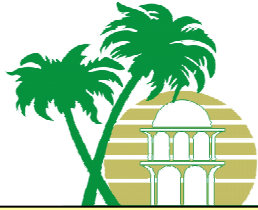


République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Centre Universitaire D'EL-Oued
Institut des Sciences Et Technologie
Département de Génie Electrique Et Hydraulique



Centre Universitaire D'EL-Oued



Centre Universitaire D'EL-Oued

Mémoire de fin d'étude pour l'obtention du diplôme d'ingénieur d'état
En : Electrotechnique
Option :
Réseaux Electriques
Thème



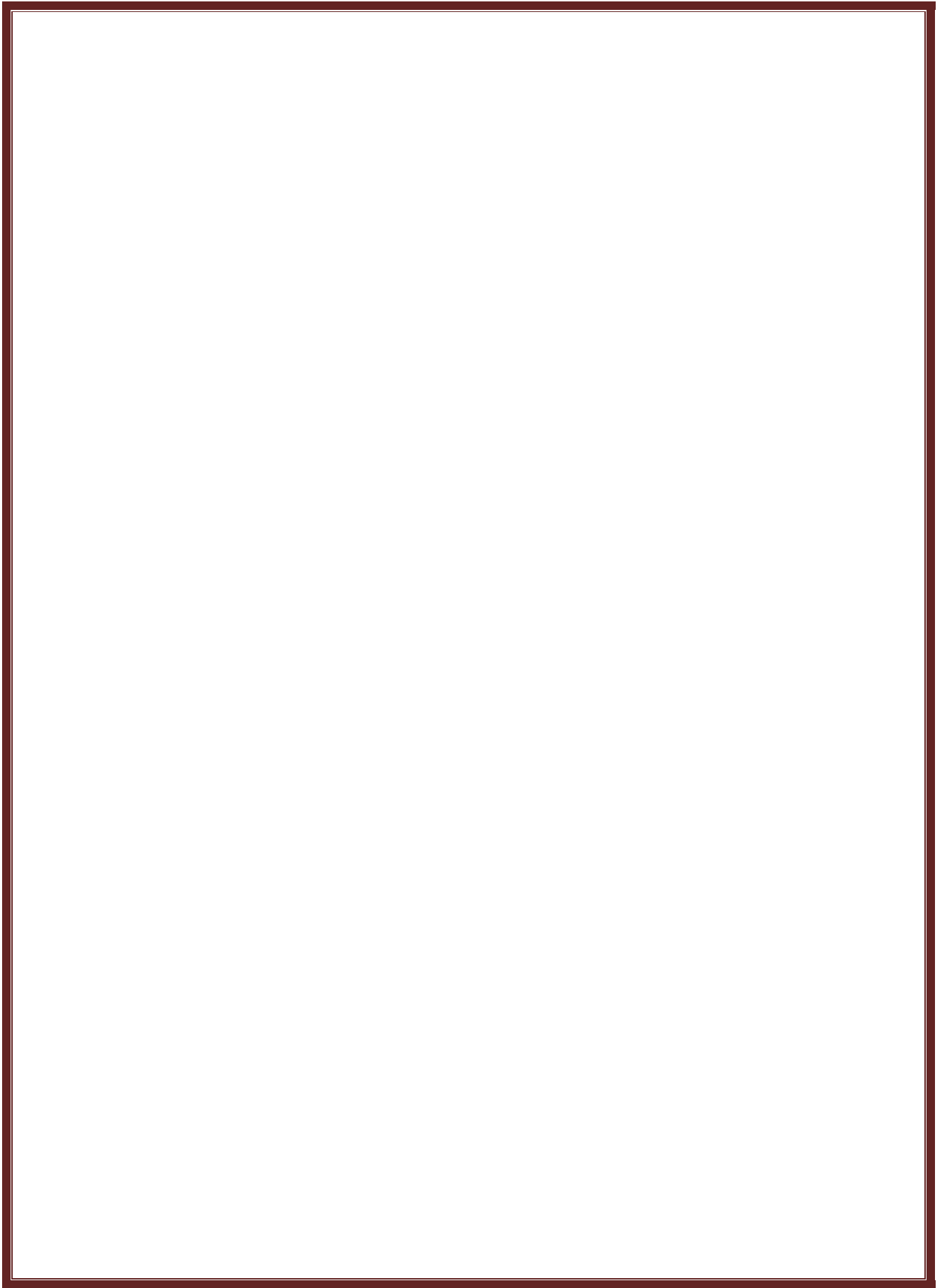
Présenté par :

Boukhcheche Sofiane
Ben Ammar M.Elbachir
Chelligu Salah

Encadrée par :

Mr. Ruiha Redha

Promotion : 2008



INTRODUCTIOOC GENERAL

CHAPITRE I : LES SYSTEMES AUTOMATISES

I-1-Fonction globale d'un système	1
I- 1-1.Matières d'œuvre.....	1
I- 1-2.Valeur ajoutée.....	1
I-1-3.Contexte et valeur ajoutée.....	2
I-2-système de production.....	2
I-3-Commande automatique.....	2
I-3-1.Objectifs de l'automatisation.....	3
I-3-2.Surveillance d'un système automatisé.....	3
I-4- Structure d'un système automatisé.....	3
I-4-1.La partie commande.....	4
I-4-2.La partie Opératives.....	4
I-4-3.Les interfaces.....	4
I-5-Etude des éléments d'un système automatisé.....	5
I-5-1.Etude des éléments de La Partie Opérative.....	5
- I- Les actionneurs.....	5
a- les actionneurs statiques.....	6
b- les actionneurs dynamiques.....	6
II - Les effecteurs.....	7
I-5-2.Partie commande.....	7
I – Partie traitement.....	8
II- Pupitre.....	8
I-5-3.Etude des éléments d'interfaces.....	8
I- Les pré actionneurs.....	8
II - Les Capteurs.....	9

CHAPITRE 2 : LES AUTOMATES PROGRAMMABLES

II-1-Introduction.....	11
II-2- Principe de fonctionnement.....	12
II-2.1. La lecture.....	13
II-2.2.Le traitement.....	13
II- 2.3.La commande.....	13
II-3- Architecture des automates.....	13

II-3.1. Aspect extérieur.....	13
II-3.2 Structure interne.....	15
a-Module d'alimentation.....	15
b-Unité centrale.....	15
c- Le bus interne.....	15
d-Mémoires.....	15
e-Interfaces d'entrées / sorties.....	16
-Interface d'entrée.....	16
-Interface de sortie.....	16
II-3.3Alimentation et câblage des entrées/sorties du TSX27.....	16
II-4-Organisation fonctionnelle de l'API.....	20
II-4.1. Unité de traitement un processeur.....	21
-L'unité logique (UL).....	21
-L'accumulateur.....	21
-le décodeur d'instruction.....	21
-Le compteur ordinal ou compteur programme.....	21
II-4.2. Éléments de stockage et de liaison.....	21
II-4.3. Modules d'entrées/sorties (E/S).....	22
a-Interfaces d'entrées.....	23
b- Interfaces de sorties.....	23
II-4.4. Auxiliaires.....	24
II-5-Communication d'API.....	25
II-5.1. Besoins de communication.....	25
II-5.2. Outils de communication.....	25
II-5.3. Outils de programmation.....	26
-Consoles de réglage.....	26
-Consoles spécifiques.....	26
-PC console.....	26
II-6-Choix d'un API.....	27
-le nombre d'E/S.....	27
-la taille du programme.....	27
-la vitesse de traitement.....	27

-les fonctions complémentaires.....	27
-les langages de programmation.....	27
-le nombre de voies analogiques.....	27
-le nombre et le type de liaisons inter processeurs.....	28
-la capacité de traitement arithmétique.....	28
-la bibliothèque fonctionnelle.....	28
-les modes de communication.....	28
II-6- Quelques types des automates programmables industrielles.....	28
II-6.1. Groupe Siemens.....	28
• Simatic S7-300.....	28
• simatic S7-400.....	28
• simatic S7-200.....	28
II-6.2. Groupe Schneider.....	29
-TSX NANO.....	29
-TSX PREMIUM.....	29
-TSX MICRO.....	30
II-6.3. Les automates TSX27 (télémechanique).....	30
-Fiabilité.....	31
-Auto- tests intégrés.....	31
✓ Le processeur.....	31
✓ La cartouche mémoire.....	31
✓ Les entrées/sorties.....	31
-Aide au diagnostic	32
➤ configuration TSX27.....	32
1. Configuration 20 E/S.....	32
2. Configuration 40 E/S.....	32
3. Configuration 60 E/S	32
3. Configuration 80 E/S.....	33

CHAPITRE III : LE LANGAGE GRAPHIQUE PL7-2

III-1-Introduction.....	34
III-2- Les objets adressables du PL7-2.....	35
III-2.1. Adressages des entrées/sorties du TSX27.....	35
III-2.2. Objets mots.....	36

III-2.3. Objets blocs fonctions.....	36
❖ Blocs fonctions d'automatisme.....	36
III-3. Cartouches langage et cartouches utilisateur.....	38
a)-Cartouches langage.....	38
b)- Cartouches utilisateur.....	38
III-4. Les bits système :TSX 27.....	39
III-5-Présentation des langages PL7-2.....	41
III-5.1 Langage a contacts.....	41
a) Principe.....	41
b) Le traitement postérieur.....	43
III-5.2 Langage a grafcet.....	44
a) principes du grafcet.....	44
-graphiques simples.....	45
- Les objets spécifiques au grafcet.....	47
- Utilisation des renvois.....	48
III-6. Le terminal de programmation TSX T407.....	48
III-6.1. Ecran de visualisation.....	49
-Caractéristiques.....	49
-Informations visualisées.....	50
III-6.2. Clavier.....	51
- Caractéristiques.....	51
- Organisation de clavier.....	51
III-6.3 Mise sous tension	52
- Accès au menu sélection des modes.....	52
III-7. Création de programme.....	54
III-1. Langage contacts.....	55
-Saisie d'un réseau de contacts.....	55
-Saisie d'une bobine de sortie	56
-Saisie d'un bloc fonction/opération.....	56
*Bloc fonction temporisateur.....	57
*Bloc fonction compteur.....	58
III-2. langage Grafcet.....	59
III-2.1. Saisie d'un graphe.....	59

-Validation d'un graphe.....	61
-Validation d'un graphe saisi sur plusieurs pages.....	61
-Saisie d'une action.....	61
-Saisie d'une réceptivité.....	61
-Suppression d'un élément et de son contenu.....	62
-Suppression d'un élément d'aiguillage ou de fin d'aiguillage.....	62
-Suppression de la totalité d'une liaison (aiguillage ou activation simultanée d'étapes).....	62
-Suppression de la totalité d'une liaison (aiguillage ou activation simultanée d'étapes).....	62

CHAPITRE IV : SURVEILLANCE DES PROCÉDES DE PRODUCTION D'EAU POTABLE PAR API

IV-1-Introduction.....	64
IV-2-Description de l'unité de filtrage.....	65
IV-2. 1. Présentation de l'UCM.....	65
-Schéma de principe.....	65
IV-2.2. Les techniques membranaires utilisées.....	66
IV-2.3. Utilité de l'UCM.....	67
IV-2.4. Exemple d'application pratique de L'UCM.....	67
• Production d'une eau potable	67
IV-2.5-La chaîne de production étudiée.....	69
IV-3-Equipements électroniques.....	71
IV-3.1. Démarreur progressif.....	71
• Fonctionnement.....	72
Réglages et mise en service.....	72
IV-3.2. Bloc de contrôle multifonction IGBT.....	72
IV-3-3. Les pompes primaires.....	77
IV-3.4 Pompe Osmose Inverse.....	77
IV-3.5 Electrovanne.....	77
IV-4-Circuits électriques.....	78
- Circuit de puissance.....	78
- circuit de commande et de puissance.....	79
-Circuit de commande et de puissance.....	80

IV-5- Programmation.....	81
IV-5-1.LONGAGE CONTACT (ladder).....	81
IV-5-2. LANGAGE GRAFCET.....	87
-Tableau d'adressage de l'entrées/sorties.....	93

Introduction générale :

L'automatique est à la fois une science et une technique qui étudie les méthodes scientifiques et les moyens techniques pour la conception et la réalisation des systèmes automatisés. L'automatisation est l'exécution automatique de tâches domestiques, industrielles, administratives ou scientifiques sans intervention humaine. L'homme a, depuis toujours, cherché à économiser son énergie musculaire en exploitant d'autres formes d'énergie animale, énergie hydraulique, énergie éolienne. L'invention du moteur électrique et du moteur thermique a renforcé cette évolution. L'homme est cependant indispensable pour le pilotage des opérations. Il assure les tâches informationnelles, bien qu'il soit soulagé des tâches opérationnelles.

Aujourd'hui, l'automatisation permet de remplacer l'homme aussi bien dans les tâches opérationnelles, que dans les tâches informationnelles. Les progrès réalisés sont dus aux développements de l'informatique. Cependant, certains systèmes ne nécessitent aucun moyen de calcul.

Des systèmes automatisés permettent :

- De réaliser des opérations trop complexes ou dangereuses et ne pouvant être confiées à l'homme.
- De substituer l'homme dans les opérations répétitives ou pénibles.
- D'accroître la précision
- D'accroître la rentabilité et la productivité par la diminution de la main-d'œuvre.

Dans le domaine des automatisés industriels, on rencontre des systèmes dont le but est de contrôler des procédés physiques. Ce qui peut se manifester par un nombre élevé de capteurs dont il faut tenir compte, et d'actionneurs auxquels fournir des commandes. Les architectures sur lesquelles les contrôleurs s'exécutent (matériel et système d'exploitation de la plateforme d'exécution) peuvent être massivement parallèles, hétérogènes et connectées par des canaux de communications.

La conception de système automatique requiert alors le support d'outils pour la construction des modèles, leur validation et analyse, et la génération de mises en œuvre qui soient correctes vis-à-vis de la spécification. Les systèmes automatisés sont souvent réalisés au moyen d'automates programmables, en utilisant des standards de conception (selon des normes).

I -1-FONCTION GLOBALE D'UN SYSTEME :

La fonction globale de tout système automatisé est de conférer une valeur ajoutée à un ensemble de matières d'œuvre dans un environnement ou contexte donné.

I -1-1.Matières d'œuvre :

Une matière d'œuvre peut se présenter sous plusieurs formes. Par exemple :

- un produit, c'est-à-dire de la matière, à l'état solide, liquide ou gazeux, et sous une forme plus ou moins transformée :
 - des objets techniques.
 - des produits chimiques.
 - des produits textiles.
 - des produits électroniques.
 - etc.
 - qu'il faut : concevoir, produire, stocker, transporter, emballer, utiliser...
- de l'énergie :
 - sous forme : électrique, thermique, hydraulique...
 - qu'il faut : produire, stocker, transporter, convertir, utiliser...
- de l'information :
 - sous forme écrite, physique, audiovisuelle...
 - qu'il faut : produire. stocker, transmettre, communiquer, décoder, utiliser...

I -1-2.Valeur ajoutée :

La Valeur Ajoutée à ces matières d'œuvre est l'objectif global pour lequel a été défini conçu, réalisé, puis éventuellement modifié, le système. Cette Valeur Ajoutée peut résulter par exemple :

- d'une modification physique des matières d'œuvre
 - traitement mécanique : usinage, formage, broyage, impression...
 - traitement chimique ou biologique
 - conversion d'énergie
 - traitement thermique : cuisson, congélation...
 - traitement superficiel : peinture, teinture...
- d'un arrangement particulier, sans modification des matières d'œuvre
 - montage, emballage, assemblage...
 - couture, collage...

- d'une mise en position particulière, ou d'un transfert, de ces matières d'œuvre
 - manutention, transport, stockage
 - commerce
 - communication
- prélèvement d'information sur ces matières d'œuvre
 - contrôle mesure lecture examens...

I -1-3.Contexte et valeur ajoutée :

La nature, la quantité et la qualité de la valeur ajoutée peuvent varier pour tenir compte de l'évolution des besoins de la société dans laquelle s'insère le système. Ce qui peut conduire à modifier le système, voire l'abandonner pour en construire un nouveau.

L'environnement, c'est-à-dire le contexte physique, social, économique, politique, ... joue un rôle essentiel dans le fonctionnement du système et influe sur la qualité et/ou la quantité de la Valeur Ajoutée.

I -2. SYSTEME DE PRODUCTION :

Un système de production est un système à caractère industriel possédant les caractéristiques suivantes :

- l'obtention de la valeur ajoutée présente, pour un ensemble de matières d'œuvre donné, un caractère reproductible,
- la valeur ajoutée peut être exprimée et quantifiée en termes économiques Un système de production répond au besoin d'élaborer des produits, de l'énergie ou de l'information à un coût rentable pour l'utilisateur du système.

L'élaboration progressive de la valeur ajoutée sur les matières d'œuvre est obtenue :

- au moyen d'un ensemble d'éléments ou de dispositifs opératifs, appelés partie opérative et plus ou moins mécanisés,
- par l'action, à certains moments, d'opérateurs humains et/ou de dispositifs de commande pour assurer la coordination des dispositifs opératifs.

Exemples de système de production

- usine de fabrication chimique, métallurgique, électronique...
- société de service (informatique...), groupe de presse, banque...

I -3.COMMANDE AUTOMATIQUE :

L'automatisation de la production consiste à transférer tout ou partie des tâches de

coordination, auparavant exécutées par des opérateurs humains, dans un ensemble d'objets techniques appelé partie commande.

La Partie Commande mémorise le savoir faire des opérateurs pour obtenir la suite des actions à effectuer sur les matières d'œuvre afin d'élaborer la valeur ajoutée.

Elle exploite un ensemble d'informations prélevées sur la Partie Opérative pour élaborer la succession des ordres nécessaires pour obtenir les actions souhaitées.

I -3-1.Objectifs de 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 du système c'est-à-dire augmenter la quantité de produits élaborés pendant une durée donnée. Cet accroissement de productivité exprime un gain de valeur ajoutée sous forme :

. D'une meilleure rentabilité.

. D'une meilleure compétitivité.

- améliorer la flexibilité de production.

- améliorer la qualité du produit grâce à une meilleure répétabilité de la valeur ajoutée.

- s'adapter à des contextes particuliers :

. Adaptation à des environnements hostiles pour l'homme (milieu salin, spatial, nucléaire...).

. Adaptation à des tâches physiques ou intellectuelles pénibles pour l'homme (manipulation de lourdes charges, tâches répétitives parallélisées...).

-augmenter la sécurité,..... etc.

D'autres objectifs, à caractères sociaux, financiers... peuvent s'ajouter à ceux-ci.

I -3-2.Surveillance d'un système automatisé :

Il s'avère très difficile en pratique d'intégrer dans une Partie Commande la totalité des savoir-faire humains de sorte que l'automatisation reste souvent partielle : certaines tâches restent confiées à des intervenants humains.

I -4. STRUCTURE D'UN SYSTEME AUTOMATISÉ :

Un système technique automatisé est composé de deux parties essentielles :

Partie commande (PC).

Partie opérative (PO).

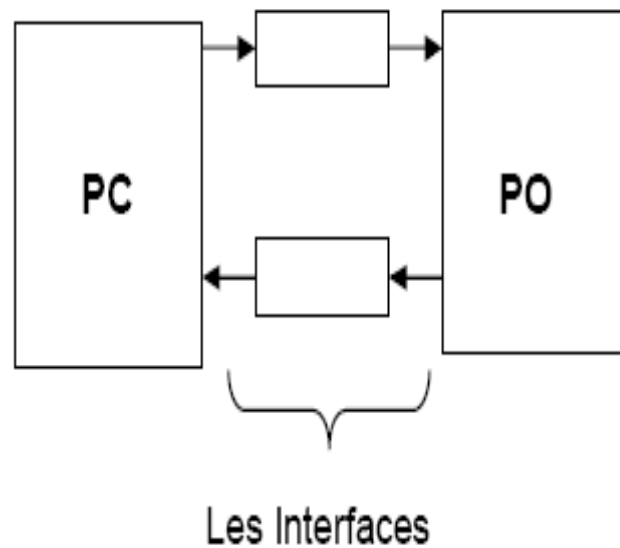
Les deux parties sont en relations par l'intermédiaire des éléments d'interface.

I-4-1.La partie commande :

C'est l'ensemble des moyens de traitement de l'information qui assure la commande et la coordination des tâches de la partie opérative, à partir de programmes pré-établis.

I -4-2.La partie Opératives :

C'est l'ensemble des moyens techniques qui effectuent directement la transformation de la matière d'oeuvre a partir des ordres de la partie commande et de l'opérateur.



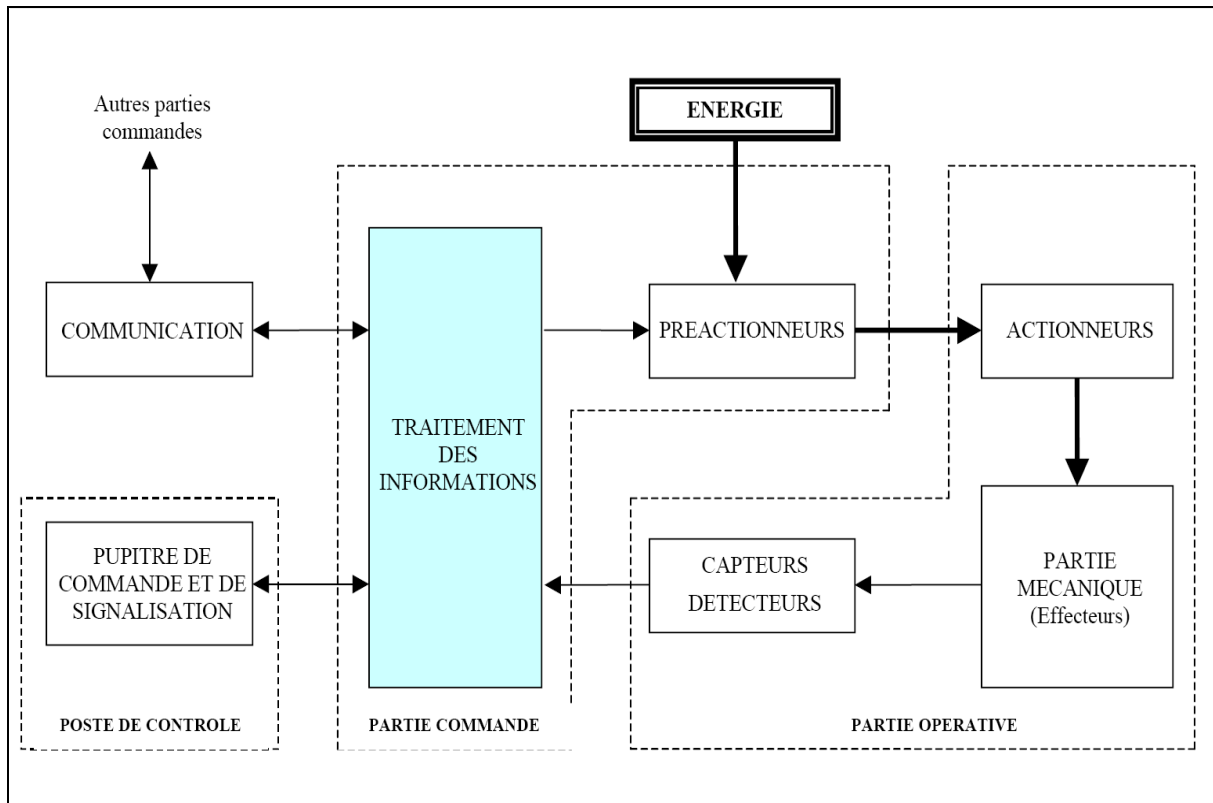
Fig(I-1) Structure d'un système automatisé 1.

I -4-3.Les interfaces :

Ces éléments sont des constituants de communication chargés de rendre possible le dialogue entre (PC) et (PO).

On distingue deux types d'organes.

- Préactionneurs : qui traduisent les ordres de la (P.C)
- Capteurs : qui captent les changements d'états de la partie opérative .



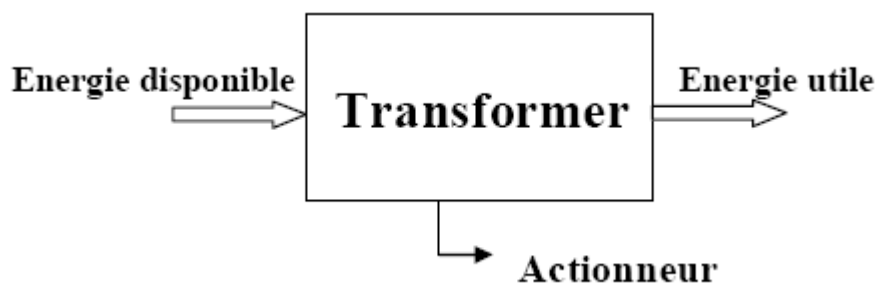
Fig(I-2) Structure d'un système automatisé 2.

I -5-ETUDE DES ÉLÉMENTS D'UN SYSTEME AUTOMATISÉ:

I -5-1.Etude des éléments de La Partie Opérative :

I- Les actionneurs :

L'actionneur a pour fonction de transformer l'énergie disponible en une énergie utile.

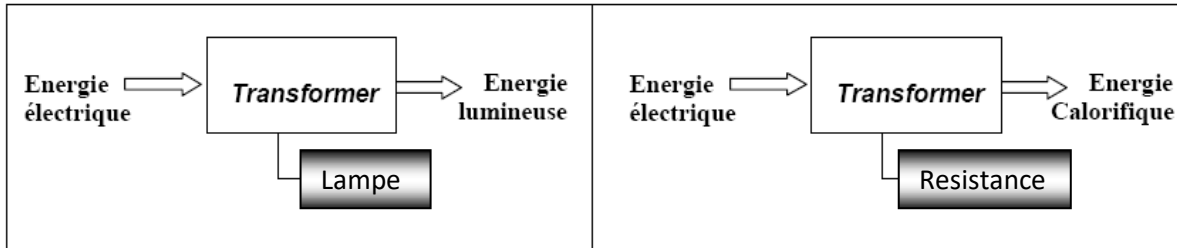


Suivant l'énergie de sortie on distingue deux types d'actionneurs.

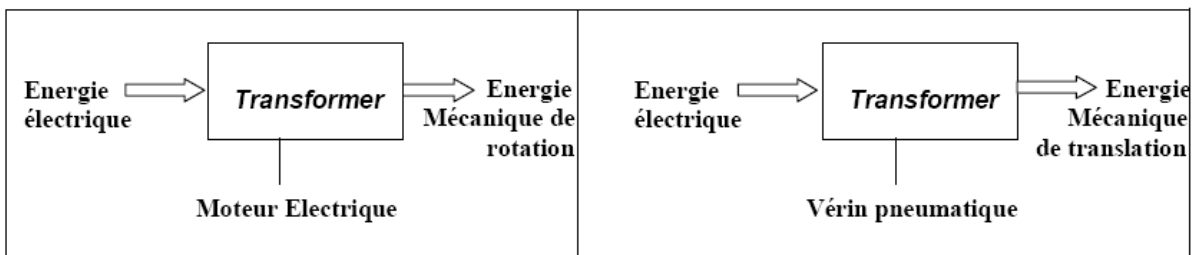
- Les actionneurs statiques dans l'énergie de sortie n'est pas mécanique.
- Les actionneurs dynamiques dans l'énergie de sortie est de type mécanique.

a- les actionneurs statiques :

Resistance

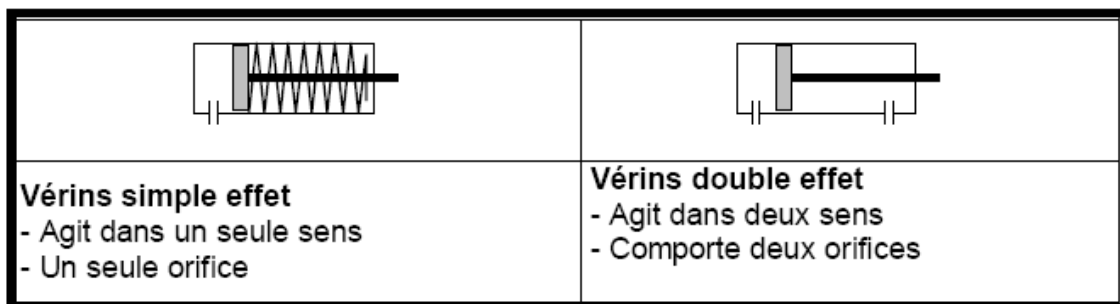


b- les actionneurs dynamiques :



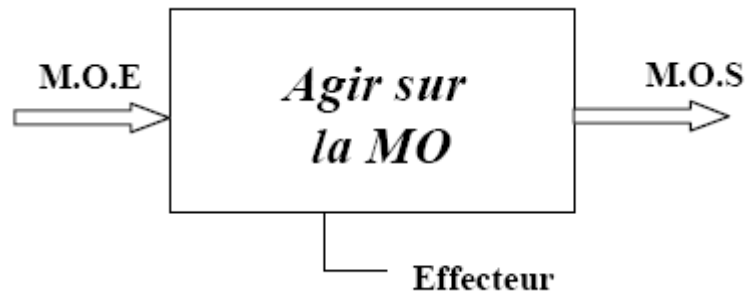
Il y a deux types de moteurs électriques

- Moteur a courant continu. (Ex : 6v continu)
- Moteur a courant alternatif. (Ex : 220v alternatif)
- Les électrovannes. (CA-220-24V et CC 24-48V)

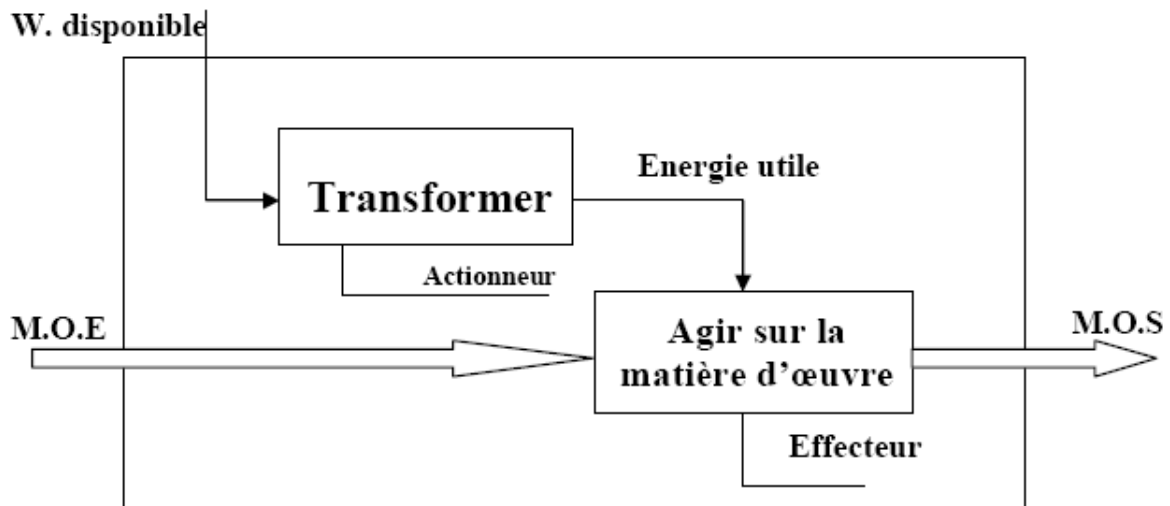


I I - Les effecteurs :

L'effecteur est un élément qui convertit l'énergie reçue de l'actionneur en une opération ou un effet pour agir directement sur la matière d'oeuvre en lui attribuant une valeur ajoutée.



Relation entre actionneurs et effecteurs :

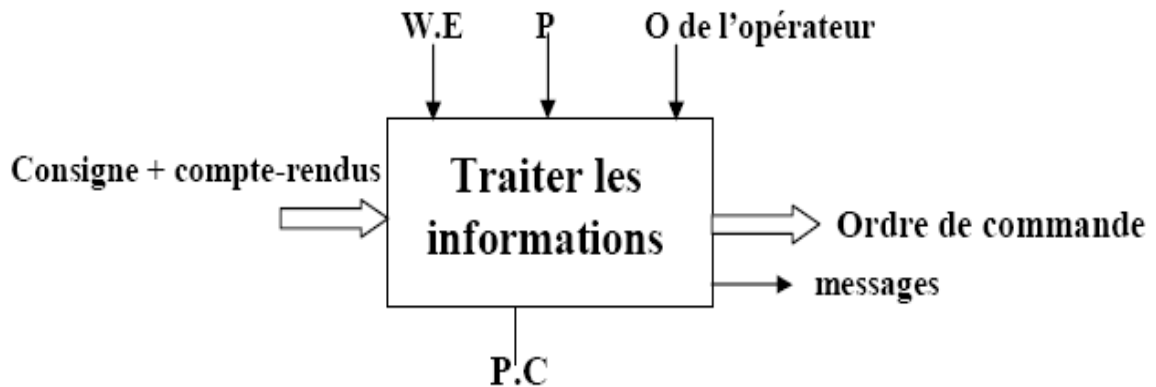


I -5-2.Partie commande :

La partie commande est constitué par deux parties :

- Partie traitement (partie commande).
- Les éléments de dialogue avec l'opérateur (Pupitre).

I – Partie traitement :



II- Pupitre :

Il est nécessaire de :

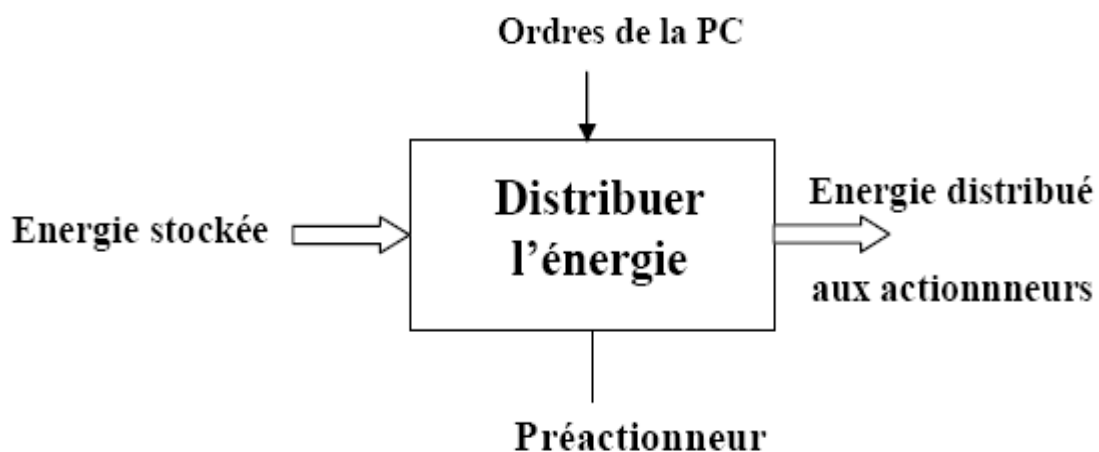
Donner en permanence à l'utilisateur des informations sur l'état du système à l'aide des :

- voyants : qui délivrent une information visuelle logique.
- Sonneries : qui délivrent une information sonore.
- Ecrans d'affichages numériques : qui délivrent des informations écrites.

I -5-3. Etude des éléments d'interfaces :

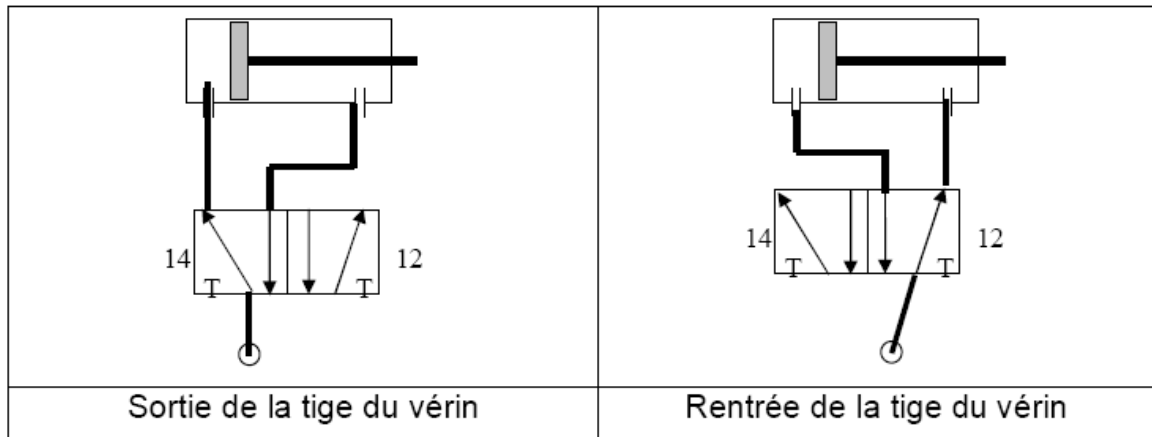
I -Les préactionneurs :

Le préactionneur est un organe qui assure la distribution de l'énergie disponible aux actionneurs sous un ordre de la P.C.



On distingue deux types de préactionneurs:

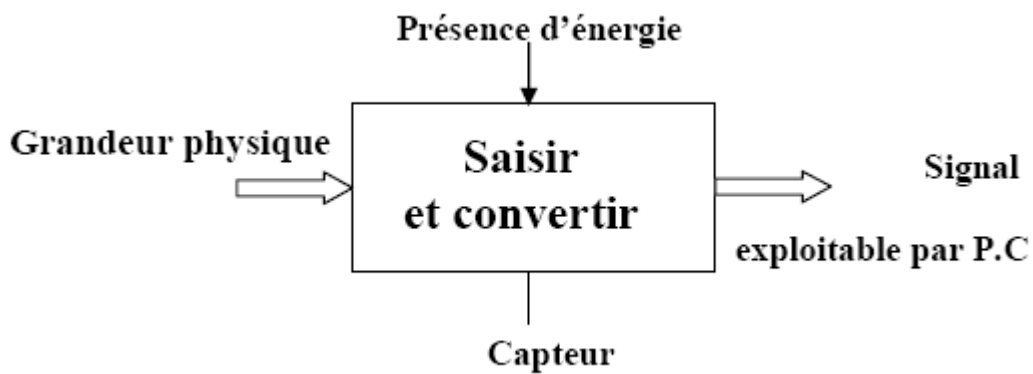
- Les préactionneurs électriques: les contacteurs.
- Les préactionneurs pneumatiques : les distributeurs.

Pilotage d'un vérin double effet par un distributeur 5/2 :

Exemple : Utilisation dans un filtre à translation.

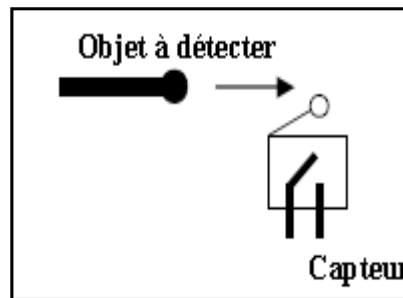
II – Les Capteurs :

Un capteur est un organe d'un système technique permettant de saisir (ou détecter) une grandeur physique (position, vitesse, température ...) et la transformer en signal exploitable par la P.C.



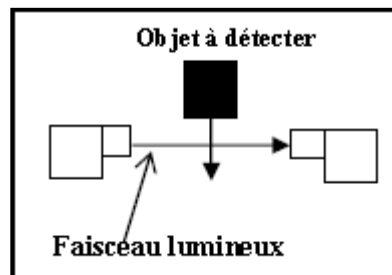
- Capteur à contact :

Pour la détection des objets.



-Capteur à distance :

Emission- Réception.



-Capteur à chute de pression.

-Détecteur de Niveau : Utilisé pour le contrôle de niveau des liquides.

-Les Pressostats : Contrôle de pression dans les conduites d'air comprimé et canalisation d'eau.

II-1. INTRODUCTION :

Les automates programmables industriels (API) sont des mémoires programmables pour contrôler les systèmes automatisés. Ces appareils de contrôle/commande largement répandus dans l'industrie, sont apparus aux États-Unis vers 1969 et en France vers 1971. Ils sont couramment utilisés dans des systèmes critiques (avions, ascenseurs, trains, . . .), aussi leur validation est primordiale.

Un API (en anglais PLC pour Programmable Logic Controller) est un ordinateur simplifié : il reçoit des données en entrée, celles-ci sont ensuite traitées par un programme et les résultats obtenus forment des sorties. Par exemple, un API peut commander les feux de route d'un carrefour, et en ouvrant et en fermant des vannes électriques. Un mécanisme plus complexe pourrait impliquer une balance sous le réservoir (comme entrée) et un contrôleur d'écoulement (comme sortie) permettant à l'eau de couler avec un débit commandé. Une application industrielle typique pourrait commander plusieurs réservoirs intégrés dans un processus tels que chaque réservoir doit satisfaire une variété de conditions comme : n'être ni trop plein ni trop vide, avoir le pH dans une certaine fourchette, etc.

Un API est donc un instrument de calcul et de commande qui est relié physiquement par une interface d'entrée à des capteurs et par une interface de sortie à des actionneurs figure (II-1). Il contient une unité centrale (CPU) qui gère l'ensemble du processus, elle contient le processeur, les mémoires vives et des mémoires mortes pour une taille débutant à 40 K octets. Elle est programmable par le biais d'une liaison spécifique et d'un logiciel adapté.

Un programme contient donc des variables spéciales permettant de représenter les entrées et les sorties. Ces variables, comme nous le précisons à la section suivante, sont mises à jour par l'API selon une procédure précise

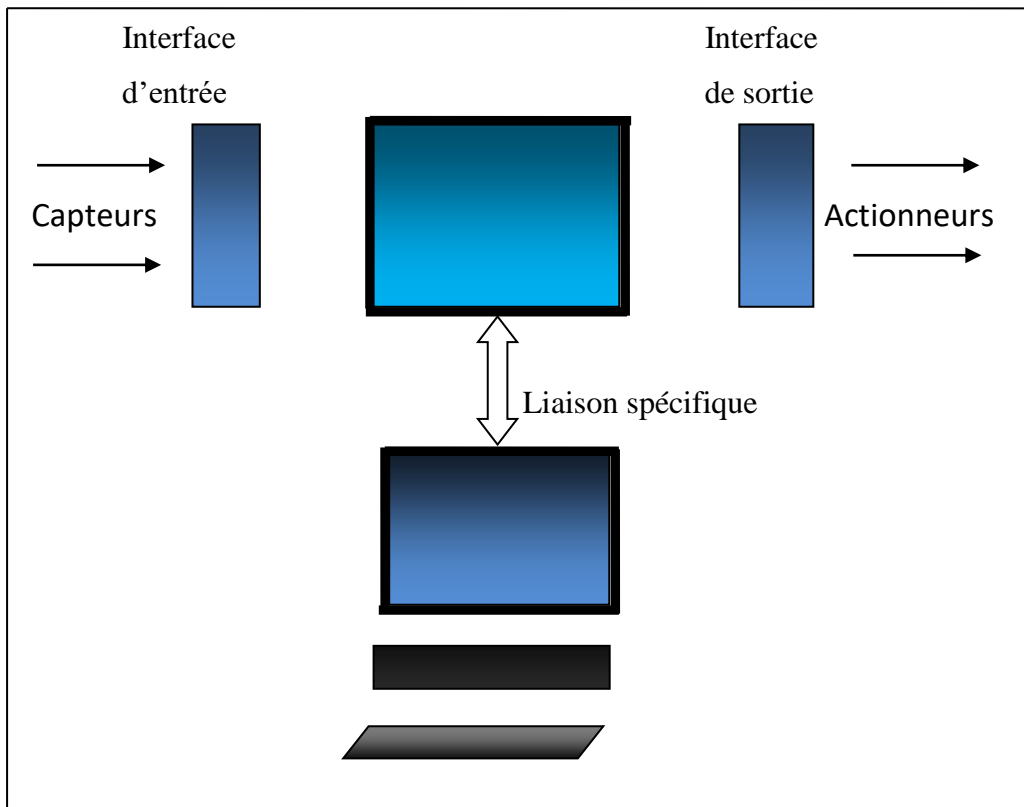


Fig (II-1) Automate programmable industriel.

II-2.PRINCIPE DE FONCTIONNEMENT :

L'automate programmable fonctionne par déroulement cyclique du programme. Le cycle comporte trois opérations successives qui se répètent normalement comme suit :

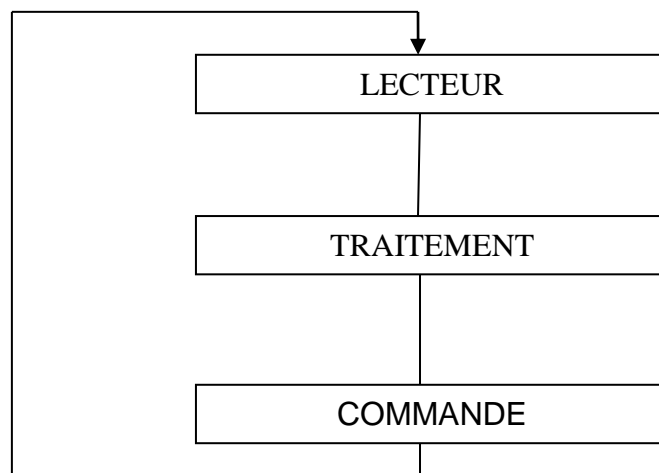


Fig (II-2) Principe de fonctionnement.

II-2.1. La lecture :

La scrutation des entrées binaires pour transférer leurs états da zone image des entrées.

II-2.2.Le traitement :

Le processeur exécute les instructions de la mémoire programme en fonction des informations de la mémoire des données. Cette exécution se traduit par la modification de certaines variables et leur mise à jour dans la zone correspondante.

II-2.3.La commande :

Les images des sorties dans la mémoire des données sont transférées dans le module de sortie pour être converti en signaux électriques pour la commande des préactionneurs et des dispositifs de visualisation. Ces valeurs sont verrouillées jusqu'au cycle prochain.

Ce cycle se répète infiniment tant qu'il n'y a pas d'interruption interne ou externe qui engendre l'arrêt temporaire ou permanent de l'automate. A chaque cycle seul, l'automate fait une mise à jour de ses données en entrée, garde cet état des entrées et passe à la phase de traitement. Cette dernière phase nécessite un temps prédéfini pour qu'elle se termine, dépendant de la fréquence du processeur et de la technologie interne et de la nature du traitement aussi.

Une fois terminée, on est dans la troisième et finale phase de sortie, où l'automate mets à jour ses signaux de sortie dépendant des résultats obtenus lors du traitement des entrées. Ces sorties restent figées jusqu'au prochain cycle.

Chaque fois que l'on minimise le temps d'un cycle, on améliore l'efficacité de notre automate. Malheureusement, le constructeur joue le rôle principal dans ce cas puisqu'il fixe la fréquence interne en se référant au processeur qu'il a utilisé. Mais l'utilisateur peut minimiser ce temps écoulé en améliorant le coût de son algorithme.

Par exemple, le temps d'un cycle pour une TSX de faible gamme est de l'ordre de 40ms.

Et son temps de conversion par une voie peut atteindre 15ms.

II-3. ARCHITECTURE DES AUTOMATE :**II-3.1. Aspect extérieur :**

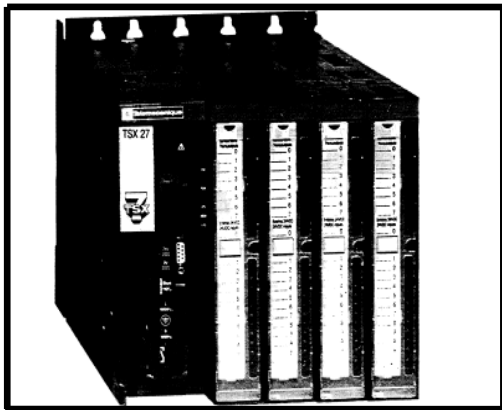
Les automates peuvent être de type compact ou modulaire.

Du type compact, on distinguera les modules de programmation (LOGO de Siemens, ZELIO de Schneider, MILLENIUM de Crouzet ...) des microautomates.

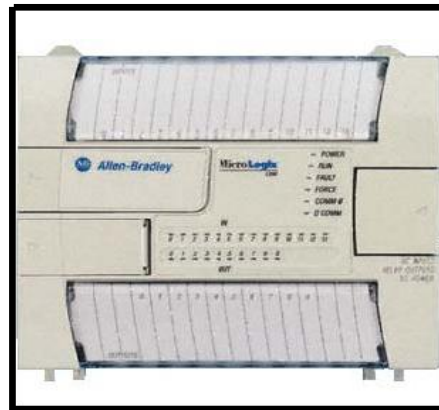
Il intègre le processeur, l'alimentation, les entrées et les sorties. Selon les modèles et les fabricants, il pourra réaliser certaines fonctions supplémentaires (comptage rapide, E/S analogiques ...) et recevoir des extensions en nombre limité.

Ces automates, de fonctionnement simple, sont généralement destinés à la commande de petit automatismes.

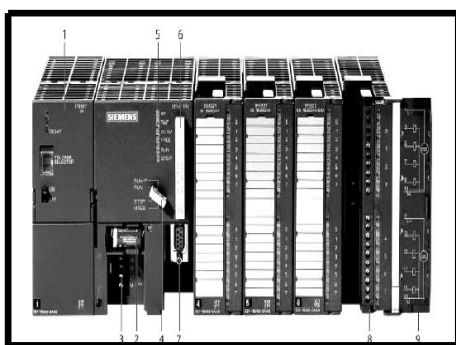
De type modulaire, le processeur, l'alimentation et les interfaces d'entrées / sorties résident dans des unités séparées (modules) et sont fixées sur un ou plusieurs racks contenant le "fond de panier" (bus plus connecteurs).



Automate modulaire TSX27



Automate compact (Allenbradley)



Automate modulaire (Siemens)



Automate modulaire (Modicon)

II-3.2 Structure interne :

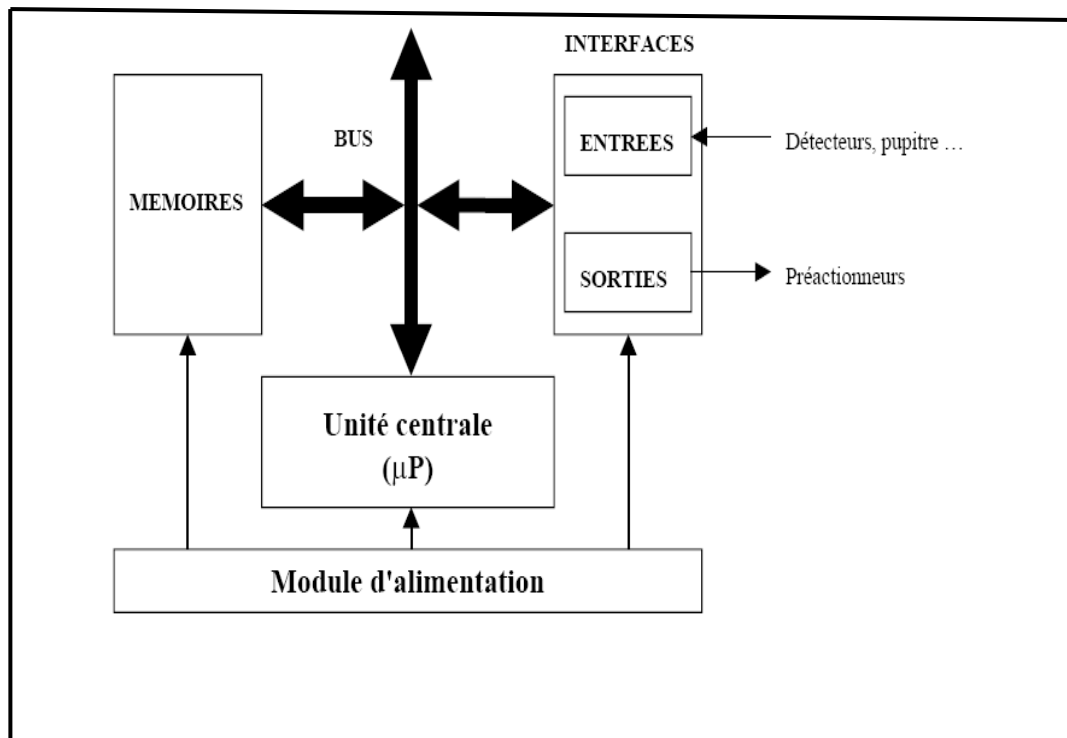


Fig (II-3) Structure interne de l'API.

a-Module d'alimentation :

Il assure la distribution d'énergie aux différents modules.

b-Unité centrale :

À base de microprocesseur, elle réalise toutes les fonctions logiques, arithmétiques et de traitement numérique (transfert, comptage, temporisation ...).

c- Le bus interne :

Il permet la communication de l'ensemble des blocs de l'automate et des éventuelles extensions.

d-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.

e-Interfaces d'entrées / sorties :✓ **Interface d'entrée :**

Elle permet de recevoir les informations du pupitre et de mettre en forme (filtrage, ...) ce signal tout en l'isolant électriquement (optocouplage).

✓ **Interface de sortie :**

Elle permet de commander les divers préactionneurs et éléments de signalisation du tout en assurant l'isolement électrique.

II-3.3Alimentation et câblage des entrées/sorties du TSX27 :

L'automate est alimenté généralement par le réseau monophasé 230V ; 50 Hz mais d'autres alimentations sont possibles (110V...ets).

La protection sera de type magnétothermique (voir les caractéristiques de l'automate et les préconisations du constructeur).

Il est souhaitable d'asservir l'alimentation de l'automate par un circuit de commande spécifique (contacteur KM1).

De même, les sorties seront asservies au circuit de commande et alimentées après validation du chien de garde, qui :

-Surveillance de C.P.U de façon à éviter les graves conséquences d'un dérèglement de celui-ci.

* A chaque cycle le C.P.U doit réarmer le chien de garde, sinon ce dernier entame les actions suivantes:

.Mise à 0 de toute les sorties.

.Arrêt de l'exécution du programme.

.Signalisation de la défaillance.

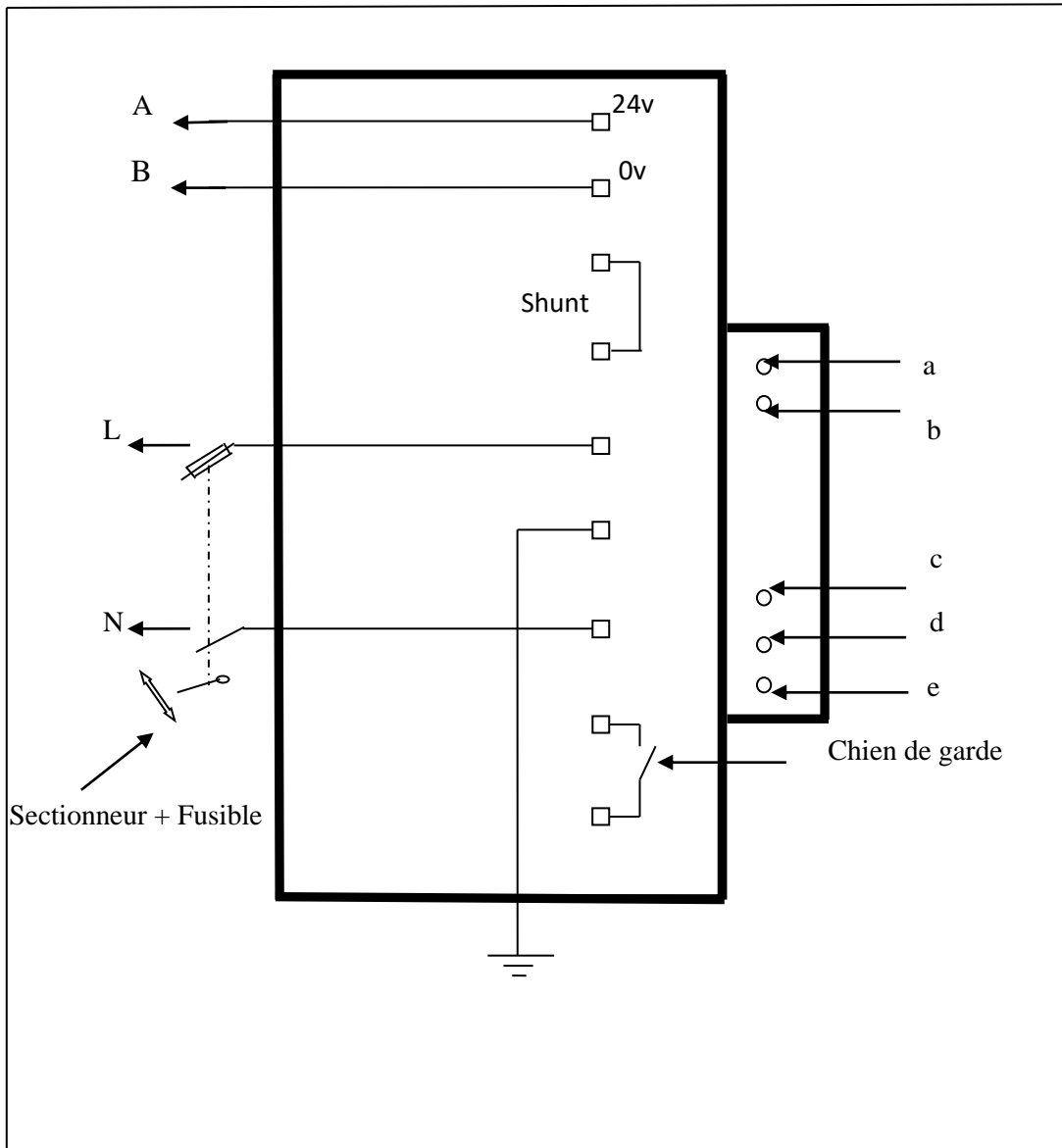


Fig (II-4) Alimentation et câblage des entrées/sorties du TSX27

- a) **PW (Vert)** : tension internes correctes.
- b) **RUN (Vert)** : programme utilisateur en cours d'exécution.
- c) **CPU (Rouge)** : défaut processeur.
- d) **MEM (Rouge)** : défaut cartouche mémoire utilisateur.
- e) **I/O (Rouge)** : défaut entrée/sorties.

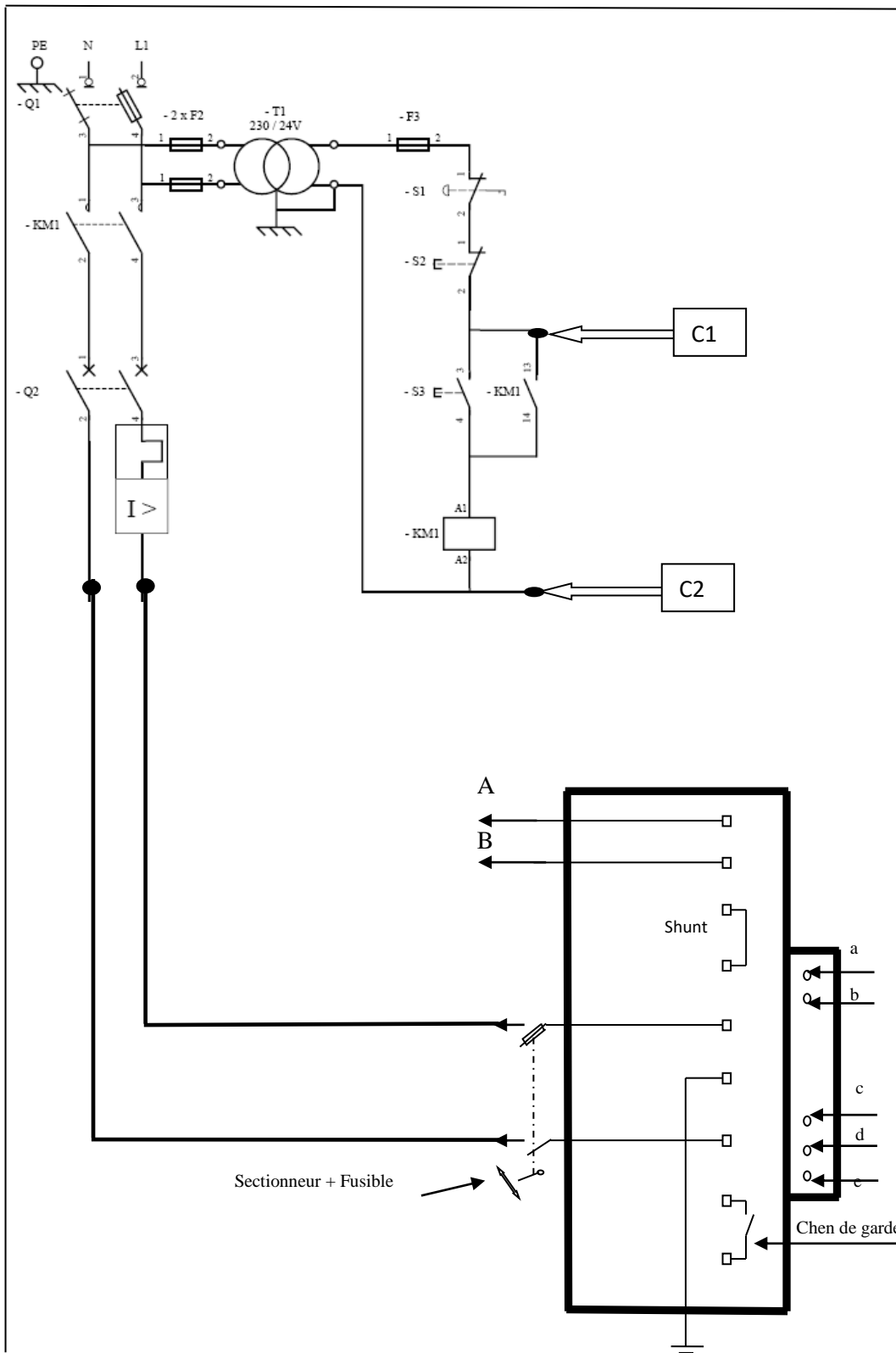
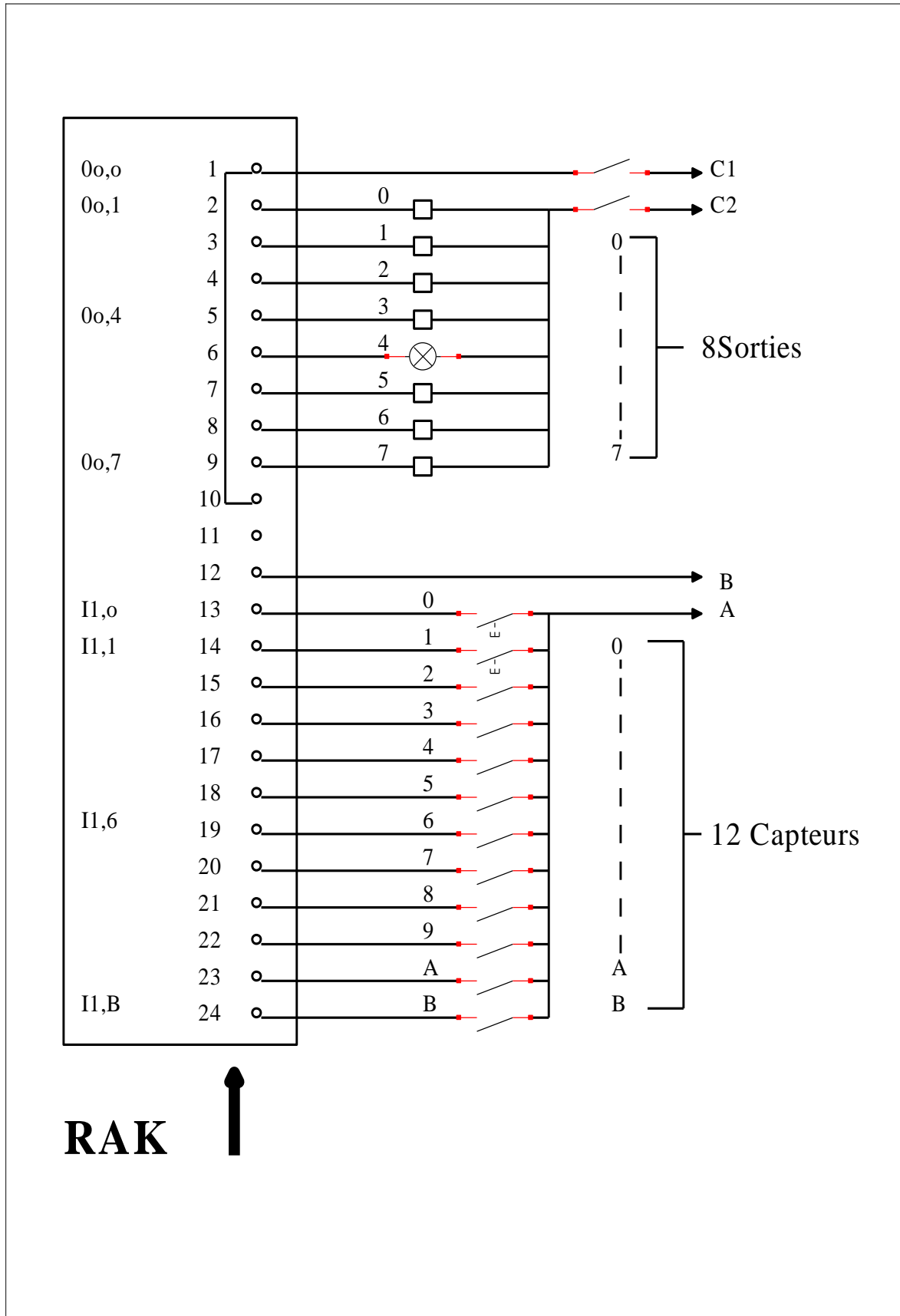


Fig (II-5) Alimentation et câblage des entrées/sorties du TSX27



II-4. ORGANISATION FONCTIONNELLE DE L'API :

La structure matérielle interne d'un API est donnée par le schéma ci-dessous :

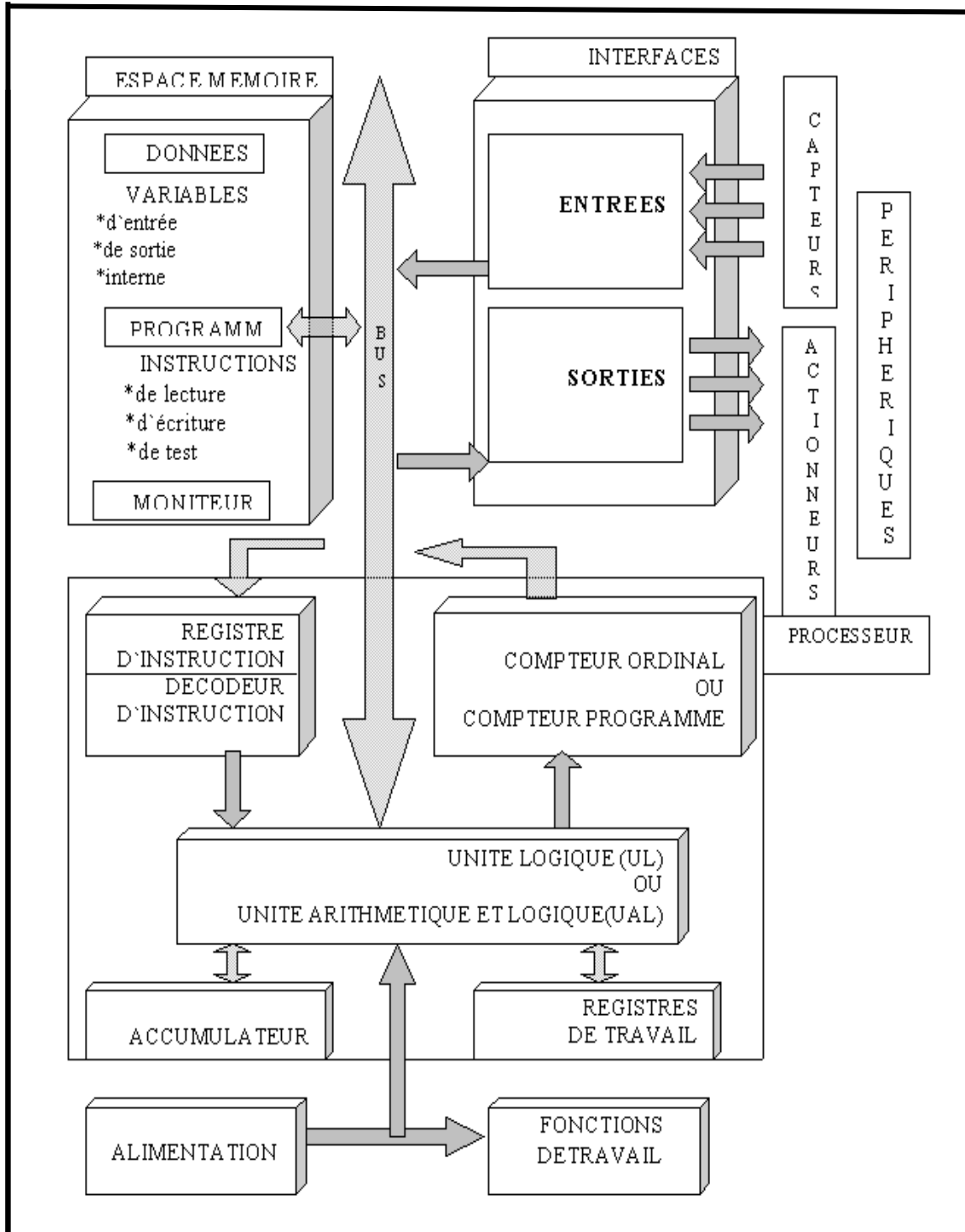


Fig (II-7) organisation fonctionnelle de l'API.

Un API se compose :

II-4.1. Unité de traitement un processeur :

Cœur de l'appareil, dans l'unité centrale, ce n'est sans doute pas, paradoxalement, le point le plus caractéristique, mais il conditionne tout de même largement les performances.

Dans le processeur on trouve à l'identique de l'unité de traitement du microprocesseur

-L'unité logique (UL) :

Qui traite les opérations logiques ET, OU et Négation ou l'unité arithmétique et Logique (UAL) qui exécute en plus les opérations arithmétiques.

-L'accumulateur :

Registre de travail, dans lequel sont rangés, temporairement, données et résultats.

-registre d'instruction :

Qui contient l'instruction à exécuter pendant la durée du traitement.

-le décodeur d'instruction :

Qui associe les microprogrammes de traitement au décodage de l'instruction.

-Le compteur ordinal ou compteur programme :

Où se trouve l'adresse de la prochaine instruction à exécuter, ce qui lui permet de gérer la chronologie de l'exécution des instructions du programme.

Pour assurer la liaison entre l'UC et les cartes d'entrées/sorties, un réseau, certaines consoles ou unités de dialogue, il faut une carte électronique spécialisée d'interfaçage

II-4.2. Éléments de stockage et de liaison :

Le stockage des données et des programmes s'effectue dans des mémoires. La mémoire vive (RAM) est volatile mais secourue par batterie. La mémoire morte (ROM) dont l'utilisateur ne peut que lire le contenu, éventuellement programmable (PROM) à l'aide d'outils spéciaux, contient le système d'exploitation, tandis que les programmes au point et utilisables peuvent se stocker dans des mémoires reprogrammables (EPROM), là encore avec un matériel spécifique.

La mémoire de données contient elle-même plusieurs zones :

-une zone de bits, dont certains peuvent être secourus en cas de défaillance de l'alimentation électrique ;

-une zone de mots, permettant de soumettre des données à un traitement plus large que le traitement booléen (traitement numérique ou alphanumérique).

La capacité de stockage d'une mémoire s'exprime en k octets (Ko) : (1Ko =1024 bits). Il faut connaître la capacité minimale utile de l'API, mais aussi la capacité maximale que l'on peut obtenir par diverses extensions. C'est la spécialisation des traitements qui permet de se satisfaire de telles capacités.

Le système d'exploitation impose aussi des limites par type, dans la programmation comme dans les données.

II-4.3. Modules d'entrées/sorties (E/S) :

Ils assurent le rôle d'interface de la partie commande (PC), qui distingue une partie opérative, où les actionneurs agissent physiquement sur le processus, et une partie commande récupérant les informations sur l'état de ce processus et coordonnant en conséquence les actions pour atteindre les objectifs prescrits (matérialisés par des consignes). Pour ce faire, ils doivent :

- Regrouper des variables de même nature, pour diminuer complexité et coût.
- Assurer le dialogue avec l'UC.
- « Traduire » les signaux industriels en informations API et réciproquement, avec une protection de l'UC et un traitement adéquats.

Les modules d'entres/sorties (E/S) Présentés en format standard ou en demi -format on distingue :

- Les modules d'entrées/sorties tout ou rien qui permettent, d'une part, de recevoir Les signaux en provenance des capteurs de réaliser des fonctions D'acquisition, d'adaptation, d'isolement galvanique, de filtrage et de protection contre les signaux parasites et d'autre part de réaliser des fonctions de mémorisation des ordres donnés par le processeur et la commande des préactionneurs au travers des circuits de découplage et d'amplification.
- Les modules d'entrées/sorties analogiques et les modules de régulation qui permettent la mise en œuvre simple de boucles de régulation.
- Les modules de comptage qui assurent le contrôle de pièces ou d'évènements le groupage d'objets, les mesures de vitesse de fréquence de position d'éléments, le contrôle de flux entrant et de flux sortant.
- Les raccordements doivent être effectués par bornes ou par connecteurs.
- Les entrées reçoivent des informations en provenance des éléments de détection et du pupitre opérateur.

➤ Les sorties transmettent des informations aux préactionneurs et aux éléments de signalisation du pupitre.

a- Interfaces d'entrées:

Elles sont destinées à :

Recevoir l'information en provenance du capteur Traiter le signal en le mettant en forme, en éliminant les parasites et en isolant électriquement l'unité de commande de la partie opérative.

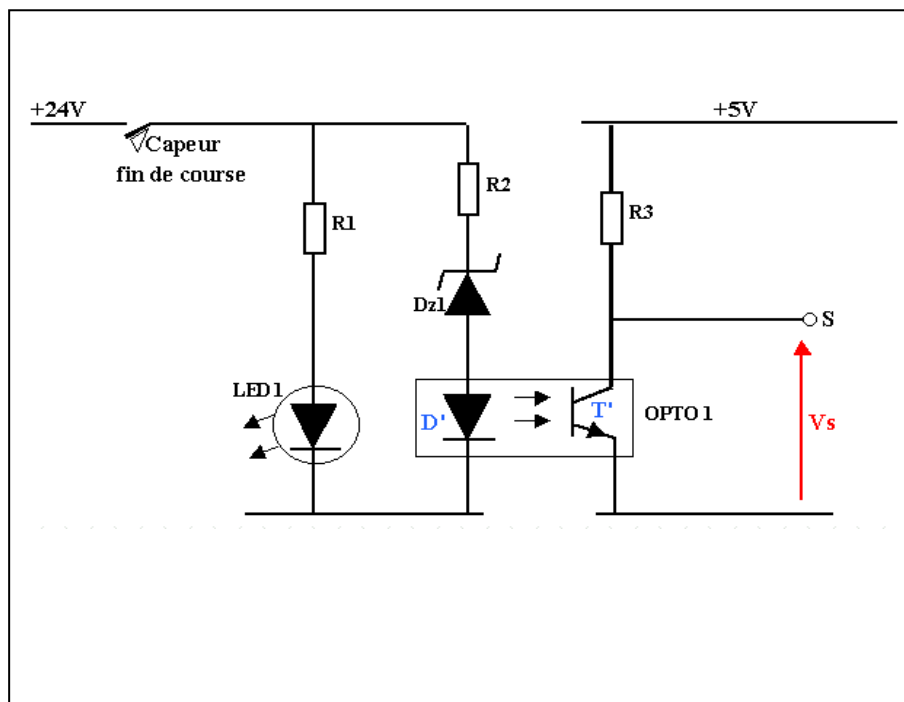


Fig (II-8):Interfaces d'entrées

b- Interfaces de sorties:

Elles sont destinées à :

Commander les préactionneurs et éléments des signalisations du système

Adapter les niveaux de tensions de l'unité de commande à celle de la partie opérative du système en garantissant une isolation galvanique entre ces dernières

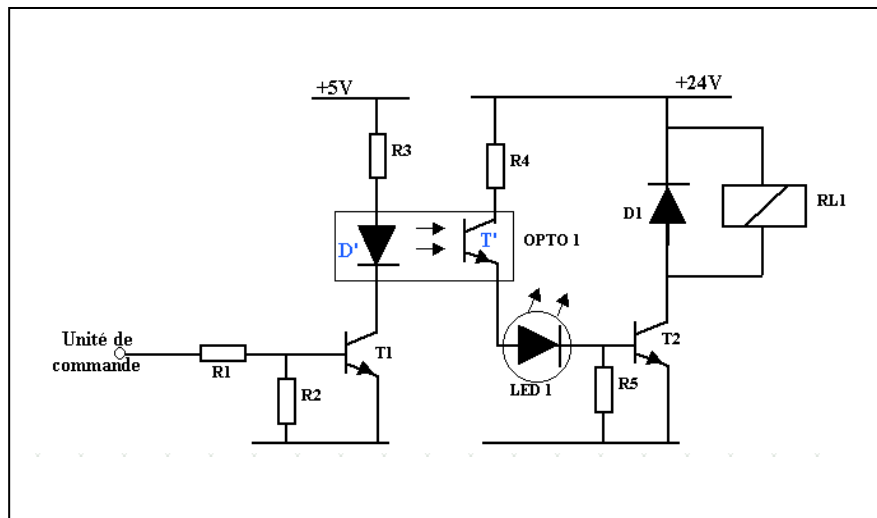


Fig (II-9): Interfaces de sorties

II-4.4. Auxiliaires

Il s'agit principalement :

- De l'alimentation électrique qui a pour rôle de fournir les tensions continues que nécessitent les composants (5 V, 12 V...) avec de bonnes performances, notamment face aux microcoupures du réseau. Sa source d'énergie est normalement le réseau électrique, parfois du 24 V continu. Il ne faut pas oublier que les châssis d'extension et les entrées/sorties déportées doivent aussi disposer d'une alimentation. Il est parfois nécessaire, pour lutter contre les perturbations électriques, d'introduire un transformateur d'isolement. C'est notamment le cas pour les raccordements à un réseau électrique à « neutre flottant ». Un onduleur évite les risques de coupure si celles-ci risquent de dépasser les tolérances admises.

- D'un ventilateur indispensable dans les châssis comportant de nombreux modules ou dans le cas où la température ambiante est susceptible de devenir assez élevée (plus de 40 °C);

- Du support mécanique. Il peut s'agir d'un rack (structure métallique accueillant des cartes avec généralement un raccordement arrière), l'automate se présentant alors sous forme d'un ensemble de cartes, d'une armoire, d'une grille et des fixations correspondantes

- D'indicateurs d'état concernant la présence de tension, l'exécution du programme (mode RUN), la charge de la batterie, le bon fonctionnement des coupleurs...

II-5. COMMUNICATION D'API :

L'automate doit pouvoir se connecter à d'autres matériels à processeur, et dialoguer avec les agents d'exploitation.

II-5.1. Besoins de communication :

L'API ne se borne pas à communiquer avec le processus qu'il pilote via ses modules d'E/S. Parmi les autres types de relations susceptibles d'être assurées, nous citerons seulement

- la communication avec un opérateur par un pupitre ou un terminal industriel.
- l'affichage local de valeurs numériques ou de messages.
- les échanges d'informations avec d'autres API ou systèmes de commande.
- les échanges d'informations avec des capteurs et actionneurs intelligents.
- les échanges d'informations avec une supervision.
- les échanges d'informations avec un processeur maître, ou, au contraire, avec des esclaves, dans le cadre d'un réseau.

II-5.2. Outils de communication :

Nous nous intéressons ici aux outils « directs », les outils indirects via un réseau local. Ce sont essentiellement :

- Des éléments de saisie d'information (boutons poussoirs, inverseurs et dispositifs à deux positions). Il s'agit là d'outils simples et robustes, mais limités à une faible quantité d'information (ordres de marche automatique, cycle par cycle, pas à pas, d'arrêt, consignes numériques...);
- Des éléments transmettant des informations (voyants, alarmes sonores, afficheurs sept segments)

Il existe des outils plus élaborés :

- *les afficheurs à circuits spécialisés, qui servent de boîte à message pour des avertissements, des informations sur le déroulement de la production, etc.
- *les terminaux industriels, qui permettent une communication homme- machine plus large que précédemment, et ce dans les deux sens (clavier alphanumérique, écran à affichage graphique ou semi graphique, etc.).
- *les périphériques informatiques (une imprimante peut être connectée à un API, généralement au moyen d'une liaison série) ;
- *la supervision dont le rôle dépasse largement la communication entre API et opérateur, car il concerne l'ensemble du système automatisé de production, mais le(s) poste(s) de

supervision n'en constitue pas moins un outil de communication à distance pour recevoir des informations de l'automate, lui donner des ordres (la supervision n'échangeant pas directement avec les capteurs et actionneurs), voire changer certains de ses paramètres. Une supervision se compose au moins d'un processeur support, d'un logiciel spécialisé, d'une base de données, d'un clavier opérateur, d'un écran (souvent de grand format), et de liaisons avec les processeurs supervisés.

II-5.3. Outils de programmation:

Leur rôle consiste à :

- écrire et modifier le programme de l'API ;
- le stocker ;
- le visualiser en temps réel via l'état des E/S et variables intermédiaires des algorithmes de commande ;
- modifier certaines données (bits, mots, présélection de compteurs et temporisateurs) ;
- aider au diagnostic des défauts.

Ils se composent d'une partie matérielle (console spécifique ou ordinateur) et d'un logiciel de programmation.

-Consoles de réglage :

Terminaux de poche, ces outils permettent, par connexion directe sur site, de modifier des valeurs de temporisation, de compteurs, des bits ou des mots, parfois le programme lui-même. Leur faible coût, leur faible encombrement et même dans certains cas leurs possibilités limitées (empêchant l'opérateur d'exploitation d'accéder à des parties du logiciel considérées comme intangibles) en font des outils complémentaires appréciables.

-Consoles spécifiques :

Munis d'un clavier spécifique, d'un écran plat permettant l'affichage de 5 à 20 lignes de programme et/ou d'une représentation graphique de type GRAFCET ou schéma à contacts, ces outils dont le prix peut dépasser celui d'un automate disparaissent au profit du PC console, aux possibilités plus étendues. On continue à les utiliser dans le cas où la mise au point et la maintenance doit s'effectuer au plus près de la partie opérative, car elles offrent par contre des fonctionnalités plus étoffées que les consoles de réglage.

-PC console :

Il apporte les facilités liées à l'emploi de cet outil de grande diffusion : visualisation sur écran, fenêtrage (pour les logiciels sous Windows, de plus en plus répandus), outils d'édition et d'affichage, sauvegarde externe, structuration des applications en pages correspondant

chacune à un écran... La mise au point se trouve également facilitée par l'affichage des valeurs prises par différents bits ou mots (à plus large échelle que sur les consoles spécifiques), la mise en surbrillance ou le changement de couleurs d'éléments graphiques. De même encore sont facilités les changements de langage, la création de bibliothèques de blocs Fonctionnels.

Tous les grands constructeurs ont aujourd'hui leur logiciel pour PC :

PL7 pour la gamme Télémécanique chez Schneider Electric, P8 chez Alstom (Cegelec) Les logiciels indépendants (AUTOMGEN) sont plus rares car il est difficile de garantir leur adéquation au matériel en toutes circonstances.

II-6.CHOIX D'UN API :

Il faut considérer particulièrement les points suivants :

-le nombre d'E/S :

Attention aux possibilités d'extension et à leur forme (même châssis, rajoutés, E/S déportées) .

-la taille du programme :

Outre la limitation de taille globale par la mémoire programme, peuvent exister des limitations catégorielles liées à un type de programmation ; peu d'API pourraient traiter un programme correspondant à des milliers d'étapes GRAFCET, ce qu'atteignent des applications dans le nucléaire, par exemple.

-la vitesse de traitement:

Rappelons qu'elle est variable et notons que, sans exiger plusieurs processeurs, certaines machines offrent la possibilité de « tâche rapide », avec un gain en vitesse mais des fonctionnalités restreintes pour ces tâches.

-les fonctions complémentaires:

Temporisateurs et compteurs; leurs caractéristiques (gamme de base de temps, cadence de comptage, types de fonctionnement tels que sorties impulsionnelles, types de remises à zéro) sont aussi importantes que leur nombre;

-les langages de programmation:

-le nombre de voies analogiques:

Attention au nombre de sorties, à la quantification et aux modes de conversion analogique/numérique et réciproquement, qui influent sur les temps de réaction ; ainsi qu'à la manière dont est géré le temps réel en régulation.

-le nombre et le type de liaisons inter processeurs:

L'existence de plusieurs liaisons potentielles, en particulier de liaisons série, est un avantage.

-la capacité de traitement arithmétique:

Elle est encore plus difficile à évaluer que pour le TOR ; certains tests portant sur une catégorie d'opérations apportent des indications partielles mais utiles.

-les modes de communication:

Il est souhaitable de disposer de plusieurs protocoles hors le protocole propriétaire ; noter aussi les capacités d'échange : MODBUS est un protocole très répandu, mais qui gère des échanges assez lents.

II-6. QUELQUES TYPES DES AUTOMATES PROGRAMMABLES INDUSTRIELLES:

Nous citons quelques types d'API du groupe siemens et Schneider et on s'intéresse au TSX27 (télé mécanique) qui fait partie de notre travail :

II-6.1. Groupe Siemens :**• Simatic S7-300 :**

- API : puissant, économique
- la pile qui devra être changée tous les ans
- CPU 316-2DP, les temps d'exécution pour : opération sur mots 1us.
- est un automate sans maintenance
- logiciel de programmation STEP7

• Simatic S7-400 :

- PS (power supply):PS405/PS407, input voltage (24/48/60) VDC et courant (2/1/0.8) A. output voltage (5.1/24) VD Cet courant (4/0.5) A.
- on a placement de pile mais c'est machine ne utilise pas pile.
- les E/S : 160/128.

-logiciel de programmation STEP7.

• Simatic S7-200 :

La famille S7-200 est constituée de micro automates programmables utilisables dans des applications d'automatisation variées. Ses possibilités d'expansion, son faible prix et son important jeu d'opérations en font une solution idéale pour la commande de petites

applications. En outre, le large choix de tailles et de tensions de CPU vous offre la souplesse nécessaire pour résoudre vos problèmes d'automatisation.

Un système d'automate programmable (AP) S7-200 de base, comprenant. Une unité centrale (CPU) S7-200, un ordinateur personnel, le logiciel de programmation STEP 7-Micro/WIN et un câble de communication. Pour pouvoir utiliser un ordinateur personnel (PC), vous devez disposer :

- _ D'un câble PC/PPI,
- _ D'une carte de processeur de communication (CP) et d'un câble d'interface multipoint (MPI)
- _ Ou d'une carte d'interface multipoint (MPI). Un câble de communication est fourni avec la carte MPI.

II-6.2. Groupe Schneider :

-TSX NANO :

-Sa faible taille permet son installation dans les coffrets, directement dans les machines ou sur les installations mobiles.

-Il est disponible en trois tailles et permet d'obtenir de 10 à 48 entrées/sorties.

-Possède de nombreuses fonctions: mémoire EEPROM, batterie, horodateur, potentiomètre et d'autre fonction d'automatismes:

Temporisateurs, compteurs/décompteurs, registres mots, registres bits à décalage pas à pas. Programmateurs cycliques.

-compatible avec des constituants d'automatisme: détecteurs de proximité, cellules photoélectriques, contacteur.

-Programmation: Logiciel PL7 sous DOS (langages liste d'instructions et à contacts).

-mémoire RAM interne sauvegardée 1k instructions langage List, 256mots de données et 68 mots de constantes.

-TSX PREMIUM :

Éléments de base de la plate-forme d'automatisme TSX premium montés sur rack:

PS: module d'alimentation.

00: module processeur de régulation.

De 01à06: autre modules:

-modules d'entrées/sorties tout ou rien.

-modules d'entrées/sorties Analogiques.

- module de sécurité intégré.
- module coupleur.
- module de communication.
- module ventilation.
- un rack peut comporter de 4 à 16 emplacements.
- Mémoire RAM interne sauvegarde de 48 K mots, extensible par PCMCIA jusqu'à 128 K mots.
- Logiciel PL7 Junior, liste d'instructions (IL), langage à contacts (LD), langage littéral (ST), langage Grafset (SFC).

-TSX MICRO :

- API modulaire, puissant, économique et convivial.
- La gamme comprend trois configurations de base.
- Elles intègrent un grand nombre de fonctions: visualisation, liaison multipoint, alimentation capteurs.
- Rapidité et puissance adaptées aux machines.
- système d'exploitation multitâche.
- 16 tâches événements.
- 17 k instructions programme/22 k données.
- Entrées/sorties analogiques 8/12/16 bits multi gammes.
- Fonction intégrées: régulation, dialogue homme -machine.
- utilisation d'un logiciel PL7 Micro (3 langages: schéma à contact, liste instructions, grafset).

II-6.3.L' automate TSX27 (télé mécanique) :

L'automate programmable TSX27 a été conçu, industrialisé est fabriqué par TELEMECANIQUE avec le plus grand soin pour constituer, compte tenu des technologies électroniques actuelles, un produit présentant les meilleures caractéristiques de sûreté, sécurité et maintenabilité (aptitude à la maintenance).

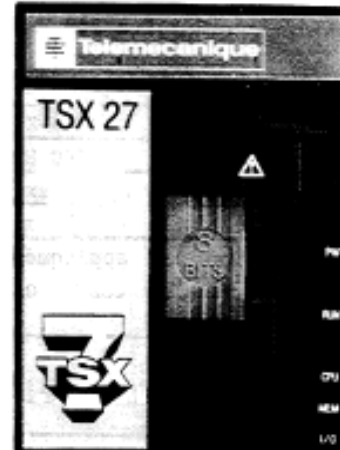
Ces caractéristiques ont pu être atteintes grâce, entre autre, à une automatisation poussée des fabrications (insertion automatique de composante...) accompagnée d'un contrôle permanent de la qualité. Les nombreux tests effectués de la réception des électroniques (circuits intégrés, de microprocesseur.....) au contrôle finale (déverminage pendant 72 heures avec cycle de température), sont le garant d'une qualité de produits

permettant la conception d'automatismes ayant le degré de sûreté et de disponibilité les plus élevés.

-Fiabilité :

La fiabilité d'un dispositif est son aptitude à accomplir une fonction requise dans des conditions d'utilisation définies pendant un temps donné.

Quel que soit le constructeur de dispositifs électroniques et malgré tous les efforts entrepris par celui-ci les taux de défaillance n'est jamais nul. Il faut donc s'attendre à ce qu'un automate pendant sa durée d'exploitation ait des défaillances. Il y a ainsi lieu de prévoir, lors de l'étude de l'automatisme, les dispositions nécessaires pour pallier ces défaillances et en limiter les conséquences, notamment vis-à-vis de la sécurité.



-Auto- tests intégrés :

Le TSX27 intègre l'autodiagnostic, c'est-à-dire qu'il comporte les auto-tests nécessaires à la détection d'une fraction importante des éventuelles défaillances internes ou de fausses manœuvres pouvant être commises lors des opérations de mise en œuvre, d'exploitation et de maintenance de l'automatisme. Selon leur nature, ces tests sont effectués lors de la mise sous tension, ou cycliquement (sur un ou plusieurs cycles de scrutation).

Ils portent principalement sur :

- ✓ Le processeur :

Contrôle du microprogramme, des mémoires RAM internes par calcul de « checksum », gestion du chien de garde,

- ✓ La cartouche mémoire :

Vérification de sa présence, de son type, de son contenu par calcul de « checksum »...

- ✓ Les entrées/sorties :

Vérification de leur présence, surveillance de l'absence de court-circuit en sortie, contrôle de l'échange des informations sur le bus.

Les résultats de ces auto-tests sont signalés par des voyants de signalisation en face avant et peuvent être approfondies à l'aide du terminal TSX407. Un contact du relais d'alarme incorporé au TSX27 est disponible pour les chaînes d'asservissement et de sécurité. Ces tests permettent également d'alerter l'opérateur et de lui faire prendre les décisions qui s'imposent, dont l'une, la plus souvent, est l'arrêt de l'automatisme.

-Aide au diagnostic :

Cependant, il est important de noter qu'une grande partie des défaillances, constatées au niveau d'une (90 à 95% du total) , sont externes au produit automate (capteurs, actionneurs, organes mécaniques, etc...).Il est donc recommandé d'utiliser la puissance de traitement de celui-ci pour fournir une assistance automatisée à la recherche de tels défauts.

➤ configuration TSX27 :

L'automate programmable TSX27 est proposé en configurations prêtes à l'emploi.

1. Configuration 20 E/S :

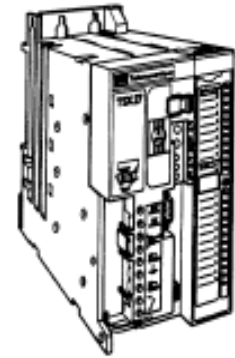
❖ Alimentation courant alternatif :

(110-127 /220-240VCA 50-60Hz)

-12 entrées 24VCC, 8 sorties relais, (réf. TSX 27 2220).

Alimentation courant continu (24VCC) :

-12 entrées 24VCC, 8 sorties 24VCC-2A, (réf. TSX 27 2211).

**2. Configuration 40 E/S :**

❖ Alimentation courant alternatif :

(110-127 /220-240VCA 50-60Hz)

-24 entrées 24VCC, 16 sorties relais,

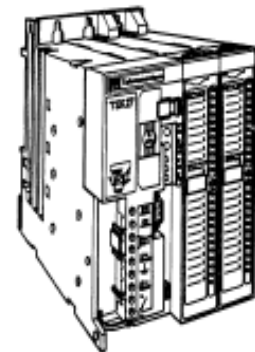
(Réf. TSX 27 2420).

-24 entrées 24VCC-2A, 8sorties relais,

8 sorties 24VCC, (Réf. TSX 27 2430).

❖ Alimentation courant continu (24VCC) :

-24 entrées 24VCC, 16sorties 24VCC-2A, (Réf. TSX 27 2411).

**3. Configuration 60 E/S :**

❖ Alimentation courant alternatif :

(110-127 /220-240VCA 50-60Hz)

-36 entrées 24VCC, 24 sorties relais,

(réf. TSX 27 2620).

-36 entrées 24VCC, 16 sorties relais,

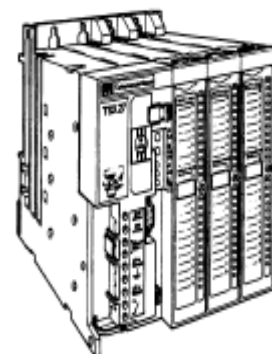
8 sorties 24VCC-2A, (réf. TSX27 2630).

-36 entrées 24VCC, 8 sorties 24VCC relais, 16 sorties 24VCC-2A

(réf. TSX 27 2640).

❖ Alimentation courant continu (24VCC) :

-36 entrées 24VCC, 24VCC sorties 24VCC-2A (réf. TSX 27 2611).



3. Configuration 80 E/S :

❖ Alimentation courant alternatif :

(110-127 /220-240VCA 50-60Hz)

-48 entrées 24VCC, 32sorties, 24VCC-2A,

(Réf. TSX 27 2810).

-48 entrées 24VCC, 16 sorties relais,

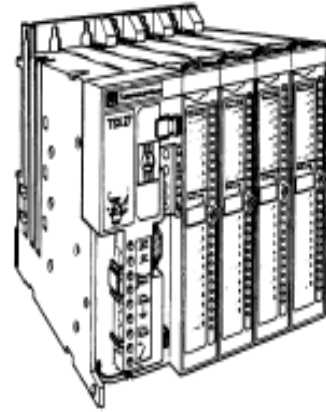
16 sorties 24VCC-2A (Réf. TSX 27 2830).

-48 entrées 24VCC, 8 sorties relais, 24 sorties 24VCC-2A

(Réf. TSX 27 2840).

❖ Alimentation courant continu (24VCC) :

-48 entrées 24VCC, 32 sorties 24VCC-2A (Réf. TSX 27 2811).



III-1.INTRODUCTION :

Un API est programmé à l'aide des langages spécialisés, fournis par son constructeur et utilisable à travers d'une interface (un logiciel sur PC, un pupitre ..). Ces langages peuvent être classés en cinq grandes familles. Cependant, deux langages de la même famille et fournis par deux constructeurs différents ne sont pas forcément compatibles, ce qui est de nature à nuire à la portabilité des applications et à limiter la réutilisation du code. C'est pour cette raison que la commission électrotechnique internationale a entrepris un grand effort de normalisation visant à uniformiser les langages utilisés dans les domaines de la programmation des API, ce qui a donné naissance à une norme. Ce standard définit cinq langages correspondant aux familles des langages les plus utilisées pour la programmation des API.

Les langages sont :

- Instruction List (IL) : un langage textuel de type assembleur.
- Structure text (ST) : un langage textuel structuré similaire au Pascal.
- Ladder Diagram (LD) : un langage graphique.
- Function Blok Diagram (FBD) : un langage graphique.
- Sequential Function Charts (SFC) [grafcet]:un langage graphique.

PL7-2 est le nom donné aux langages utilisés pour programmer les automates TSX27, TSX47. Ce sont deux langages graphiques :

- Langage à contacts : ce langage comprend tous les éléments nécessaires pour programmer des réseaux en schémas à contacts : contacts, bobine, blocs fonction.....etc
- Langage Grafecet : ce langage comprend tous les éléments ci-dessus avec en plus instructions d'étapes et de transition relatives au grafcet, les actions associées transitions étant écrites en schémas à contacts.

III-2. LES OBJETS ADRESSABLES DU PL7-2 :

III-2.1. Adressages des entrées/sorties du TSX27 :

Le TSX 27 est un automate modulaire qui possède 1, 2, 3 ou 4 cartes(RAK) d'entrées/sorties. Chaque carte possède 20 entrées/sorties (12 entrées et 8 sorties).

L'adressage d'une entrée ou d'une sortie est défini dans le programme par quatre caractères:

I/Oy,i I: Entrée (Input) O: Sortie (Output)

y : emplacement

, : virgule

i : numéro de voie dans la carte

0 à 7 pour les sorties

0 à B pour les entrées

Chaque carte d'entrées/sorties possède:

- en partie supérieure un ensemble de 8 sorties repérées par les emplacements 0, 2, 4 ou 6 dans l'automate, et par leurs numéros de voies de 0 à 7.

Exemple: O4, 5 = sixième voie de la carte de l'emplacement 4.

- en partie inférieure un ensemble de 12 entrées repérées par les emplacements 1, 3, 5 ou 7 dans l'automate, et par leurs numéros de voies de 0 à B.

Exemple: I3, A = onzième voie de la carte de l'emplacement 3.0

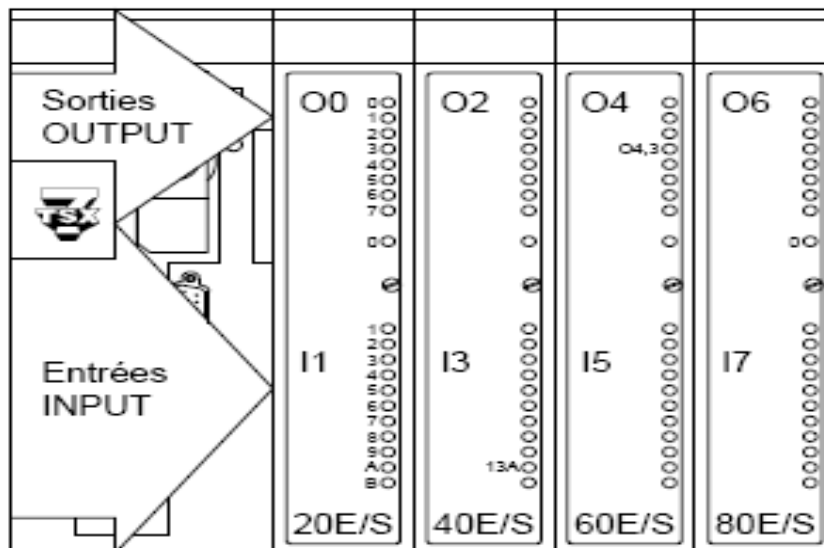


Fig (III-1) Adressages des entrées/sorties du TSX27

Tableau récapitulatif :

Sorties (Output)	O _{y,i}	y = 0, 2, 4, 6 i = 0, 1, ..., 7
Entrées (Input)	I _{y,i}	y = 1, 3, 5, 7 i = 0, 1, ..., B

III-2.2. Objets mots :

Les objets mots sont adressés par le logiciel PL7-2 sous le format mots de longueur 16 bits, situés dans la mémoire de données.

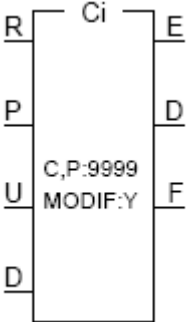
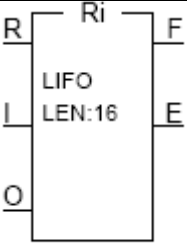
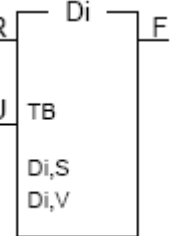
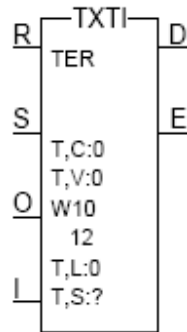
III-2.3. Objets blocs fonctions :

Le langage à contacts utilise une ressource appelée : blocs fonctions, lesquels se répartissent en deux groupes :

- les blocs fonctions d'automatismes : ils mettent à disposition de l'utilisateur des fonctions préprogrammées telles que temporisateurs, monostables, compteurs, registres,...etc.
- le bloc fonction texte : il permet d'échanger des données entre programme utilisateur et coupleur intelligent, ou prise terminal, ou système, ou autre programme utilisateur. Les blocs fonctions mettent en oeuvre des objets bits de sortie et mots spécifiques.

❖ **Blocs fonctions d'automatisme**

Temporisateur TSX 27(16)		<p>E : entrée "armement" C : entrée "contrôle" Ti,D : bit de sortie temporisation écoulée Ti,R : bit de sortie temporisation en cours Ti,P : mot valeur de présélection Ti,V : mot valeur courante</p>
00Monostable TSX 27 (8)		<p>S entrée sur front "départ" Mi, R : bit de sortie monostable en cours Mi, P : mot valeur de présélection Mi, V : mot valeur courante</p>

<p>Compteur/décompteur TSX 27(16)</p>		<p>R entrée "remise à zéro" P entrée "présélection" U entrée sur front "comptage" D entrée sur front "décomptage" Ci,E bit de sortie "débordement décomptage" Ci,D bit de sortie "présélection atteinte" Ci,F bit de sortie "débordement comptage" Ci,P mot valeur de présélection Ci,V mot valeur courante</p>
<p>Registre LIFO/FIFO TSX 27(4)</p>		<p>R entrée "remise à zéro" I entrée sur front "stockage" O entrée sur front "déstockage" Ri,F bit de sortie "registre plein" Ri,E bit de sortie "registre vide" Ri,I mot d'entrée du registre Ri,O mot de sortie du registre L nombre de mots du registre (999 max)</p>
<p>Programmeur R entrée "Remise A Zéro" cyclique TSX 27(8)</p>		<p>R entrée "Remise A Zéro" U entrée sur front "Avance de pas" cyclique Di,F bit de sortie "dernier pas en cours" Di,S mot "numéro du pas actif" Di,V mot "temps enveloppe" Di,Wjmot 16 bits d'ordres</p>
<p>Bloc fonction texte TSX 27(8)</p>		<p>R entrée validation arrêt échange S entrée sur front "lancement échange" O entrée sur état "émission" I entrée sur état "réception" TXTi,D bit de sortie "transfert terminé et reçu" TXTi,E bit de sortie "erreur de transfert" TXTi,L mot "longueur du message" (en octets) TXTi,S mot status : compte-rendu de l'échange TXTi,M mot adresse coupleur et numéro de voie TXTi,C mot code requête TXTi,R mot compte rendu échange TXTi,T num. bloc TXT en communi. TXTi,A adresse station réseau</p>

III-3.CARTOUCHES LANGAGE ET CARTOUCHES UTILISATEUR :

Il existe deux types de cartouches :

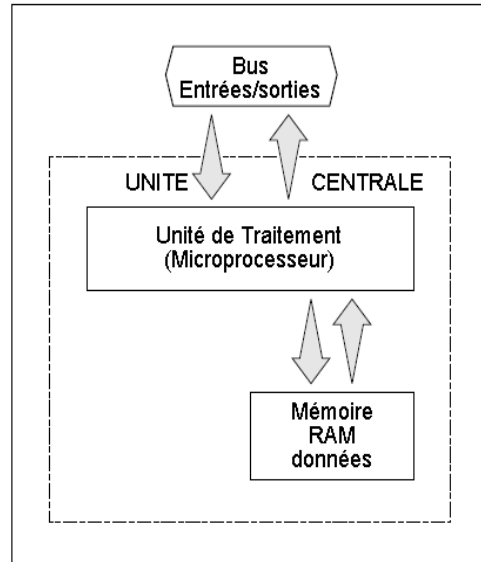
- ✓ Les cartouches langage.
- ✓ Les cartouches utilisateur.

a)-Cartouches langage :

Les cartouches langage contiennent tous les éléments des langages PL7-2 pour saisir, mettre au point, régler ou modifier un programme d'automate TSX27 ou TSX47. deux types de cartouches langage sont disponibles :

- Programmation et réglage en langage à contacts : Réf. TSX TS4 20*.
- Programmation et réglage en grafset et langage à contacts : Réf. TSX TS4 21*.

*Ces cartouches existent en 3 versions :
F Française, G : Allemande, E : Anglaise



fig(III-2) Cartouches langage

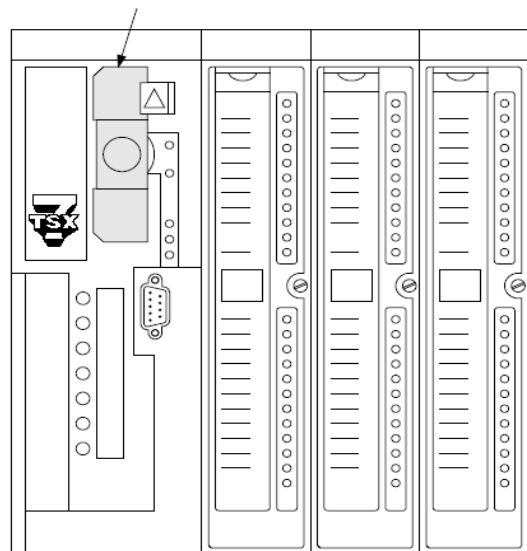
Les cartouches langage sont toujours insérées dans le terminal. La mise en place et le débrogage de ces cartouches peut s'effectuer sous tensions. En effet dès l'ouverture du couvercle. Une sécurité interrompt l'alimentation électrique de la cartouche.

b)-Cartouches utilisateur :

Cartouches utilisateur RAM CMOS ou EPROM

Les cartouches utilisateur stockent le programme de l'application à automatiser. En programmation avec le terminal connecté, ou en exploitation, une cartouche utilisateur doit être insérée dans l'automate.

Cette cartouche peut également être insérée dans le terminal, mais uniquement pour transférer ou dupliquer une mémoire.



Fig(III-3) Cartouches utilisateur

Deux types de cartouche utilisateur sont disponibles :

-Cartouche mémoire vive, pour créer et mettre au point un programme en RAM CMOS secourue par batterie : 8,16 ou 32 octets, référencée TSX RAM88 ,168 OU 328.La durée de la sauvegarde est de 6 semaines en dehors de l'automate.

Ne pas omettre de charger cette batterie en insérant préalablement la cartouche dans un automate sous tension pendant 24 heures.

-Cartouche mémoire reprogrammable, pour stocker le programme définitif en EPROM 8,16 ou 32 k octets, référencée TSX RPM 88,168 ou 328.

III-4.LES BITS SYSTEME : TSX 27

Les automates du type TSX 27 et TSX 47 disposent de 24 bits systems pour indiquer l'état de fonctionnement interne de l'automate.

L'état de ces bits peut etre testé pour détecter par exemple une panne dans l'automate et lancer une procedure de secours.

Bit	Fonction	Etat init.	Gestion (*)	TSX 27
SY0	1 = démarrage à froid (reprise secteur avec perte des données)	0	S ou U → S	x
SY1	1 = démarrage à chaud (reprise secteur sans perte des données)	0	S ou U → S	x
SY2	0 = adres. indexé W0 à W127 1 = adres. indexé W0 à W1023	0	U	
SY3	BT tâche rapide 0 = 10ms ; 1 = 5 ms	0	U	V≥3
	1 = gel de l'horodateur	0	U	
SY6	BT = 100ms		S	X
SY5	BT = 1s		S	X
SY7	BT = 1mn		S	X
SY8	0 = maintien des sorties sur STOP auto- mate	1	U	X
SY9	1 = mise à zéro des sorties automate en	0	U	X
SY10	0 = défaut E/S	1	S	X

SY11	1 = rafraîchissement variables COM réseau émises par station	0	S → U	
SY12	1 = réseau et coupleur, station réseau OK	0	S	
SY13	1 = pile absente ou défectueuse	0	S	
SY14	1 = visualisation du contenu de SW16	0	U	X
SY15	1 = le compteur rapide C31 est égal à sa valeur de présélection	0	S → U	
SY16	1 = demande de réarmement toutes les 10s sorties statiques déclenchées sur surintensité ou court-circuit	0	U → S	
SY17	1 = débord. sur calcul non signé ou décal. circulaire	0	S → U	X
SY18	1 = débordement sur calcul (16 bits)	0	U → S	X
SY19	1 = inhibition tâche rapide	1	U	$V \geq 3$
SY20	1 = débordement d'index	0	S → U	$V \geq 3.1$
SY21	1 = initialisation du GRAFCET provoque la mise à 0 des étapes et la mise à 1 des étapes initiales	0	U → S	X
SY22	1 = RAZ de GRAFCET	0	U → S	X
SY23	1 = validation du pré positionnement du GRAFCET Maintenu à 1 provoque le figeage du GRAFCET	0	U → S	X

(*) Abréviations utilisées :

S = mis à 1 ou 0 par le système,

U = mis à 1 ou 0 par l'utilisateur,

S → U = mis à 1 par le système, remis à 0 par l'utilisateur,

U → S = mis à 1 par l'utilisateur, remis à 0 par le système.

III-5.PRESENTATION DES LANGAGES PL7-2 :

III-5.1 Langage a contacte:

a) Principe

Un programme écrit en langage à contacts se compose d'une suite de réseaux exécutés séquentiellement par l'automate.

Dessiné entre deux barres de potentiel, un réseau est un ensemble d'éléments graphiques représentant :

- les entrées/sorties de l'automate (boutons-poussoirs, détecteurs, relais, voyants...),
- des fonctions d'automatismes (temporisateurs, compteurs...),
- des opérations arithmétiques et logiques et des opérations de transfert,
- les variables internes de l'automate (bits, mots, etc..).

Ces éléments graphiques sont reliés entre eux par des connexions horizontales et verticales, définissant ainsi des "réseaux de contacts".









Chaque réseau ainsi constitué comporte au plus 4 lignes et 10 colonnes et s'organise en 2 zones :

- la zone test, dans laquelle figurent les conditions nécessaires à une action,
- la zone action, qui sanctionne le résultat consécutif à un enchaînement de test.

PL7-2		Formulaire programmation schéma à contacts										
N°	ZONE TEST							ZONE ACTION	OBSERVATIONS			
<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>											()	Zone commentaires
											()	non mémorisés dans
	Commentaire mémorisé dans mémoire automate										()	l'automate
											()	(sur formulaire
											()	uniquement)

Fig(III-4)Formulaire programmation schéma à contacts

Des formulaires de programmation "schéma à contacts" sont prévus pour la représentation des réseaux. Ils sont conformes à la structure des réseaux, pour la programmation en langage PL7-2 : un réseau peut être visualisé sur l'écran du terminal, même en mode Run.

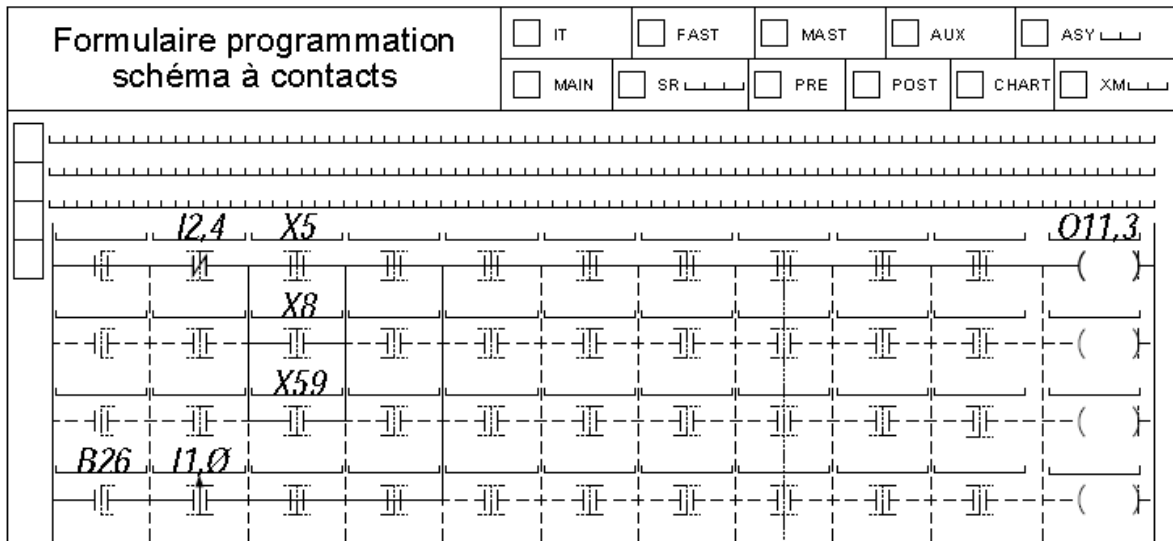
Désignation	Graphe	Fonctions
Elément à Contact	• Contact à fermeture 	Contact passant quand l'objet bit qui le test à l'état 1.
Elément à Contact	• Contact à ouverture 	Contact passant quand l'objet bit qui le pilote est à l'état 0.
Elément de liaison	• Connexion de liaison. 	Permet de relier en série les éléments graphiques de test et d'action entre les deux barres de potentiel.
Elément de liaison	• Connexion verticale 	Permet de relier en parallèle les éléments graphiques de test et d'action.
Elément d'action	• Bobine directe 	L'objet bit associé prend la valeur du résultat logique de la zone test.
Elément d'action	• Bobine inverse 	L'objet bit associé prend la valeur inverse du résultat logique de la zone test
Elément d'action	• Bobine d'enclenchement 	L'objet bit associé est mis à 1 et garde cet état, lorsque le résultat de la zone test est à 1. Il est remis à 0 par la bobine de déclenchement.
Elément d'action	• Bobine de déclenchement 	L'objet bit associé est mis à 0 et garde cet état, lorsque le résultat de la zone test est à 1.

<p>Elément d'action</p>	<p>•Bobine saut à un autre réseau(JUMP) maximum42</p> <p style="text-align: center;">-(J)-</p>	<p>Permet un branchement à un Réseau étiqueté, amont ou aval. Les sauts ne sont effectifs qu'au sein d'une même entité de programmation (programme principal, traitement préliminaire...)</p> <p>L'exécution d'un saut provoque :</p> <ul style="list-style-type: none"> • l'arrêt de la scrutation du réseau en cours, • l'exécution du réseau étiqueté demandé, • la non scrutation de la partie du programme située entre l'action de saut et le réseau désigné. <p>L'instruction de saut n'est pas permise en tâche rapide.</p>
<p>Elément d'action</p>	<p>•Bobines sauvegardées</p> <p style="text-align: center;">-() -(/) -(S) -(R)</p>	<p>Bobines sauvegardées en cas coupure secteur. Les états des objets bits associés sont sauvegardés durant le première tour de cycle.</p>

b) Le traitement postérieur :

Saisi en langage à contacts, le traitement postérieur est scruté de haut en bas. Ce traitement est le dernier exécuté avant l'activation des sorties et permet de programmer la logique de sortie :

- Définition des actions associées ou non à des étapes.
- Gestion des sécurités inhérentes à ces actions (prise en compte d'un arrêt d'urgence, capteur de surcourse).
- Gestion des fonctions d'automatisme (temporisateur, compteur,..).



Fig(III-5) Le traitement postérieur

Actions associées au grafcet :

L'activité d'une étape du grafcet peut être testée à l'aide d'un contact dans le traitement postérieur afin de piloter les actions associées. Ceci permet d'assurer l'unicité de la commande (une seule bobine commandée par plusieurs étapes). De même le traitement postérieur permet de compléter les consignes émises par le traitement séquentiel en intégrant à l'équation d'une sortie les modes de marche et d'arrêt et les sécurités indirectes à l'action.

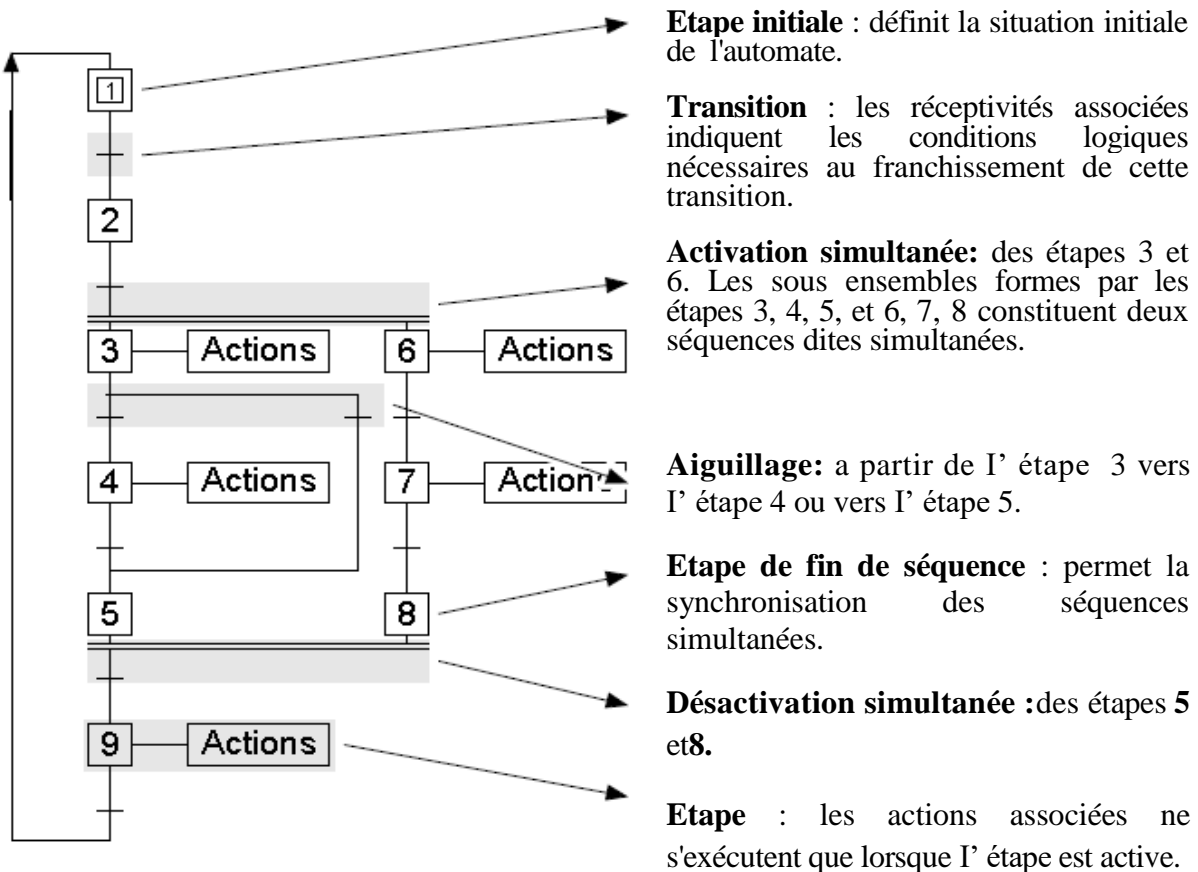
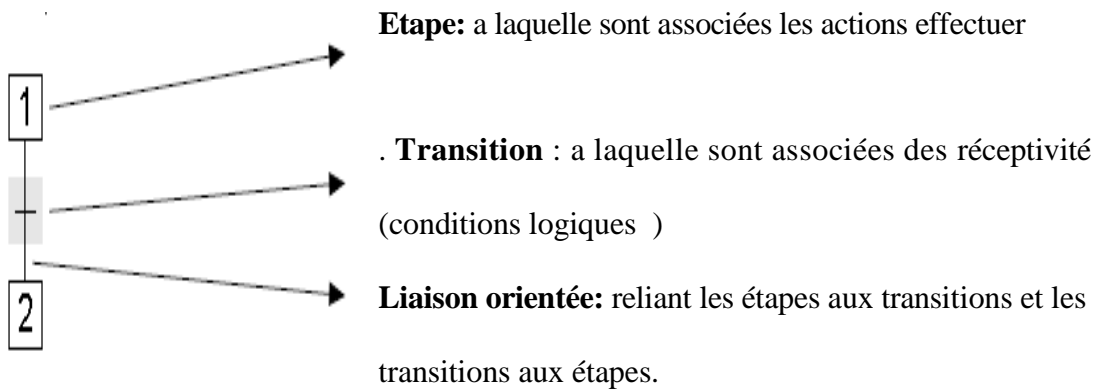
III-5.2 Langage a grafcet:

a) principes du grafcet :

Le grafcet permet de représenter graphiquement et de façon structure le fonctionnement d'un automatisme séquentiel.

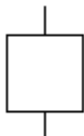
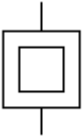

Cette description graphique du comportement séquentiel de l'automatisme, et des différentes situations qui en découlent, s'effectue à l'aide de symboles

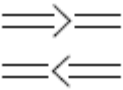
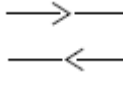



-graphiques simples :



Les transitions et les liaisons orientées symbolisent les possibilités dévolution des étapes actives.

Les actions associées aux étapes traduisent d'une façon générale "ce qui doit être fait" lorsque celles-ci sont actives. Elles décrivent en particulier les ordres qu'il faut transmettre à la partie opérative (processus à automatiser) ou à d'autres systèmes à automatiser.

Désignation	Graphisme	Nb.max	Commentaires
Étapes		96(*)	Symbolisent un état stable de l'automatisme. Bit étape associé : Xi. Mot temps enveloppe : Xi, V. Nombre maximum d'étapes actives simultanément : 16
Étapes		16	Symbolisent les étapes initiales actives en début de cycle après une initialisation ou une reprise à froid. Bit étape associé : Xi. Mot temps enveloppe : Xi, V.
Transitions		128	Séparent deux étapes et permettent le passage de l'une à l'autre. Nombre maximum de transitions Validées simultanément : 24

Activations et désactivation simultanées d'étape			Permettent l'activation et la désactivation simultanée de 4 étapes maximum
Aiguillage et fin d'aiguillage			Nombre maximum de transitions en aval et en amont d'un aiguillage : 4
aval et en amont d'un aiguillage Renvois destination		42	Permettent d'assurer la continuité de la lecture du graphe
Passage d'une étape à une autre (réceptivité)		128	Symbolise le passage d'une étape à une autre, apparaît automatiquement lors de la saisie d'une réceptivité
Liaisons orientées			Permettent de relier une étape à une transition ou une transition à une étape.

- Les objets spécifiques au grafcet :

L'utilisateur dispose d'objets bits associés aux étapes, de bits systèmes spécifiques au langage grafcet et d'objets mots indiquant le temps d'activité des étapes.

Désignation	Adresse	Adresse
Bits associés aux étapes	Xi	Etape i du grafcet principal 0 <= i <= 95
Bits système associés au grafcet (1)	SY21	Provoque l'initialisation du grafcet

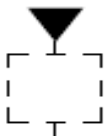
	SY22	Provoque la remise à zéro générale du grafcet
	SY23	Validation du pré positionnement et gel des graphes
Mots temps d'activité des étapes	Xi,V	Temps d'activité de l'étape i $0 \leq i \leq 95$ $0 \leq V \leq 9999$ secondes

- Utilisation des renvois :

On utilise deux types de renvois :



• renvoi de destination : le numéro qui lui est associé indique l'étape de destination "où l'on va".



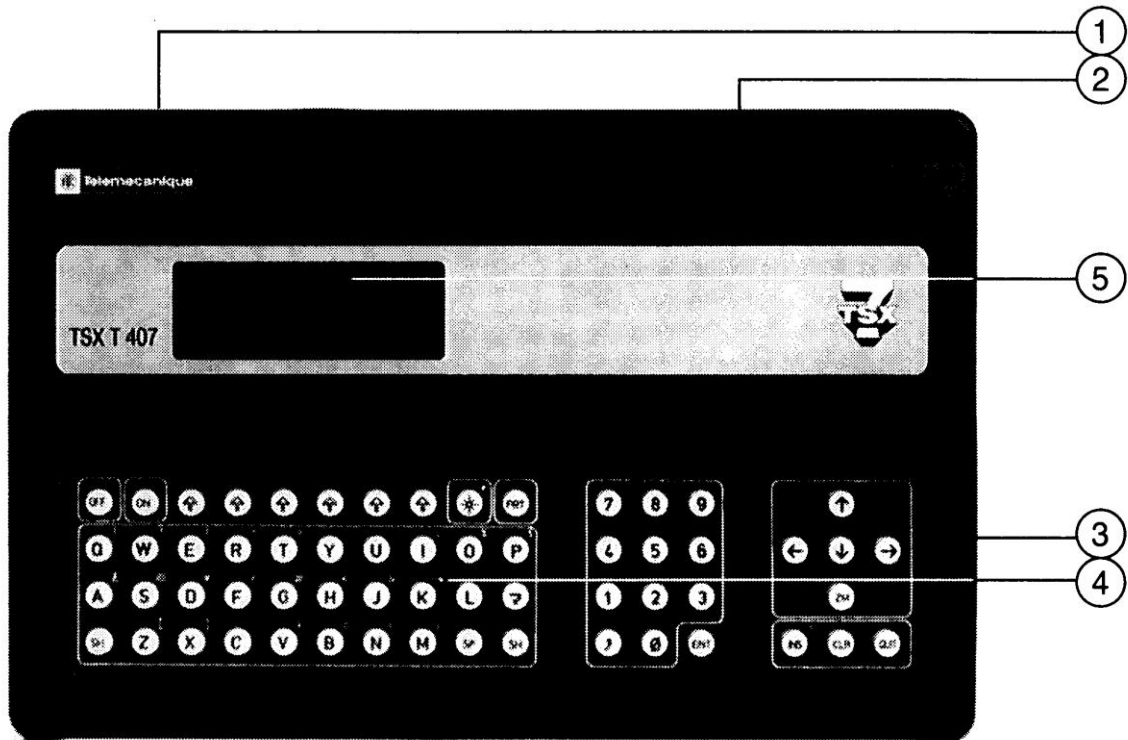
• renvoi d'origine : le numéro qui lui est associé indique l'étape d'origine "d'où l'on vient"

Toute liaison transition ® étape peut être interrompue dans une page ou dans deux pages différentes. Le nombre maximum de renvois est de 42 renvois d'origine et de 42 renvois de destination.

III-6.LE TERMINALE DE PROGRAMMATIONTSX T407:

Le terminal TSX T407, par la cartouche programme dont il est équipé, est l'outil de programmation en PL7-2 de l'automate, TSX 27.

Il permet d'effectuer aisément une programmation "graphique" à partir d'études réalisées en langage à contacts ou en grafcet. Il sert également comme terminal de réglage et de mise au point pour cet automate .programmés en langage PL7-2



Fig(III-6) - Le terminal de programmation TSX T407

- ① - Câble de connexion et d'alimentation.
- ② - Connecteur périphérique (imprimante, modem, magnétophone.....).
- ③ - Cartouche langage.
- ④ - Clavier.
- ⑤ - Ecran de visualisation.

TSX T407 1 qui fonctionne comme la version zéro en mode connecté, et autonome ; il est alimenté par le boîtier d'alimentation TSX TA4 2.

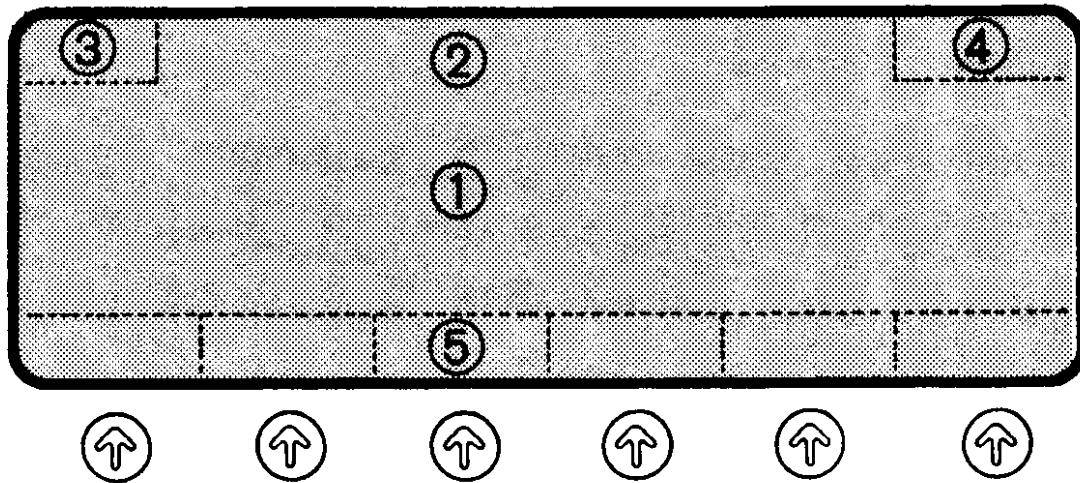
La mémoire de travail est:

- *Soit la mémoire utilisateur de l'automate (mode connecté),
- *Soit la mémoire interne du terminal (mode autonome).

III- 6.1.Ecran de visualisation :

- Caractéristiques :

L'écran du terminal TSX T407 est réalisé par un afficheur à cristaux liquides constitué de 6 lignes de 24 caractères.

-Informations visualisées :

Fig(III-7) Ecran de visualisation

- ① -Zone programme: composée de 4 lignes, elle visualise un réseau de contacts ou une fenêtre du graphe.
- ② - Bandeau : il permet de saisir les commentaires ou les adresses des contacts, bobines et blocs fonction.
- ③ - Zone événements: en fonctionnement normal elle peut indiquer les informations suivantes:

! Défaut automate (mémoire, E/S) ou bit système SY8 mis à 1

!B défaut de la batterie de sauvegarde de la mémoire RAM du terminal.

!F présence d'un ou plusieurs bit forces.

A la mise sous tension cette zone affiche le résultat de l'auto-test du terminal.

- ④ -Zone état automate: elle visualise le mode de marche de l'automate ou le type de point d'arrêt utilisé pour la mise au point du programme:

RUN l'automate exécute le programme mémorisé dans sa mémoire utilisateur.

STOP l'automate est à l'arrêt.

WAIT le chien de garde surveillant l'exécution du programme, signale un défaut. Il faut initialiser la cartouche utilisateur [CLM], ou modifier le programme.

INT le terminal TSX T407 1 fonctionne en mode autonome.

DCN le terminal n'est pas connecté à un automate série 7.

S/CY l'exécution du programme s'arrête au prochain défaut de cycle , après mise à jour des entrées /sorties.

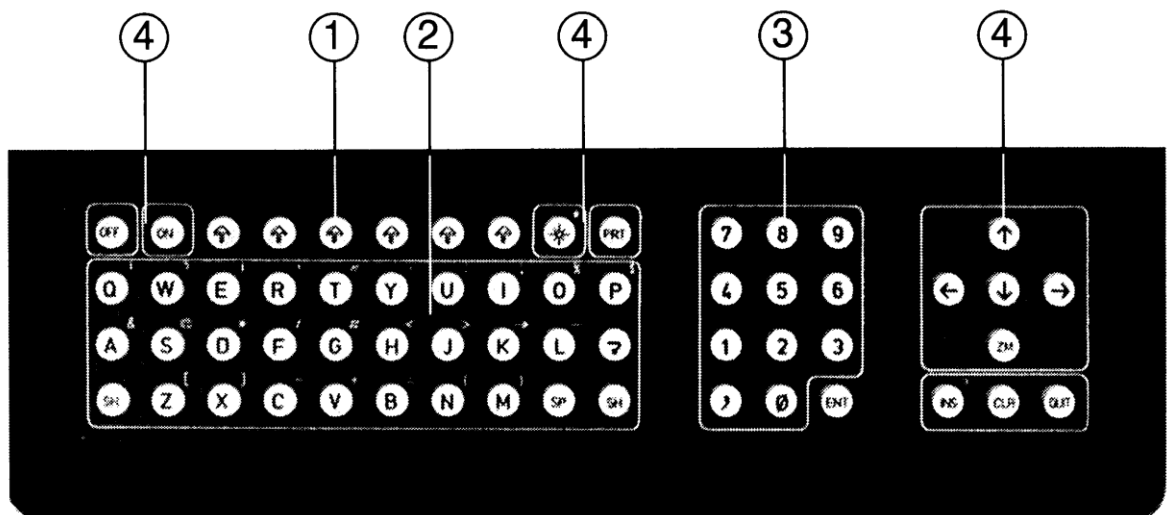
- ⑤ -Zone d'affichage des touches dynamiques : elle indique la fonction des touches dynamiques.

III-6.2 Clavier:

- Caractéristiques :

Le clavier est intégré au terminal. Il comprend 60 touches regroupées par fonctions. Il permet de saisir du programme, des données en mémoire ou de dialoguer avec l'automate connecté. Un contrôle permanent de l'utilisation des touches évite toute erreur de manipulation

- Organisation de clavier :



Fig(III-8) Clavier

- ① - 6 touches dynamiques : la fonction de ces touches gérée dynamiquement par le terminal, varie selon le contexte d'utilisation. Ces spécifiques du langage (contacts, bobines, blocs fonctions, étapes Grafcet, transitions)

La fonction des touches dynamiques est visualisée sur la ligne inférieure de l'écran.

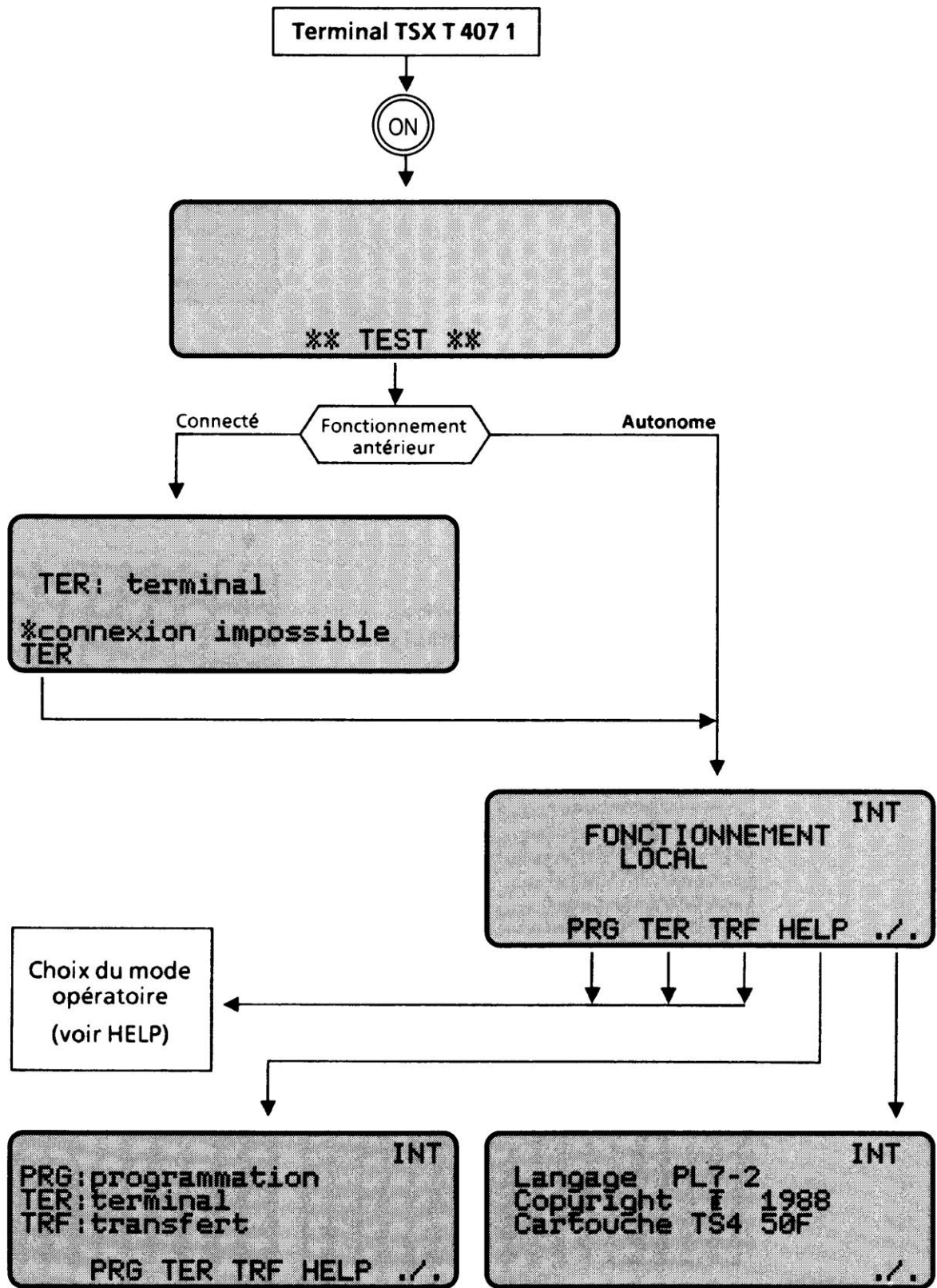
- ② - Clavier alphabétique : ce clavier "QWERTY" permet de saisir en majuscule toutes les lettres de l'alphabet pour écrire programme. Données ou commentaires. Utilisation simultanée d'une touche avec <SH> permet de saisir la deuxième fonction de cette touche.
- ③ - Clavier numérique : il permet de saisir tous les chiffres (0 à 9) et la virgule.
- ④ - Touches fonction : elle permettent la mise sous tension du terminal et les fonction déplacement du curseur , zoom , insertion ou suppression d'un réseau de contacts , impression ,etc

III-6.3 Mise sous tension :

a - Accès au menu sélection des modes:

Si le mode fonctionnement choisi (autonome ou connecté) reste inchangé par rapport au fonctionnement antérieur, l'accès au menu sélection des modes est direct paré l'auto-test du terminal.

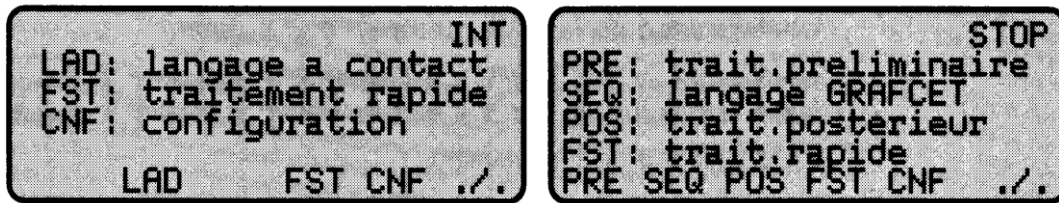
Dans le cas contraire se reporter aux étapes données par la figure suivante :



Fig(III-9) Utilisation du terminal en fonctionnement autonome

III-7.CREATION DE PROGRAMME:

Lors de la programmation il apparait les deux écrans suivants :



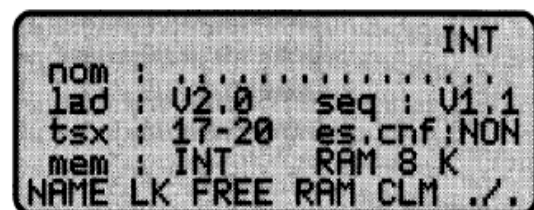
Fig(III-10) Création de programme

- [LAD]** Accède à la programmation en langage à contacts.
- [PRE]** Accède à la programmation du traitement préliminaire, en langage à contacts.
- [SEQ]** Accède à la programmation séquentiel.
- [POS]** Accède à la programmation en postérieur.
- [FST]** Accède à la programmation de la tâche rapide.
- [CNF]** Accède à la fonction configuration de l'automate.
- [./.]** Visualise l'écran suivant.

[NAME] Nommer le programme.

[LK] Diagnostic et retassage.

[FREE]



[RAM] Accède à la fonction configuration de la mémoire.

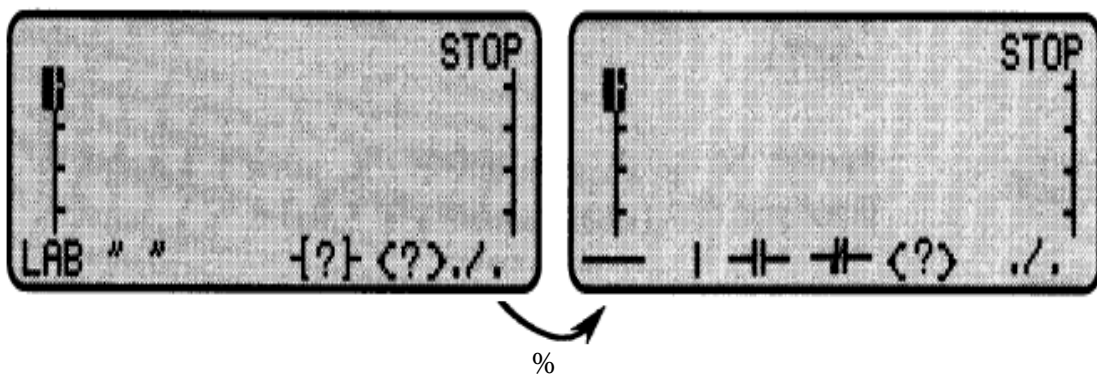
[CLM] Accède à la fonction initialisation de la mémoire (effacement total de la mémoire programme et des paramètres de configuration).

En plus des touches dynamiques, l'écran ci-dessus renseigne l'utilisateur sur l'automate et sur les langages utilisés.








III-7.1. Langage contacts :

Toute création de symbole graphique nécessite que le curseur soit visualisé sur l'écran. L'élément graphique est saisi par la touche dynamique correspondante à l'endroit pointé par le curseur et son adresse est saisie dans le bandeau. Cet ensemble étant validé par < ENT >, la saisie d'un nouveau symbole set alors possible.


-Saisie d'un réseau de contacts :





Fig(III-11) réseau de contacts


-  Permet de saisir le numéro d'étiquette du nouveau réseau de contacts, valider par < ENT >.
-  Visualise les touches dynamiques des blocs fonctions/opérations.
-  Visualise les touches dynamiques des bobines de sorties.
-  Saisit une connexion horizontale.
-  Saisit une connexion verticale. Si une connexion verticale est pointée par le curseur, elle la supprime.
-  Permet de saisir un contact à fermeture (symbole graphique et adresse). Valider par < ENT >.
-  Permet de saisir un contact à ouverture (symbole graphique et adresse). Valider par < ENT >.


-Saisie d'une bobine de sortie :


- 

Permet de saisir une bobine directe (symbole graphique et adresse). Valider par < ENT >.
- 

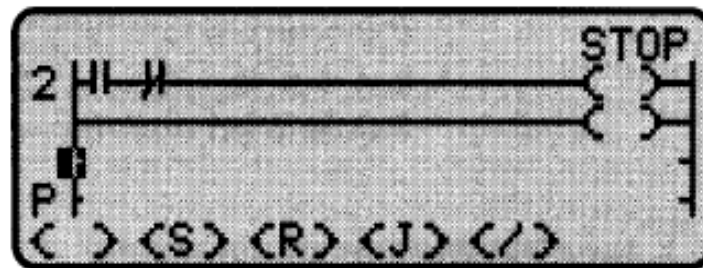
Permet de saisir une bobine directe d'enclenchement SET (symbole graphique et adresse). Valider par « ENT ».
- 

Permet de saisir une bobine de déclenchement RESET (symbole graphique et adresse). Valider par < ENT >.
- 

Permet de saisir une bobine de saute (JUMP) (symbole graphique et étiquette du réseau à scruter dès que cette bobine est active). Valider par < ENT >.
- 

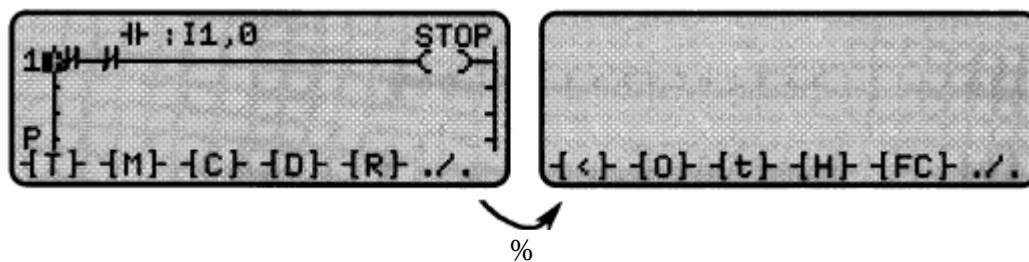
Permet de saisir une bobine inverse (symbole graphique et adresse). Valider par < ENT >.
- 

Visualise à nouveau les touches dynamiques connexions et contacts.



Fig(III-12) Saisie d'une bobine de sortie

- Saisie d'un bloc fonction/opération :

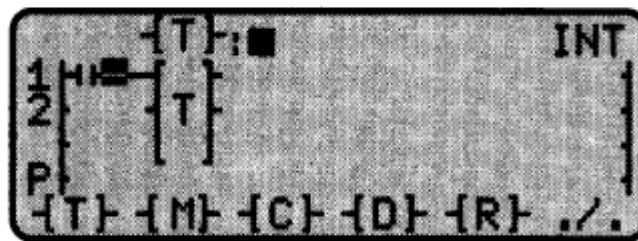


Fig(III-13) Saisie d'un bloc fonction/opération

- {T} Bloc fonction temporisation.
- {M} Bloc fonction monostable
- {C} Bloc fonction compteur.
- {R} Bloc fonction registre
- {D} Bloc fonction programmateur cyclique
- {<} Bloc opération comparaison.
- {o} Bloc opération calcul et transfert.
- {t} Bloc fonction texte.

*Bloc fonction temporisateur :

- {T} Permet de saisir un temporisateur :
-Symbole graphique et numéro :
T0 à T15 (TSX 27)



Fig(III-14) Bloc fonction temporisateur



Permet l'accès au deuxième niveau de visualisation (écran suivant).

- [TB] Par appuis successifs, modifie la base de temps :
1mn, 1s, 100ms, 10ms (1mn par défaut).
- [PRE] Permet de modifier la valeur de
présélection(PRESET) :0 à 9999(9999 par défaut).
- [MOD] Autorise ou interdit la modification des paramètres en
REGLAGE (YES par défaut).

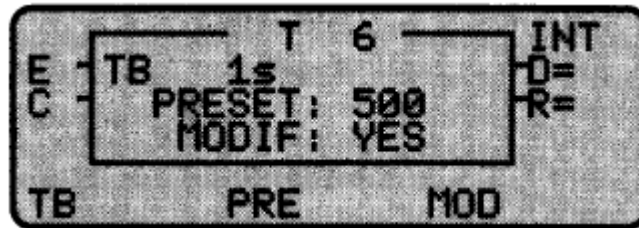
❖ Entrées/sorties :

E : Armement (sur front descendant initialise le temporisateur).

C : contrôle (à l'état 1 valide l'écoulement du temps).

D : temporisateur écoulée.

R : temporisateur en cours.



Fig(III-15) Les entrées/sorties

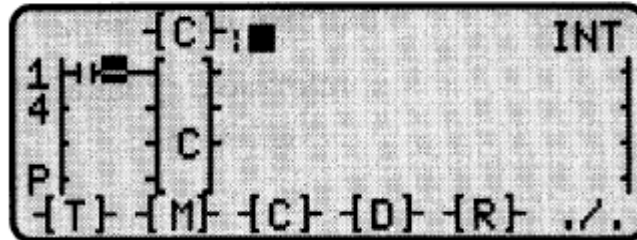
*Bloc fonction compteur :



Perme de saisir un compteur :

-Symbole graphique et numéro :

C0 à C15 (TSX 27)



Fig(III-16) Bloc fonction compteur



Permet l'accès au deuxième niveau de visualisation (écran ci-dessous).

[PRE] Permet de modification la valeur de Présélection (PRESET) :0 à 9999 (9999 par défaut).

[MOD] Autorise ou interdit la modification des Paramètres en REGLAGE (YES par défaut).

❖ Entrées/sorties :

R : R.A.Z (à l'état 1 met la valeur courante à 0).

P : Présélection (à l'état 1 initialise le compteur).

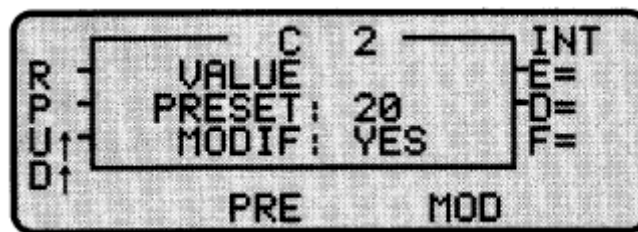
U↑ : Comptage (sur front montant incrémente la valeur courante).

D↑ : Décomptage (sur front montant décrémente la valeur courante).

E : Débordement décomptage.

D : Présélection attente.

F : Débordement comptage.



Fig(III-17) Les entrées/sorties de compteur

III-7.1.langage Grafcet :

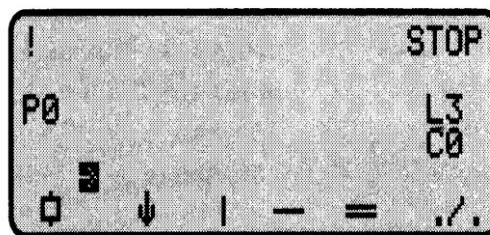
Le choix du langage de programmation s'effectue lors de l'initialisation de la mémoire utilisateur : PRG –SEQ – CLM – SEQ.

- Saisie d'un graphe :



Permet de saisir une étape

(le curseur est position sur une ligne impaire). Valider par <ENT>



Fig(III-18) Saisie d'un graphe

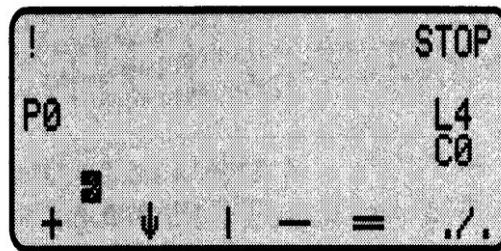


Saisit une transition (le curseur est positionné une ligne paire) .



Permet de saisir un renvoi de destination ou d'origine.

Valider par <ENT>.



Fig(III-19) Saisie d'un graphe

Saisit une liaison orientée vers le bas.



Permet de saisir un aiguillage ou une fin d'aiguillage. Cette touche visualise les 2 touches suivantes permettant de choisir sens de représentation de l'élément graphique.



Saisit un aiguillage vers la droite .



Saisit un aiguillage vers la gauche.



Permet de saisir une activation ou une désactivation simultanée d'étapes . Cette touche visualise les 2 touches suivantes permettent de choisir le sens de représentation de l'élément graphique .



Saisit une activation simultanée d'étapes vers la droite



Saisit une activation simultanée d'étapes vers la gauche.



Visualise la suite des touches dynamiques.

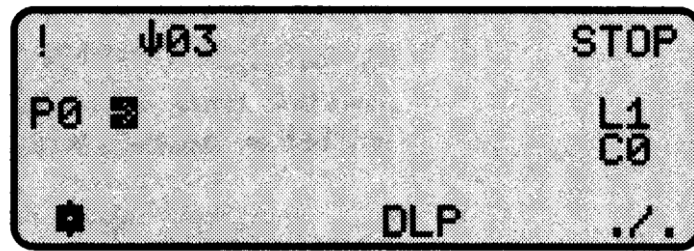


Saisit une étape initiale.



Delete Page : supprime le contenu de la page





Fig(III-20) Saisit une étape initiale et delete Page

A chaque étape est associé un numéro: 0 à 95.

A chaque renvoi de destination est associé le numéro de l'étape de destination: 0 à 95.

A chaque renvoi d'origine est associé le numéro de l'étape d'origine 0 à 95.

-Validation d'un graphe :



Valide toute page contenant un ou plusieurs graphes. Le terminal contrôle l'exécutabilité du ou des graphes saisis et signal le éventuelles.

-Validation d'un graphe saisi sur plusieurs pages :



Si un graphe occupe plusieurs pages il ne pourra être validé qu'en fin de saisie. Il sera alors possible de valider les différentes pages dans l'ordre inverse de leur saisie . Le terminal se positionne automatiquement sur les pages restant à valider et visualise le message "reste pages non valides".

-Saisie d'une action:

La présentation s'effectue en mode (POS).

-Saisie d'une réceptivité :



Le curseur étant positionné à gauche d'une transition . cette touche permet de saisir le réseau de contacts associé à cette transition .



La bobine située dans la zone action du réseau, symbolisant le passage d'une étape à l'autre, est saisie automatiquement .

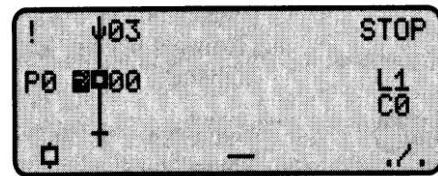
Il est toutefois nécessaire de saisir la zone test . Une réceptivité non saisie n'est jamais franchie.

La saisie d'une réceptivité peut se faire après la saisie de la transition ou en fin de saisie du graphe.


-Suppression d'un élément et de son contenu :

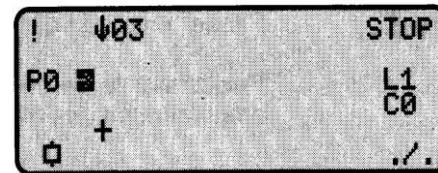
Positionner le curseur à gauche de l'élément (étape, transition, renvoi) à effacer.

-  En appuyant simultanément sur touches supprimé l'élément graphique.
-  Les action ou la réceptivité associées cet élément sont supprimées.



Saisir le nouvel élément à l'emplacement libéré.




-  Valider à totalité du graphe

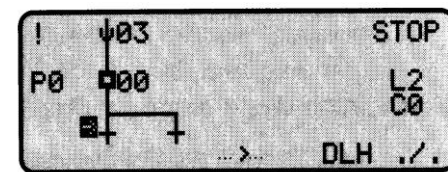
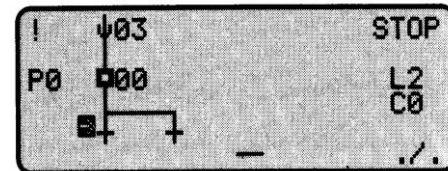


Fig(III-21) Suppression d'un élément et de son contenu

-Suppression d'un élément d'aiguillage ou de fin d'aiguillage :

Positionner le curseur en dessous (à gauche ou à droite) de la liaison à supprimer.

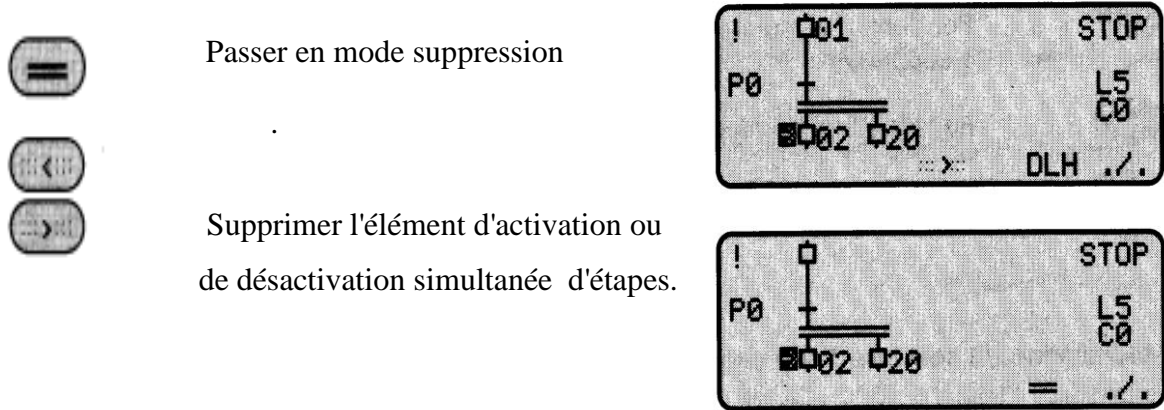
-  Passer en mode suppression
-  Supprimer l'aiguillage ou la fin d'aiguillage
- 



Fig(III-22) Suppression d'un élément d'aiguillage ou de fin d'aiguillage

-Suppression de le totalité d'une liaison (aiguillage ou activation simultanée d'étapes) :

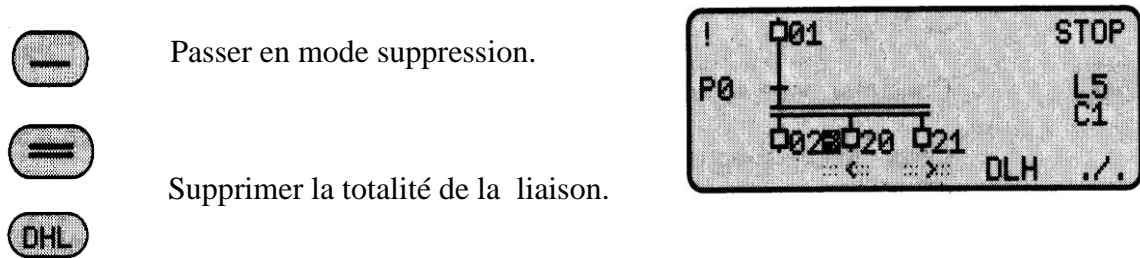
Positionner le curseur en dessous (à gauche ou à droite) de l'élément à supprimer.



Fig(III-23) Suppression de le totalité d'une liaison

-Suppression de le totalité d'une liaison (aiguillage ou activation simultanée d'étapes) :

Positionner le curseur en dessous (à gauche ou à droite) de l'élément à supprimer.



Fig(III-24) Suppression de le totalité d'une liaison

INTRODUCTION :

L'eau recouvre 70% de la superficie du globe, mais malheureusement 97 % de cette eau est salée et non potable et ne convient pas à l'irrigation. L'eau douce, elle représente 3% de l'eau totale de notre planète. Dans ce faible pourcentage, les rivières et les lacs représentent 0,3%, alors que tout le reste est stocké dans les calottes polaires glacières.

Un des facteurs majeurs qui gouvernent le développement de sociétés humaines est la préoccupation d'obtenir et de maintenir une provision adéquate d'eau. Le fait de disposer d'une quantité d'eau suffisante a dominé les premières phases de développement. Cependant, les augmentations des populations ont poussé à puiser de façon intensive dans les sources en surface de bonne qualité mais qui sont en quantité limitée ou les ont contaminées ou ont laissé perdurer des gaspillages humains qui ont amené à détériorer la qualité de l'eau. La qualité de l'eau ne pourra plus être oubliée dans le processus de développement. La conséquence inévitable de l'augmentation de la population et du développement économique est le besoin de concevoir des installations de traitement de l'eau pour fournir une eau de qualité acceptable issue de sources en surface contaminées.

La production d'eau potable peut être définie comme la manipulation d'une source d'eau pour obtenir une qualité de l'eau qui satisfait à des buts spécifiés ou des normes érigées par la communauté au travers de ses agences régulatrices.

L'eau est le composé le plus abondant sur la surface du globe. Sans elle, la vie comme nous le savons cesserait d'exister. Pour l'ingénieur de l'eau, la microbiologie est importante pour ses effets sur la santé publique, sur la qualité de l'eau (propriétés physiques et chimiques), et sur la bonne marche de l'unité de traitement. Les micro-organismes flottants peuvent être responsables de problèmes de santé publique divers qui incluent des maladies bactériennes telles que le choléra et la gastro-entérite, des infections virales telles que l'hépatite.

La commande et la surveillance des installations de production d'eau potable deviennent de plus en plus importantes et ce quel que soit l'endroit dans le monde.

IV- 1.DESCRPTION DE L'UNITÉ DE FILTRAGE :

IV-1.1.Présentation de l'UCM :

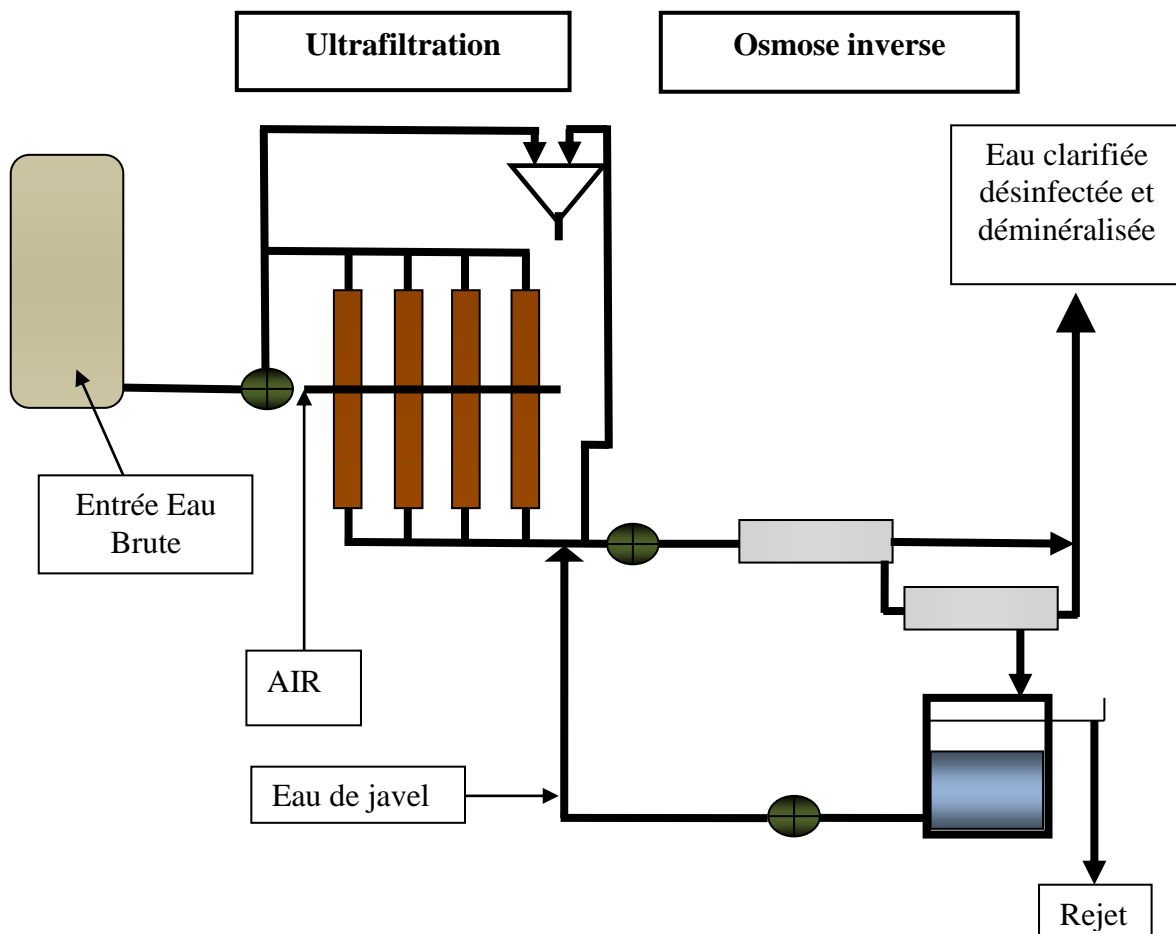
L'unité compacte Membranaire (UCM) est un équipement original de purification des eaux , développé par technique membranaires :l'ultrafiltration et l'osmose inverse .

L'association innovante de ces deux procédés sur un seul équipement permet d'obtenir une purification à la fois bactériologique et virale ainsi qu'une purification minérale de l'eau (déminéralisation totale ou partielle).

Les caractéristiques principales de la gamme UCM sont :

- un débit d'eau traitée allant de 3.5 à 42m³/h (pour une température de 15°C),
- l'utilisation de modules d'Ultrafiltration en polysulfone, résistants au chlore,
- l'utilisation de membranes d'Osmose Inverse en polyamide, basse énergie.

-Schéma de principe:



Fig(IV-1)Production d'une eau potable

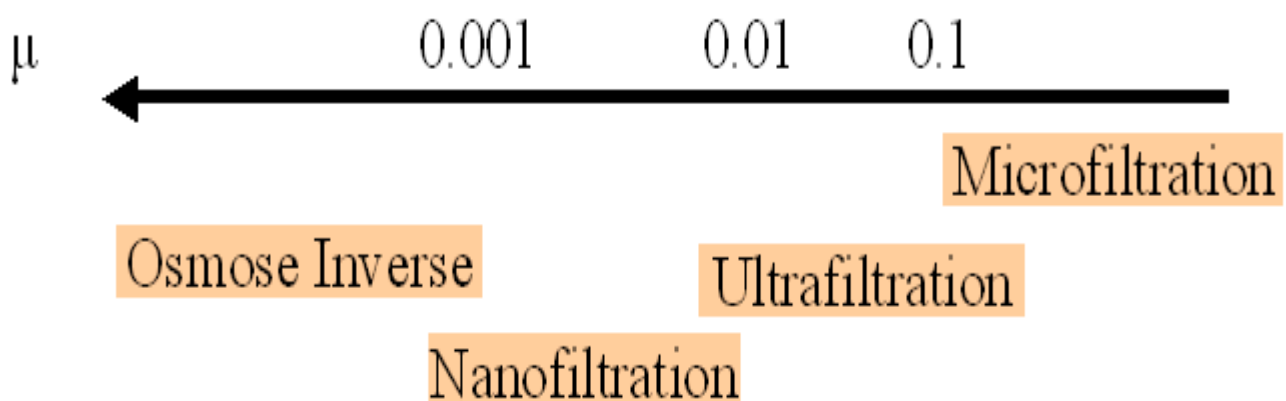
V-1.2. Les techniques membranaires utilisées :

Les techniques de filtration membranaires sont caractérisées par :

- une filtration mécanique au travers d'un média plus ou moins poreux permettant de retenir des particules de tailles plus ou moins importantes .
- des conceptions simples et universelles reposant sur des lois éprouvées.

Les membranes se classent selon leur porosité:

- La Microfiltration utilise des pores relativement grands (0.1 μm) qui laissent passer certaines impuretés et micro-organismes (virus).
- L'Ultrafiltration utilise des pores plus fins (0.01 μm) . Cette finesse des pores répond à la fois au besoin de clarification et désinfection de l'eau (les bactéries et les virus sont retenus) et aussi à l'abattement des particules non ioniques , particules dites colloïdales (argiles, silice) présentant un haut pouvoir colmatant.
- La Nanofiltration utilise des pores plus étroits (0.001 μm) qui retiennent certains sels minéraux contenus dans l'eau . En général, il s'agit d'ions bivalent tels que le calcium, magnésium , sulfate . La technique de nanofiltration est utilisée en désulfatation de l'eau et est efficace dans la lutte contre les pesticides.
- L'Osmose inverse utilise des pores très petits, d'une taille de 0.0001 μm qui retiennent la majorité des minéraux présents dans l'eau et permettent donc d'obtenir une eau déminéralisée.



Fig(IV-2) Production d'une eau potable

IV-1.3. Utilité de l'UCM :

Dans le monde industriel, l'utilisation des ressources en eau et tout particulièrement des ressources naturelles (eau de forage, de canal ou de rivière) est aujourd'hui une priorité permanente.

Une des originalités de l'UCM réside dans sa capacité à exploiter pratiquement tous les types d'eau disponibles, y compris celles qui n'étaient pas utilisées dans le passé compte tenu de la difficulté à les traiter.

En effet, l'étage d'Ultrafiltration permet de retenir toutes les particules incompatibles avec le traitement d'Osмосe Inverse et de réduire:

- la charge bactériologique avant l'Osмосe Inverse.
- tous les paramètres non ionisés, tels que la silice colloïdale.
- l'indice de colmatage de l'eau.

L'étage d'Osмосe Inverse permet, quant à lui, la déminéralisation de l'eau à destination de nombreux procédés industriels :

- circuits de refroidissement,
- procédés d'évaporation,
- chaudières de production de vapeur.

L'eau produit par sa très faible salinité , autorise des rapports de concentration élevés et des taux de purge optimum réduisant ainsi les rejets vers la station d'épuration.

En résumé:

La principale application de l'UCM est la déminéralisation des eaux de surface sur un seul et même équipement , sans l'utilisation notable de produits chimiques dangereux .

L'UCM permet également de produire , dans la quasi-totalité des cas , une eau de qualité potable , clarifiée , désinfectée et dont la salinité peut être ajustée à la valeur voulue

IV -1.4.Exemple d'application pratique de L'UCM :**• Production d'une eau potable :**

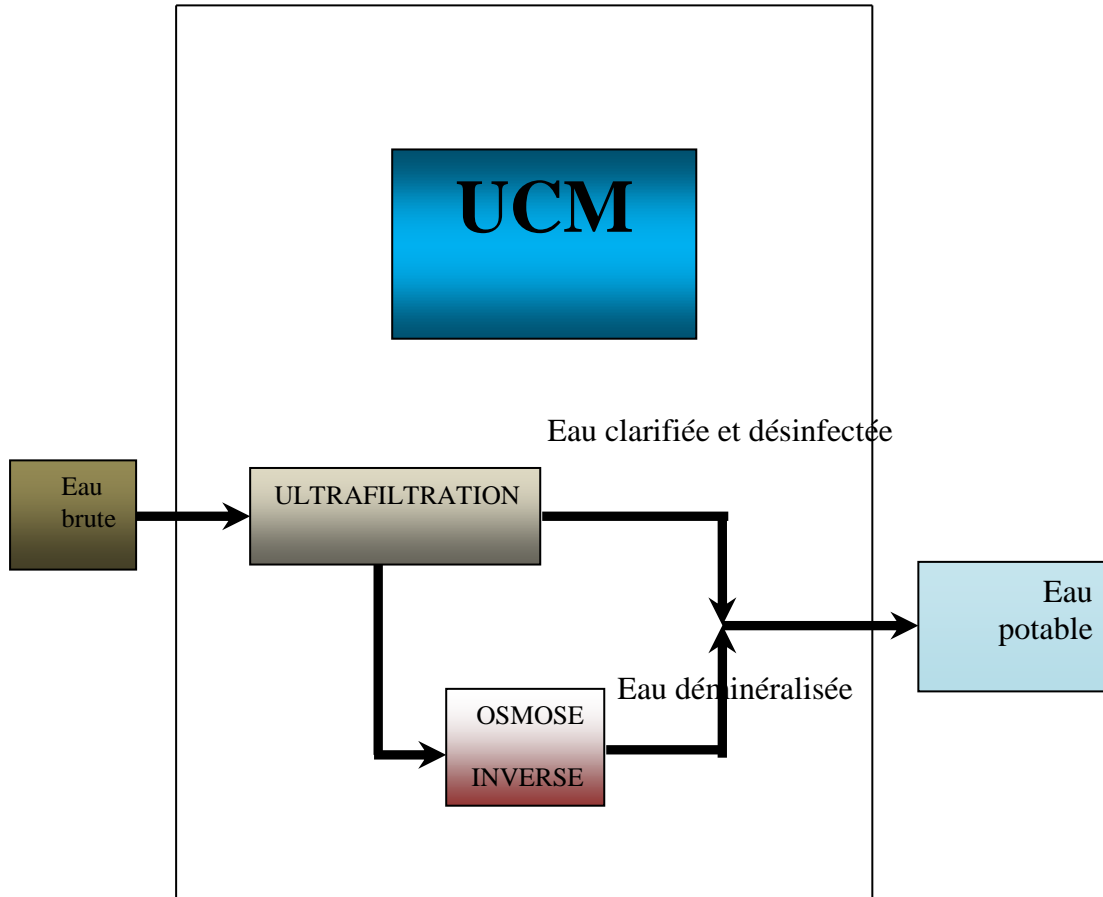
A partir d'une ressource généralement délaissée parce que :

- Trop polluée d'un point de vue bactériologique.
- Trop polluée par sa teneur en matières en suspension (MES).
- Inadaptée par une trop forte salinité.

L'UCM est la technologie la plus performante, la plus simple d'utilisation et la plus économique pour obtenir une eau clarifiée, désinfectée et dont la minéralisation est ajustée.

Le traitement s'établit comme suit :

- Abattement de la charge bactériologique sur l'étage d'ultrafiltration de tête.
- Déminéralisation complète de l'eau ultrafiltrée.
- Mélange de l'eau déminéralisée avec une proportion d'eau pour ajuster la salinité de l'eau à une valeur donnée, sans apport de produits chimiques dangereux



Fig(IV-3) Production d'une eau potable

Fig(IV-4) Production d'une eau potable et l'eau déminéralisée.

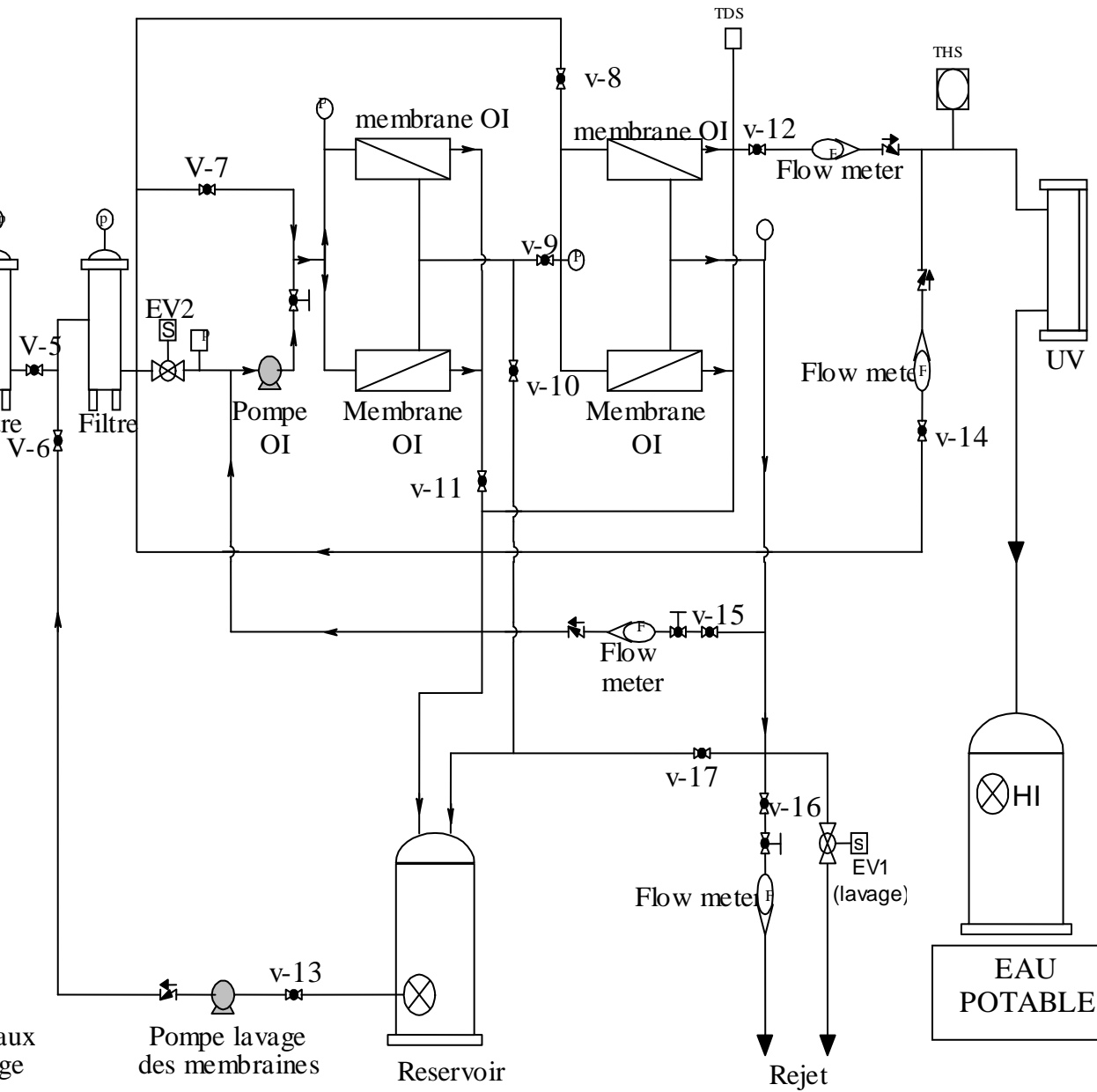


schéma fonctionnel de la station

IV-2.EQUIPEMENTS ELECTRIQUES:

L'unité compacte se compose de deux parties formant le système de production, l'une hydraulique, l'autre électrique.

La partie hydraulique représente la majorité du matériels, telque les canalisations les vannes à commande manuel et les filtres.

Alors que la partie électrique est basée sur l'utilisation d'une armoire de commande équipée d'un automate, d'un bloc de contrôle IGBT de la pompe osmose inverse, pompes d'amenés, les électrovannes et une pompe de forage équipée d'un démarreur progressif.

IV-2.1. Démarreur progressif :



Fig(IV-6) Démarreur progressif

- Le démarreur progressif **LH4 N1** est utilisé pour les moteurs monophasés et triphasés à chaque fois que les à-coups dus aux démarrages doivent être supprimés. Son utilisation est limitée aux moteurs de petite puissance.

Le démarreur-ralentisseur progressif **LH4 N2** permet le contrôle du couple de démarrage, une réduction importante du courant de démarrage et le ralentissement en douceur.

D'une façon générale :

Les démarreurs électroniques **LH4** permettent :

- un ajustement du couple de démarrage.
- la suppression des chocs mécaniques (causes d'usure).
- la réduction des temps d'arrêt de la production.

- Les démarreurs-ralentisseurs progressifs **LH4 N2** sont utilisés pour les ventilateurs,

pompes, compresseurs et toutes les machines à fortes inerties. Sur les machines, où l'isolement galvanique n'est pas nécessaire, ils évitent l'utilisation du contacteur de ligne.

- **Fonctionnement :**

Le démarreur progressif LH4 délivre au démarrage une tension réduite et la montée progressive de celle-ci se fait jusqu'à sa valeur nominale. Ceci a pour conséquence de réduire les à-coups de couple préjudiciables aux moteurs, à la mécanique entraînée, ainsi que les courants de démarrage dans le cas du LH4 N2.

Pour le LH4 N2, un relais signalant les défauts du produit et permettant la commande du contacteur de ligne (isolement), délivre une information lorsque le produit fonctionne. Cette information peut être utilisée pour piloter le contacteur de ligne.

Réglages et mise en service :

Sur tous les démarreurs et démarreurs-ralentisseurs, il y a 2 potentiomètres qui permettent :

- ✓ de régler le temps d'accélération
- ✓ d'obtenir le couple de décollage nécessaire pour démarrer immédiatement après l'ordre de marche.

Sur les LH4 N2, un troisième potentiomètre permet le réglage du temps de décélération.

Ces fonctions ajustables peuvent être plombées pour éviter tout risque de dérèglement.

4-2.2.Bloc de contrôle multifonction IGBT :

Vu l'impotence de la pompe osmose inverse, celui-ci est commandée par un bloc de contrôle qui surveille, la courant, la tension, la vitesse, la fréquence.....etc.

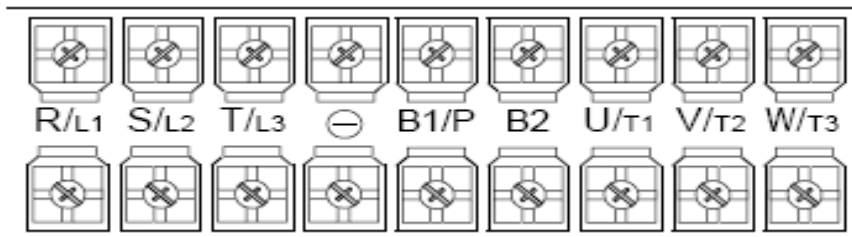
Le IGBT peut être programmé par le console de programmation intégré sur le module ou peut être placé sur l'armoire de contrôle speecom 7200 MA.

* Bornes de connexions du 7200MA

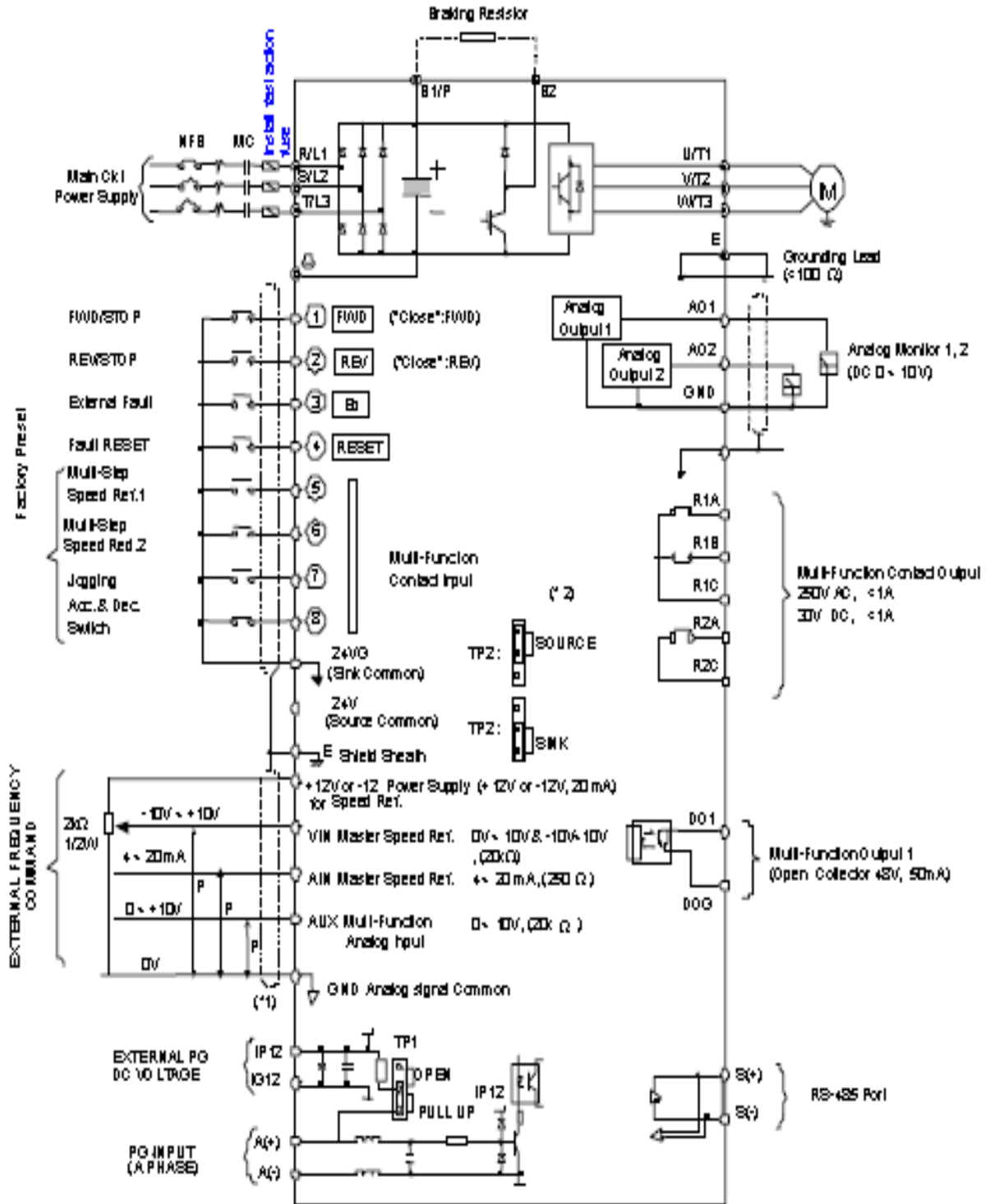
Terminal	220V :1~20HP, 440V :1~20MP	220V :25~40HP, 440V :25~75MP
R/L1	Main circuit input power supply (For single phase power supply, please use R/L1, S/L2 as input terminal)	
S/L2		
T/L3		
B1/P	B1/P, B2 :External braking resistor	-
⊖ B2	B1/P, ⊖ : DC power supply input • ⊕ -⊕:DC power supply or braking unit	
⊕		
B1/R	Unused	
U/T1	Output	
V/T2		
W/T3		
E	Grounding lead (3rd type grounding)	

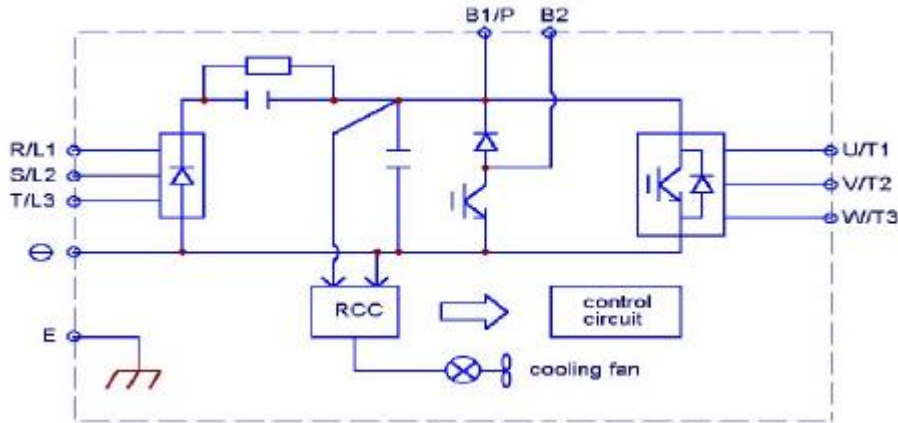
*Configuration du bloc de contrôle :

- 220V/440V : 15~20HP



* Schéma de connexion :

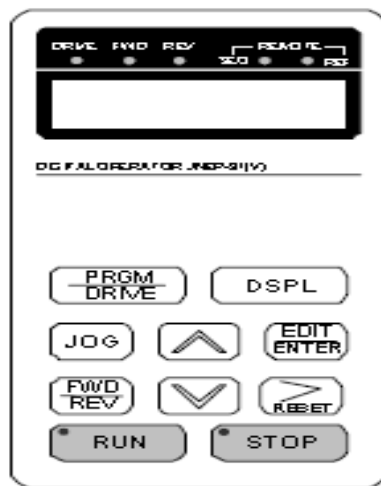




* CONSOLE DE PROGRAMMATION (JNEP-31(V))

Le consol de programmation possède deux modes de fonctionnement(commande – programmation).

Ces deux modes peuvent être obtenu par la touche :



Console de programmation

* Exemple d'utilisation :

Description	Key Sequence	Digital Operator Display	Remark
<p>(1) When Power on</p> <ul style="list-style-type: none"> Select frequency reference value displayed Select PRGM mode 	<p>press 3 times</p> <p>PRGM DRIVE</p> <p>DSPL</p> <p>EDIT ENTER</p> <p>RESET</p> <p>EDIT ENTER</p>	<p>Freq. Cmd.000.00Hz</p> <p>TECO</p>	<p>LED <input type="checkbox"/> DRIVE</p> <p>OFF</p>
<p>(2) Input voltage setting (e.g. AC input voltage is 380V)</p> <ul style="list-style-type: none"> Select CONTROL PARAMETER Display Cn-01 setting Input Voltage 380V <p>(continued)</p>		<p>Cn -01-</p> <p>Input Voltage</p> <p>Cn-01 = 440.0V</p> <p>Input Voltage</p> <p>Cn-01 = 380.0V</p> <p>Input Voltage</p> <p>Entry Accepted</p>	

Description	Key Sequence	Digital Operator Display	Remark
(continued)			
(3) FWD JOG	<ul style="list-style-type: none"> PRGM DRIVE DSPL JOG 	<p>Freq. Cmd. 0.00 Hz TECO</p> <p>O/P Freq. 6.00 Hz Freq. Cmd. 6.00 Hz</p>	<p>LED ON DRIVE</p> <p>LED ON FWD</p>
(4) Frequency setting	<ul style="list-style-type: none"> DSPL (press 4 times) RESET ↑ ↓ EDIT ENTER 	<p>Freq. Cmd. 0.00 Hz TECO</p> <p>Freq. Cmd. 015.00 Hz TECO</p> <p>Freq. Cmd. 015.00 Hz TECO</p> <p>Entry Accepted</p>	<p>Displayed for 0.5sec Confirm the display.</p>
(5) FWD run	<ul style="list-style-type: none"> DSPL * RUN 	<p>O/P Freq. 0.00 Hz Freq. Cmd. 15.00 Hz</p> <p>O/P Freq. 15.00 Hz Freq. Cmd. 15.00 Hz</p>	<p>LED ON * RUN</p>
(6) Frequency command change	<ul style="list-style-type: none"> DSPL (press 4 times) RESET ↑ ↓ EDIT ENTER 	<p>Freq. Cmd. 015.00 Hz TECO</p> <p>Freq. Cmd. 060.00 Hz TECO</p> <p>Freq. Cmd. 060.00 Hz TECO</p> <p>Entry Accepted</p>	<p>Displayed for 0.5sec Confirm the display.</p>
(7) REV RUN	<ul style="list-style-type: none"> DSPL FWD/REV 	<p>O/P Freq. 60.00 Hz Freq. Cmd. 60.00 Hz</p> <p>O/P Freq. 60.00 Hz Freq. Cmd. 60.00 Hz</p>	<p>LED ON REV</p>
(8) STOP	<ul style="list-style-type: none"> * STOP 	<p>O/P Freq. 0.00 Hz Freq. Cmd. 60.00 Hz</p>	<p>LED ON * STOP</p> <p>(Blinking while decel.) * RUN</p>

IV- 2-3. Les pompes primaires :

2917 tr/min	H max	43.9 m	} pompe Grund Fox
Q 21m ³ /h	H	34.6 m	

4KW	} Moteur Siemens
220-240 / 380-415 V	
13.8 / 8 A	
Imax15.2/ 8.95 A	
2910 / 2930 tr/min	
Cos 0.88/ 0.84	
CL :F	IP 55 50HZ

IV-2.4 Pompe Osmose Inverse :

2933tr/min	H max	207m	} pompe Grund Fox
Q 21m ³ /h	H	165.4 m	

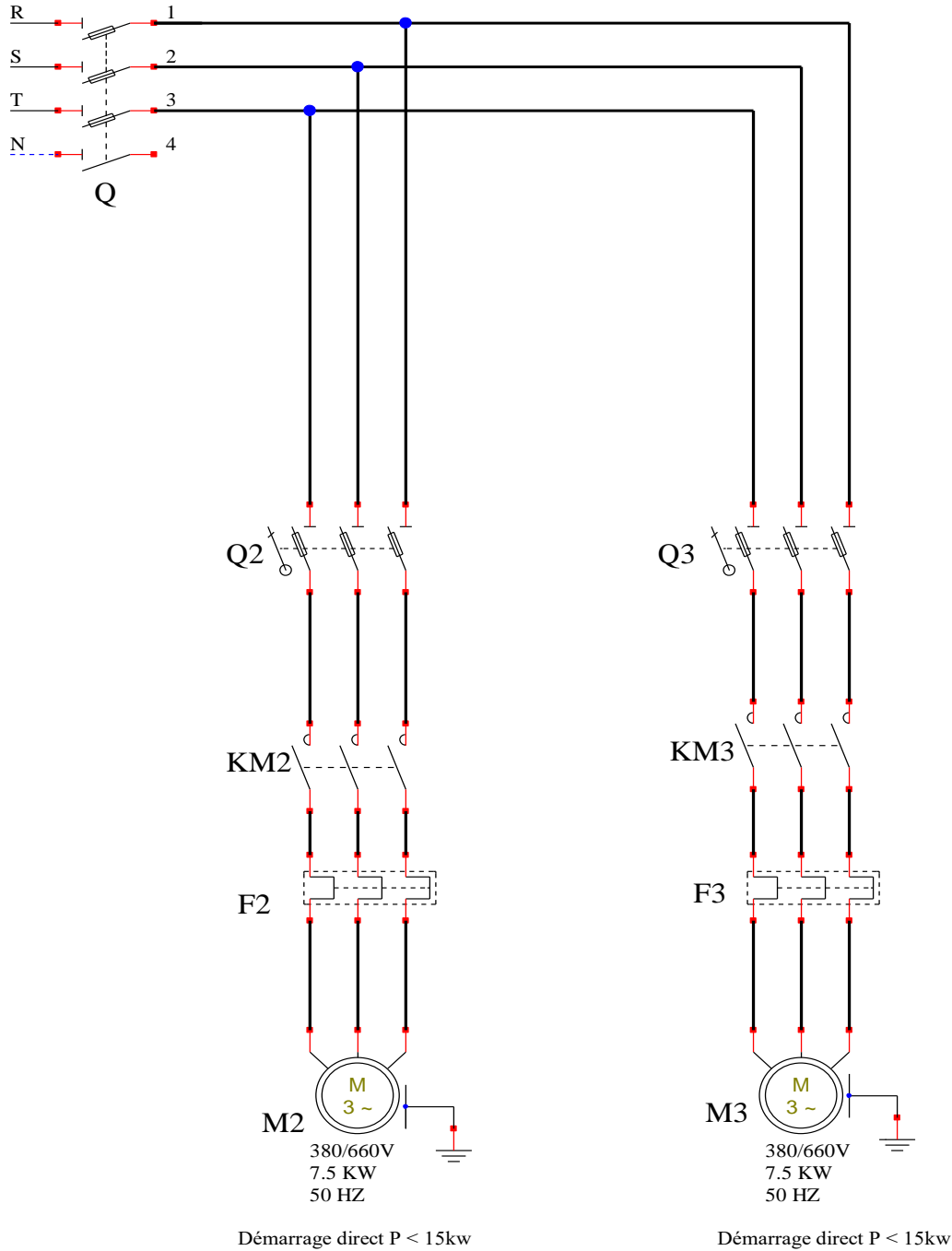
220-277/380-480V	} Moteur Siemens
Cos 0.9	
2945tr/min	
16KW	

IV-2.5 Electrovanes :

220V	0.3A
50/60 HZ	
To/Tf 12 Secondes.	

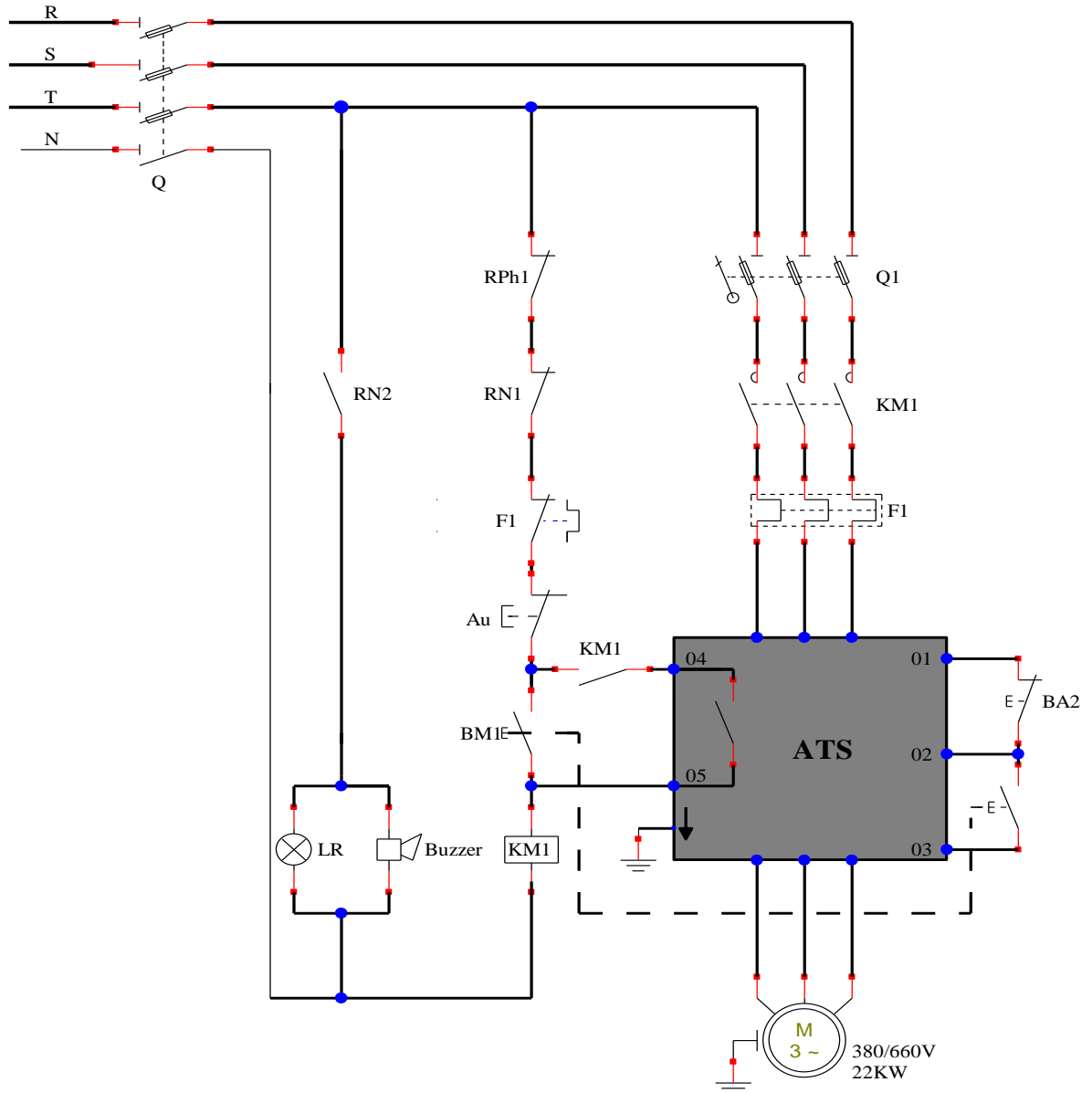
IV-3. CIRCUITS ELECTRIQUES :

- Circuit de puissance :



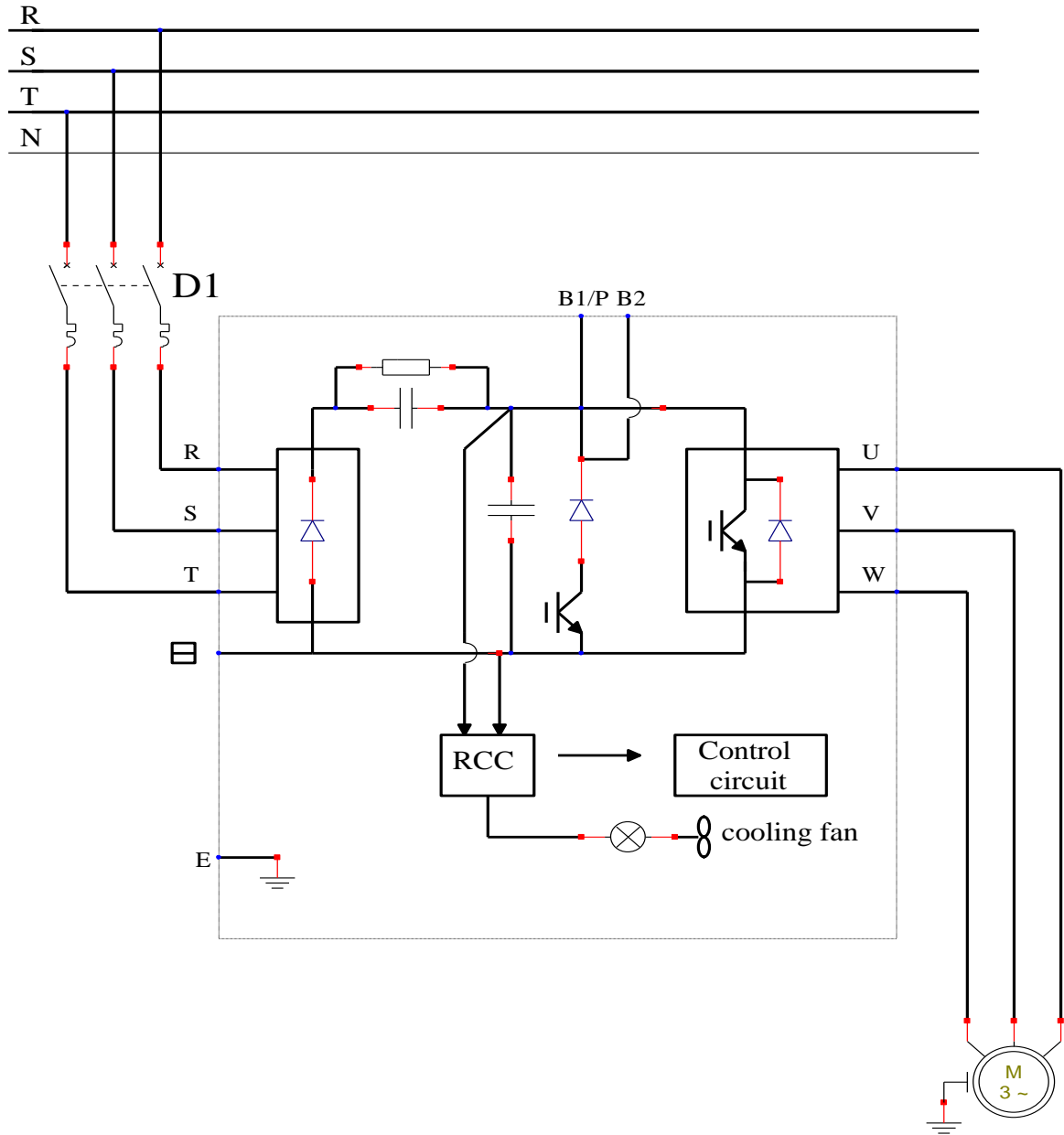
Moteurs pompes primaires

- circuit de commande et de puissance



MOTEUR POMPE FORAGE

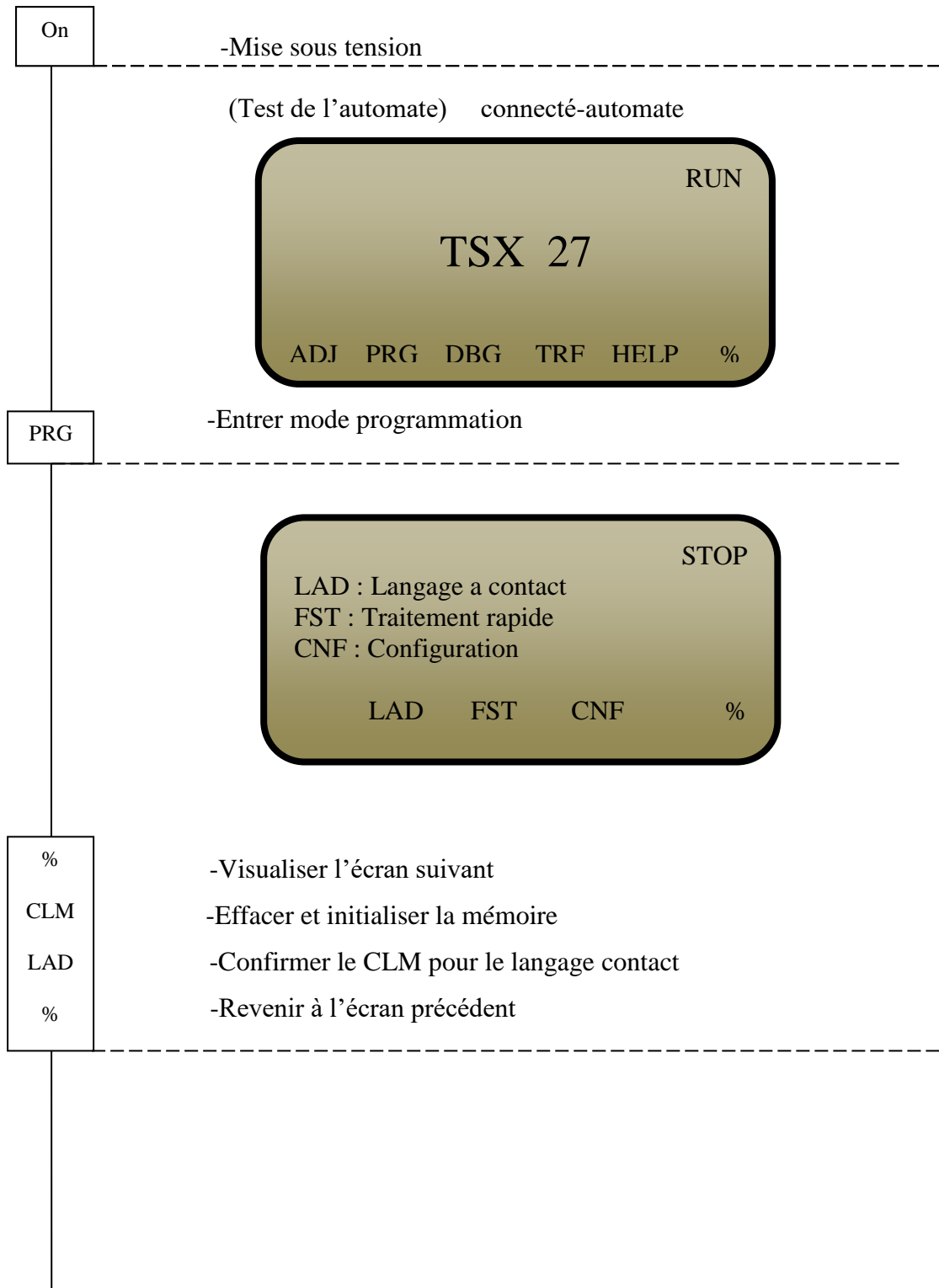
- Circuit de commande et de puissance :

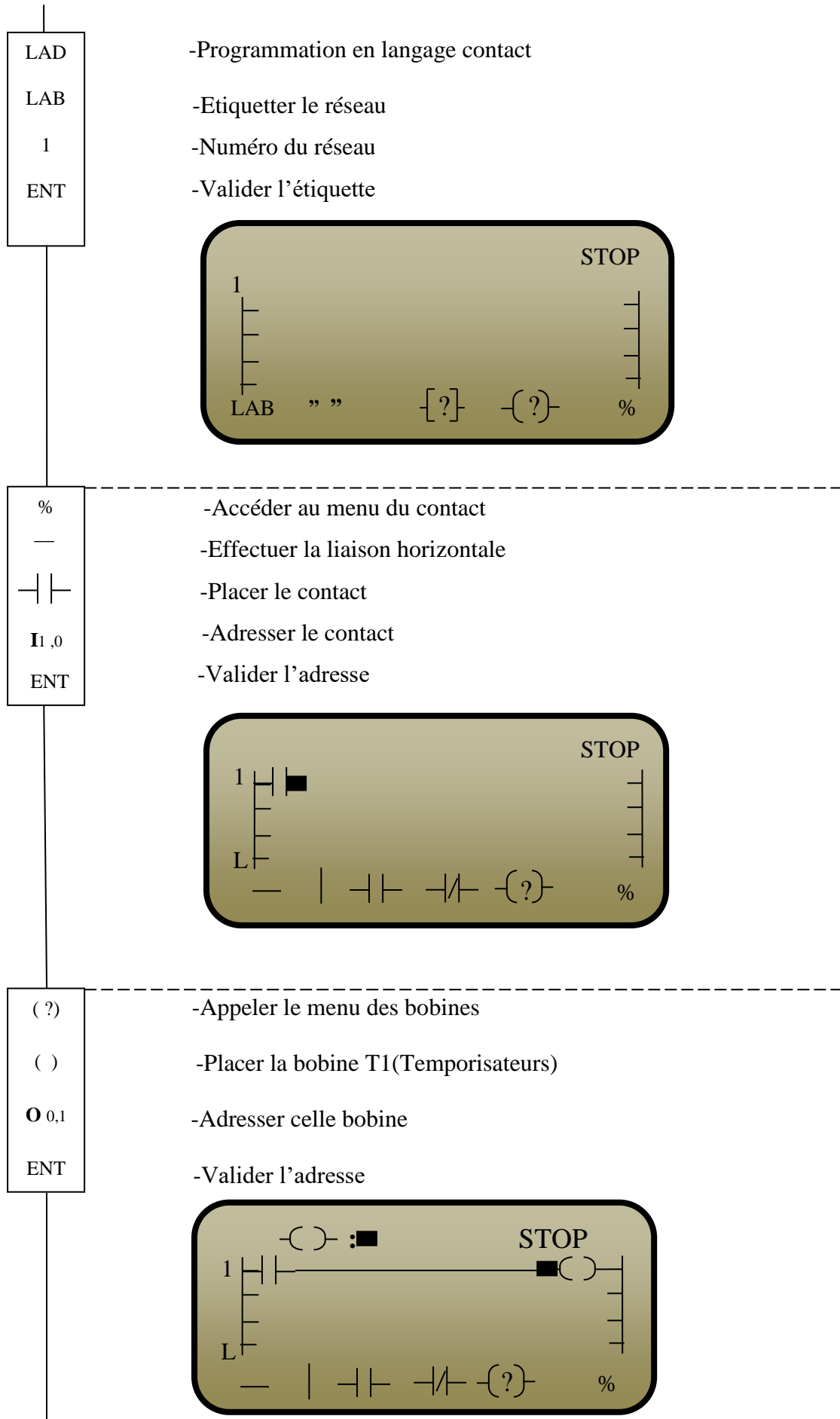


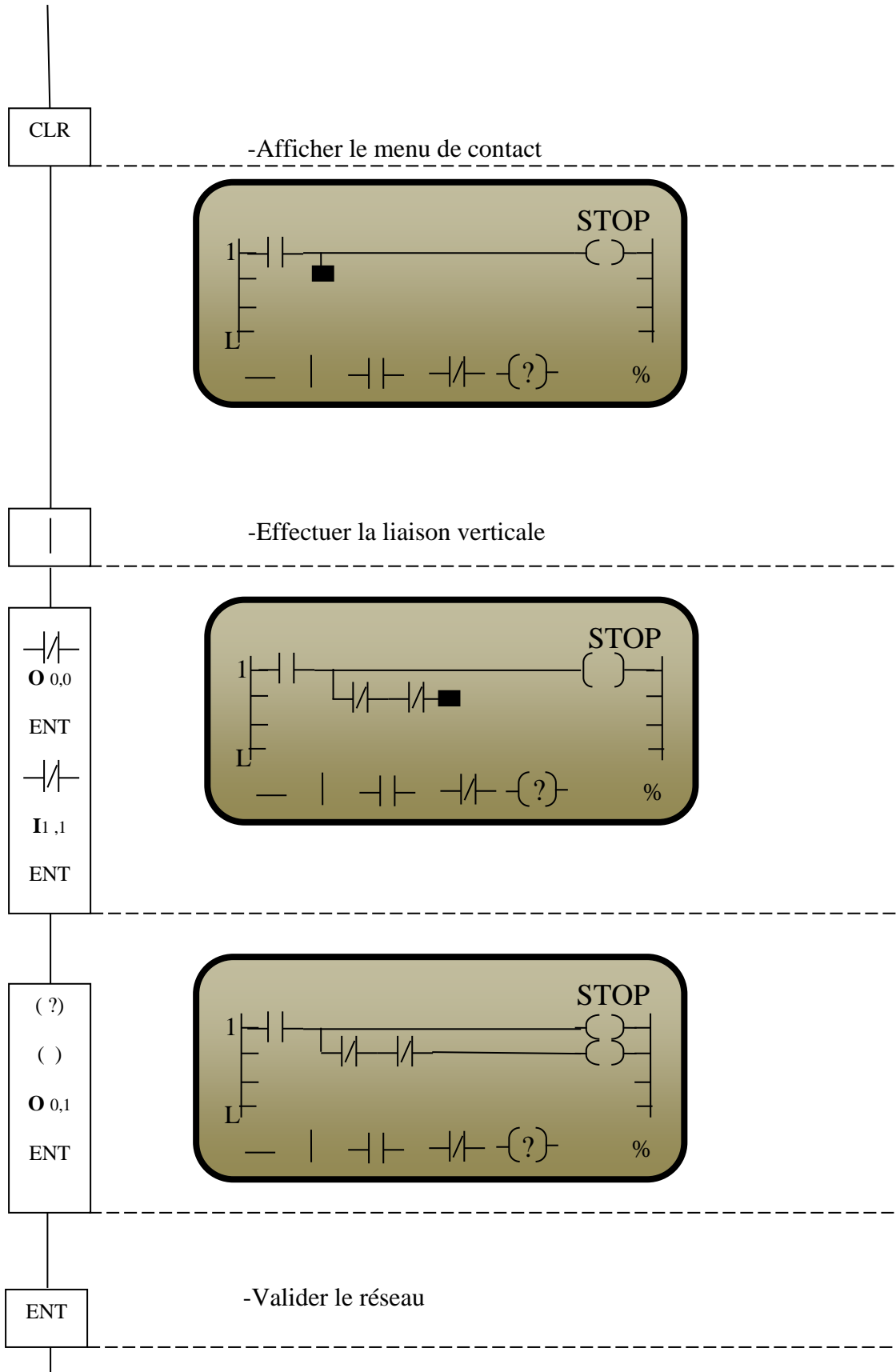
Moteur osmose inverse












IV-4. PROGRAMMATION :

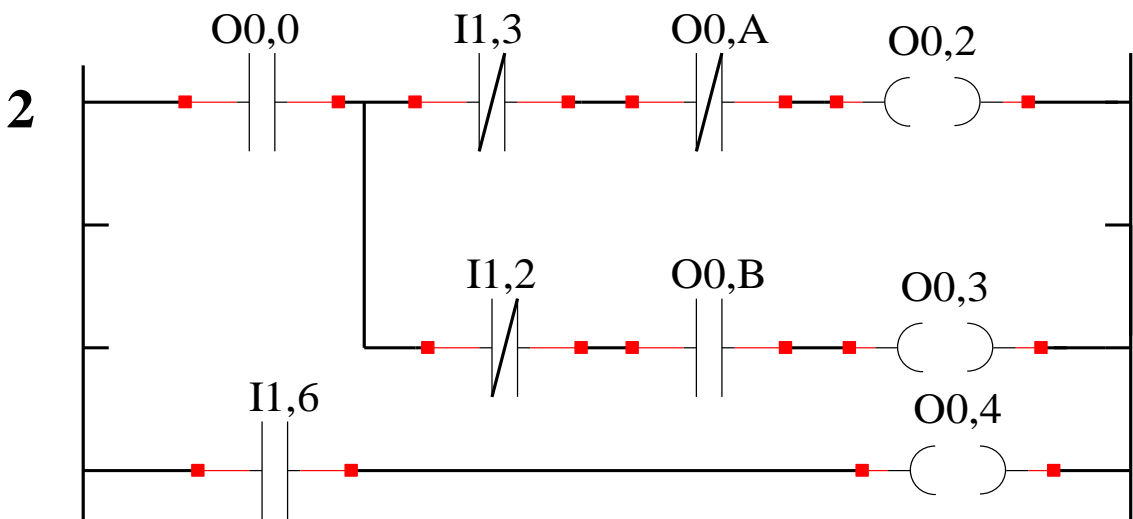
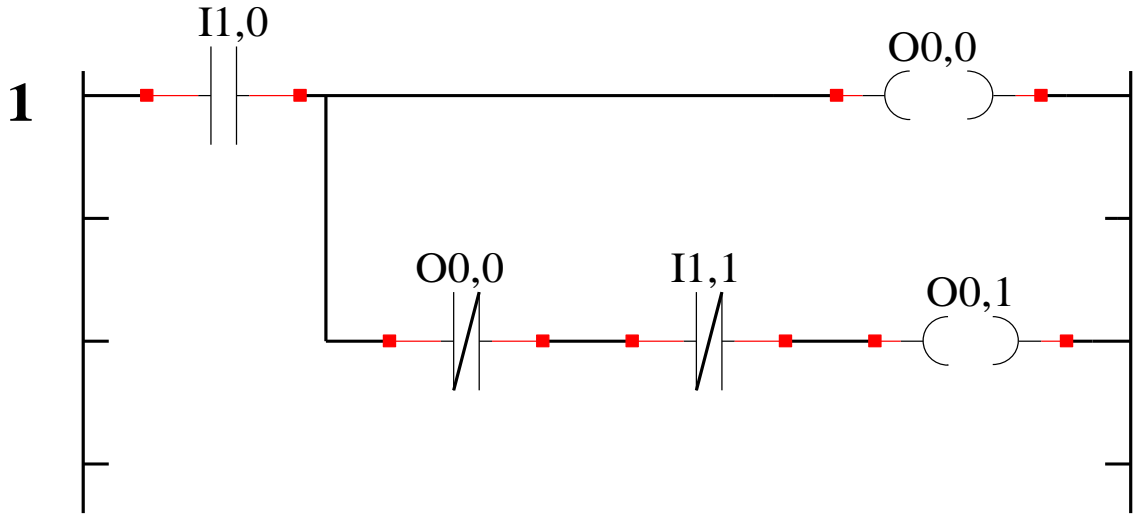
IV-4.LONGAGE CONTACT (ladder) :

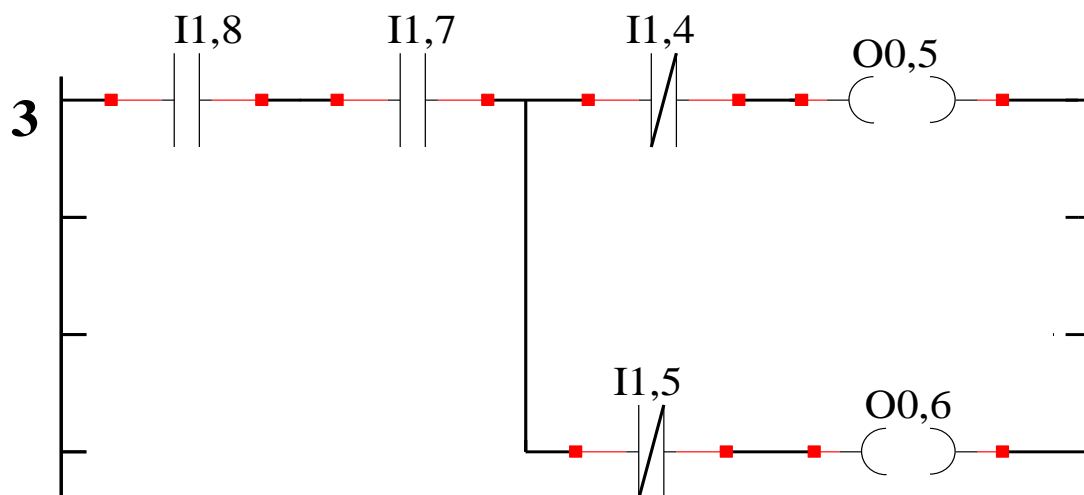










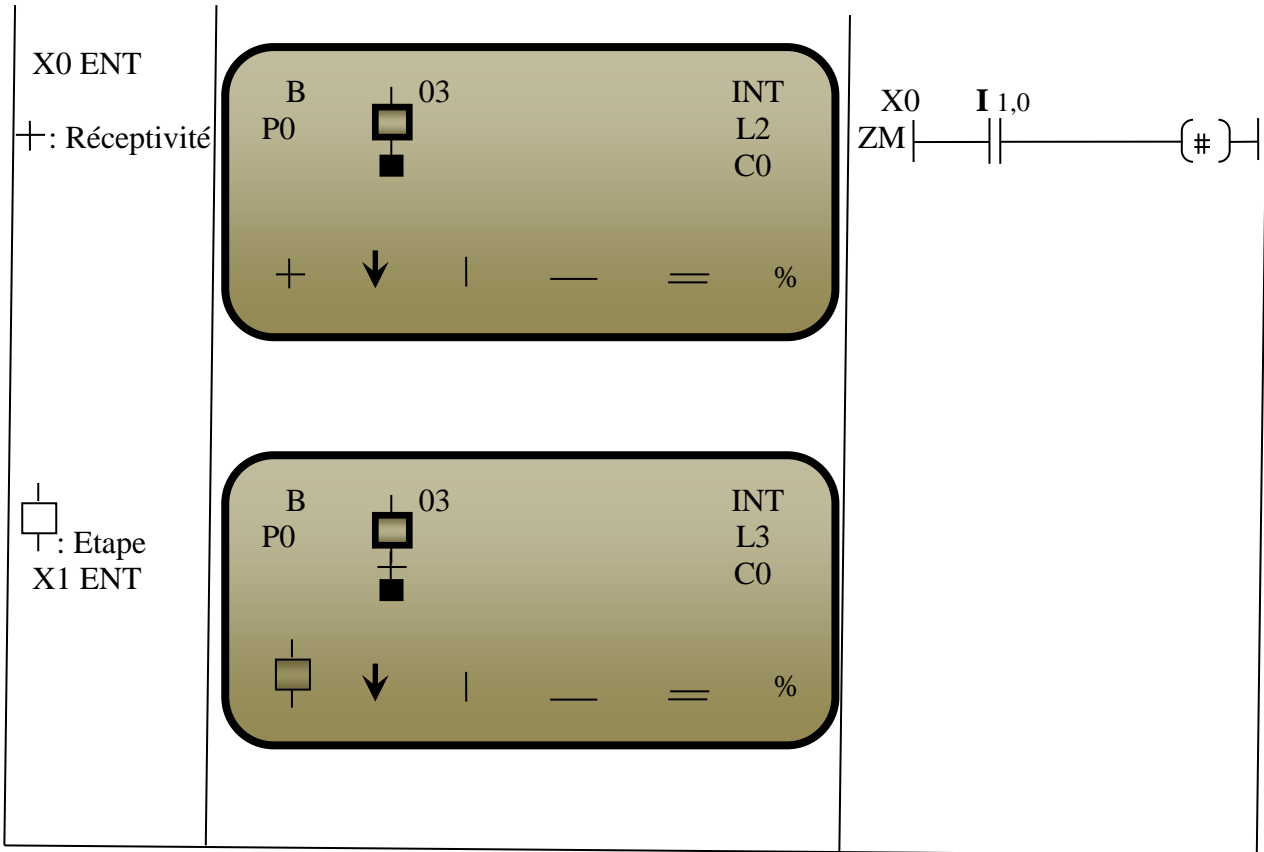
- CLR : Afficher le menu des contacts
 - LAB, 2 ENT : Etiqueter le second réseau
 -  **O0,0** ENT
 -  **I1,3** ENT
 -  **O0,A** ENT
 - (?) : Menu de bobine
 - () **O0,4** ENT
 - | : Liaison verticale.
 -  **I1,2** ENT
 -  **O0,B** ENT
 - CLR : Afficher le menu des contacts
 -  **I1.6** ENT
 - (?) : Menu de bobine
 - (?) **O0,4** ENT
 - ENT : Valider le réseau
 - LAB 3 ENT Etiqueter le troisième réseau
 -  **I1,8** ENT
 -  **I1,7** ENT
 -  **I1,4** ENT
 - (?) : Menu des bobines
 - () **O0,5** ENT
 - | : Liaison verticale
 -  **I1,5** ENT
 - (?) : Menu de bobine
 - () **O0,6** ENT
 - CLR : Afficher le menu des contacts
 -  **O0,6**
 - (?) : Menu de bobine
 - () **O0,7** ENT
- ENT
- QUIT : Retour à l'écran de sélection.
- DBG : Passer en mode mise au point.
- RUN/STOP : lancer le programme.
- YES : confirmation.





IV-4.2 LANGAGE GRAFCET :

Saisie du graphe	L'écran	Saisie des transitions
<p>PRG SEQ</p> <p>↓</p> <p>X3 ENT : Étiqueter L'étape</p> <p>% : Visualiser L'écran suivant</p> <p>□ : Étape</p> <p>□ : Étape initiale</p>		
		
		
		



+ : Réceptivité

□ : Etape

X2 ENT

+ : Réceptivité

□ : Etape

X3 ENT

+ : Réceptivité

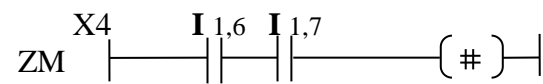
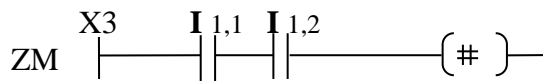
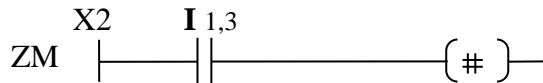
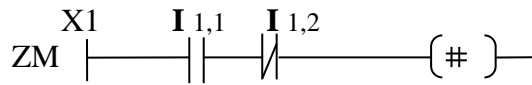
↓ : Renvois à l'origine X0

— : Aiguillage en OU

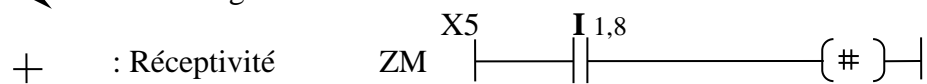
+ : Réceptivité

↓ : Renvois à l'origine X1

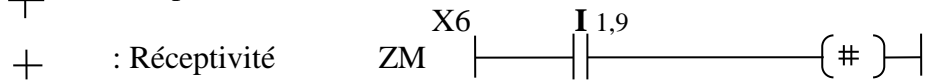
— : Aiguillage en OU



— ← : Vers la gauche



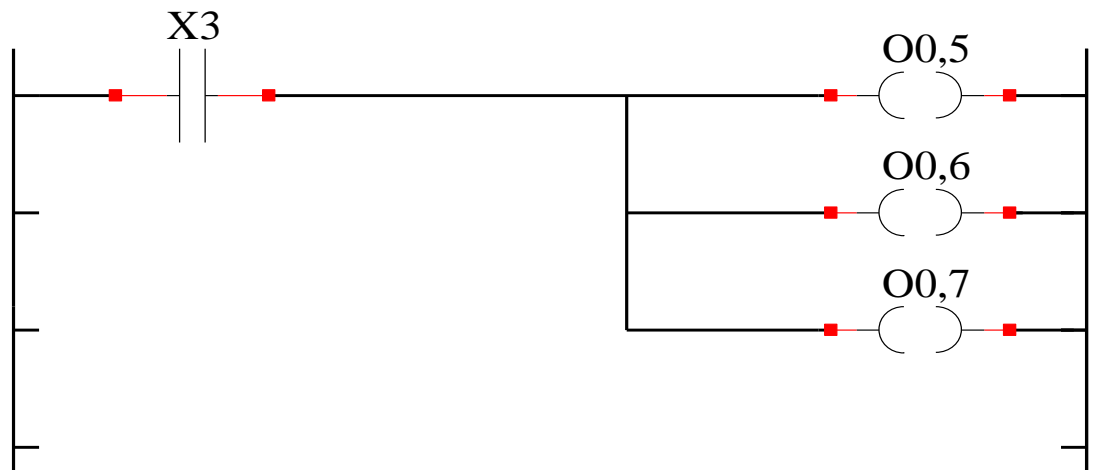
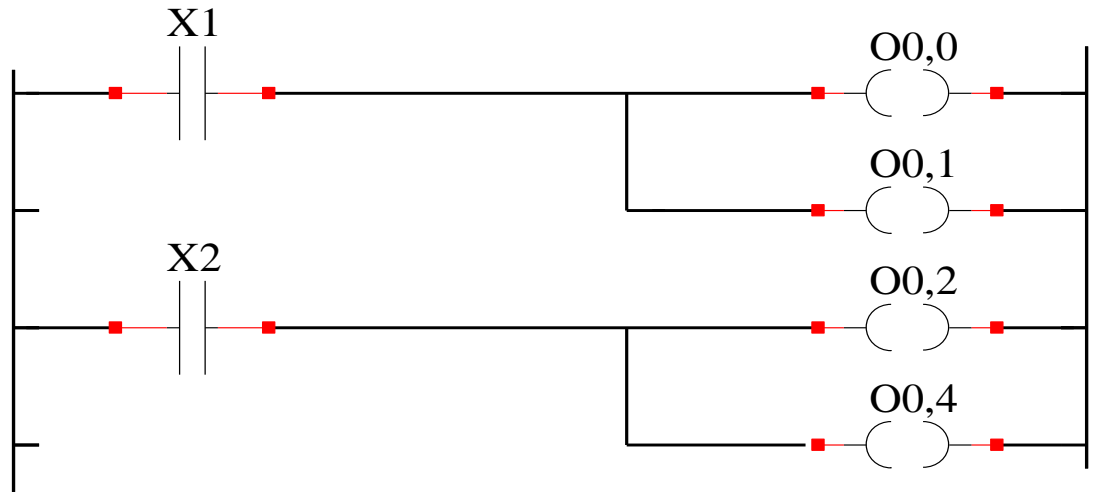
▣ : Etape

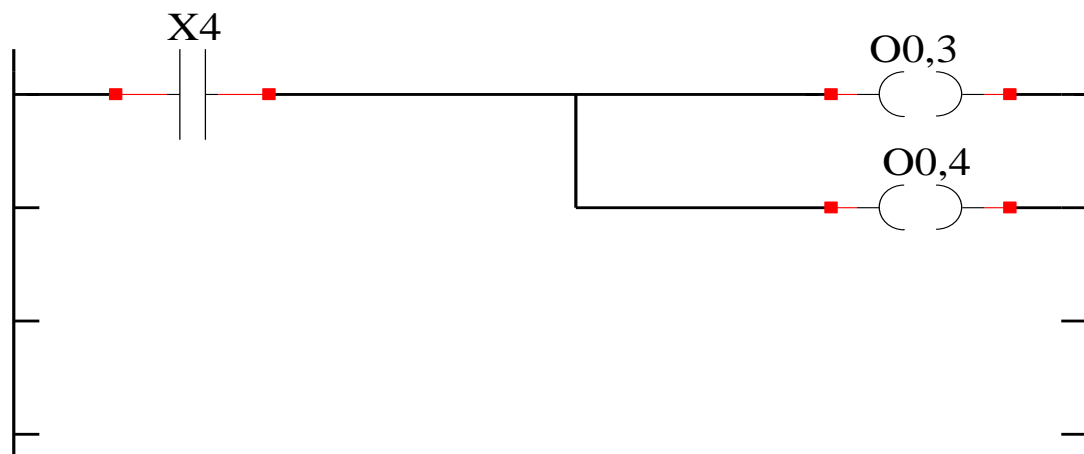


— : Aiguillage en OU

→ — : Vers la droite

Programmation des actions





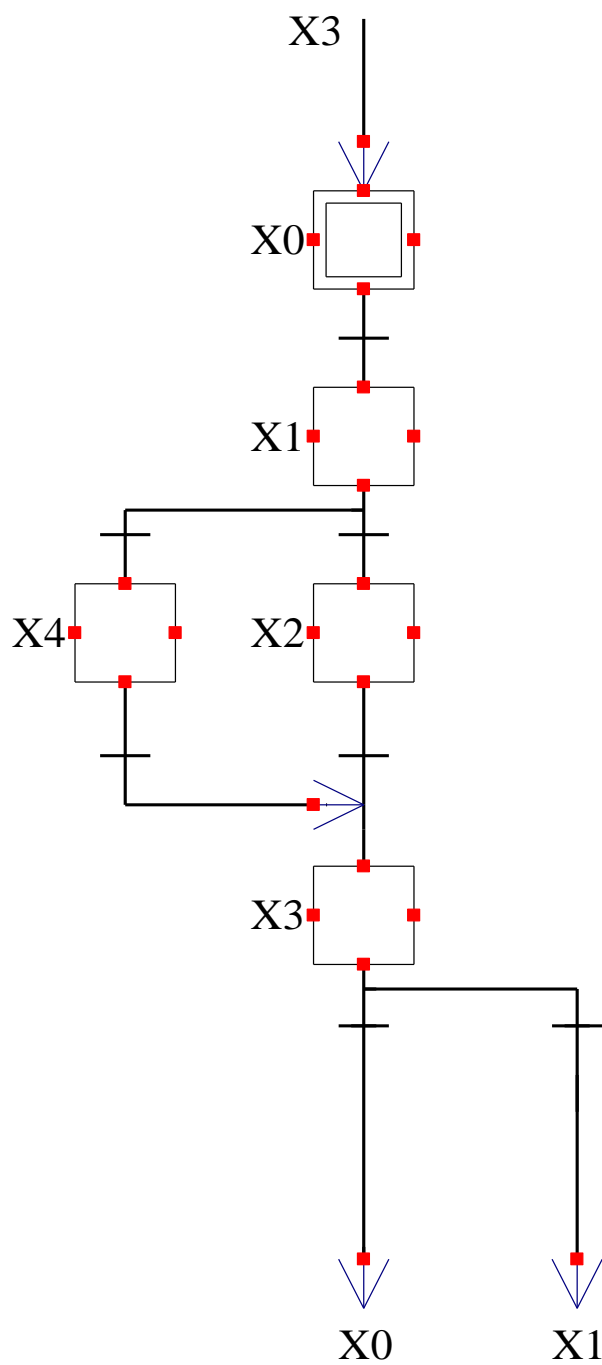


Tableau d'adressage des entrées/sorties :

I1, 0	Contact fermeture Marche Automatique
I1, 1	Contact ouverture Marche Manuel
I1, 2	Contact à ouverture Marche Manuel
I1, 3	
I1, 4	
I1, 5	
I1, 6	Contact fermeture Marche Automatique
I1, 7	Contact du Pressostat
I1, 8	Contact fermeture Marche Automatique
O0, 0	Contact Temporisé
O0, 6	Contact fermeture Relais pompe OI
O0, A	Contact ouverture horloge
O0, B	Contact fermeture horloge
O0, 0	Temporisateur à deux contact : 10+1F
O0, 1	Relais de commande Electrovanne 1
O0, 2	Contacteur de commande PA
O0, 3	Contacteur de commande PB
O0, 4	Contacteur de commande horloge
O0, 5	Relais de commande Electrovanne 2
O0, 6	Relais de commande pompe OI
O0, 7	Relais de commande pompe Jav et UV

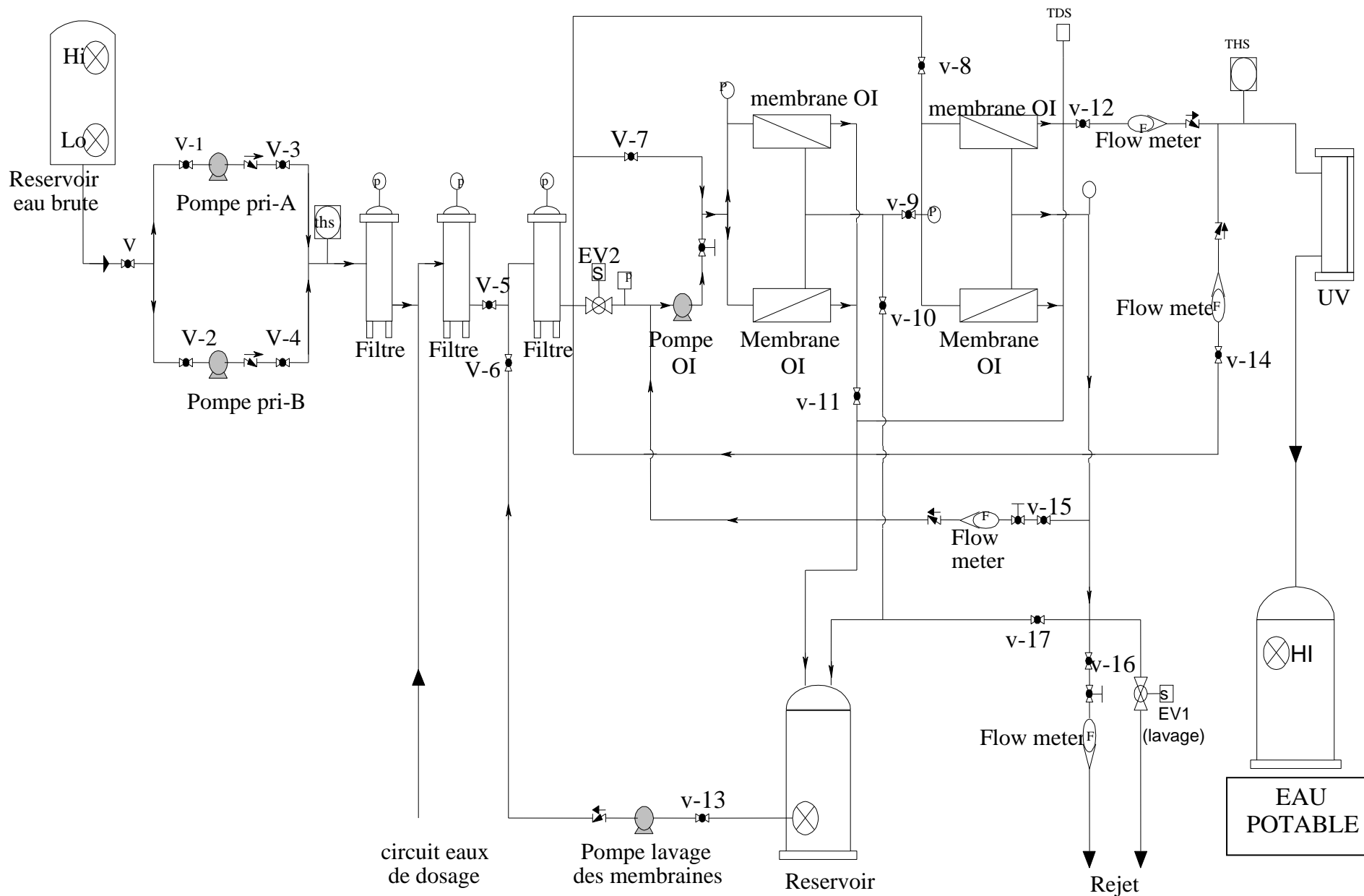


Diagramme fonctionnel de la station

Bibliographie

- [1] Séminaire Batna 93 sur le TSX.
- [2] Guide d'utilisation et programmation télémechanique d'API 1984/1986.
- [3] TECO SPECON 72000 MA instruction manuel 08/07.
- [4] Schneider Electric-catalogue automatismes industriels 2001.
- [5] Mémoires de fin d'étude
- [6] Etude des API (cours elec) Roizot bébotien.
- [7] Document infilco 06/2006 sur UCM.
- [8] L'automatisation présenté par-VRIGNON et M.THENAISSIE.
- [9] Les automates programmables industrielles (Doc).
- [10] Les automates programmables Technologie de ponté bossy