

**Introduction :**

Les méthodes de fabrication actuelles subissent des mutations importantes et font de plus en plus appel à l'automatisation. Cette évolution est caractérisée par un développement spectaculaire des systèmes programmés.

Les matériels mis en œuvre sont de plus en plus sophistiqués. Les machines automatiques actuelles utilisent des technologies très différentes hydraulique, pneumatique, électrique et électronique. Il est fondamental que l'ingénieur acquière au cours de sa formation une compétence nécessaire à la conception et la maintenance de ces divers matériels [2].

**I-1- L'automatisation :**

L'automatisation d'un procédé (c'est à dire une machine, un ensemble de machines ou plus généralement un équipement industriel) consiste à en assurer la conduite par un dispositif technologique. Une « machine » ou un « système » est dit automatisé lorsque le processus qui permet de passer d'une situation initiale à une situation finale se fait sans intervention humaine et que ce comportement est répétitif chaque fois que les conditions qui caractérisent la situation initiale sont remplies [1].

**I-2- Les objectifs de l'automatisation**

La compétition économique entraîne les industriels à vivre en permanence dans un esprit de concurrence, qui oblige à toujours améliorer les performances en termes de quantité et surtout de qualité (d'où le maître mot de l'économie : rapport qualité/prix). L'automatisation des processus industriels a pour finalité de réaliser ces vœux. Ses objectifs principaux au nombre de quatre sont complémentaires et liés. Ils peuvent s'énoncer ainsi :

**a- Produire à qualité constante :**

Contrairement à l'être humain, il est clair que la machine n'est pas sujette à la fatigue de fin de journée, par conséquent la qualité des produits sortant des chaînes est généralement la même [3].

**b- Fournir les quantités nécessaires :**

Cette notion fait référence à l'adaptativité, c'est-à-dire pouvoir adapter l'offre à la demande, l'objectif est de produire les quantités nécessaires à un instant donné, de façon à tendre vers la notion de stock zéro, pour pouvoir adapter l'offre à la demande, cela se fait rapidement et efficacement dans un environnement automatisé (arrêter par exemple une chaîne de fabrication ayant une faible demande, ou au contraire mettre en route d'autres pour répondre à une forte demande) [3].

**c- Augmenter la productivité :**

Il s'agit donc d'augmenter le rendement. Pour se faire l'automatisation a consisté à remplacer une grande partie des opérateurs humains par des machines, qui ont des cadences de travail élevées, ne connaissent ni les pauses café ni les congés payés [3].

**d- Améliorer les conditions de travail :**

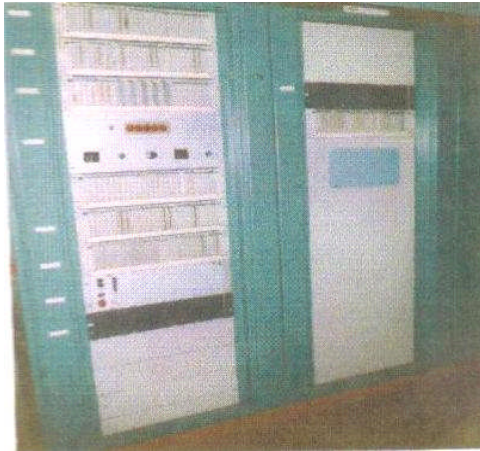
Il s'agit d'une part de remplacer l'homme par la machine pour les tâches pénibles ou qu'il ne peut pas faire (pour l'effectuer ailleurs où il est censé faire un travail plus noble) , d'autre part d'augmenter les possibilités de réaliser les objectifs "a", "b", "c". En effet un employé qui mange bien, est bien soigné, et a bonnes conditions de vie et de travail, n'est pas souvent malade, n'est pas fatigué, a peu d'absentéisme, et devient donc plus rentable économiquement [3].

**I-3- Technologies des automatismes :**

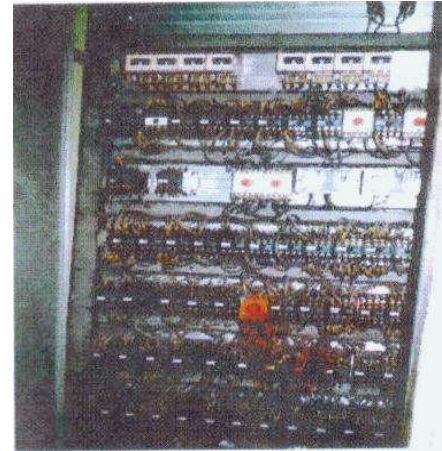
Nous disposons de nombreux outils technologiques pour réaliser l'organe de commande de notre système que l'on regroupe habituellement en deux catégories fondamentales : les solutions câblées (fig I.2) et les solutions programmées (fig I.1). Le tableau (I.1) illustre une comparaison entre ces deux solutions [5]:

Logique Câblée	Logique Programmable
Toutes les fonctions technologiques (électromagnétiques, électroniques, pneumatiques...), sont câblées entre elles à partir d'un diagramme à caractère booléen.	Le programme est établi à partir d'un langage compris puis rangé suivant un ordre dans les mémoires. Ce programme est divisé en une succession d'ordre. Les fonctions logiques ne sont pas dépendantes du câblage mais sont reliées entre elles par un câblage binaire.
Faible capacité	Grâce à l'API, l'adaptation à une nouvelle fabrication est rapide.
Gros frais de reconversion des installations.	Moins coûteuse.
Un automate à base d'une logique câblée est prévu pour des productions industrielles de mode répétitif.	L'automate étant ; modifiable à volonté, il est programmable.
L'encombrement, le poids et le volume des composants sont des contraintes majeurs lorsqu'il s'agit de maîtriser des applications plus au moins complexes.	La flexibilité, la souplesse vis-à-vis de la mise au point de commande et de l'évolution de celles-ci et la performance font de la logique programmée, une solution préférable.
Connexion figée, valable pour une seule application	Toutes les connexions sont en soft (logique).
Longueur de câble utilisée très importante	Gain en longueur de câble électrique.
Mise au point difficile.	Diagnostic rapide : (recherche d'éventuelles pannes).
Prix de revient (installation) coûteux.	Gain important en temps : (amélioration du taux de marche).

Tableau (I.1) Comparaison entre la solution câblée et la solution programmée.



**Fig-I.1- Armoire de commande à logique programmable SIEMENS**



**Fig-I.2- Armoire de commande à logique câblée**

#### **I-4- Les avantages des automates programmables :**

Pour la réalisation d'un système d'automatisme, il faut considérer les avantages apportés par l'automate à tous les stades du projet [7].

##### **I-4-1- Pré étude, Devis :**

Le chiffrage d'une solution automate est particulièrement simple. Il suffit, en effet, de compter le nombre d'entrées et de sorties qui définissent la taille du système [7].

##### **I-4-2- Etude du système :**

L'utilisation d'un automate permet de réduire au minimum le temps d'étude de la partie électrique d'un système. Le concepteur peut en effet attendre que le processus soit bien défini pour réaliser le programme de l'automate au dernier moment. De plus pas de plans d'implantation ni de câblage électrique compliqués [7].

##### **I-4-3- Construction du système :**

Le montage-câblage est réduit au minimum avec un automate. Il suffit de raccorder seulement les entrées et les sorties. Dans ce cas nous n'avons pas de tôlerie spéciale [7].

**I-4-4- Mise en route :**

La mise en route d'un système comporte toujours certains imprévus. Y remédier est simple avec l'automate : aucune modification de câblage, de plans, pour adjonction de dernière minute: quelques modifications de programme seulement [7].

**I-4-5- Fiabilité et maintenance :**

Le système d'automate est compact ou modulaire. Le changement d'un élément demande quelques minutes.

La fiabilité du système est accrue par rapport aux systèmes câblés traditionnels (moins de composants, moins de connexions) [7].

**I-4-6- Evolution et modification :**

La modification de fonction ou l'adjonction de fonctions supplémentaires n'oblige pas une intervention de câblage. Seul le programme est à modifier et cela peut se faire sans arrêter l'ancien système [7].

**I-5- Structure d'un système automatisé :**

On distingue, dans tout système automatisé, la machine ou l'installation (partie de puissance) et la partie commande constituée par l'appareillage d'automatisme. Cette dernière partie, est assurée par des constituants répondant schématiquement à cinq fonctions de base : [6]

- L'acquisition des données.
- Traitement de données.
- Commande de puissance (prés actionneur).
- Le dialogue homme machine.
- La protection du système.

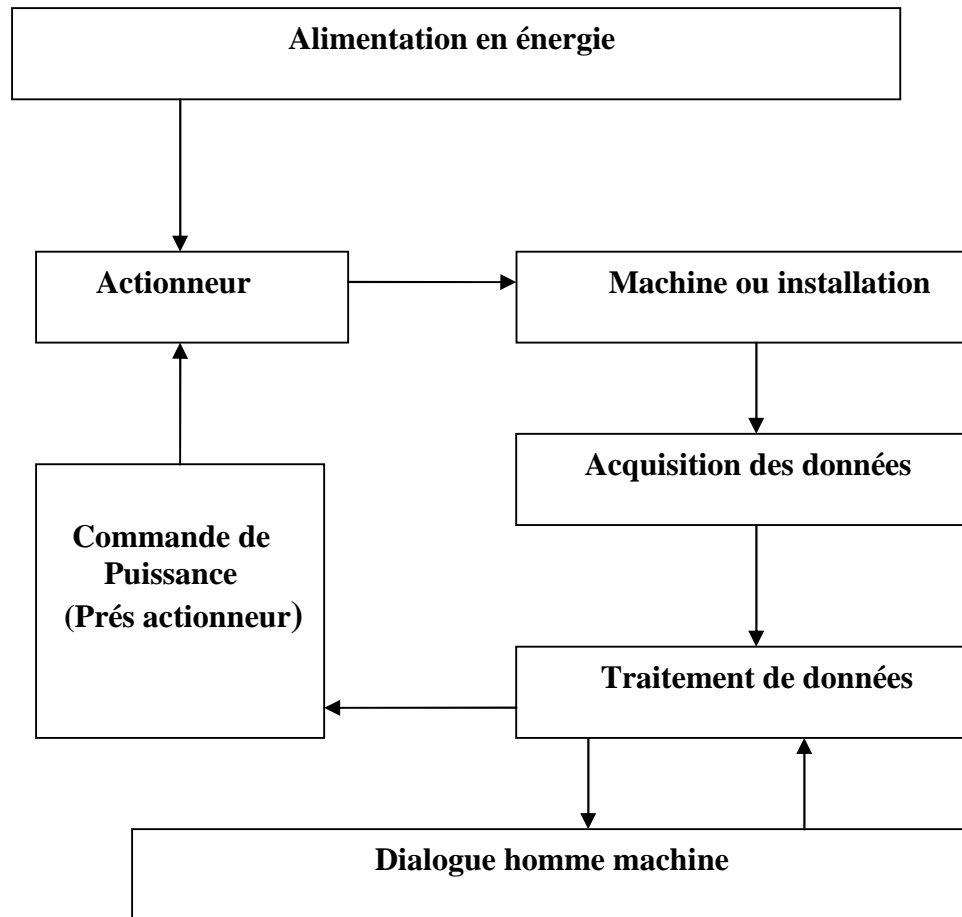


Fig –I.3-Structure d'un système automatisé.

## I-6- Les Automates Programmables Industriels :

### I-6-1- Définition :

Un automate programmable industriel (API) est un dispositif électronique capable de piloter (assurer la commande) un processus industriel, dans des environnements très diversifiés : mécanique, fonderie, agroalimentaire, électronique etc...

Il est programmable, donc possède une mémoire pour enregistrer des programmes, qui seront écrits dans un langage propre à chaque API. Sa programmation ne nécessite pas d'être informaticien ou la connaissance d'un langage informatique [3].

### I-6-2- Architecture de l'API :

Un API se compose de trois parties : l'unité centrale, les interfaces et l'alimentation.

### I-6-2-1- L'unité centrale :

C'est l'ensemble des dispositifs nécessaire en fonctionnement logique interne de l'API, c'est le cœur de l'automate [5].

a- Architecteur de l'unité centrale (UC) : l'unité centrale comporte la zone mémoire et le processeur [5].

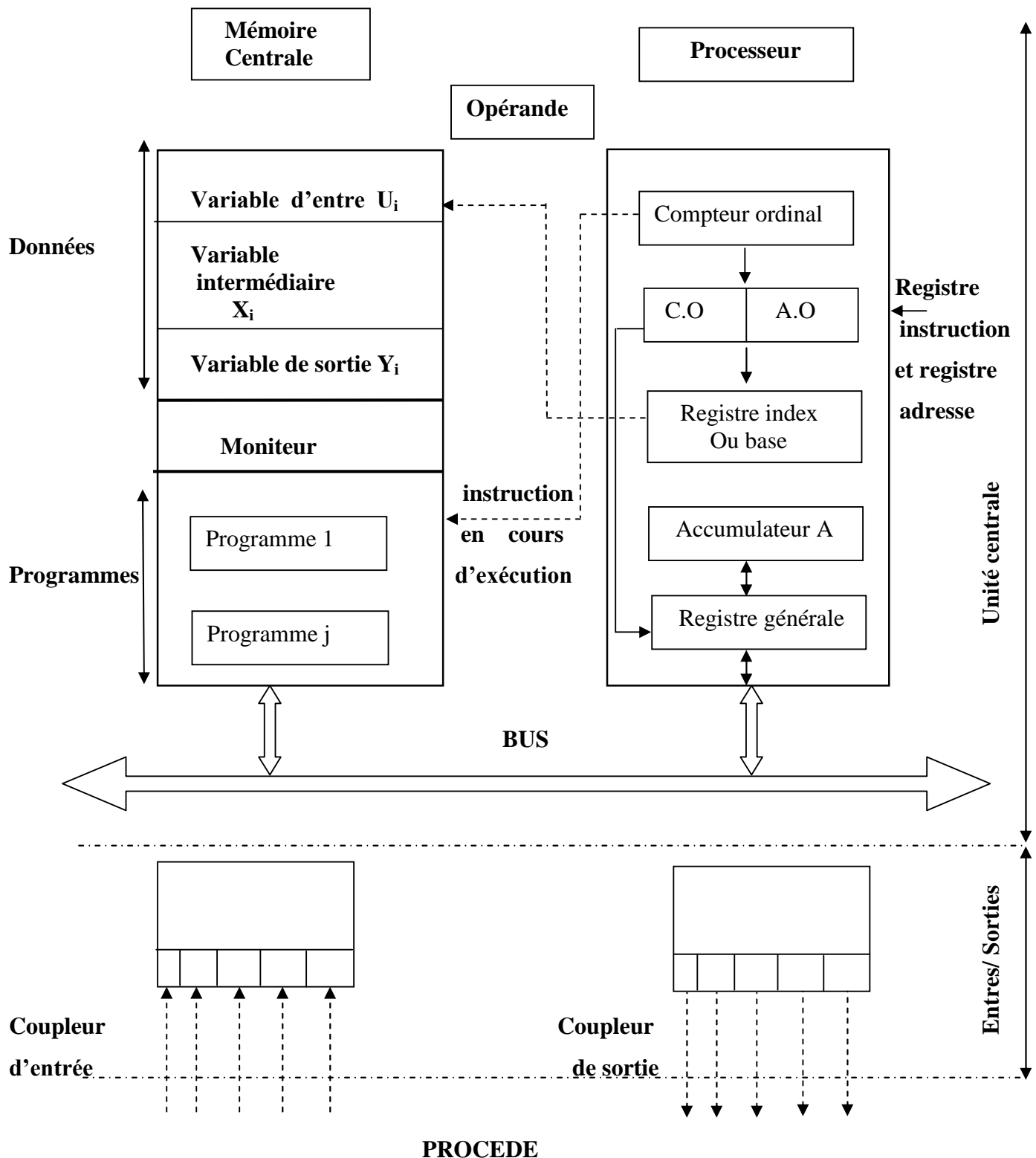


Fig –I.4- Structure de l'Unité Centrale d'un API.

**b- Le processeur :**

C'est le cerveau de l'automate : en plus de son rôle de calculateur. Il gère et coordonne les relations entre les zones mémoires et les interfaces. Il procède à [3]:

- La lecture des informations d'entrée.
- L'interprétation de ces informations et l'exécution des instructions du programme contenues en mémoire.
- La commande des sorties.

**c- La zone mémoire :**

Dans un API on rencontre différents types de mémoires, ayant des rôles différents en fonction de leur technologie [3].

**RAM (Random Access Memory) :**

C'est une mémoire vive dite volatile car son contenu « s'efface » dès qu'elle n'est plus alimentée. C'est la zone de travail du processeur (et de l'utilisateur éventuellement) [3].

**ROM (Read Only Memory) :**

Appelée mémoire morte, seule la lecture y est possible. On y stocke généralement les programmes de gestion de l'automate appelés « système d'exploitation ». Son contenu ne s'efface pas en cas de coupure de l'alimentation [3].

**EPROM (Eraseable programable ROM) :**

C'est une ROM dont le contenu peut être effacé électriquement. Elle est généralement utilisée pendant la phase de développement des programmes. A cause de sa souplesse d'utilisation par rapport à la ROM, elle tend de plus en plus à la remplacer pour stocker les programmes de l'utilisateur pendant la phase d'exploitation. Elle se présente sous forme de cartouche à enficher directement sur le support qui lui est réservé dans l'automate [3].

**d- Fonction de la zone mémoire :**

Fonctionnellement la mémoire est organisée pour [3]:

- Recevoir les informations issues des capteurs d'entrée et les mémoriser.
- Recevoir les informations générées par le processeur, qu'elles soient de type variables internes (résultats de calculs intermédiaires temporaires logiques ou numériques), ou de type variables de sortie destinées à la commande des pré actionneurs.



- Conserver les programmes à exécuter écrits dans le langage propre à l'automate.
- Conserver le système d'exploitation et les données associées.

### I-6-2-2-Les interfaces :

L'API dispose d'un module d'interface pour chaque entrée (capteur) et chaque sortie (préactionneur), dont l'état est indiqué par une LED. A chaque module est associée une adresse. L'ensemble des modules d'entrée constitue l'interface (ou les interfaces) d'entrée, et ceux de sortie l'interface de sortie [3].

#### a- Interfaces d'entrée :

Elles assurent principalement trois fonctions : la mise en forme du signal d'entrée pour produire un signal logique bas niveau (écrêtage par diode zener ou seuillage par trigger de Schmitt), l'élimination des parasites industriels et l'effet de rebondissement des contacts par filtrage passe bas retardateur, et le découplage ou isolation galvanique en isolant électriquement les signaux logiques de la partie commande des signaux analogiques de la partie opérative.

Associée à un circuit retardateur écrêteur (diode zener), une résistance de limitation de courant et une diode de protection contre les inversions de courant, l'interface d'entrée la plus utilisée est le coupleur optoélectronique car il réalise une bonne isolation galvanique, il permet une protection contre les perturbations électromagnétiques, les défaillances des capteurs (effet de rebondissement des lames), et les défauts de câblage (courant de fuite et mauvaise mise à la terre) [3].

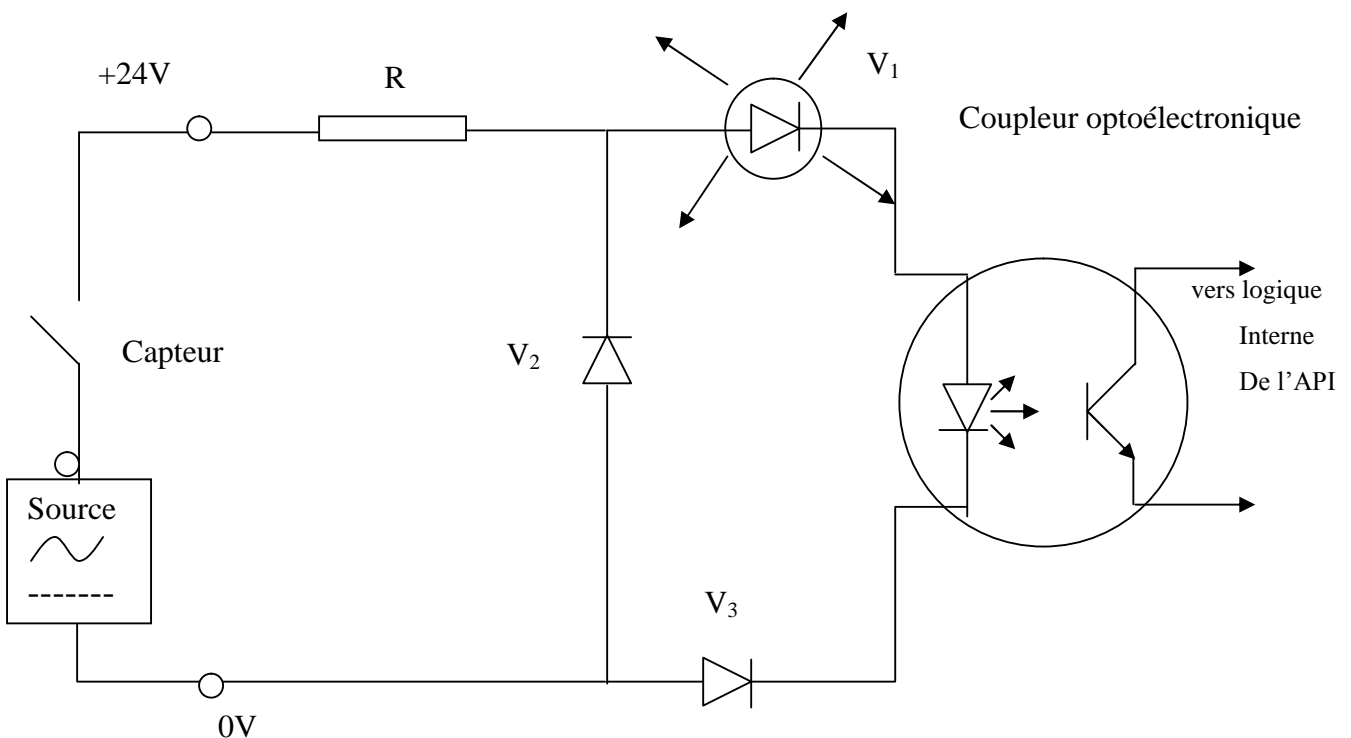


Fig –I.5- Schéma de principe d'interface d'entrée logique en 24V continu

### b- Interface de sortie :

En plus des fonctions (par coupleur optoélectronique ou relais Reed) et de protection contre les courts-circuits et les surcharges (par un dispositif thermique), contre l'inversion de polarité de l'alimentation extérieure (par diode), ils assurent une fonction de pré amplification car ils sont destinées à convertir les signaux bas niveau de la partie commande en signaux ayant une puissance suffisante pour piloter les pré-actionneurs (contacteurs, électrovannes, distributeurs...) [3].

La communication et l'amplification de puissance adaptées à la charge sont réalisées :

- **En alternatif** : le plus souvent par un relais statique ou électromagnétique (dans ce dernier cas le coupleur optoélectronique n'est pas indispensable) [3].

On peut également rencontrer un amplificateur intégré ou un triac.

- **En continu** : on utilise souvent un transistor de puissance ou un amplificateur intégré.

Le choix de la solution de communication dépend en fait de la charge : Puissance, nature de courant, inductif ou résistif, etc.... [3].

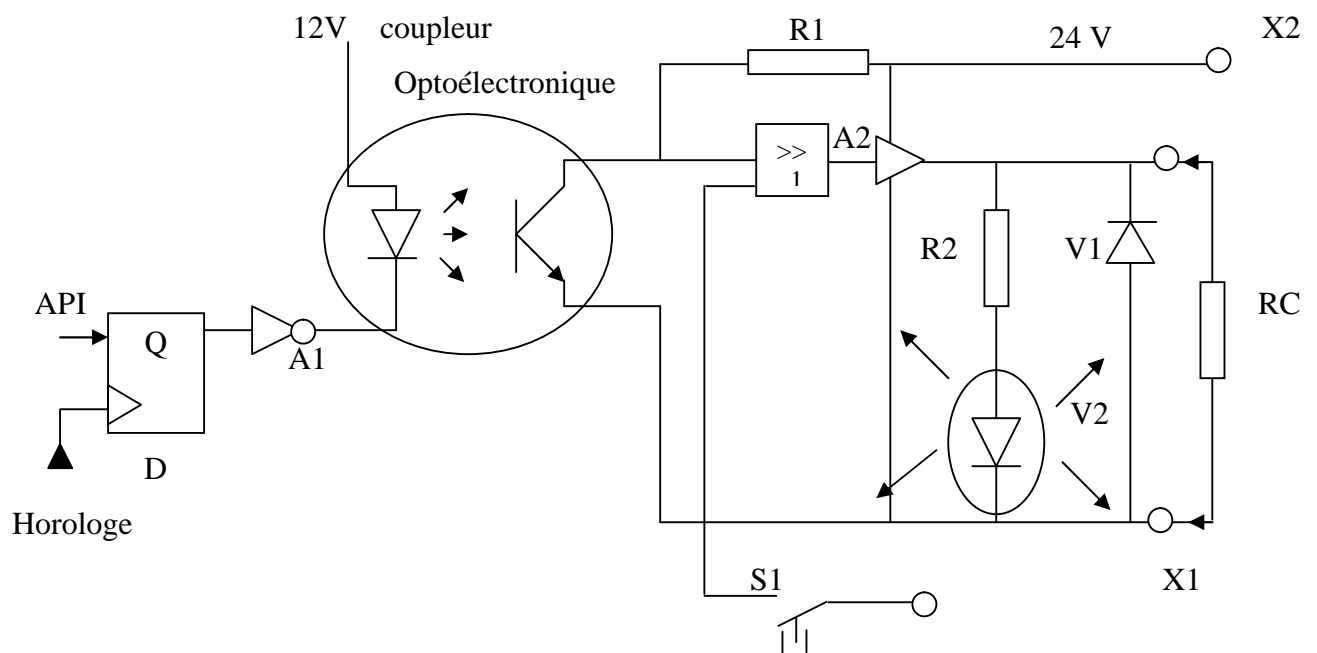


Fig –I.6 - Schéma de principe d'une interface de sortie en 24V continu

### I-6-2-3 Alimentations :

A l'intérieur de l'automate il y a un bloc d'alimentation fournissant les différentes tensions nécessaires aussi bien à la logique de commande (5 V à  $\pm 15$  V) qu'aux entrées sorties (24 V à 220 V). La tension normalisée la plus courante est de 24 V (alternatif ou continu).

Généralement l'alimentation des capteurs est fournie par l'API (en 24 V) alors que celle des actionneurs est fournie par une alimentation externe. En courant continu, la tension des signaux de sortie de l'API sont de l'ordre de 24, 48 ou 110 V, alors que l'intensité du courant varie entre 0.5 et 2 A. En alternatif 50HZ, elles peuvent aller jusqu'à 220 V et 5 A respectivement [3].

### I-6-3- Programmation des API :

Un programme est une suite d'instructions écrite dans le langage de programmation particulier de l'automate. Les automates modernes acceptent plusieurs langages de programmation dont certains sont adaptés aux problèmes d'automatisation.

Ils permettent tous de traiter les opérations de base que l'on rencontre en logique câblée : les opérations arithmétiques élémentaires, les opérations logiques (ET, OU, NON,...), les opérations séquentielles (comptage, décomptage, temporisation, mémoire bistable, saut, ...) [3].

Les principaux types de langages rencontrés sur les API sont :

#### I-6-3-1- Les langages graphiques :

On peut avoir plusieurs langages graphiques, parmi les quelles on cite :

##### a- Le diagramme-échelle (schéma à contacts) :

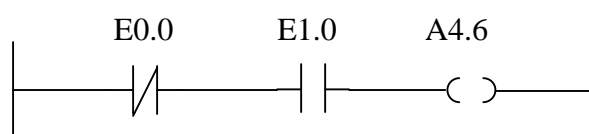
Un diagramme-échelle est également nommé "schéma des contacts" ou "logique des contacts". Il s'agit d'une représentation graphique, laquelle peut être utilisée, d'une part en tant qu'instrument pour l'élaboration des projets, et d'autre part en tant que langage de programmation directe. Les représentations graphiques sont basées sur la méthode de dessin américaine, utilisées pour représenter des contacts, des sorties, etc... [4].

—| |— : Contact normalement ouvert d'un contacteur

—|/|— : Contact normalement fermé d'un contacteur

—( )— : Bobine (sortie ou mémoire interne)

Exemple d'un schéma à contact Fig(I.7) :

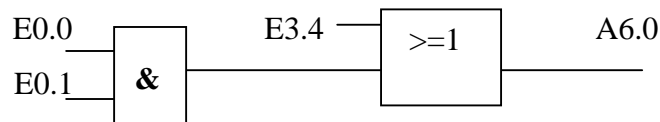


**Fig-I.7- Schéma à contact**

**b- Langage logigramme :**

Le logigramme ou diagramme logique ou schéma logique, est une représentation graphique d'un système de commande (fig.I.8) [4].

Exemple :



**Fig-I.8- Schéma d'un logigramme**

**c- Le GRAFCET :**

Le GRAFCET est une méthode de représentation graphique qui décrit les comportements successifs de la partie commande d'un système automatisé (ordres à émettre, action à effectuer, événement à surveiller). Cette représentation graphique concise et facile à lire est aisément compréhensible par toutes personnes en relation avec le système automatisé, du concepteur à l'utilisateur sans oublier l'agent de maintenance [4].

**Symbolisation du GRAFCET : [4]****Etape :**

Chaque étape est représentée par un carré repéré numériquement.

**Etape initiale :**

Représente une étape qui est active au début de fonctionnement. Elle se différencie de l'étape en doublant les cotés du carré.

**Actions associées à l'étape :**

Elles sont décrites littéralement ou symboliquement à l'intérieur d'un ou plusieurs rectangles reliés par un trait à la partie droite de l'étape.

**Transition :**

La transition est représentée par une barre.

**Réceptivité :**

Les conditions de réceptivité sont inscrites à droite de la transition.

**Liaison orientée :**

Indique le sens du parcours.

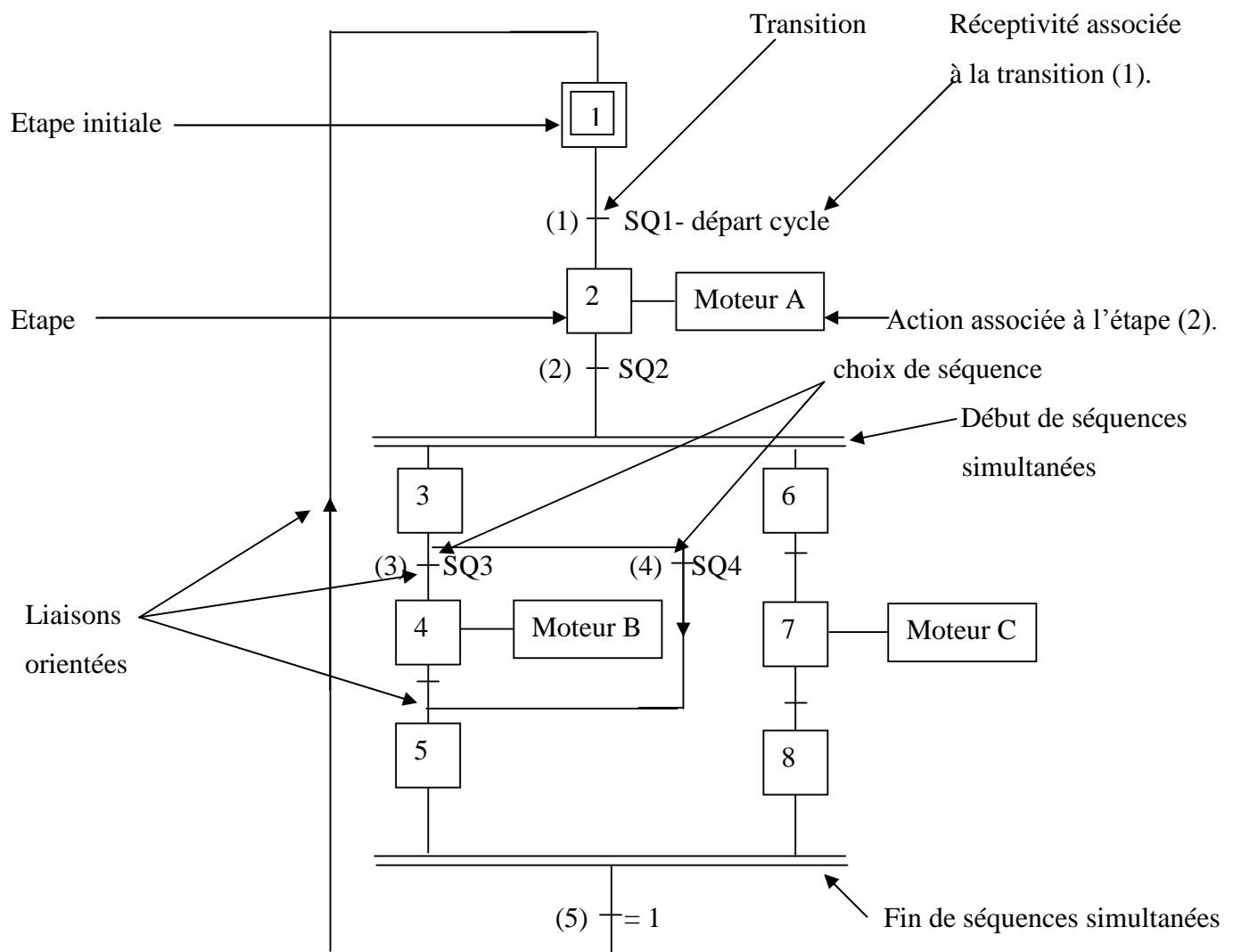


Fig -I.9- Structure d'un Grafcet

### I-6-3-2- Les langages mnémoniques :

Dans ce cas les instructions s'écrivent sous forme des expressions littérales utilisant des parties textuelles, mots réservés [5].

Ils sont définis comme suite : [5]

#### a- Les langages littéraux :

Un tel langage est une transcription élémentaire et immédiate des instructions du langage machine.

Exemple :

$y = (a.\overline{b} + c).d$  Cette équation est programmée comme suite :

Code	Commentaire
SI a	Charger a
SI / b	Charger $\overline{b}$
ET X	$X = a.\overline{b}$
SI c	Charger c
OU Z	$Z = X + c = a.\overline{b} + c$
SI Z	
SI d	Charger d
ET Y	$Y = Z.d = (a.\overline{b} + c).d$

**Tableau-I.2- Exemple d'une programmation avec un langage littéral**

#### b- Liste d'instruction (LIST) :

C'est la représentation du programme sous forme d'une suite d'abréviations d'instruction.

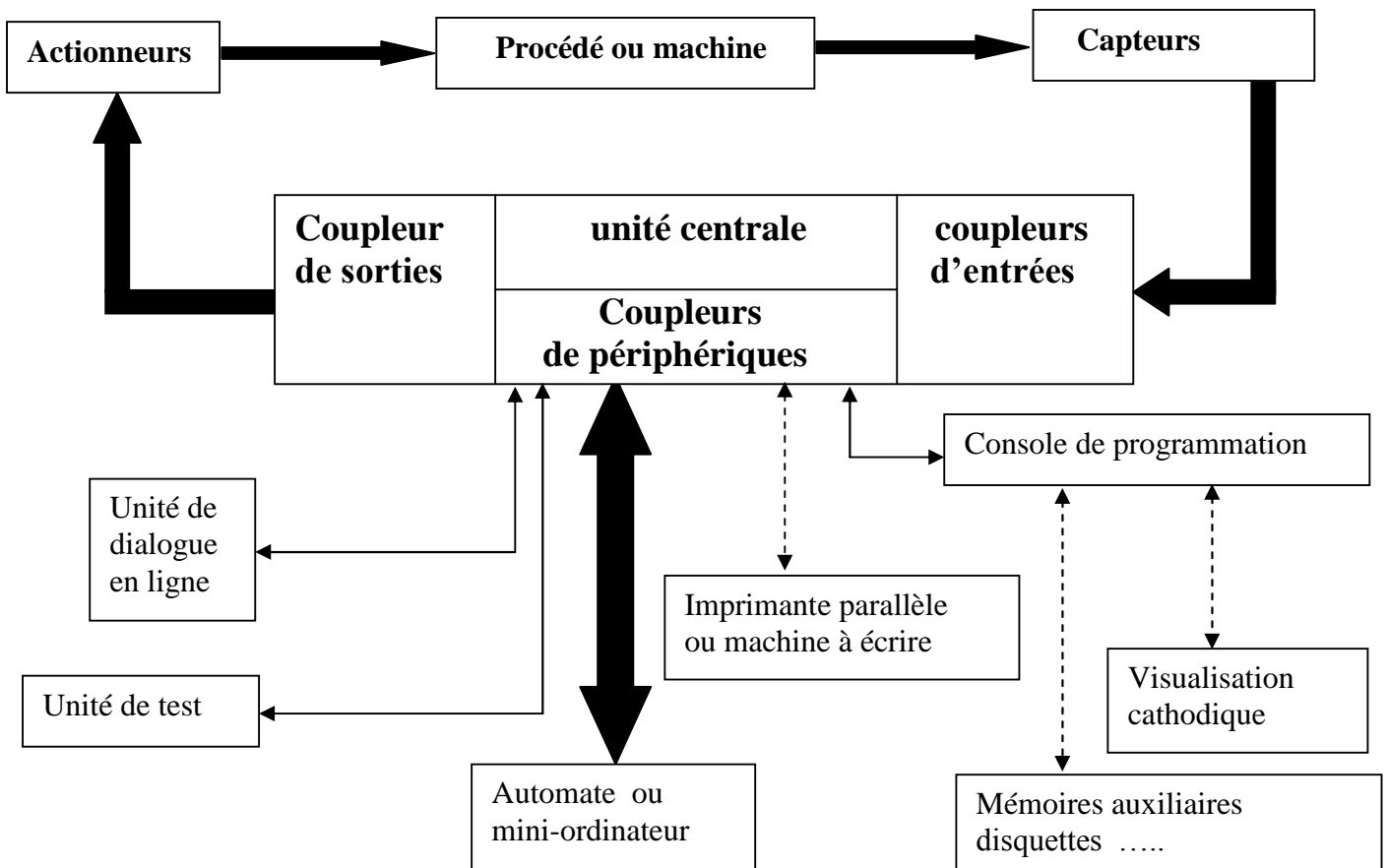
Exemple :

List	Commentaire
O E 1.2	Combinaison OU, interrogation à "1" d'un signal.
O E 1.7	Combinaison OU, interrogation à "1" d'un signal.
O E 1.5	Combinaison OU, interrogation à "1" d'un signal.
= A 3.2	Sortie.

**Tableau-I.3- Exemple d'une programmation avec un langage LIST**

#### I-7- L'environnement de l'API :

Pour conduire sa mission vis-à-vis d'un procédé, un API doit disposer des liaisons connectées temporairement ou non, directement ou non, à l'API. Ces dispositifs peuvent être, par exemple, les périphériques et les auxiliaires des API (l'imprimante, les mémoires de masse...), les consoles d'exploitation, etc... [6] .

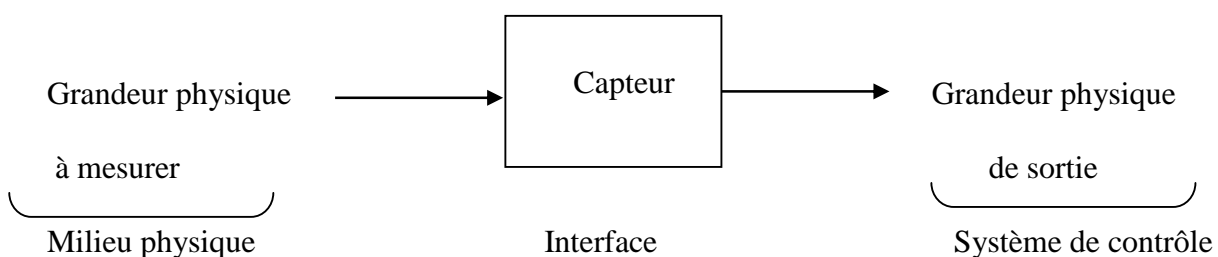


**Fig –I.10- L’Automate Programmable et son environnement.**

(**■** Liaisons permanentes, ——— temporaires, - - - - - éventuelles).

### I-7-1- Les capteurs :

Un capteur est un organe de prélèvement d'information qui élabore à partir d'une grandeur physique (information entrante), une autre grandeur physique de nature différente (très souvent électrique). Cette grandeur représentative de la grandeur prélevée est utilisable à des fins de mesure ou de commande.



**Fig –I.11 - Schéma de principe d’un capteur**

**I-7-2- Les actionneurs :**

Ce sont des organes destinés à remplacer l'énergie humaine par une énergie électrique, pneumatique ou hydraulique, ils permettent d'obtenir l'énergie nécessaire au bon fonctionnement de la machine à partir de l'énergie disponible dans l'équipement.

**I-8- Les applications des API :**

Les applications des (API) sont innombrables. On peut citer quelques domaines d'application industrielles comme suite : [7]

- Machines spéciales et d'assemblage.
- Manutention-Convoyage.
- Alimentation de secours-Groupes électrogènes.
- Station de pompage et traitement des eaux.
- Traitement des surfaces.
- Chauffage- Ventilation- Climatisation.
- Systèmes d'économie d'énergie.
- Systèmes de surveillance et d'alarme pour toutes industries.

**Conclusion**

Dans ce chapitre on a présenté une étude théorique sur les systèmes automatisés, on a illustré les avantages des automates programmables par rapport aux systèmes câblés.

Vu la souplesse, la rapidité, facilité du diagnostic lors des pannes, l'utilisation d'un outil programmé à base des (API) dans le monde industriel est devenu indispensable.