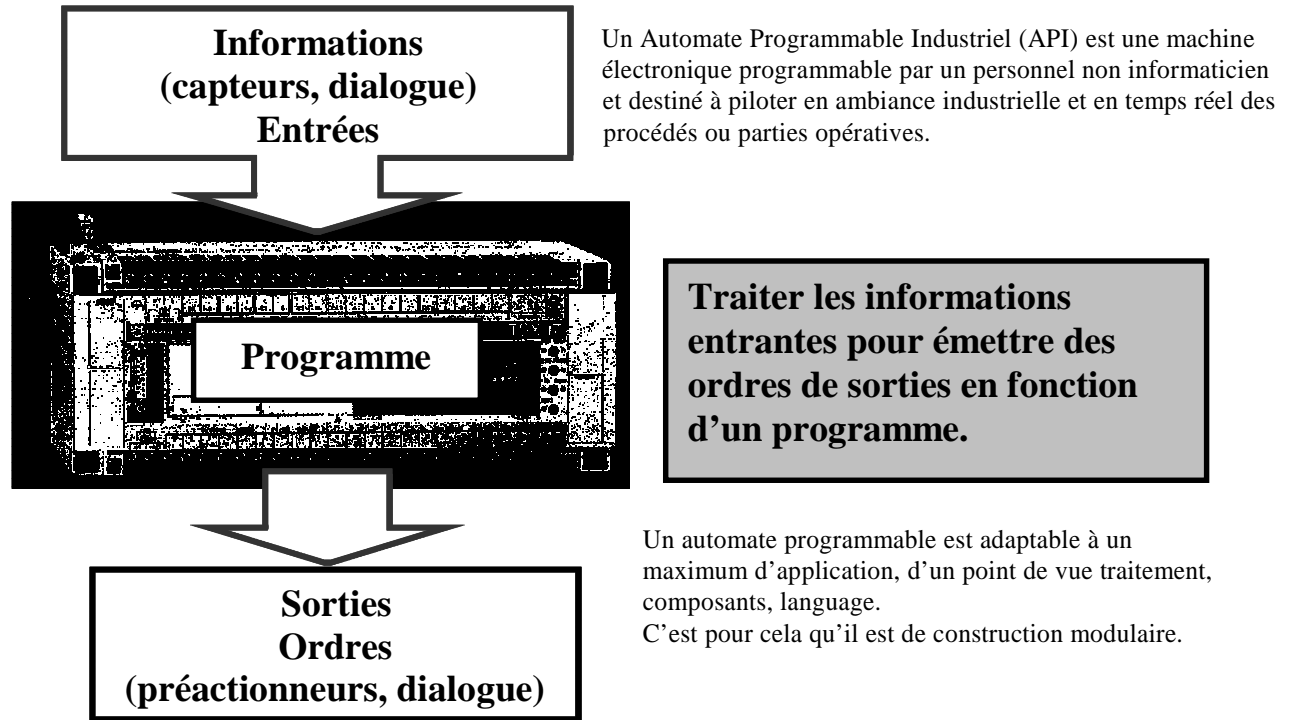
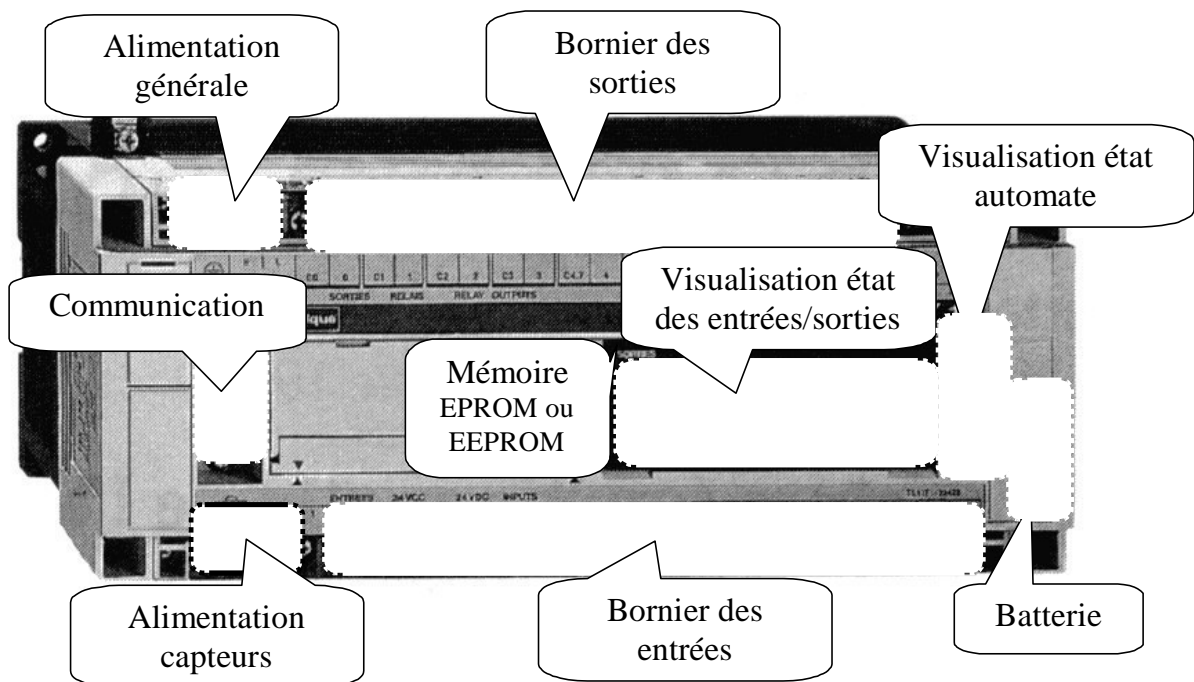


# L'Automate Programmable Industriel

## 1. Définition



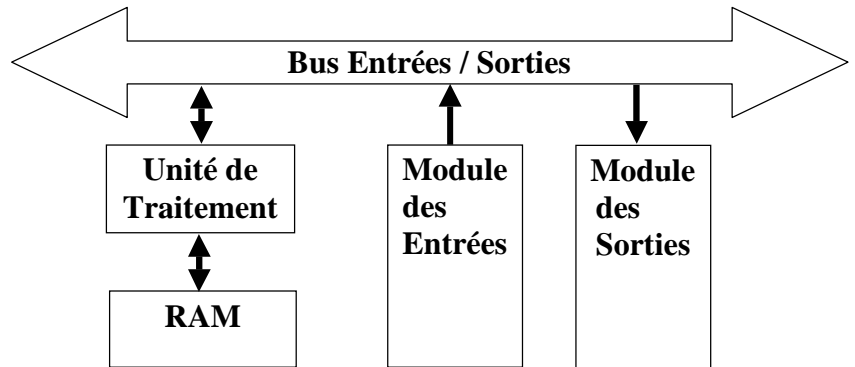
## 2. Structure générale



### 3. Principe de fonctionnement

Le traitement à lieu en quatre phases :

- **Phase 1 : Gestion du système**  
• Autocontrôle de l'automate
- **Phase 2 : Acquisition des entrées**  
Prise en compte des informations du module d'entrées et écriture de leur valeur dans RAM (zone DONNEE).
- **Phase 3 : Traitement des données**  
Lecture du programme (située dans la RAM programme) par l'unité de traitement, lecture des variables (RAM données), traitement et écriture des variables dans la RAM données.
- **Phase 4 : Emissions des ordres**  
Lecture des variables de sorties dans la RAM données et transfert vers le module de sorties.



### 4. Caractéristiques techniques

Les caractéristiques principales d'un API sont :

<ul style="list-style-type: none"> <li>• Compact ou modulaire</li> <li>• Tension d'alimentation</li> <li>• Taille mémoire</li> <li>• Temps de scrutation</li> </ul>	<ul style="list-style-type: none"> <li>• Sauvegarde (EPROM, EEPROM, pile, ...)</li> <li>• Nombre d'entrées / sorties</li> <li>• Modules complémentaires (analogique, communication,...)</li> <li>• Langage</li> </ul>
---	---

#### 4.1 Unité Centrale

L'unité centrale est le regroupement du processeur et de la mémoire centrale. Elle commande l'interprétation et l'exécute des instructions programmes. Les instructions sont effectuées les unes après les autres, séquencées par une horloge.  
Exemple: Si deux actions doivent être simultanées, l'API les traite successivement.

##### Caractéristiques principales :

- Vitesses de traitement : C'est la vitesse de l'UC pour exécuter 1 K-instructions logiques. (10 à 20 ms/Kmots).
- Temps de réponse : scrutation des entrées, vitesse de traitement et affectation des sorties.

#### 4.2 Mémoire

Deux types de mémoire cohabitent :

- **La mémoire Langage** où est stocké le langage de programmation. Elle est en général figé, c'est à dire en lecture seulement. (ROM : mémoire morte)
- **La mémoire Travail** utilisable en lecture-écriture pendant le fonctionnement c'est la RAM (mémoire vive).

##### Attribution des zones mémoire travail en RAM

Nature des Inform.	Désignations	Exploitation	Zones Mémoires
Etats des Capteurs	Variable d'entrée	Evolution de leur valeur en fonction du déroulement du cycle	Zone mémoire des Données
Ordres aux préactionneurs	Variable de sortie		
Résultats de fonctions comptage, tempo...	Variable Interne et / ou Variable mot		
Résultats intermédiaires			
Instructions du cycle dans l'API	Programme	Ecrit 1 fois et lu à chaque scrutation	Zone mémoire PROGRAMME

##### • Sauvegarde :

Sauvegarde de la RAM (programmes, configuration, données)		Sauvegarde Externe (programme, configuration)
1 heure minimum par pile interne	1an par pile externe	permanente par EPROM (effaçable par ultraviolet), EEPROM (effaçable par courant électrique)....

Le transfert de l'EPROM ou EEPROM vers la mémoire RAM de l'automate, s'effectue à chaque reprise secteur et si le contenu de celle-ci est différent.

### 4.3 Les Modules Entrées - Sorties

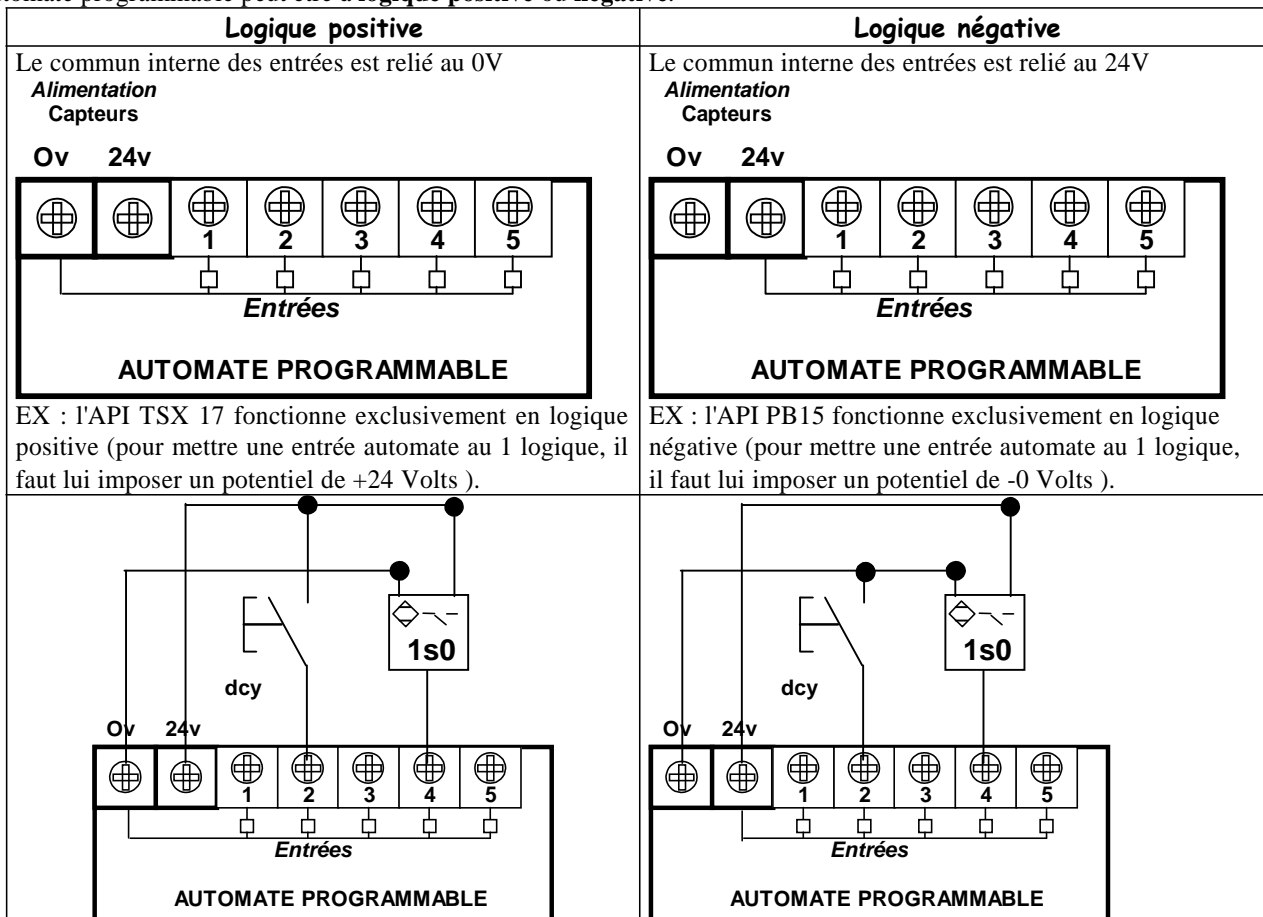
Module d'extension d'Entrées/Sorties TOR  
Module réseau : communication entre automate

Module d'extension d'Entrées Analogiques 0-10V Module  
d'extension de Sorties Analogiques 0-10V

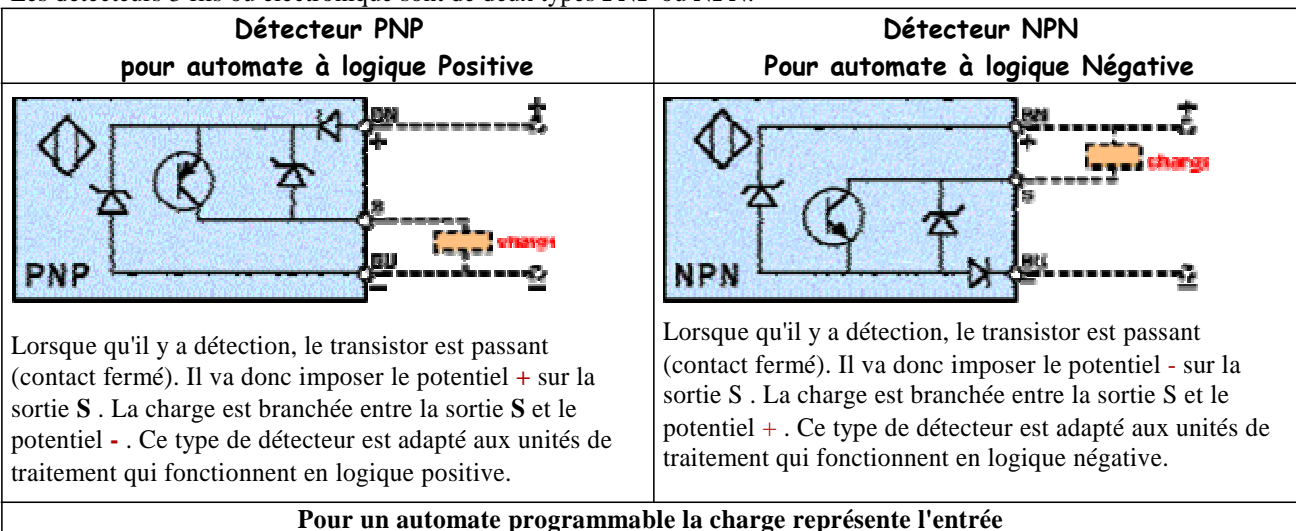
#### 4.3.1 Branchement des Entrées TOR

Le principe de raccordement consiste à envoyer un signal électrique vers l'entrée choisie sur l'automate dès que l'information est présente.

L'alimentation électrique peut être fourni par l'automate (en général 24V continu) ou par une source extérieure. L'automate programmable peut être à **logique positive** ou **logique négative**.



Les détecteurs 3 fils ou électronique sont de deux types PNP ou NPN.



### 4.3.2 Branchement des sorties

Le principe de raccordement consiste à envoyer un signal électrique vers le préactionneur connecté à la sortie choisie de l'automate dès que l'ordre est émis.

L'alimentation électrique est fournie par une source extérieure à l'automate programmable.

