

Les Automates Programmables Industriels (API)

Introduction

Les Automates Programmables Industriels (API) sont apparus aux Etats-Unis vers 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 Automate Programmable Industriel (API) est une machine électronique programmable par un personnel non informaticien et destiné à piloter en ambiance industrielle et en temps réel des procédés industriels. Un automate programmable est adaptable à un maximum d'application, d'un point de vue traitement, composants, langage. C'est pour cela qu'il est de construction modulaire.

Il est en général manipulé par un personnel électromécanicien. Le développement de l'industrie à entraîner une augmentation constante des fonctions électroniques présentes dans un automatisme c'est pour ça que l'API s'est substitué aux armoires à relais en raison de sa souplesse dans la mise en œuvre, mais aussi parce que dans les coûts de câblage et de maintenance devenaient trop élevés.

Pourquoi 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 (rentabilité, compétitivité) du système.
- Améliorer la flexibilité de production.
- Améliorer la qualité du produit.
- Adaptation à des contextes particuliers tel que les 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...

Structure générale des API

Les systèmes API sont principalement disponibles sous deux formes : en boîtier unique et en version modulaire/rack. Les systèmes non modulaires ont un nombre d'entrées sorties fixe (entre 10 et 30 E/S), avec souvent des performances limitées, ce sont les gammes les moins onéreuses. La majorité des installations comporte une solution modulaire, permettant grâce à des extensions d'étendre les E/S de l'automate ainsi que les interfaces de communications.

Un API modulaire est constitué de modules séparés pour : l'alimentation, le processeur, les entrées/sortie, les interfaces de communication. Les modules sont branchés les uns à la suite des autres dans un rack. Il suffit d'insérer un module sur le rack et de le configurer dans le logiciel pour l'ajouter, le rack de fond fournit le bus de communication et l'alimentation du module.

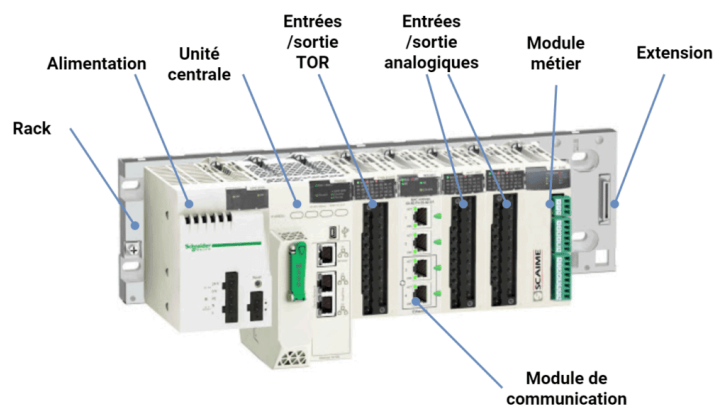


Figure 1: Modicon M580

Les différents sous-ensembles sont :

- **Le rack/Châssis** : Certains châssis sont connus pour un montage en fond de panier, tandis que d'autres le sont pour un montage en rack. Il existe aussi des automates modulaires, sans châssis, ou les modules se connectent les uns aux autres.
- **Unité centrale** : Les processeurs sont disponibles dans plusieurs variantes en termes de capacité d'E/S, de mémoires. Certaines unités peuvent comporter une ou plusieurs interfaces de communication, ainsi que quelque entrée/sortie. Un connecteur de communication permet la programmation de l'automate. L'unité centrale comporte régulièrement un mémoire externe (carte SD) et une pile pour garder la sauvegarde.
- **Entrées-sorties TOR** : Modules d'entrées/sortie digitale (tout ou rien), il existe en version 8, 16, 32 ou 64 E/S. Il existe avec des variantes de tensions, ou de courant admissibles. Les sorties peuvent être à commande à relais (puissance), ou à transistors.
- **Entrées-sorties analogiques** : Les modules d'entrées-sorties analogiques réalisent les conversions A/N et N/A, avec une résolution allant jusqu'à seize bits. Nous avons plusieurs gammes de tension/courant disponibles, les plus utilisés étant le 0-10V et le 4-20mA.
- **Modules de communication** : Des modules de communication peuvent être utilisés pour augmenter le nombre de ports de communication du processeur ou utiliser d'autres protocoles de communications.
- **Module métier** : les fabricants proposent des modules pour une utilisation plus spécifique ; comme le commande d'axe permettant d'assurer le positionnement avec précision d'élément mécanique selon un ou plusieurs axes, le comptage rapide permettant d'acquérir des informations de fréquences élevées incompatibles avec le temps de traitement de l'automate ; la mesure de température.
- **Extensions** : il est possible d'étendre son rack par un autre rack, des entrées/sortie déportées par un bus de communication, ou un autre automate en esclaves.

Structure interne d'un automate programmable industriel (API)

Afin de recevoir les informations concernant l'état du système et de commander les préactionneurs selon le programme inscrit dans sa mémoire,

De manière générale, l'automate programmable industriel est composé de plusieurs éléments de base décrite ci-dessous :

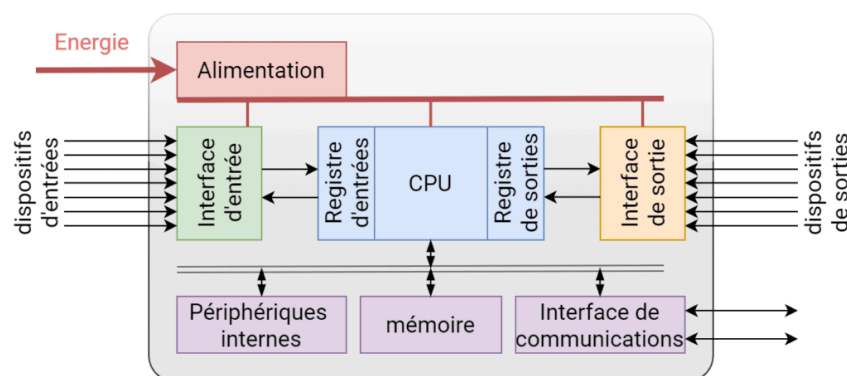


Figure 2: Modicon M580

- **Un processeur (ou Central Processing Unit, CPU)** : Son rôle consiste à traiter les instructions qui constituent le programme de fonctionnement de l'application, à gérer les entrées et sorties, à surveiller et diagnostiquer l'automate (par des tests lancés régulièrement), à mettre en place un dialogue avec le terminal de programmation.
- **Une mémoire** : Elle permet le stockage des instructions constituant le programme de fonctionnement ainsi que diverses informations. Il peut s'agir de mémoire vive RAM (modifiable à

volonté, mais perdue en cas de coupure de tension) ou de mémoire morte EEPROM (seule la lecture est possible).

- **Des interfaces entrées/sorties** : Elles permettent au processeur de recevoir et d'envoyer des informations. Ces dispositifs d'entrée et sortie peuvent produire des signaux discrets, numériques (ce sont des sorties de type « tout ou rien ») ou analogiques. Les dispositifs qui génèrent des signaux discrets ou numériques sont ceux dont les sorties sont de type tout ou rien. Par conséquent, un interrupteur est un dispositif qui produit un signal discret : présence ou absence de tension. Les dispositifs numériques peuvent être vus comme des dispositifs discrets qui produisent une suite de signaux tout ou rien. Les dispositifs analogiques créent des signaux dont l'amplitude est proportionnelle à la grandeur de la variable surveillée.
- **L'alimentation** : est indispensable puisqu'elle convertit une tension alternative en une basse tension continue (24V) nécessaire au processeur et aux modules d'entrées-sorties. L'alimentation ne fait pas toujours partie de l'automate qui sera donc directement alimenté par une basse tension.
- **Interface de communication** : est utilisée pour recevoir et transmettre des données sur des réseaux de communication qui relient l'API à d'autres API distants ou à des équipements en fonction des protocoles choisis (voir le chapitre Protocole de communication industriel). Elle est impliquée dans des opérations telles que la vérification d'un périphérique, l'acquisition de données, la synchronisation entre des applications et la gestion de la connexion.
- **Périphérique de programmation** : est utilisé pour entrer le programme dans la mémoire du processeur. Ce programme est développé sur le périphérique, puis transféré dans la mémoire de l'API.

Fonctionnement des API

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. Généralement les automates programmables industriels ont un fonctionnement cyclique (Figure 3). Le microprocesseur réalise toutes les fonctions logiques ET, OU, les fonctions de temporisation, de comptage, de calcul... Il est connecté aux autres éléments (mémoire et interface E/S) par des liaisons parallèles appelées "**BUS**" qui véhiculent les informations sous forme binaire. Lorsque le fonctionnement est dit synchrone par rapport aux entrées et aux sorties, le cycle de traitement commence par la prise en compte des entrées qui sont figées en mémoire pour tout le cycle.

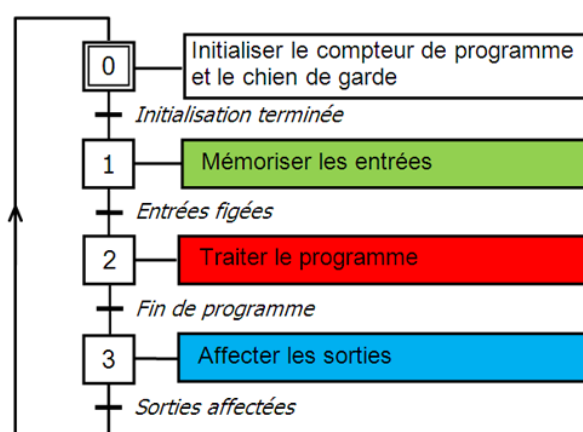


Figure 3: Fonctionnement cyclique d'un API

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. Dans ce cas, le temps de réponse à une

variation d'état d'une entrée peut être compris entre un ou deux temps de cycle (durée moyenne d'un temps de cycle est de 5 à 15 ms (Figure .4).

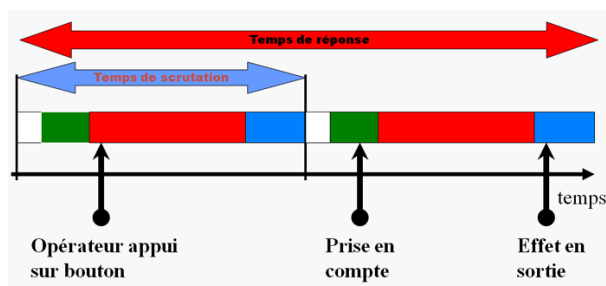


Figure 4: Temps de scrutation vs Temps de réponse

Il existe d'autres modes de fonctionnement, moins courants :

- synchrone par rapport aux entrées seulement ;
- asynchrone.

Réseaux d'automates

Principe:

Avec le développement des systèmes automatisés et de l'électronique, la recherche de la baisse des coûts et la nécessité actuelle de pouvoir gérer au mieux la production et à partir du moment où tous les équipements sont de type informatique, il devient intéressant de les interconnecter à un mini-ordinateur ou à un automate de supervision . L'interconnexion entre deux automates peut être réalisée très simplement en reliant une ou plusieurs sorties d'un automate à des entrées de l'autre et vice-versa Cette méthode ne permet pas de transférer directement des variables internes d'un automate sur l'autre, de sorte que celles-ci doivent être converties par programme en variables de sortie avant leur transfert. Elle devient coûteuse en nombre d'entrées/sorties mobilisé pour cet usage et lourde du point de vue du câblage, lorsque le nombre de variables qui doivent être échangées devient important.

Bus de terrain

Pour diminuer les coûts de câblage des entrées / sorties des automates, sont apparus les bus de terrains. L'utilisation de blocs d'entrées / sorties déportés a permis tout d'abord de répondre à cette exigence.

Les interfaces d'entrées/sorties sont déportées au plus près des capteurs. Avec le développement technologique, les capteurs, détecteurs ... sont devenus intelligents“ et ont permis de se connecter directement à un bus.

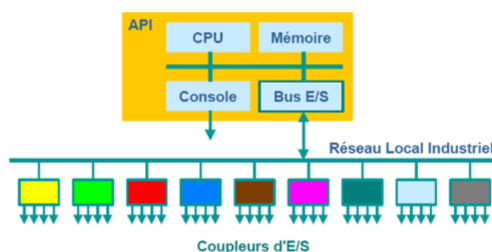


Figure 5: Interconnexion par entrées/sorties déportées

Plusieurs protocoles de communication et des standards sont apparus pour assurer le “multiplexage” de toutes les informations en provenance des capteurs / préactionneurs par exemple le bus ASi (Actuators Sensors interface) est un bus de capteurs/actionneurs de type Maître / Esclave qui permet

de raccorder 31 esclaves (capteurs ou préactionneurs) sur un câble spécifique (deux fils) transportant les données et la puissance. Ce bus est totalement standardisé et permet d'utiliser des technologies de plusieurs constructeurs.

- **Avantages des bus de terrain :**

- Réduction des coûts de câblage et possibilité de réutiliser le matériel existant
- Réduction des coûts de maintenance

- **Inconvénients des bus de terrain :**

- Taille du réseau limitée
- Latence dans les applications à temps critique
- Coût global

Différents types de réseaux d'automates :

Réseau en étoile :

Un centre de traitement commun échange avec chacune des autres stations. Deux stations ne peuvent pas échanger directement entre elles (Figure .6). Exemple le réseau de terrain BITBUS de la société INTEL.

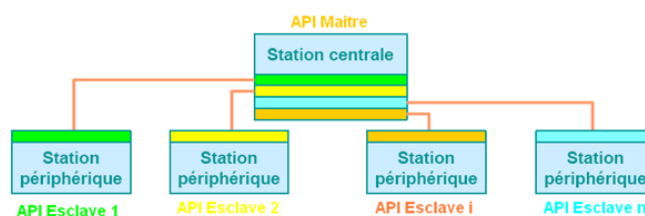


Figure 6: Topologie étoile

- **Avantages :**

- Grande vitesse d'échange.
- Différent types de supports de transmission.
- Pas de gestion d'accès au support.

- **Inconvénients :**

- Coût global élevé.
- Evolutions limitées.
- Tout repose sur la station centrale.

Réseau en anneau :

Chaque station peut communiquer avec sa voisine. Cette solution est intéressante lorsqu'une station doit recevoir des informations de la station précédente ou en transmettre vers la suivante (Figure .7).

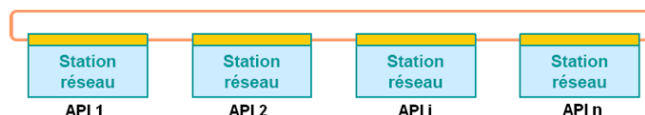


Figure 7: Topologie Anneau

- **Avantages :**

- Signal régénéré donc fiable.
- Contrôle facile des échanges (le message revient à l'émetteur).

- **Inconvénients :**

- Chaque station est bloquante.
- Une extension interrompt momentanément le réseau.

Réseau hiérarchisé :

C'est la forme de réseaux la plus performante. Elle offre une grande souplesse d'utilisation, les informations pouvant circuler entre-stations d'un même niveau ou circuler de la station la plus évoluée (en général un ordinateur) vers la plus simple, et réciproquement (Figure .8).

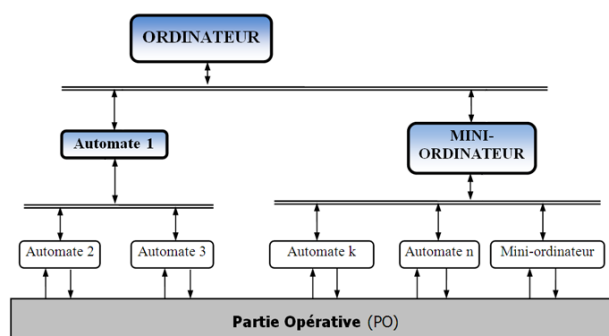


Figure 8: Réseau hiérarchisé

Sécurité .

Les systèmes automatisés sont, par nature, source de nombreux dangers (tensions utilisées, déplacements mécaniques, jets de matière sous pression ...). Placé au coeur du système automatisé, l'automate se doit d'être un élément fiable car un dysfonctionnement de celui-ci pourrait avoir de graves répercussions sur la sécurité des personnes, de plus les coûts de réparation et un arrêt de la production peuvent avoir de lourdes conséquences sur le plan financier. Aussi, l'automate fait l'objet de nombreuses dispositions pour assurer la sécurité :

- **Contraintes extérieures** : l'automate est conçu pour supporter les différentes contraintes du monde industriel et à fait l'objet de nombreux tests normalisés.
- **Coupures d'alimentation** : l'automate est conçu pour supporter les coupures d'alimentation et permet, par programme, d'assurer un fonctionnement correct lors de la réalimentation (reprises à froid ou à chaud)
- **Mode RUN/STOP** : Seul un technicien peut mettre en marche ou arrêter un automate et la remise en marche se fait par une procédure d'initialisation (programmée)
- **Contrôles cycliques** :
 - Procédures d'autocontrôle des mémoires, de l'horloges, de la batterie, de la tensions d'alimentation et des entrées / sorties.
 - Vérification du temps de scrutation à chaque cycle appelée Watchdog (chien de garde), et enclenchement d'une procédure d'alarme en cas de dépassement de celui-ci (réglé par l'utilisateur).
- **Visualisation** : Les automates offrent un écran de visualisation où l'on peut voir l'évolution des entrées / sorties

Les normes interdisent la gestion des arrêts d'urgence par l'automate ; celle-ci doit être réalisée en technologie câblée.