
2.

Structures des Automates Programmables

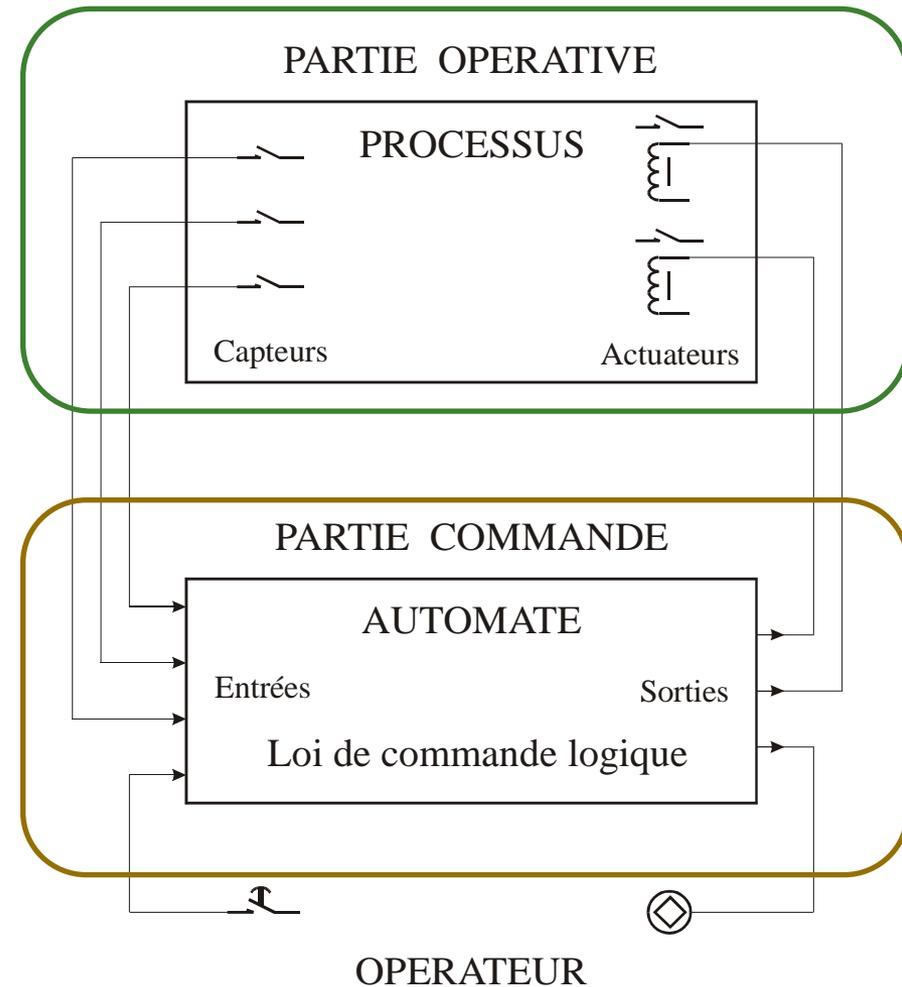
Pierre Duysinx

Université de Liège

Année académique 2021-2022

Rôle d'un automate

- On distingue
 - La **partie opérative** = processus et la machine physique
 - La **partie commande** = l'automatisme et le système programmé
- Entrées = capteurs
 - Fin de courses, détecteurs de présences, etc.)
- Sorties = actionneurs
 - Contacteurs, relais, commandes de vanne de verrin, robots, etc.



Rôle d'un automate

- Le rôle d'un automate est de réagir aux changements d'état de ses entrées en modifiant l'état de ses sorties selon une loi de contrôle déterminée **a priori**.
- On distingue
 - **Programme combinatoire**: à chaque instant, les sorties sont déterminées uniquement par les entrées (pas d'effet mémoire)
 - **Programme séquentiel**: les sorties sont déterminées par les entrées et un nombre fini de variables logiques internes tenant compte de l'évolution du système

Logique câblée

- **Logique câblée** → Réalisation de la loi de contrôle en interconnectant judicieusement des opérateurs matériels (électriques, pneumatiques, hydrauliques) réalisant des fonctions logiques de base.
- Opérateurs logiques de base
 - AND, OR, NOT
 - NAND, NOR
 - Relais normalement ouvert / normalement fermé

Exemple

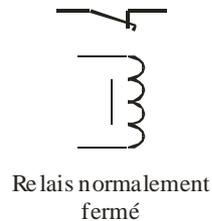
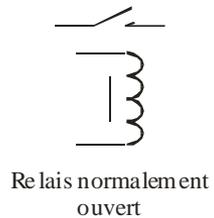
Exemple de fonction logique combinatoire

$$O1 = (\bar{I1} \cdot I2 + I3) \cdot (I4 + I5)$$

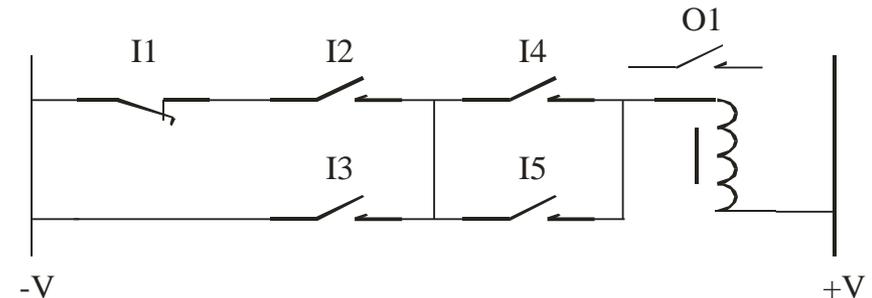
$$O1 := ((\text{NOT } I1 \text{ AND } I2) \text{ OR } I3) \text{ AND } (I4 \text{ OR } I5)$$

Solution à relais

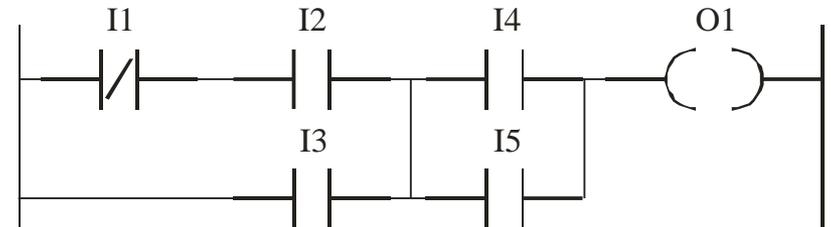
Opérateurs disponibles



Réalisation de la fonction



Représentation symbolique



Logique programmée

- Idée de la **logique programmée** = n'utiliser qu'un seul jeu d'opérateurs de base (unité logique) permettant de réaliser n'importe quelle fonction logique donnée par combinaison de ces opérateurs de base
- Exécution séquentielle des opérations logiques élémentaires
→ Nécessité d'un balayage des opérations à une cadence suffisamment rapide pour donner l'illusion d'une exécution parallèle et du caractère instantané de l'évaluation.
- La manière de balayer est décrite par une suite d'instructions mémorisées c.-à-d. un programme

Exemple

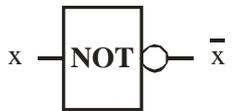
Exemple de fonction logique combinatoire

$$O1 = (\bar{I1} \cdot I2 + I3) \cdot (I4 + I5)$$

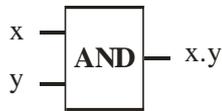
$$O1 := ((\text{NOT } I1 \text{ AND } I2) \text{ OR } I3) \text{ AND } (I4 \text{ OR } I5)$$

Solution électronique:

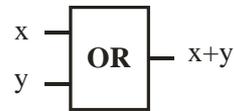
Opérateurs disponibles



Inversion
logique

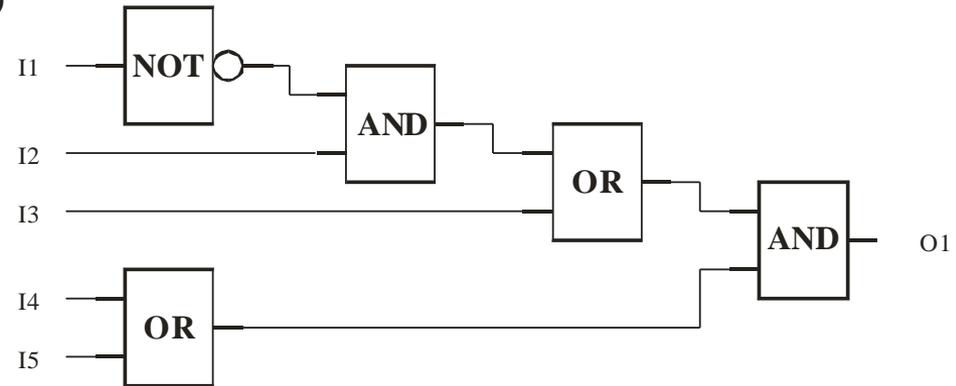


ET logique



OU logique

Réalisation de la fonction



Logique câblée v.s. programmée

- Logique câblée:
 - Inconvénients:
 - Volume du contrôleur proportionnel à la complexité du problème
 - Des modifications de la commande implique des modifications de câblage
 - Avantages:
 - Vitesse, car fonctionnement simultané des opérateurs

Logique câblée v.s. programmée

■ Logique programmée

□ Avantages

- Uniformisation du matériel: même matériel quel que soit la fonction logique à réaliser
- Facilité de modification de la loi de contrôle: il suffit de modifier le programme. Simplification de la maintenance!
- Faible liaison entre le volume matériel et la complexité du problème (effet simplement sur les entrées / sorties et taille mémoire)

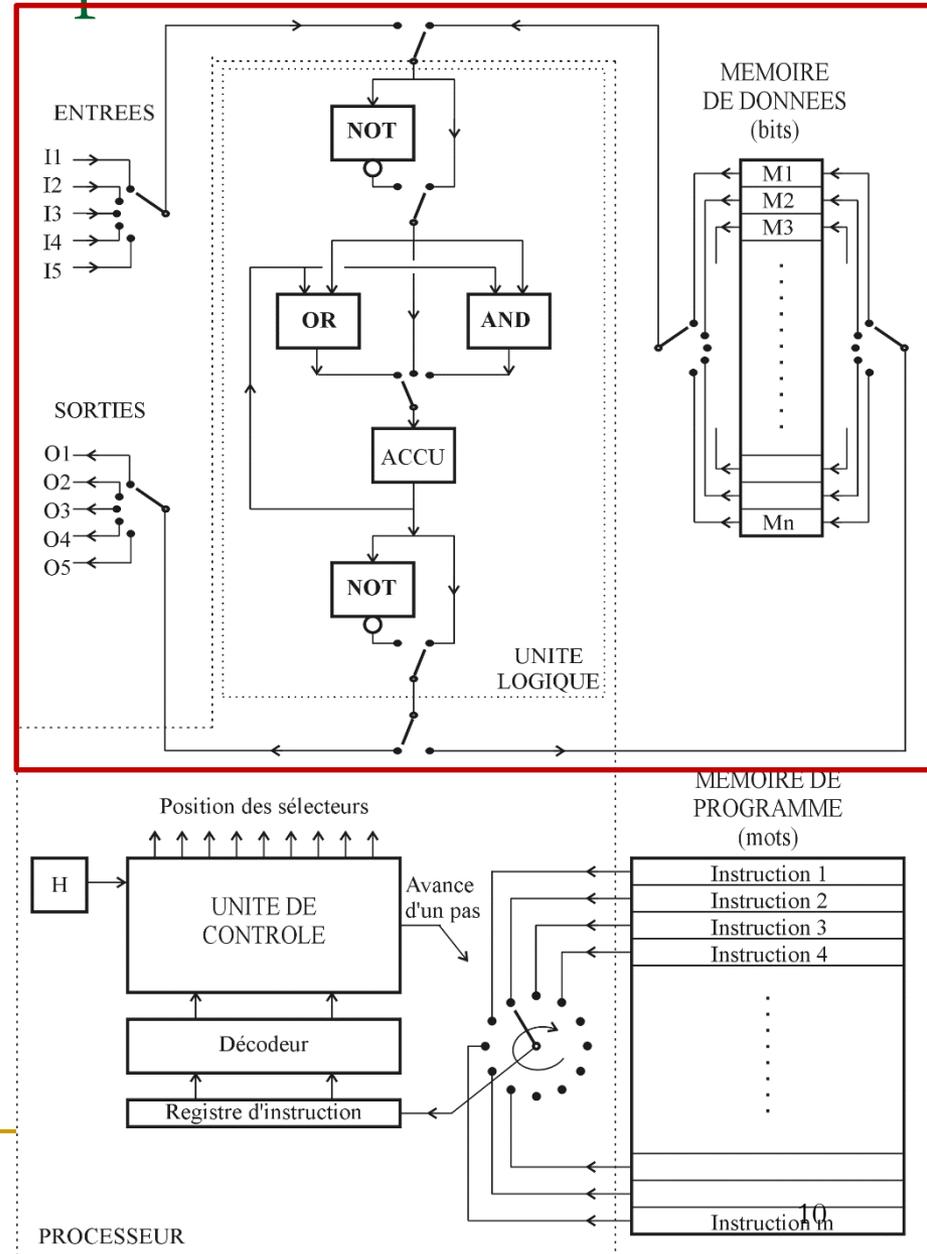
□ Inconvénients:

- Vitesse inversement proportionnelle à la complexité du problème. Ceci peut être une limitation pour des processus électromécaniques rapides

Structure fonctionnelle du processeur d'un automate programmable

■ L'UNITE LOGIQUE

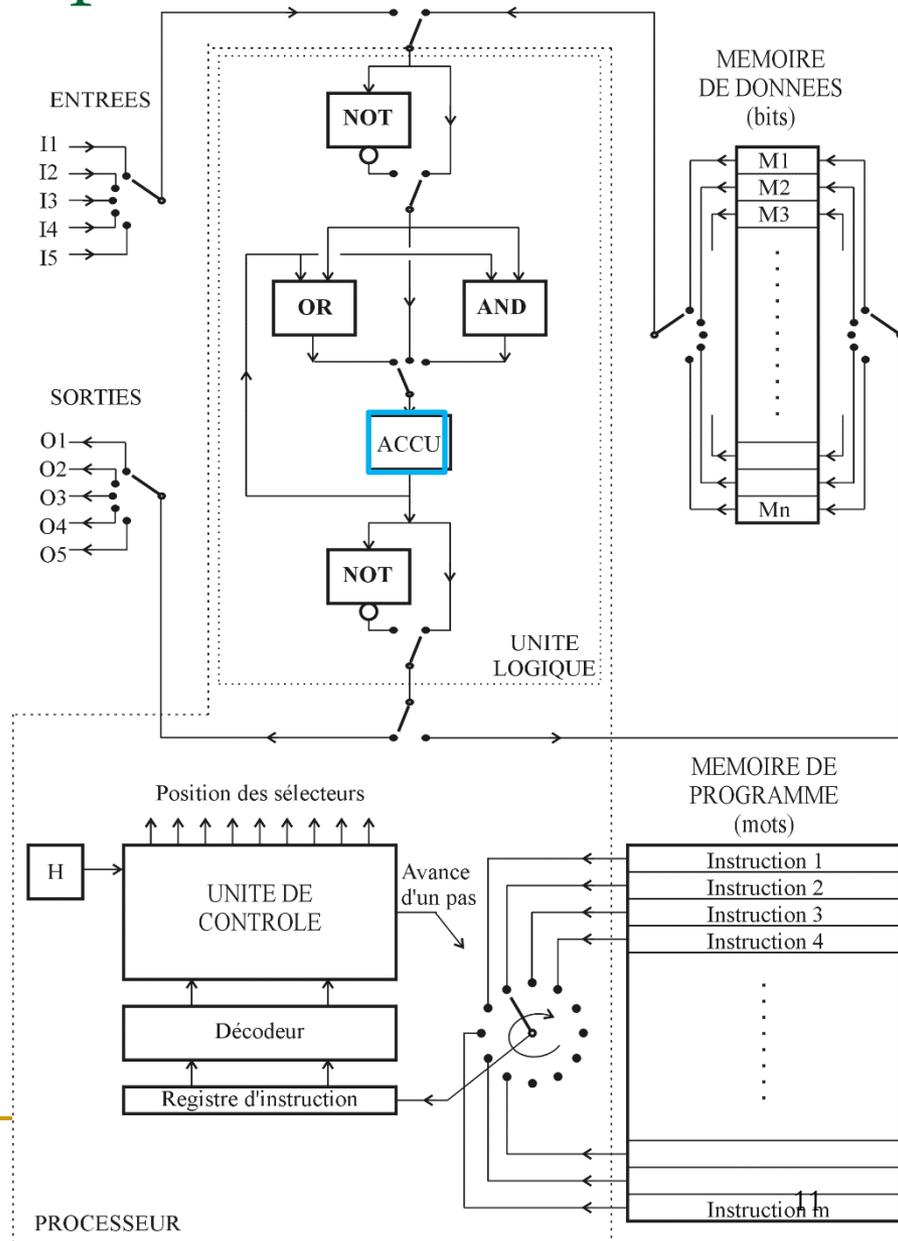
- Comporte 3 opérateurs logiques de base (AND/OR/NOT)
- Associée à une mémoire de bistables (bits) utilisées pour la mémorisation d'informations séquentielles et le stockage de résultats intermédiaires
- Une série de sélecteurs pour mettre en communication les opérateurs logiques, les entrées, les sorties et les mémoires.



Structure fonctionnelle du processeur d'un automate programmable

■ ACCUMULATEUR LOGIQUE

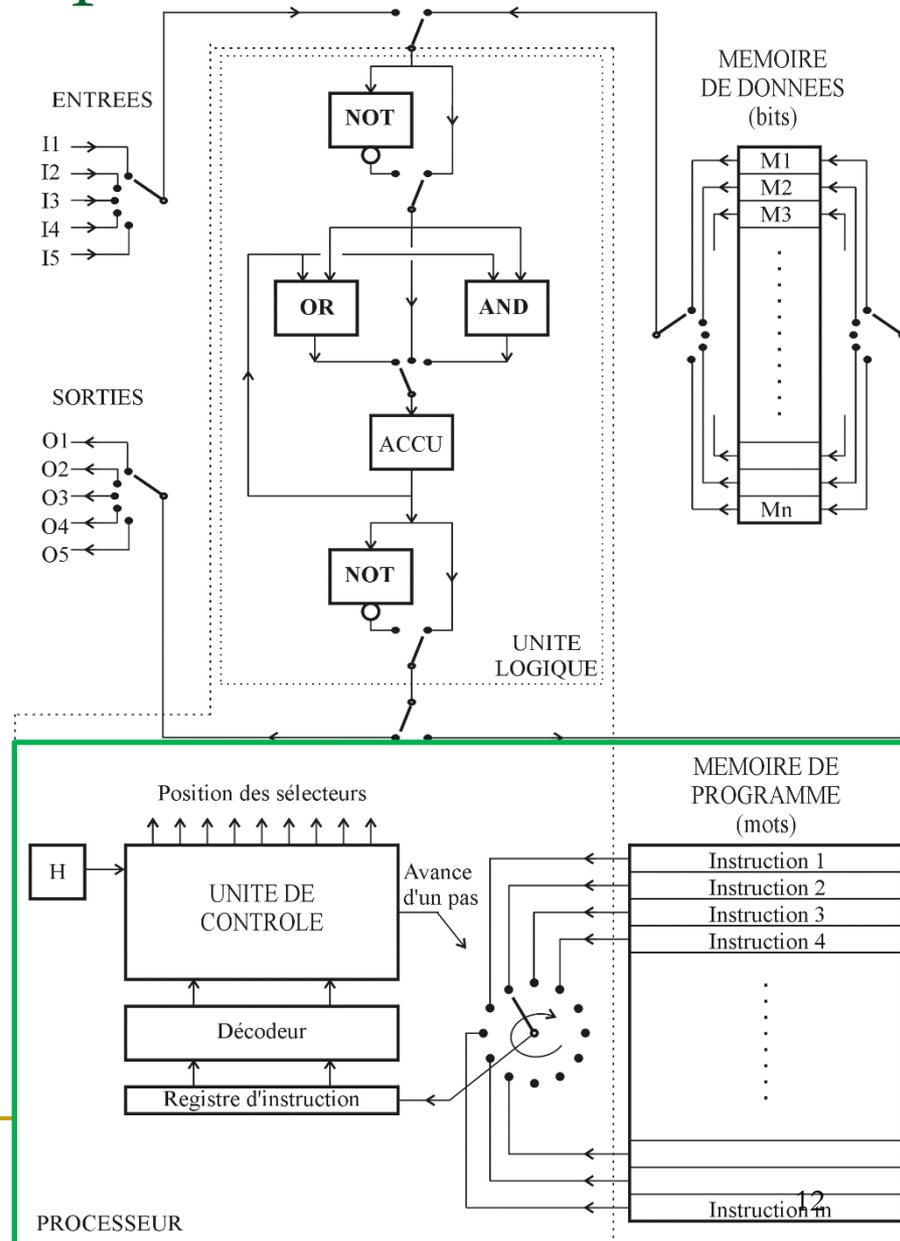
- Une cellule de mémoire particulière intervenant de manière privilégiée dans les opérations logiques
- Constitue une opérande obligée des opérateurs AND, OR
- Sert au **stockage** du résultat obtenu par l'opération logique



Structure fonctionnelle du processeur d'un automate programmable

■ UNITE DE CONTROLE

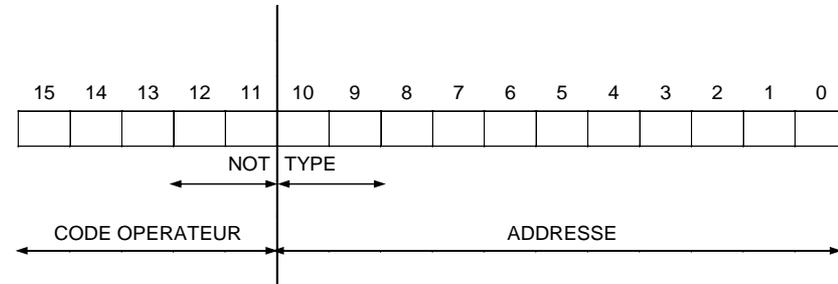
- Gère le fonctionnement de l'ensemble sous le contrôle des instructions stockées dans le programme
- Les instructions du programme sont menées successivement dans le registre d'instruction, puis sont décodées et en fonction des opérations qu'elles indiquent, l'unité de contrôle engendrent des ordres nécessaires à leur exécution
- Exécution séquentielle et cyclique du programme



Structure fonctionnelle du processeur d'un automate programmable

■ FORMAT DES INSTRUCTIONS

- Doit fournir toutes les indications nécessaires à l'exécution de l'opération
- Utilisation d'un bit formant un mot
- Exemple de format d'instruction
 - Inversion de l'opérande
 - Code opération
 - Trois types d'opérandes %I (entrée), %Q (sortie), %M (mémoire)
 - Adresse de l'opérande

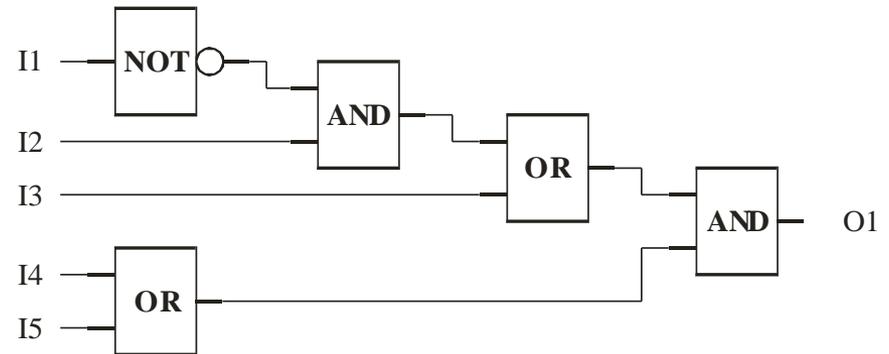


SYMBOLES			
Dir/Inv			
N		1	Inversion logique
-		0	Utilisation directe
Type			
I		0 1	Entrée
O		1 0	Sortie
M		1 1	Mémoire
Fonction			
AND	0 0 0 1		AND avec l'accumulateur
OR	0 0 1 0		OR avec l'accumulateur
LD	0 0 1 1		Charger l'opérande dans l'ACCU
ST	0 1 0 0		Placer l'ACCU dans l'opérande
RET	0 0 0 0		Boucler à la première instruction

Exemple

Exemple de fonction logique combinatoire

$O1 := ((NOT I1 AND I2) OR I3) AND (I4 OR I5)$

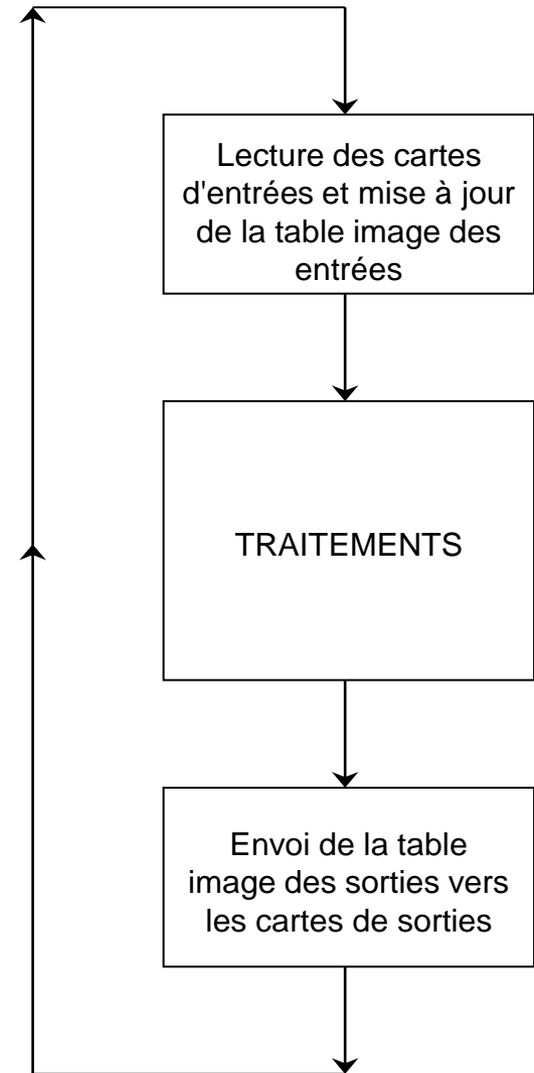


Solution programmée

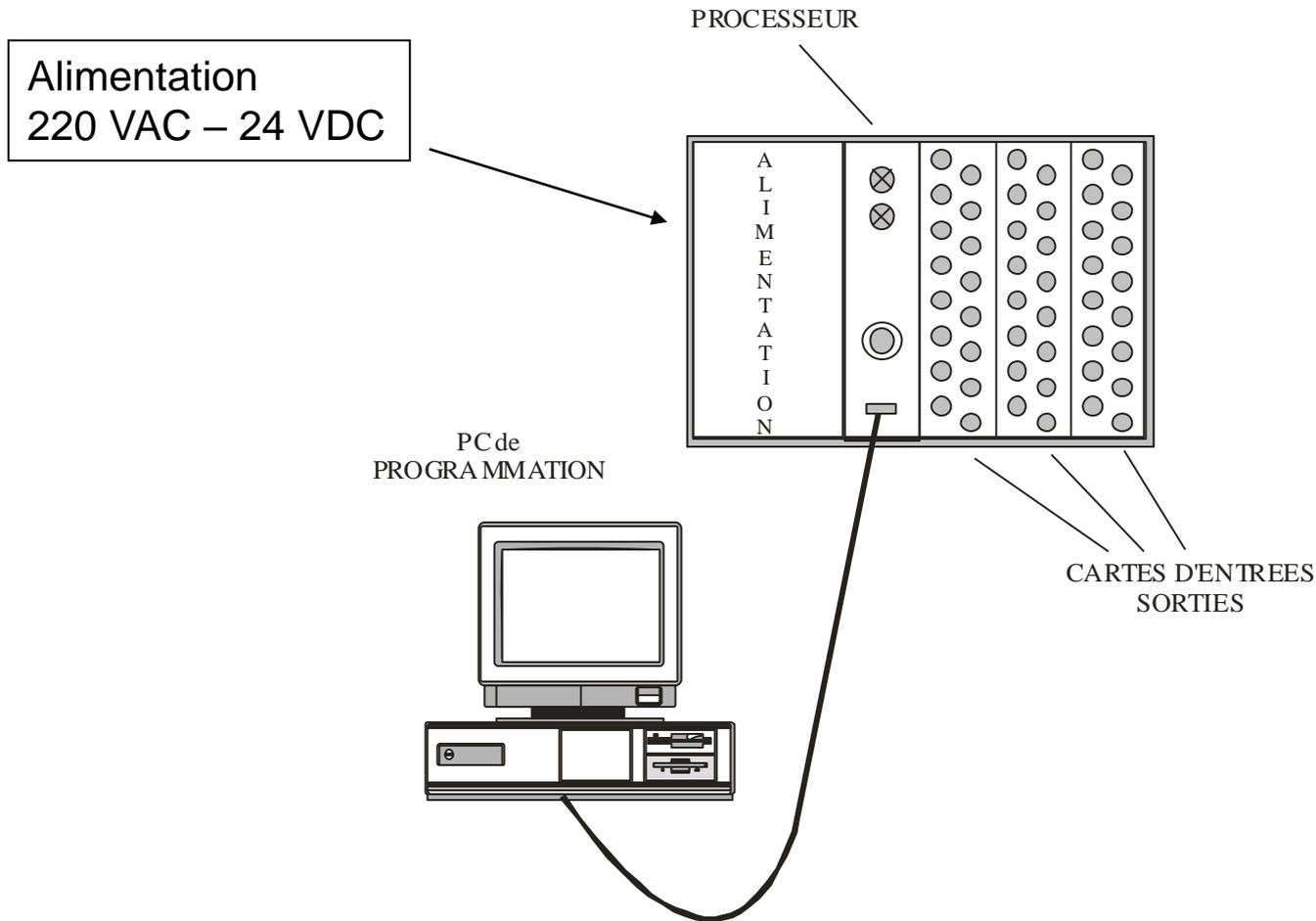
1	LDN	I1	mettre I1 inversé dans l'ACCU
2	AND	I2	ET entre ACCU et I2 (résultat dans ACCU)
3	OR	I3	OU entre ACCU et I3 (résultat dans ACCU)
4	ST	M1	sauver ACCU dans mémoire M1
5	LD	I4	mettre I4 dans ACCU
6	OR	I5	OU entre ACCU et I5 (résultat dans ACCU)
7	AND	M1	ET entre ACCU et M1 (résultat dans ACCU)
8	ST	O1	mettre la valeur de l'ACCU dans la sortie O1
9	RET		boucler sur l'instruction n° 1

Structure fonctionnelle du processeur d'un automate programmable

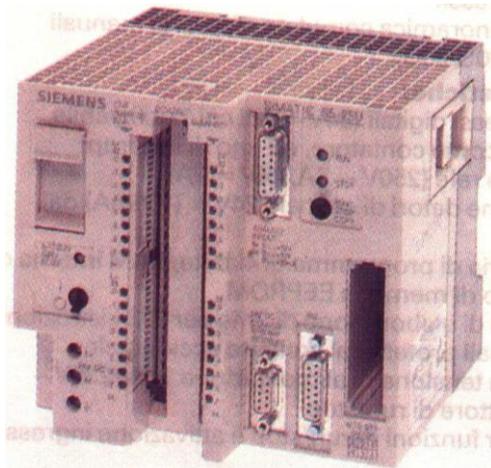
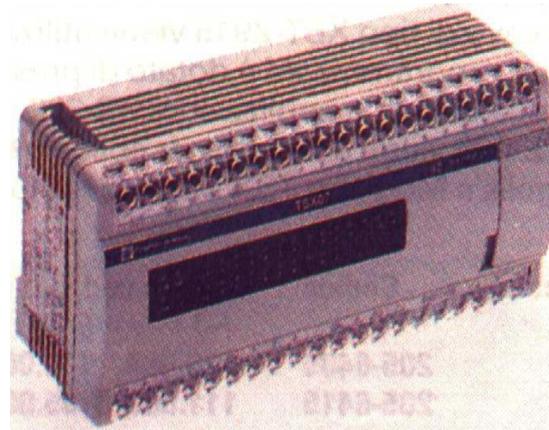
- Le langage machine
 - = configuration de 0 et de 1
- Le langage de programmation
 - Liste d'instructions
 - Proche du raisonnement de l'automaticien
- **Exécution cyclique** du programme



Structure matérielle d'un automate programmable



Structure matérielle d'un automate programmable

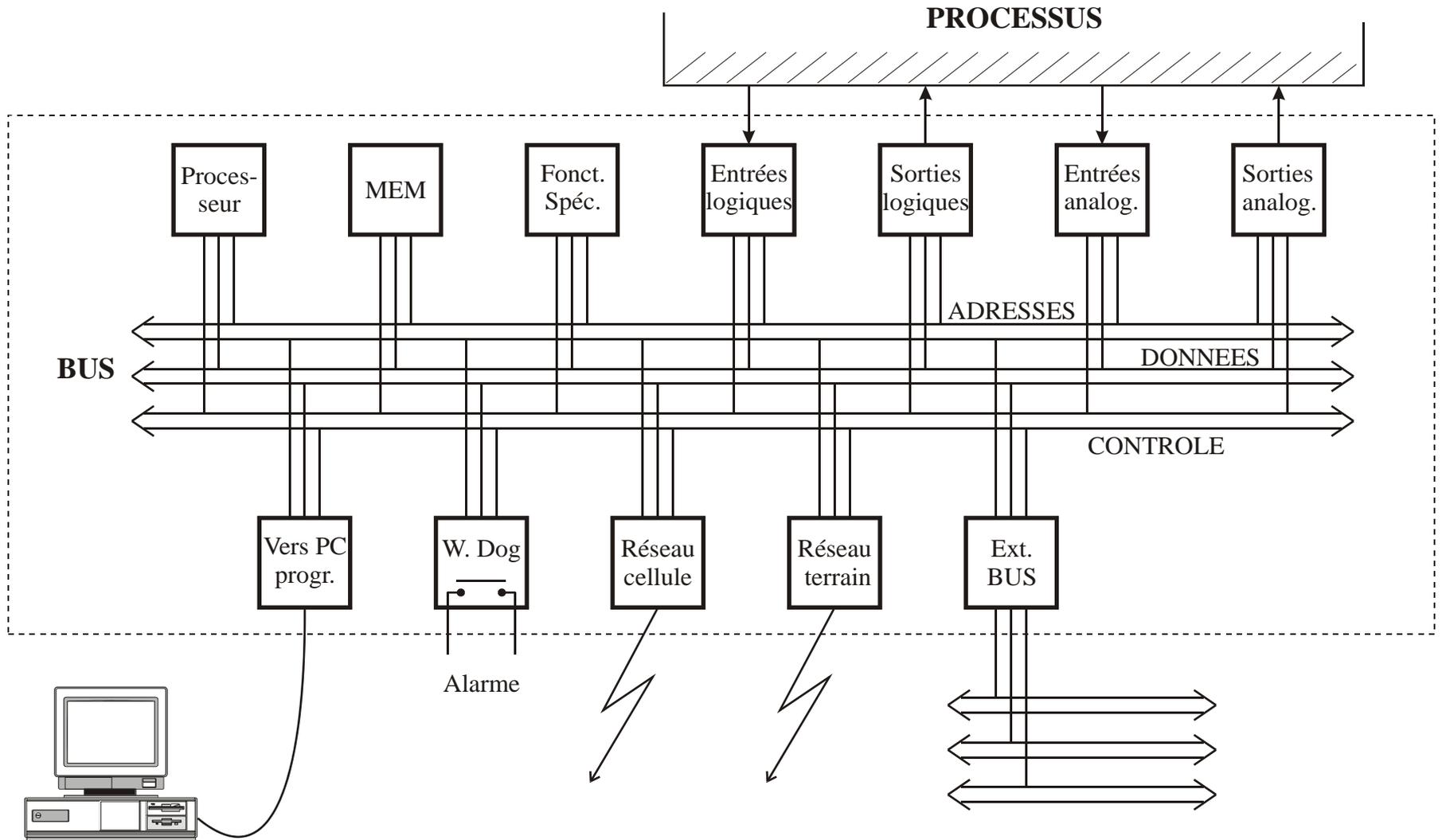


Technologie de réalisation

■ BUS D'ÉCHANGE

- On constate que:
 - Les échanges d'informations entre les différents éléments de l'automates (entrées, sorties, mémoires) transitent toujours par le processeur
 - A cause du fonctionnement séquentiel, il n'y a jamais qu'un seul élément à la fois en communication avec le processeur
- Il est possible d'utiliser un chemin commun et une procédure commune pour les échanges. Celui-ci est appelé **BUS**.
- Le BUS est constitué de lignes d'adresse, de lignes de données et de lignes de contrôle
- L'automate programmable adopte ainsi la structure typique d'un ordinateur.

Technologie de réalisation

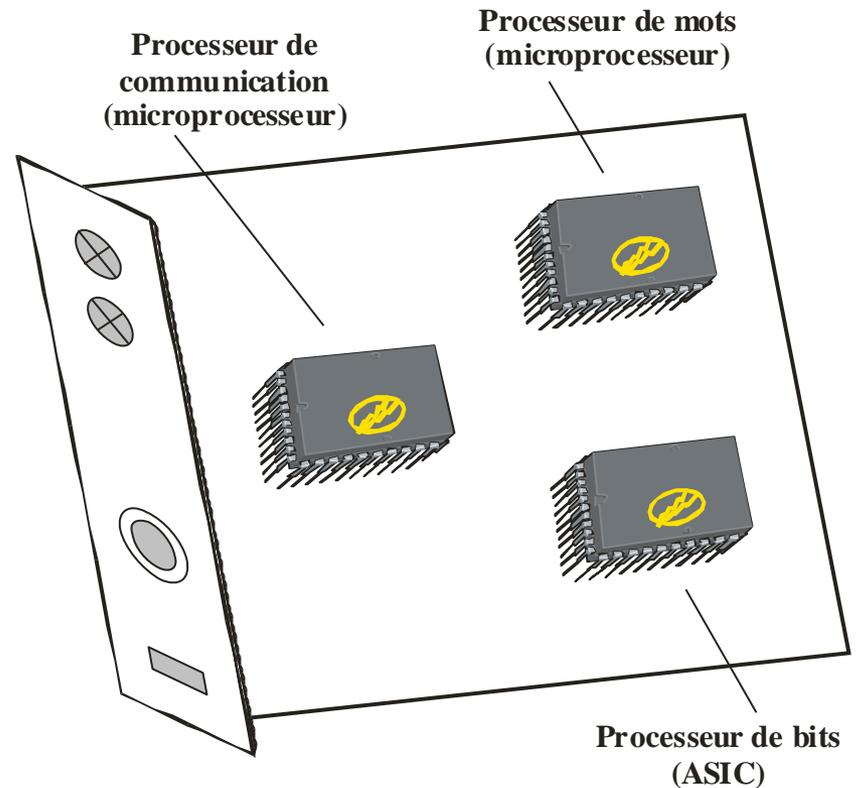


Processeur

- La logique programmée repose sur **le microprocesseur**
- Un jeu d'instructions se traduit par un jeu de sous-routines écrites dans le langage machine. Ce jeu de sous-routines est le logiciel système et est stocké dans la mémoire système de l'automate
- Utilisation de microprocesseurs standards
 - Inconvénients:
 - Les microprocesseurs utilisés sont prévus pour travailler sur des mots et pas des bits, ce qui pénalise le temps d'exécution
 - Avantages
 - Vitesses de traitement importantes
 - Facilité d'implémentation
 - Solution économique
 - Ouvre la porte à toute espèce d'extension du jeu d'instructions

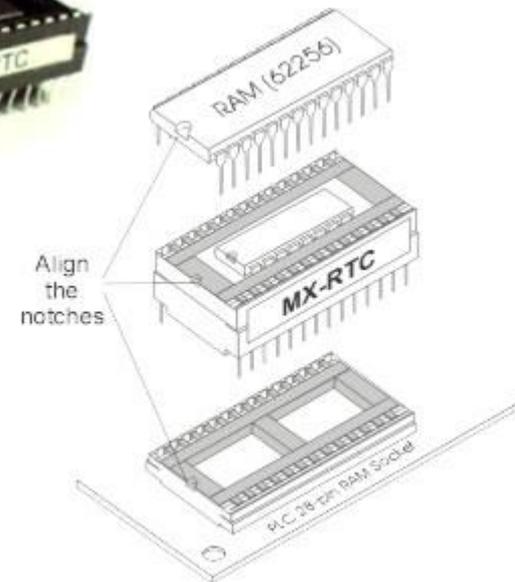
Processeur

- Structure d'une carte processeur moderne:
 - Un **micro-processeur standard** utilisé pour les fonctionnalités additionnelles (arithmétiques)
 - Un **circuit ASIC** (Application Specific Integrated Circuit) pour les **traitements purement logiques**
 - Un **processeur pour la prise en charge des communications** vers la console et les réseaux informatiques



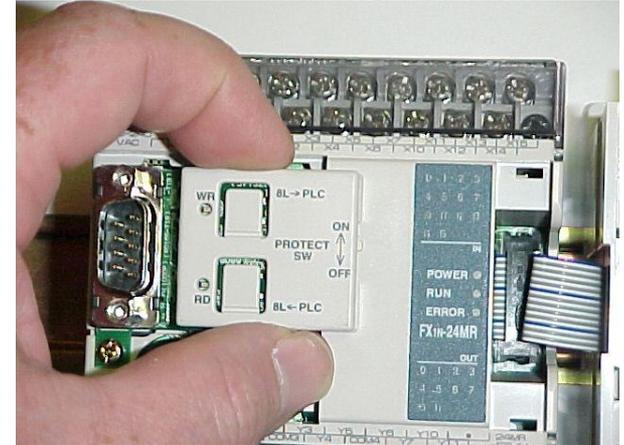
Mémoires

- Utilisation de mémoires à semi-conducteurs
- Données: RAM (Random Access Memory) mémoire vive
 - Lecture / écriture
 - Perte d'information en cas de coupure d'alimentation
 - Technique CMOS (faibles consommation)



Mémoires

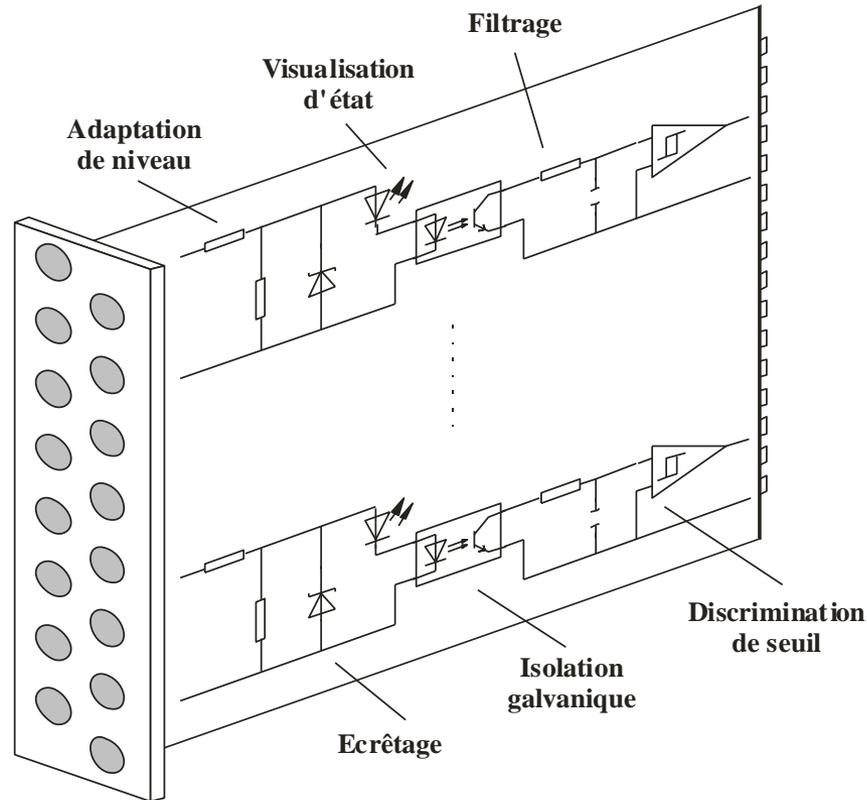
- Programme:
 - La mémoire doit pouvoir subir sans dommage les coupures de courant
 - EPROM (Erasable Programmable Read Only Memory)
 - Programmé et effacé après exposition aux UV
 - EEPROM (Electrically ...)
 - Effacement par voie électrique in situ
 - RAM gardées
- Systèmes:
 - EPROM ou PROM



Modules d'entrées / sorties industriels

- Modules réalisant l'interface entre les signaux du processus (vers actionneurs ou venant des capteurs) et les signaux du bus interne de l'automate
- Fonctions:
 - Découplage mécanique (bornier)
 - Découplage électrique entre les signaux du processus et l'électronique PLC
 - Synchronisation des transferts conformément aux procédures d'échange du BUS du PLC
- Entrées / sorties
 - Logiques
 - Analogiques pour les PLC qui le permettent

Modules d'entrées / sorties industrielles



Signaux de processus:

24, 48 V (CC ou AC)
110, 220 V (AC)

Signaux électroniques:

5 V (CC)

Types de signaux

- Deux grandes catégories de signaux
 - Analogiques
 - Digitaux ou logiques

- Signaux logiques
 - Signaux « tout ou rien » représentant l'état logique de contacteurs, boutons poussoirs, voyants lumineux, détecteurs de présence, etc.
 - Typiquement
 - Tensions continues: 12, 24, 48 ou 60 V
 - Tensions alternatives: 110 ou 220 V
 - Contacts libres de potentiel : information binaire = état ouvert ou fermé du circuit

Types de signaux

■ Signaux analogiques

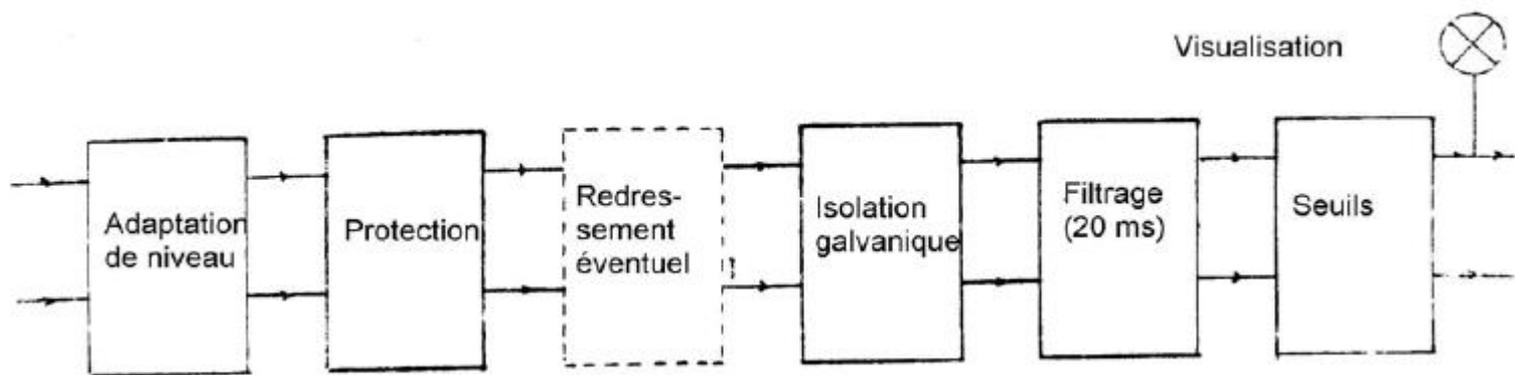
- Résultent de la **transduction** de grandeurs physiques: vitesse, température, pression, etc.
- Typiquement
 - Tensions de haut niveau: ex. -10 à +10 V
 - Tensions de bas niveau: ex. -50 à +50 mV
 - Courants: ex. 0 – 20 mA ou 4-20mA
- Remarques
 - Pour les courants, passage en tension dans l'automate via une résistance calibrée
 - Limite supérieure de 20 mA: empêche la formation d'arc de court circuit et limite les risques d'explosion
 - Limite inférieure de 4 mA permet de détecter la coupure du conducteur car 0 A est en dehors de la gamme.
 - Certains automates ont des entrées analogiques spécialement adaptées à certains types de capteurs courants: ex. sonde PTt100 de température

Rôle des interfaces industrielles

- Pour les entrées
 - Adaptation mécanique aux standards industriels de câblage (borniers à vis par exemple);
 - Adaptation électrique aux standards industriels (signaux en courant, en tension, signaux bas niveaux, hauts niveaux, etc.);
 - Protection contre les tensions de mode commun;
 - Atténuation des signaux parasites (bruit capté par les lignes de transmission, rebondissement des contacts, etc.).
- Pour les sorties
 - Idem entrées +
 - Prendre en compte les anomalies et pannes de la partie commande
 - de détecter de manière précoce les anomalies de fonctionnement du calculateur;
 - de placer les commandes dans un état de sécurité défini à l'avance.

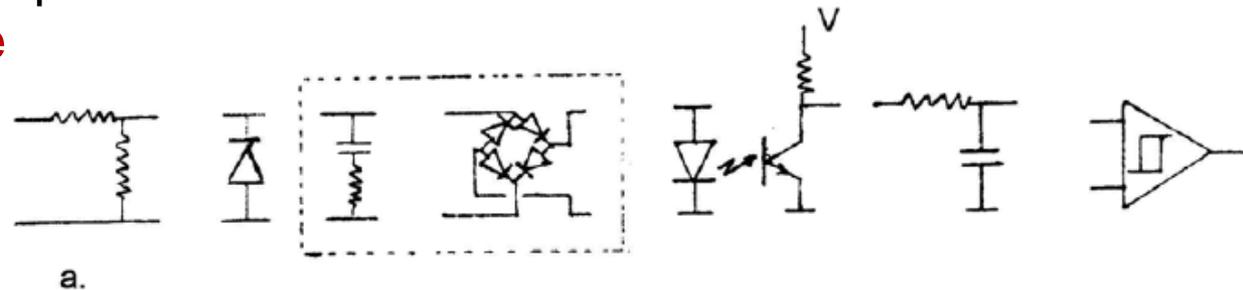
Conditionnement des entrées logiques

- Les principaux éléments d'un conditionneur **d'entrée logique** typique:
 - adaptation de niveau en fonction des standards;
 - protection contre les surtensions;
 - redressement éventuel dans le cas de signaux alternatifs;
 - isolation galvanique entre processus et calculateur;
 - filtrage destiné à éliminer les parasites hautes fréquences;
 - seuils de discrimination des états logiques 1 et 0;



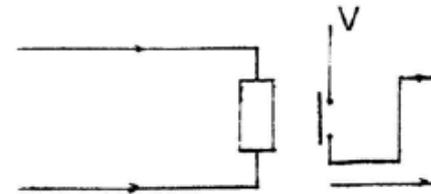
Conditionnement des entrées logiques

- Solutions technologiques:
isolation galvanique

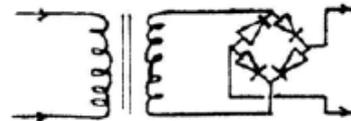


- (a) solution toute électrique avec isolation galvanique réalisée par opto coupleur
- (b) isolation galvanique par relais
- (c) isolation galvanique par transformateur dans le cas de signaux alternatifs.

b.

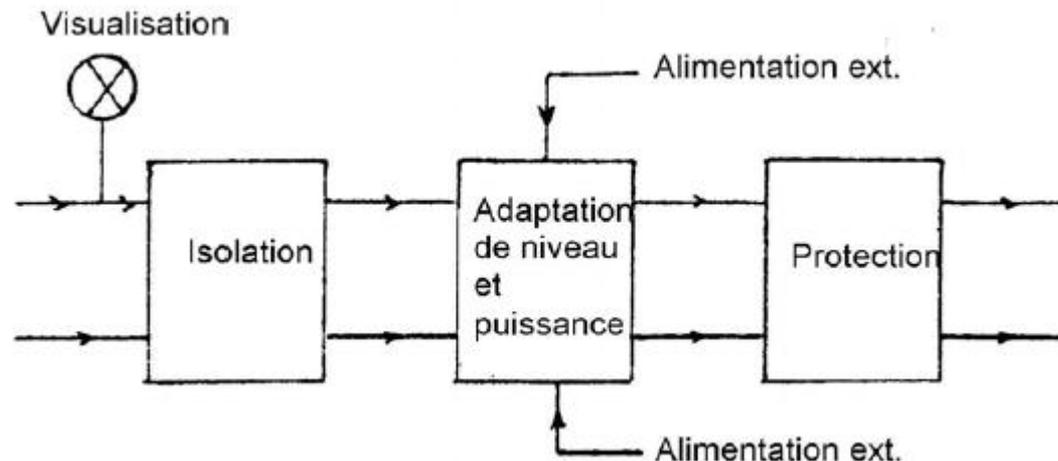


c.



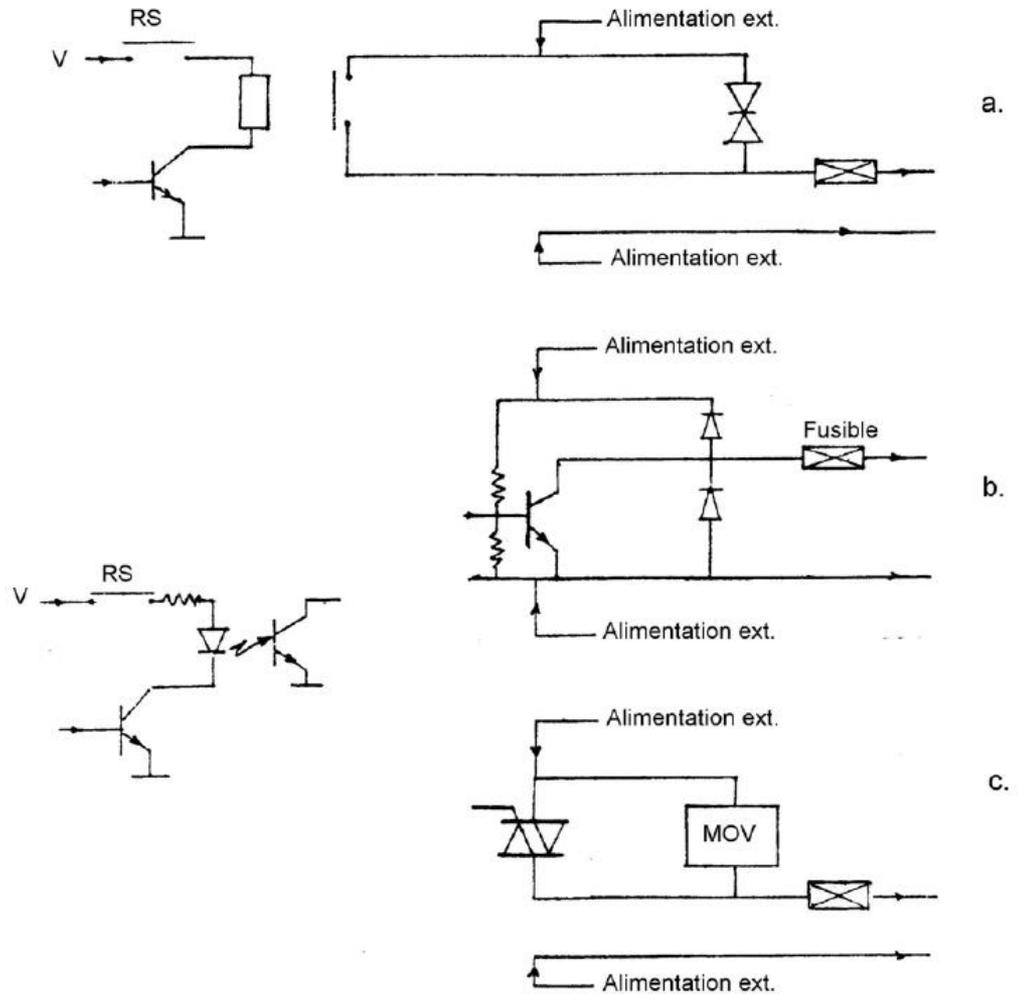
Conditionnement des sorties logiques

- Les principaux éléments d'un conditionneur de sortie logique typique:
 - visualisation d'état
 - isolation galvanique
 - adaptation de niveau et de puissance (en général, à partir d'une alimentation extérieure)
 - protection contre les surcharges (courts-circuits), les surtensions (charges inductives); absorption de l'énergie de coupure des relais, etc.



Conditionnement des sorties logiques

- Quelques exemples de solutions technologiques: **adaptation de puissance**
 - (a) sortie à relais (contact libre de potentiel) assurant à la fois l'isolation et l'adaptation de puissance
 - (b) sortie à transistor avec isolation par opto coupleur
 - (c) sortie à triac pour les signaux alternatifs



Conditionnement des sorties logiques

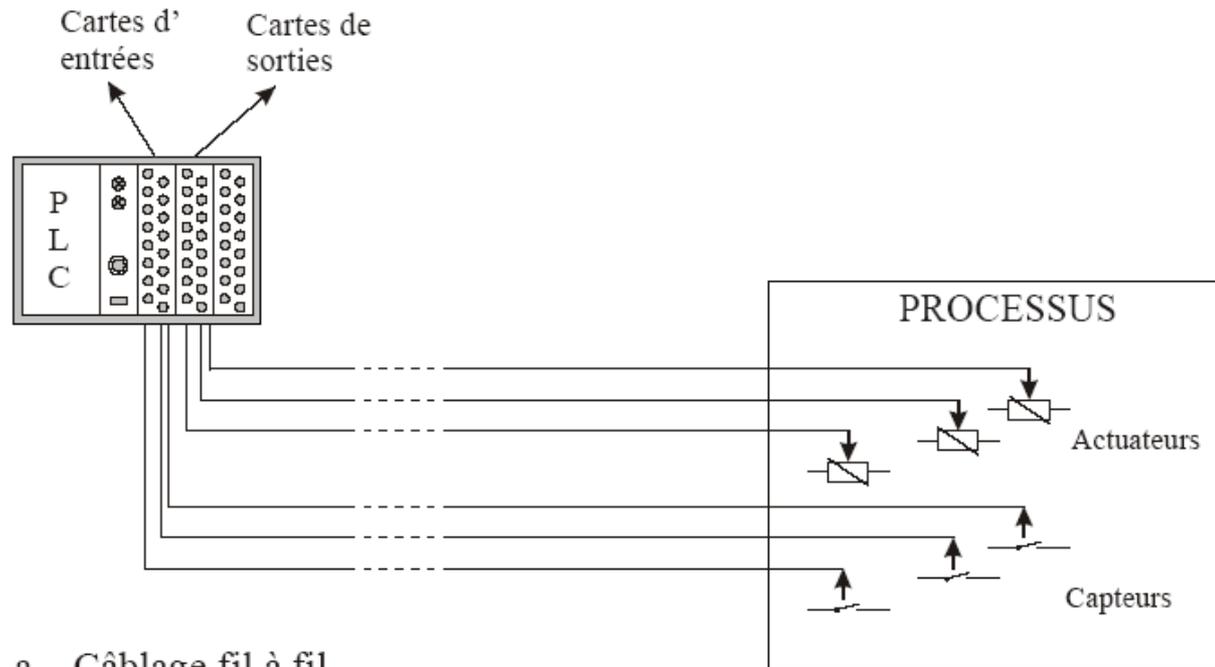
- Pour les sorties à relais: procédure très simple basée sur des bascules RS
 - En cas d'anomalie, le système retombe sur état déterminé (normalement ouvert ou normalement fermé)
 - Lors du retour à la normale, fermeture après réinitialisation cohérente des circuits d'interface
- Pour les opto-coupleurs: piloter l'alimentation des opto-coupleurs de sortie par un relais de sécurité
 - Position de repli := retombée générale des commandes à zéro

PRINCIPES D'ORGANISATION

- L'accès à l'automate d'un signal de mesure venant du processus ou inversement sortant vers le processus met en jeu:
 - Le bornier: découplage entre le câblage venant du processus de l'automate
 - Les circuits de conditionnement des signaux: adaptations, isolations, filtrages, conversions requis pour arriver avec des signaux compatibles avec l'électronique de l'automate.
 - L'interface entre les signaux et le BUS de l'automate.

- Dans la pratique 3 grandes solutions
 - 1° Solutions intégrées
 - 2° Périphéries déportées
 - 3° Réseaux de capteurs et actuateurs

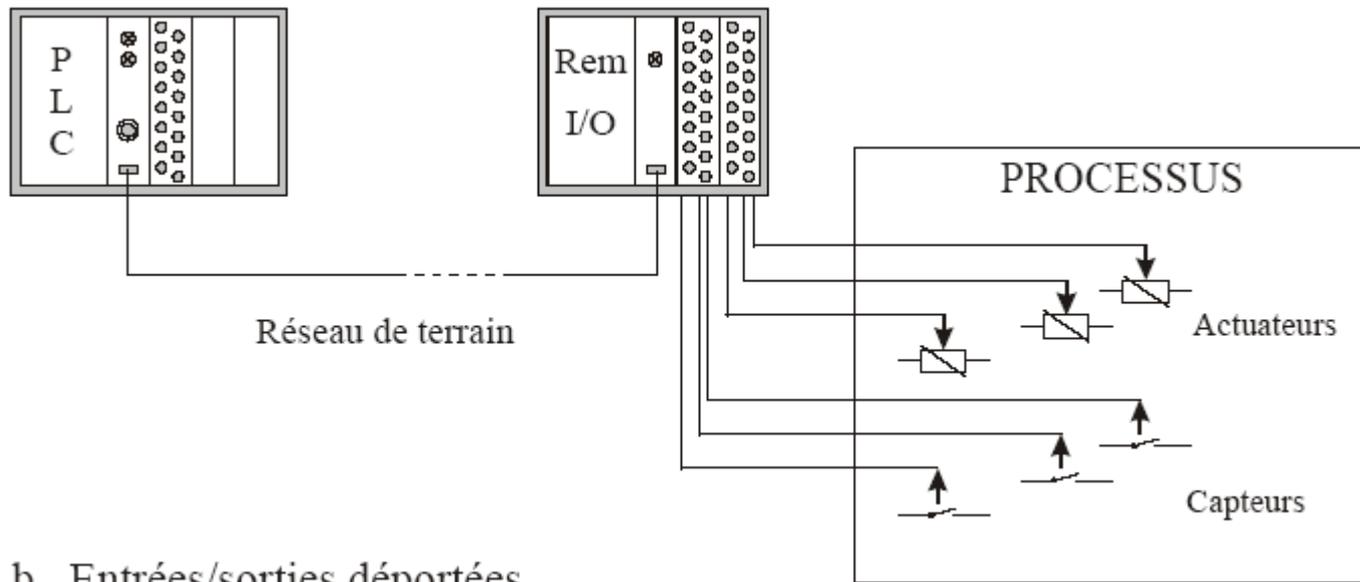
Solution intégrée



a. Câblage fil à fil

- Les trois éléments regroupés sur une même carte et greffés sur le BUS de l'automate
- Petits systèmes
- Inconvénients:
 - Bruits au voisinage de l'automate
 - Câblage point à point

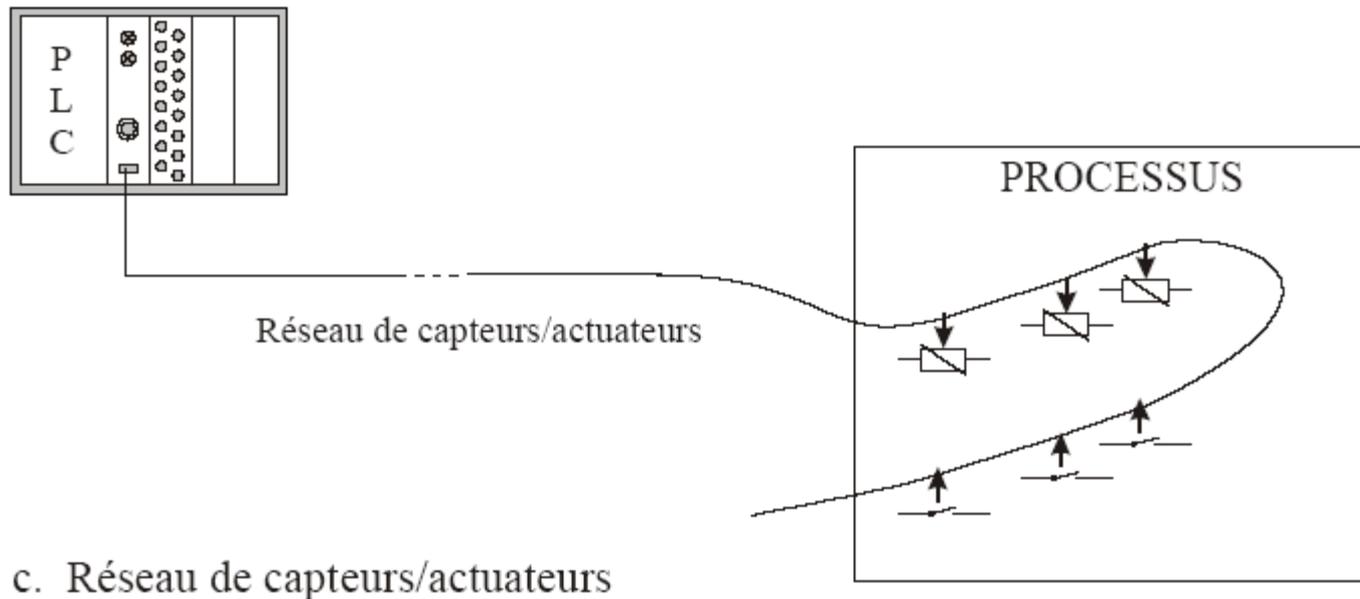
Périphériques déportés



b. Entrées/sorties déportées

- Utilisation du réseau de terrain pour centraliser les entrées sorties dans des modules comportant leur propre BUS et leur propre alim.
- On peut placer ces modules au voisinage du processus ou parties du processus contrôlé
- Réduction du câblage

Réseaux de capteurs et actionneurs

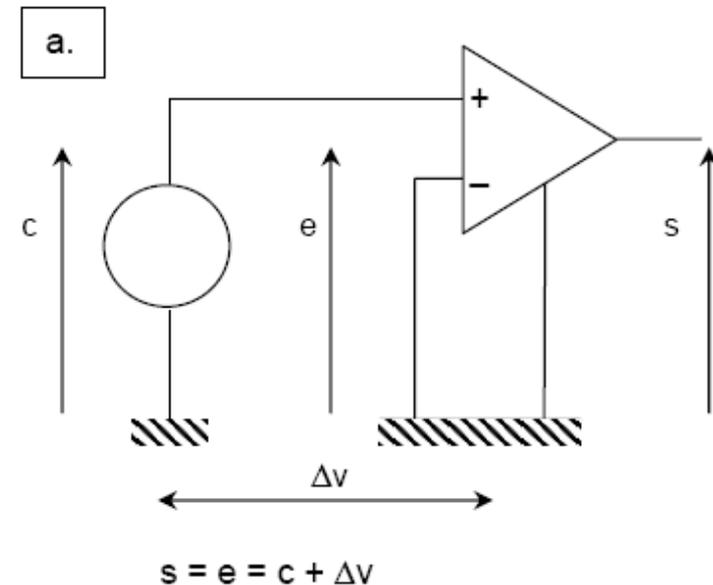


- Les capteurs et actionneurs sont directement greffés sur le réseau qui est relié à l'automate
- Conditionnement local des signaux dans les capteurs / actionneurs
- Réduction maximale du câblage
- Utiliser uniquement des capteurs / actionneurs conçus pour le réseau

Tensions de mode commun

ORIGINE

- Les **tensions de mode** commun trouvent leur origine dans les différences de potentiels qui peuvent exister entre les références de tension des capteurs et les références de tension des systèmes de traitement auxquels ils sont raccordés.
- Si une différence de potentiel existe entre les masses, le signal reçu sera évidemment biaisé.
- Lorsque la tension est nulle aux bornes du capteur, les entrées positive et négative de l'amplificateur vont donc se trouver à une tension Δv par rapport à la référence de cet amplificateur. Cette tension commune aux deux entrées est précisément ce que l'on appelle le mode commun.

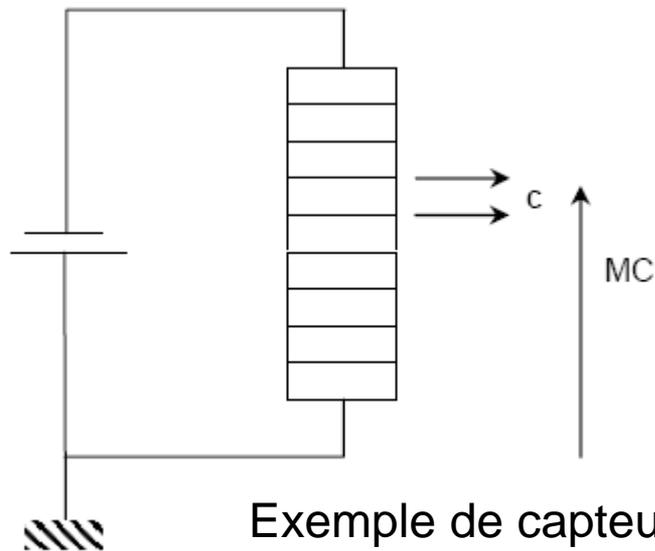


Raccordement unifilaire

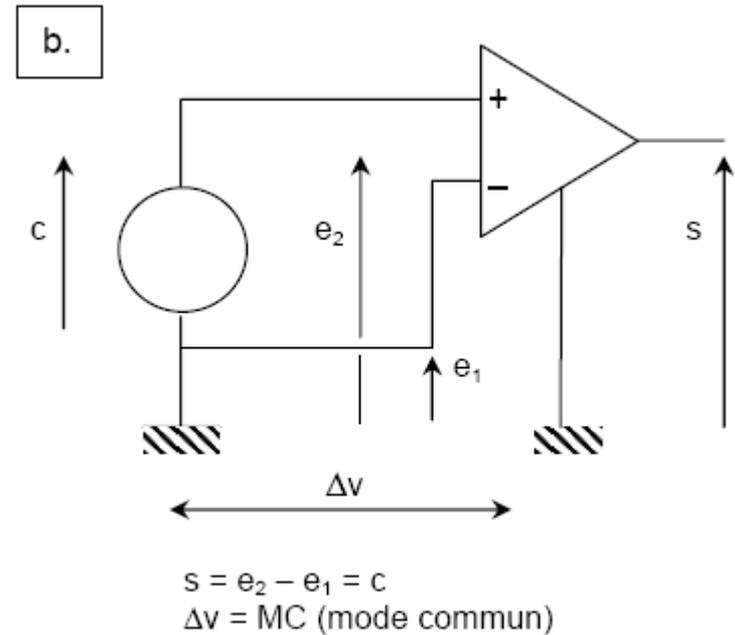
Tensions de mode commun

ORIGINE

- Pour éviter ce risque, on travaille généralement en différentiel.



Exemple de capteur différentiel sur cellule d'électrolyse



Raccordement différentiel

Tensions de mode commun

- Causes des tensions de mode commun
 - Existence de courants de circulation dans le sol qui, en fonction de la distance, peuvent donner lieu à des chutes ohmiques non négligeables, quelques volts typiquement. Ces courants de circulation résultent généralement de l'emploi, volontaire ou involontaire, de la terre comme circuit de retour.
 - Mode commun induit par des parasites électromagnétiques de forte intensité, par exemple, ceux provenant de contacteurs, thyristors, postes de soudure, etc. Les tensions de mode commun induites peuvent atteindre quelques centaines de volts. Dans cette même catégorie, on peut ranger les mises sous tension accidentelles des capteurs ou des conducteurs.
 - Capteurs différentiels : c'est un cas où, par sa disposition même, le capteur se trouve à un certain potentiel par rapport à la terre. Exemple de la mesure de tension aux bornes d'une cellule d'électrolyse.

Tensions de mode commun

PROBLÈMES

- Erreur de mesure :
 - L'impédance des entrées vis-à-vis de la masse de l'amplificateur, si grande soit elle, n'est cependant jamais infinie.
 - Un courant de fuite existera donc toujours et, par suite, une chute ohmique proportionnelle au mode commun.
 - La valeur de cet effet résiduel est une caractéristique importante des amplificateurs, elle s'exprime en terme de taux de réjection du mode commun. Une valeur typique sera, par exemple, 120 dB de 0 à 60 Hz. [Source : Computer Products, inc.]

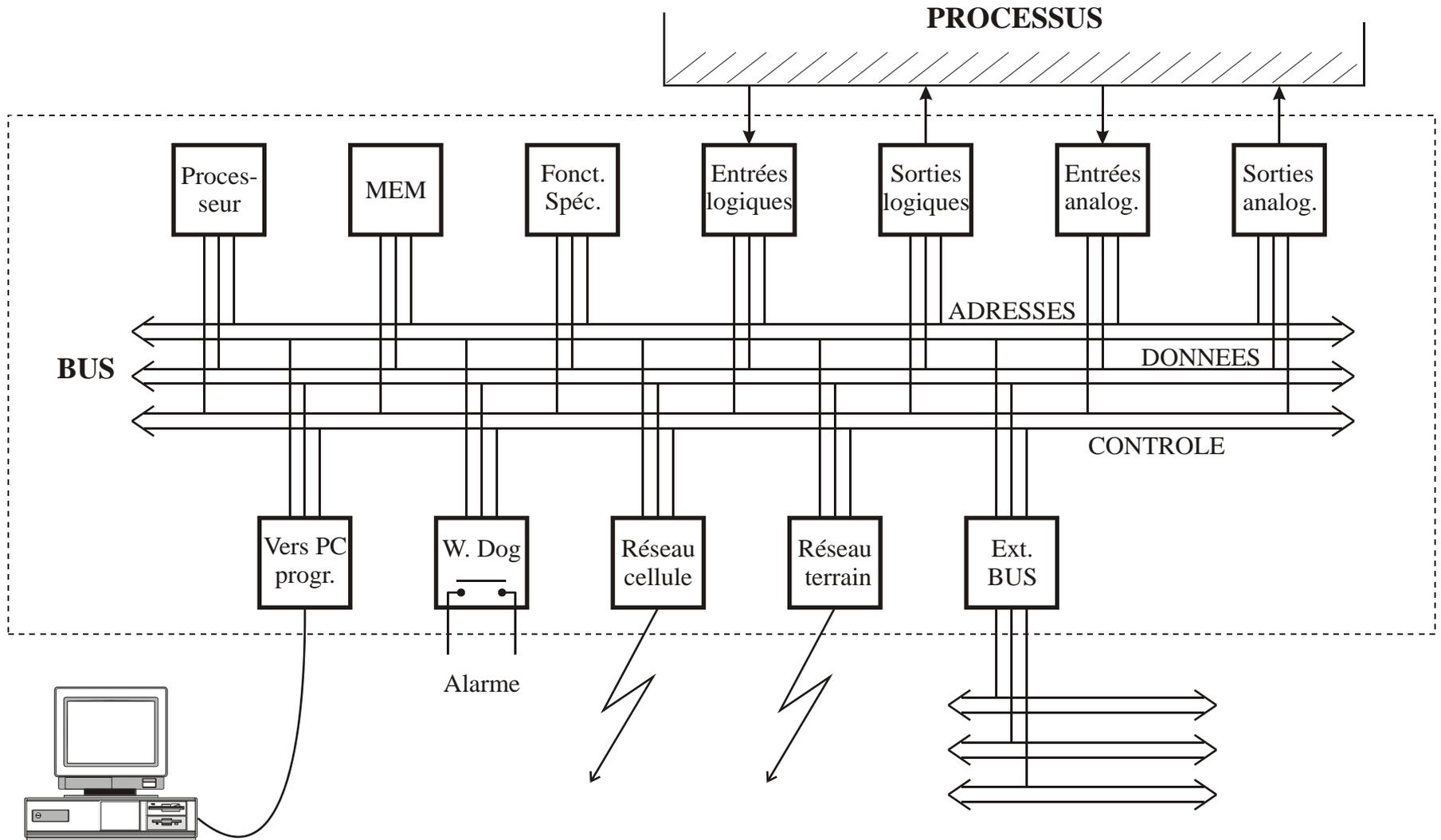
- Claquage de circuits :
 - Pour des raisons purement électroniques, les amplificateurs sont détruits lorsque (signal + mode commun) dépassent des valeurs de l'ordre de 15V

Tensions de mode commun

SOLUTIONS

- Le **travail en différentiel** élimine bien entendu la propagation directe du mode commun à la sortie de l'amplificateur.
- Si pas de risque de claquage: choix d'un **ampli avec taux de réjection adéquat**,
- Si risque de claquage, **isolation galvanique** indispensable entre le capteur et l'amplificateur.
 - Interposition d'un élément non électrique (coupleurs optiques, magnétiques ou relais)

Technologie de réalisation (rappel)



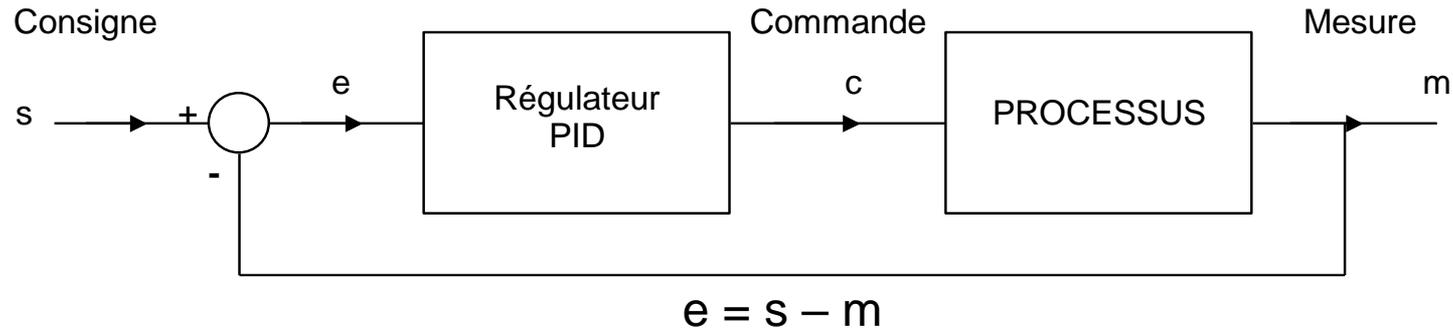
Modules de couplage

- Modules qui permettent de connecter l'automate à d'autres systèmes de traitement et de travailler dans le contexte de signaux informatiques (pas industriels)
- Console de programmation et de test
 - Coupleurs avec la console de programmation
- Extension de bus
 - Etendre le bus de l'automate pour des configurations multi-châssis (longueur limitée)
- Communications
 - Communications à distance par lignes série (téléphoniques, coaxes, fibres optiques...)
- Décentralisation des entrées / sorties
 - Décentraliser des châssis entrées / sorties industrielles sur des distances importantes (ordre du km)

Fonctions spéciales

- En utilisant des microprocesseurs standards pour les automates, ces derniers sont capables de prendre en charge toute espèce de fonctions spéciales:
 - Arithmétiques
 - Régulation PID
 - Commande d'axes
 - Comptage rapide
 - Carte Web
 - ...

Régulateur PID



- Le régulateur PID ouvre la porte à un grand nombre de contrôles de processus continus et répond à la plupart des cas de régulation industrielle
- Signal de commande:

$$c = K \left(e + \frac{1}{T_i} \int e dt + T_d \frac{de}{dt} \right)$$

- Terme (P) proportionnel à l'erreur
- Terme intégral (I) afin d'annuler l'erreur statique
- Terme dérivé (D) proportionnel aux variations de l'erreur qui réalise une anticipation

Régulateur PID

- Algorithme PID

$$c = K \left(e + \frac{1}{T_i} \int e dt + T_d \frac{de}{dt} \right)$$

- Approximation discrète pour les contrôleurs digitaux

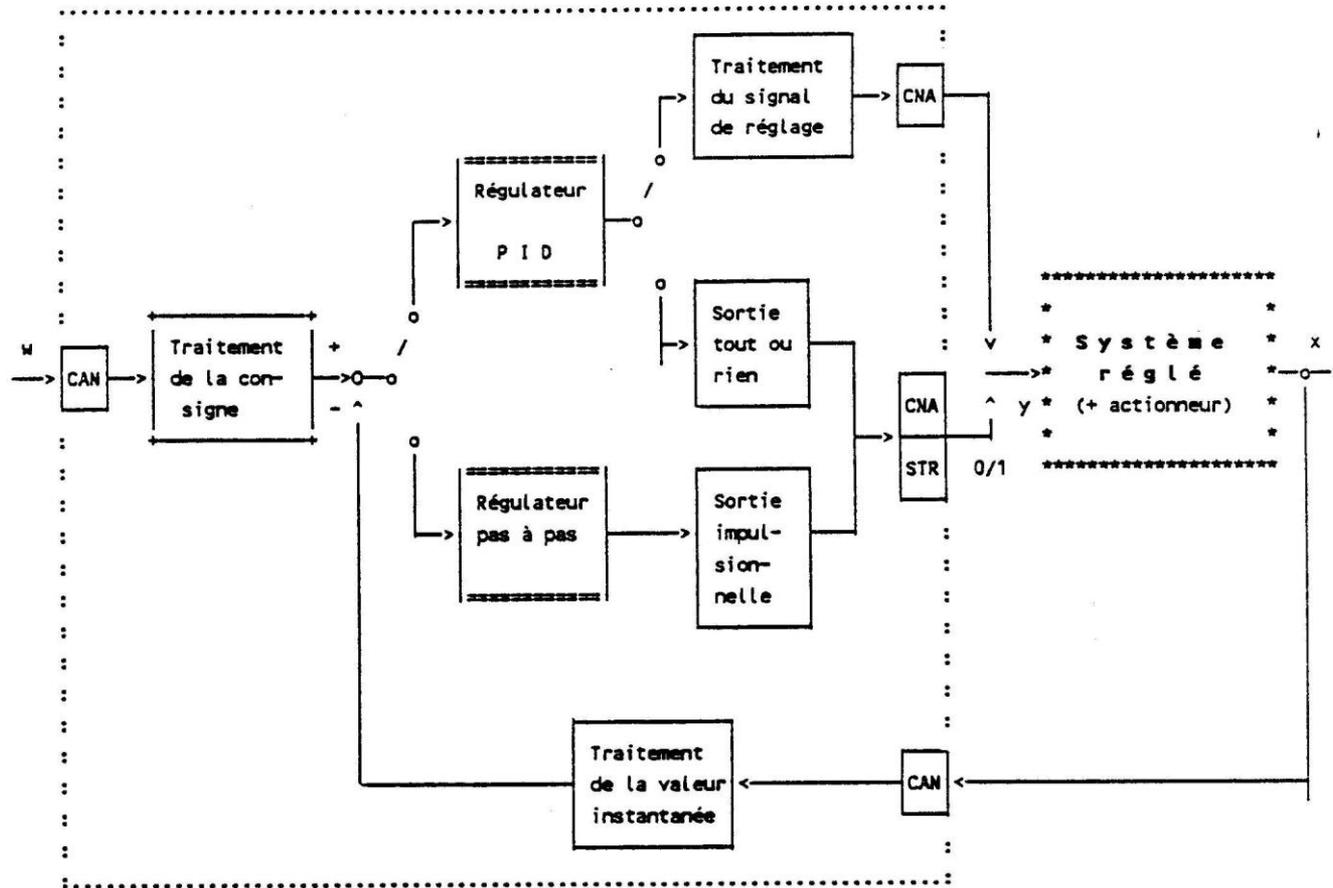
$$c_n = K \left[e_n + \frac{T}{T_i} \sum_{j=0}^n e_j + \frac{T_d}{T} (e_n - e_{n-1}) \right]$$

- Mise en œuvre = choix de K , T_i , T_d

Régulateur PID

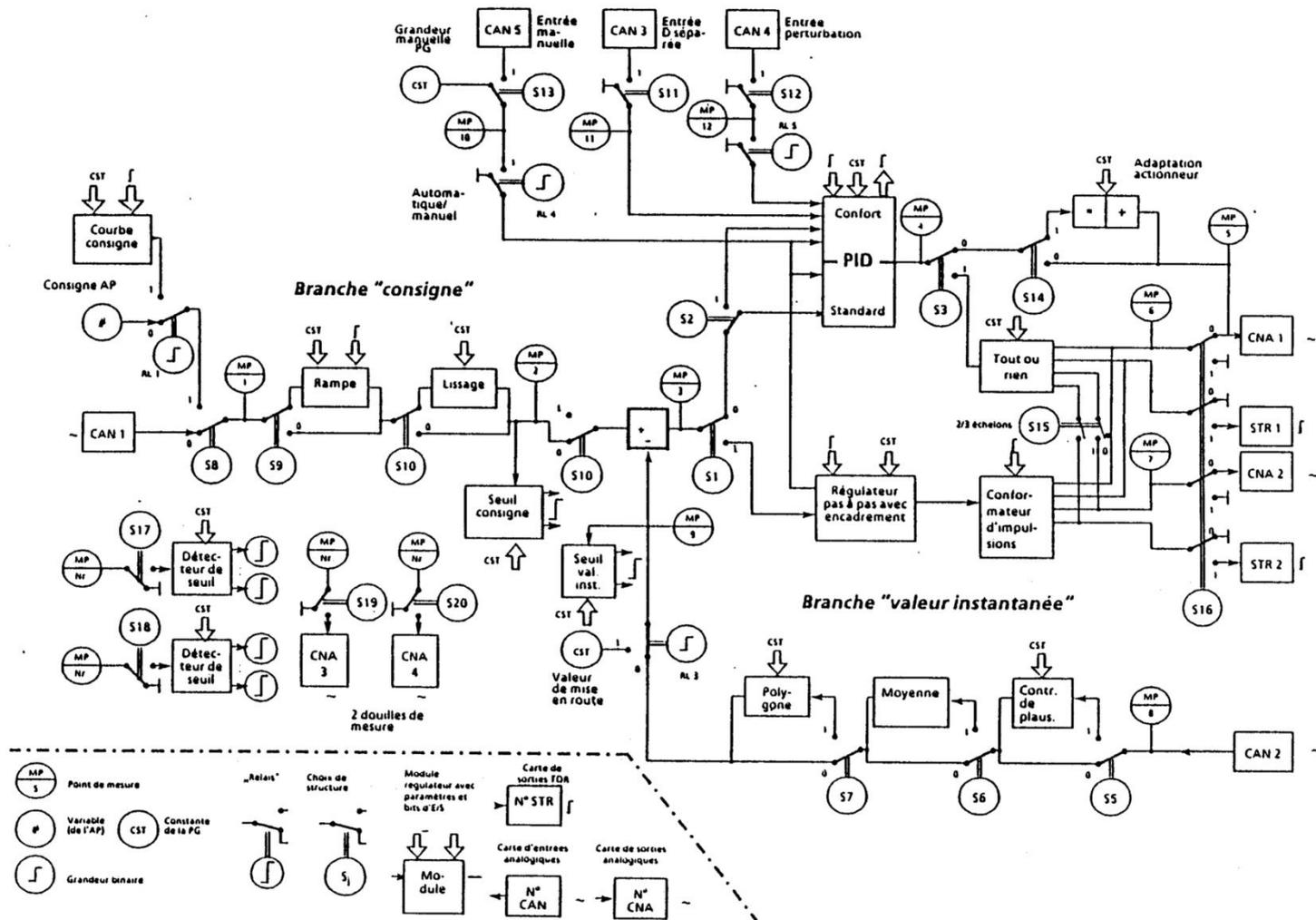
- Programmation en langage automate possible, mais fastidieuse
 - Fonctions annexes (passage manuel / auto, contrôle terme intégral, régulateurs en cascade, contrôle de l'erreur...) difficiles à programmer
- Mise en œuvre informatique
 - Programmation du régulateur en langage informatique
 - Charge exagérée du processeur
 - Solution multiprocesseur
- Carte PID
 - Programmation à l'aide de la console
 - Choix d'options prévues par le constructeur

Régulateur PID



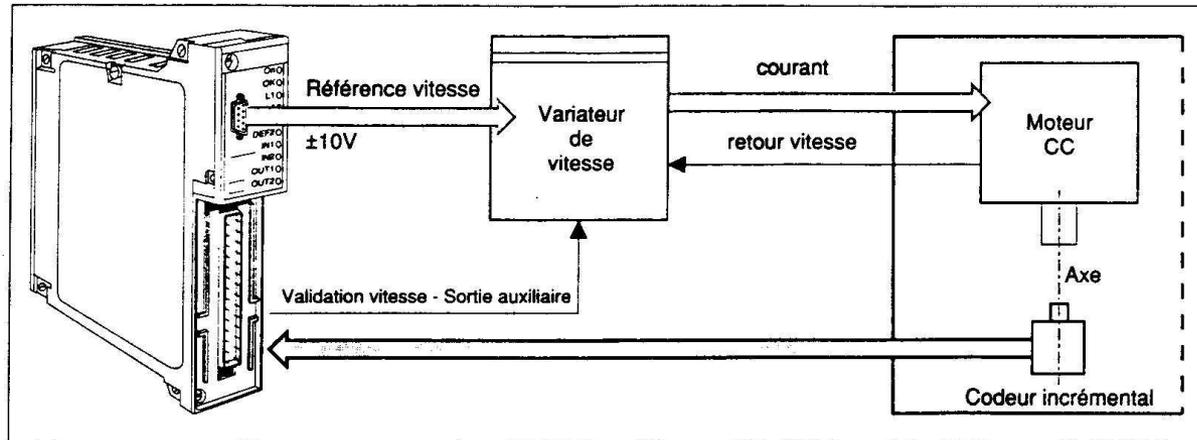
Structure de principe du régulateur PID
(Source SIEMENS)

Régulateur PID



Structure détaillée du régulateur PID
(Source SIEMENS)

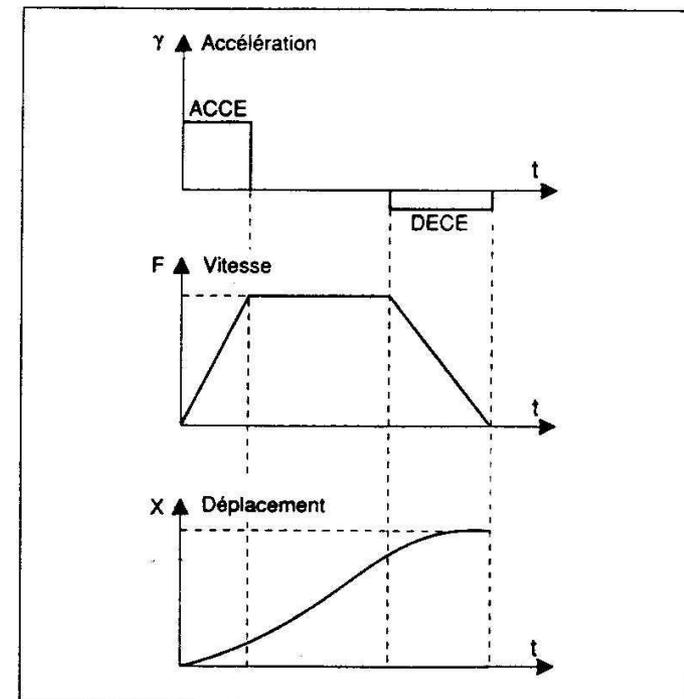
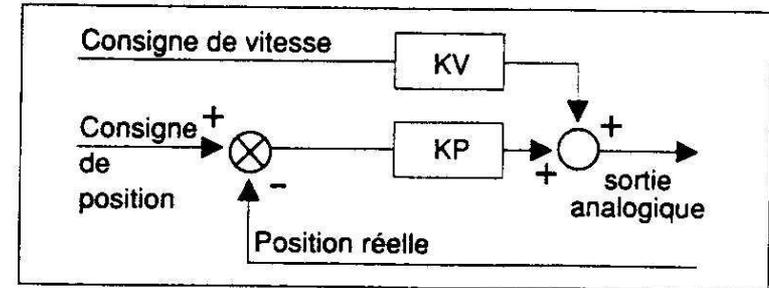
Commande d'axe



- Le position d'un mobile le long d'un axe: machine de découpe, d'emballage, lignes de transfert, de manutention...
- Commande assez facile à standardiser
 - Position mesurée par un codeur incrémental (train d'impulsions rapides): une carte de comptage
 - Contrôle de l'enchaînement des mouvements usuels

Commande d'axe

- Principe de fonctionnement d'une commande d'axe
 - Asservissement en position (gain en position K_p)
 - Asservissement en vitesse assurée par un dispositif extérieur, le variateur (gain en vitesse K_v)
 - Limitations d'accélérations
- Jeu d'instructions simple et complet pour décrire les trajectoires, les enchaîner et assurer la coordination avec le processeur

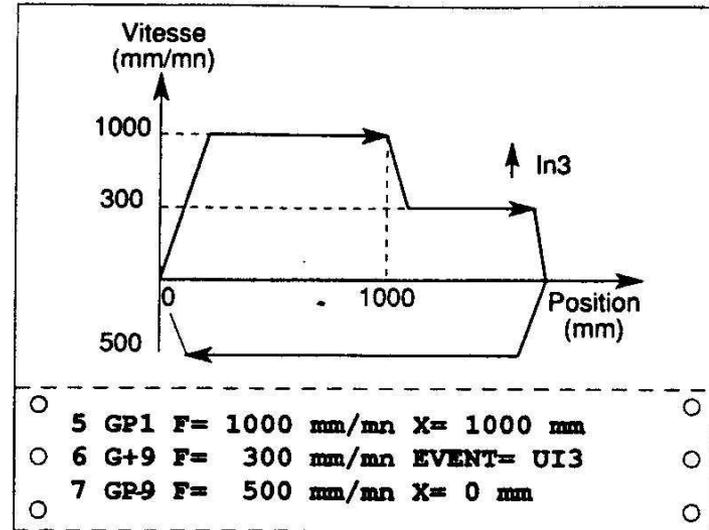


Commande d'axe

■ Programmation

- Instructions de déplacement
 - Jusqu'à une position d'arrêt ou sans arrêt
 - Positions fixées, avec apprentissage, communiquées en temps réels
- Organisation des mouvements
 - Appel à des sous routines
 - Sauts conditionnels ou inconditionnels
 - Attente d'évènement
- Gestion des paramètres
 - Calcul, affectation de valeurs de position

Exemple 1



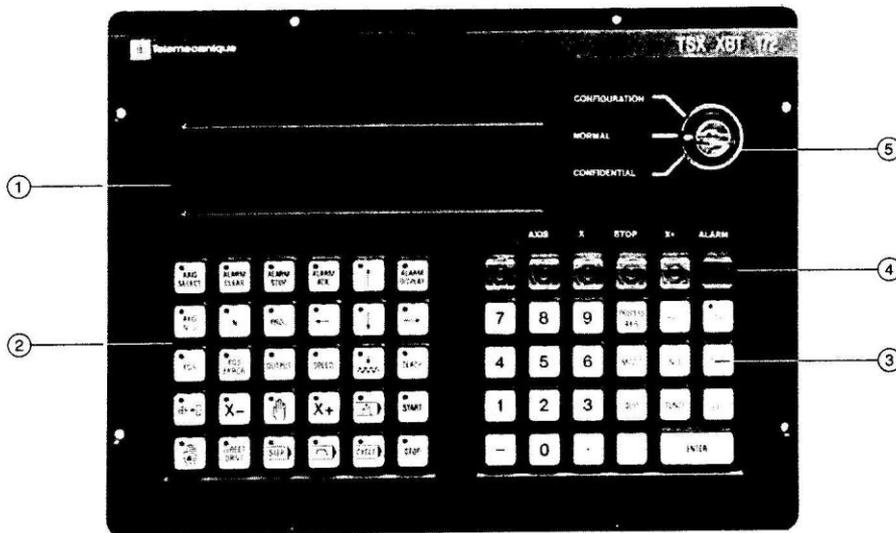
Exemple 2

Exécuter 10 fois la séquence ci-dessus

```
○ 0 LDC CN0 VAL=10 ○  
○ 1 CALL N=5 ○  
○ 2 DEC CN0 ○  
○ 3 JNZ N=1 CN0 ○  
○ 4 END ○
```

Programmation d'une commande d'axe
a. Déplacements
b. Organisation des mouvements

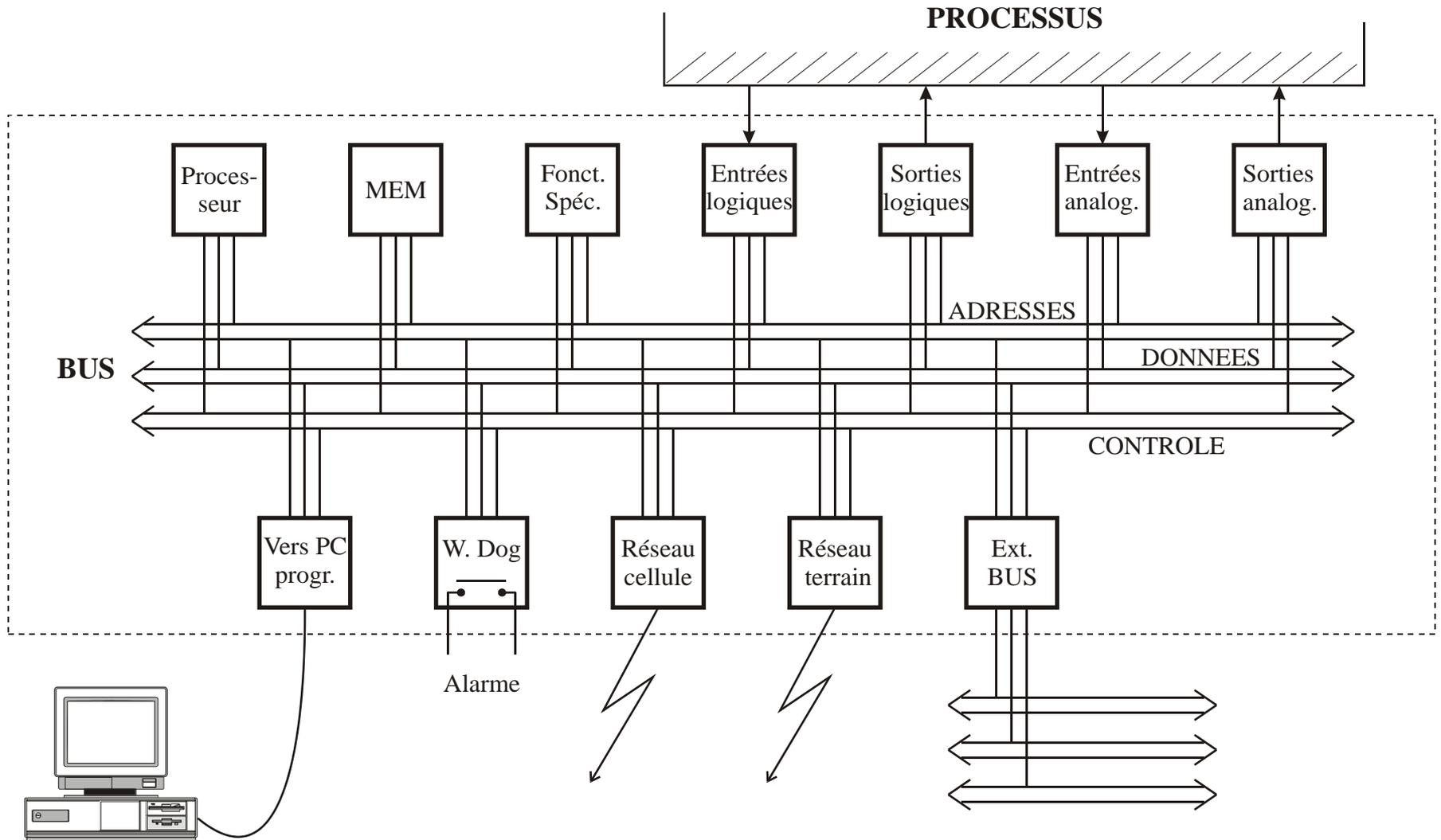
Commande d'axe



Commande d'axe. Exemple de terminal d'exploitation

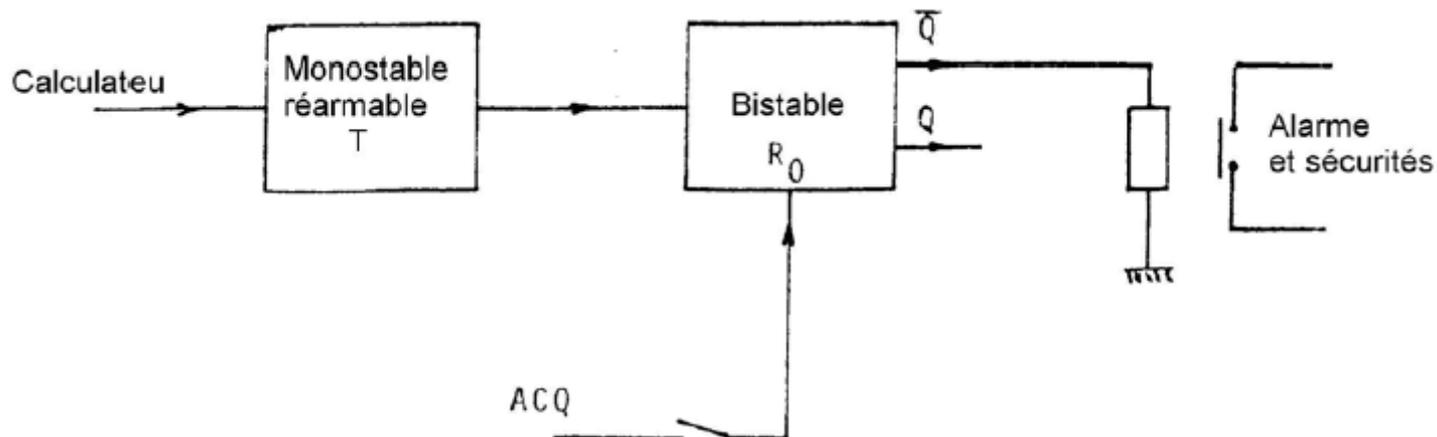
- Terminaux d'exploitation adjoints aux commandes d'axe
 - Objet
 - Sélection des modes de marche
 - Mode de mise au point
 - Modification des paramètres
 - Apprentissage des positions, prise de l'origine
 - Test des programmes
 - Fonctionnement automatique
 - Visualisation des paramètres et du programme
- Observation des déplacements : position et vitesse

Technologie de réalisation (rappel)

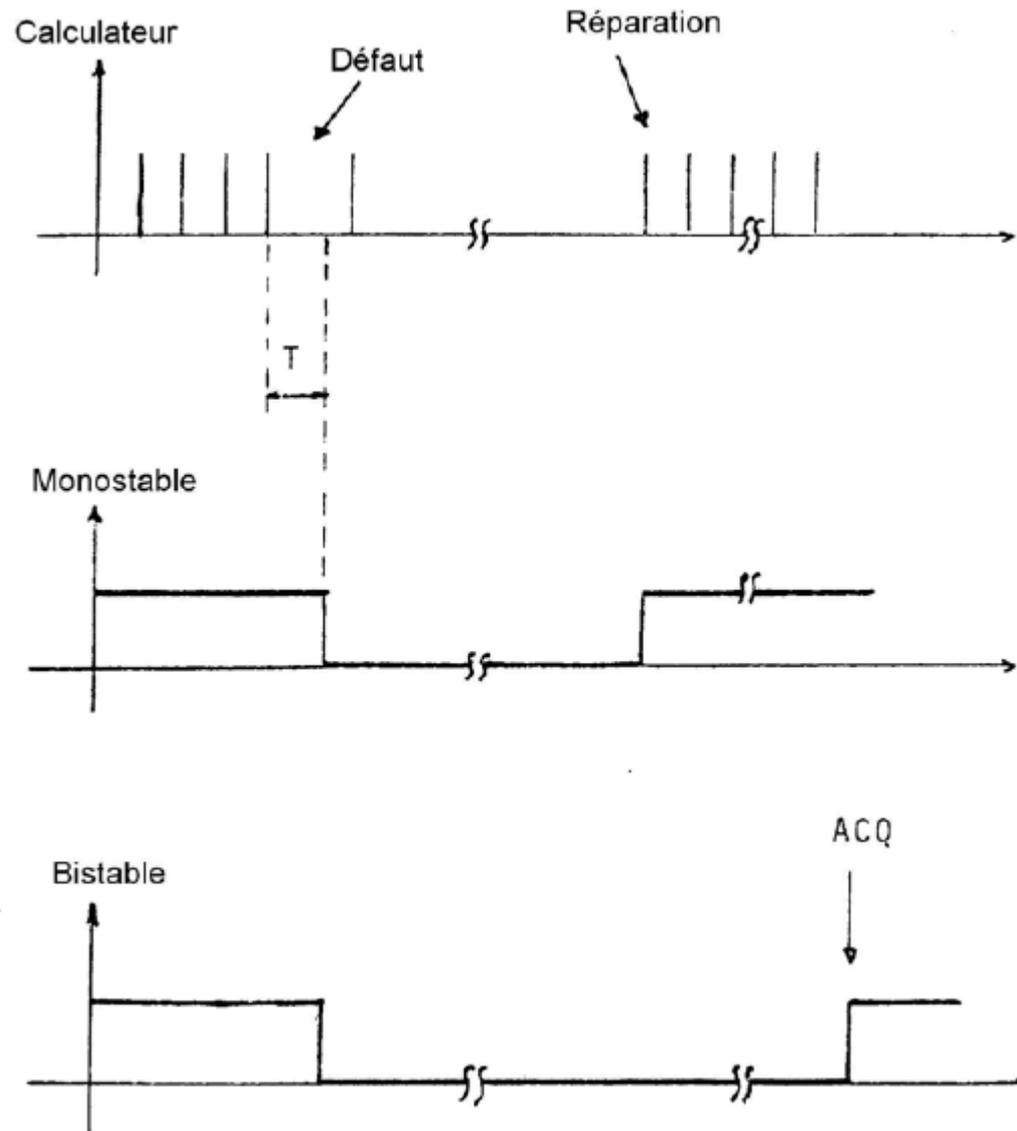


Modules de surveillance et de contrôle

- Modules chargés de surveiller et de contrôler le bon fonctionnement du matériel et du logiciel
- Fonction de « **chien de garde** » (**watch dog**) qui contrôle le cycle de l'automate
 - Principe:
 - Obliger l'automate à envoyer à chaque cycle une impulsion au système de surveillance
 - Vérifier que le temps entre 2 impulsions ne dépasse pas une limite
 - Détecter ainsi une panne de processeur, un bouclage intempestif...



Dispositifs de sécurité

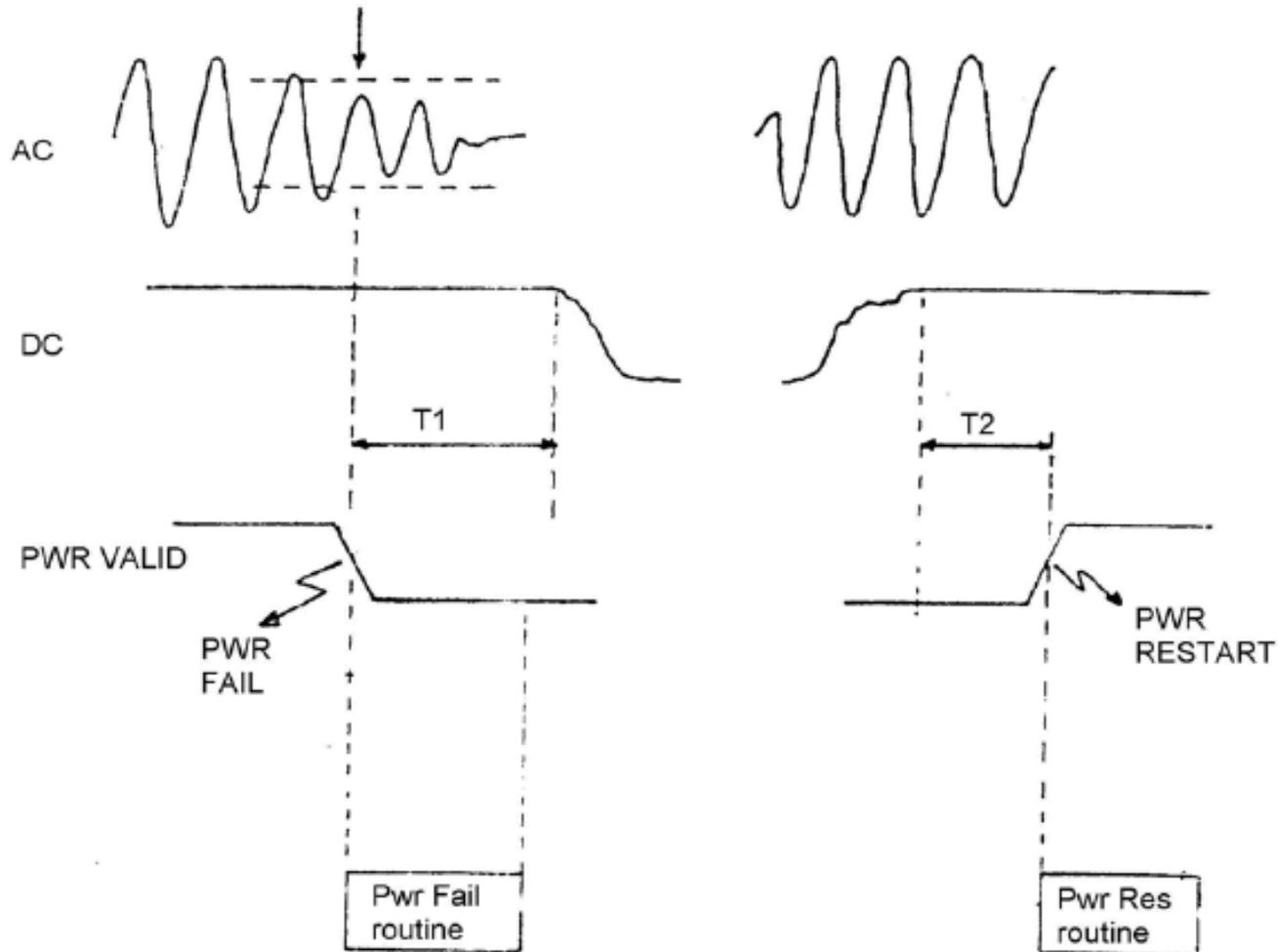


Systeme watch dog

Dispositifs de sécurité

- SURVEILLANCE D'ALIMENTATION
 - Assurer le comportement déterminé du calculateur en cas de coupure d'alimentation ou de microcoupure du réseau et de remise sous tension
- Coupure de tension « **power fail** »
 - Comportement du système informatique aléatoire
 - Tampons d'énergie grâce au filtrage pour prendre en minimum de mesures de sauvegarde
 - Lorsque le système détecte une variation de la tension du réseau (par ex -10%), flag *power valid* est mis à 0
 - Le processeur dispose d'un temps T1 pour faire un certain nombre de sauvetages dans une mémoire non volatile et s'arrêter spontanément

Dispositifs de sécurité



Exemple de séquence de coupure et rétablissement du réseau

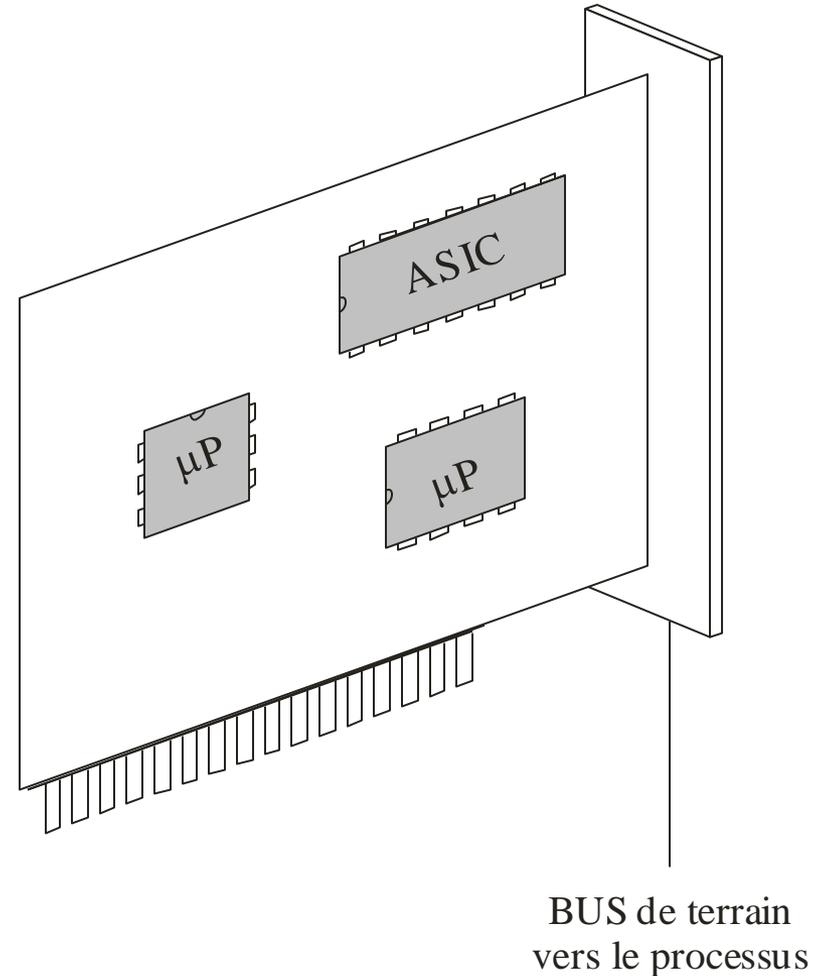
Dispositifs de sécurité

- SURVEILLANCE D'ALIMENTATION
- Remise sous tension « power restart »
 - Système de surveillance détecte le rétablissement des alimentations
 - Après un délais T2 le signal *power valid* est remis à 1
 - La routine d'interruption restaure le système au moment de la coupure et reprend le traitement au point où on l'a abandonné
- Autres scénarios possibles sur la base des deux interruptions engendrées par *power valid*

Soft PLC

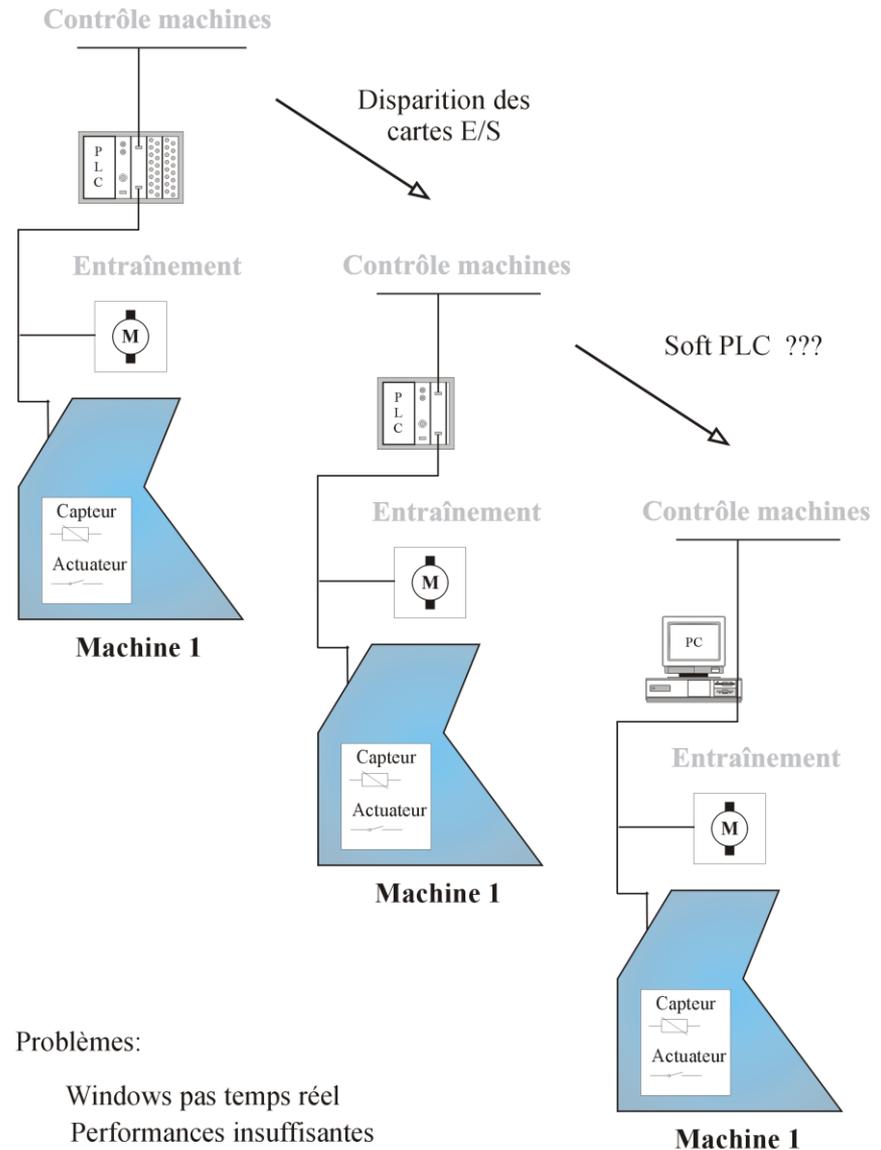
- Développement des PC et apparition des systèmes d'exploitation multitâches (Win NT et suivants)
- Tâche exécutée sous Windows simulant le fonctionnement d'un automate programmable
- Priorité élevée pour obtenir un temps de cycle aussi constant que possible (quelques ms)
- Mise en œuvre des langages classiques

Slot PLC



Soft PLC

- Pour les PC pas de cartes entrées sorties industrielles ⇒ contourner le problème avec des réseaux de terrains
- Point faible: les systèmes de sécurité (coupures de courant, etc.) ne sont pas disponibles
- Avantage: le coût !
- Applications didactiques, applications industrielles non critiques

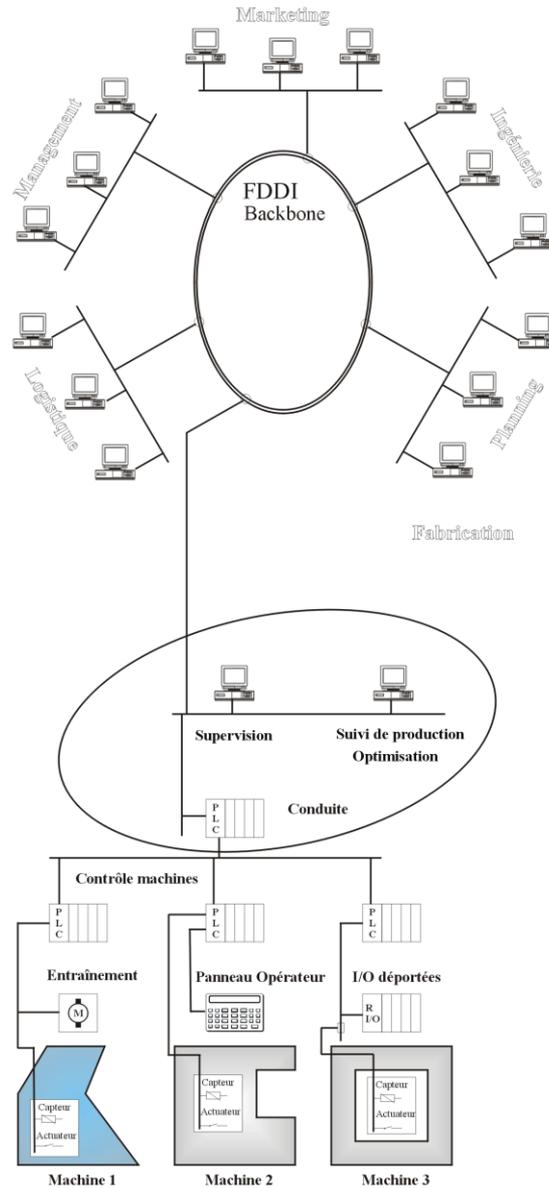


Problèmes:

Windows pas temps réel
Performances insuffisantes
Pas de sécurités (alim., sauvegarde, reprise,...)

En route vers l'ouverture

5
COMPAGNIE



4
USINE

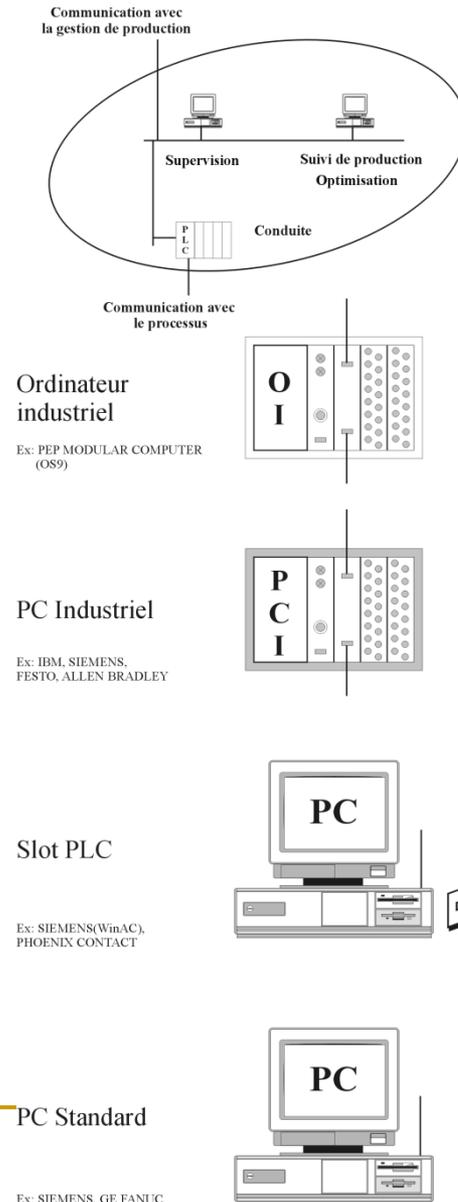
3
ATELIER

2
CELLULE

1
TERRAIN

0
CAPTEUR / ACTUATEUR

En route vers l'ouverture



Performances		Ouverture monde PC	
Temps réel	Sécurité	Matériel	Logiciel
OUI	OUI	NON	NON
(NON)	OUI	NON	(OUI)
OUI	OUI	OUI	OUI
(NON)	NON	OUI	OUI