

CHAPITRE I

Systemes Automatisés

I. Mise en situation

I.1. Définitions

- *L'automatisation* consiste à rendre automatique les opérations qui exigeaient auparavant l'intervention humaine.
- *L'automatisation* est considérée comme l'étape d'un progrès technique ou apparaissent des dispositifs techniques susceptibles de seconder l'homme, non seulement dans ses efforts musculaires, mais également dans son travail intellectuel de *surveillance* et de *contrôle*.

I.2. Historique

- Les précurseurs tels que Blaise PASCAL (1623- 1662) qui invente la première calculatrice "La Pascaline".
- Bardeen, Brattain & Shockley inventent le Transistor en 1949.
- Moore (1966) invente le microprocesseur et crée Intel.
- Grenelle (1972) invente le 1^{er} micro-ordinateur.
- 1974 : Premiers Automates Programmables Industriels.

I.3. Les buts (ou objectifs) de l'automatisation

Objectifs : La compétitivité de l'entreprise et des produits passe par la qualité, la maîtrise des coûts et l'innovation. On cherche donc à améliorer la *productivité*, les conditions de travail, et surtout la *sécurité*.

Les buts (ou objectifs) de l'automatisation sont donc:

- Eliminer les tâches répétitives,
- Simplifier le travail de l'humain,
- Augmenter **la sécurité** en améliorant les conditions de travail,
- Accroître et améliorer **la productivité**.

C'est également :

- Economiser les matières premières et l'énergie,
- S'adapter à des contextes particuliers : **flexibilité**,
- Améliorer **la qualité**.

Moyens : Les moyens permettant d'atteindre ces objectifs sont d'ordre financier, matériel, organisationnel, et humain. On distingue notamment : informatiser, automatiser, robotiser, rationaliser, organiser, etc, ...

I.4. Niveau d'automatisation, de rentabilité, et de sûreté.

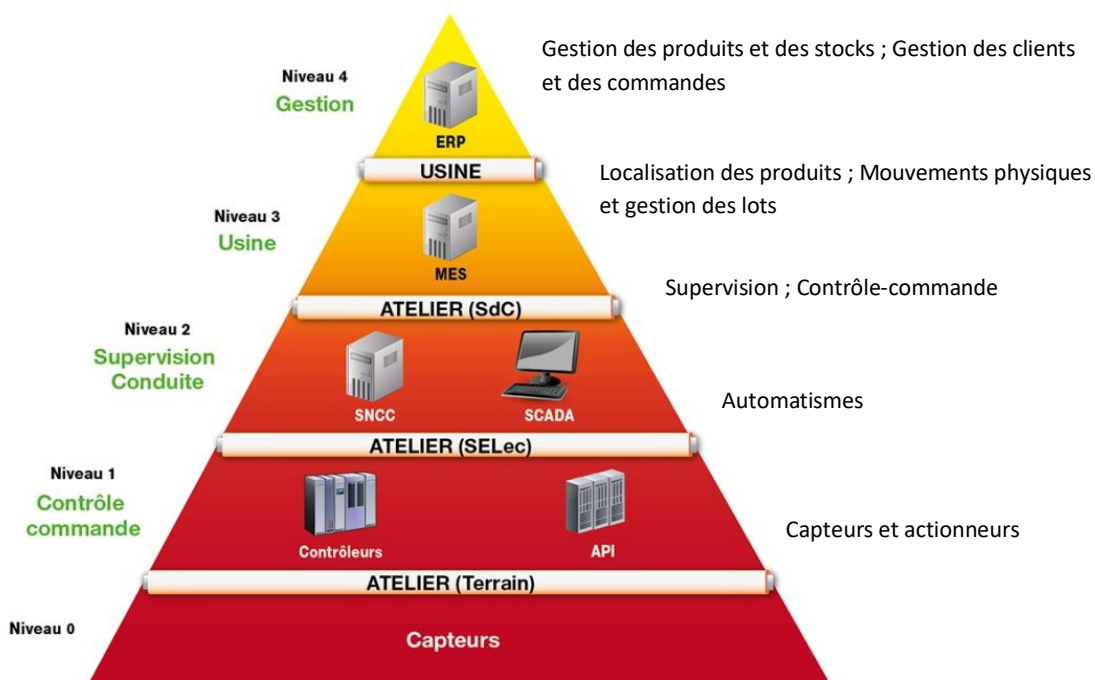
Il y a différents degrés d'automatisation qui vont du niveau totalement manuel correspondant à 0% d'automatisation à un niveau total d'automatisation (100%) dans lequel l'homme ne joue aucun rôle dans l'accomplissement de la tâche. Cependant, dans la plupart des applications, l'opérateur humain et la machine

automatisée coopèrent pour mener à bien la tâche ou la mission à accomplir. Dans les systèmes de haut degré d'automatisation, l'opérateur humain joue seulement le rôle de superviseur.

Ainsi, l'automatisation nécessite une reconception simultanée du produit, des moyens de production, et du processus. Comment mettre en œuvre un système automatisé complexe, sûr et efficace intégrant une multitude de moyens techniques et des moyens humains à des rythmes différents ?

L'International society of automation (ISA) a défini une organisation pyramidale à quatre étages. Le **C.I.M.** (*Computer Integrated Manufacturing*), ou production intégrée par ordinateur, est un concept décrivant l'automatisation complète des procédés de fabrication. C'est-à-dire que tous les équipements de l'usine fonctionnent sous le contrôle permanent des ordinateurs, automates programmables et autres systèmes numériques.

De bas en haut : la partie *matérielle*, le logiciel de *supervision* (SCADA), le système *d'exécution* de la production (MES) et le système de *gestion* (ERP).



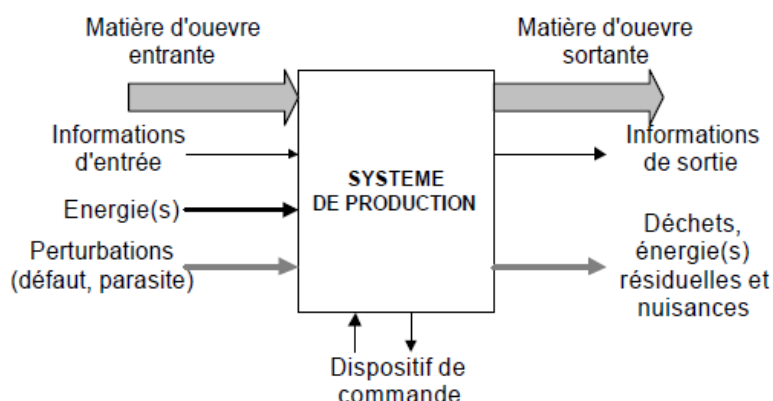
- **ERP** (Enterprise Resource Planning), pour la gestion des ressources de l'entreprise).
- **MES** (Manufacturing Execution System), pour le suivi de la production).
- **SCADA** (Supervisory Control and Data Acquisition), pour le pilotage des machines automatisées).

II. Systèmes automatisés

II.1. Système technique

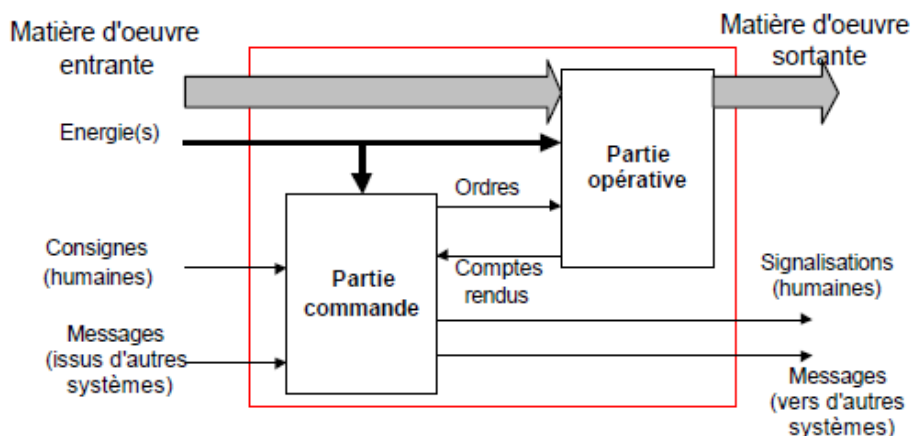
C'est un ensemble technique conçu pour répondre à un besoin. Sa fonction globale est d'apporter une valeur ajoutée : partant de matériaux, de pièces, de sous-ensembles, ..., il élabore des produits de valeur supérieure qui peuvent être des produits finis ou intermédiaires.

Un système de production est dit «industriel» si l'obtention de cette valeur ajoutée, pour un ensemble de matières d'œuvre donné, à un caractère reproductible et peut être exprimée et quantifiée en termes économiques.



II.2. Structure d'un système automatisé

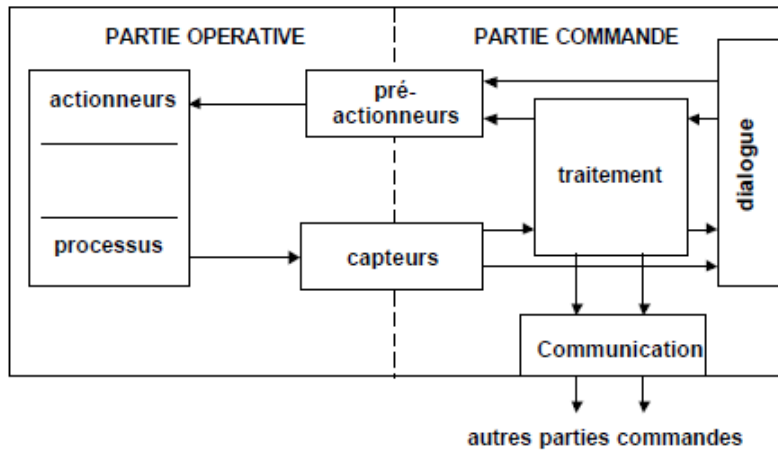
Tout système automatisé se décompose en deux parties interdépendantes : la partie opérative (P.O.) et partie commande (P.C.). Les informations échangées entre ces deux parties sont des informations internes au système alors que les consignes, signalisation proviennent de l'extérieur.



i- Partie opérative : C'est elle qui opère sur la matière d'œuvre, elle comporte en général :

- Processus : outillages et moyens divers (moules, outils de coupe, pompes, têtes de soudures,...)
- Actionneurs : destinés à mouvoir ou mettre en œuvre ces moyens (moteur électrique pour actionner une pompe, vérin hydraulique pour fermer un moule, vérin pneumatique pour mouvoir une tête de marquage,...).

ii- *Partie commande* : un automatisme qui élabore en sortie *des ordres* destinés à la partie opérative et des *signaux de visualisation*, en fonction du contrôle du processus. Cette commande est basée sur les *consignes* utilisateur et sur les *informations* reçues concernant l'état du système.



Dans la partie commande, l'unité de traitement reçoit les informations *des capteurs* et émet des *ordres* vers les *pré-actionneurs*. Bien connaître ces organes répartis sur la machine est indispensable pour réaliser l'automatisme qui doit les commander.

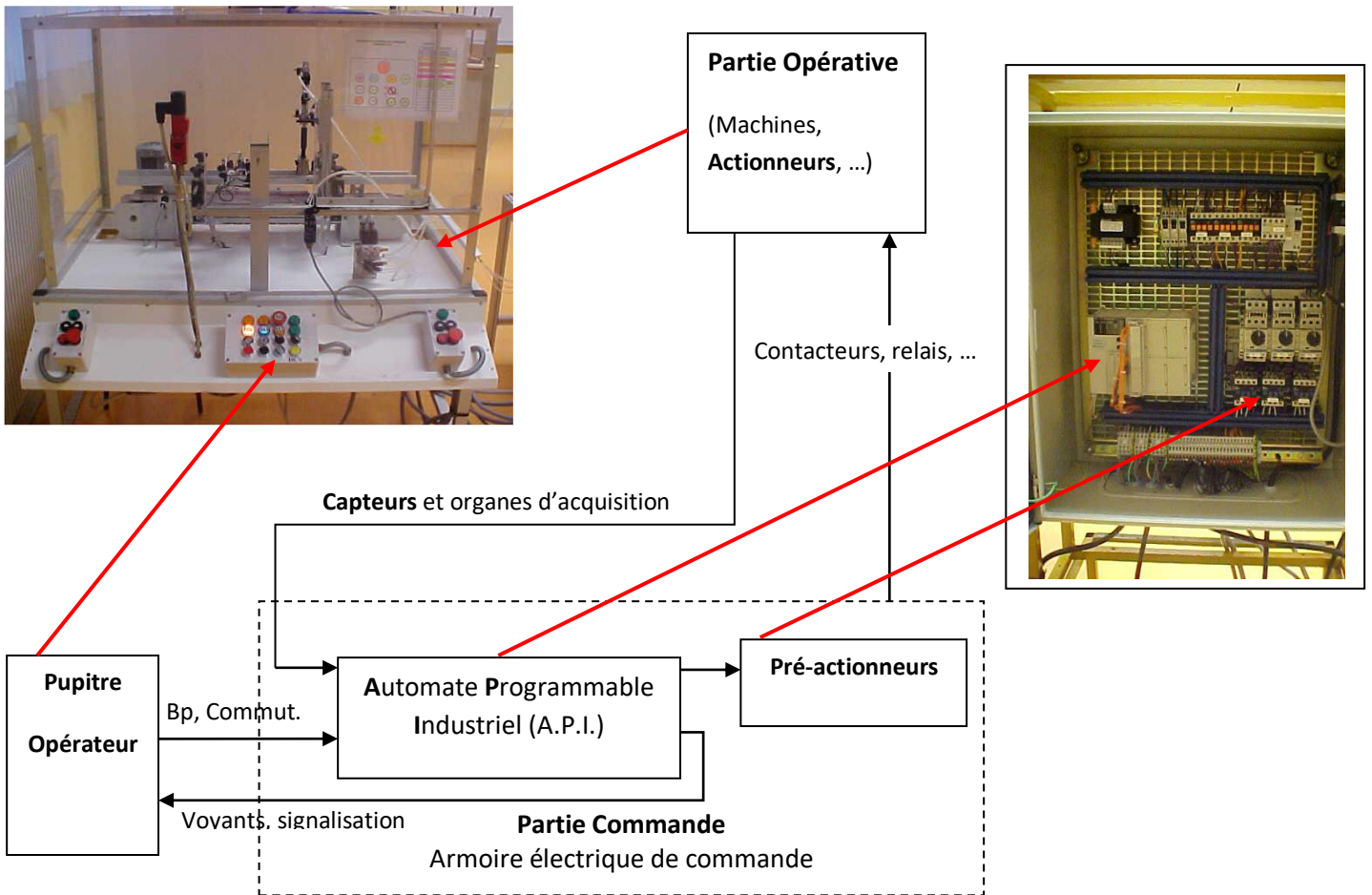


Schéma fonctionnel d'un système automatisé

CHAPITRE III Structure des Automates Programmables

I. Historique

Les automates programmables industriels sont apparus à la fin des années soixante, à la demande de l'industrie automobile américaine (GM), qui réclamait plus d'adaptabilité de leurs systèmes de commande.

Les coûts de l'électronique permettant alors de remplacer avantageusement les technologies actuelles.

- **Avant** : utilisation de relais électromagnétiques et de systèmes pneumatiques pour la réalisation des parties commandes ⇒ *logique câblée*

Inconvénients : cher, pas de flexibilité, pas de communication possible.

- **Solution** : utilisation de systèmes à base de microprocesseurs permettant une modification aisée des systèmes automatisés ⇒ *logique programmée*

Les ordinateurs de l'époque étant chers et non adaptés aux contraintes du monde industriel, les automates devaient permettre de répondre aux attentes de l'industrie.

- **Contraintes du monde industriel :**

- Influences externes :

- *poussières,*
- *température,*
- *humidité,*
- *vibrations,*
- *parasites électromagnétiques, ...*

- Personnel :

- *mise en œuvre du matériel aisée (pas de langage de programmation complexe)*
- *dépannage possible par des techniciens de formation électromécanique*
- *possibilité de modifier le système en cours de fonctionnement*

- Matériel :

- *évolutif*
- *modulaire*
- *implantation aisée*

- **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.

- **Nature des informations traitées par l'automate**

Les informations peuvent être de type :

- *Tout ou rien* (T.O.R.) : l'information ne peut prendre que deux états (vrai/faux, 0 ou 1 ...). C'est le type d'information délivrée par un détecteur, un bouton poussoir ...
- *Analogique* : l'information est continue et peut prendre une valeur comprise dans une plage bien déterminée. C'est le type d'information délivrée par un capteur (pression, température ...)
- *Numérique* : l'information est contenue dans des mots codés sous forme binaire ou bien hexadécimale. C'est le type d'information délivrée par un ordinateur ou un module intelligent.

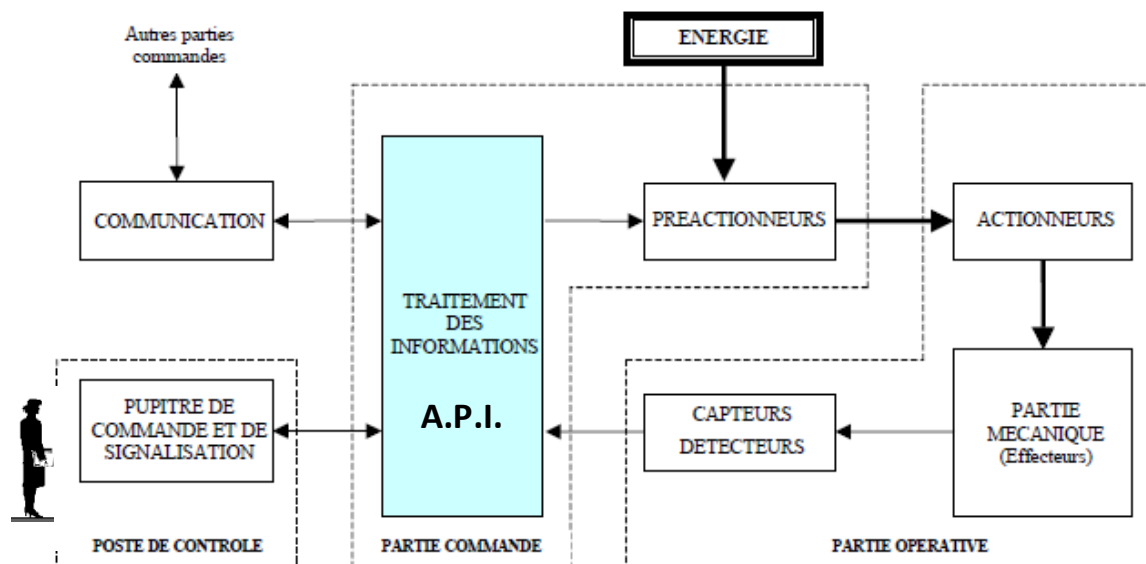
II. Définition

L'Automate Programmable Industriel (API) est un appareil électronique programmable, adapté à l'environnement industriel, qui réalise des fonctions d'automatisme pour assurer la commande de préactionneurs et d'actionneurs à partir d'informations logique, analogique ou numérique.

III. Place de l'API dans le système automatisé de production (S.A.P.)

Le système automatisé est soumis à des *contraintes* : énergétiques, de configuration, de réglage et d'exploitation qui interviennent dans tous les modes de marche et d'arrêt du système.

Les automates occupent la partie commande d'où émergent les ordres de fonctionnement vers la partie opérative.



- Le poste de contrôle, composé des pupitres de commande et de signalisation, permet à l'opérateur de *commander le système* (consignes de 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).

- Le bloc de traitement des informations *reçoit les consignes* du pupitre de commande (opérateur) et les *informations* de la partie opérative transmises par les capteurs/détecteurs.

En fonction de ces *consignes et* du *programme* de gestion des tâches implanté dans l'automate programmable, il va *commander les préactionneurs* et *renvoyer des informations au pupitre* de signalisation ou à d'autres systèmes de commande et/ou de supervision en utilisant un réseau et un protocole de communication.

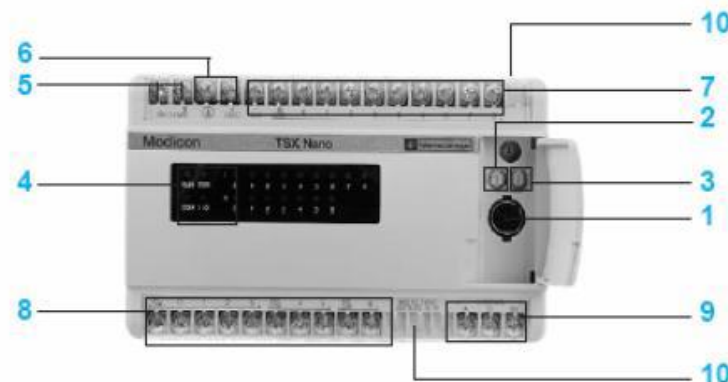
- Les *préactionneurs*, permettent de *commander les actionneurs* et *assurer le transfert d'énergie* entre la source de puissance (réseau électrique, pneumatique ...) et les actionneurs. Exemple : contacteur, distributeur ...

IV. Architecture des automates

IV.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 de Schneider, MILLENIUM de Crouzet ...) des *microautomates*. 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.

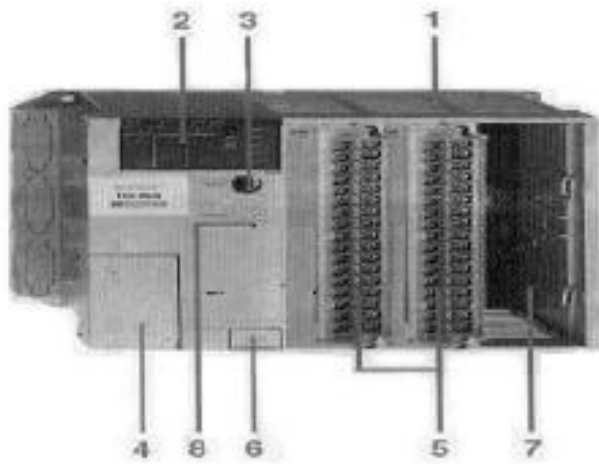


Automate monobloc
TSX Nano

Les automates *monobloc* comprennent en général :

- 1 - Une prise pour raccordement du terminal permettant la programmation à partir d'une console ou d'un ordinateur.
- 2 et 3 - Des prises pour des fonctions de communication spécialisées avec d'autre équipement.
- 4- Ecran sur lequel on peut visualiser: - l'état des entrées et sorties relais "TOR" - l'état automate.
- 5 - Un raccordement de l'alimentation secteur.
- 6 - Une alimentation capteurs (en général 24 VDC/150mA).
- 7 - Un raccordement des capteurs d'entrées.
- 8 - Un raccordement des pré actionneurs de sorties.
- 9 - Un raccordement pour des entrées spécifiques.
- 10 - Un cache amovible pour protection des borniers à vis.

- 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
modulaire TSX37-08

- 1- Un bac à 3 emplacements.
- 2- Un bloc de visualisation centralisé.
- 3- Une prise terminale repérée TER.
- 4- Une trappe d'accès aux bornes d'alimentation.
- 5- Deux modules à 16 entrées et 12 sorties TOR positionnés dans le premier et le deuxième emplacement (positions 1, 2, 3 et 4).
- 6- Une trappe d'accès à la pile optionnelle.
- 7- Un emplacement disponible.
- 8- Un bouton de réinitialisation

IV.2. Structure interne

IV.2.1. Module d'alimentation :

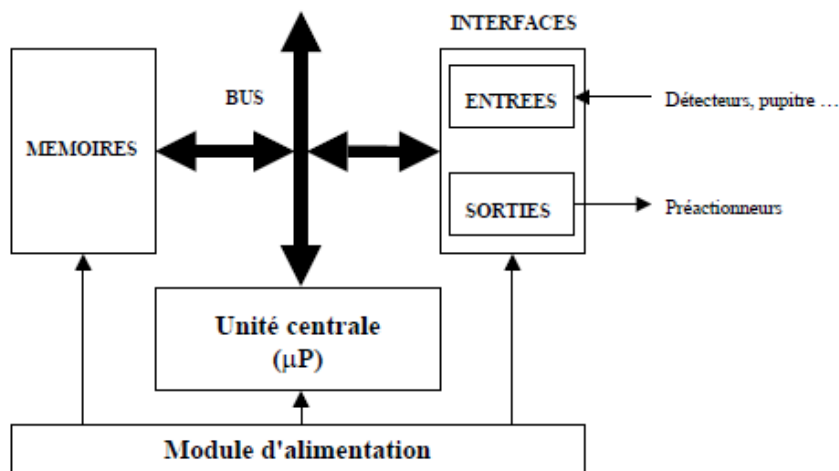
Composé de blocs qui permettent de fournir à l'automate l'énergie nécessaire à son fonctionnement. Cette énergie sera dimensionnée en fonction des consommations des différentes parties. En règle générale, un voyant positionné sur la façade indique la mise sous tension de l'automate. On distingue trois sources d'alimentation : alimentation *alternative*, *continue* ou *auxiliaire*.

- *L'alimentation alternative*: délivre, à partir du secteur 110V ou 220V, les principales tensions dont l'automate a besoin : 24V, 12V ou 5V en continu. Ces alimentations comportent deux parties : le redressement et la régulation. En cas de défaillance de l'automate, certaines alimentations sont dotées d'un contact à relais et d'un voyant.
- *L'alimentation continue*: certains automates fonctionnent à partir de tensions continues fournies par des alimentations externes. Elle nécessite l'utilisation de transformateurs à enroulements séparés.
- *Les alimentations auxiliaires* : Lorsque l'alimentation principale ne peut pas assurer le fonctionnement de toutes les cartes, il faut prévoir des alimentations auxiliaires. Celles-ci sont implantées soit à l'intérieur du coffret, soit dans des racks d'extension, soit dans les colonnes d'interface, ou encore à l'extérieur de l'automate.

IV.2.2. Unité centrale : à base de **microprocesseur**, elle réalise toutes les fonctions logiques, arithmétiques et de traitement numérique (transfert, comptage, temporisation ...).

IV.2.3. Le bus interne : il permet la communication de l'ensemble des blocs de l'automate et des éventuelles extensions.

IV.2.4. 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.



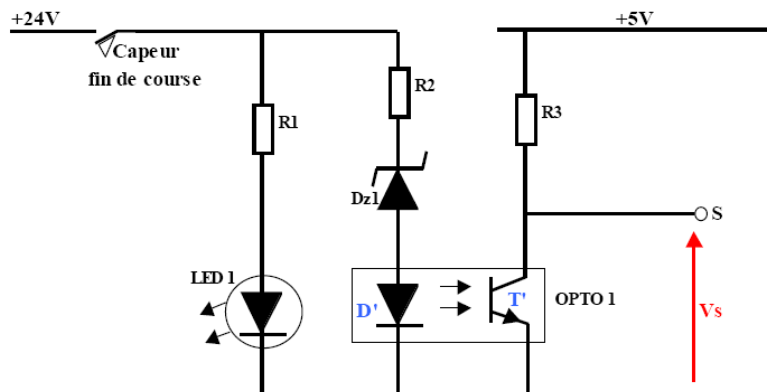
IV.2.5. Interfaces d'entrées/sorties

IV.2.5.1. 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).

a. Interfaces d'entrées TOR

Elles sont destinées à :

- Recevoir l'information en provenance des capteurs logiques tels que les boutons poussoirs, les pressostats, thermostats, fins de course, capteurs de proximité inductifs ou capacitifs, capteurs photo-électriques, roues codeuses etc.
- Assurer l'adaptation, l'isolement, le filtrage et la mise en forme des signaux électriques. Une diode électroluminescente située sur la carte donne l'état de chaque entrée. Le nombre d'entrées sur une carte est de : 4, 8, 16, 32.



➤ **Fonctionnement:**

Lors de la fermeture du capteur ;

- LED1 signale que l'entrée automate est actionnée
- La led D' de l'optocoupleur s'éclaire
- Le photo transistor T' de l'optocoupleur devient passant
- La tension $V_s = V_{ce} = 0V$ (transistor saturé)

Donc lors de l'activation d'une entrée automate, l'interface d'entrée envoie *un 0 logique* à l'unité de traitement et *un 1 logique* lors de l'ouverture du contact du capteur (entrée non actionnée).

V.1 Jeu d'instructions

Le processeur peut exécuter un certain nombre d'opérations logiques; l'ensemble des instructions booléennes des instructions complémentaires de gestion de programme (saut, mémorisation, adressage ...) constitue un jeu d'instructions.

Sur une ligne de programme, on trouve systématiquement un *code d'instruction*, suivi éventuellement de *l'adresse de l'opérande* (variable) sur laquelle s'applique l'opération.

V.2 Opérations logiques de base

Les opérations logiques de base sont :

- Lecture de l'état d'une variable (**Load**, **If**, ... etc);
- Et logique (**AND**);
- OU logique (**OR**, + ...);
- Affectation (=, **SET**, **OUT**);
- Négation (**NOT**, **Non**, **Pas**).

V.3. Instructions complémentaires

Les instructions complémentaires sont la mémorisation, la temporisation, le comptage, le saut (avant ou arrière ou le deux),...

V.4. 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.

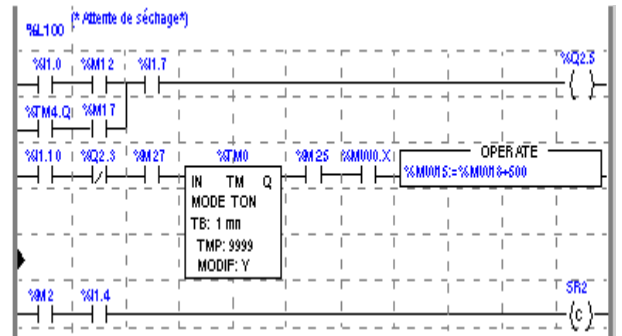
On distingue:

➤ **Les langages graphiques**

- **LD : Ladder Diagram (Diagramme à contacts).**

Langage graphique développé pour les *électriciens*. Il utilise les symboles tels que: contacts, relais et blocs fonctionnels et s'organise en réseaux (labels).

C'est le plus utilisé.



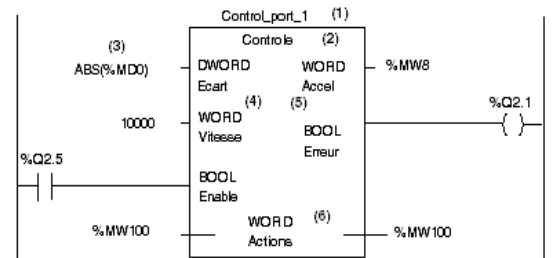
Le langage à relais (Ladder Diagram) est basé sur un symbolisme très proche de celui utilisé pour les schémas de câblage des systèmes à relais. Les symboles les plus utilisés sont donnés au tableau suivant :

Fonction	Symbole	
	Européen	Américain
Contact normalement ouvert	---o o---	
Contact normalement fermé	---o---o---	
Début de branchement		
Fin de branchement		
Affectation	---()---	---()

- **FBD : Function Block Diagram (Logigrammes).**

Langage graphique où des fonctions sont représentées par des rectangles avec les entrées à gauche et les sorties à droites. Les blocs sont programmés (bibliothèque) ou programmables.

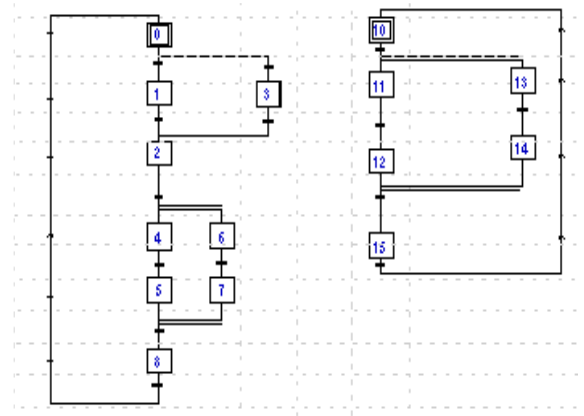
Utilisé par les automaticiens.



- **SFC : Sequential Function Chart (Grafcet).**

Le GRAFCET, langage de spécification, est utilisé par certains constructeurs d'automate (Schneider, Siemens) pour la programmation. Parfois associé à un langage de programmation, il permet une programmation aisée des systèmes séquentiels tout en facilitant la mise au point des programmes ainsi que le dépannage des systèmes.

On peut également traduire un Grafcet en langage à contacts et l'implanter sur tout type d'automate.



➤ *Les langages textuels*

- **IL** : Instruction List (Liste d'instructions).

De même nature que l'assembleur (programmation des microcontrôleurs).
Très peu utilisé par les automaticiens.

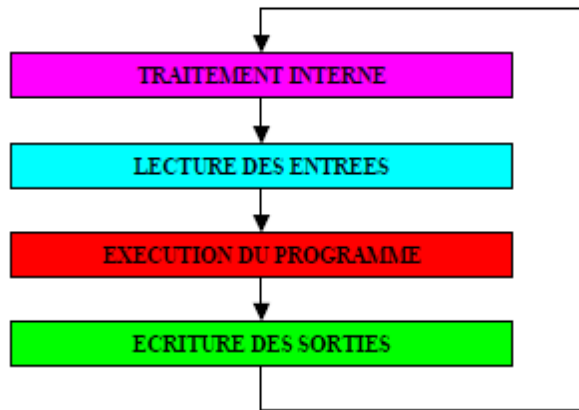
```
! %L0: LD      %I1.0
      ANDN   %M12
      OR (   %TM4.Q
      AND   %M17
      )
      AND   %I1.7
      ST    %Q2.5
```

- **ST** : Structured Text (Texte structuré).

Langage informatique de même nature que le Pascal, il utilise les fonctions comme if ... then ... else ... (si ... alors ... sinon ...)
Peu utilisé par les automaticiens.

```
IF %M0 THEN
  FOR %MW99 := 0 TO %I DO
    IF %M00 [%MW99] > 0 THEN
      %MW10 := %M00 [%MW99];
      %MW11 := %MW99;
      %M1 := TRUE;
    EXIT; (*Sortie de la boucle FOR*)
  ELSE
    %M1 := FALSE;
```

V.6. Traitement du programme automate



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