

Chapitre 1

Généralités sur les Systèmes Automatisés

I.1. Introduction

Un système automatisé est un ensemble d'éléments organisés pour réaliser de manière autonome des opérations, qui exigeaient auparavant l'intervention humaine, dans un but précis (**donner une valeur ajoutée à une matière d'œuvre**). Ou agir sur une **matière d'œuvre** afin de lui donner une **valeur ajoutée**.

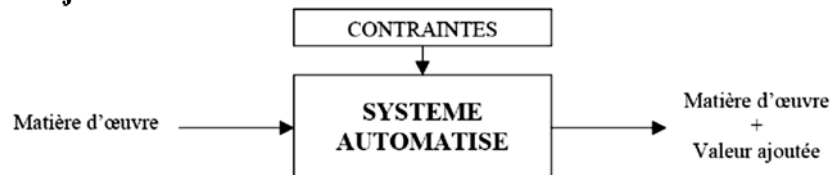


Figure I.1 système automatisé

L'automatisation permet :

- Accroître la productivité (rentabilité, compétitivité) du système
- Améliorer la flexibilité de production ;
- Améliorer la qualité du produit
- Adaptation à des environnements hostiles pour l'homme (milieu toxique, dangereux, nucléaire...), adaptation à des tâches physiques ou intellectuelles pénibles pour l'homme (manipulation de lourdes charges, tâches répétitives parallélisées...),
- Augmenter la sécurité, etc...

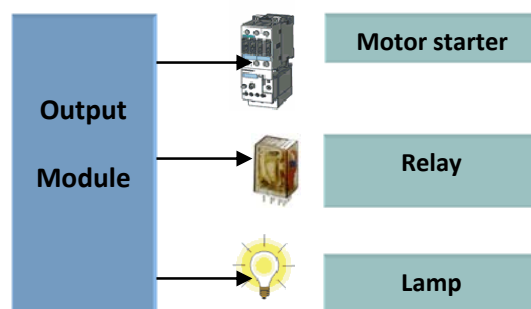
I.2 les différentes parties d'un système automatisé :

I.2.1 Partie opérative :

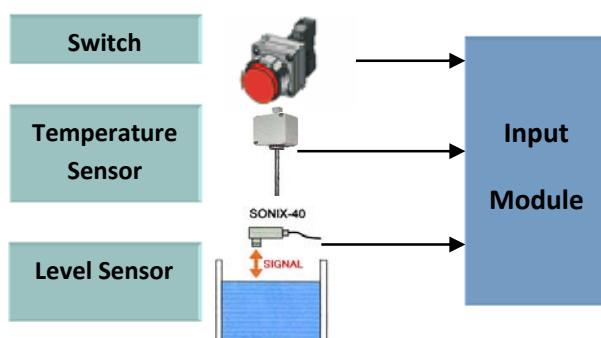
Elle agit sur la matière d'œuvre afin de lui donner sa valeur ajoutée.

Eléments de la Partie Opérative :

a/ **Les actionneurs** (moteurs, vérins) agissent sur la partie mécanique du système qui agit à son tour sur la matière d'œuvre.



b/ **Les capteurs / détecteurs** permettent d'acquérir les divers états du système.



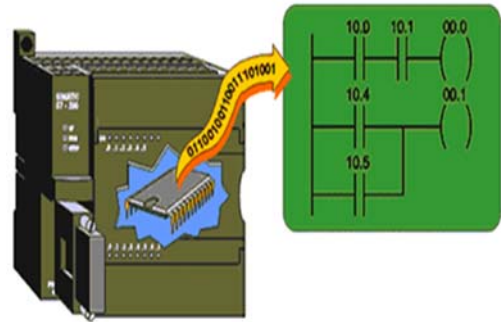
I.2.2 Partie commande :

Elle donne les ordres de fonctionnement à la partie opérative.

Les **préactionneurs** permettent de commander les actionneurs ; ils assurent le **transfert d'énergie** entre la source de puissance (réseau électrique, pneumatique ...) et les actionneurs. Exemple : contacteur, distributeur ...

Ces préactionneurs sont commandés à leur tour par le bloc **traitement des informations**.

Celui-ci reçoit les consignes du **pupitre de commande** (opérateur) et les informations de la partie opérative transmises par les capteurs / détecteurs.



I.2.3 Poste de contrôle :

Composé des **pupitres de commande et de signalisation**, il permet à l'opérateur de commander le système (marche, arrêt, départ cycle ...).

Il permet également de visualiser les différents états du système à l'aide de voyants, de terminal de dialogue ou d'interface homme-machine (IHM).

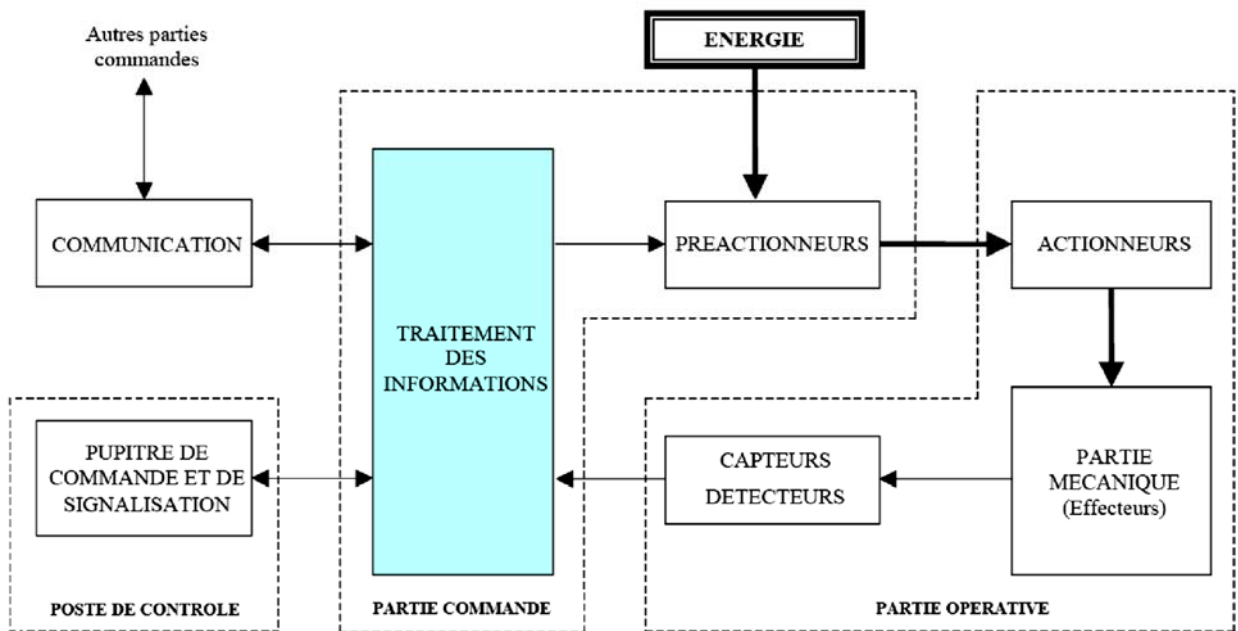
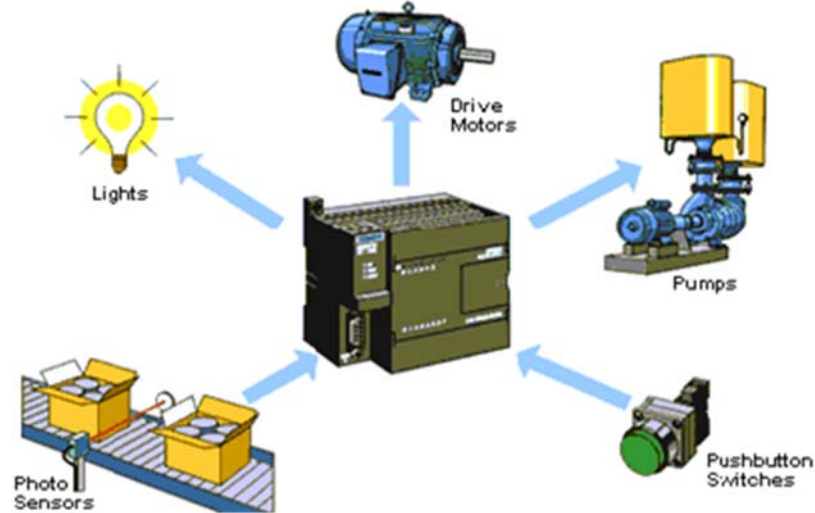


Figure I.2 Les différentes parties d'un système automatisé

I.3 AUTOMATES PROGRAMMABLES INDUSTRIELS

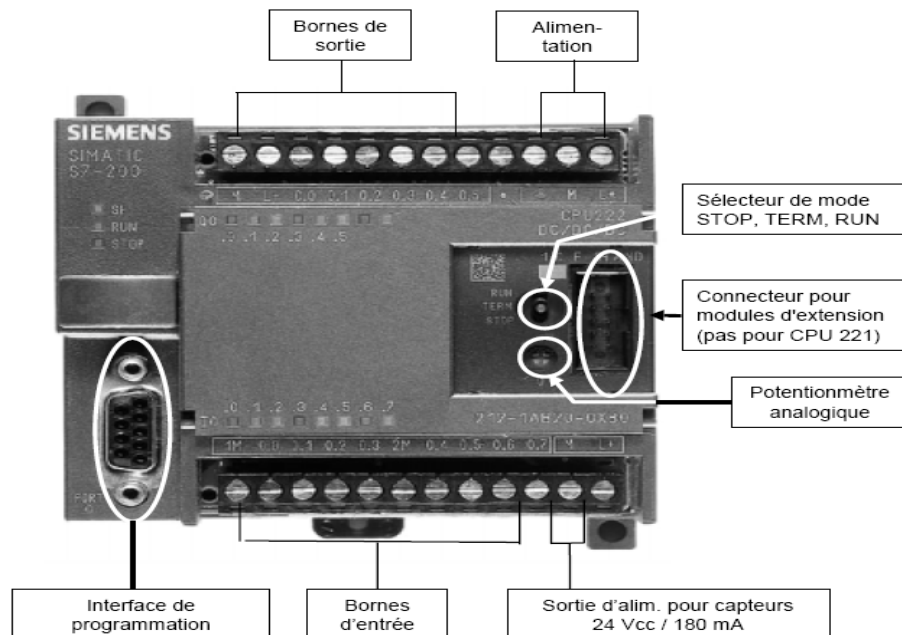
Un automate programmable industriel, ou API (programmable logic controller, PLC), est un dispositif électronique numérique programmable destiné à la commande de processus industriels par un traitement séquentiel. Il envoie des ordres vers les *préactionneurs* (partie opérative ou *PO* côté actionneur) à partir de données d'entrées (capteurs) (partie commande ou *PC* côté capteur), de consignes et d'un programme informatique.



I.3.1 Structure générale des API :

Les caractéristiques principales d'un automate programmable industriel (API) sont : coffret, rack, baie ou cartes

- Compact ou modulaire
- Tension d'alimentation
- Taille mémoire
- Sauvegarde (EPROM, EEPROM, pile, ...)
- Nombre d'entrées / sorties
- Modules complémentaires (analogique, communication,)
- Langage de programmation

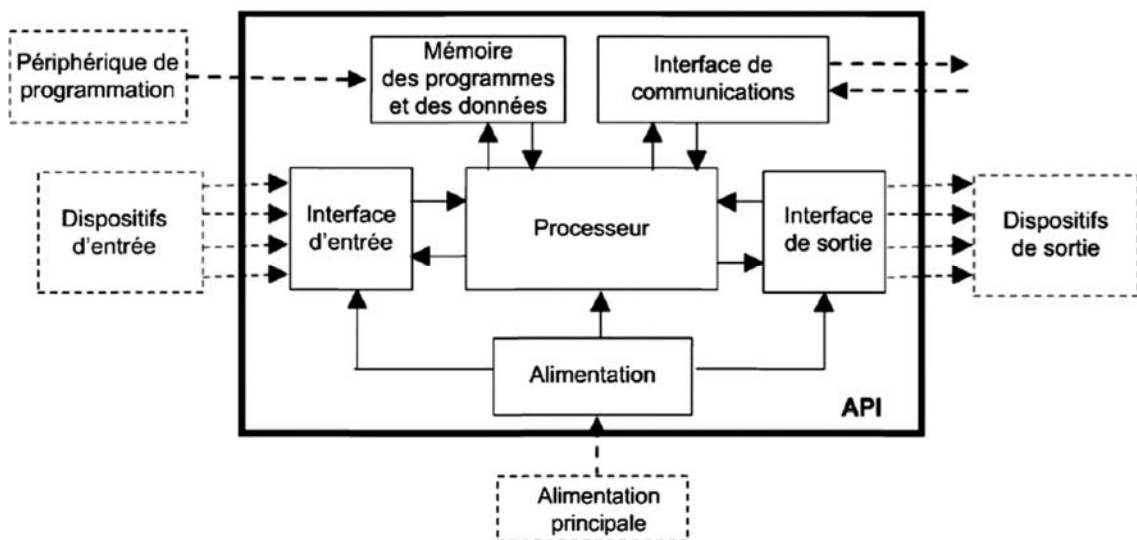


I.3.2 La structure interne d'un automate programmable industriel

La structure interne d'un automate programmable industriel (API) est assez voisine de celle d'un système informatique simple,

Principales parties des API :

- Une unité de traitement (un processeur CPU);
- La mémoire Programme où est stocké le langage de programmation. Elle est en général figée, c'est à dire en lecture seulement. (ROM : mémoire morte);
- La mémoire de données utilisable en lecture-écriture pendant le fonctionnement c'est la RAM (mémoire vive). Elle fait partie du système entrées-sorties. Elle fige les valeurs (0 ou 1) présentes sur les lignes d'entrées, à chaque prise en compte cyclique de celle-ci, elle mémorise les valeurs calculées à placer sur les sorties ;
- Des modules d'entrées-sorties ;
- Des interfaces d'entrées-sorties ;
- Une alimentation 230 V, 50/60 Hz (AC) – 24 V (DC).



Par rapport aux ordinateurs, les API se caractérisent par :

- leur robustesse : conçus pour pouvoir travailler en milieu hostile, ils utilisent des circuits durcis et sont prévus pour résister aux vibrations, aux températures des ateliers etc ;
- leur réactivité aux indications fournies par les capteurs (dispositifs anti-collision, alarmes diverses) ;
- leur facilité de maintenance (bien que les ordinateurs industriels atteignent également un très bon degré de fiabilité). Les modules peuvent être changés très facilement et le redémarrage des API est très rapide.

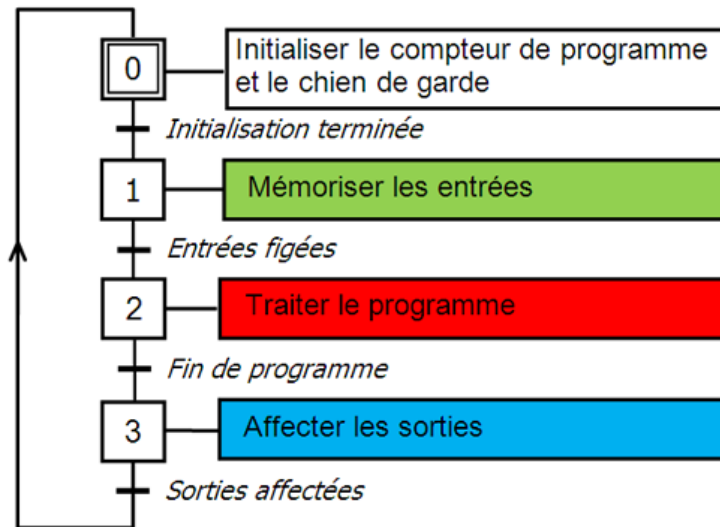
I.3.3 - Fonctionnement :

Généralement les automates programmables industriels ont un fonctionnement cyclique. L'automate programmable reçoit les informations relatives à l'état du système et puis commande les pré-actionneurs suivant le programme inscrit dans sa mémoire.

le cycle de traitement commence par :

- la prise en compte des entrées qui sont figées en mémoire pour tout le cycle.
- Le processeur exécute alors le programme instruction par instruction en rangeant à chaque fois les résultats en mémoire.

- En fin de cycle les sorties sont affectées d'un état binaire, par mise en communication avec les mémoires correspondantes.



I.3.4. Domaines d'emploi des automates :

On utilise les API dans tous les secteurs industriels pour la *commande des machines* (convoyage, emballage ...) ou des *chaînes de production* (automobile, agroalimentaire ...) ou il peut également assurer des fonctions de *régulation de processus* (métallurgie, chimie ...). Il est de plus en plus utilisé dans le domaine du *bâtiment* (tertiaire et industriel) pour le contrôle du chauffage, de l'éclairage, de la sécurité ou des alarmes.

I.3.5. Différents langages de programmation :

Il existe différents langages de programmation définis par la CEI 61131-3 :

- IL (Instruction List), le langage List est très proche du langage assembleur on travaille au plus près du processeur en utilisant l'unité arithmétique et logique, ses registres et ses accumulateurs
- ST (Structured Text), Ce langage structuré ressemble aux langages de haut niveau utilisés pour les ordinateurs
- LD (Ladder Diagram), le langage Ladder (échelle en anglais) ressemble aux schémas électriques et permet de transformer rapidement une ancienne application faite de relais électromécaniques en un programme. Cette façon de programmer exploite une approche visuelle du problème longtemps appréciée en industrie, mais qui s'appuie sur une logique de moins en moins adaptée mais toujours utilisée (2013). On parle également de langage à contacts ou de schéma à contacts pour désigner ce langage Ladder.
- Boîtes fonctionnelles (FBD), le FBD se présente sous forme diagramme : suite de blocs, connectables entre eux, réalisant des opérations, simples ou très sophistiquées.

Dans la programmation d'un automate, il est possible également de choisir de programmer en SFC, dérivé du grafset. À chaque action élémentaire est associé un programme écrit en IL, ST, LD ou FBD.

I.3.6. Avantages et inconvénients :

Les API présentent de nombreux intérêts :

- Les éléments qui les composent sont particulièrement robustes (absence de mécanique tournante pour le refroidissement et le stockage des données, matériaux renforcés) leur

permettant de fonctionner dans des environnements particulièrement hostiles (poussière environnante, perturbations électromagnétiques, vibrations des supports, variations de température...)

- Ils possèdent des circuits électroniques optimisés pour s'interfacer avec les entrées et les sorties physiques du système, les envois et réceptions de signaux se font très rapidement avec l'environnement. Avec de plus une exécution séquentielle cyclique sans modification de mémoire, ils permettent d'assurer un temps d'exécution minimal, respectant un déterminisme temporel et logique, garantissant un temps réel effectif (le système réagit forcément dans le délai fixé).

En contrepartie, ils sont plus chers que des solutions [informatiques](#) classiques à base de microcontrôleurs par exemple mais restent à l'heure actuelle les seules plateformes d'exécution considérées comme fiables en milieu industriel (avec les ordinateurs industriels). Le prix est notamment dépendant du nombre d'entrées/sorties nécessaires, de la mémoire dont on veut disposer pour réaliser le programme, de la présence ou non de modules métier. De plus ils nécessitent la maîtrise de langages spécifiques conformes à la norme [CEI 61131-3](#) qui reprennent dans leur forme la logique d'exécution interne de l'automate. Ces langages apparaissent toutefois à beaucoup d'utilisateurs plus accessibles et plus visuels que les langages informatiques classiques.

I.3.7. Cahier des charges

Le cahier des charges est le descriptif fourni par l'utilisateur au concepteur de l'automatisme pour lui indiquer les différents modes de marches et les sécurités que devra posséder l'automatisme.

Ce cahier des charges décrit le comportement de la partie opérative par rapport à la partie commande.

On utilise des outils de description :

Le GRAFCET, les organigrammes, les logigrammes, les chronogrammes, pour décrire le cahier des charges.