sont scrutées les

unes à la suite des autres par un multiplexeur (MUX). Par contre, les sorties analogiques disposent d'un seul convertisseur par voie.

Les E/S analogiques sont caractérisées par l'amplitude du signal (VH et VL), par la vitesse de conversion et par la grandeur électrique (courant ou tension).

Il existe, au niveau des entrées, trois types d'entrées analogiques :

- Haut niveau : 0-10V, 0-20mA, 4-20mA
- Pour thermocouple : 0-20mV, 0-50mV, 0-100mV
- Pour sondes Pt100 : 0-100mV, 0-250mV, 0-400mV

## **5-Architecture des automates :**

### **5-1-Aspect extérieur :**

Les automates peuvent être de type compact ou modulaire.

De type compact, on distinguera les modules de programmation (LOGO de Siemens, ZELIO descendric, MILLENIUM de Crozet ...) du micro automates.

Il intègre le processeur, l'alimentation, les entrées et les sorties. Selon les modèles et les fabricants, il pourra réaliser certaines fonctions supplémentaires (comptage rapide, E/S analogiques ...) et recevoir des extensions en nombre limité.

Ces automates, de fonctionnement simple, sont généralement destinés à la commande de petits automatismes.

De type modulaire, le processeur, l'alimentation et les interfaces d'entrées / sorties résident dans des unités séparées (modules) et sont fixées sur un ou plusieurs racks contenant le "fond de panier" (bus plus connecteurs).

Ces automates sont intégrés dans les automatismes complexes où puissance, capacité de traitement et flexibilité sont nécessaires.

Automate compact (Allen Bradley) Automate modulaire (Mo dicton)

## Exemple : Automate modulaire (Siemens)

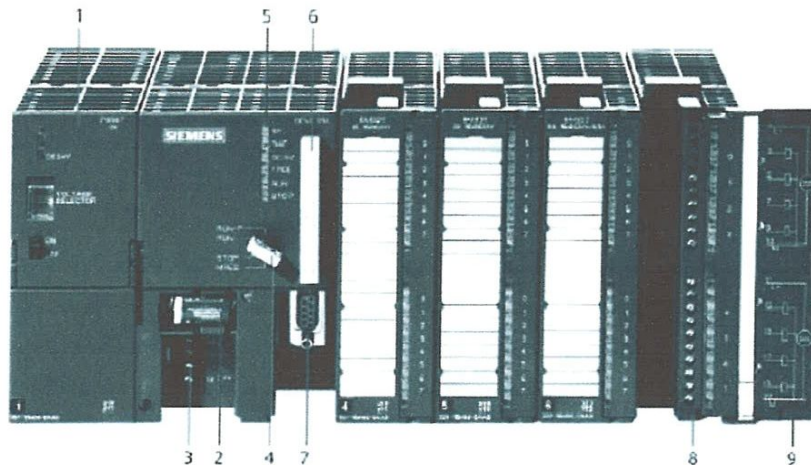


Fig.1.5 : Automate modulaire (siemens)

- |  |                              |
|--|------------------------------|
| 1- Module d'alimentation                     | 6- Carte mémoire             |
| 2 -Pile de sauvegarde                        | 7-Interface multipoint (MPI) |
| 3 -Connexion au 24V CC                       | 8-Connecteur frontal         |
| 4 -Commutateur de mode (à clé)               | 9-Volte en face avant        |
| 5 -LED de signalisation d'état et de défauts |                              |

**5-2-Structure interne :**

- Module d'alimentation** : il assure la distribution d'énergie aux différents modules
- **Unité centrale** : à base de microprocesseur, elle réalise toutes les fonctions logiques, arithmétiques et de traitement numérique (transfert, comptage, temporisation ...).
- Le bus interne** : il permet la communication de l'ensemble des blocs de l'automate et des éventuelles extensions.
- Mémoires** : Elles permettent de stocker le système d'exploitation (ROM ou PROM), le programme (EEPROM) et les données système lors du fonctionnement (RAM). Cette dernière est généralement secourue par pile ou batterie. On peut, en règle générale, augmenter la capacité mémoire par adjonction de barrettes mémoires type PCMCIA.
- **Interfaces d'entrées / sorties** :
  - **Interface d'entrée** : elle permet de recevoir les informations du S.A.P. ou du pupitre et de mettre en forme (filtrage, ...) ce signal tout en l'isolant électriquement (optocouplage).
  - Interface de sortie** : elle permet de commander les divers pré-actionneurs et éléments de Signalisation du S.A.P. tout en assurant l'isolement électrique.

### 5-3-Fonctions réalisées :

Les automates compacts permettent de commander des sorties en T.O.R et gèrent parfois des fonctions de comptage et de traitement analogique.

Les automates modulaires permettent de réaliser de nombreuses autres fonctions grâce à des modules intelligents que l'on dispose sur un ou plusieurs racks. Ces modules ont l'avantage de ne pas surcharger le travail de la CPU car ils disposent bien souvent de leur propre processeur.

### 5-4-Principales fonctions :

- **Cartes d'entrées / sorties** : Au nombre de 4, 8, 16 ou 32, elles peuvent aussi bien réaliser des fonctions d'entrées, de sorties ou les deux. Ce sont les plus utilisées et les tensions disponibles sont normalisées (24, 48, 110 ou 230V continu ou alternatif ...). Les voies peuvent être indépendantes ou posséder des "communs".

Les cartes d'entrées permettent de recueillir l'information des capteurs, boutons ... qui lui sont raccordés et de la matérialiser par un bit image de l'état du capteur.

Les cartes de sorties offrent deux types de technologies : les sorties à relais électromagnétiques (bobine plus contact) et les sorties statiques (à base de transistors ou de triacs).

-**Cartes de comptage rapide** : elles permettent d'acquérir des informations de fréquences élevées incompatibles avec le temps de traitement de l'automate.

Exemple : signal issu d'un codeur de position.

-**Cartes de commande d'axe** : Elles permettent d'assurer le positionnement avec précision d'élément mécanique selon un ou plusieurs axes. La carte permet par exemple de piloter un servomoteur et de recevoir les informations de positionnement par un codeur. L'asservissement de position pouvant être réalisé en boucle fermée.

- **Cartes d'entrées / sorties analogiques** : Elles permettent de réaliser l'acquisition d'un signal analogique et sa conversion numérique (CAN) indispensable pour assurer un traitement par le microprocesseur. La fonction inverse (sortie analogique) est également réalisée.

Les grandeurs analogiques sont normalisées : 0-10V ou 4-20mA.

#### -Autres cartes :

- Cartes de régulation PID

-Cartes de pesage

- Cartes de communication (Ethernet ...)



-Cartes d'entrées / sorties déportées

### 6-Traitement du programme automate :

Tous les automates fonctionnent selon le même mode opératoire.

**-Traitement interne** : des passages en RUN / STOP, L'automate effectue des opérations de contrôle et met à jour certains paramètres systèmes (détection mises à jour des valeurs de l'horodateur, ...).

**-Lecture des entrées** : L'automate lit les entrées (de façon synchrone) et les recopie dans la mémoire image des entrées.

**-Exécution du programme** : L'automate exécute le programme instruction par instruction et écrit les sorties dans la mémoire image des sorties.

**-Ecriture des sorties** : L'automate bascule les différentes sorties (de façon synchrone) aux positions définies dans la mémoire image des sorties.

Ces quatre opérations sont effectuées continuellement par l'automate (fonctionnement cyclique). On appelle scrutation l'ensemble des quatre opérations réalisées par l'automate et le temps de scrutation est le temps mis par l'automate pour traiter la même partie de programme. Ce temps est de l'ordre de la dizaine de millisecondes pour les applications standard

Le temps de réponse total (TRT) : est le temps qui s'écoule entre le changement d'état d'une entrée et le changement d'état de la sortie correspondante :

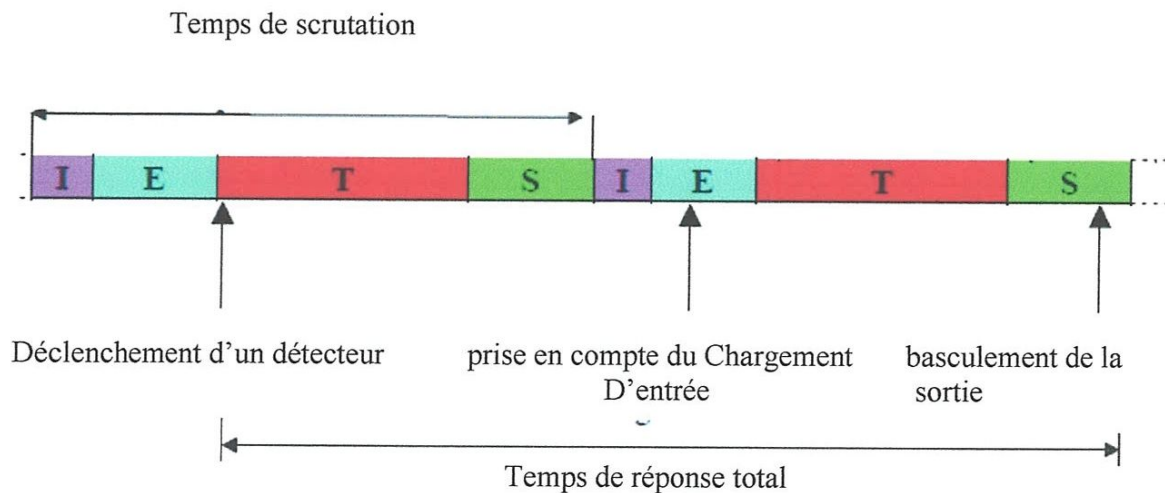


Fig.1.6 : temps de réponse totale

Le temps de réponse total est au plus égal à deux fois le temps de scrutation (sans traitement particulier).

Le temps de scrutation est directement lié au programme implanté. Ce temps peut être fixé à une valeur précise (fonctionnement périodique), le système indiquera alors tout dépassement de période. Dans certains cas, on ne peut admettre un temps de réponse aussi long pour certaines entrées : ces entrées pourront alors être traitées par l'automate comme des événements (traitement événementiel) et prises en compte en priorité (exemples : problème de sécurité, coupure d'alimentation ...).

Certains automates sont également pourvus d'entrées rapides qui sont prises en compte avant le traitement séquentiel mais le traitement événementiel reste prioritaire.

### 7-- Langages d'automates :

Les automates programmables industriels doivent pouvoir être utilisés facilement par du personnel habitué aux techniques classiques d'automatisation et peu à l'informatique. Ceci a conduit les constructeurs des API à concevoir des langages d'application spécialement adaptés à la réalisation d'automatisme. On distingue :

-Les langages graphiques :

- **LD** : **Ladder Dia** gram (Diagrammes échelle)
- **FBD** : **Fonction Block Diagrammes** (Logigrammes)
- **SFC** : **Séquentiel Fonction Charte** (Graf cet)

-Les langages textuels :

- **IL** : **Instruction List** (Liste d'instructions).
- **ST** : **Structure Texte** (Texte structuré).

**Chapitre II :**

**Grafcet**

### 1-Introduction :

Le GRA.F.C.E.T. (Élaboré en 1977) est l'abréviation de l'expression Graphe Fonctionnel de Commande Etape/Transition. Il permet de décrire tous les comportements attendus d'un automate de commande face aux événements ou informations issus d'un processus automatisé. Il résulte d'une synthèse réalisée par l'A.F.C.E.T., de tous les outils de description existants. En d'autres termes, c'est un modèle graphique de représentation du cahier des charges d'un automate logique. Le grafcet est devenue une norme d'abord française (AFNOR C03190) en 1982, puis une norme internationale (IEC 848) en 1987.

Le fonctionnement d'un automate comporte des durées pendant lesquelles les grandeurs physiques restent fixes (étapes ou états) et des durées pendant lesquelles certaines variables varient d'une valeur à une autre (transitions).

### 2-Définition :

Le **GRAFCET** (Graphe Fonctionnel de Commande par Etapes et Transitions) est un diagramme fonctionnel dont le but est de décrire graphiquement, suivant un cahier des charges, les différents comportements de l'évolution d'un automate séquentiel. Il est parfois simple à utiliser et rigoureux sur le plan formel et constitue un unique outil de dialogue entre toutes les personnes collaborant à la conception, à l'utilisation ou à la maintenance de la machine à automatiser.

### 3- Types de représentation du Grafcet :

Il y a deux types de représentation :

-La représentation fonctionnelle ou de niveau 1 donne une interprétation de la solution retenue pour un problème posé, en précisant la coordination des tâches opératives. Elle permet une compréhension globale du système.

-La représentation technologique ou de niveau 2 donne une interprétation en tenant compte des choix technologiques relatifs à la partie de commande de l'automatisme ; le type et la désignation des appareillages (S1, KM, Ka...).

### 4-Les éléments de bases :

Un GRAFCET se compose de 3 éléments principaux:

- Etapes et actions associées aux étapes
- Transitions
- Liaisons orientées

#### 4.1 - Etape :

Une étape est une situation dans laquelle les variables d'entrée et de sortie de la partie commande restent inchangées. Autrement dit, l'étape représente un état du système dans lequel les informations d'entrée (consignes et comptes-rendus) et les informations de sortie (ordres et visualisations) de la partie commande restent identiques à elles-mêmes. L'étape est représentée par un carré repéré numériquement (fig1.1). Les actions associées sont marquées en clair dans un rectangle à droite du carré représentant l'étape.

La situation initiale d'un système automatisé est indiquée par une étape dite étape initiale et représentée par un carré double.

- Représentée par:

1 carré et 1 repère alphanumérique

- Une étape est:

ACTIVE ou INACUVE

- Entrée/Sortie de l'étape:

Au -DESSUS \ au -DESSOUS



Fig.2.1 : Etape

#### 4.2 - Actions associées aux étapes :

D'une façon générale dans un GRAFCET une action est associée à chaque étape, mais on peut rencontrer aussi une même action associée à plusieurs étapes ou plusieurs actions associées à la même étape.

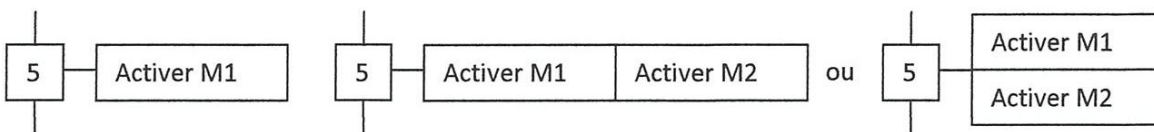


Fig.2.2 : Actions associées aux étapes

- Actions : obligatoirement exprimées par un verbe à l'infinitif
- Pour une étape : une ou plusieurs actions sans aucune notion d'ordre
- Etape sans action : étape d'attente ou étape assurant le respect de la syntaxe du GRAFCET

##### 4.2.1- Classification des actions associées aux étapes :

###### a- Actions – Ordres – Effets :

- Le choix technologique de la partie opérative pour produire les effets attendus.
- Le choix de la technologie pour la PC permet de décrire les ordres à donner pour provoquer les actions qui produisent les effets attendus.

Exemple :

Ordre	Action	effet
Enclencher le contacteur KM1	Rotation du moteur M1	Avance du plateau

### b- Nature et classification des actions ou des ordres :

Les actions peuvent être classées en fonction de leur durée par rapport à celle de l'étape. On distingue :

- les ordres continus ;
- les ordres conditionnels ;
- les ordres retardés ;
- les ordres de durées limitées ;
- les ordres fugitifs ;
- les ordres maintenus sur plusieurs étapes ;
- les actions mémorisées.

#### 1- Ordres continus:

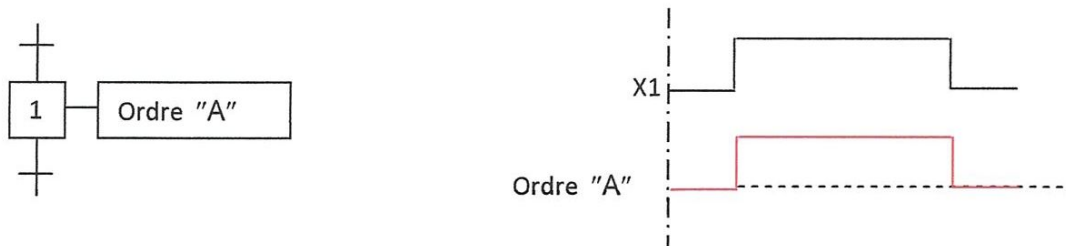


Fig.2.3: Action continue

L'ordre est émis, de façon continue, tant que l'étape, à laquelle est active.

#### 2- Ordres conditionnels:

L'ordre ne peut être émis que si l'étape à laquelle il est associé est active et si une condition logique spécifiée est réalisée. Cette condition peut être indiquée à l'intérieur ou à l'extérieur du rectangle d'action, suivant la place disponible.

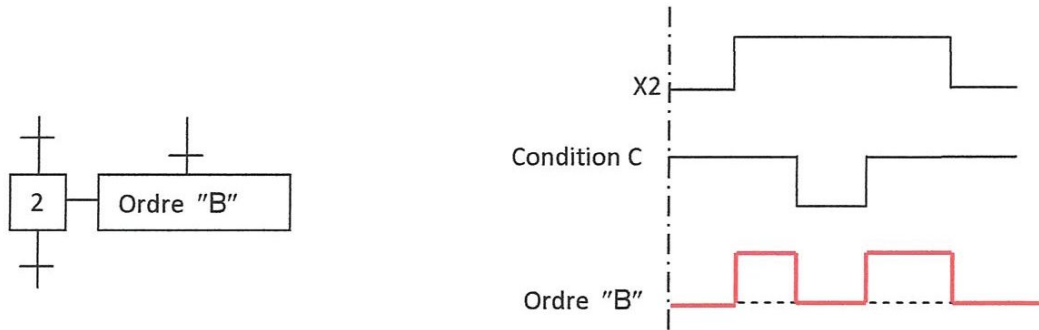


Fig.2.4: Ordre conditionnel

**3- Ordre retardé:**

Cet ordre est un cas particulier de l'ordre conditionnel où le temps intervient comme condition logique. L'indication du temps s'effectue par la notation générale " t / xi / q " dans laquelle "xi" indique l'étape prise comme origine du temps et "q" est la durée du retard.

Exemple : "t/x3/ 5s" : prendra la valeur logique 1, 5s après la dernière activation de l'étape 3.

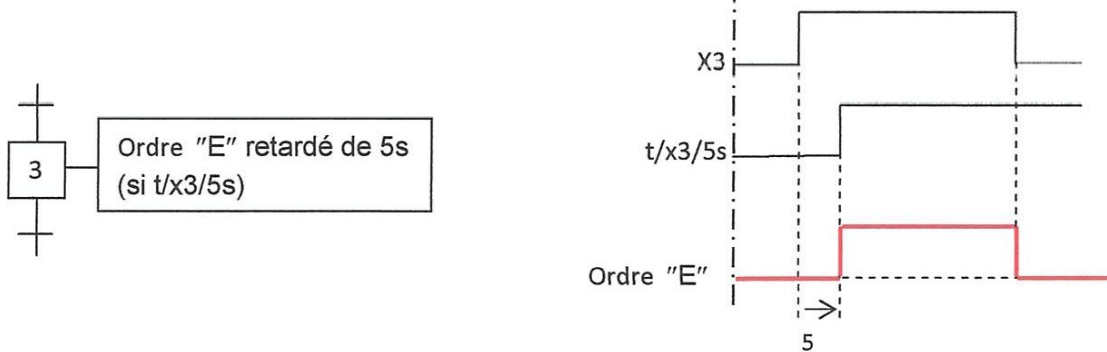


Fig.2.5: Action retardée

**4- Ordre de durée limitée:**

L'ordre est émis dès l'activation de l'étape à laquelle il est associé ; mais la durée de cet ordre sera limitée à la valeur spécifiée.

L'ordre "G" est limité à 5s après l'activation de l'étape 4.

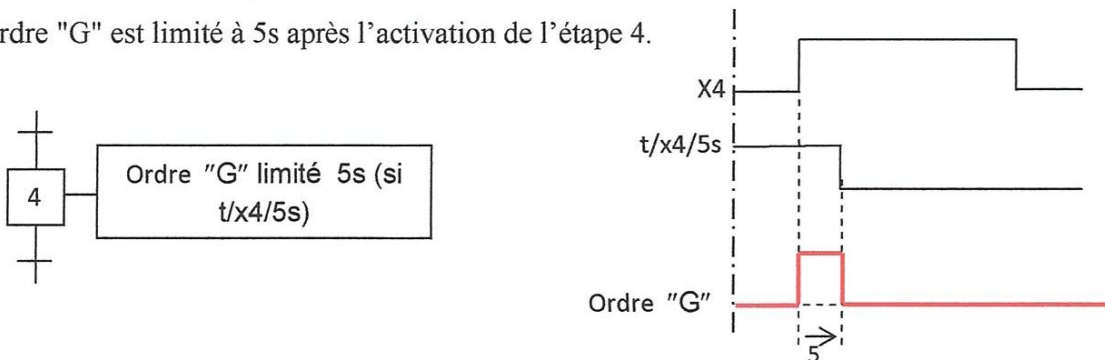


Fig.2.6 : Action de durée limitée

### 5- Ordre fugitif:

- A l'activation ou à la désactivation d'une étape :

Dans ce cas l'ordre fugitif, émis à l'activation ou à la désactivation de l'étape, est de durée théorique infiniment courte. En réalité cette durée sera suffisante pour pouvoir effectuer des actions ponctuelles sur la partie commande telles que positionnement des mémoires, présélection des compteurs, incrémentation des compteurs, etc.

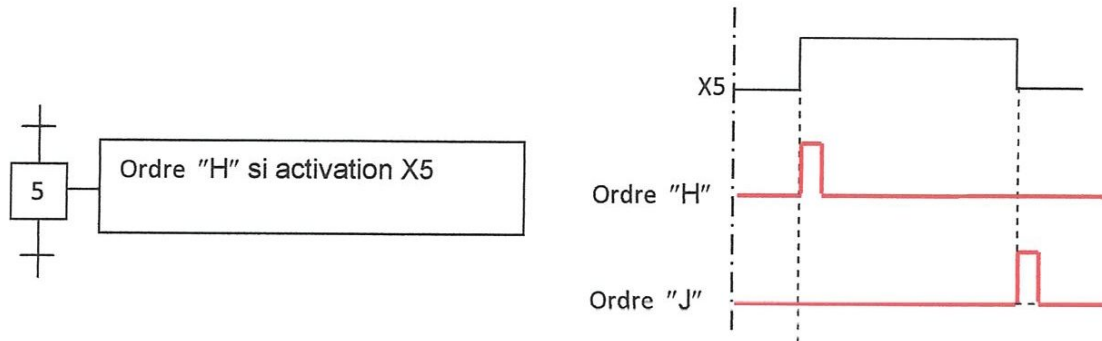


Fig.2.7: Action fugitive (étape)

- A l'apparition ou à la disparition d'une variable :

L'ordre fugitif est émis, pendant l'activation de l'étape, lors de l'apparition ou la disparition de la variable spécifiée.

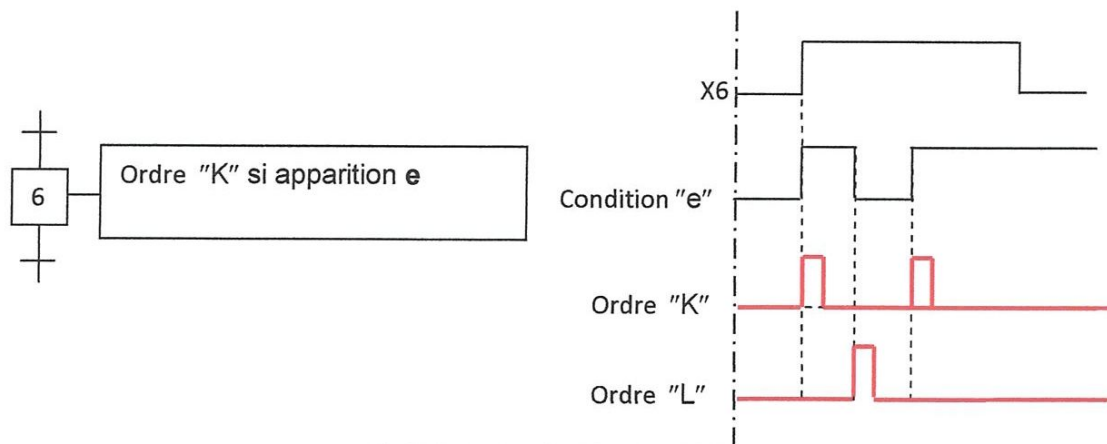


Fig.2.8: Action fugitive (variable)

### 6- Ordre maintenu sur plusieurs étapes:

Afin de maintenir la continuité d'une action devant se prolonger pendant l'activité de plusieurs étapes, il est possible :

- soit de répéter l'ordre continu relatif à cette action, dans toutes les étapes concernées;
- soit d'utiliser une description sous forme de séquences simultanées.



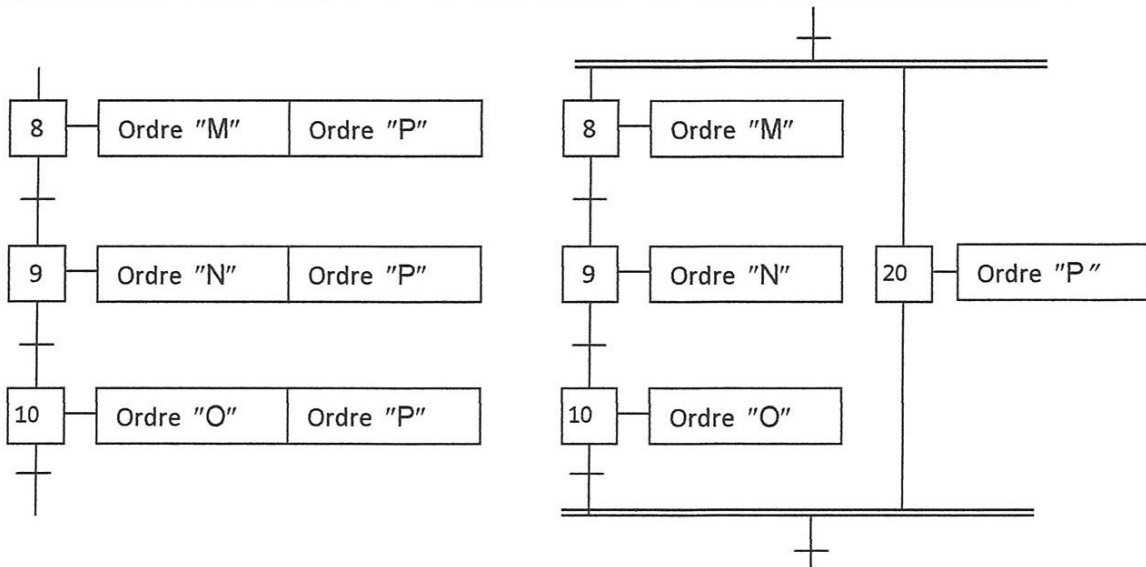


Fig.2.9: Ordre maintenue

**7- Action mémorisée par une fonction opérative auxiliaire:**

Le maintien d'un ordre, sur la durée d'activation de plusieurs étapes consécutives, peut également être obtenu par la mémorisation de l'action, obtenue par l'utilisation d'une fonction opérative auxiliaire appelée fonction de mémorisation (ou mémoire).

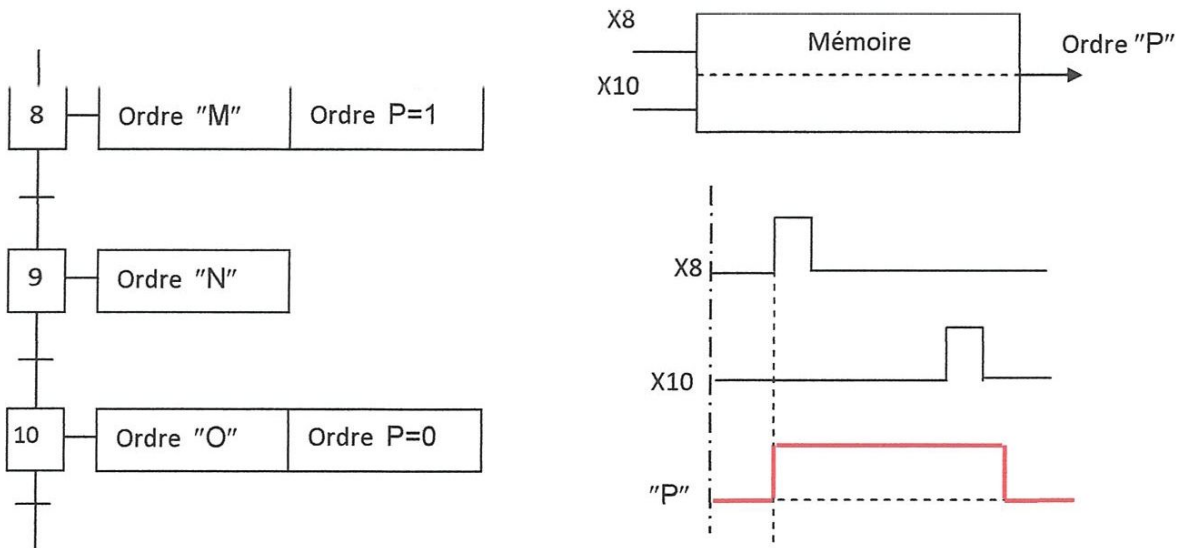


Fig.2.10: Action mémorisée par une fonction auxiliaire

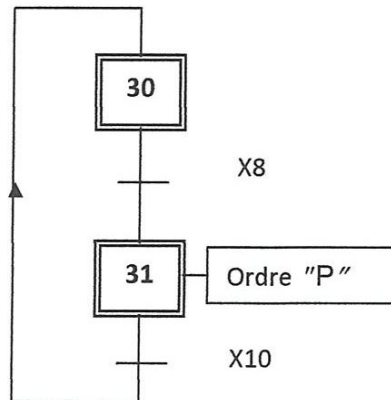
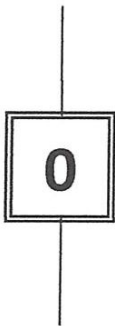


Fig.2.11: GRAFCET de fonction mémoire

Cette fonction pourra être décrite par un GRAFCET auxiliaire.

**4.3-Les étapes initiales :**



- Représentée par:  
1 double carré et 1 repère alphanumérique
- Une étape initiale est:  
ACTIVE à l'initiation du processus

Fig.2.12 : Les étapes initiales

**4.4 : la transition :**

Elle est située entre deux étapes consécutives ; elle indique l'évolution d'une étape vers l'étape suivante.

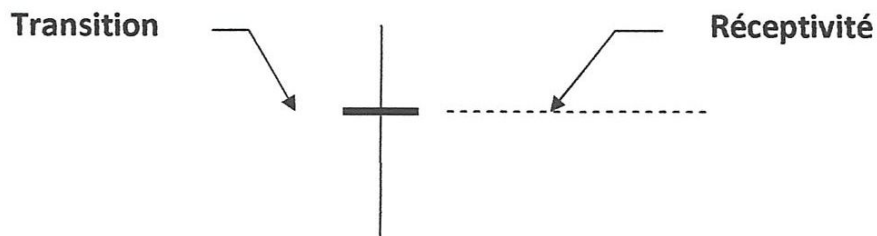


Fig2.13: Transition

A chaque transition est associée une condition logique appelée réceptivité.

La réceptivité qui est une information d'entrée (expression logique) est fournie par :

- l'opérateur : pupitre de commande ;
- la partie opérative : états des capteurs ;
- d'autres systèmes : dialoguent entre systèmes.

#### 4.5 - Liaisons orientées :

Elles relient les étapes aux transitions et les transitions aux étapes. Elles sont normalement orientées de haut vers le bas. Une flèche est nécessaire dans le cas contraire ou pour une meilleure compréhension.

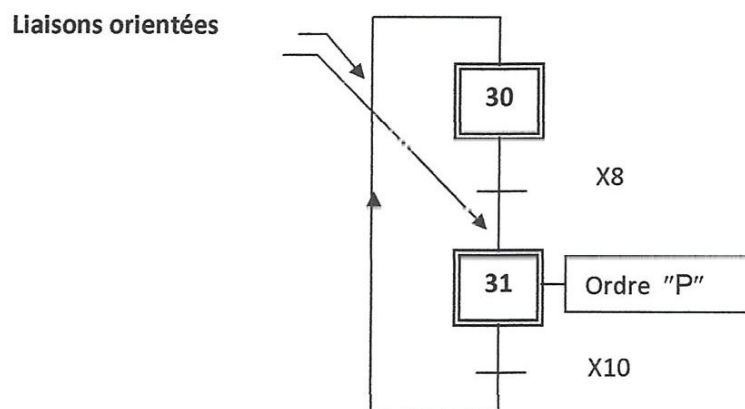


Fig2.13 : Liaisons orientées

#### 5-Règles d'évolution d'un GRAFCET :

On étudie les conditions dans lesquelles le GRAFCET évolue : conditions de passage d'une étape active vers une autre étape active.

##### 5.1 - Règle 1 : Initialisation

La situation initiale du 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 (étapes initiales ou étapes d'attente).

Dans un GRAFCET, il doit y avoir au moins une étape initiale.

##### 5.2 - Règle 2 : Franchissement d'une transition

Pour qu'une transition soit franchissable il faut qu'elle soit validée et que la réceptivité associée soit vraie.

On dit qu'une transition est validée (susceptible d'être franchie) lorsque toutes les étapes précédentes sont actives.

**5.3 - Règle 3 : Evolution des étapes actives**

Cette règle s'applique dans le cas d'un GRAFCET à une ou plusieurs séquences.

**Enoncé de la règle 3 :** Le franchissement d'une transition entraîne l'activation de toutes les étapes immédiatement suivantes et la désactivation de toutes les étapes immédiatement précédentes.

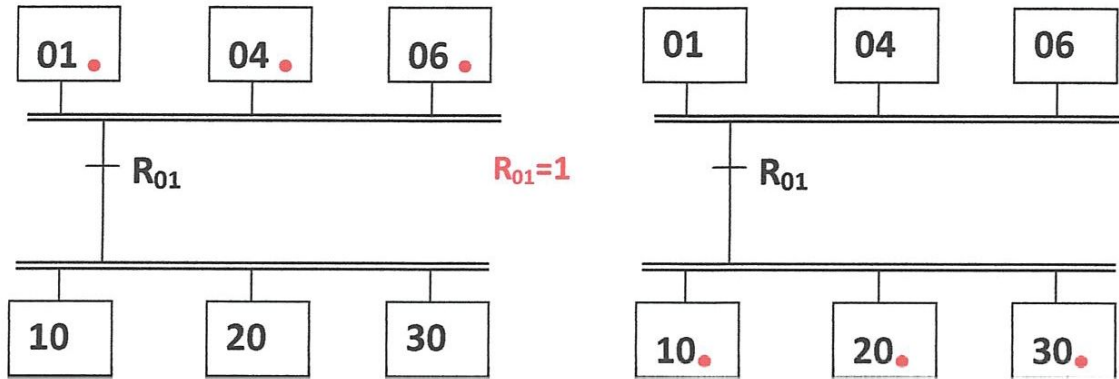


Fig.2.14: Illustration de la règle 3

**5.4 - Règle 4 :**

Plusieurs transitions simultanément franchissables, sont simultanément franchies. Cette règle servira, à la décomposition du GRAFCET en plusieurs autres GRAFCET, ou, à un GRAFCET à plusieurs séquences.

**5.5 - Règle 5 :**

Si, au cours de l'évolution d'un GRAFCET, une même étape doit être activée et désactivée simultanément, elle reste active.

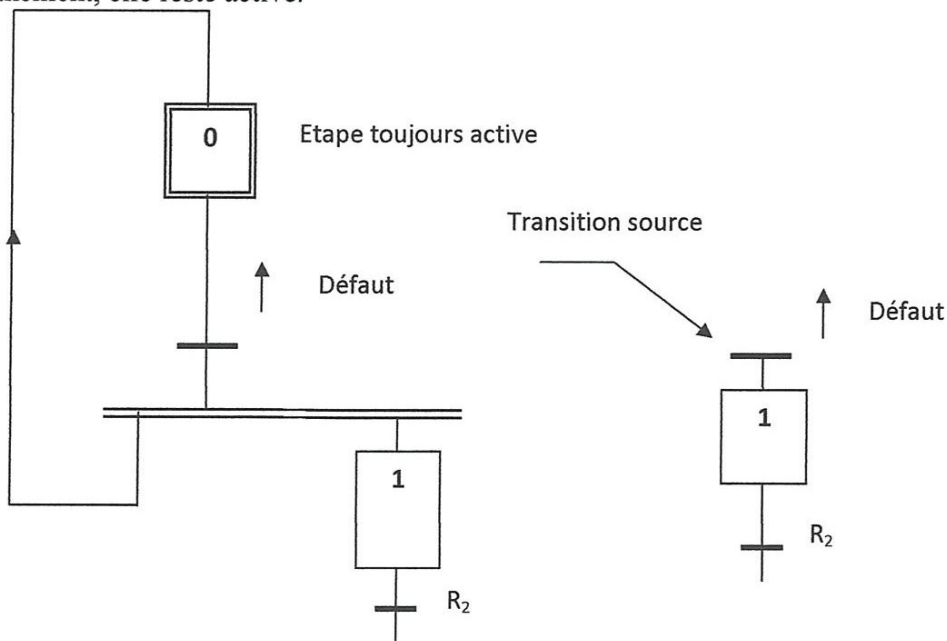


Fig.2.15: Illustration de la règle 5

**6 - GRAFCET à séquences multiples :**

**6.1 - Séquence :**

Une séquence, dans un Graf cet, est une suite d'étapes à exécuter l'une après l'autre.

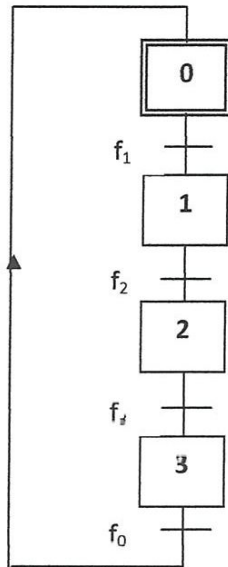


Fig.2.16: Graf cet à séquence unique.

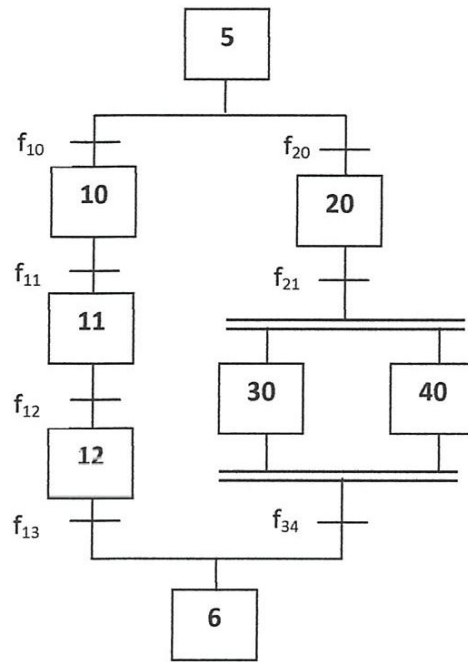


Fig.2.17: Graf cet à plusieurs Séquences

**6.2-Aiguillage (ou divergence en OU) :**

On dit qu'il y a divergence en OU lorsque le grafcet se décompose en plusieurs séquences selon un choix conditionnel.

Comme la divergence en OU on rencontre aussi la convergence en OU. On dit qu'il y a convergence en OU, lorsque deux ou plusieurs séquences du grafcet converge vers une seule séquence.

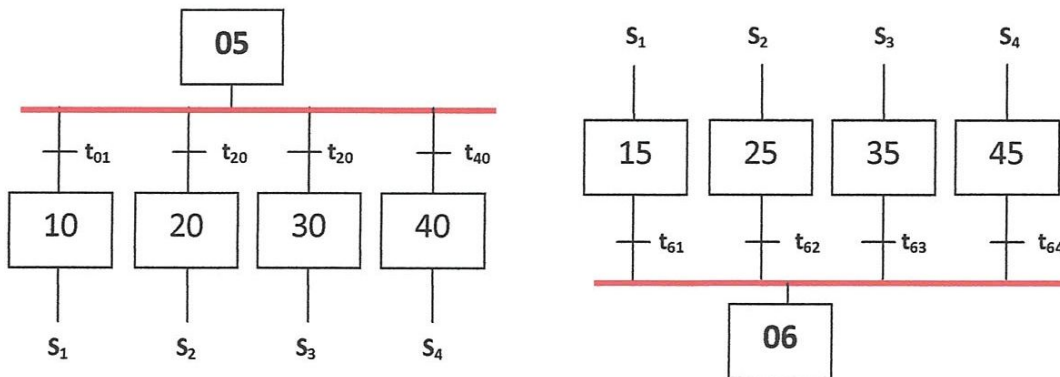


Fig.2.18: Divergence et convergence en OU

**6.3-Saut d'étapes et reprise de séquence :**

- Le saut d'étapes est une divergence en OU particulière qui permet de sauter des étapes lorsque les actions associées sont inutiles à réaliser.
- La reprise de séquence (ou boucle) permet de reprendre, une ou plusieurs fois, une séquence tant qu'une condition n'est pas obtenue.

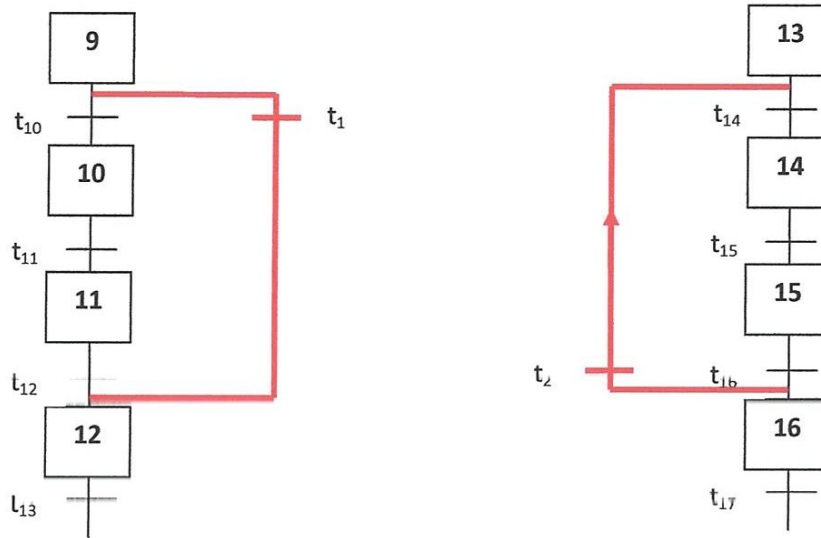


Fig.2.19: Saut d'étapes et Reprise de séquences.

**6.4 - Séquences simultanées (ou divergence -convergence en ET) :**

On dit qu'on se trouve en présence d'un parallélisme structurel, si le franchissement d'une transition conduit à activer plusieurs étapes en même temps, ces étapes déclencheront des séquences dont les évolutions seront à la fois simultanées et indépendantes.

Le début (divergence en ET) et la fin (convergence en ET) d'un parallélisme structurel sont représentés par deux traits parallèles.

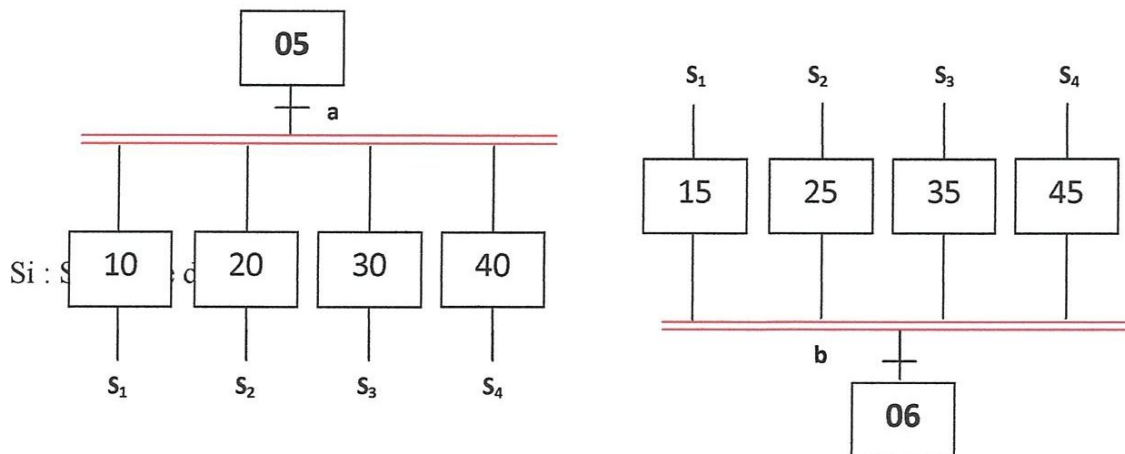


Fig.2.20: Divergence et convergence en ET

### 7 - Liaison de deux Grafcets :

Chaque étape est associée à une mémoire (bit image) lui permettant de fournir à la sortie un signal logique X qui peut servir comme réceptivité à une autre étape.

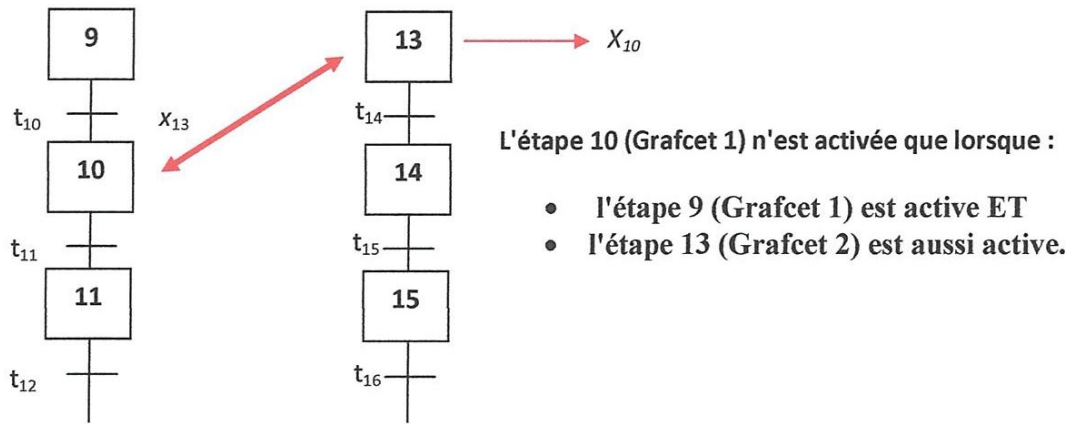


Fig.2.20 : Liaisons entre Grafcet

### 8-Synchronisation de deux Grafcets ;

#### 8.1 - Principe :

Synchroniser deux Grafcets revient à rendre l'évolution de l'un dépend de l'évolution de l'autre. Une machine et son poste de chargement automatique, par exemple, doivent avoir un fonctionnement synchronisé. En effet, la machine ne peut fonctionner qu'après le chargement d'une pièce; le poste de chargement ne peut fonctionner qu'après avoir dégagé la pièce c'est à dire que la machine aura effectué un cycle complet.

#### 8.2 - Méthode :

L'interdépendance est obtenue par les réceptivités : utilisation des mêmes variables d'entrée dans chacun des deux Grafcet

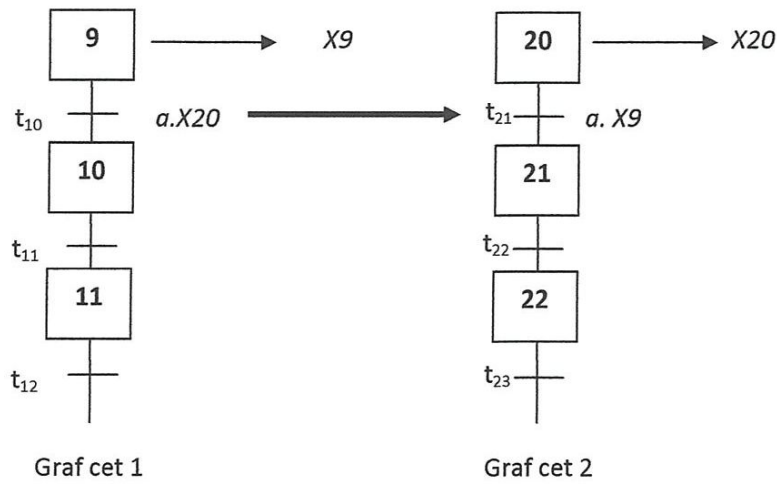


Fig.2.21: Synchronisation de deux Grafkets.

Chacune des étapes 10 (Grafcet 1) et 21 ne peuvent être activées qu'en même temps. Cela revient comme s'il s'agit d'une divergence en ET.

:



**Chapitre III:**

**STEP 7**

## 1-Introduction :

STEP 7 est un logiciel de programmation pour les systèmes SIMATIC S7 et par conséquent le logiciel de programmation de S7-300. STEP 7 offre toutes les fonctionnalités nécessaires pour configurer, paramétrer et programmer S7-300. Il met à disposition du programmeur des fonctions d'assistance. Pour résoudre efficacement les problèmes d'automatisation. Les caractéristiques de STEP 7 facilitent la tâche de programmation pour l'utilisateur.

STEP 7 fonctionne sous Windows 95/98/2000/NT/XP et son utilisation est tout aussi simple que les autres applications Windows! STEP 7 présente une interface graphique typique des autres programmes Windows : ce sont par exemple les stations, les modules et les programmes! STEP 7 autorise une programmation orientée objet. Tous les objets sont représentés à l'écran par des icônes. STEP 7 épaulé le programmeur avec une aide en ligne qui informe et qui propose une aide confortable et contextuelle.

## 2-Définition du STEP7 :

STEP 7 est le logiciel de base pour la configuration et la programmation de systèmes D'automatisation SIMATIC. Il fait partie de l'industrie logicielle SIMATIC. Le logiciel de base STEP 7 Existe en plusieurs versions :

- STEP 7-Micro/DOS et STEP 7-Micro/Win pour des applications autonomes simples sur SIMATIC S7 - 200.

- STEP 7 pour des applications sur SIMATIC S7-300/400, SIMATIC M7-300/400 et SIMATIC C7 Présentant des fonctionnalités supplémentaires : modules fonctionnels et de modules de communication

- Forçage et fonctionnement

- Possibilité d'extension grâce aux applications proposées par l'industrie logicielle SIMATIC (voir aussi Possibilités d'extension du logiciel de base STEP7)

- Possibilité de paramétrage d multiprocesseur

- Communication par données globales

- Transfert de données commandé par événement à l'aide de blocs de communication et de Blocs fonctionnels

- Configuration de liaisons

### 3-Fonctionnalités:

Le logiciel STEP 7 Professional fédère toutes les fonctions requises pour concevoir, configurer, programmer, tester, mettre en service et maintenir les systèmes d'automatisation SIMATIC.

STEP 7 Professional apporte une productivité de l'ingénierie sans précédent. Il concrétise tous les atouts du concept "Totale Intégrates Automation" :

- Interface utilisateur orientée objet, commune à tous les outils logiciels SIMATIC Industriel Software
- Base de données uniques des projets garantissant la cohérence des applications même complexes
- Cohérence des réseaux de communication entre les composants d'une application

### 4-Caractéristiques techniques:

STEP 7 Professional intègre en particulier les outils suivants :

- Interface utilisateur SIMATIC Manager commune à tous les outils logiciels intégrés et optionnels.
  - Tous les langages de programmation pour automates programmables définis dans le standard CEI 61131-3: schémas contact, logigrammes, listes d'instructions, graphes séquentiels (S7-GRAPH) et langages structurés (S7-SCL)
  - Le logiciel de simulation automate S7-PLCSIM pour la mise au point de programmes sans disposer des automates cible.
  - Outil configuration graphique des composants matériels et des réseaux de communication
- STEP 7 Professional comporte de nombreuses fonctions évoluées qui contribuent à l'efficacité du travail des automaticiens, par exemple:
- Programmation structurée, avec des blocs fonctions paramétrables et instanciers
  - Vérification globale de la cohérence d'un programme
  - Gestion multilingue de projet.
  - Interfaces d'import/export de code source des programmes, de la liste des noms symboliques de variables, de configurations matérielles.
  - Modèle objet STEP 7 pour réalisation de scripts destinés à "automatiser le travail de l'automaticien" .

### 5-Fonctions du logiciel de base :

Le logiciel de base est utilisé dans toutes les phases du processus de création de vos solutions d'automatisation, comme par exemple :

- La création et la gestion de projets,
- La configuration et le paramétrage du matériel et de la communication,
- La gestion des mnémoniques,
- La création de programmes, par exemple pour les systèmes cible S7,
- Le chargement de programmes dans des systèmes cible,
- Le test de l'installation d'automatisation,
- Le diagnostic lors de perturbations de l'installation.

La conception de l'interface utilisateur du logiciel STEP 7 répond aux connaissances ergonomiques modernes et son apprentissage est très facile

### 6-Interaction du logiciel et du matériel :

A l'aide du logiciel STEP 7 on peut créer le programme S7 dans un projet. L'automate S7 est constitué d'un module d'alimentation, d'une CPU et de modules d'entrées ou de sorties (modules d'E/S). L'automate programmable (AP) contrôle et commande à l'aide du programme STEP7 la machine. L'adressage des modules d'E/S se fait par l'intermédiaire des Adresses du programme STEP 7.

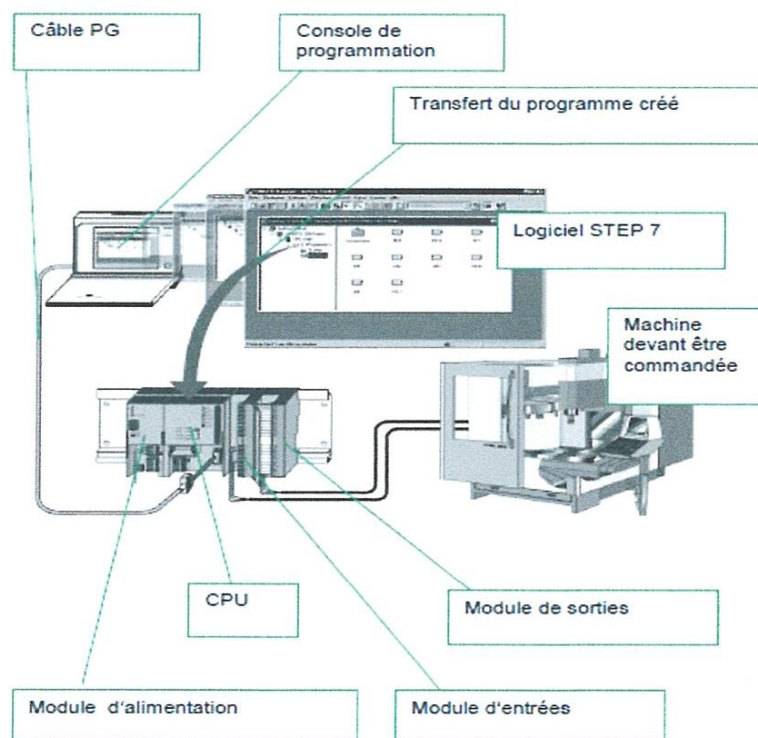


Fig.3.1. Interaction du logiciel et du matériel

### 7-SIMATIC Manager :

SIMATIC S7 désigne un produit de la société SIEMENS et est synonyme de la nouvelle gamme des automates programmables. STEP 7 est le logiciel permettant de configurer et de programmer les systèmes d'automatisation ; SIMATIC S7-300/400, ainsi que les systèmes intégrés compacts SIMATIC C7. La programmation et la configuration d'un système C7 se font de manière identique à celles d'un SIMATIC S7-300. STEP 7 est constitué d'un logiciel de base et de logiciels optionnels s'exécutant sous Windows. Le logiciel de base STEP 7 assiste son utilisateur dans toutes les phases du processus de création des solutions d'automatisation, comme par exemple :

- \* La création et la gestion de projets ;
- \* La configuration et le paramétrage du matériel et de la communication ;
- \* La gestion des mnémoniques ;
- \* la création de programmes pour systèmes cible S7;
- \* Le chargement de programmes dans des systèmes cible ;
- \* Le test de l'installation d'automatisation ;
- \* Le diagnostic lors de perturbations de l'installation ;
- \* Possibilité de paramétrage de modules fonctionnels et de modules de communication ;
- \* Forçage et fonctionnement multiprocesseur ;
- \* Communication par données globales ;
- \* Transfert de données commandé par événement à l'aide de blocs de communication et de blocs fonctionnels ;
- \* Configuration de liaisons

### 8- Utilisation de STEP 7 :

STEP 7 permet l'accès "de base" aux automates Siemens. Il permet de programmer individuellement un automate (en différents langages). Il prend également en compte le réseau des automates, ce qui permet d'accéder à tout automate du réseau (pour le programmer), et éventuellement aux automates de s'envoyer des messages entre eux. Il ne permet pas d'incorporer les ordinateurs dans le réseau (durant le fonctionnement, il n'y a pas de dialogue entre les PC et les automates, donc pas de supervision du processus par un logiciel centralisé, comme ce serait possible sous WINCC)

## 9-Les différents blocs :

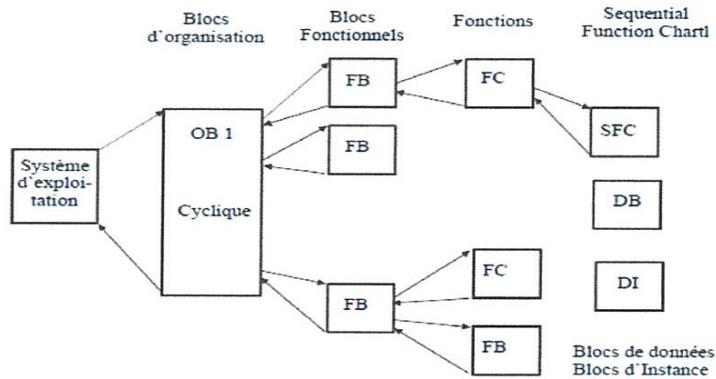


Fig.3.2.Les différents bloc

### 9.1-Bloc utilisateur :

#### 9.1.1-OB (bloc d'organisation) :

Un OB est appelé cycliquement par le système d'exploitation et constitue donc l'interface entre le programme utilisateur et le système d'exploitation. L'OB contient des instructions d'appels de blocs indiquant à l'unité de commande de l'automate l'ordre dans lequel il doit traiter les blocs.

#### 9.1.2-FB (bloc fonctionnel) :

Le FB dispose d'une zone de mémoire qui lui est affectée en propre. Il est possible d'affecter un bloc de données (DB) au FB à l'appel du bloc. Il est possible d'accéder aux données du DB d'instance via les appels contenus dans le FB. On peut affecter plusieurs DB à un FB. Il est possible d'appeler d'autres FB et FC dans un bloc fonctionnel via des instructions d'appels de blocs.

#### 9.1.3-FC (fonction) :

Une FC ne possède pas une zone de mémoire propre. Les données locales d'une fonction sont perdues après l'exécution de la fonction. Il est également possible d'appeler d'autres FB et FC dans une fonction via des instructions d'appels de blocs.

#### 9.1.4-DB (bloc de données) :

Les DB sont utilisés pour la mise à disposition d'espace mémoire pour les variables types donnés. Il existe deux types de blocs de données. Les DB globaux dans lesquels tous les OB, FB et FC peuvent lire les données enregistrées ou écrire des données et les DB d'instance qui sont affectés à un FB donné.

## 9-2- Bloc système pour fonctions standard et fonctions système :

Les blocs système sont des fonctions prêtes à l'emploi stockées dans la CPU.

Ces blocs peuvent être appelés par l'utilisateur et utilisés dans le programme, mais on ne peut pas changer ni accéder à leur programme.

On dispose dans STEP 7 des blocs système suivants :

### 9.2.1-SFB (bloc fonctionnel système) :

Bloc fonctionnel stocké dans le système d'exploitation de la CPU et pouvant être appelé par l'utilisateur.

### 9.2.2-SFC (fonction système) :

Fonction stockée dans le système d'exploitation de la CPU et pouvant être appelée par l'utilisateur.

### 9.2.3-SDB (données système) :

Zone de mémoire dans le programme configuré par différentes applications de STEP 7 (par exemple S7 Configuration, Communication Configuration...), pour le stockage des données dans le système d'automatisation.

## 10-Programmation symbolique :

### 10.1- Adresse absolue :

Chaque entrée et chaque sortie possèdent par défaut une adresse absolue déterminée par la configuration matérielle. Celle-ci est indiquée de manière directe, C'est à dire absolue. Cette adresse absolue peut être remplacée par des noms symboliques pouvant être librement choisis.

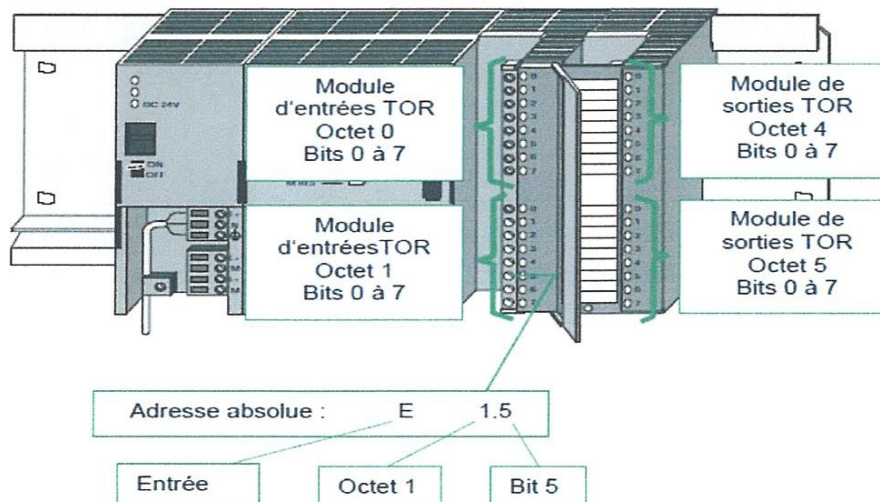


Fig.3.3.Adresse absolue

## 10.2-Programmation symbolique :

Nous pouvons affecter, dans la table des mnémoniques, un nom symbolique à toutes les Adresses absolues que nous voulons appeler dans le programme ainsi que le type de données, par exemple pour l'entrée E0.1 le mnémonique Commutateur 1. Ces noms sont utilisables dans toutes les sections du programme, C'est pourquoi on les appelle des variables globales.

La programmation symbolique permet d'alléger l'écriture de programme qui y gagne en clarté.

## 11-Les modes de programmation :

Le langage de programmation STEP7 dispose de plusieurs modes de représentation, selon les goûts de l'état de connaissance. En respectant certaines règles, le programme peut être conçu sous forme de liste d'instructions puis converti en un autre mode de représentation

### 11.1-Le langage CONT :

Le langage contacte est une représentation graphique de la tâche d'automatisation. Il ressemble à un schéma des circuits dans lesquels les circuits de courant sont disposés horizontalement afin d'être visibles à l'écran.

Les entrées sont représentées par des interrupteurs -| | - (ou -|/| - si entrée inversée), les sorties par des bobines -( ) - ou des bascules -(S) - -(R)

#### Exemple :

<Opérande      \_\_\_\_\_( )

L'opérande indique le bit auquel est affecté l'état de signal de la séquence combinatoire

<Opérande      \_\_\_\_\_| |\_\_\_\_\_

L'opérande indique le bit dont l'état de signal est interrogé

<Opérande      \_\_\_\_\_|/|\_\_\_\_\_

L'opérande indique le bit dont l'état de signal est interrogé

<Opérande      \_\_\_\_\_(S)

L'opérande indique le bit qui doit être mis à 1

<Opérande      \_\_\_\_\_(R)

L'opérande indique le bit qui doit être mis à 0



### 11.2-Langage LIST :

La tâche d'automatisation est écrite dans la Liste d'instructions à l'aide des différentes instructions. L'instruction (opération et opérande) symbolise la tâche à l'aide de mnémoniques.

Chaque mode de représentation du programme a ses avantages mais aussi ses limitations. Si quelques règles ont été respectées lors de la programmation, la compilation est possible dans les trois modes de représentation. Les programmes d'automatisation programmés en CONT ou LOG sont en principe toujours traduisibles en LIST. Le langage liste à l'avantage d'être traité plus rapidement par l'automate que les langages LOG et CONT dans la mémoire de programme de l'automate, le programme est toujours stocké en LIST.

### 11.3-langage LOG :

C'est un langage graphique, utilisant les symboles de l'électronique numérique (portes logiques). Il n'y a rien de spécial à dire, c'est très intuitif. On peut utiliser plusieurs entrées pour une même porte, placer des inverseurs sur les entrées.... Ici, on découpe son programme en plusieurs réseaux (en général quand un ensemble de blocs n'est pas relié au reste, ou un réseau par sortie...).

# Chapitre IV: Application

station de distribution et station de tri

### 1-Le système MPS (Système de Production Modulaire)

C'est un système qui simule une mini usine : il s'agit d'une ligne de fabrication constituée de différentes cellules. Chaque cellule du processus de fabrication a une fonction bien définie (distribution, contrôle, usinage, manipulation, assemblage, stockage). En choisissant parmi une multitude de stations, on peut réaliser une application ou un processus répondant à nos besoins. La combinaison judicieuse de différentes stations permet ainsi de constituer une installation de fabrication.

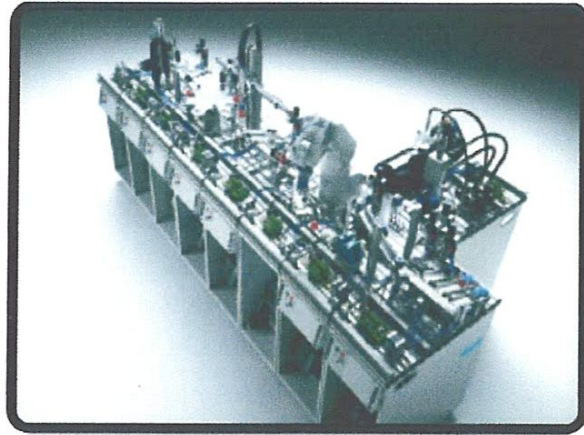
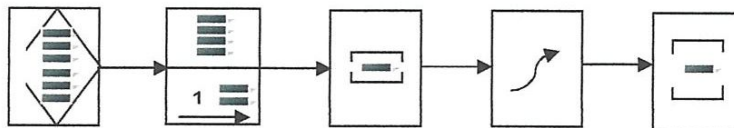


Fig.4.1 : Système MPS

### 2-Station de distribution :

#### 2.1- Fonction :

La station de distribution se charge de séparer les pièces du module de magasinage. Huit pièces au maximum se trouvent dans le module de magasinage à empilage. La détection de la pièce dans le magasin s'effectue à l'aide d'une cellule photoélectrique. Un vérin à double effet éjecte les pièces une à une. Le module de transport saisit chaque pièce éjectée au moyen d'une ventouse. Le bras pivotant du module de transport, commandé par un vérin oscillant, amène la pièce au point de transfert de la station suivante.



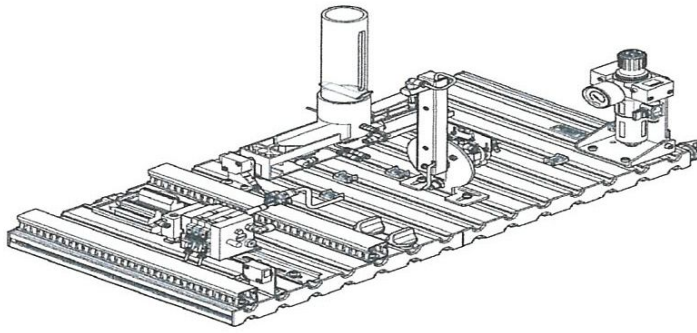


Fig.4.2: Station de distribution

## 2.2-Station de distribution contient

- le module de magasinage à empilage
- le module de transfert
- Plaque aluminium profilé
- Chariot
- Pupitre de commande
- Tableaux d'API

## 2.3-Objectifs de formation des projets étudiés

### Mécanique

- Structure mécanique d'une station

### Pneumatique

- Raccordement par tuyaux de composants pneumatiques
- Technique du vide
- Vérins pneumatiques linéaires et rotatifs

### Electrotechnique

- Câblage dans les règles de l'art de composants électriques capteurs
- Utilisation dans les règles de l'art de capteurs de fin de course

### Programmation et utilisation d'un API

Structure d'un programme d'API

- Programmation d'un élément de mode de fonctionnement
- Déroulement correct des réglages

## 2.4- Les principaux composants :

### 2.4.1-Le module de magasinage à empilage :

Le module de magasinage à empilage sépare les pièces du magasin, 8 pièces maximum se trouve dans le module de magasinage à empilage.

Un vérin à double effet pousse la plus basse pièce du magasin jusqu'à arrêt mécanique. Cette position est un point de transfert au prochain module (module de transfert). La pièce disponible dans le magasin est détectée par une cellule photoélectrique. La position du vérin d'éjection est électriquement détectée via des capteurs inductifs. La vitesse d'avancement et de recul du vérin d'éjection est infiniment réglable à l'aide des soupapes de commande à sens unique d'écoulement.

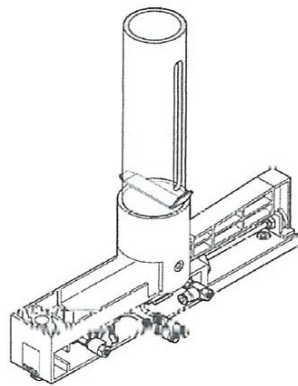


Fig.4.3 : Module de magasinage

### 2.4.2-Module de transfert:

Le module de transport est un dispositif de manipulation pneumatique. Les pièces sont élevées utilisant une ventouse et transférées par le bras pivotant. L'intervalle de pivotation est réglable entre  $0^\circ$  et  $180^\circ$  par un arrêt mécanique. La détection des fins de course s'effectue à l'aide de microcontacteurs.

Les positions de fin de course le bras pivotant doivent être ajusté selon la station aval choisie.

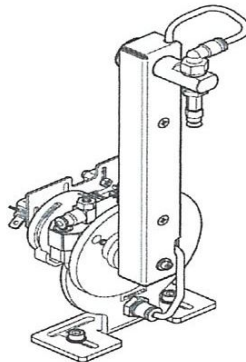


Fig.4.4 : Module de transfert

### 2.4.3-Régulateur de pression :

Filtre régulateur avec manomètre, distributeur de mise sous pression, raccords enfichables et coupleur, monté sur un support orientable.

Le filtre, comportant un purgeur d'eau, élimine les impuretés, les battitures, la rouille et l'eau de condensation contenues dans l'air comprimé. Le régulateur de pression assure la régulation de l'air comprimé d'alimentation à la pression de service réglée et compense les variations de pression. Une vis de purge du condensât est située au niveau de la cuve de filtre. Le distributeur de mise sous pression/d'arrêt assure la mise sous pression/à l'échappement de l'ensemble de la commande. L'actionnement du distributeur 3/2 s'effectue au moyen du bouton rotatif.

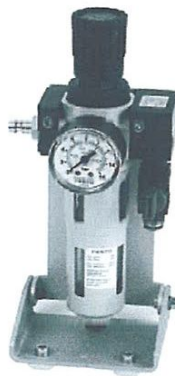


Fig.4.5 : Régulateur de pression

### 2.4.4- Vacuostat :

Vacuostat mécanique à seuil de commutation réglable et indication de l'état de commutation (LED).



Fig. 4.6 : Vacuostat

### 2.4.5-Pupitre de commande :

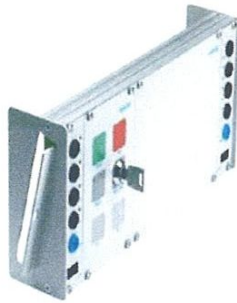


fig.4.7: Pupitre de commande

#### a-Pupitre de commande MPS® Sys Link :

Le pupitre de commande MPS® permet de commander aisément la station MPS®. Sys Link ou AS-Interface différentes interfaces assurent une grande souplesse d'utilisation. Entièrement assemblé, avec panneau de commande, panneaux de communication, panneau vide et châssis à connexion Sys Link.

Clavier à membrane avec : touche Start avec LED, touche Stop, touche Reset avec LED, 2 témoins utilisables à loisir. Prises bananes de 4 mm avec affichage d'état par LED en vue d'un couplage simple des E/S). A l'arrière, embases Sys Link et Sub-D pour la connexion à n'importe quel API.

#### b-Pupitre de commande MPS® AS-Interface :

Entièrement assemblé, avec panneau de commande AS-Interface, panneaux de Communication, panneau vide et châssis à connexion AS-Interface.

### 2.5-Description de la séquence :

#### -Condition initiale :

Le magasin est plein des pièces

#### -Position initiale :

- Vérin d'éjection en arrière.
- Le bras pivotant en position (magasin).
- Le vide est éteint.

#### -La séquence :

- Le bras pivotant tourne dans la position "station en aval" si les pièces sont Présents dans le magasin et le bouton marche est appuyé.
- Le vérin d'éjection recule et pousse les pièces hors du magasin.
- Le bras pivotant tourne dans la position "magasin"

- Le vide est allumé. Quand la pièce est levée, le vacuostat est activé.
- Le vérin d'éjection avance et relâche la pièce.
- Le bras pivotant tourne dans la position "station en aval".
- Le vide est coupé.
- Le bras pivotant tourne dans la position "magasin"

#### **2.6-Grafcet du système :**

Les figures 4.8.a et 4.8.b illustrent les Grafquets élaborés pour cette station et le programme correspondant en langage contact ainsi que la table des mnémoniques sont donnés dans l'annexe.



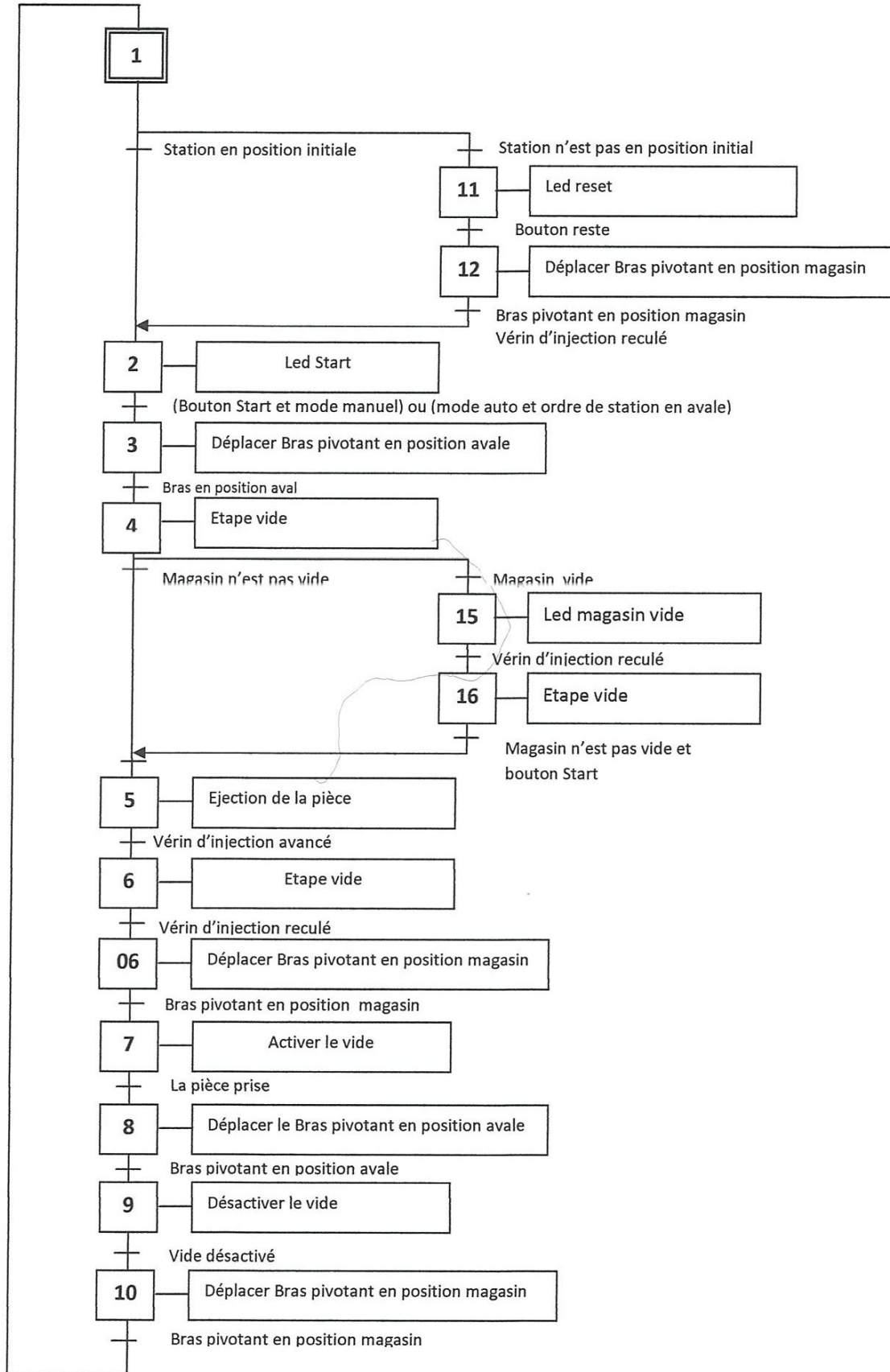


Fig.4.8.a : Grafcet niveau 1 de la station de distribution

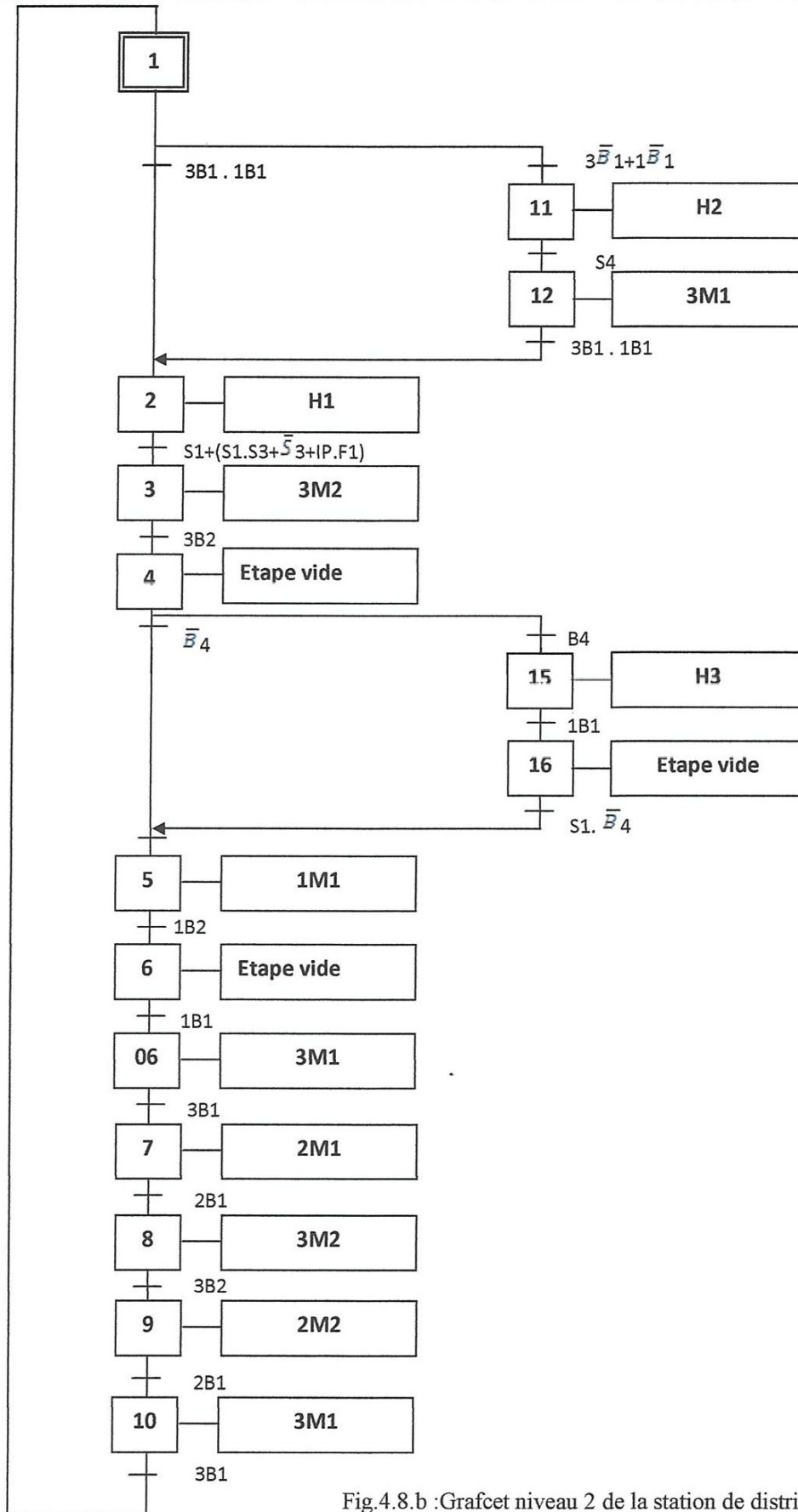


Fig.4.8.b :Grafcet niveau 2 de la station de distribution

### 3-Station de tri

#### 3.1- Fonction :

La station de tri reconnaît différentes propriétés des pièces insérées. A l'aide d'un capteur optique et d'un capteur capacitif, elle est capable de différencier des pièces. Une barrière photoélectrique surveille si la zone de travail est libre avant de procéder à l'élévation de la pièce à l'aide d'un vérin linéaire. Un capteur analogique détermine la hauteur de la pièce. Un vérin linéaire achemine les pièces conformes vers la station voisine en empruntant la glissière à coussin d'air supérieure. Les pièces non conformes sont rebutées via la glissière à coussin d'air inférieure.

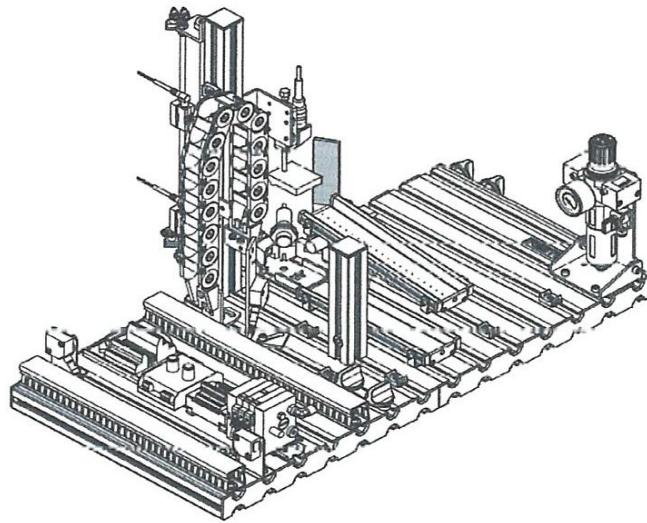


Fig.4.9 : Station de tri

#### 3.2- Thème central : Capteurs

La station de tri réunit toutes les formes de base des capteurs industriels et en démontre la mise en œuvre : détecteurs de proximité optiques et capacitifs et barrière photoélectrique. S'y ajoutent différents détecteurs de fin de course (inductifs, magnétiques) pour les vérins

#### 3.3-Capteur inductif :

Le capteur inductif est fourni complet avec support de fixation sur le rail de sécurité du convoyeur ou d'une glissière.

- Distance de commutation nominale : 4 mm
- Tension d'alimentation : 24 V DC
- Sortie de commutation : PNP, NO
- Câble de raccordement : 3 pole



Fig.4.10 : Capteur inductif

### 3.4- Barrières photoélectrique reflex:

Le capteur optique et le réflecteur sont livrés complets, avec les équerres de fixation sur un profilé ou une plaque profilé.



Fig.4.11 : Barrières photo –électrique reflex

### 3.5-Option : Traitement de valeurs analogiques

Un capteur analogique détecte la hauteur de la pièce, amenée en contact avec le dispositif de mesure par le vérin linéaire. Un comparateur analyse le signal du capteur et le transmet sous forme d'information numérique. Le signal analogique est en outre appliqué à une borne distincte pour connexion au boîtier de simulation ou à un API à module analogique.

### 3.6-Objectifs de formation des projets étudiés :

Mécanique :

- Architecture mécanique d'une station pneumatique :
- Utilisation de vérins sans tige électrotechnique :
- Câblage dans les règles de l'art de composants électriques

Capteurs :

- Mode de fonctionnement et domaines d'utilisation de capteurs optiques et capacitifs à commutation numérique

- Mode de fonctionnement et domaines d'utilisation de capteurs analogiques, en prenant pour exemple un système analogique de mesure de course

API:

- Programmation et utilisation d'un API
- Traitement de signaux analogiques

Miser en service :

- Mise en service de l'ensemble du montage

Dépannage :

- Dépannage systématique sur une installation de production

### 3.7-Les principaux composants :

#### 3.7.1-Module de détection :

Le module de détection se compose de deux capteurs différents et d'une équerre de fixation.

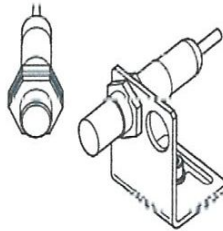


Fig.4.12 :Module de detection

#### 3.7.2-Module de levage :

Levage d'une pièce au moyen d'un vérin sans tige de piston. Dans cette position, la pièce peut faire l'objet d'un contrôle à l'aide du module de mesure. Elle set en suite éjective par un second veina.

- Hauteur : 290 mm
- Large : 220 mm
- Profonde : 130 mm

Le guide de câblage permet la pose en toute sécurité des câbles de raccordement du module de levage.

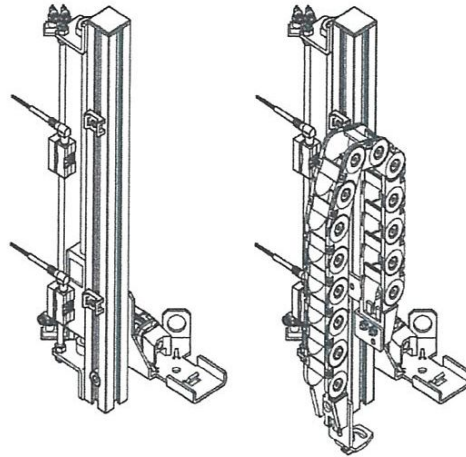


Fig.4.13 : Module de levage

### 3.7.3- Module de mesure :

Le module de mesure permet de mesurer la hauteur d'une pièce à l'aide d'un palpeur de course linéaire. Il se fixe directement sur le module de levage. Le palpeur de course linéaire est relié à un comparateur. Le comparateur permet une exploitation aisée du signal de mesure. Palpeur de course (potentiomètre à plastique conducteur) :

- Game de mesure : 25 mm
- Résistance connecté: 1 k.

Comparateur:

- Distance de commutation nominale : jusqu'à 400 mm maximum (réglable)
- Tension d'alimentation : 24 V DC
- Sorties de commutation : 3, PNP, NO
- Entrée analogique : 0 – 10 V

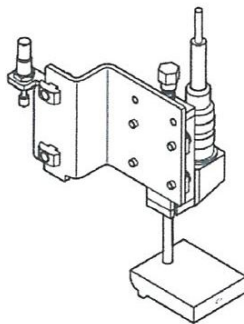


Fig.4.14 : Module de mesure

### 3.7.4-Module de glissière pneumatique :

La glissière est fixée sur un profilé universel. Ses propriétés s'adaptent à l'aide d'un limiteur de débit situé sous la glissière.

- Utilisation : Alimentation matière en amont d'un convoyeur ou d'une station, transfert vers une station
- Longueur : 220 mm
- Pression de service : 600 kPa (6 bar)
- Réglage : Via limiteur de débit sur la face inférieure de la glissière



Fig 4 15 : Module de glissière pneumatique

### 3.7.5-Compateur analogique :

Le compateur permet de convertir le signal analogique du déplacement linéaire du capteur en un signal numérique. Il ya trois sorties numériques :

- 1 : valeur mesurée plus petite que le seuil 1
- 2 : valeur mesurée entre le seuil 1 et le seuil 2
- 3 : valeur mesurée plus grande que le seuil 2

Les seuils sont réglables à l'aide de deux potentiomètres LEVEL1 et LEVEL2.une LED permet la visualisation de la sortie active.

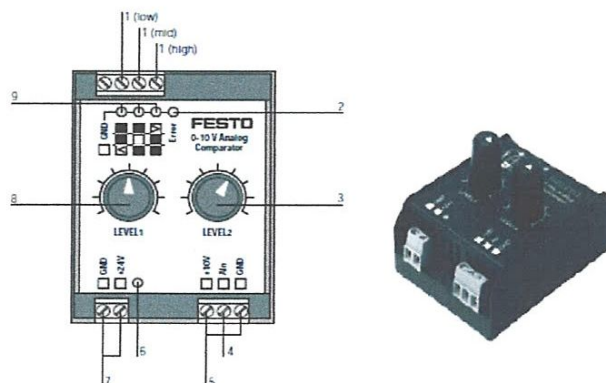


Fig.4.16 : Compateur analogique

- 1-Sorties numériques
- 2-Affichage d'erreur de sortie
- 3-Potentiomètre LEVEL2 (seuil supérieur)
- 4-Entrée analogique
- 5-Référence de tension de capteur de déplacement linéaire.
- 6-Affichage de tension externe d'opération
- 7-Tension externe d'opération
- 8-Potentiomètre LEVEL2 (seuil inférieur)
- 9- Affichage d'état de commutation des sorties

### **3.8-Grafcet du système :**

Les figures 4.17.a et 4.17.b illustrent les Grafkets élaborés pour cette station et le programme correspondant en langage contact ainsi que la table des mnémoniques sont donnés dans l'annexe.



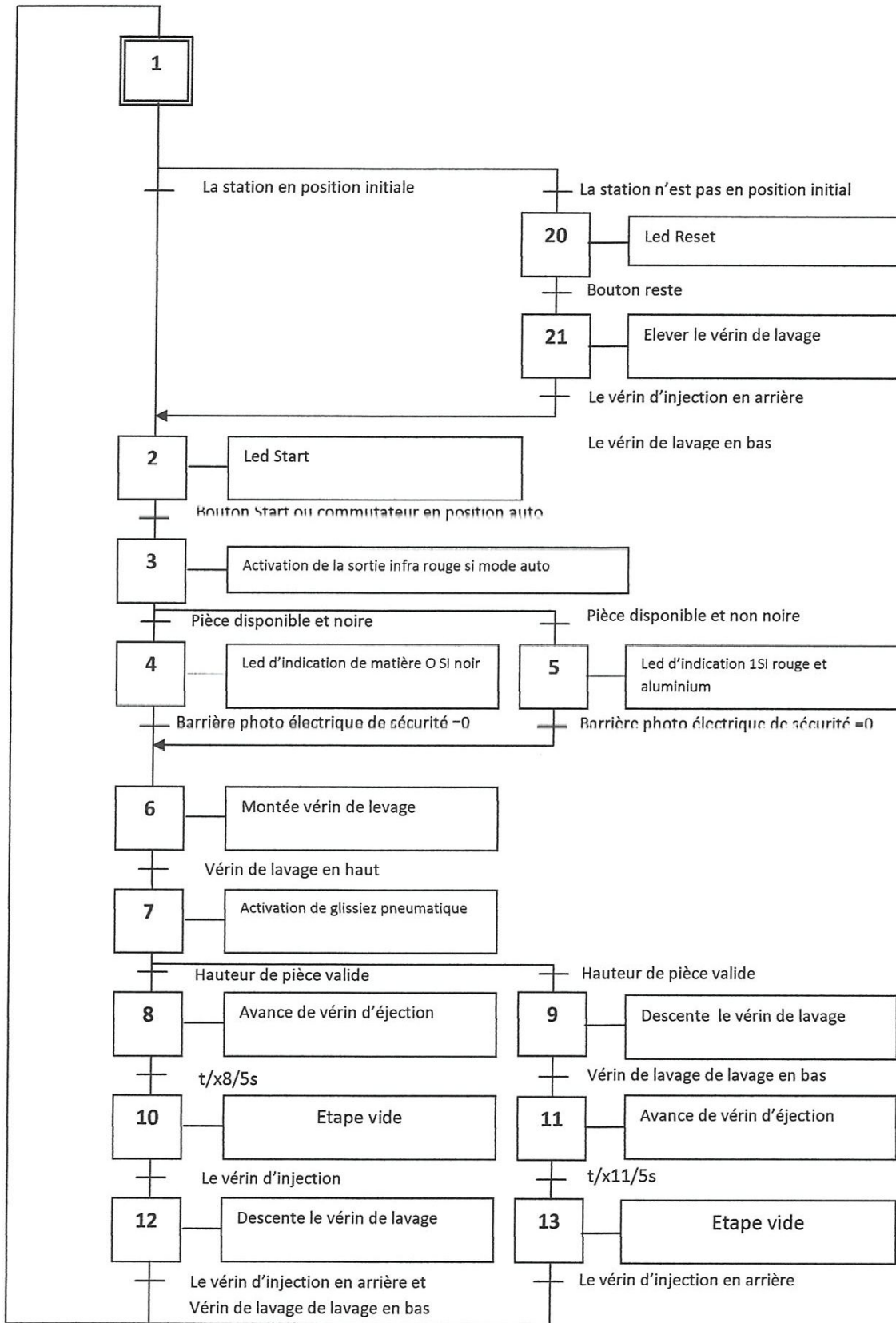


Fig.4.17.a: Grafcet niveau 1 de la station de tri

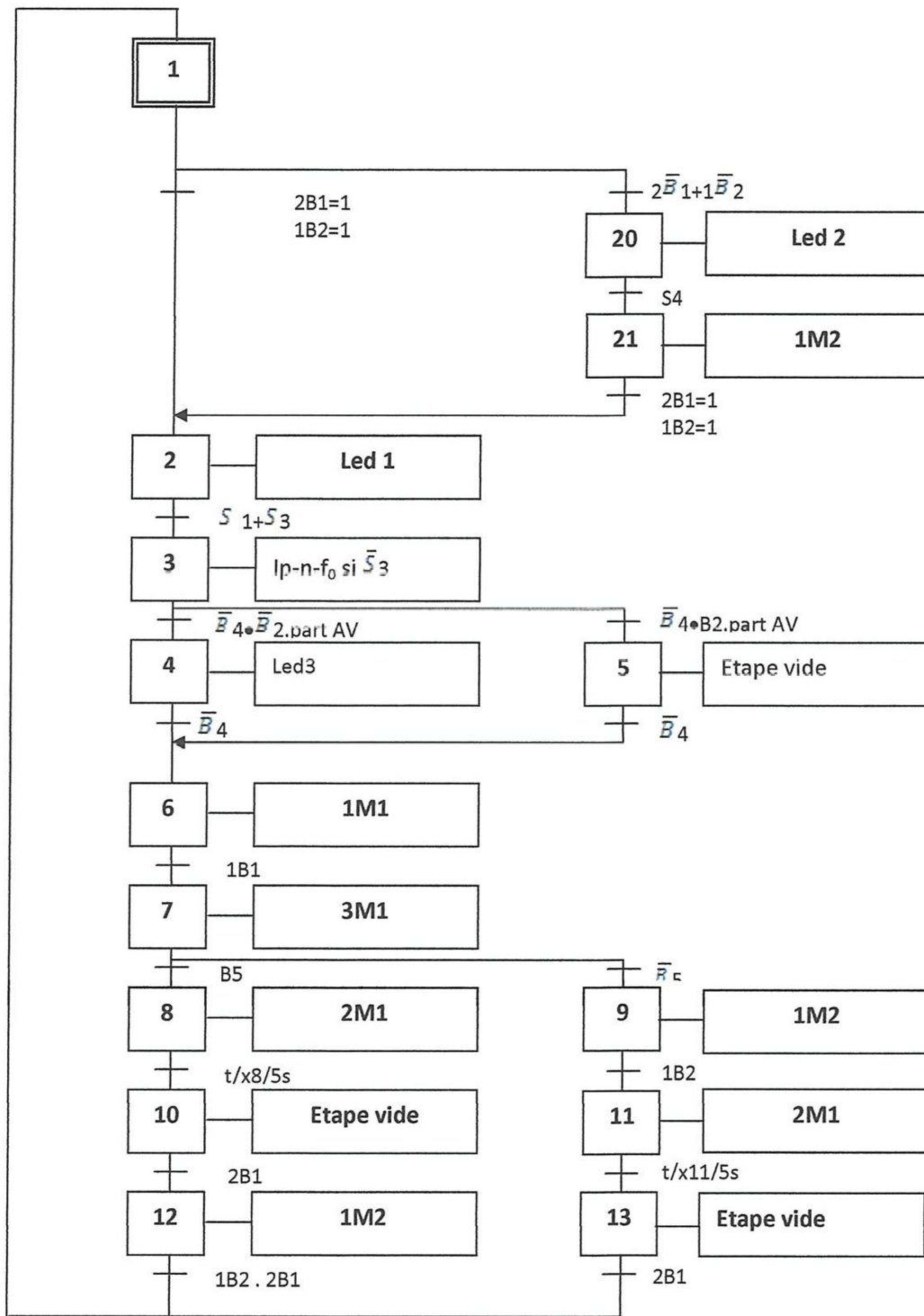


Fig.4.17.b:Grafctet niveau 2 de la station de distribution

# Annexe

**Table des mnémoniques de la station de distribution**

**Programme de la station de distribution**

**Propriétés de la table des mnémoniques**

Nom : Mnémoniques  
 Commentaire :  
 Date de création : 10/05/2011 10:36:17  
 Dernière modification : 29/05/2011 10:12:42  
 Dernier filtre sélectionné : Tous les mnémoniques  
 Nombre de mnémoniques : 35/35  
 Dernier tri : Opérande ordre décroissant

Etat	Mnémonique	Opérande	Type de données	Commentaire
	x06	M 1.7	BOOL	
	x16	M 1.6	BOOL	
	x15	M 1.5	BOOL	
	x12	M 1.4	BOOL	
	x11	M 1.3	BOOL	
	x10	M 1.2	BOOL	
	x9	M 1.1	BOOL	
	x8	M 1.0	BOOL	
	x7	M 0.7	BOOL	
	x6	M 0.6	BOOL	
	x5	M 0.5	BOOL	
	x4	M 0.4	BOOL	
	x3	M 0.3	BOOL	
	x2	M 0.2	BOOL	
	x1	M 0.1	BOOL	
	s4	E 1.3	BOOL	
	s3	E 1.2	BOOL	
	s2	E 1.1	BOOL	
	s1	E 1.0	BOOL	
	ip-f1	E 0.7	BOOL	
	b4	E 0.6	BOOL	
	3b2	E 0.5	BOOL	
	3b1	E 0.4	BOOL	
	2b1	E 0.3	BOOL	
	1b2	E 0.2	BOOL	
	1b1	E 0.1	BOOL	
	h3	A 1.2	BOOL	
	h2	A 1.1	BOOL	
	h1	A 1.0	BOOL	
	3m2	A 0.4	BOOL	
	3m1	A 0.3	BOOL	
	2m2	A 0.2	BOOL	
	2m1	A 0.1	BOOL	
	1m1	A 0.0	BOOL	
X			BOOL	

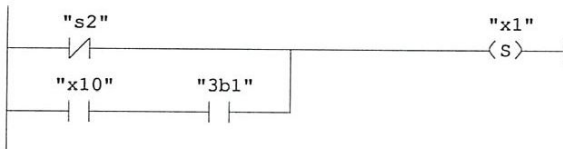
**FC1 - <offline>**

""  
 Nom :                      Famille :  
 Auteur :                    Version : 0.1  
                               Version de bloc : 2  
 Horodatage Code :        29/05/2011 15:57:40  
                               Interface :    10/05/2011 10:37:01  
 Longueur (bloc/code /données locales) : 00444 00276 00000

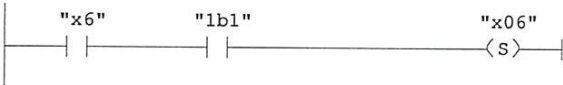
Nom	Type de données	Adresse	Valeur initiale	Commentaire
IN		0.0		
OUT		0.0		
IN_OUT		0.0		
TEMP		0.0		
RETURN		0.0		
RET_VAL		0.0		

Bloc : FC1

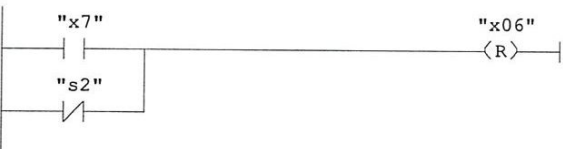
Réseau : 1



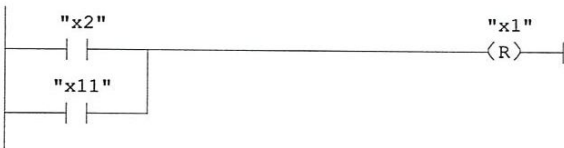
Réseau : 2



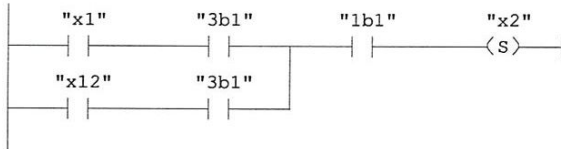
Réseau : 3



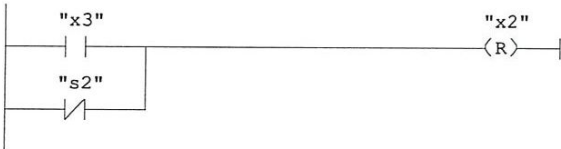
Réseau : 4



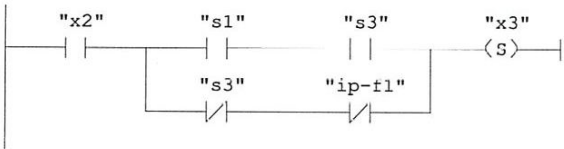
Réseau : 5



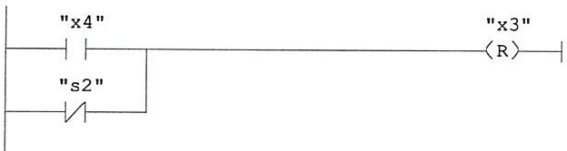
Réseau : 6



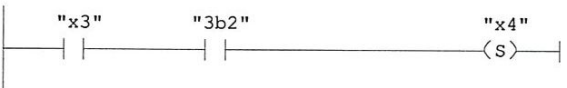
Réseau : 7



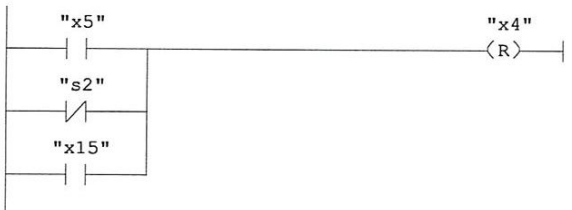
Réseau : 8



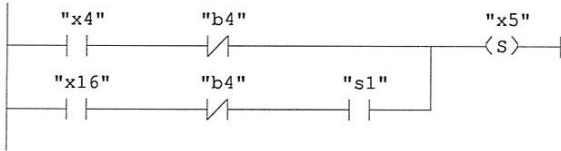
Réseau : 9



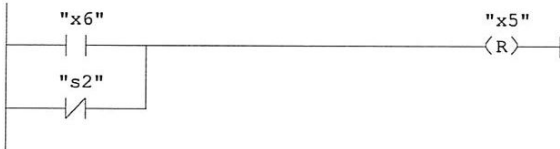
Réseau : 10



Réseau : 11



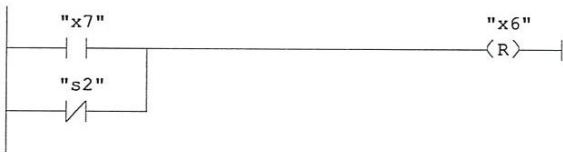
Réseau : 12



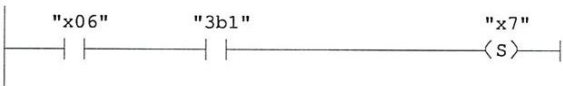
Réseau : 13



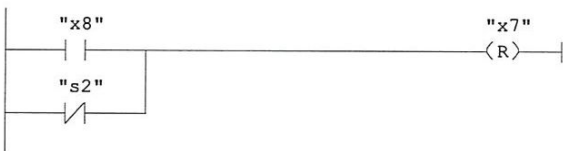
Réseau : 14



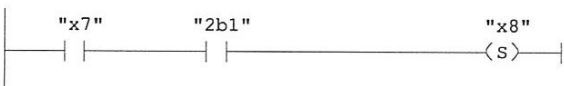
Réseau : 15



Réseau : 16

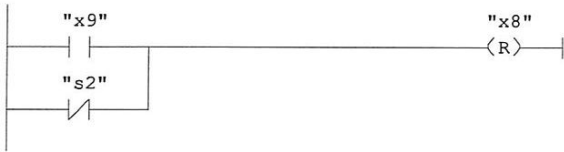


Réseau : 17

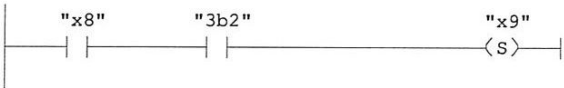




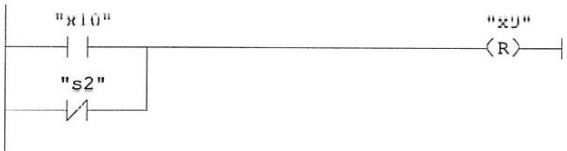
Réseau : 18



Réseau : 19



Réseau : 20



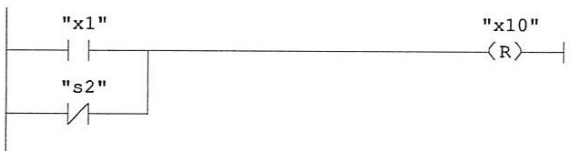
Réseau : 21



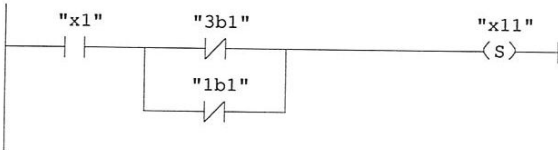
Réseau : 22



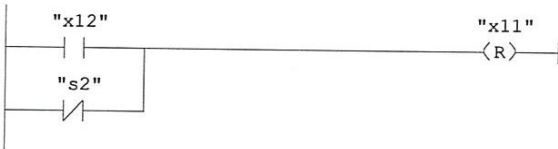
Réseau : 23



Réseau : 24



Réseau : 25



Réseau : 26



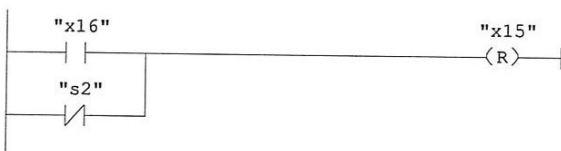
Réseau : 27



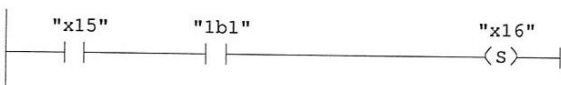
Réseau : 28



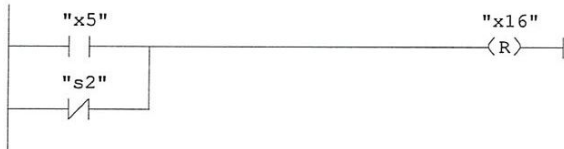
Réseau : 29



Réseau : 30



Réseau : 31



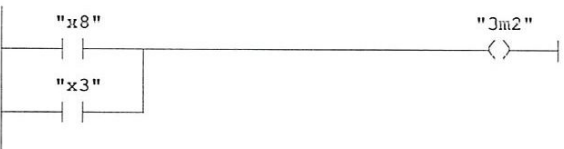
Réseau : 32



Réseau : 33



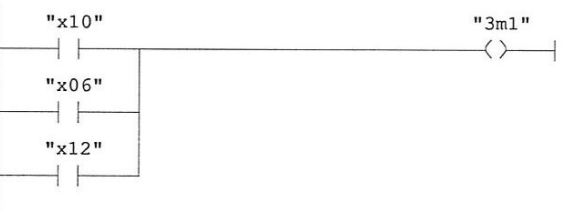
Réseau : 34



Réseau : 35



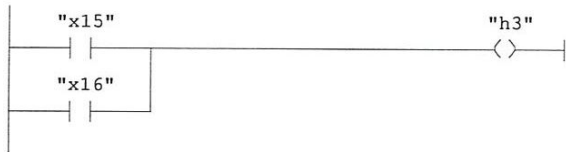
Réseau : 36



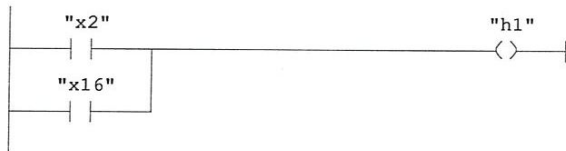
Réseau : 37



Réseau : 38



Réseau : 39



**Table des mnémoniques de la station de tri**

**Programme de la station de tri**

**Propriétés de la table des mnémoniques**

Nom : Mnémoniques  
 Commentaire :  
 Date de création : 15/05/2011 11:14:54  
 Dernière modification : 22/05/2011 11:32:55  
 Dernier filtre sélectionné : Tous les mnémoniques  
 Nombre de mnémoniques : 37/37  
 Dernier tri : Opérande ordre croissant

Etat	Mnémonique	Opérande	Type de données	Commentaire
	1m2	A 0.0	BOOL	
	1m1	A 0.1	BOOL	
	2m1	A 0.2	BOOL	
	3m1	A 0.3	BOOL	
	ip-n-fo	A 0.7	BOOL	
	led start	A 1.0	BOOL	
	led2	A 1.1	BOOL	
	led3	A 1.3	BOOL	
	lh-mat	A 2.6	BOOL	
	part-av	E 0.0	BOOL	
	b2	E 0.1	BOOL	
	b4	E 0.2	BOOL	
	b5	E 0.3	BOOL	
	1b1	E 0.4	BOOL	
	1b2	E 0.5	BOOL	
	2b1	E 0.6	BOOL	
	ip-fi	E 0.7	BOOL	
	s1	E 1.0	BOOL	
	s2	E 1.1	BOOL	
	s3	E 1.2	BOOL	
	s4	E 1.3	BOOL	
	x0	M 0.0	BOOL	
	x1	M 0.1	BOOL	
	x2	M 0.2	BOOL	
	x3	M 0.3	BOOL	
	x4	M 0.4	BOOL	
	x5	M 0.5	BOOL	
	x6	M 0.6	BOOL	
	x7	M 0.7	BOOL	
	x8	M 1.0	BOOL	
	x9	M 1.1	BOOL	
	x10	M 1.2	BOOL	
	x11	M 1.3	BOOL	
	x12	M 1.4	BOOL	
	x13	M 1.5	BOOL	
	x20	M 1.6	BOOL	
	x21	M 1.7	BOOL	

**FC1 - <offline>**

```

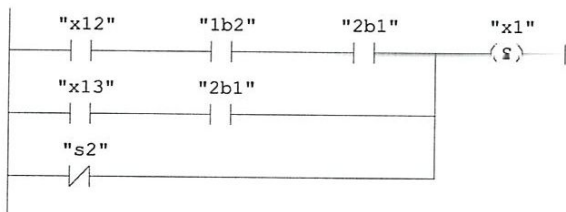
""
Nom :                               Famille :
Auteur :                             Version : 0.1
                                       Version de bloc : 2
Horodatage Code :                   29/05/2011 15:50:32
                                       Interface : 15/05/2011 11:27:13
Longueur (bloc/code /données locales) : 00466 00292 00000

```

Nom	Type de données	Adresse	Valeur initiale	Commentaire
IN		0.0		
OUT		0.0		
IN_OUT		0.0		
TEMP		0.0		
RETURN		0.0		
RET_VAL		0.0		

Bloc : FC1

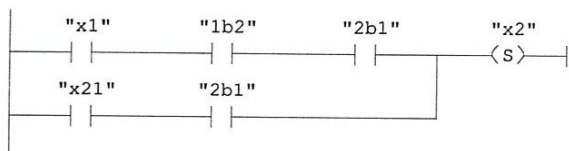
Réseau : 1



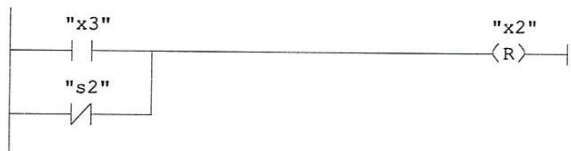
Réseau : 2



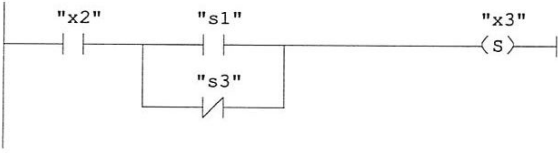
Réseau : 3



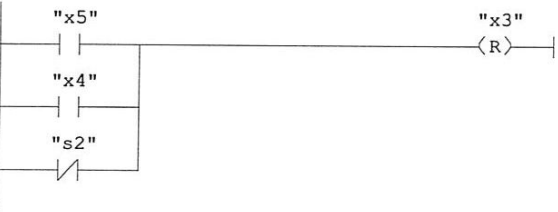
Réseau : 4



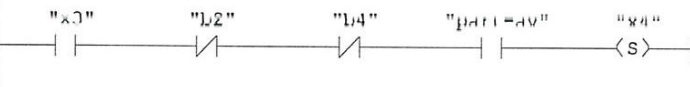
Réseau : 5



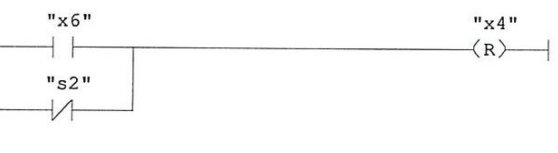
Réseau : 6



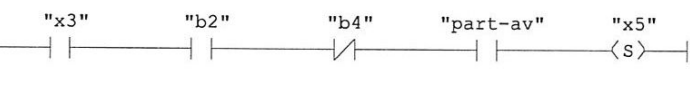
Réseau : 7



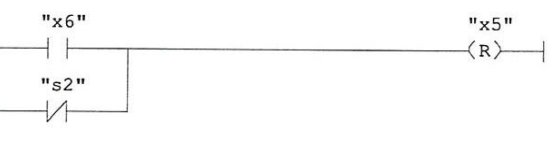
Réseau : 8



Réseau : 9

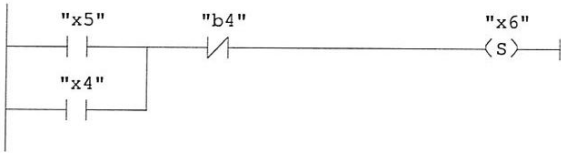


Réseau : 10

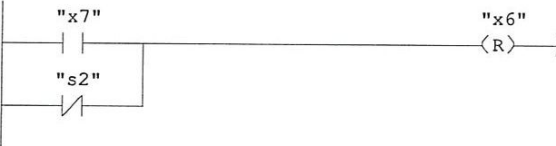




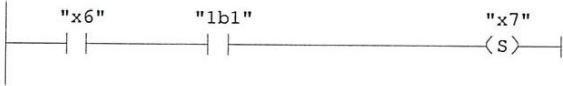
Réseau : 11



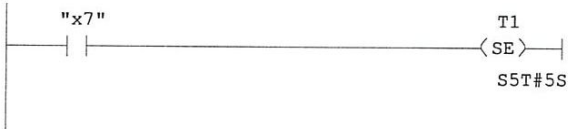
Réseau : 12



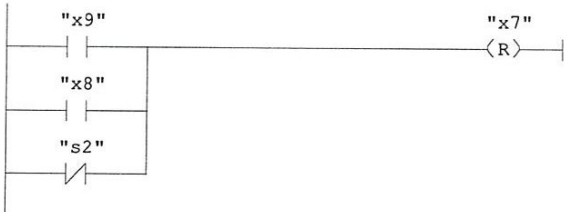
Réseau : 13



Réseau : 14



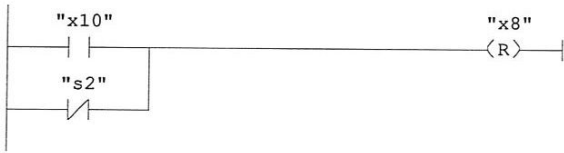
Réseau : 15



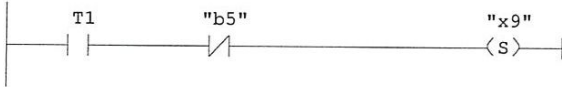
Réseau : 16



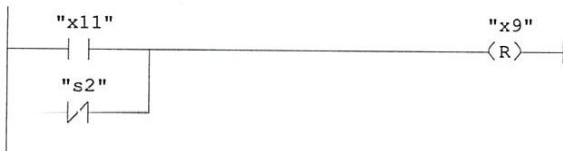
Réseau : 17



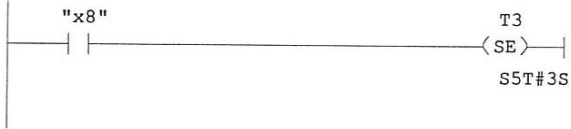
Réseau : 18



Réseau : 19



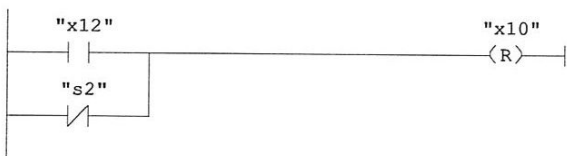
Réseau : 20



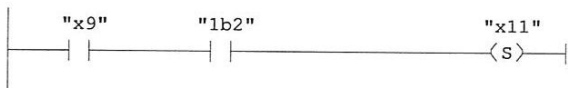
Réseau : 21



Réseau : 22



Réseau : 23



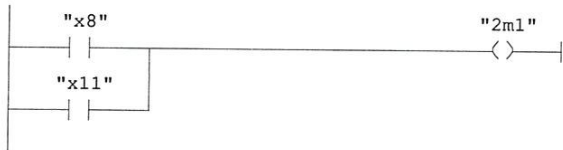
Réseau : 37



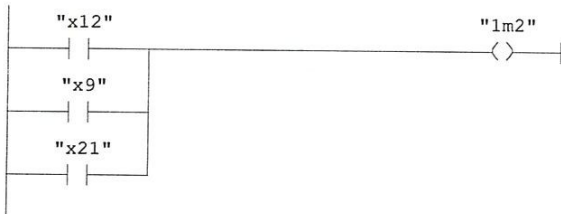
Réseau : 38



Réseau : 39



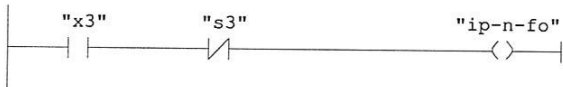
Réseau : 40



Réseau : 41



Réseau : 42



**Conclusion**