



REPUBLIQUE ALGERIENNE
DEMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la
Recherche Scientifique



UNIVERSITE MOULOU D MAMMERI DE TIZI OUZOU

FACULTE DE GENIE ELECTRIQUE ET INFORMATIQUE
DEPARTEMENT D'ELECTROTECHNIQUE

PROJET DE FIN D'ETUDES
D'INGENIEUR D'ETAT EN ELECTROTECHNIQUE
OPTION MACHINE ELECTRIQUE

THEME

*ETUDE ET AMELIORATION DE LA STATION DE TRAITEMENT
DE SURFACE AU NIVEAU DE L'ENIEM PILOTEE PAR UN
AUTOMATE TELEMECANIQUE TSX 57 203*

Présenté par :

M^r KADI HOCINE
M^r MASSAOUDI M'HAMED
M^{elle} AIT HAMOU ASSIA

Dirigé par :

M^r S.FELLAG

Proposé par :

M^r M.CHALLAL

Promotion 2007/2008



REMERCIEMENTS

On exprime notre profonde gratitude à Monsieur S.FELLAG, de nous avoir encadré et d'avoir dirigé ce travail avec dévouement. On le remercie sincèrement pour ses conseils, sa patience et sa disponibilité tout au long de notre projet.

Nos reconnaissances vont également à notre co-promoteur Monsieur M.CHALLAL qui nous a proposé ce sujet.

Nous tenons à remercier Monsieur S.YAHIA CHERIF pour tous ses précieux conseils qui nous a donnés.

Nous tenons à remercier Mr A.Bouamara et le groupe S.N.C générale maintenance pour l'aide qu'ils nous ont fournit.

Nous remercions vivement les membres de jury qui nous feront honneur de juger notre travail.

Nous remercions aussi tout ceux qui ont contribué de près ou de loin à l'aboutissement de ce mémoire.



Introduction générale

On peut toujours augmenter la production en accélérant les cadences de l'opérateur. C'est facile mais dangereux, l'homme ne supporte pas.

On peut aussi produire sans se soucier du temps et donc du prix de revient, mais une telle production serait vite stoppée par les lois du marché.

L'évolution prévisible de toute société industrielle dans les années à venir nous permet d'affirmer que la production à moindre coût sera encore plus que par le passé, l'une des données essentielles. Si nous projetons sur cette toile de fond toujours présente, les domaines où l'homme ne peut, ou ne veut intervenir directement, soit parce qu'il serait obligé d'évoluer dans une atmosphère insalubre voir mortelle, soit parce qu'il devrait effectuer des tâches répétitives dénuées d'intérêts, soit enfin parce que la fiabilité et la sécurité d'un processus l'imposent, on peut prévoir un développement toujours plus intense des systèmes automatisés.

La commande des processus par l'automate programmable est la solution recherchée de plus en plus dans l'industrie vue la justesse des traitements qu'il effectue pour gérer une commande adéquate à tout moment et dans toutes les conditions.

Les exemples de système automatisé de production sont nombreux, les plus représentatif et les plus gourmands en automate sont les centres de fabrication d'automobile.

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

La technologie d'automate permet une automatisation de plus en plus flexible et évolutive des équipements de production.

Notre travail consiste à étudier et améliorer l'installation de traitement de surface au niveau de l'unité cuisson de l'ENIEM.

Le contenu de ce travail est organisé comme suit :

- Le premier chapitre est consacré à la description de l'installation et le processus de production ainsi que le cycle de travail.
- Dans le second nous présentons des généralités sur les systèmes automatisés et l'API TSX 57203.
- Le troisième chapitre est consacré à la modélisation de la station.
- Dans le quatrième chapitre nous décrivons le langage de programmation PL7 et illustrons un exemple de programmation.
- Dans le cinquième chapitre nous présentons quelques propositions pour amélioration au niveau temps de cycle, sécurité et performance pour la station.
- Notre travail se termine par une conclusion générale.

I. Introduction

Le traitement de surface des matériaux est une étape importante dans la fabrication de différents types d'équipements.

A l'origine, les surfaces présentent des caractéristiques physiques, chimiques ou électriques non étudiées et peuvent donc être non adaptées à leur utilisation. Afin de modifier ces caractéristiques et les rendre exploitables dans les domaines voulus, un ensemble de traitement leur est appliqué. Il existe plusieurs techniques de traitement de surface tel que la peinture, le traitement mécanique, thermodynamique, traitement de conversion et par voie aqueuse.

L'installation de zingage utilise le traitement par voie aqueuse.

II. Présentation de l'installation [14]

L'installation est du genre automatique en ligne avec deux chariots, les traitements principaux appliqués sont le zingage acide et la passivation.

L'installation est composée de suites de cuves formant des chaînes de production. Les bains correspondent à des bains de traitement, de rinçage et un bain de séchage. [figure I.1]

Les pièces subissent donc une succession de trempage dans les différents bains afin d'obtenir le traitement désiré. L'enchaînement nécessaire au traitement complet d'une pièce se fait par le passage successif dans les différents bains qui est assuré par les deux portiques guidés automatiquement. Le passage n'est pas forcément d'un bain à celui qui le succède dans la chaîne.

Les photos des portiques sont représentées en figure I.2.

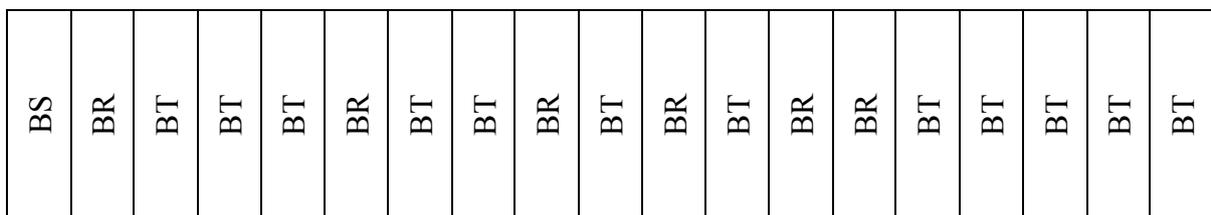


Figure I.1 : *structure de la chaîne*

BT : bain de traitement.

BR : bain de rinçage

BS : bain de séchage



Figure I.2 : *photos de l'installation*

II.1. Description des bains

Les bains sont construits en tôle d'acier et ils sont adéquatement renforcés et indéformables, ils sont devisés en deux parties séparées par une paroi, le revêtement intérieur en pvc, tandis qu'à l'extérieur ils sont protégés par du plastisol donné au pistolet et cuit au four. Il sont conçus de manière que l'écoulement des fluides par la vanne de vidange disposé en un point bas se fasse complètement et rapidement. Ils sont complétés par une décharge à bride et d'un couple d'appuis pour les barres portes montages.

II.1.1. Bain de traitement

L'immersion d'une pièce dans l'un de ces bains permet d'obtenir un traitement bien spécifique selon la composition du bain, les différents bains de traitement sont :

➤ **Bain de dégraissage chimique**

Il permet de nettoyer les pièces de toutes traces de l'huile et de graisse, il est complété par :

- Un serpentin pour le chauffage en acier inox.
- Une gaine pour le thermostat.
- Quatre hottes d'aspiration en acier inox placées transversalement à la cuve et complétées de couvercle d'inspection.
- Un dispositif pour l'agitation mécanique.

➤ **Bain de décapage**

Il permet d'enlever la rouille par l'attaque des acides sur l'acier à l'aide d'acide sulfurique et chlorique.



Il est complété par six hottes d'aspiration placées transversalement à la cuve et complétée par un couvercle d'inspection, et un agitateur.

L'installation comprend deux bains de décapage identiques.

➤ **Bain de dégraissage anodique**

Il permet d'enlever les traces de décapage et les impuretés positives, il est complété par :

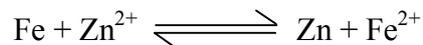
- Une armature galvanique en cuivre de section adéquate et isolée avec des crochets.
- Un serpentin en acier inox pour le chauffage.
- Une gaine pour le thermostat.
- Quatre hottes d'aspirations en pvc placées transversalement à la cuve et complétées de couvercles d'inspection.

➤ **Bain de neutralisation**

Contient de l'acide chlorique, il permet d'enlever les rouilles de décapage et d'arrêter la réaction chimique après traitement.

➤ **Bain de zingage**

Le zingage permet de déposer une couche de zinc sur les surfaces des pièces à l'aide de réaction oxydoréduction.



Il est complété par :

- Une contre bride close.
- Une armature galvanique en cuivre de section adéquate et isolée avec des crochets
- Un serpentin en titane pour le refroidissement.
- Six hottes d'aspiration en pvc placées transversalement à la cuve et complétée de couvercle d'inspection.
- Un agitateur.
- Des barres de zinc pur émergées dans ce bain et alimentées via un redresseur, permettant d'obtenir des molécules de Zn.

L'installation comprend quatre bains de zingage identiques.

➤ **Récupération (bain mort)**

C'est un bain vide qui permet de récupérer les gouttes du zinc acide des pièces sortantes du bain du zingage.

➤ **Bain de passivation**

Il permet l'esthétique de la pièce.

Il est complété par une soufflerie en tube de PVC.

II.1.2. Bain de rinçage

Les bains de rinçage permettent de :

- ✓ Nettoyer les pièces pour qu'elles soient aptes à subir le traitement chimique de l'opération suivante.
- ✓ Eviter le mélange des différentes substances des bains.
- ✓ Arrêter les réactions chimiques sur les surfaces des pièces.

L'installation comprend deux types de rinçage :

- ✓ Le rinçage chaud : ce bain est muni d'un serpentin pour le chauffage et d'une gaine pour le thermostat, il permet aussi d'enlever les traces du chrome.
- ✓ Le rinçage froid (à température ambiante), avec six bains identiques et ils sont complétés par un dispositif pour l'alimentation de l'eau avec une vanne de réglage.

II.1.3. Bain de séchage

Il est constitué en tôle d'acier, fonctionne avec le système de la ventilation forcée à air chaud.

L'air chaud fait circuler par deux ventilations opposées aux radiateurs, passe contre les pièces en décrochant les gouttes et les séchant. Un système de hottes soufflantes est attaché autour de la cuve et tout l'ensemble est isolé avec des panneaux synthétiques protégés par une tôle d'acier peint avec du plastisol donné au pistolet et cuit au four. L'intérieur du séchoir est peint avec un vernis à base de céramique.

La puissance des radiateurs à eau surchauffée est de 42000Kcal/h. La barre porte montage est placée sur le bord supérieur de la cuve, tandis que sur le fond il y a un manchon de décharge pour l'eau d'égouttement. Extérieurement, le séchoir est protégé par du plastisol donné au pistolet et cuit au four, tandis que l'intérieur est peint avec un vernis anti-acide à base de céramique.

II.2. Caractéristiques des bains

Les caractéristiques des bains sont regroupées dans le tableau suivant :

Bain	Nature	V (m ³)	T° C	concentration	Produit	régénération
01	Dégraissage chimique	3	70	60 à 66 g/l 5ml/l	PRESOL 7030 AB 40	Alcalinité du bain
02	Rinçage	2,6	Ambiante			
03	Décapage	5,2	Ambiante	10% V 20% V 3% V	H ₂ SO ₄ conc HCl conc NBT 12	50g/l Fe
04	Rinçage	2,6	Ambiante			
05	Dégraissage Anodique	3,5	40	50 à 57g/l 0,02ml/l	PRESOL 7030 AB 40	Selon l'alcalinité libre du bain
06	Rinçage	2,6	Ambiante			
07	Neutralisation	2,6	Ambiante	4% V	HCl conc	
08	Rinçage	2,6	Ambiante			
09	Zingage	7	25	170g/l 70g/l 40g/l 1g/l	NH ₄ Cl ZnCl ₂ ZnB ZnL	Filtration continue
10	Bain mort	2,6	Ambiante	Contient de la solution de Zn		
11	Rinçage	2,6	Ambiante			
12	Passivation (bleutée)	2,6	Ambiante	10g/l 1% V	F71 HNO ₃	Tout les 15 jours
13	Rinçage	2,6	Ambiante			
14	Rinçage	3	50		Eau chaude	
15	Séchage	2,8	100		Air chaud	

II.3. Quelques éléments de la chaîne

II.3.1. Barres portes montages

Les barres portes montages sont construites avec des plats en cuivres et elles sont complétées de crochets pour le transport et de guides en bronze pour les appuyer sur les cuves.

II.3.2. Accrochage des montages

L'accrochage et le détachage des montages, donc la charge et la décharge des pièces s'effectuent manuellement. Là où on accroche les montages aux barres de transport est la zone de travail la plus proche de l'installation ; elle est située au début de l'installation et quand le chariot amène la barre de transport avec les pièces traitées et la dépose au poste de chargement, deux ouvriers détachent les montages et accrochent d'autres montages avec des pièces à traiter.

II.3.3. Couvre bord

Où il est nécessaire, il y a des couvre bords en PVC pour éviter l'égouttement entre les cuves.

II.3.4. Groupe de transport

II.3.4.1 Guide pour les portiques

Les guides pour les portiques sont des railles composées par deux tubes en acier et une chaîne qui sert de crémaillère. Ces guides sont fixés à des poteaux accrochés aux cuves.

II.3.4.2. Portiques

Pour déplacer les barres d'une cuve à l'autre, on emploie des chariots, chaque chariot est construit en acier inoxydable, il est équipé de deux moteurs, un pour le soulèvement et l'autre pour le déplacement.

Le mouvement du déplacement est assuré par un pignon qui engrène sur une chaîne disposé le long des rails. Ce système donne une excellente précision en freinage et il évite un quelconque débandade.

L'équipement électrique comprend aussi des fins de course magnétiques pour la montée, la descente, les surettes et le contrôle de la position et des cuves.

L'installation est munie de deux chariots identiques.

II.3.4.3. Feston

L'alimentation électrique des moteurs et des services des portiques est faite à travers un câble à feston. À côté des rails il y a un guide dans lequel se déplacent des chariots de soutien des câbles. La longueur du câble permet le déplacement du premier portique entre le premier et le quinzième bain et entre le vingtième et le cinquième bain pour le deuxième portique.

II.3.4.4. Cames de contrôle

Groupe de toutes les Cames nécessaires pour l'arrêt des chariots sur les cuves et le compte des positions.

Ils sont logés sur des guides fixés sur les rails pour rendre plus faciles les réglages.

II.3.4.5. Dispositif de sécurité

Les systèmes de transmission des chariots sont protégés par des carters tandis qu'il y a pour chaque mouvement des fins de course de sûreté qui surveillent le dispositif électrique de contrôle de chariot.

II.4. Capteur [4]

Les capteurs sont des composants de la chaîne d'acquisition dans une chaîne fonctionnelle.

Ils prélèvent une information sur le comportement de la partie opérative et la transforment en une information exploitable par la partie commande. Cette information est une grandeur abstraite qui précise un événement particulier parmi l'ensemble d'événements possibles. Cette information est portée par un support physique (énergie) qui est un signal. Les signaux sont généralement de nature électrique ou pneumatique.

L'information donnée par le capteur pourra être logique, numérique ou analogique.

II.4.2. Choix d'un capteur

Pour choisir un capteur, il faudra définir

- le type et la nature d'événement à détecter
- la grandeur de l'événement
- l'environnement de cet événement.

II.4.3. Différents types de capteurs utilisés

- a) **capteurs de position** : ce sont des capteurs de contact, ils peuvent être équipés d'un galet, d'une tige souple ou d'une bille. L'information donnée par ce type de capteurs est de type TOR. La détection est un contact entre la tige et l'objet à détecter.
- b) **Capteur optique** : un capteur photoélectriques est un capteur de proximité. Il se compose d'un émetteur de lumière associé à un récepteur. La détection d'un objet se fait par coupure ou variation d'un faisceau lumineux, cela provoquera l'amplification du signal pour être exploité par la partie commande.
- c) **capteur inductif** : les capteurs inductifs produisent à l'extrémité de leurs tête de détection un champ magnétique oscillant. Ce champ est généré par une self. Lorsque un objet métallique pénètre dans ce champ, il y'a perturbation de ce dernier, puis son atténuation. Cette variation est exploitée par un amplificateur qui délivre un signale de sortie provoquant la commutation du capteur.

Le positionnement des différents capteurs sur les portiques est représenté en figure I.3.

La vue de haut de l'installation est représentée en figure I.5.

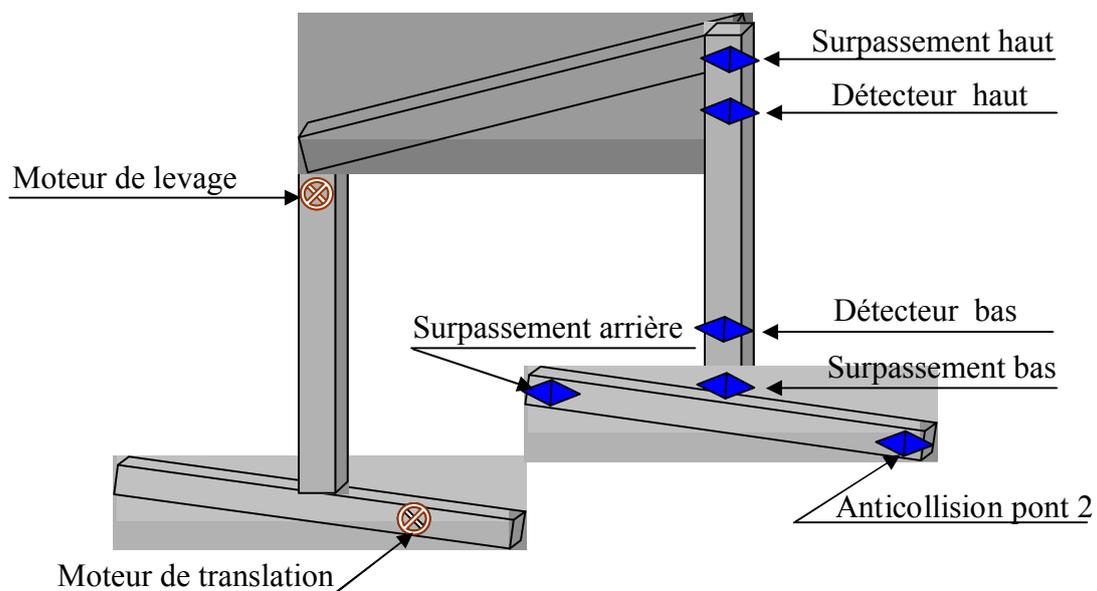


Figure I.3 : Image du portique 1.

II.5. Armoire de commande

Il comporte l'ensemble : l'automate programmable, un redresseur, deux variateurs de vitesse électroniques, l'ensemble des relais et interrupteur et il est muni d'un pupitre.



Figure I.4 : *armoire de commande.*

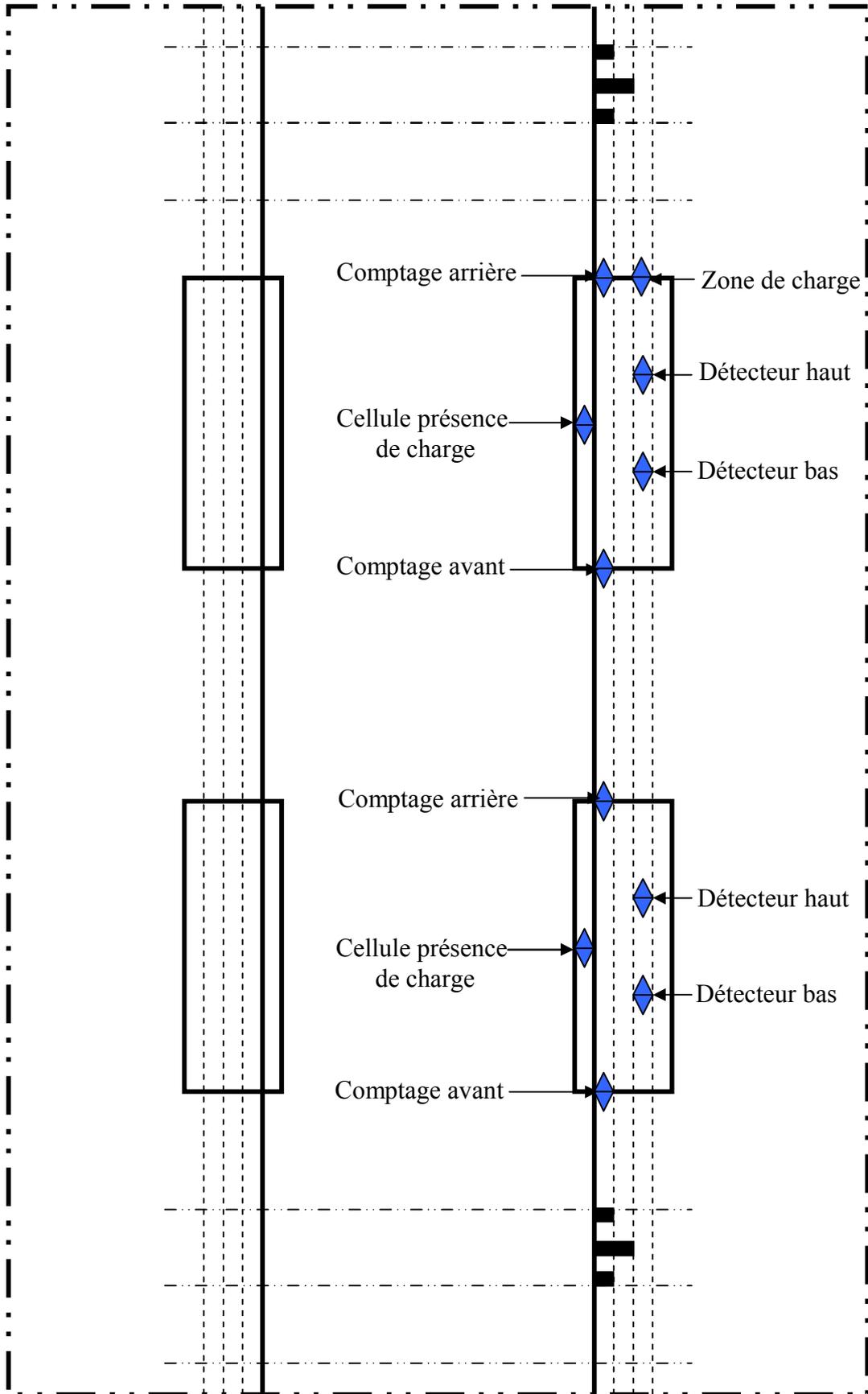


Figure I.5 : Vue de haut de l'installation.

II.5.1 L'automate

C'est le cerveau du système automatisé, c'est lui qui gère tout le processus (le mouvement des portiques, le cycle de travail, les alarmes, l'arrêt d'urgence...).

II.5.2. Pupitre

C'est un terminal de dialogue qui a pour fonction principales de :

- visualiser des données issues de l'automatisme.
- Modifier des paramètres de l'automatisme.
- Commander l'automatisme par des commandes TOUT ou RIEN.

II.5.3. Redresseur

Les montages redresseurs sont les convertisseurs de l'électronique de puissance qui assurent directement la conversion alternative – continue, alimentés par une source de tension alternative, ils permettent d'alimenter en courant continu le récepteur branché à leur sortie.

On utilise un redresseur chaque fois qu'on a besoin du continu alors que l'énergie électrique est disponible à chaque fois en alternatif. Comme c'est sous cette seconde forme que l'énergie électrique est presque toujours générée et distribuée, les redresseurs ont un très vaste domaine d'application.

Dans notre cas, l'installation de traitement de surface possède deux redresseurs alternatif – continu

- Le premier débite sur des barres de zinc émergé dans le bain de zingage afin que la réaction chimique s'effectue par l'électrolyse celui-ci donne à sa sortie une tension de 8V à partir de la tension du réseau et un courant de 500 A.
- Le deuxième alimente les différents capteurs et pré actionneur qu'utilise le système d'automatisme, la tension à sa sortie est de 24V et débite un courant de l'ordre de quelques ampères.

II.5.4. Variateur de vitesse [10]

Les deux moteurs de translation et de levage des deux portiques fonctionnent à vitesse variable. La vitesse de rotation réelle η_r d'un moteur asynchrone en fonctionnement normal est toujours inférieure à la vitesse de synchronisme η_s (champ tournant).

$$\eta_r = \eta_s (1-g) \text{ avec :}$$

g : glissement de (3 à 5%)

$$\eta_s = 60f/p \text{ avec :}$$

f : fréquence du réseau d'alimentation

p : nombre de paires de pôle

Il y a donc deux moyens de faire varier la vitesse de rotation d'un moteur asynchrone :

- agir sur le nombre de pôles à la construction (moteur à plusieurs bobinages)
- faire varier la fréquence du réseau d'alimentation (variateur de vitesse électronique).

L'installation possède deux variateurs de vitesses électroniques identiques, un pour chaque portique ; (un variateur commun pour le moteur de translation et le moteur de levage).

Un variateur de vitesse électronique est constitué d'un pont redresseur, d'un filtre (condensateur) et d'un pont onduleur.

La tension monophasée ou triphasée du réseau est convertie en une tension continue par l'intermédiaire du pont redresseur et des condensateurs de filtrage. Cette tension continue est découpée par un pont onduleur pour donner une succession d'impulsion de largeur variable (MLI : modulation à largeur d'impulsion).

L'ajustement de la largeur des impulsions et leur répétition permettent d'obtenir une fréquence variable tout en maintenant le rapport tension sur fréquence constante.

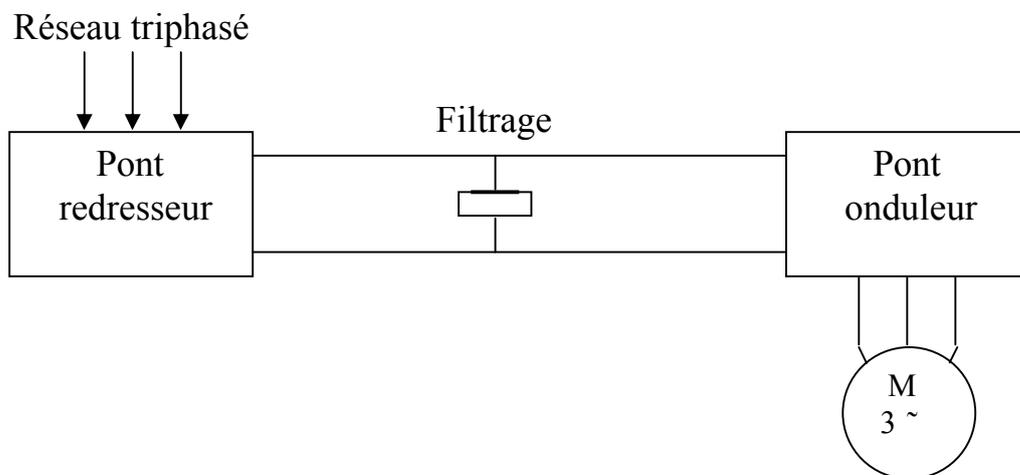


Figure I.6 : constitution d'un variateur de vitesse.

La tension monophasée ou triphasée du réseau est convertie en une tension continue par l'intermédiaire du pont redresseur et des condensateurs de filtrage.

Cette tension continue est découpée par un pont onduleur pour donner une succession d'impulsions de largeur variable (modulation de largeur d'impulsion).

L'ajustement de la largeur d'impulsions et leur répétition permettent d'obtenir une fréquence variable tout en maintenant le rapport V/f constant.

II.6. Moteurs électriques asynchrones [14]

II.6.1. Moteurs de levages

Les moteurs de levage sont des moteurs asynchrones triphasés à double sens de rotation muni de réducteurs de vitesse.

II.6.1.1. Caractéristiques des moteurs de levage

- vitesse de rotation 1400 tr/mn.
- puissance 0,75 KW.
- courant nominal 1,9 A.
- $\cos \varphi$ 0,90.
- montage triangle

II.6.2. Moteurs de translations

Les moteurs de translation sont des moteurs asynchrones triphasés, munis de réducteurs de vitesse et possèdent deux sens de rotation.

II.6.2.1. Caractéristiques des moteurs de translation

- vitesse de rotation 1350 tr/mn
- puissance 0,37 KW
- courant nominal 0,83 A
- $\cos \varphi$ 0,91 A
- montage triangle

II.6.3. Agitateurs

Les agitateurs sont des moteurs asynchrones triphasés munis de réducteurs de vitesse à une seule vitesse de rotation et un seul sens de rotation.

II.6.3.1. Caractéristiques des moteurs

- vitesse de rotation 28,4 tr/mn
- puissance 0,18 KW
- courant nominal 0,77 A
- $\cos \varphi$ 0,66
- montage triangle

III. Fonctionnement de l'installation**III.1. La commande des portiques**

La commande des portiques est assurée par un automate programmable industriel ; celui-ci gère l'ensemble des mouvements des moteurs de levage et de translation ainsi que les alarmes et les sécurités. La flexibilité du programme de l'automate permet de varier le cycle et de modifier le mode de traitements des pièces selon les critères et les exigences du besoin. à titre d'exemple, on peut jouer sur le temps d'immersion des pièces dans les différents bains et les temps d'attentes des portiques.

III.2. Mouvement de translation

Juste à l'entrée du bain où le portique est sensé s'arrêter, le moteur de translation subit un ralentissement afin de s'arrêter définitivement à la zone de chargement. L'arrêt brusque peut provoquer la tombée des pièces suspendues sur les corbeilles de transport.

III.3. Mouvement de levage

Pour éviter l'égouttement et afin de limiter la pollution du bain suivant, le moteur de levage marque un petit temps d'arrêt à la fin du levage des montages de chaque bain avant le démarrage du moteur de translations.

Le temps de montée des moteurs de levage est de 8.5s et le temps de descente est de 8s.

III.4. Le cycle de fonctionnement des portiques

Le cycle de travail des deux portiques illustré dans le diagramme fonctionnel est conçu de manière à traiter le maximum de pièces d'une part et d'accorder le temps nécessaire pour le traitement d'autre part.

III.4.1. Cycle de travail

- 1- Poste de charge des montages.
- 2- Dégraissage chimique.
- 3- Lavage.
- 4- Décapage.
- 5- Lavage.
- 6- Dégraissage anodique.
- 7- Lavage.
- 8- Neutralisation.
- 9- Lavage.
- 10- Zingage acide.
- 11- Bain mort.
- 12- Lavage chaud.
- 13- Activation sulfurique
- 14- Passivation bleue.
- 15- Lavage.
- 16- Séchage.
- 17- poste de décharge des montages.

III.4.2. Le temps de cycle minimum

Le temps de cycle minimum est le moindre temps que met le portique pour effectuer toutes les opérations sur les cuves. Généralement, on l'utilise pour définir la productivité de l'installation, on pourra l'augmenter pour certaines exigences mais jamais le diminuer. Le zingage se fait avec un temps nominal d'immersion de 22 minutes et 40 secondes.

Le temps de cycle minimum est de 6 minutes et 30 secondes.

III.4.3. Table de charge et de décharge des portiques**Portique 1****Portique 2**

N°	PRISE	POSE	SYN
01	501	0	0
02	1	6	0
03	2	1	0
04	3	2	0
05	502	0	20
06	8	10	0
07	7	8	0
08	508	0	21
09	511	0	0
10	511	0	22
11	15	0	0
12	515	0	0
13	0	5	0
14	5	0	0
15	505	0	0
16	0	4	0
17	4	0	0
18	504	0	0
19	0	3	0
20	501	0	0

N°	PRISE	POSE	SYN
01	512	0	0
03	10	11	0
04	511	0	0
05	13	14	0
06	16	20	0
07	14	0	0
08	514	0	0
09	0	16	0
10	20	15	0
11	515	0	0
12	518	0	0
13	518	0	10
14	11	12	0
15	512	0	11
16	507	0	0
17	507	0	12
18	6	7	0
19	512	0	0
20	0	0	0

III.5. Démarrage de l'installation

La régulation de la température est un facteur très important, le bain de dégraissage chimique doit être chauffé à 70°C, le bain de dégraissage anodique à 40°C et celui du rinçage chaud à 50°C, les sondes de contrôle de température sont reliées à des afficheurs avec voyants situés en face avant de l'armoire de commande.

Un ouvrier est chargé de mettre en marche le dispositif de chauffage de ces bains au préalable trois heures avant le début de traitement des pièces pour atteindre les températures désirées. Il faut s'assurer que les deux portiques sont en position initiale.

Lors de la mise à l'arrêt précédent, le cycle revient à son état initial qui est la présence de la pièce dans les bains 1,2,3,7,8,9,10,12,16,17,18,19 (en réalité le dispositif reconnaît les barres porte montage comme étant un état de présence de charge), dans le cas contraire saisir les numéros des bains où les charges sont présentes dans les tables des ponts sur le pupitre de commande.

Après avoir vérifié ces contraintes on accède au démarrage de l'installation en suivant les instructions suivantes :

- Appuyer sur les boutons marche générale.
- Tourner les boutons hors sécurité à droite.
- Appuyer sur le bouton marche mobile.
- Valider le chargement en appuyant sur le bouton validation chargement.
- Appuyer sur le bouton marche des ventilateurs.
- Appuyer sur le bouton marche agitateur 1 et 2 (bains de zinc).
- Appuyer sur le bouton marche des aspirateurs 1 et 2 (bains mort).
- Appuyer sur le bouton acquittement de défaut.
- Tourner le bouton démarrage en automatique.
- Tourner le bouton hors sécurité à gauche.
- Appuyer sur le bouton départ cycle.

IV. Conclusion

Dans ce chapitre nous avons décrit les principaux éléments constituant l'installation et le cycle de fonctionnement des portiques.

L'automate programmable utilisé et le pupitre de commande n'ont pas été décrits dans ce chapitre, vu que l'automate constitue l'objet de notre étude et afin de bien décrire cet élément (constitution et fonctionnement), la majeure partie du chapitre suivant lui est consacré.

I. Introduction

Une machine ou un système est dit automatisé lorsque le processus qui permet de passer d'une situation initiale à une situation finale se fait sans intervention humaine et que le comportement est répétitif chaque fois que les conditions qui caractérisent la situation initiale sont remplies. Autrement dit l'automatisation est considérée comme l'étape d'un progrès technique où apparaissent des dispositifs techniques susceptibles de seconder l'homme, non seulement dans ses efforts musculaires, mais également dans son travail intellectuel de surveillance et de contrôle.

II. Définition d'un automate programmable industriel (API) [1]

Un automate programmable industriel est une machine (dispositif) électronique programmable par un personnel non informaticien et destiné à piloter en ambiance industrielle et en temps réel des procédés ou parties opératives. Il permet de traiter les informations entrantes pour émettre des ordres de sortie en fonction d'un programme.

Un automate programmable est adaptable à un maximum d'applications d'un point de vue traitement, composants et langage, c'est pour cela qu'il est généralement de construction modulaire.

III. Caractéristiques d'un API

- Compact ou modulaire
- Tension d'alimentation
- Taille mémoire
- Temps de scrutation
- Sauvegarde (EPROM, EEPROM, PILE,...)
- Nombre d'entrées/sorties
- Modules complémentaires (analogique, communication.....)
- Langage

IV. Structure d'un système automatisé [2]

IV.1. Composant d'un système automatisé

Le schéma ci-après illustre des composants d'un système automatisé

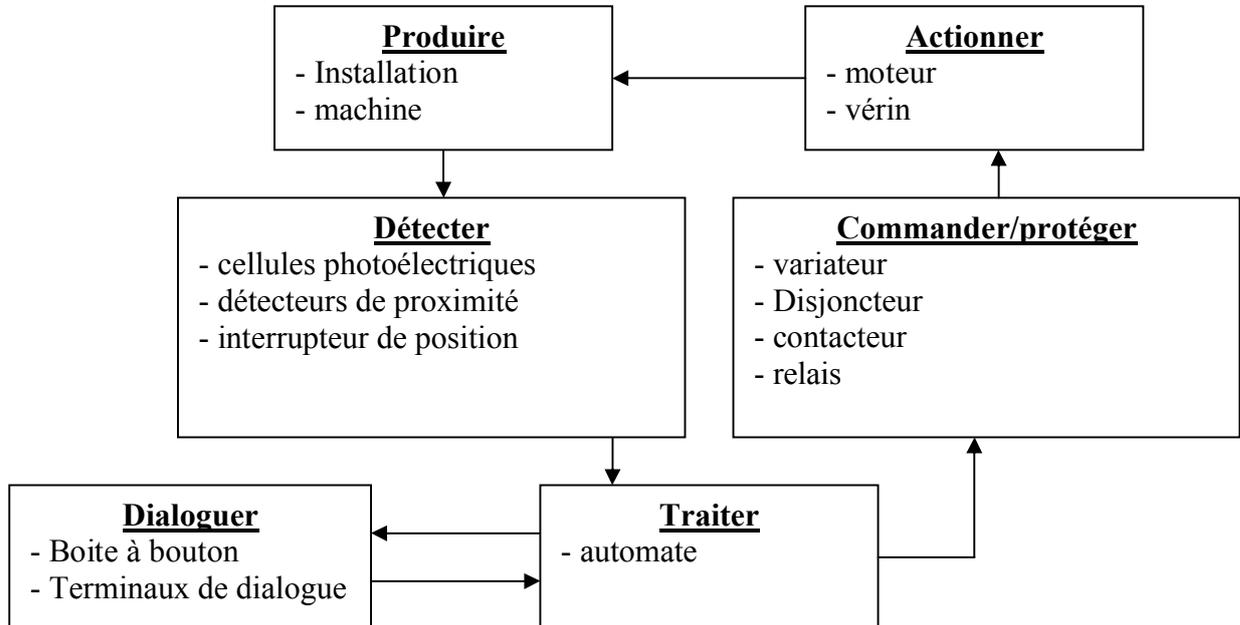


Figure II.1 : Composant d'un système automatisé.

L'ensemble d'un système automatisé comprend généralement une partie opérative, une partie commande et une partie relation.

IV.1.1. partie opérative

C'est celle qui opère ou agit sur la matière d'œuvre ou le produit. Elle comporte en général des actionneurs et outillage mécanique permettant leur élaboration.

Elle agit selon les ordres donnés par la partie commande, les exécute et émet des informations à celle-ci.

IV.1.2. partie commande

C'est elle qui émet des ordres vers la partie opérative et en reçoit des informations en retour afin de coordonner ses actions.

Elle peut être réalisée selon deux types de technologies :

IV.1.2.1. Logique câblée : qui correspond à un traitement parallèle de l'information, c'est-à-dire que l'on peut avoir une sollicitation simultanée de plusieurs constituants.

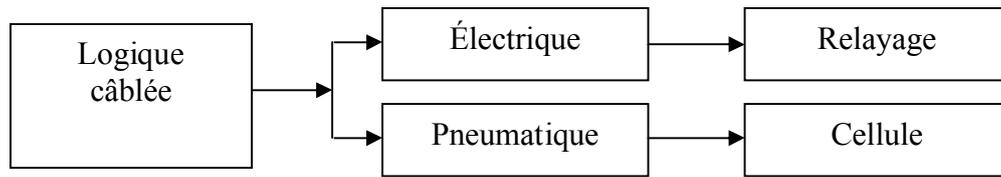


Figure II.2 : Représentation de la logique câblée.

IV.1.2.2. Logique programmée : qui correspond à une démarche séquentielle, c'est-à-dire : une seule opération exécutée à la fois.

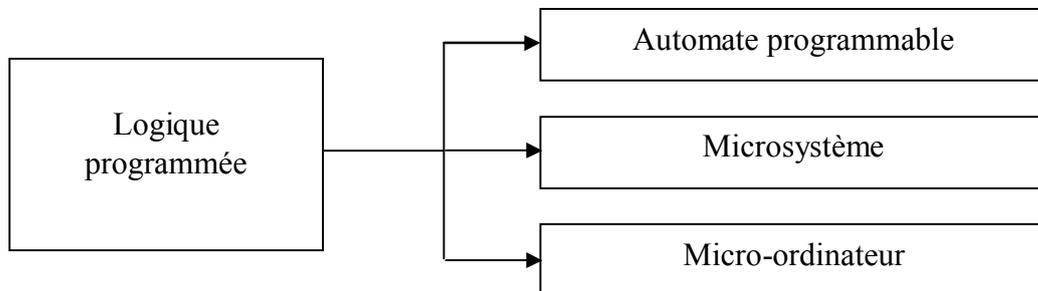


Figure II.3 : Représentation de la logique programmée

IV.1.2.3. Technologie câblée :

Le fonctionnement de l'installation est défini par câblage entre les différents éléments.
Une modification de ce fonctionnement impose une modification du câblage.

IV.1.2.4. Technologie programmée

Le fonctionnement de l'installation est défini par un programme exécuté de manière cyclique par un processeur.

Un changement de fonctionnement consiste à modifier le programme sans avoir à toucher aux raccordements des capteurs et des prés actionneurs.

IV.1.3. partie relation

La partie relation est composée de différent pupitre et terminaux, celle-ci assure la communication entre l'utilisateur et les organes à commander et permet de visualiser et de contrôler le système automatisé.

IV.2. Schéma de structure d'un système automatisé [3]

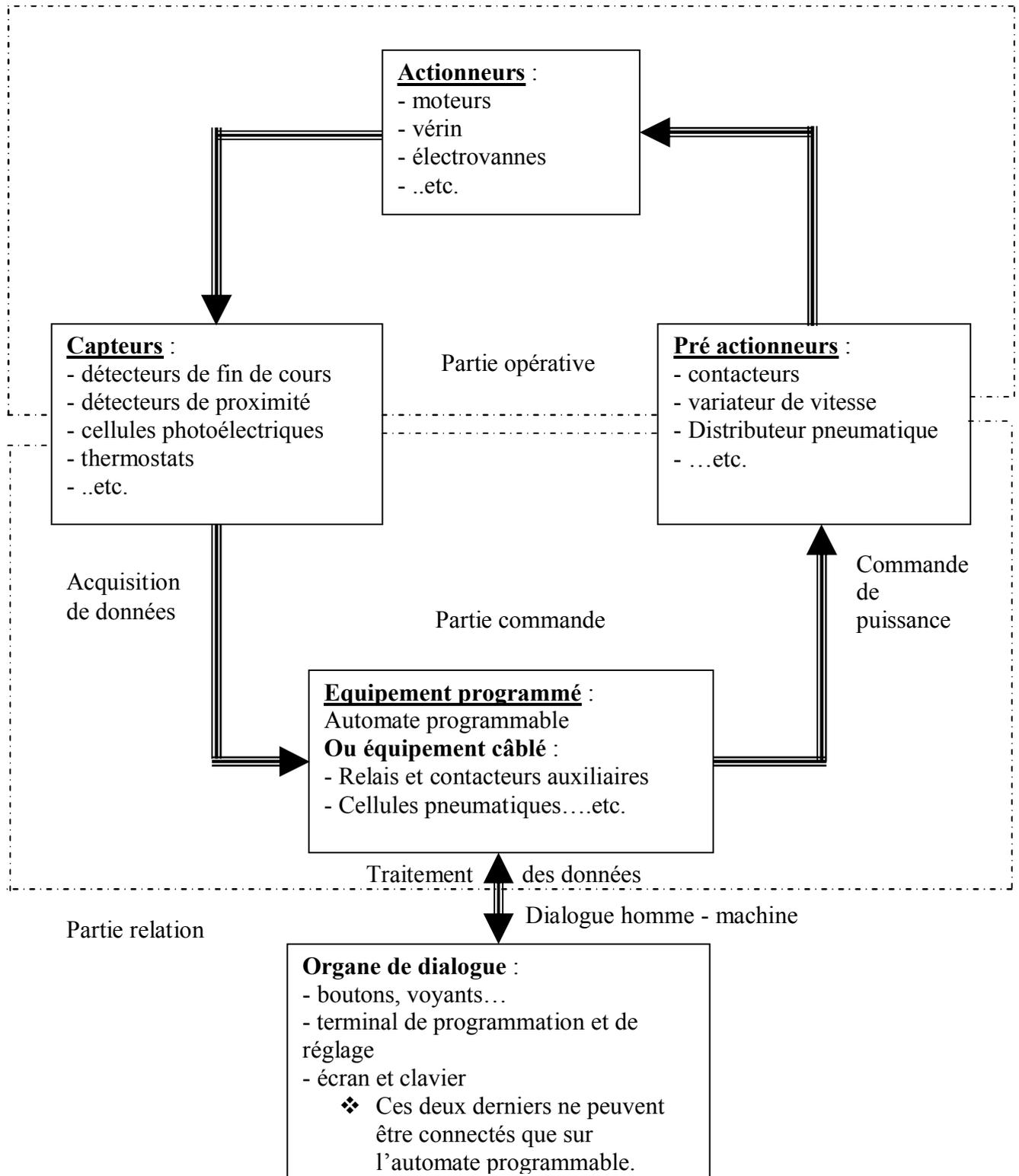


Figure II.4 : Schéma de structure d'un système automatisé

V. Architecture interne d'un API [7]

La conception modulaire des API permet une grande souplesse de configuration pour les besoins de l'utilisateur.

Un API est essentiellement constitué de 4 modules.

- l'alimentation
- l'unité centrale
- module entrée/sortie
- terminal de programmation

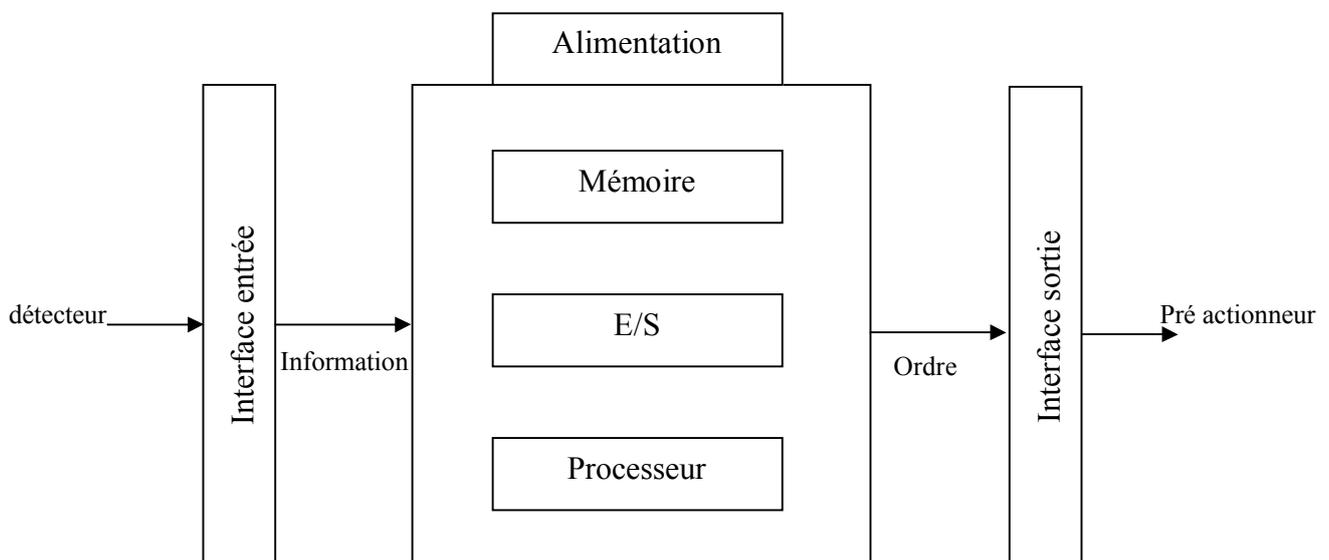


Figure II.5 : Schéma de structure interne de l'automate.

V.1. Alimentation

L'API est alimenté à partir du réseau alternatif (220), la CPU (processeur) et tout les autres composants utilisent une tension continue de 24V. Le module d'alimentation convertit la tension du réseau en tension de service.

V.2. Unité centrale

L'unité centrale est le regroupement du processeur et de la mémoire centrale.

V.2.1. le processeur : il comporte :

- une unité logique (UL) qui traite les opérations logiques ET, OU et négation

- une unité arithmétique et logique (UAL) qui traite les opérations de temporisation, de comptage et de calcul.
- Un accumulateur qui est un registre de travail dans lequel se range une donnée ou un résultat.
- Un registre d'instruction qui contient, durant le temps de traitement l'instruction à exécuter.
- Un décodeur d'instruction qui décode l'instruction à exécuter en y associant les microprogrammes de traitement.
- Un compteur programme ou un compteur ordinal qui contient l'adresse et la prochaine instruction à exécuter et gère ainsi la chronologie de l'exécution des instructions du programme.

V.2.2. La zone mémoire : la zone mémoire permet de :

- Recevoir les informations issues des capteurs d'entrée
- Recevoir les informations générées par le processeur et destinées à la commande des parties (valeurs des compteurs, des temporisations.....)
- Recevoir et conserver les programme du processeur.

V.2.2.1. Actions possibles sur une mémoire

- Ecrire pour modifier le contenu d'un programme
- Effacer pour faire disparaître les informations qui ne sont plus nécessaires
- Lire pour en lire le contenu d'un programme sans le modifier.

V.2.2.2. Technologie des mémoires

- RAM (read acces memory) mémoire vive dans laquelle on peu lire, écrire et effacer, elle contient le programme.
- ROM (read only memory) mémoire morte accessible uniquement en lecture
- EPROM mémoire morte reprogrammable, effacement au rayon ultra violet
- EEPROM mémoire morte reprogrammable, effacement électrique.

V.3. Les modules entrées/sorties

V.3.1. Module TOR : le principe de raccordement consiste à envoyer un signal électrique vers l'entrée choisie de l'automate dès que l'information est présente. Il permet donc d'effectuer une lecture de l'état logique des capteurs qui lui sont associés.

Un module d'entrée est principalement défini par sa modularité (nombre de voies) et les caractéristiques électriques (alimentation,...), ce module permet de raccorder à l'automate différents capteurs logiques (boutons poussoirs, de fin de cours...).

V.3.1.1. Les modules de sorties TOR : le principe de raccordement consiste à envoyer un signal électrique vers les pré actionneurs connectés à la sortie choisie de l'automate, dès que l'ordre est émis. Ce module permet à l'automate d'agir sur les actionneurs. Il réalise la correspondance entre l'état logique et le signal électrique.

V.3.2. Les modules d'entrées analogiques : ils permettent de gérer les grandeurs analogiques en faisant varier un code numérique au sein du module, ces modules convertissent les signaux analogiques issus du processeur en valeurs numériques par un convertisseur analogique/numérique.

V.3.3. Les modules de sorties analogiques : ces modules assurent la conversion numérique/analogique. L'intensité ou la tension sont proportionnelles à la valeur numérique. La conversion est assurée par des convertisseurs numériques/analogiques.

VI. Terminal de programmation : l'API doit communiquer avec :

- Le personnel d'étude et de réalisation des premières mise en œuvre (édition programme, transfert, sauvegarde, ...)
- Le personnel de mise au point et de maintenance de réalisation des opérations sur le système (forçage, visualisation de l'état, modification des paramètres, temporisation, compteurs...)
- Une console : elle est utilisée sur site, elle comporte un clavier, un écran de visualisation et le langage de programmation.
- Un micro-ordinateur avec un logiciel d'assistance à la programmation : il comprend plusieurs modules pour permettre l'édition, l'archivage, la mise au point des applications.

VII. Principe de fonctionnement [1]

L'automate programmable reçoit les informations relatives à l'état du système, puis commande les prés actionneurs suivant le programme inscrit dans sa mémoire.

Le traitement se fait en quatre étapes :

- **Phase 1** : gestion du système : autocontrôle de l'automate
- **Phase 2** : acquisition des entrées : prise en compte des informations des modules d'entrées et écriture de leurs valeurs dans la zone de données (RAM)
- **Phase 3** : traitement des données : lecture du programme (situé dans la RAM) par l'unité de traitement, lecture des variables (RAM données), traitement et écriture
- **Phase 4** : émission des ordres : lecture des variables de sorties dans la RAM de données et les transférer vers le module de sortie.

VIII. Choix de l'automate [13]

Pour l'automatisation d'une station (machine), il existe une partie primordiale qu'est le choix d'un API qui est lié à certains critères importants à savoir :

- La capacité du processeur au traitement des données.
- La nature du traitement.
- Le nombre des entrées sorties.
- La nature des ses entrées sorties (analogiques ou numériques).
- Le dialogue (la console détermine le langage).
- La nature de la station (si elle nécessite une flexibilité de changement de processus de travail).
- La fiabilité et la robustesse (par rapport au milieu et à la température).

La station étudiée permet le traitement de différents types de pièces, donc différents cycles de travail ; ce qui nous rend la programmation d'un tel procédé très délicat. Pour palier à ce problème le choix est porté sur un automate télémechanique possédant un pupitre de commande, ce dernier est muni d'une carte programmable permettant ainsi d'intervenir sur le changement du cycle sans pour autant toucher ni à l'automate ni à la machine.

IX. Présentation de l'automate TSX 57 203 [8]

Une large gamme de processeurs, de performances et de capacités croissantes est proposée pour répondre au mieux aux différents besoins des utilisateurs.

Les processeurs TSX57 sont intégrables sur racks TSXRKY. Ils gèrent l'ensemble d'une station automate constituée de modules d'entrées/sorties TOR, de modules analogiques et de

module métier (comptage, commandes d'axes, commandes pas à pas, communication,...) qui sont répartis sur un ou plusieurs racks.



Figure II.6 : Automate TSX 57203.

Chaque processeur intègre :

- une mémoire RAM interne sauvegardée qui peut recevoir le programme application et qui peut être étendue par une carte d'extension mémoire PCMCIA.
- Un horodateur.
- 2 prises terminales (TER et AUX) qui permettent de raccorder simultanément plusieurs équipements (terminal de programme, terminal de dialogue opérateur,...).
- Un emplacement pour carte de communication PCMCIA.

Le programme application est réalisé à partir du logiciel PL7 junior ou PL7 pro qui propose :

- 4 langages de programmation.
- Une structure logicielles multi-tâches (tâche maître et rapide, traitement événementiels).
- La modification d'un programme.

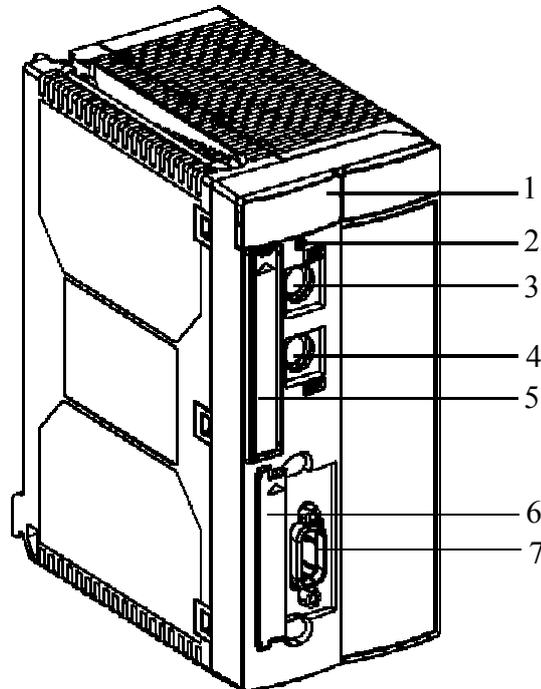
IX.1. Description physique du processeur TSX 57203

Figure II.7 : *Processeur TSX 57203.*

- 1- bloc de visualisation comprenant 4 voyant.
Voyant RUN, ERR, I/O, TER.
- 2- bouton RESET à pointe de crayon provoquant un démarrage à froid de l'automate lorsqu'il est actionné.
- 3- Prise terminale TER permettant le raccordement d'un périphérique automatique ou non.
- 4- Prise de dialogue opérateur AUX permettant le raccordement d'un périphérique auto-alimenté.
- 5- Emplacement pour carte d'extension au format PCMCIA types 1 en l'absence de carte mémoire, cet emplacement est équipé d'un cache qu'il est obligatoire de maintenir en place, son extraction provoque l'arrêt du processeur.

- 6- Emplacement pour carte de communication au format PCMCIA type3 pour raccordement au processeur d'une voie de communication.
- 7- Connecteur SUBD9 points pour raccordement du bus FIPIO maître.

❖ Rappel catalogue

Rack TSX RKY 12EX	8
Rack TSX RKY 4 EX/6EX/8EX	16
Emplacement modules	87 [111]
E/S TOR en rack	1024
E/S analogiques en rack	80
Voies métier	24
Boucle de régulation	30
Mémoire interne	48 (K mot)
Extension mémoire	160 (K mot)

Pour les emplacements des modules, l'automate TSX 57203 comprenant 87 emplacements avec 8 racks TSXRKY 12 EX et 111 emplacements avec 16 racks TSXRKY 4EX/6EX/8EX.

Les voies métier sont les voies de comptage, commande d'axes, commande pas à pas, communication,...

La capacité mémoire se mesure en K mots, chaque mot est composé de 16 bits.

IX.2. Implantation d'un module processeur double format

Le module processeur TSXP57 double format s'implante sur un rack TSX RKY-, en position 00 et 01 ou 01 et 02 selon le type de module d'alimentation utilisé (format standard ou double format).

- si le rack est équipé d'un module d'alimentation au format standard TSXPSY 161/2600, le processeur s'installe en position 00 et 01 (positions préférentielles) ou en position 01 et 02, dans ce dernier cas la position 00 est indisponible.
- Si le rack est équipé d'un module d'alimentation double format TSXPSY 3610/5500/5520, 8500, le processeur s'installe en position 01 et 02.

Le montage du module processeur sur le rack doit obligatoirement s'effectuer avec l'alimentation du rack HORS TENSION.

L'automate qui gère l'installation de zingage comporte un rack équipé d'un module d'alimentation au format standard et d'un module processeur double format.

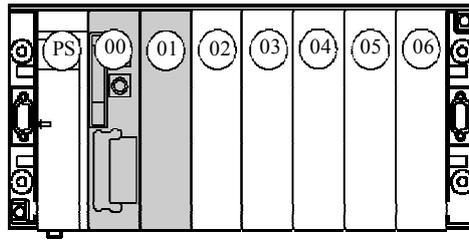


Figure II.8: Rack TSX RKY.

IX.3. Mémoire interne

Cette mémoire reçoit l'application (données, programme et constantes) et sa capacité varie selon le type de processeur.

Elle est de 48 K mots sur processeur TSX 57203.

Si la taille de l'application est supérieur à celle de la RAM, il est possible d'étendre la mémoire par une carte d'extension mémoire PCMCIA, dans ce cas le programme est données dans la mémoire RAM interne.

La mémoire RAM interne peut être sauvegardée par une pile optionnelle située dans le module alimentation. La sauvegarde de l'application n'est effective que si les modules alimentation et processeur restent en place sur le rack.

IX.3.1. Emplacement pour carte d'extension mémoire PCMCIA

Cet emplacement, en face avant du processeur, permet de recevoir une carte mémoire optionnelle au format PCMCIA.

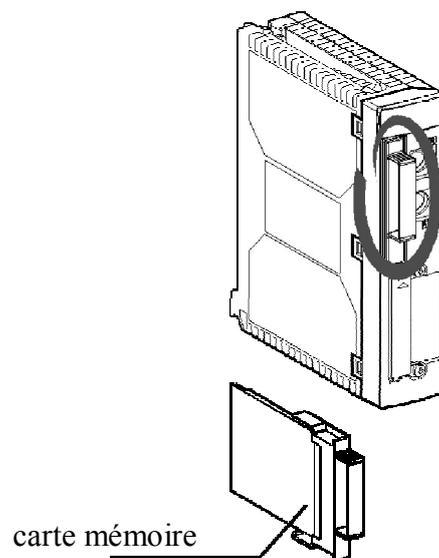


Figure II.9 : Carte d'extension mémoire PCMCIA

3 familles de cartes sont proposées :

IX.3.1.1. Carte mémoire standard : il existe deux types :

- RAM sauvegardée pour les phases de création et de mise au point du programme application. La sauvegarde est réalisée par une pile amovible contenue dans la carte.
- FLASH EPROM lorsque le programme application est opérationnel (mise au point terminée).

IX.3.1.2. Cartes mémoire de type BACKUP

Pour charger le programme en RAM interne sans l'utilisation d'un terminal. Une telle carte nécessite d'être au préalable chargée avec le programme application dont la taille doit être inférieure à 32 Kmots.

IX.3.1.3. Carte mémoire de type application + fichier

Elles disposent en plus de la zone de stockage application traditionnelle :

- D'une zone fichier permettant d'archiver des données par programme.
- D'une zone permettant l'archivage de la base de symbole de l'application, cette base de symboles est compressée pour tenir sans aucune contrainte dans la zone qui lui est allouée.

❖ **Référence des carets d'extension mémoire PCMCIA de type standard et backup compatible avec le processeur TSX 57203**

Référence	Type	Capacité
TSX MRP 032P	RAM	32 K16
TSX MRP 064P	RAM	64K16
TSX MRP 0128P	RAM	128 K16
TSX MRP 0256P	RAM	256 K16
TSX MFP 032P	FLASH EPROM	32 K16
TSX MFP 064P	FALSH EPROM	64 k16
TSX MFP 0128P	FLASH EPROM	128 K16
TSX MFP BAK 032P	RAM/BACKUP	32 K16

❖ **Référence des cartes mémoire de type application + fichier ou symboles compatibles avec le processeur TSX 57203**

référence	type	capacité		
		application	Zone fichier	Zone symbole
TASX MRP 232P	RAM	32 K16	128 K16	/
TSX MRP 264P	RAM	64K16	128K16	/
TSX MRP 2128P	RAM	128KP	128K16	128K16
TSX MRP 3256P	RAM	256K16	640K16	128K16
TSX MRP 3384P	RAM	384K16	640K16	/
TSX MRP 0512P	RAM	512K16	/	256K16
TSX MFP 232P	FALSH EPROM	32K16	128K16	/
TSX MFP 264P	FLASH EPROM	64K16	128K16	/

❖ **Bouton reset :**

Une action sur un bouton poussoir à pointe de crayon, provoque un démarrage à froid de l'application :

- processeur en fonctionnement : démarrage en STOP ou en RUN suivant configuration.
- Processeur en défaut : démarrage forcé en STOP.

❖ **Fonction RUN/STOP :**

Elle permet la mise en exécution ou l'arrêt du programme application, depuis un terminal de programmation ou une entrée TOR définie en configuration. La mise en STOP depuis cette entrée physique est prioritaire par rapport à la mise en RUN depuis un terminal.

❖ **Horodateur :**

L'horodateur intégré au processeur gère la date et l'heure courante ainsi que la date et l'heure du dernier arrêt de l'application. Cette gestion s'effectue même lorsque le processeur est hors tension, à la condition qu'il soit monté sur le rack avec le module d'alimentation, équipé d'une pile de sauvegarde.

IX.4. Diagnostic à partir des voyants de visualisation

4 voyants situés en face avant du processeur permettent le diagnostic rapide sur l'état de l'automate :

- **RUN** (vert): état de l'application (**allumé** : fonctionnement normal ; **clignotant** : automate en STOP ou en défaut logiciel bloquant ; **éteint** : automate non configuré, application absente, non valide, incompatible avec le type de processeur ou automate en erreur, défaut processeur ou système.).
- **ERR** (rouge): défauts processeur ou carte mémoire ou carte de communication PCMCIA (**allumé** : automate en erreur, défaut processeur ou défaut système ; **clignotant** : automate non configuré, application absente, non valide ou incompatible avec le type de processeur, automate en défaut logiciel bloquant, défaut pile carte mémoire, défaut bus X (1) ; **éteint** : fonctionnement normal).
- **I/O** (rouge) : défaut d'E/S (**allumé**: défaut entrées sorties, provenant d'un module ou d'une voie ou défaut de configuration, **clignotant** : défaut bus X (1) ; **éteint** : fonctionnement normal).
- **TER** (jaune) : il signale l'activité sur la prise terminale (**clignotant** : échange en cours sur la prise terminal).



Figure II.10 : voyants de visualisation du processeur.

IX.5. Présentation générale des modules entrées/sorties TOR

IX.5.1. Description physique

➤ **module à connecteurs HE10**

- 1 bloc visualisation.
- 2 Connecteur HE10 protégés par un capot. Ces connecteurs permettent le raccordement des capteurs et pré actionneurs soit directement via des torons pré câblés ; soit au travers d'embases de raccordement TELEFAST2.

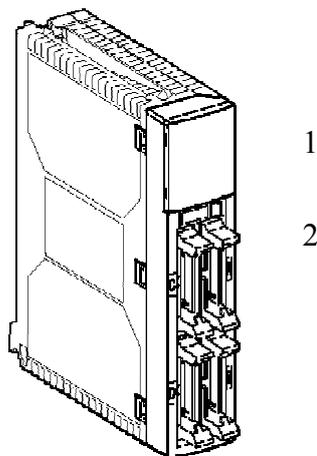


Figure II.11 : Module à connecteurs HE10.

❖ Rappel catalogue

Module d'entrée TSX DEY...

référence	modularité	Connect	Tension	logique	filtrage
DEY 08 D2	8	BORNIER	24Vcc	positive	4ms
DEY 16 D2	16	BORNIER	24Vcc	positive	4ms
DEY 16 D3	16	BORNIER	48Vcc	positive	4ms
DEY16 A2	16	BORNIER	24Vcc 24Vca	Négative /	10ms 50/60ms
DEYA3	16	BORNIER	48Vca	/	50/60ms
DEY16A4	16	BORNIER	115Vca	/	50/60MS
DEY 16A5	16	BORNIER	230Vca	/	50/60ms
DEY 16FK	16	HE10	24Vcc	POSITIVE	7,5ms
DEY 32D2K	32	HE10	24Vcc	POSITIVE	4ms
DEY 32D3K	32	HE10	48Vcc	POSITIVE	4ms
DEY 64D2K	64	HE10	24Vcc	POSITIVE	4ms

Module de sorties TSX DSY...

Référence	modularité	connect	tension	Courant	logique	Temps de réponse
DSY 08 T2	8	BORNIER	24Vcc	0,5A	POS	1,2ms
DSY 08 T22	8	BORNIER	24Vcc	2A	POS	0,2ms
DSY 08 T31	8	BORNIER	48Vcc	1A	POS	0,2ms
DSY 16 T2	16	BORNIER	24Vca	0,5A	POS	1,2ms
DSY 16 T3	16	BORNIER	48Vca	0,5A	POS	1,2ms
DSY 08 R5	8	BORNIER	24Vcc 24..240Vca	3A	-	0 v 1 < 8ms 1 v 0 < 10ms
DSY 08 R4D	8	BORNIER	24..110Vcc	5A	-	0 v 1 < 10ms 1 v 0 < 15ms
DSY 08 R5A	8	BORNIER	24..48Vcc 24..240Vca	5A	-	
DSY 16 R5	16	BORNIER	24Vcc 24..240Vca	3A	-	
DSY 08 S5	8	BORNIER	48..220Vca	2A	-	0 v 1 < 10ms 1 v 0 < 10 ms
DSY 16S4	16	BORNIER	24..110Vca	1A	-	
DSY 16S5	16	BORNIER	48..220Vca	1A	-	
DSY32 T2K	32	HE10	24Vcc	0,1A	POS	1,2ms
DSY 64 T2K	64	HE10	24Vcc	0,1A	POS	1,2ms

Les modules DSY 08 T2, DSY 08 T22, DSY 08 T31, DSY 16 T2, DSY16 T3, DSY 32 T2K, DSY 32 T2K, DSY 64 T2K ont des sorties à transistor.

Les modules DSY 08 R5, DSY 08 R4D, DSY 08 R5A, DSY 16R5, ont des sorties à relais.

Les modules DSY 08 S5 et DSY 16 S4 ont des sorties triacs.

Modules mixtes d'entrées sorties

Les modules TSX MY 28FK/28RFK possèdent simultanément 16 entrées et 12 sorties (à transistor).

modularité	connect	tension	courant	logique	filtrage	Temps de réponse
16 entrées	HE10	24Vcc	-	positive	7,5ms	/
12 sorties	HE10	24Vcc	0,5A	positive	-	0,5ms

Les sorties intègrent un dispositif de protection contre les courts circuits et les surcharges, et contrent les inversions de polarité pour les modules à courant continu.

IX.5.2. Implantation et montage

Les modules d'entrée/sortie TOR, se positionnent indifféremment sur un rack TSXRKY..

Le montage et le démontage d'un module peuvent s'effectuer rack sous tension, mais il est obligatoire de couper la tension des capteurs et des pré-actionneurs et de déconnecter le bornier.

IX.5.3. Entrée à générateur de courant

Les entrées à courant continu 24Vcc et 48Vcc sont du type « générateur de courant ». Quelle que soit la tension d'entrée supérieure à 11V (pour les entrées 24Vcc) ou 20 V (pour les entrées 48Vcc), le courant d'entrée est constant.

IX.5.4. Protection des sorties statiques à courant continu

Toutes les sorties statiques protégées, sont équipées d'un dispositif qui permet, lorsqu'une sortie est active, de détecter l'apparition d'une surcharge ou d'un court circuit. Un tel défaut provoque la désactivation de la sortie (disjonction) et la signalisation du défaut (le voyant de la voie en défaut clignote et le voyant I/O du processeur s'allume). Pour réactiver une sortie disjonctée, il est nécessaire de la réarmer.

IX.5.5. Réarmement des sorties

Le réarmement d'une sortie disjonctée peut être automatique ou commandé, selon le choix effectué en configuration. Le réarmement est demandé pour les sorties statiques à courant continu ou, pour les sorties à relais et triacs protégées par un fusible interchangeable. Il s'effectue par groupe de 8 voies, mais reste sans effet pour les voies non activées ou sans défaut.

- Si le réarmement est automatique, il est exécuté par le module toutes les 10s. jusqu'à la disparition du défaut qui permet de le prendre en compte.
- Si le réarmement est commandé par le programme application ou via une console, il sera pris en compte si le défaut a disparu. Il est nécessaire d'attendre au minimum 10s entre deux réarmements.

IX.5.6. Repli des sorties

Lors d'un défaut bloquant, toutes les sorties d'un module sont positionnées dans un état déterminé par l'utilisateur en configuration : maintien dans l'état, repli à 0 ou repli à 1.

IX.5.7. Partage des entrées/sorties

Chaque module est découpé fonctionnellement en groupes de 8 voies qui peuvent être affectés à des tâches différentes de l'application (par exemple, pour un module à 16 voies, les voies 0 à 7 peuvent être affectées à la tâche MAST et les voies 8 à 15 à la tâche FAST).

Les voies d'un même groupe possèdent les modes de marche et la gestion de fonctionnalités communs (repli et réarmement des sorties).

IX.5.8. Filtrage programmable sur les entrées

Les modules d'entrée permettent de configurer le temps de filtrage des entrées en 0 et 7,5 ms (4ms par défaut).

Pour éviter le temps de rebonds lors de la fermeture des contacts mécaniques, il est conseillé d'utiliser un temps de filtrage supérieur à 3ms.

IX.5.9. Mémorisation d'état

Certains modules permettent, à travers la mémorisation d'état, la prise en compte d'impulsions très courtes et de durée inférieure à un temps de cycle automate. Le changement d'état de l'entrée est pris en compte pour être traité au cycle suivant dans la tâche.

Le temps qui sépare l'arrivée de deux impulsions sur une même entrée doit être au moins égal à deux temps de cycle.

La durée minimale de l'impulsion doit être supérieure au temps de filtrage configuré. Ces modules permettent de configurer jusqu'à 16 entrées qui permettent la prise en compte d'événements et leur traitement immédiat.

IX.5.10. Contrôle des courts-circuits et surcharges

Les modules d'entrées statiques sont équipés d'un dispositif qui contrôle l'état de la charge. Le court-circuit ou la surcharge d'une ou plusieurs sorties provoquent l'apparition d'un défaut et la disjonction des sorties incriminées.

IX.5.11. Contrôle de la tension des capteurs et des pré-actionneurs

Tous les modules d'entrées et de sorties statiques sont équipés d'un dispositif qui contrôle que la tension d'alimentation des capteurs, des pré-actionneurs et du module est suffisante pour garantir le bon fonctionnement des voies d'entrées et de sorties. Si cette tension devient inférieure à un seuil, un défaut est signalé.

IX.6. Les principaux blocs fonction

- Bloc fonction temporisateur type travail.
- Bloc fonction temporisateur type repos.
- Bloc fonction temporisateur type travail et repos.
- Bloc fonction temporisateur avec deux valeurs.
- Bloc fonction temporisateur type travail/repos avec deux valeurs.
- Bloc fonction monostable redéclenchable.
- Bloc fonction monostable temporisé, non redéclenchable.
- Bloc fonction monostable avec deux valeurs.
- Bloc fonction oscillateur.
- Bloc fonction compteur à deux seuils.
- Bloc fonction compteur à un seuil avec monostable.
- Bloc fonction intervalomètre permettant de mesurer un temps ou une longueur.
- Bloc fonction Burst permettant de générer un nombre défini de période d'oscillations
- Bloc fonction PWM permettant de générer une oscillation continue à fréquence fixe mais à rapport cyclique variable.
- Bloc fonction détection de sous vitesse.
- Bloc fonction surveillance de vitesse.
- Bloc fonction commande/contrôle permettant de commander une action et de vérifier qu'au bout d'un certain temps elle s'est bien effectuée.
- Bloc fonction commande pendant un nombre de points de comptage (positionnement simple).
- Bloc fonction signalisation de défaut.
- Bloc fonction bascule D, mémorisation de front.
- Bloc fonction bascule T, division par 2.

IX.7. Raccordement des modules à connecteurs HE10

IX.7.1. Toron pré-câblé de 20 fils

Il permet le raccordement fil à fil des entrées/sorties vers capteurs, pré-actionneurs ou borne de raccordement

Borne / Fil	Borne / Fil
1 blanc	2 marron
3 vert	4 jaune
5 gris	6 rose
7 bleu	8 rouge
9 noir	10 violet
11 gris-rose	12 rouge-bleu
13 blanc-vers	14 marron-vert
15 blanc-jaune	16 jaune-marron
17 blanc-gris	18 gris-marron
19 blanc-rose	20 rose-marron

IX.7.2. Nappe de raccordement

Elle permet le raccordement des entrées /sorties vers l'interface de raccordement et d'adaptation à câble rapide TELEFAST2.

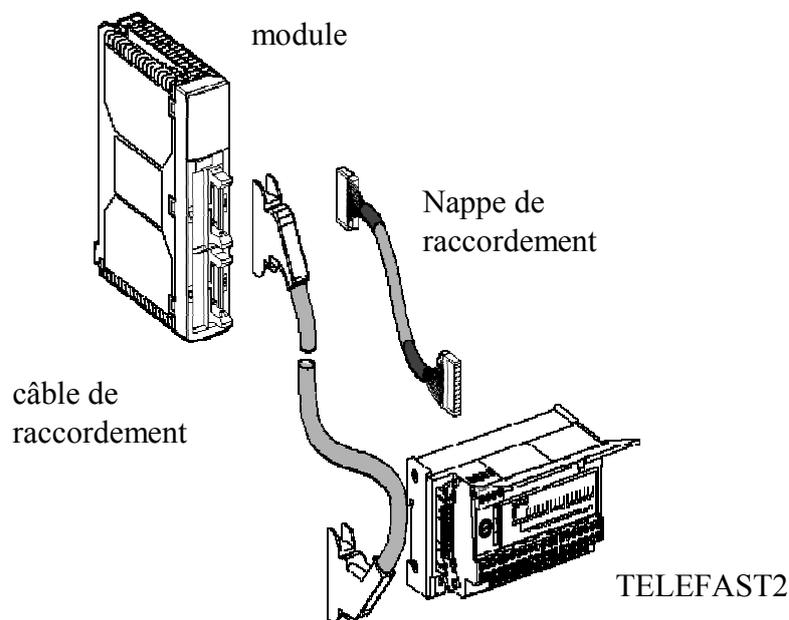


Figure II.12 : Nappe de raccordement.

IX.8. Diagnostic des modules entrée/sortie

Les voyants d'état, en face avant du module, permettent un diagnostic rapide de celui-ci : Trois voyants d'état du module renseignent sur le mode de fonctionnement du module.

-RUN (vert) : état du module (allumé : marche normale ; éteint : module en défaut).

-ERR (rouge) : défauts internes (allumé : module en panne ; clignotant : défaut de communication).

-I/O (rouge) : défauts externes (allumé : surcharge, court-circuit, défaut tension capteurs/pré-actionneurs ; clignotant : défaut bornier).

- 8, 16 ou 32 voyants d'état des voies renseignent sur l'état de chaque entrée ou sortie (allumé : voie à l'état 1 ; clignotant : voie en défaut, surcharge ou court-circuit ; éteint : voie à l'état 0).

De plus, le voyant **+32**, présent sur les modules à 64 voies, indique le groupe de voies qui est visualisé (éteint : voies 0 à 31 ; allumé : voies 32 à 63).

Un bouton poussoir (présent uniquement sur les modules à 64 voies) permet de sélectionner le groupe de voies.

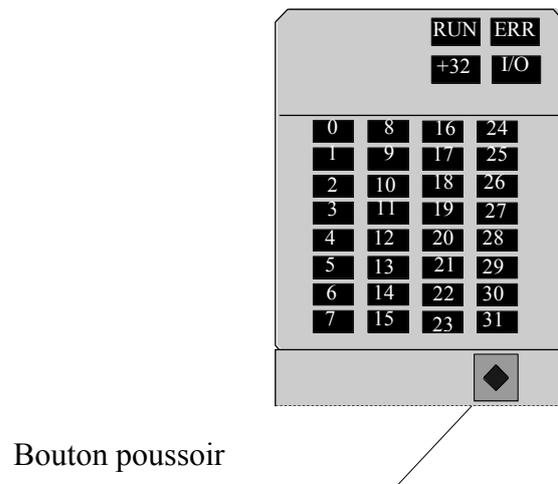


Figure II.13 : *Voyants de visualisation.*

IX.9. Représentation d'un module d'entrée de modularité 64

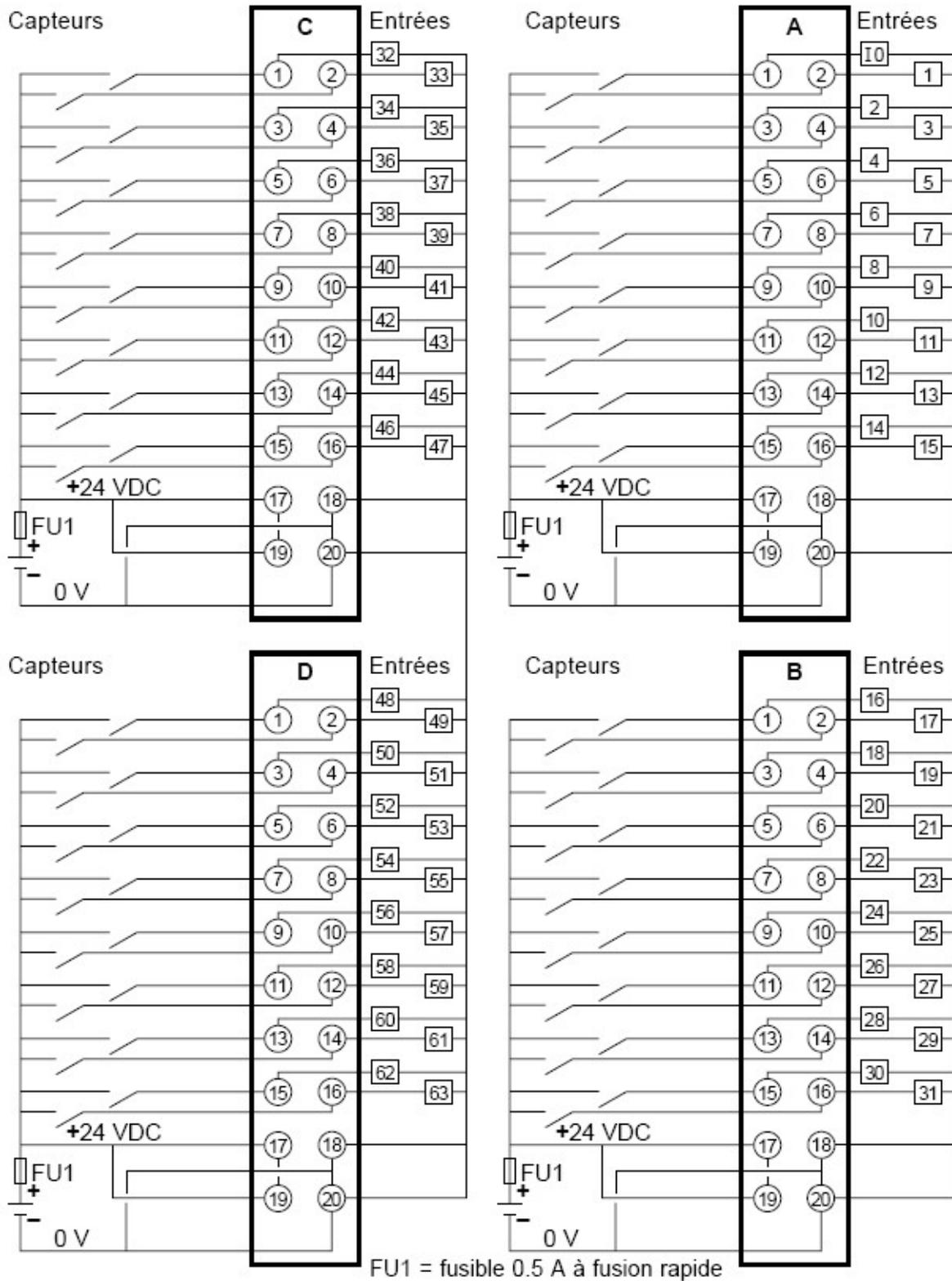


Figure II.14 : Module d'entrée TSX DEY 64D2K.

IX.10. Représentation d'un module de sortie de modularité 64

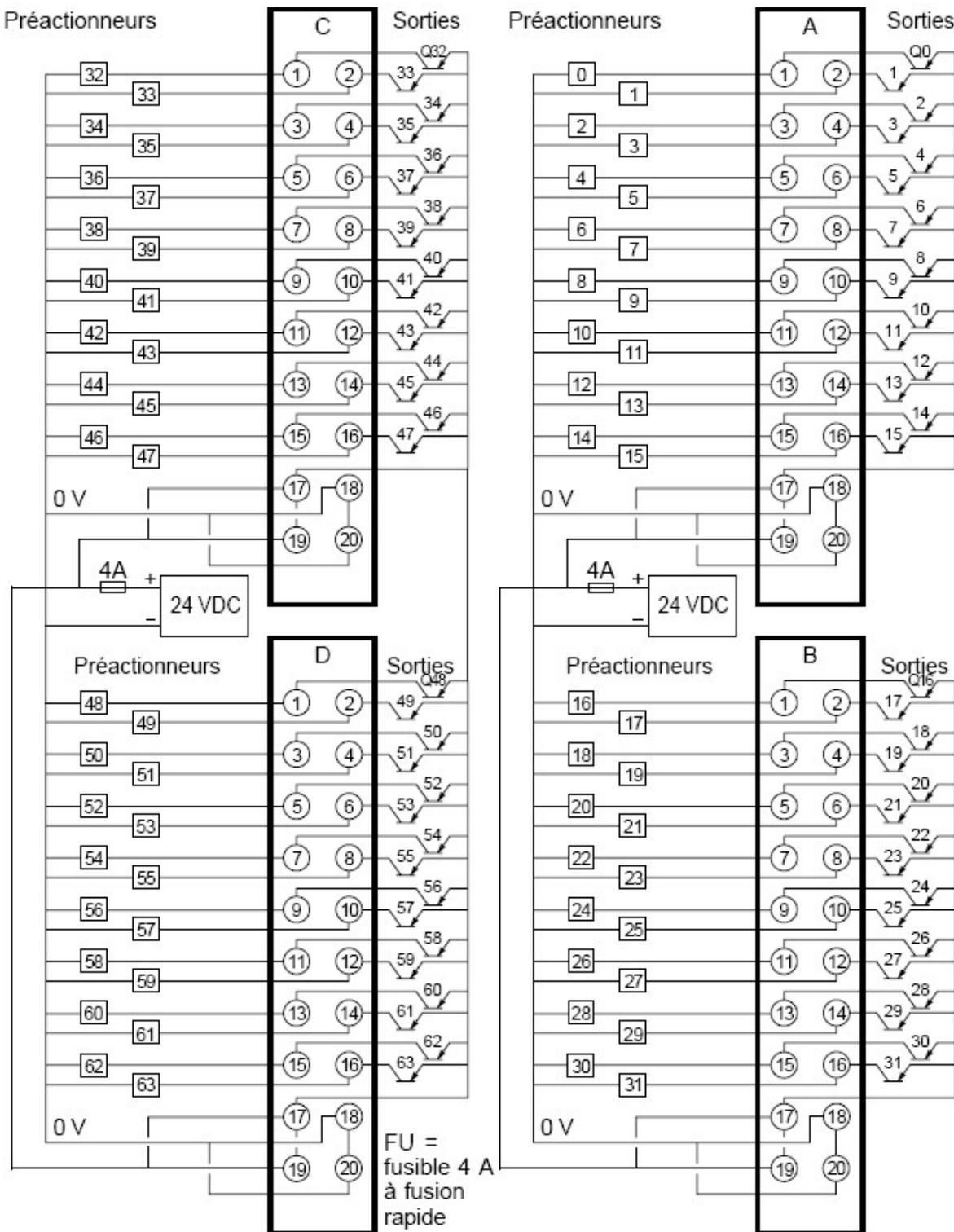


Figure II.15 : Module de sortie TSX DSY 64T2K.

X. Présentation du pupitre de commande [14]

SATS 2000 est un automatisme qui permet le pilotage d'une ligne de traitement de surface. Il gère ainsi le déplacement d'un ou de plusieurs mobiles (portique et transfert), assure la commande de toutes les parties du processus de l'installation et permet de visualiser en temps réel l'état des divers éléments de la chaîne.

Ce pupitre est un terminal opérateur (nommé XBT) pour les applications du système d'automatisme. Par son intermédiaire, l'utilisateur pourra effectuer les opérations suivantes :

- Saisie des tables et des paramètres des ponts.
- Saisie des paramètres de la cuve.
- Saisie des paramètres de la chaîne.
- Gestion des alarmes et des défauts.
- Visualisation de l'état de la chaîne.

X.1. Architecture du pupitre

Les éléments du système sont organisé comme suit :

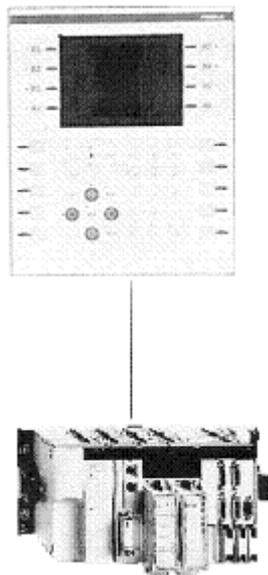


Figure II.16 : *Eléments d'un système automatisé.*

L'XBT est connecté à un automate par le réseau TELEMECANIQUE UNITELWAY c'est cet automate qui contrôle l'ensemble de l'installation.

X.2. Description fonctionnelle générale

Le système de pilotage est composé de trois niveaux :

- Niveau 1- c'est le niveau XBT : le terminal opérateur est l'outil de suivi en temps réel du fonctionnement de la ligne de traitement de surface. par ailleurs, il permet le paramétrage de l'installation ainsi que la commande de ces composants.
- Niveau 2- c'est le niveau automate .celui-ci gère le pilotage et la synchronisation des différents postes robotisés et manuels. Il acquière les états et les alarmes pour les remonter au niveau 3.
- Niveau 3- c'est le niveau des postes de travail. Ceux-ci sont deux types : manuels avec ou sans pupitres, ou automatiques (robots ou transferts). Les postes manuels sont de trois types différents : postes de chargement des supports, postes de chargement des pièces, postes de déchargement des pièces.

L'installation ne pourra fonctionner correctement sans l'XBT qui assure les fonctions suivantes :

- surveillance et suivi des moyens de production.
- acquisition des différents traitements réalisables sur les pièces.
- gestion des alarmes et des défauts.

X.3. principes généraux [5]

Les utilisateurs de SATS 2000 accèdent aux différents écrans (également appelés vues) en utilisant les touches situées de part et d'autre du terminal. Celles-ci sont de deux natures différentes :

- les touches R1 à R10 sont des touches dynamiques dont les fonctions changent suivant la vue sur laquelle se trouve l'utilisateur. Dans ce cas des indications apparaissent à l'écran pour désigner l'effet de chacune d'entre elles.
- les touches F1 à F12 possédant quand à elles toujours les mêmes fonctions quelque que soit l'écran sur lequel se trouve l'utilisateur, il faut noter au passage que toutes ces touches n'ont pas forcément de fonctions attribuées.

Les vues sont développées en utilisant une représentation fonctionnelle de type blocs et pavée qui se veut être la plus claire et la plus précise possible. Une attention est donnée à la facilité d'utilisation de l'application.

Au centre de l'écran, se trouvent les informations disponibles. Celles-ci sont fréquemment fournies dans des tableaux possédant le code couleur suivant :

- les cases bleues ne peuvent être modifiées.
- les cases jaunes peuvent être modifiées. (pour ce faire, il suffit d'appuyer sur la touche « Mod », de se positionner sur la case à modifier en se déplaçant à l'aide des touches « flèches », on peut alors entrer les valeurs souhaitées à l'aide du clavier numérique).

Sur les côtés, on trouve les indications concernant les touches R1 à R10 qui permettent d'accéder aux autres vues de l'application.

En bas de l'écran, enfin, se trouve un bandeau permettant de visualiser la dernière page alarme en cours. (Dans ce bandeau, il n'est possible de détecter au niveau de l'XBT qu'une seule alarme à la fois).

X.4. détail des touches de fonction

Voici détaillée ci-dessous, la représentation du terminal opérateur : les fonctions des principales touches y sont explicitées. Dans le tableau qui suit sont consignées les autres fonctions de l'XBT.

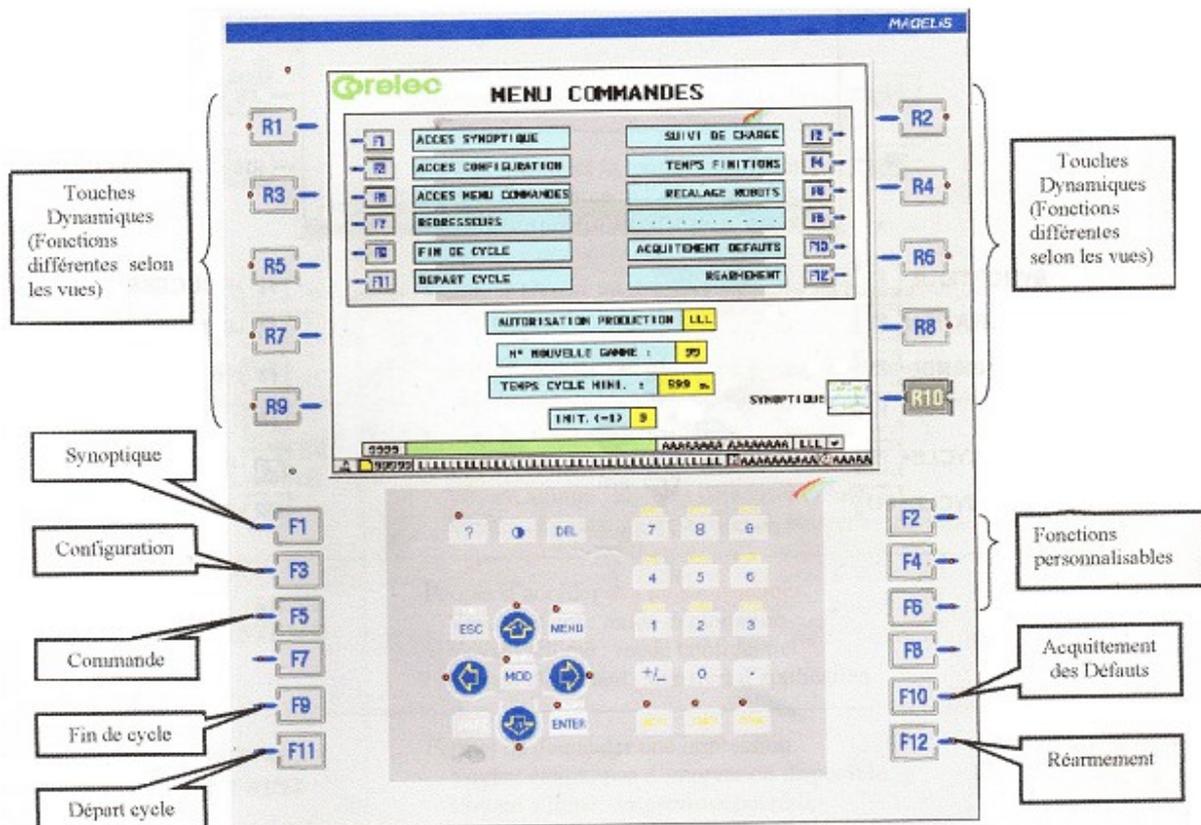


Figure II.17 : Représentation du terminal opérateur.

X.5. Détails des touches du pavé numérique

	<p>Permet de se déplacer dans une page ou dans une liste. Voyant éteint : touche inactive. Voyant allumé : déplacement possible. Voyant clignotant : possibilité de modifier une valeur dans une liste.</p>
ENTR	<p>Permet de valider un choix ou une saisie. Permet d'acquitter une alarme.</p>
MENU	Permet d'accéder à la page menu.
MOD	<p>Permet d'activer le mode modification. Permet également de faire un choix dans une liste.</p>
ESC	<p>Permet d'annuler la modification d'un paramètre. Permet de retourner à l'écran précédent. Permet de sortir d'un écran.</p>
DEL	Permet d'effacer le caractère à gauche du curseur.
SHIFT	Permet d'accéder aux fonctions grisées au dessus des touches.
SHIFT+ENTRER → ALARME	<p>Permet d'accéder à la liste des alarmes. Voyant éteint : pas d'alarme. Voyant allumé : alarmes déjà visualisées. Voyant clignotant : apparition de nouvelle alarme.</p>
SHIFT+MENU → SYSTEME	<p>Permet d'accéder aux pages systèmes. Voyant éteint : mode exploitation. Voyant allumé : mode confidentiel. Voyant clignotant : transfert d'application.</p>
SHIFT+MOD → PRINT	<p>Permet de demander une impression. Voyant éteint : pas d'impression disponible. Voyant allumé : impression possible. Voyant clignotant : impression en défaut.</p>

XI. Conclusion

L'automatisation d'un système quelconque nécessite non seulement la connaissance du processus décrit dans son cahier des charges mais aussi le dispositif qui permet l'automatisation. Généralement il s'agit d'un automate programmable.

Ce chapitre décrit les principales composantes d'un système automatisé ainsi que l'architecture détaillées de l'automate ; ceci permet d'optimiser la procédure et d'effectuer le choix de l'automate le plus adapté au bon fonctionnement du système à automatiser. Pour se faire il est primordial de modéliser le système ; le grafcet est un moyen de le faire.

Le chapitre suivant illustre le mode de fonctionnement d'une manière exacte et précise l'installation de zingage.

I. Introduction

Le GRAFCET (**g**raphe **f**onctionnel de **c**ommande **é**tapes de **t**ransitions) est un outil graphique de définition pour l'automatisme séquentiel ; c'est un diagramme fonctionnel qui permet de décrire graphiquement les différents comportements de l'évolution d'un automatisme séquentiel. Il est à la fois simple à utiliser et rigoureux sur le plan formel et constitue un outil de dialogue entre toutes les personnes collaborant à la conception, à l'utilisation ou à la maintenance de la machine à automatiser.

II. Eléments constituant un GRAFCET [6]

Le GRAFCET est composé d'étapes, de transition et de liaisons.

- une liaison ne peut être parcourue que dans un sens. A une extrémité d'une liaison, il y a une seule étape, à l'autre une transition.
- Une étape correspond à une phase durant laquelle on effectue une action pendant une certaine durée (même faible mais jamais nulle). L'action doit être stable, c'est-à-dire que l'on fait la même chose pendant toute la durée de l'étape. Une action peut être la composition de plusieurs actions ou l'opposé, l'inaction (étape dite d'attente).

Une étape est dite active lorsqu'elle correspond à une phase en fonctionnement c'est-à-dire qu'elle effectue l'action qui lui est associée (si non elle est désactivée).

- une transition est une condition de passage d'une étape à une autre, elle est logique (dans son sens vrai ou faux) et sans notion de durée. la condition est définie par une réceptivité qui est généralement une expression booléenne de l'état des capteurs.

III. Règles d'évolution

L'évolution (la modification) de l'état de l'automatisme est régie par cinq règles :

- règle 1

Situation initiale : se sont des étapes qui sont actives au début du fonctionnement qui est le moment où le système n'a pas besoin de se souvenir de ce qui s'est passé auparavant.

Les étapes initiales correspondent habituellement au comportement au repos du procédé.

Si l'automatisme n'est pas cyclique (état initial dépendant des entrées dès la mise en marche du procédé), alors il peut être utile de forcer des étapes initiales.

- **Règle 2**

Franchissement d'une étape : une transition est validée lorsque toutes les étapes immédiatement précédentes sont actives. Elle ne peut être franchie que lorsqu'elle est validée et que sa réceptivité est vraie.

Une transition franchissable est obligatoirement et automatiquement franchie.

- **Règle 3**

Évolution des étapes actives : le franchissement d'une transition entraîne simultanément la validation de toutes les étapes immédiatement suivantes et la désactivation de toutes les étapes immédiatement précédentes.

- **Règle 4**

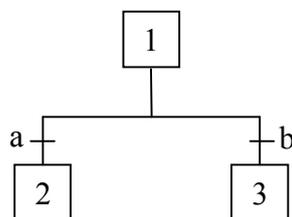
Plusieurs transitions simultanément franchissables sont simultanément franchies.

- **Règle 5**

Activation et désactivation simultanée d'une même étape : si au cours d'un même balayage, une étape est simultanément activée et désactivée elle reste active.

IV. Configurations courantes

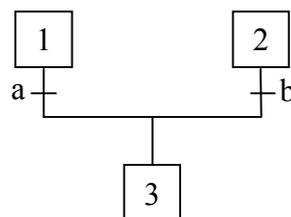
IV.1. Divergence en OU



Si 1 est active et a seul, alors désactivation de 1 et activation de 2 et 3 reste inchangée.

Si a et b et 1 est active, alors désactivation de 1 et activation de 2 et 3.

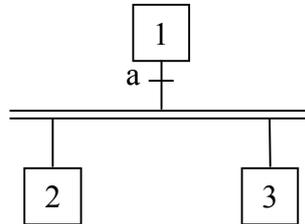
IV.2. Convergence en OU



Si 1 est active et a sans b, alors activation de 3 et désactivation de 1 et 2 reste inchangée.

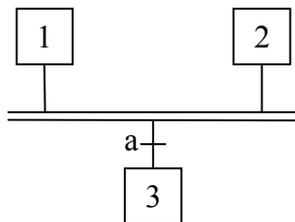
Si 1 et 2 sont actives et a et b, alors seule 3 active.

IV.3. Divergence en ET



Si 1 est active et si a, alors désactivation de 1 et activation de 2 et 3.

IV.4. Convergence en ET

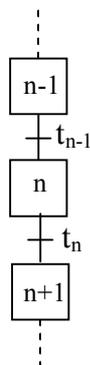


Si 1 est active seule et a, alors aucun changement.

Si 1 et 2 sont actives et a, alors activation de 3 et désactivation de 1 et 2.

V. Mise en œuvre du GRAFCET

Soit la partie du GRAFCET représentée ci après :



Nous utilisons les notations suivantes :

- Pour décrire l'activation de l'étape n :

$X_n = 1$ si l'étape n est active.

$X_n = 0$ si l'étape n est inactive.

- Pour la réceptivité t_n :

$t_n = 0$ si la réceptivité est fausse.

$t_n = 1$ si la réceptivité est vraie.

VI. Mise en équation d'une étape

La deuxième règle d'évolution du GRAFCET permet de déduire les variables qui interviennent dans l'équation d'activation de l'étape n.

D'après cette règle, la condition d'activation de l'étape n (CAX_n) est :

$$CAX_n = X_{n-1} t_{n-1}.$$

La troisième règle permet de déduire les variables qui interviennent dans la désactivation de l'étape n.

La condition de désactivation de l'étape n (CDX_n) est :

$$CDX_n = X_n t_n = X_{n+1}$$

Si la condition d'activation et la condition de désactivation de l'étape ne sont pas fausses, alors cette étape reste dans son état (effet mémoire); donc l'état de X_n à l'instant t+1 dépend de l'état précédent de X_n à l'instant t.

La table de vérité de l'activation d'une étape résume tout les cas possible et donne son état pour chaque cas.

Table de vérité de l'activité de l'étape n : X_n

$X_n(T)$	CAX_n	CDX_n	$X(T+1)$
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

Le tableau de Karnaugh associé nous donne l'équation de X_n :

$$X_n = CAX_n + \overline{CDX_n} X_n$$

$$X_n = X_{n-1} t_{n-1} + \overline{X_{n+1}} X_n$$

VII. Application aux configurations courantes

VII.1. Divergence en OU

Etape 2 :

$$CAX_2 = X_1 a$$

$$CDX_2 = X_3 + X_5$$

Etape 3 :

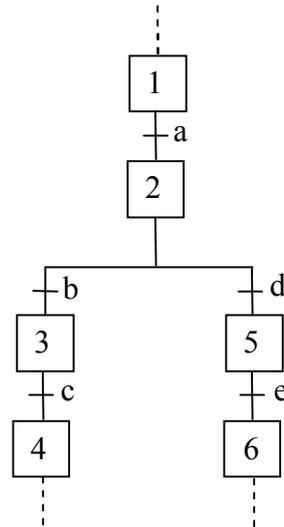
$$CAX_3 = X_2 + b$$

$$CDX_3 = X_4$$

Etape 5 :

$$CAX_5 = X_2 d$$

$$CDX_5 = X_6$$



VII.2. Convergence en OU

Etape 2 :

$$CAX_2 = X_1 a$$

$$CDX_2 = X_5$$

Etape 4 :

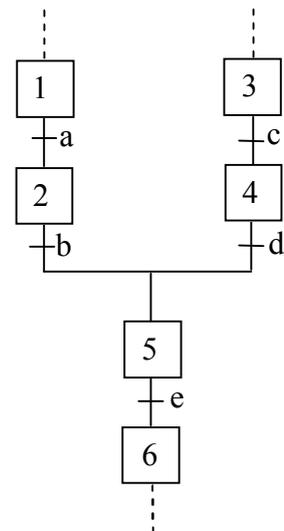
$$CAX_4 = X_3 c$$

$$CDX_4 = X_5$$

Etape 5 :

$$CAX_5 = X_2 b + X_4 d$$

$$CDX_5 = X_6$$



VII.3. Divergence en ET

Etape 2 :

$$CAX_2 = X_1 a$$

$$CDX_2 = X_3 X_5$$

Etape 3 :

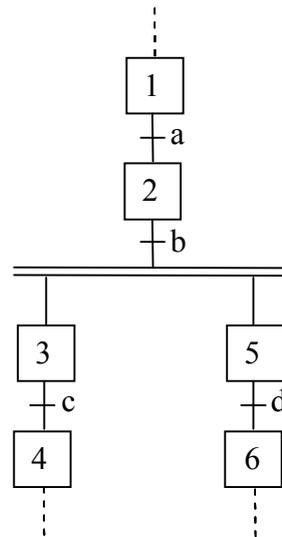
$$CAX_3 = X_2 b$$

$$CDX_3 = X_4$$

Etape 5 :

$$CAX_5 = X_2 b$$

$$CDX_5 = X_6$$



VII.4. Convergence en ET

Etape 2 :

$$CAX_2 = X_1 a$$

$$CDX_2 = X_5$$

Etape 4 :

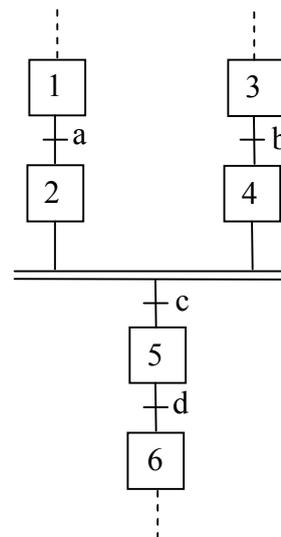
$$CAX_4 = X_3 b$$

$$CDX_4 = X_5$$

Etape 5 :

$$CAX_5 = X_2 X_4 c$$

$$CDX_5 = X_6$$



VIII. Gestion des modes marche/arrêt et des arrêts d'urgences

Soit la variable *Init* telle que :

Init = 1 : initialisation du GRAFCET : mode ARRET.

Init = 0 : déroulement du cycle : mode MARCHE.

Soient les variables ARRET D'URGENCE (AUdur et AUdoux) tel que :

AUdur = 1 : désactivation de toutes les étapes.

AUdoux = 1 : désactivation des actions, les étapes restent actives.

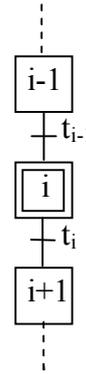
IX. Equations des étapes

IX.1. Equation d'une étape initiale

$$CAX_i = X_{i-1} t_{i-1} + Init$$

$$CDX_i = X_{i+1} + \overline{Init}$$

$$X_i = (CAX_i + \overline{CDX_i} X_i + Init) \overline{AUdur}$$

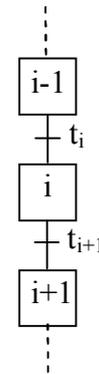


IX.2. Equation d'une étape non initiale

$$CAX_i = X_{i-1} t_{i-1} \overline{Init}$$

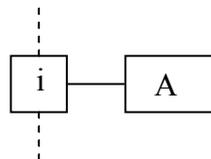
$$CDX_i = X_{i+1} + Init$$

$$X_i = (CAX_i + \overline{CDX_i} X_i) Init \overline{AUdur}$$



X. Equation des actions

$$A = X_i \overline{AUdoux}$$



XI. Modélisation de l'installation

XI.1. Liste des actions et des capteurs

X1.1.1. liste des actions

Pour le portique 1

1MLH : moteur de levage haut.

1MLB : moteur de levage bas.

1MTAV : moteur de translation avant.

1MTAR : moteur de translation arrière.

1MTAVF : freinage du moteur de translation avant.

1MTARF : freinage du moteur de translation arrière.

Pour le portique 2

2MLH : moteur de levage haut.

2MLB : moteur de levage bas.

2MTAV : moteur de translation avant.

2MTAR : moteur de translation arrière.

2MTAVF : freinage du moteur de translation avant.

2MTARF : freinage du moteur de translation arrière.

XI.1.2. Liste des capteurs

Pour le portique 1

%I 2.0 : détecteur haut.

%I 2.1 : détecteur bas.

%I 2.2 : cellule présence charge.

%I 2.3: détecteur poste

%I 2.4: détecteur comptage avant.

%I 2.5 : détecteur comptage arrière.

%I 2.6 : Zone de chargement.

%I 2.7 : détecteur surpassement haut.

%I 2.8 : détecteur surpassement bas.

%I 2.9 : détecteur surpassement arrière

%I 2.10 : anti collision pont 2.

%I 2.11 : pont1 en hors sécurité.

%I 2.12 : contact platine.

%I 2.13 : disjoncteur de puissance.

%I 2.14 : disjoncteur de frein

Pour le portique 2

%I 2.16 : détecteur haut.

%I 2.17 : détecteur bas.

%I 2.18 : cellule présence charge.

%I 2.19: détecteur poste

%I 2.20: détecteur comptage avant.

%I 2.21 : détecteur comptage arrière.

%I 2.23 : détecteur surpassement haut.

%I 2.24 : détecteur surpassement bas.

%I 2.25 : détecteur surpassement arrière

%I 2.27 : pont2 en hors sécurité.

%I 2.28 : contact platine.

%I 2.29 : disjoncteur de puissance.

%I 2.30 : disjoncteur de frein

XI.1.3. Autres désignations

TM0, TM1, TM2, TM3, TM4, TM5, TM6, TM7 : temporisateurs.

C1 et C2 sont respectivement les compteurs avant du pont1 et du pont2.

dc1 et dc2 sont respectivement les compteurs arrière du pont1 et du pont2.

XII. Commentaire

La structure de ce grafcet se décompose en deux suites de séquences exécutables simultanément.

- Les étapes allant de 1 à 52 décrivent le fonctionnement du portique1
- Les étapes allant de 53 à 109 décrivent le fonctionnement du portique 2

Les étapes 52 et 109 sont des étapes d'attente, elles permettent la synchronisation et reprise du cycle.

La réceptivité de la dernière convergence en ET est égale à 1 c'est-à-dire toujours vraie.

- M1 macros étape de décharge du portique 1.
- M2 macros étape de charge du portique1.
- M3 : macros étape de décharge du portique2.
- M4 : macros étape de charge du portique2.

XIII. Grafcet sécurité

Lorsqu'un arrêt sera enclenché la partie opérative se figera, c'est-à-dire qu'à l'instant même aucun mouvement ne sera effectué.

$CA = \%I\ 2.7 + \%I\ 2.8 + \%I\ 2.9 + \%I\ 2.10 + \%I\ 2.11 + \%I\ 2.12 + \%I\ 2.13 + \%I\ 2.14 +$
 $\%I\ 2.23 + \%I\ 2.24 + \%I\ 2.25 + \%I\ 2.27 + \%I\ 2.28 + \%I\ 2.29 + \%I\ 2.30.$

XIV. Conclusion

Le modèle de grafset conçu présente toutes les actions qu'exécute la partie opérative et toutes les conditions qui doivent être satisfaites pour qu'une action soit exécutée.

Le model pourra constituer l'étape préliminaire de la programmation de l'automate car il inclut tout les cas que l'on peut envisager sur le comportement des portiques et ceci en fonction de tous les capteurs qu'utilise le programme automate qui gère le fonctionnement de ces portiques.

I. Introduction

L'automate programmable est utilisé depuis plusieurs années dans le milieu industriel. Le développement d'une application complexe peut être ardu sans méthode de travail, car il faut tenir compte autant de la séquence de production que des modes d'opérations, de la gestion des arrêts et de la sécurité, ce qui est assuré par un programme mis en œuvre sur un logiciel de programmation compatible avec l'automate choisi. La station étudiée est gérée par un automate de type télémechanique et le logiciel de programmation qui lui est compatible est le PL7 qui fera l'objet de ce chapitre.

II. Présentation du logiciel PL7 [15]

Le PL7 est un logiciel de programmation destiné à la conception et la mise en œuvre des applications pour les automates de type télémechaniques, il offre quatre méthodes de programmation à savoir :

- le langage à contacte (ladder).
- langage liste d'instructions.
- Littéral structuré.
- Le grafcet.

De plus ces différentes méthodes peuvent être utilisées simultanément dans le même programme.

Ce logiciel propose des fonctions d'aides étendues et du diagnostic grâce auxquelles la programmation et la configuration deviennent très simples.

III. Ergonomie générale du logiciel

L'écran principal du logiciel PL7 est donné par la figure IV.1, il nous fournit les nombreux outils disponibles et les différents éléments le constituant.

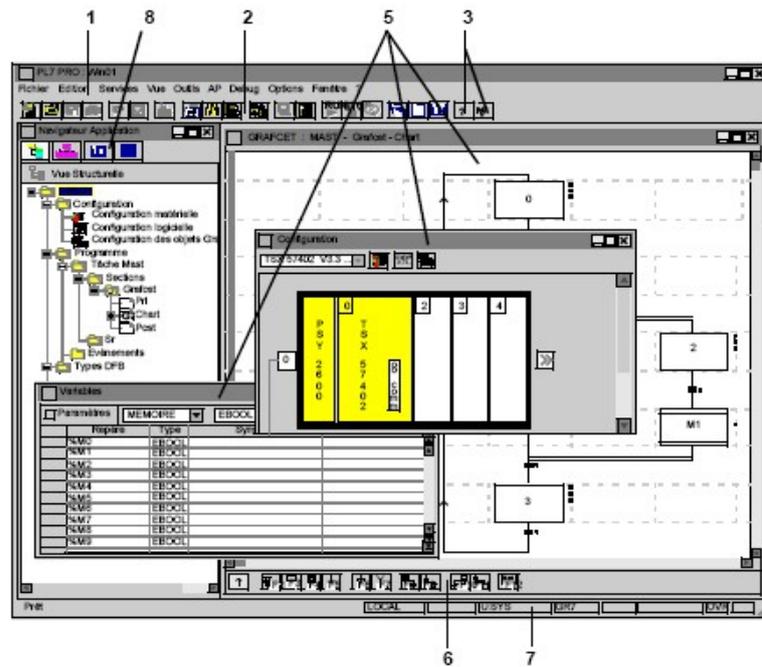


Figure IV.1. Ecran principal du logiciel PL7.

- 1- Barre de menu
- 2- Barre d'outils
- 3- Aide
- 4- Navigateur
- 5- Editeurs
- 6- Palette d'éléments graphiques
- 7- Barre d'état

III.1. Barre de menus

Elle permet l'accès à toutes les fonctions du logiciel.

III.2. La barre d'outils de PL7

Elle assure un accès rapide aux fonctions de base du logiciel, à l'aide de la souris. L'accès aux différentes fonctions est dynamique et varie suivant le contexte ; la barres d'outils se présente comme suit :

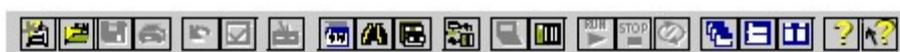


Figure IV.2. Barre d'outils du logiciel PL7.

Ce tableau donne la signification de chaque élément de la barre d'outils.

Elément	Fonction	Elément	Fonction
	Nouvelle application		Mode local
	Ouvrir une application		Mode connecté
	Enregistrer l'application		Passage de l'automate en RUN
	Imprimer tout ou partie de l'application		Passage de l'automate en STOP
	Annuler les dernières modifications		Lancer / Stopper l'animation
	Valider les modifications		Organisation des fenêtres en cascade
	Atteindre		Organisation des fenêtres en mosaïque horizontale
	Navigateur application		Organisation des fenêtres en mosaïque verticale
	Références croisées		Aide
	Bibliothèque de fonctions		Qu'est-ce que c'est ?
	Transfert automate <-> console		

III.3. Aide

Le logiciel PL7 propose deux modes d'aide qui sont :

- Aide en ligne : décrit de manière séquentielle la mise en œuvre des différents éditeurs du logiciel. Elle fournit également de nombreuses informations sur :

- Les utilisateurs.
- Les généralités de PL7.
- Les instructions de langages PL7.
- L'utilisation de PL7.
- Les métiers TSX Micro et premium.

- Aide contextuelle de PL7 : permet de d'accéder directement à l'information à partir de l'élément sélectionné.

III.4. navigateur

Le navigateur d'application permet de présenter sous forme arborescence, le contenu d'une application PL7 ; deux types de représentations sont proposés à savoir :

- Structurelle : vue traditionnelle correspondant à l'ordre de traitement par l'automate.

- Fonctionnelle : vue de l'applications en modules fonctionnels correspondant à des fonctions de l'automate.

L'écran suivant montre les deux modes de représentation.

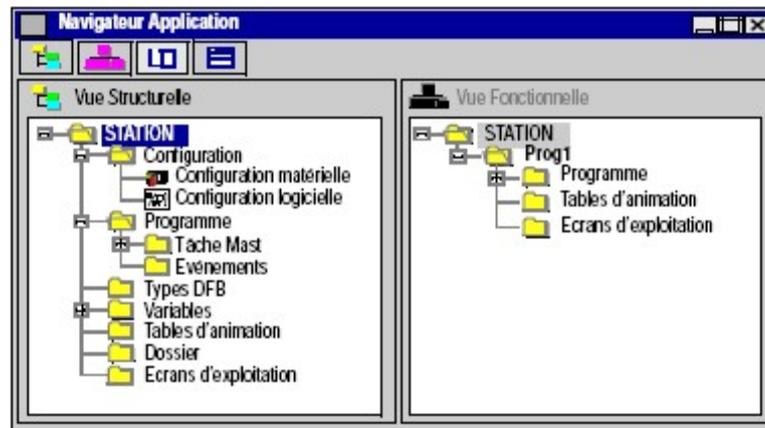


Figure IV.3. *Navigateur d'application.*

III.4.1. La représentation fonctionnelle

La représentation fonctionnelle présente l'application structurée en modules fonctionnels correspondant aux différentes fonctions de l'automatisme de l'application. Un module fonctionnel est un regroupement d'éléments de programme (sections, macro étapes, tables d'animation, écrans d'exploitation,...).

L'écran suivant présente une application structurée en modules fonctionnels.

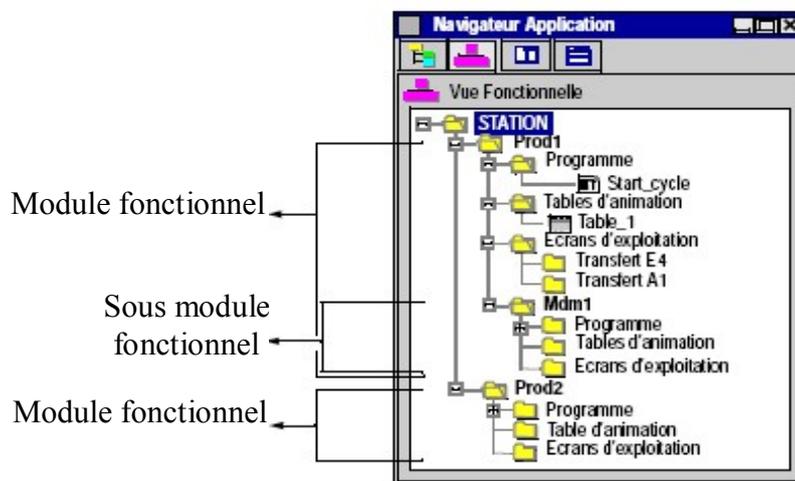


Figure IV.4. *Représentation fonctionnelle.*

III.4.2. La représentation structurale

La représentation structurale présente le contenu d'une application sous une forme arborescence ; elle permet de se déplacer à l'intérieur d'une application et offre un accès direct :

- A la configuration (matérielle, logicielle et des objets GRAFCET).
- Aux programme.
- Aux DFB contenus dans l'application.
- Aux données.
- Aux tables d'animation.
- A certaines parties du dossier (informations générales, pages de garde).
- Aux écrans d'exploitation.

L'écran suivant présent la structure arborescence d'une application et les différents éditeurs la constituant qu'on aura à définir dans la suite du chapitre.

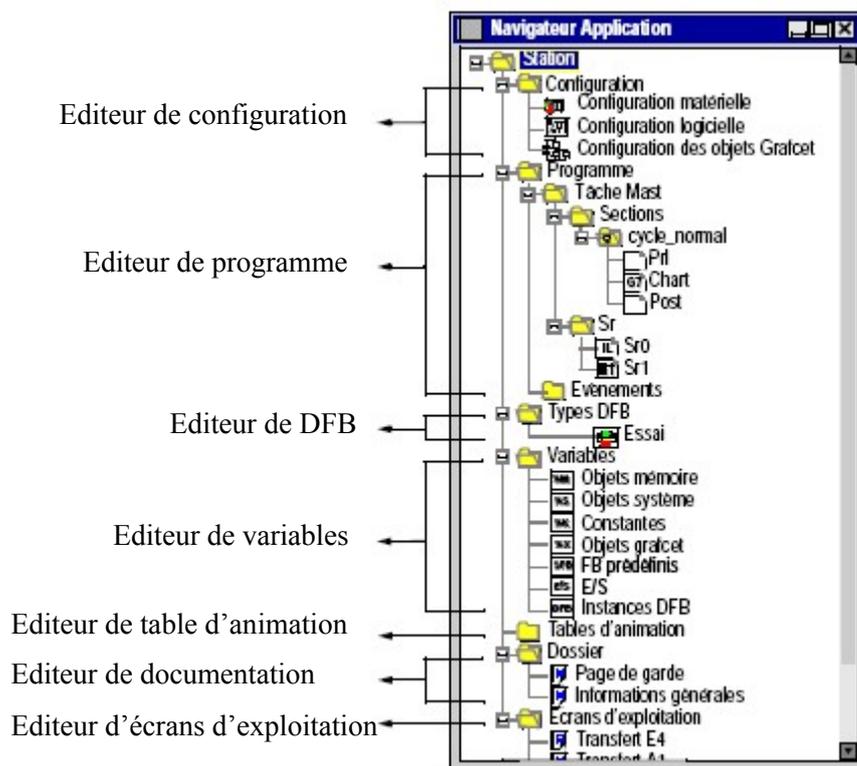


Figure IV.5. Représentation structurale.

IV. Présentation des éditeurs

IV.1. Editeur de configuration

L'éditeur de configuration de PL7 assure pour chaque application les fonctions suivantes :

- La configuration matérielle.
- La configuration logicielle.
- La configuration des objets grafcet dans le cas de programmation en langage grafcet.

L'éditeur de configuration assure également le mode connecté, des fonctions de mis au point, de réglage et de diagnostic.

IV.1.1. Configuration matérielle

Cet éditeur nous donne une représentation graphique des modules actuellement connectés dans les emplacements de l'automate.

On y accède en cliquant sur 'configuration matérielle' dans le navigateur d'application. L'éditeur de configuration permet de d'une manière intuitive de déclarer les et configurer les différents éléments constituant l'automate :

- Rack.
- Alimentation.
- Processeur.
- Module métiers.

La figure suivante illustre l'écran de configuration matérielle et différentes fonctionnalités qu'il propose.

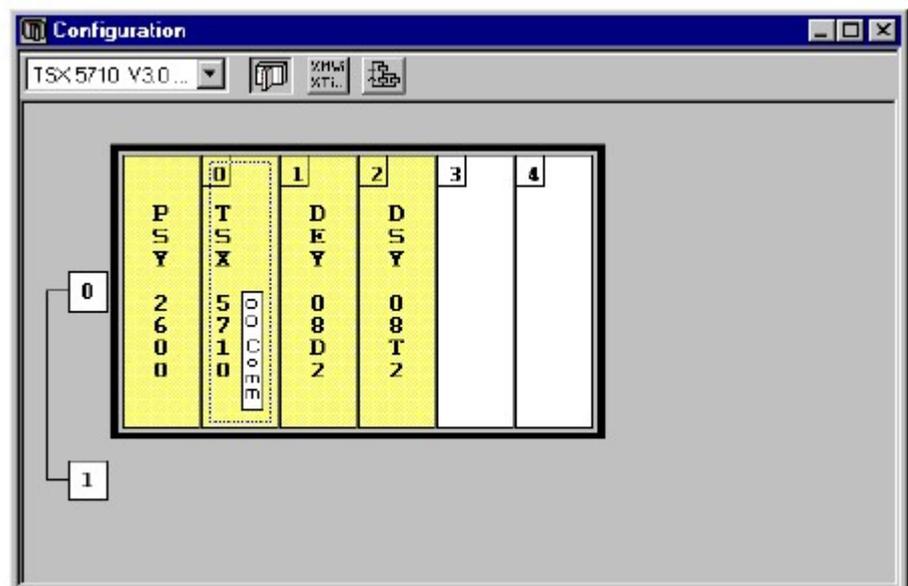


Figure IV.6.
Configuration matérielle.

IV.1.2 Configuration logicielle

L'éditeur de configuration assure le paramétrage logiciel en renseignant :

- Le nombre de blocs fonctions.
- Le nombre de registres.
- La taille des zones de variables globales.

IV.1.3. Configuration d'objets grafcet

L'éditeur de configuration permet de définir les objets Grafcet (étapes, macro étapes,...) et les paramètres d'exécution (nombre d'étapes et de transitions actives).

IV.2. Editeur de programme

Les éditeurs de programme permettent la programmation des fonctions et métiers mis en œuvre par l'application ; quatre éditeurs programme sont proposés :

- Editeur langage à contacts (LD).
- Editeur langage liste d'instructions (IL).
- Editeur langage littéral structure (ST).
- Editeur langage grafcet (G7).

IV.2.1. Langage à contacts (LD)

IV.2.1.1. 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 reliés entre eux par des connexions horizontales et verticales, ces éléments représentent:

- 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.

Chaque réseau (nommé **Rung**) comporte:

- Une zone de saisie de l'**étiquette** (cellule située en haut et à gauche du rung).
- Une zone de saisie du **commentaire** (1ère ligne à droite de l'étiquette).
- Une zone de saisie (**Test et Action**) des éléments graphiques:
 - 7 lignes et 11 colonnes (taille maximum) pour une application de niveau L1.
 - 16 lignes et 11 colonnes (taille maximum) pour une application de niveau L2

La zone Test (colonnes 1 à 10) accueille:

- Les contacts.
- Les blocs fonctions.
- Les blocs comparaison.

La zone Action (colonne 11) accueille:

- Les bobines.
- Les blocs opérations.

L'écran suivant représente les différentes zones que compose l'éditeur à langage de contact.

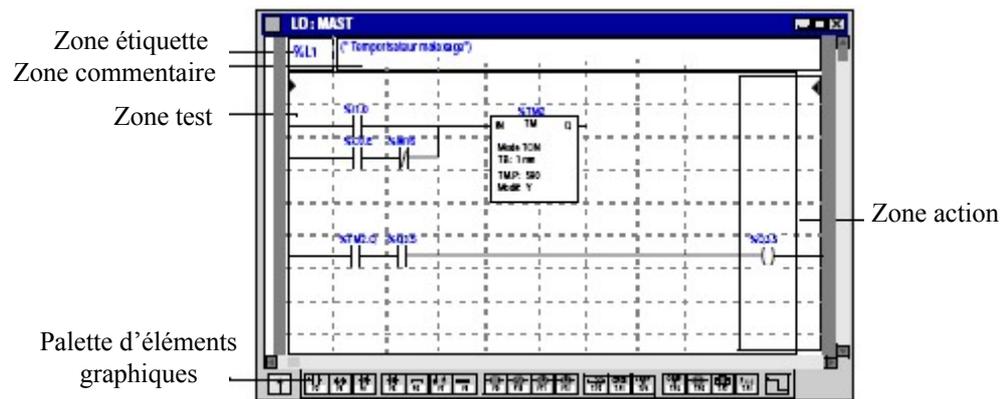


Figure IV.7. Zone de langage à contact.

IV.2.1.2. Règle de programmation

➤ Programmation d'un réseau de contact

La programmation d'un réseau de contacts s'effectue à l'aide des éléments graphiques, en respectant les règles de programmation ci-après.

Les éléments graphiques simples de test et d'action occupent chacun une cellule au sein d'un réseau.

Toute ligne de contacts commence sur la ligne de potentiel gauche et doit se terminer sur la ligne de potentiel droite. Les tests sont toujours situés sur les colonnes 1 à 10 et les actions sur la colonne 11.

Le sens de circulation du courant est le suivant :

- Pour les liaisons horizontales, de la gauche vers la droite.
- Pour les liaisons verticales, dans les deux sens.

L'écran suivant représente un exemple e réseau de contacts.

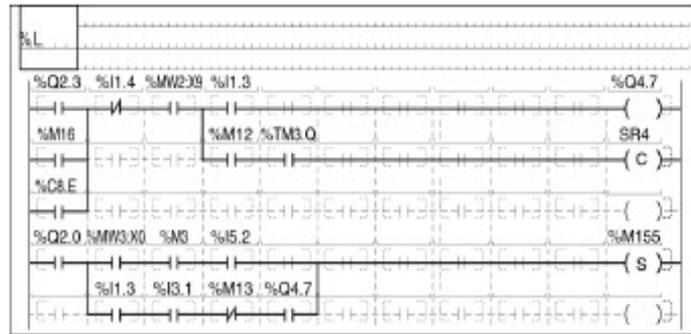


Figure IV.8. Exemple de langage à contacte.

➤ Programmation des blocs de fonction

Les blocs de fonctions standard se positionnent test des réseaux de contacts. Quelque soit le type de bloc fonction utilisé, il doit obligatoirement être relié en entrée à la barre de potentiel gauche, en direct ou à travers d'autres éléments graphiques.

- **Sorties "en l'air"** : il n'est pas nécessaire de relier à d'autres éléments graphiques les sorties des blocs fonction.
- **sorties testables** : les sorties des blocs fonction sont accessibles à l'utilisateur sous forme d'objet bit.

Les variables internes de blocs et les sorties graphiques sont des objets exploitables à distance depuis une autre partie du programme.

Les entrées non câblées des blocs fonction standard sont mises à 0. Comme pour les éléments graphiques du type contacts, il est possible de réaliser des combinaisons de blocs fonction.

L'illustration suivante présente un exemple de réseau de contacts contenant deux blocs de fonctions.

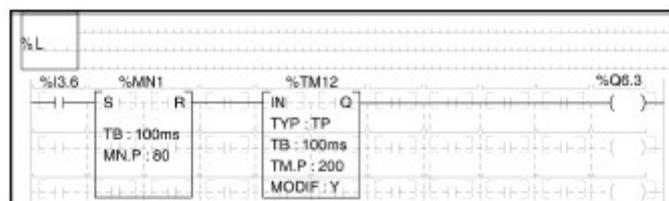


Figure IV.9. Blocs fonctions.

➤ Programmation des blocs opération

Les blocs comparaison se positionnent dans la zone test et les blocs opération se positionnent dans la zone action.

Quel que soit le type de bloc opération utilisé, il doit obligatoirement être relié en entrée à la barre de potentiel gauche, en direct ou à travers d'autres éléments graphiques.

Comme pour les éléments graphiques du type contacts, il est possible de réaliser des combinaisons de blocs fonction et opération.

L'illustration suivante présente un exemple d'un réseau de contacts contenant deux blocs de comparaison et un bloc opération.

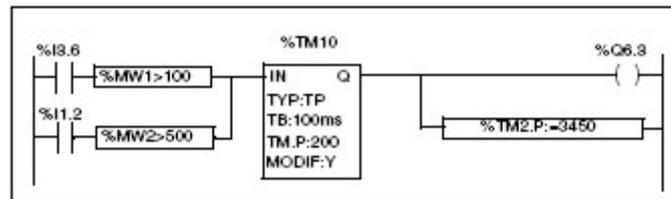


Figure IV.10. Bloc comparaison et opération.

IV.2.2. Editeur langage liste d'instruction

Une section écrite en langage liste d'instructions se compose d'une suite d'instructions exécutées séquentiellement par l'automate.

L'illustration suivante présente un programme liste d'instructions PL7 et le détail d'une instruction.

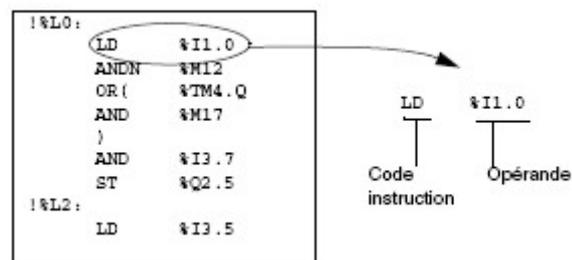


Figure IV.11. Présentation d'un programme liste d'instruction.

Comme le montre la figure une instruction est composée d'un code instruction et une opérande.

- l'opérande : Une instruction agit sur un opérande. Cet opérande peut être :
 - Une entrée/sortie de l'automate (boutons-poussoirs, détecteurs, relais, voyants...).
 - Une fonction d'automatisme (temporisateurs, compteurs...).

- Une opération arithmétique et logique ou une opération de transfert.
 - Une variable interne de l'automate.
- Code instruction : Le code instruction détermine l'opération à exécuter. Il existe deux types de codes instructions :
- **Test** : dans laquelle figurent les conditions nécessaires à une action (ex : LD, AND, OR...).
 - **Action** : qui sanctionne le résultat consécutif à un enchaînement de test. (ex : ST, STN, R).

IV.2.2.1. Structure d'un programme en liste d'instruction

Comme en langage à contacts, les instructions sont organisées en séquence d'instructions (équivalent à un réseau de contacts) appelée phrase.

L'illustration suivante présente une phrase liste d'instructions PL7

```

! (*Attente de séchage*) _____ 1
%L2 : _____ 2
      LD %I1.0
      AND %M10 _____ 3
      ST %Q2.5

```

Chaque phrase commence par un point d'exclamation (généralisé automatiquement), elle comporte les éléments suivants :

- **Commentaire** : Il facilite l'interprétation d'une phrase à laquelle il est affecté. Il est optionnel. Il peut être intégré au début d'une phrase et peut occuper 3 lignes maximum (soit 222 caractères alphanumériques), encadrés de part et d'autre par les caractères (* et *).

L'illustration ci-après repère la position du commentaire dans une phrase.

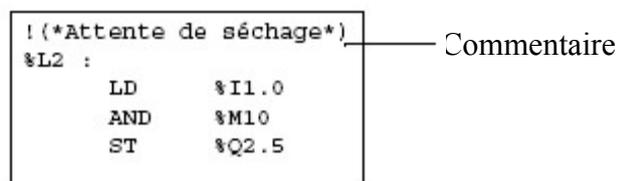


Figure IV.12. Représentation d'un commentaire.

Les commentaires s'affichent uniquement à partir de la première ligne de la phrase. En cas de suppression d'une phrase, le commentaire qui lui est associé est également supprimé.

Les commentaires sont mémorisés dans l'automate et sont accessibles à tout moment par l'utilisateur. A ce titre, ils consomment de la mémoire programme.

- **Étiquette** : Elle permet de repérer une phrase dans une entité de programme (programme principal, sous-programme, ...). Elle est optionnelle.

Cette étiquette a la syntaxe suivante : %Li avec i compris entre 0 et 999. Elle se positionne en début d'une phrase.

Le programme suivant illustre l'utilisation d'une étiquette.

```

%L0 :
    LD    %M40
    JMPC  %L10

!(*Attente de séchage*)
%L2 :
    LD    %I1.0
    AND  %M10
    ST    %Q2.5
...
%L10 : ————— Etiquette
    LD    %I3.5
    ANDN %Q4.3
    OR   %M20
    ST   %Q2.5

```

Figure IV.13. Représentation d'une étiquette.

Il est à noter que :

- Une même étiquette ne peut être affectée qu'à une seule phrase au sein d'une même entité de programme.
- Il est nécessaire d'étiqueter une phrase afin de permettre un branchement après un saut de programme.
- L'ordre des repères des étiquettes est quelconque, c'est l'ordre de saisie des phrases qui est prise en compte par le système lors de la scrutation.

- **Instructions** : il s'agit des différentes instructions que comporte le langage liste d'instruction elles sont réparties en quatre types :

- instruction de test.
- Instruction d'action.
- Instruction sur bloc de fonction.
- Instruction numérique.

IV.2.3. Langage littéral structuré

Le langage littéral structuré est un langage évolué de type algorithmique particulièrement adapté à la programmation de fonctions arithmétiques complexes, manipulations de tableaux et gestions de messages.

Il permet la réalisation de programmes par écriture de lignes de programmation, constituées de caractères alphanumériques.

Ce langage est utilisable avec les logiciels PL7 Micro, PL7 Junior et PL7 Pro sur les automates Premium et Micro.

Dans la version PL7 Pro, ce langage permet la création des blocs fonction utilisateur DFB sur les automates Premium.

L'illustration suivante présente un programme en langage structuré PL7.

```
!      (* Recherche du premier élément non nul dans un
      tableau de 32 mots, détermination de sa valeur
      (%MW10), de son rang (%MW11). Cette recherche
      s'effectue si %M0 est à 1, %M1 est mis à 1 si
      un élément non nul existe, sinon il est mis à 0*)

      IF %M0 THEN
          FOR %MW99:=0 TO 31 DO
              IF %MW100[%MW99]<>0 THEN
                  %MW10:=%MW100[%MW99];
                  %MW11:=%MW99;
                  %M1:=TRUE;
                  EXIT;      (*Sortie de la boucle*)
              ELSE
                  %M1:=FALSE;
              END_IF;
          END_FOR;
      ELSE
          %M1:=FALSE;
      END_IF;
```

Figure IV.14. Programme en langage structuré.

Comme en langage liste d'instruction, une phrase en langage littéral structuré est composé de :

- une commentaire (optionnel) : qui nous renseigne sur la phrase.
- Une étiquette (optionnelle) : sert à repérer la phrase.
- Instructions (optionnelles) : une à plusieurs instruction séparées par ";".

IV.2.4. Langage grafcet

L'éditeur Grafcet permet de représenter graphiquement et de façon structurée le fonctionnement d'un automatisme séquentiel.

Cet éditeur se compose de 8 pages de 14 lignes et 11 colonnes qui définissent ainsi des cellules pouvant accueillir chacune un élément graphique.

Il dispose de nombreux outils permettant la saisie de façon conviviale tels que :

- Une palette d'objets graphiques accessible directement par la souris ou le clavier (étapes, transitions, liaisons, renvois, macro étapes, ...).
- un accès direct à la programmation des actions ou des réceptivités.
- une numérotation automatique des étapes.
- un affichage par page Grafcet avec les lignes d'étapes et de transitions.
- une saisie simplifiée des commentaires.
- deux modes de visualisation.

L'écran suivant présente les différentes zones que compose l'éditeur de langage Grafcet.

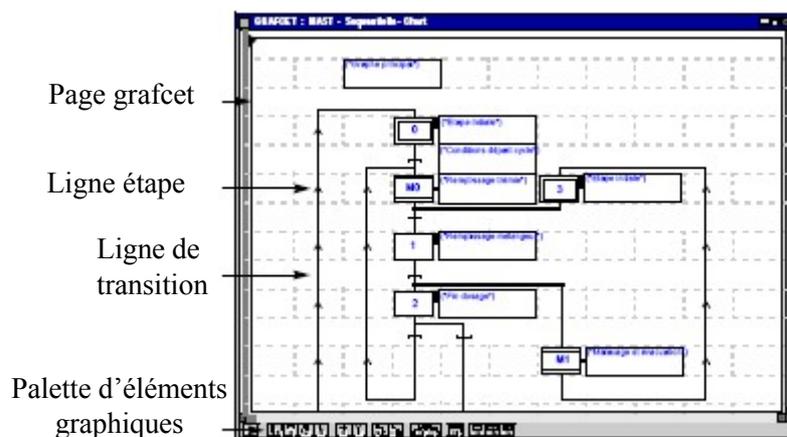


Figure IV.15. Zone de programmation en langage Grafcet.

Deux modes de visualisation sont proposés par le logiciel :

- vue normale (mode de saisie par défaut).
- vue réduite.

Ce dernier permet l'affichage d'un plus grand nombre de pages Grafcet tout en conservant le même niveau d'information.

IV.3. Editeur des types DFB

L'éditeur de DFB permet de programmer des blocs fonction propres à l'utilisateur qui répondent aux spécificités de ses applications.

Ces blocs fonction utilisateur permettent de structurer une application. Ils seront utilisés dès qu'une séquence de programme se trouve répétée dans une application ou pour figer une programmation standard.

Note : La création de DFB nécessite PL7 Pro. L'utilisation de DFB est possible avec PL7 Junior et PL7 Pro.

L'écran suivant représente la vue générique de l'éditeur des types DFB.

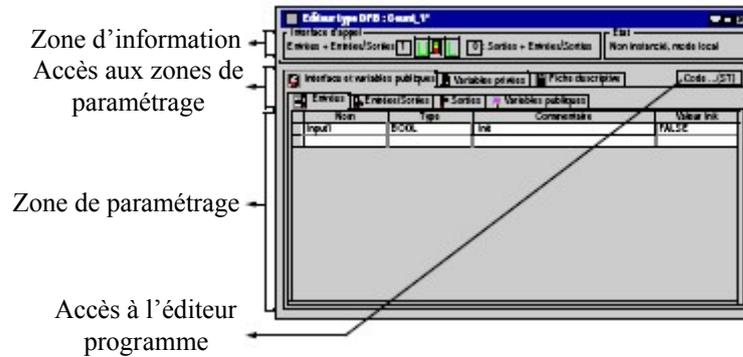


Figure IV.16. Editeur DFB.

IV.4. Editeur de variables

L'éditeur de variables permet de :

- symboliser les différents objets de l'application (bits, mots, blocs fonctions, modules métiers, ...).
- paramétrer les blocs fonction prédéfinis (temporisateurs, compteurs, ...).
- saisir les valeurs des constantes et choisir la base d'affichage (décimal, binaire, hexadécimal, flottant, message).
- instancier et paramétrer les blocs fonction utilisateur DFB.

L'éditeur de variables se présente ainsi :

Repère	type	Symbole	Commentaire
WCH1 MOD	CH		
WCH1 MOD.EKR	BOOL		
WCH1 MOD	WORD		
WCH1 MOD.1	WORD		
WCH1 MOD.2	WORD		
P.WCH1.0	CH		
P.WCH1.0	WORD		
P.WCH1.0.1	WORD		
P.WCH1.0.2	WORD		
P.0.0	BOOL	Presence piece	détection d'une pièce devant le voinv 1
P.0.1	BOOL	Capteur v1 sortie	capteur de position du voinv 1 sortie
P.0.2	BOOL	Capteur v1 sortie	capteur de position du voinv 1 sortie
P.0.3	BOOL	Capteur v2 sortie	capteur de position du voinv 2 sortie
P.0.4	BOOL	Capteur v2 sortie	capteur de position du voinv 2 sortie
P.0.5	BOOL	In	mesa en marche du système
P.0.6	BOOL	Arrêt	arrêt du cycle automatique
P.0.7	BOOL		
P.0.EKR	BOOL		

Figure IV.17. Editeur de variables

L'accès aux variables est facilité par :

- Un classement par famille et par type.
- des fonctions de tri (tri par symboles ou repères).
- la possibilité d'afficher tous les objets associés à une même variable (par exemple tous les bits d'un mot, tous les objets associés à un bloc fonction prédéfini).
- la possibilité de pré symboliser des objets de certains métiers.
- la possibilité de lancer une recherche avec joker sur symbole ou commentaire.
- la possibilité de filtre sur les E/S (permettre de ne visualiser, pour un module, que les variables d'entrée ou de sortie agissant réellement sur le process).
- la possibilité de Copier / Coller les paramètres SFB.
- la possibilité de Couper/Copier / Coller les symboles et commentaires d'une suite de variables.
- la possibilité de supprimer la Pré symbolisation.
- l'affichage en gras des variables utilisées dans le programme.
- l'affichage en rouge des chevauchements des variables utilisées dans le programme.

IV.5. Editeur de tables d'animation

L'éditeur de tables d'animation permet de créer des tables contenant des listes de variables à surveiller ou à modifier.

Cet éditeur propose des fonctionnalités telles que :

- Création manuelle de tables par saisie des variables, ou création automatique à partir de tout ou parties de sections de programme ou d'objets animés dans un écran d'exploitation.
- modification de la valeur courante des variables.
- forçage de la valeur courante des objets bits.
- choix de la base d'affichage de la valeur courante (décimal, binaire, hexadécimal).

IV.6. Editeur de documentation

L'éditeur de documentation permet de constituer, de visualiser et d'imprimer le dossier de l'application.

Il s'articule autour du navigateur de documentation qui montre sous forme arborescente la constitution du dossier.

Cet éditeur permet de définir :

- Une page de garde, comprenant le nom du concepteur et du projet.
- des pages d'informations générales.
- un cartouche.

Il génère automatiquement :

- le sommaire.
- le dossier d'application : configurations matérielle / logicielle, et programme.
- la liste des variables, triées par repère ou par symbole.
- les références croisées, triées par repère ou symbole.

IV.7. Editeur d'écrans d'exploitation

L'éditeur d'écrans d'exploitation est un outil destiné à faciliter l'exploitation d'un processus automatisé.

Il permet à un concepteur de développer des écrans adaptés au procédé et qui offre à l'exploitant :

- Un ensemble d'informations affichées en clair : texte explicatif, valeurs dynamiques, synoptiques couleurs.
- la possibilité d'agir simplement et rapidement : modification et surveillance dynamique des variables automate, ...

L'éditeur d'écrans d'exploitation comporte trois fenêtres de taille paramétrable:

- Le navigateur qui permet d'accéder aux différents écrans d'exploitation.
- l'éditeur graphique qui permet de créer, modifier puis d'exploiter l'animation des écrans.
- la fenêtre de visualisation, utilisée pour faciliter la mise au point d'une application et visualiser de façon très simple les messages de diagnostic.

V. Structure du programme de conduite de l'installation

Contrairement aux autres logiciels de programmation (Step7, CX programmer,...) des autres automates (Siemens, Allain Bradly, Omron,...), la programmation sous PL7 n'est pas devisée en bloc et sous bloc.

Le programme actuel du fonctionnement de l'installation est un programme flexible, il est conçu d'une manière à permettre le changement du cycle de travail et du temps des différentes attentes des portiques à travers le pupitre de commande en saisissant les paramètres des portiques sur la table de commande affichée sur l'écran du pupitre.

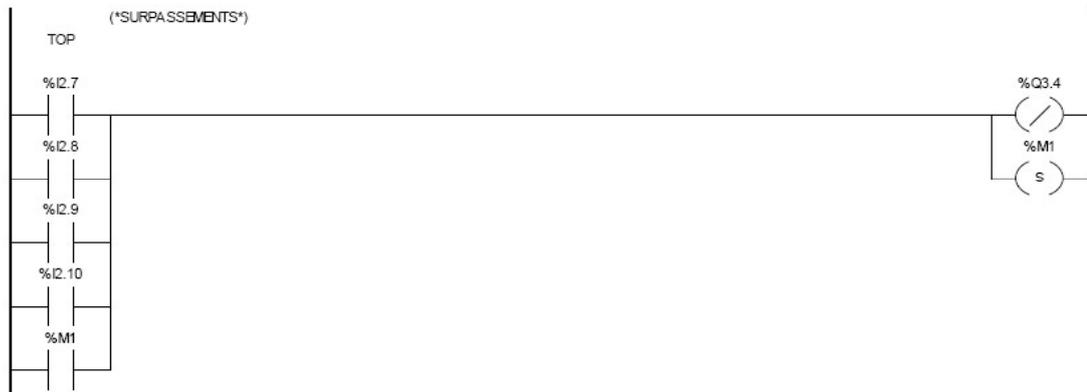
C'est un programme imbriqué c'est-à-dire il est utilise deux langages l'un dans l'autre ceci par l'attribution de chacune des touches du terminal opérateur une tache bien spécifique.

La conception d'un tel programme nécessite non seulement la connaissance du logiciel PL7 mais aussi la connaissance de l'XBT L1000 qui est le logiciel de programmation du pupitre de commande.

Dans ce qui suit nous allons illustrer le programme fait sous le logiciel PL7 en utilisant le langage LADER. Nous nous limiterons aux onze premières étapes du premier portique.

MAST-PRL

Commentaire :



Liste de Variables utilisées dans le rung :

REPERE	SYMBOLE	COMMENTAIRE
%I2.7		
%Q3.4		commande grande vitesse p1
%I2.8		
%M1		image de la grande vitesse
%I2.9		
%I2.10		

Auteur :	4.2.1 Sections 4.2.1.1 Pont_1	Imprimé le 01/07/2008
Service :		Indice :
Automate cible : TSX 57203		Folio : 4.2.1.1 - 1

Ce document est la propriété de la société XXX et ne peut être reproduit ou communiqué sans son autorisation.

CHART - PAGE 0

Commentaire :

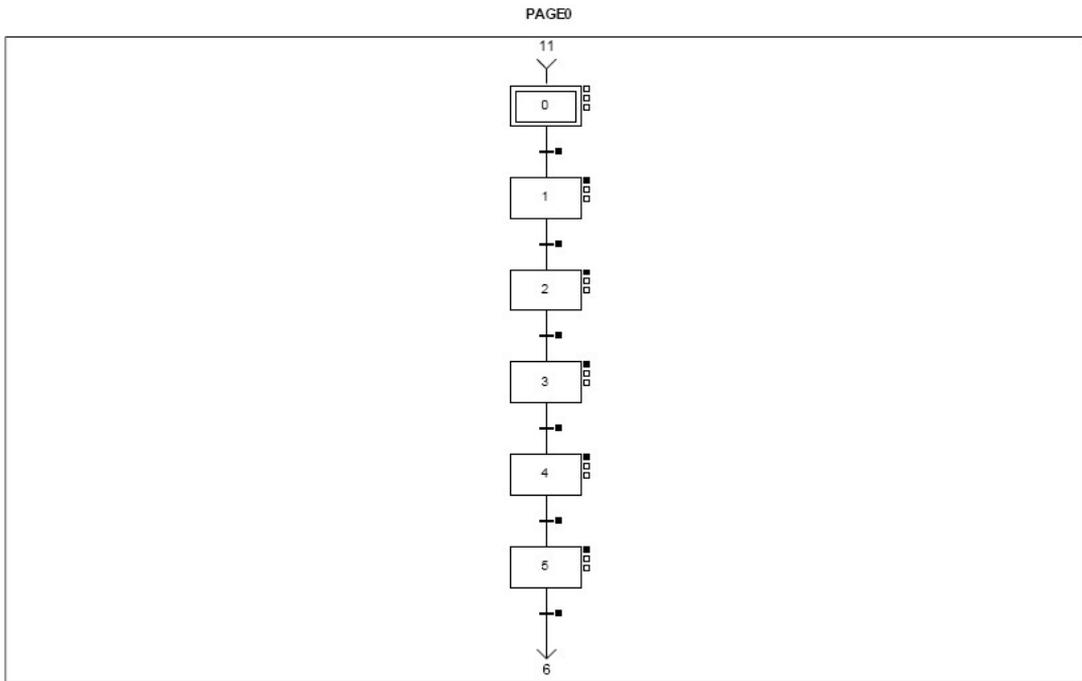
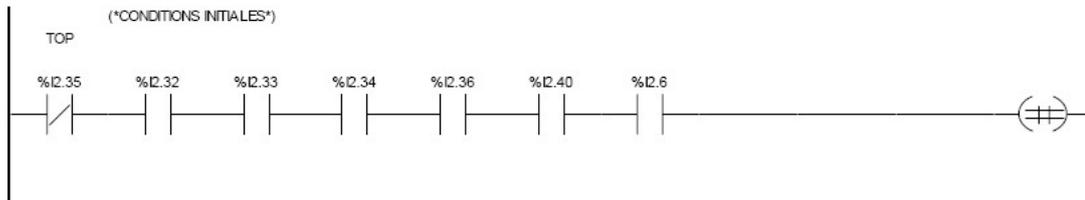


CHART - PAGE0 %X(0)->%X(1)



Liste de Variables utilisées dans le rung :

REPÈRE	SYMBOLE	COMMENTAIRE
%I2.35		arrêt d'urgence
%I2.32		ligne en automatique
%I2.33		acquiescement défaut
%I2.34		depart cycle
%I2.36		marche pont ok
%I2.40		bp validation chargement
%I2.6		zone de chargement

CHART - PAGE0 %X(1)->%X(2)



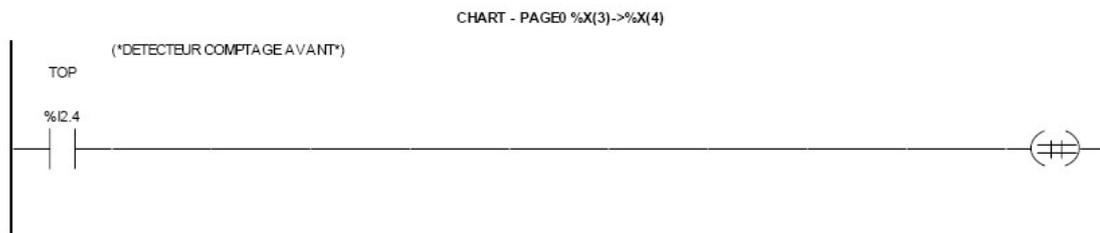
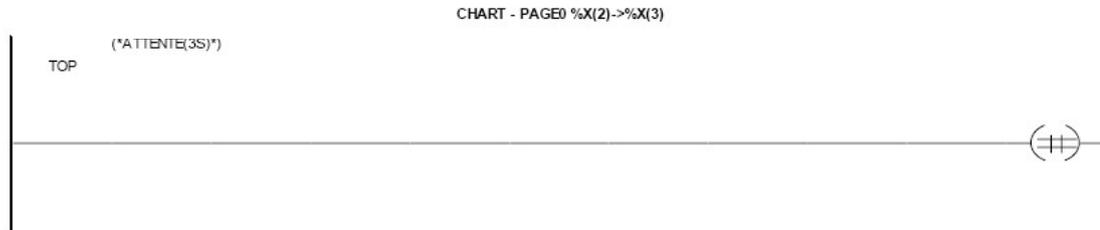
Auteur :	4.2.1 Sections	Imprimé le 01/07/2008
Service :	4.2.1.1 Pont_1	Indice :
Automate cible : TSX 57203		Folio : 4.2.1.1 - 2

Ce document est la propriété de la société XXX et ne peut être reproduit ou communiqué sans son autorisation.

CHART - PAGE0

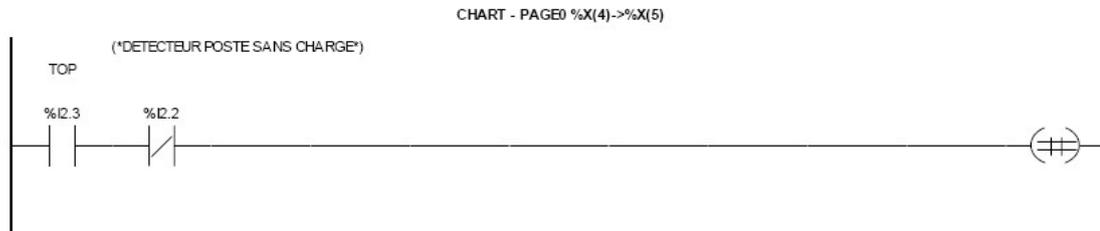
Liste de Variables utilisées dans le rung :

REPERE	SYMBOLE	COMMENTAIRE
%I2.0		detecteur haut p1



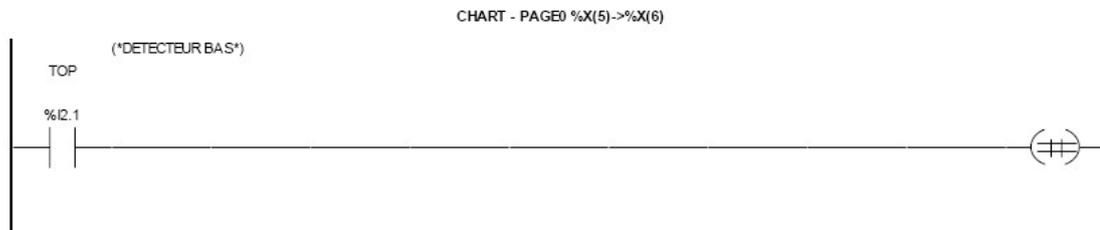
Liste de Variables utilisées dans le rung :

REPERE	SYMBOLE	COMMENTAIRE
%I2.4		detecteur comptage avant p1



Liste de Variables utilisées dans le rung :

REPERE	SYMBOLE	COMMENTAIRE
%I2.3		detecteur poste p1
%I2.2		cellule presence charge



Auteur :	4.2.1 Sections 4.2.1.1 Pont_1	Imprimé le 01/07/2008
Service :		Indice :
Automate cible : TSX 57203		Folio : 4.2.1.1 - 3

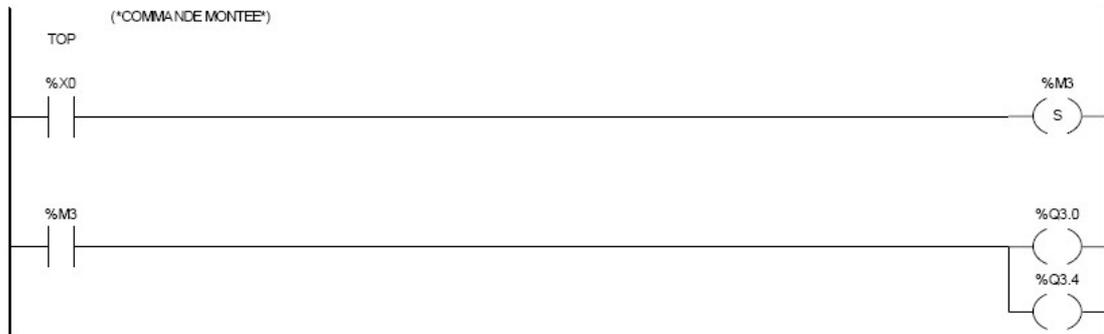
Ce document est la propriété de la société XXX et ne peut être reproduit ou communiqué sans son autorisation.

CHART - PAGE0

Liste de Variables utilisées dans le runq :

REPERE	SYMBOLE	COMMENTAIRE
%I2.1		detecteur bas p1

CHART - PAGE0 %X1 P1



Liste de Variables utilisées dans le runq :

REPERE	SYMBOLE	COMMENTAIRE
%X0		
%M3		image de sortie commande montée
%Q3.0		commande montée p1
%Q3.4		commande grande vitesse p1

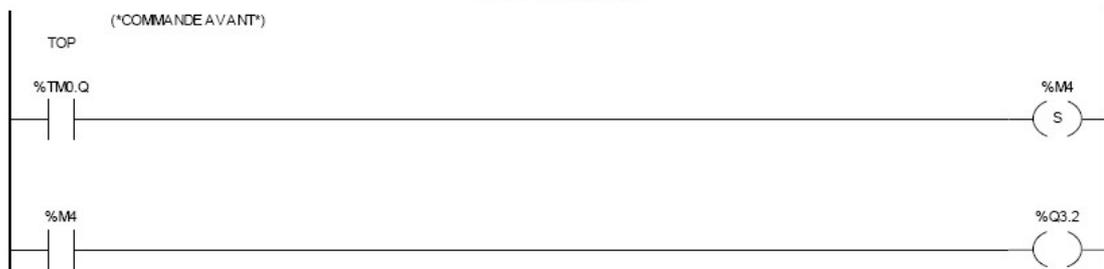
CHART - PAGE0 %X2 P1



Liste de Variables utilisées dans le runq :

REPERE	SYMBOLE	COMMENTAIRE
%I2.0		detecteur haut p1
%TMD		tempo d'attente

CHART - PAGE0 %X3 P1



Auteur :	4.2.1 Sections 4.2.1.1 Pont_1	Imprimé le 01/07/2008
Service :		Indice :
Automate cible : TSX 57203		Folio : 4.2.1.1 - 4

Ce document est la propriété de la société XXX et ne peut être reproduit ou communiqué sans son autorisation.

CHART - PAGE0

Liste de Variables utilisées dans le rung :

REPERE	SYMBOLE	COMMENTAIRE
%TMD.Q		
%M4		image de sortie commande avant
%Q3.2		commande avant p1



Liste de Variables utilisées dans le rung :

REPERE	SYMBOLE	COMMENTAIRE
%C1		detecteur comptage avant p1
%I2.4		

CHART - PAGE0 %X4 P1



Liste de Variables utilisées dans le rung :

REPERE	SYMBOLE	COMMENTAIRE
%MW100		
%KW1		
%M4		image de sortie commande avant



Liste de Variables utilisées dans le rung :

REPERE	SYMBOLE	COMMENTAIRE
%M4		image de sortie commande avant
%Q3.0		commande montée p1

Auteur :	4.2.1 Sections	Imprimé le 01/07/2008
Service :	4.2.1.1 Pont_1	Indice :
Automate cible : TSX 57203		Folio : 4.2.1.1 - 5

Ce document est la propriété de la société XXX et ne peut être reproduit ou communiqué sans son autorisation.

CHART - PAGE0

CHART - PAGE0 %X5 P1



Liste de Variables utilisées dans le rung :

REPERE	SYMBOLE	COMMENTAIRE
%X4		
%M5		image de sortie commande de descente



Liste de Variables utilisées dans le rung :

REPERE	SYMBOLE	COMMENTAIRE
%M5		image de sortie commande de descente
%Q3.1		commande descente p1

Auteur :	4.2.1 Sections 4.2.1.1 Pont_1	Imprimé le 01/07/2008
Service :		Indice :
Automate cible : TSX 57203		Folio : 4.2.1.1 - 6

Ce document est la propriété de la société XXX et ne peut être reproduit ou communiqué sans son autorisation.

CHART - PAGE 1

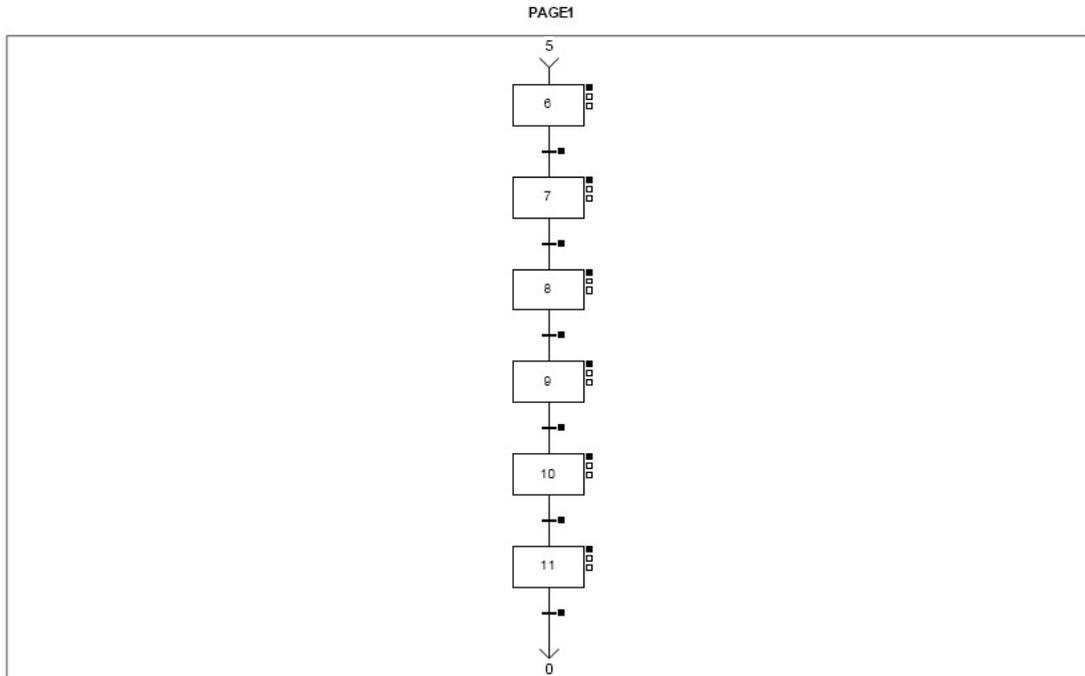
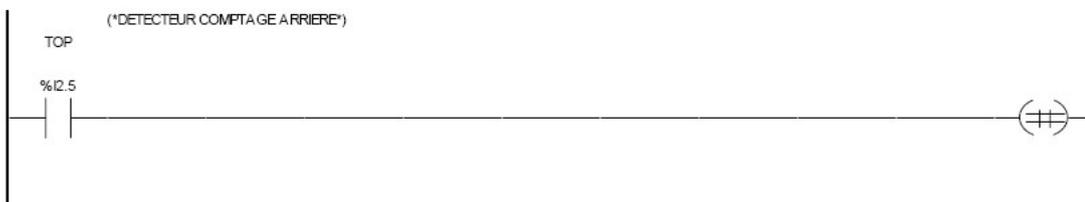


CHART - PAGE1 %X(6)->%X(7)



Liste de Variables utilisées dans le rung :

REPÈRE	SYMBOLE	COMMENTAIRE
%I2.5		detecteur comptage arriere p1

CHART - PAGE1 %X(7)->%X(8)



Liste de Variables utilisées dans le rung :

REPÈRE	SYMBOLE	COMMENTAIRE
%I2.3		detecteur poste p1
%I2.2		cellule presence charge

Auteur :	4.2.1 Sections	Imprimé le 01/07/2008
Service :	4.2.1.1 Pont_1	Indice :
Automate cible : TSX 57203		Folio : 4.2.1.1 - 7

Ce document est la propriété de la société XXX et ne peut être reproduit ou communiqué sans son autorisation.

CHART - PAGE1

CHART - PAGE1 %X(8)->%X(9)



Liste de Variables utilisées dans le rung :

REPERE	SYMBOLE	COMMENTAIRE
%I2.0		detecteur haut p1

CHART - PAGE1 %X(9)->%X(10)



CHART - PAGE1 %X(10)->%X(11)



Liste de Variables utilisées dans le rung :

REPERE	SYMBOLE	COMMENTAIRE
%I2.5		detecteur comptage arriere p1

CHART - PAGE1 %X(11)->%X(0)



Liste de Variables utilisées dans le rung :

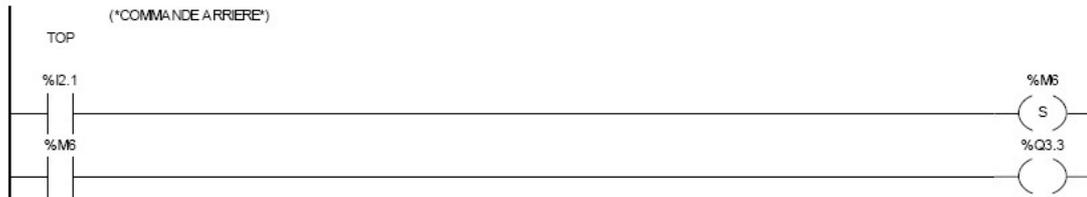
REPERE	SYMBOLE	COMMENTAIRE
%I2.3		detecteur poste p1
%I2.2		cellule presence charge

Auteur :	4.2.1 Sections	Imprimé le 01/07/2008
Service :	4.2.1.1 Pont_1	Indice :
Automate cible : TSX 57203		Folio : 4.2.1.1 - 8

Ce document est la propriété de la société XXX et ne peut être reproduit ou communiqué sans son autorisation.

CHART - PAGE1

CHART - PAGE1 %X6 P1



Liste de Variables utilisées dans le rung :

REPERE	SYMBOLE	COMMENTAIRE
%I2.1		detecteur bas p1
%M6		image de sortie commande arriere
%Q3.3		commande arriere p1



Liste de Variables utilisées dans le rung :

REPERE	SYMBOLE	COMMENTAIRE
%C2		detecteur comptage arriere p1
%I2.5		

CHART - PAGE1 %X7 P1



Liste de Variables utilisées dans le rung :

REPERE	SYMBOLE	COMMENTAIRE
%MW101		
%KW2		
%M6		image de sortie commande arriere



Auteur :	4.2.1 Sections	Imprimé le 01/07/2008
Service :	4.2.1.1 Pont_1	Indice :
Automate cible : TSX 57203		Folio : 4.2.1.1 - 9

Ce document est la propriété de la société XXX et ne peut être reproduit ou communiqué sans son autorisation.

CHART - PAGE1

Liste de Variables utilisées dans le runq :

REPERE	SYMBOLE	COMMENTAIRE
%M6		image de sortie commande arriere
%Q3.3		commande arriere p1

CHART - PAGE1 %X8 P1



Liste de Variables utilisées dans le runq :

REPERE	SYMBOLE	COMMENTAIRE
%X7		
%M7		
%Q3.0		commande montée p1

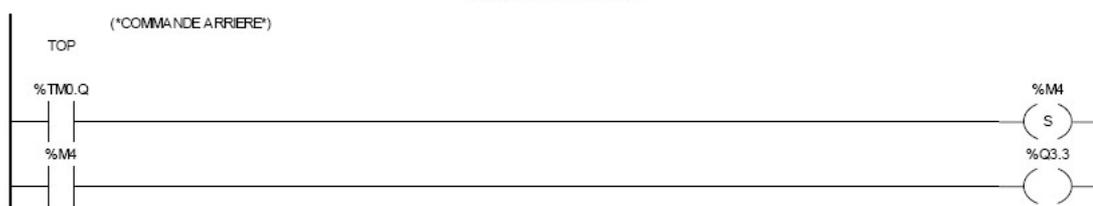
CHART - PAGE1 %X9 P1



Liste de Variables utilisées dans le runq :

REPERE	SYMBOLE	COMMENTAIRE
%I2.0		decteur haut p1
%TMD		tempo d'attente

CHART - PAGE1 %X10 P1



Liste de Variables utilisées dans le runq :

REPERE	SYMBOLE	COMMENTAIRE
%TMD.Q		image de sortie commande avant
%M4		commande arriere p1
%Q3.3		

Auteur :	4.2.1 Sections 4.2.1.1 Pont_1	Imprimé le 01/07/2008
Service :		Indice :
Automate cible : TSX 57203		Folio : 4.2.1.1 - 10

Ce document est la propriété de la société XXX et ne peut être reproduit ou communiqué sans son autorisation.

CHART - PAGE1

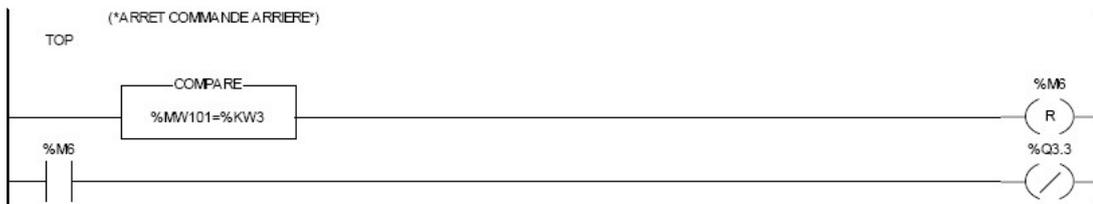
CHART - PAGE1 %X10 P1



Liste de Variables utilisées dans le rung :

REPERE	SYMBOLE	COMMENTAIRE
%C2		
%I2.5		detecteur comptage arriere p1

CHART - PAGE1 %X11 P1



Liste de Variables utilisées dans le rung :

REPERE	SYMBOLE	COMMENTAIRE
%MW101		
%KW3		
%M6		image de sortie commande arriere
%Q3.3		commande arriere p1

Auteur :	4.2.1 Sections	Imprimé le 01/07/2008
Service :	4.2.1.1 Pont_1	Indice :
Automate cible : TSX 57203		Folio : 4.2.1.1 - 11

Ce document est la propriété de la société XXX et ne peut être reproduit ou communiqué sans son autorisation.

VI. Conclusion

Le programme que nous avons développé respecte l'exactitude du modèle GRAFCET il est constitué de deux compteurs qui comptent le nombre de bains et d'un nombre de comparateurs égal au nombre de bains, ce qui permet de programmer l'arrêt des portiques au bains voulus en comparant à chaque fois le chiffre donné par les compteurs au numéro du bain où les portiques sont sensés s'arrêter.

Ce programme et le fonctionnement de toute l'installation sont susceptibles de plusieurs améliorations ce qui fera l'objet du chapitre qui suit.

I. Introduction

Aujourd'hui il sera difficile de concevoir un système de production sans avoir recours aux différentes technologies et composants formant le système de production automatisé.

La conception d'un dispositif de commande est d'autant plus difficile que le système à automatiser est complexe et peut donc présenter certaines imperfections.

La commande de l'installation de zingage n'est pas optimale, elle présente certaines insuffisances susceptibles d'être améliorées tel que le temps de cycle minimum et le dispositif de sécurité surtout par rapport au personnel et ce fera l'objet de ce chapitre.

II. Proposition pour l'amélioration du temps du cycle de travail

Actuellement le temps de cycles des portiques est de 6mn et 48s, le temps d'émersion des pièces dans différents bains ne peut être changé afin de satisfaire les exigences de qualité.

Les attentes que fait le premier portique au niveau des bains 2, 8, 11 et au niveau des bains 7, 12, 18 pour le deuxième portique ne peuvent être ni supprimées ni diminuées sinon il y aura collision entre les deux portiques. Ces temps sont calculés lors de la programmation d'origine.

Le gain de temps peut être obtenu soit par l'augmentation de la vitesse de translation et pas celle de soulèvement car les pièces risquent de tomber dans les bains soit par la suppression des temps d'attente que font les portiques au niveau de chaque bain lors de la charge ou de la décharge.

III. L'inconvénient de la commande des moteurs de translation et de levage par un seul variateur de vitesse

L'inconvénient que présente ce type de commande est que les vitesses de translation des portiques que ce soit la grande ou la petite vitesse sont les mêmes que celle de soulèvement car ces moteurs sont alimentés à partir des mêmes bornes via le même variateur de vitesse comme le montre la figure V.1.

L'autre inconvénient de cette commande, est qu'un variateur de vitesse se programme pour commander des moteurs dont leurs caractéristiques sont bien spécifiées, car lors de sa programmation, celles-ci (telles que la vitesse de rotation nominale et le courant nominal) seront introduites pour qu'elles soient prises en charge par le variateur. Dans notre cas, les caractéristiques des deux moteurs sont différentes : $N_n = 1400$ tr/mn et $I_n = 0,83$ A pour le moteur de translation et $N_n = 1350$ tr/mn et $I_n = 1,91$ A pour le moteur de levage.

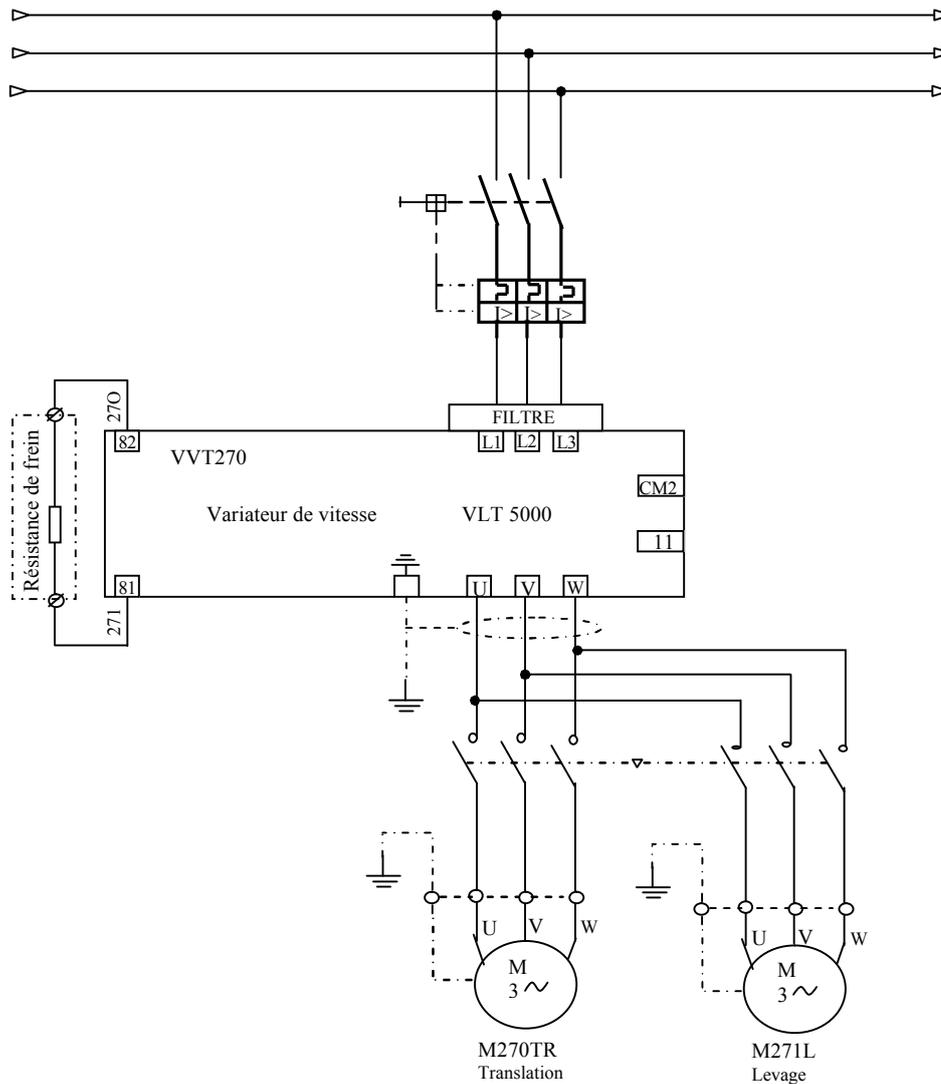


Figure V.1. Schéma de commande de deux moteurs par un seul variateur de vitesse.[14]

IV. La nécessité de l'ajout d'un variateur de vitesse

La commande des vitesses de translation des portiques et celle de levage doivent être indépendantes l'une de l'autre ; afin de pouvoir jouer sur le temps de translation sans toucher à celui de levage et vis versa. Pour se faire l'ajout d'un deuxième variateur de vitesse s'avère nécessaire. Ce ci nous permet aussi d'éviter le temps de commutation qui est marqué par une attente de trois secondes que prend le variateur pour exciter un moteur juste après la désexcitation de l'autre surtout quand il s'agit de changer le sens de rotation.

V. Rappel sur les principales fonctions du variateur de vitesse électronique [9]

V.1. accélération contrôlée

La mise en vitesse du moteur est contrôlée au moyen d'une rampe d'accélération généralement linéaire. Cette rampe est réglable et permet par conséquent de choisir le temps de mise en vitesse appropriée à l'application.

V.2. décélération contrôlée

Quand un moteur est mis hors tension, sa décélération est due uniquement au couple résistant de la machine (décélération naturelle). Les variateurs électroniques permettent de contrôler la décélération au moyen d'une rampe linéaire généralement indépendante de la rampe d'accélération. Cette rampe peut être réglée de manière à obtenir un temps de passage de la vitesse en régime établi à une vitesse intermédiaire ou nulle :

- si la décélération désirée est plus rapide que la décélération naturelle le moteur doit développer un couple résistant qui vient s'ajouter au couple résistant de la machine, on parle alors de freinage électrique qui peut s'effectuer par renvoi d'énergie au réseau d'alimentation soit par dissipation dans une résistance de freinage.
- Si la décélération désirée est plus lente que la décélération naturelle, le moteur doit développer un couple moteur supérieur au couple résistant de la machine et continuer à entraîner la charge jusqu'à l'arrêt.

V.3. Variation de vitesse

La génération de la tension à la sortie est obtenue par la découpage de la tension redressée en moyen d'impulsion dont la durée, donc la largeur, est modulée de telle manière que le courant alternatif résultant soit aussi sinusoïdale que possible. Cette technique connue sous le nom de MLI conditionne la rotation régulière à basse vitesse et limite les échauffements. La fréquence de modulation retenue est un compromis : elle doit être suffisamment élevée pour réduire l'ondulation du courant et le bruit acoustique dans le moteur sans augmenter notablement les pertes dans le pont onduleurs et dans les semi conducteurs.

V.4. inversion du sens de marche

Généralement, les variateurs permettent cette fonction en standard. L'inversion de l'ordre des phases d'alimentation du moteur est réalisée automatiquement soit par inversion de la consigne de l'entrée soit par un ordre logique sur une borne.

V.5. Freinage d'arrêt

Ce freinage consiste à arrêter un moteur sans pour autant contrôler la rampe de ralentissement. Pour les variateurs de vitesse des moteurs asynchrones ceci est réalisé d'une manière économique en injectant du courant continu dans le moteur avec un fonctionnement particulier de l'étage de puissance. Toute l'énergie mécanique est dissipée dans le rotor de la machine et de ce fait ce freinage ne peut être qu'intermittent.

V.6. Fonctionnement en U/F constant

Pour les moteurs asynchrone, le variateur de fréquence alimenté à tension et fréquence fixe par le réseau, assure au moteur en fonction des exigences de vitesses, son alimentation en courant alternatif à tension et fréquence variables.

Pour alimenter convenablement un moteur asynchrone à couple constant quelque soit la vitesse, il est nécessaire de maintenir le flux constant. Ceci nécessite que la tension et la fréquence évoluent simultanément et dans les mêmes proportions. Dans ce type de fonctionnement la référence vitesse impose une fréquence à l'onduleur et par voie de conséquence au moteur, la tension d'alimentation est en relation directe avec la fréquence.

VI. Choix du variateur de vitesse [10]

Le choix d'un variateur de vitesse étant intimement lié à la nature de la charge entraînée et aux performances visées, toute définition et recherche d'un variateur de vitesse doit passer par une analyse des exigences fonctionnelle et l'équipement, puis des performances requise pour le moteur lui-même.

A la mise en service, le metteur au point de la machine doit notamment introduire les caractéristiques plaquées sur le moteur dont les paramètres de réglage du variateur tels que tension nominale, fréquence nominale, courant nominal, vitesse nominale, cosinus moteur.

Comme les moteurs de translation et de levage sont différents, on a à choisir un variateur qui s'adapte pour chacun d'eux, donc deux types différents pour chaque portique.

Notre choix est porté sur quatre variateurs télémécanique de type ALTIVAR 31, de référence ATV31CO37N4 pour les moteurs de translation et ATV31CO75N4 pour les moteurs de levage.

Les caractéristiques électriques sont données dans le tableau suivant :

	ATV31CO37N4	ATV31CO75N4
Puissance	0,37 KW	0,75 KW
Courant du réseau (entrée)	2,2 A	3,6 A
Puissance apparente	1,5 KVA	2,4 KVA
Courant d'appel Max	10 A	10 A
Icc ligne présumé	5 KA	5 KA
Courant nominale In	1,5 A	2,3 A
Courant transitoire Max	2,3 A	3,5 A
Puissance dissipée à charge nominale	32 W	41 W

Le courant d'appel max est le courant de pointe à la mise sous tension.

Le courant transitoire max est supporté pendant 60 secondes.

VII. Présentation du variateur de vitesse électronique ATV31 [11]



Figure V.2. Variateur de vitesse ATV31.

VII.1. Schéma de raccordement du variateur ATV31

Le schéma en figure V.2 présente les différentes entres et sorties de ce variateur.

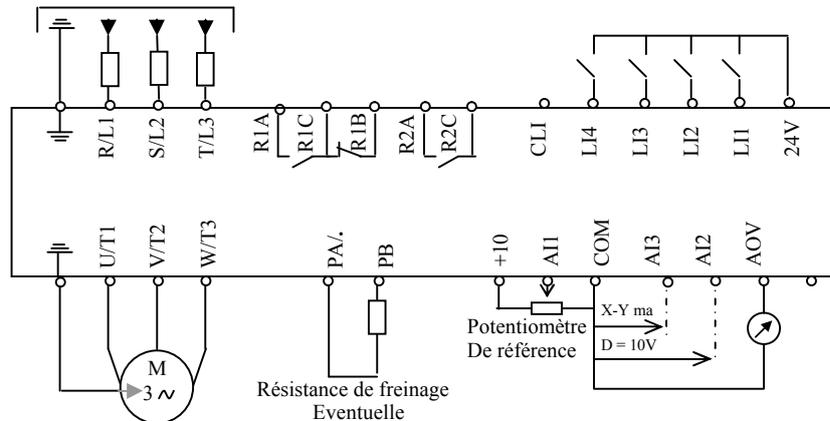


Figure V.3. Schéma de raccordement du variateur ATV31.

VII.2. Fonction des bornes de contrôle

- LI1, LI2, LI3, LI4 : entrées logiques.
- CLI : commun des entrées logiques.
- R1A, R1B, R1C : contact OF à point commun (R1C) du relais programmable R1.
- R2A, R2C : contact à fermeture du relais programmable R1.
- +10V : alimentation pour le potentiomètre de consigne (1 à 10kv).
- AI1, AI2 : entrées analogiques en tension.
- AI3 : entrées analogique en courant.
- AOV : sortie analogique en tension.
- AOC : sortie analogique en courant.

AOV e AOC, sont des sorties affectables et peuvent être utilisées comme sorties logique en tension (l'une ou l'autre mais pas les deux).

VII.3. Fonction de l'afficheur et des touches

La gamme de variateur ALTIVAR 31 possède un clavier avec écran permettant la saisie des différents paramètres des moteurs et de visualiser certains (fréquence et vitesse) pendant le fonctionnement. Cet écran est accessible en mode marche et en mode arrêt, mais on ne peut valider une modification que s'il est en arrêt.

Le schéma de figure V.4 présente l'afficheur et les différentes fonctions de chacune des touches du clavier.

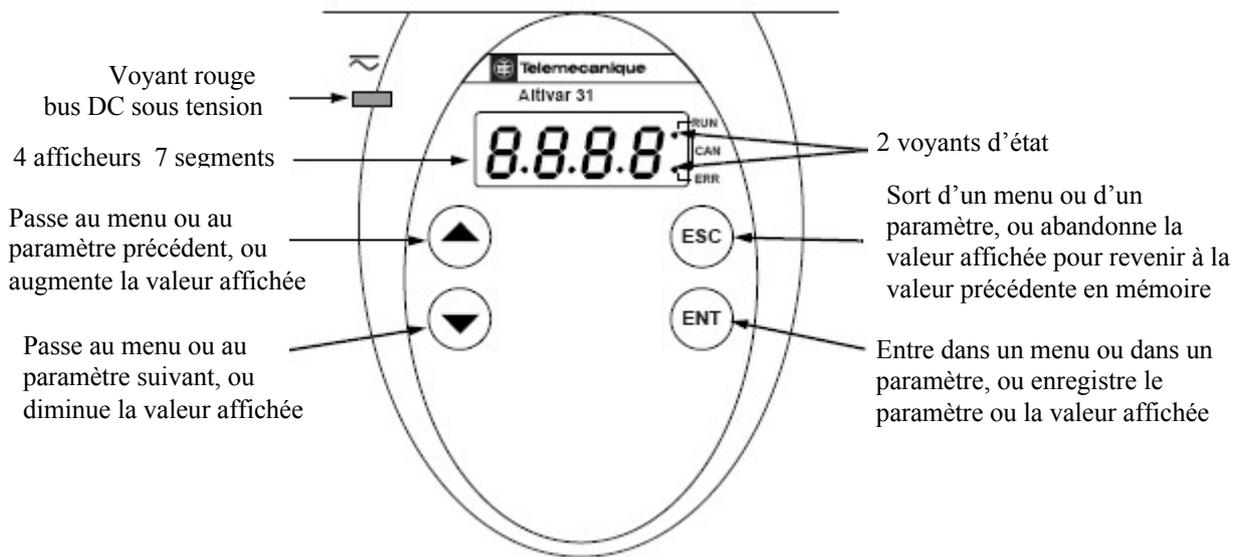


Figure V.4. Clavier et écran d'un ALTIVAR 31

VII.4. Préréglages

L'Altivar 31 est préréglé pour les conditions d'emploi les plus courantes :

- Affichage : variateur prêt, moteur à l'arrêt, et fréquence moteur en marche.
- Application à couple constant, contrôle vectoriel de flux sans capteur (UFt = n).
- Mode d'arrêt normal sur rampe de décélération.
- Mode d'arrêt sur défaut : roue libre
- Rampes linéaires.
- Petite vitesse (LSP).
- Grande vitesse (HSP).
- Courant thermique moteur (I_{tH}) = courant nominal moteur (valeur selon calibre du variateur).
- Courant de freinage par injection à l'arrêt.
- Adaptation automatique de la rampe de décélération en cas de surtension au freinage.
- Pas de redémarrage automatique après un défaut.
- Fréquence de découpage 4 kHz.
- Entrées logiques :
 - LI1, LI2 (2 sens de marche) : LI1 c'est la marche avant, LI2 c'est la marche arrière.
 - LI3, LI4 : 4 vitesses présélectionnées.
- Entrées analogiques :

- AI1 : consigne vitesse 0-10 V.
- AI2 : entrée sommatrice vitesse 0 ± 10 V.
- AI3 : 4-20 mA inactive (non affectée).
- Relais R1 : le contact s'ouvre en cas de défaut (ou variateur hors tension)
- Relais R2 : inactif (non affecté).
- Sortie analogique AOC : 0-20 mA, inactive (non affectée).

VII.5. Vitesses présélectionnées

2, 4, 8, ou 16 vitesses peuvent être présélectionnées, nécessitant respectivement 1, 2, 3 ou 4 entrées logiques.

L'ordre des affectations à respecter est le suivant : PS2, puis PS4 puis PS8, puis PS16.

Tableau de combinaison des entrées de vitesses présélectionnées est le suivant :

16 vitesses LI (SP16)	8 vitesses LI (SP8)	4 vitesses LI (SP4)	2 vitesses LI (sp2)	Consigne de vitesse
0	0	0	0	SP1
0	0	0	1	SP2
0	0	1	0	SP3
0	0	1	1	SP4
0	1	0	0	SP5
0	1	0	1	SP6
0	1	1	0	SP7
0	1	1	1	SP8
1	0	0	0	SP9
1	0	0	1	SP10
1	0	1	0	SP11
1	0	1	1	SP12
1	1	0	0	SP13
1	1	0	1	SP14
1	1	1	0	SP15
1	1	1	1	SP16

VIII. La mise en œuvre de l'amélioration proposée

Les deux moteurs de chaque portique seront commander indépendamment l'un de l'autre par deux variateurs différents, eux même seront gérés par l'automate programmable.

Les variateurs de vitesse ne reçoivent de l'automate que les consignes de mise en marche ou à arrêt des moteurs.

L'accélération et la décélération ainsi que les vitesses nominales seront programmée par les variateurs ce qui donne la possibilité de différencier la vitesse de soulèvement de celle de la translation.

VIII.1. Schéma de raccordement des variateurs à l'API pour un portique

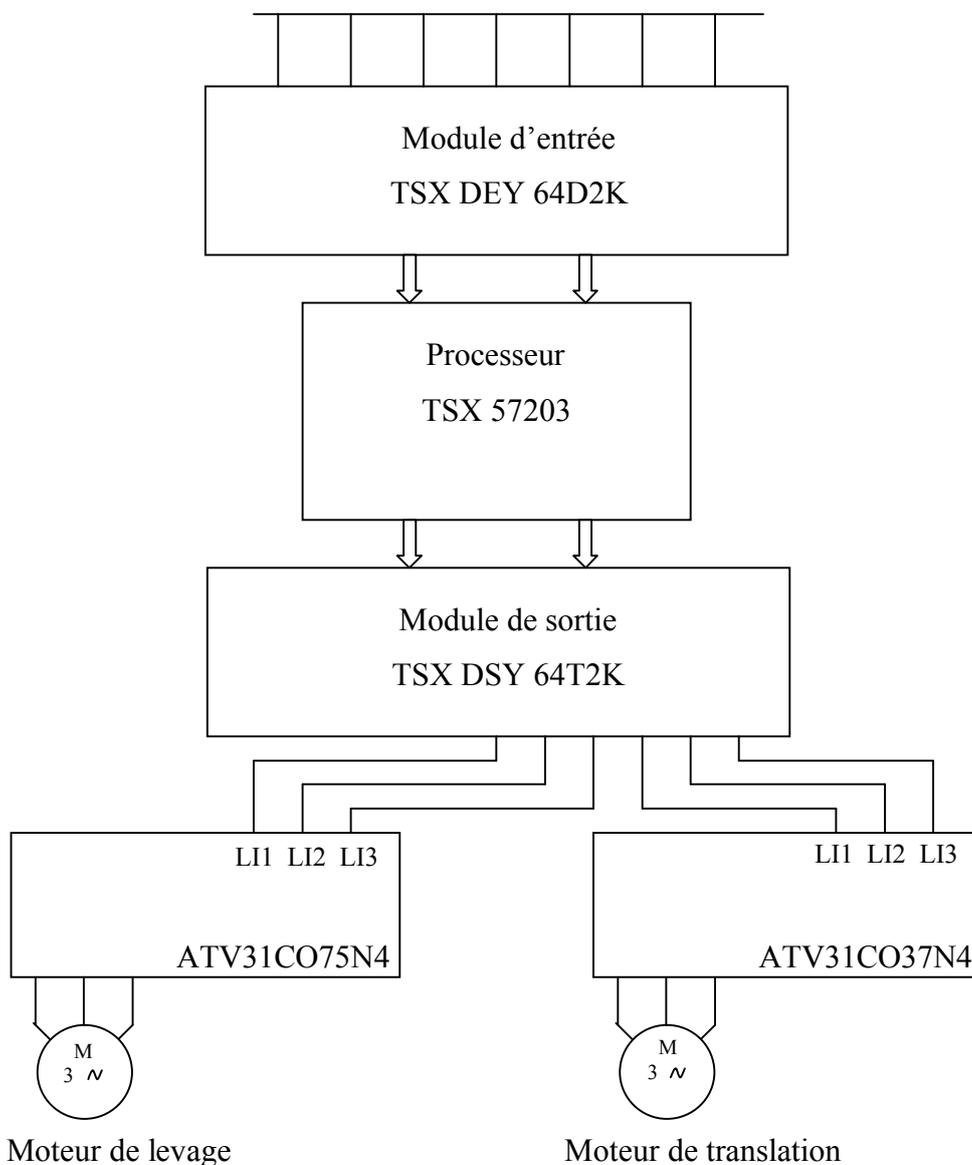


Figure V.5. Schéma de raccordement de l'automate aux variateurs

Dans ce type de montage on voit bien que les commandes des deux moteurs sont indépendantes l'une de l'autre, ceci permet d'actionner l'un sans l'autre ou les deux simultanément, ce qui n'est pas possible avec un seul variateur.

Cela nous offre la possibilité de gagner du temps lors de chaque levage en faisant démarrer le moteur de translation dès que celui du levage commence à décélérer.

VIII.2. Elaboration du GRAFCET

L'utilisation d'un variateur de vitesse pour chacun des moteurs de levage et de translation nous permet de supprimer les attentes que font les portiques lors de chaque charge et décharge.

A la différence du Grafcet conçu dans le chapitre III, celui-ci ne possède pas de macro étapes de charge et de décharge, car celles-ci deviennent une seule étape.

Le Grafcet précédent possède 109 étapes et 27 macros étapes de trois étapes chacune, ce qui nous donne en tout 163 étapes tandis que le Grafcet conçu pour cette amélioration ne contient que 106 étapes, la minimisation du nombre d'étapes rend le programme moins complexe.

IX. Autres propositions

IX.1. Coté sécurité

Cette installation présente un danger pour le personnel. En effet lors de la translation des portiques sur les guides, si un ouvrier pénètre dans le champ de leurs déplacements (pour l'inspection des bains et la maintenance de l'installation en plein fonctionnement) les portiques ne reconnaissent pas la présence de celui-ci, ce qui provoquera sa percutions.

Pour éviter ce genre d'accidents, l'ajout de nouveaux capteurs pourra y remédier à ce problème.

IX.1.1. Choix du capteur

Comme le choix du capteur se base sur le principe de détection et la nature de l'évènement à détecter, notre choix est porté sur un détecteur photoélectrique.

Pour plus de performance, le capteur ne détectera la présence d'une personne qu'à une distance bien spécifiée du portique de manière à ce que l'ouvrier se trouvant sur les guides des chariots ne soit détecté, ce qui provoquera l'arrêt du portique, que s'il est proche de l'un des portiques, environ 40cm.

Le capteur répondant à ces exigences est un détecteur de proximité représenté ci-après.

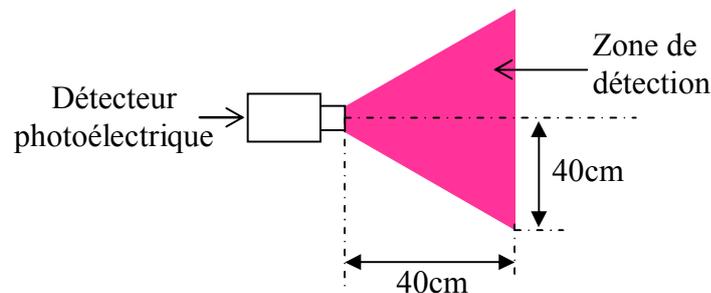


Figure V.6. *Système de proximité.*

IX.1.2. Caractéristiques du capteur [4]

Type d'émission	Infrarouge
Portée nominale	0,40M
Référence	XUM-LH 4055 (télémechanique)
Masse	80g
Température	-25°C ---- +70°C
Mode de raccordement	Par câble
Tension d'alimentation	24V
Limite de tension	10 – 30V avec protection contre la surtension
Courant commuté	< 100ma avec protection contre les surcharges et le court-circuit
Fonctionnement	Présence d'objet dans le faisceau

IX.1.3. Positionnement des capteurs à ajouter

Afin de détecter toute présence de personne sur le guide des portiques, 4 capteurs seront placés sur les bras des portiques, un de chaque côté à une hauteur de 80 cm, ce-ci offre un meilleur champ de détection.

Le schéma suivant illustre le positionnement de ces capteurs.

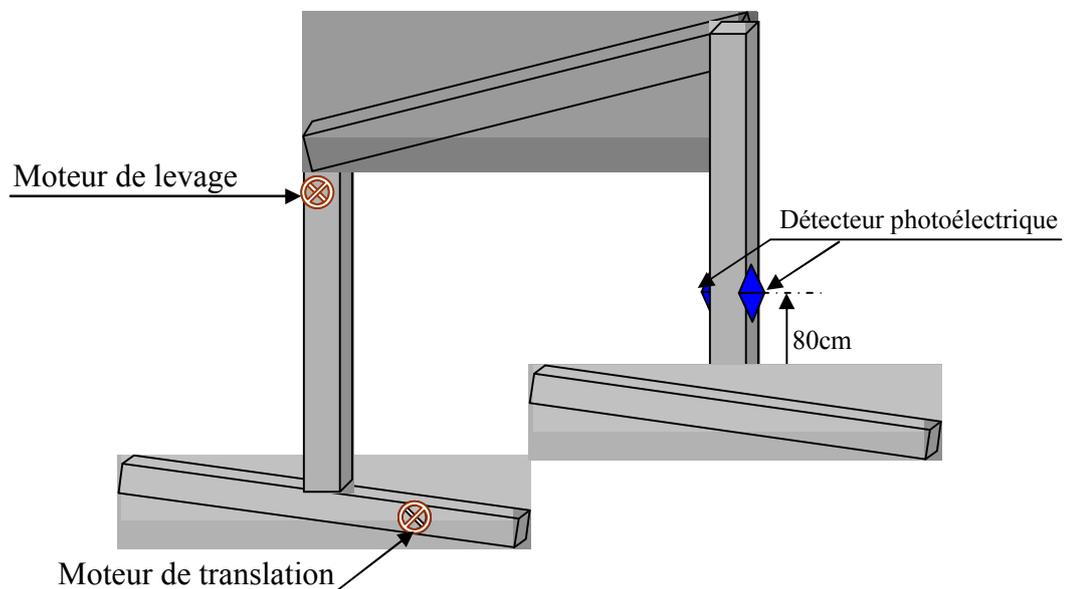


Figure V.7. Positionnement des nouveaux capteurs.

IX.1.4. Câblage des nouveaux capteurs

La détection de ces capteurs d'une personne près des portiques en état de translation présente un danger pour celle-ci, ce qui nécessite l'arrêt immédiat du chariot ; donc ces détecteurs doivent provoquer cet arrêt. Pour se faire, il suffit de les connecter parallèlement à l'un des capteurs de sécurité.

Le détecteur de sécurité le plus proche est celui de surpassement bas.

IX.1.5. Avantage de cette proposition

En plus de la sécurité qu'offre cette amélioration, sa mise en œuvre est très simple et pratique et ne demande aucun changement au niveau du programme automate et du terminal de programmation (XBT), car les entrées et les sorties de l'automate ne seront pas modifiées.

IX.2. Côté performances

L'installation comporte trois baignoires dont la substance contenue doit être chauffée avant le début du traitement des pièces. Ces baignoires sont les suivantes :

- Le dégraissage chimique à 70°C.
- Le rinçage chaud à 50°C.
- Le dégraissage anodique à 40°C.

Vu l'important volume de ces baignoires, le dispositif de chauffage prend un temps considérable pour ramener les substances aux températures nécessaires pour le traitement.

Comme les systèmes de chauffage ne sont pas intégrés dans la partie automatisée de l'installation, un ouvrier est chargé de les mettre manuellement en marche trois heures plutôt (à cinq heures du matin) avant le début de la mise en marche de l'installation.

IX.2.1 Solution proposée

L'idée porte sur la mise en marche des dispositifs de chauffage sans intervention humaine, l'intégrer dans le système automatisé n'est pas une solution pratique car celle-ci nécessite l'intervention au niveau du programme automate ce qui compliquerait la tâche.

L'alimentation des systèmes de chauffage qui n'est autre que des serpentins dont lesquelles circule de l'eau chaude est autonome, elle est assurée par une chaudière centrale allumée tout le temps.

La circulation de l'eau dans les serpentins est contrôlée par des vannes ; leurs ouvertures et fermetures se font manuellement par un ouvrier.

L'ouverture de ces vannes pourra se faire avec un déclencheur automatique réglable qui offre la possibilité de réglage au déclenchement avec précision, c'est-à-dire on peut le mettre en marche à l'heure voulue.

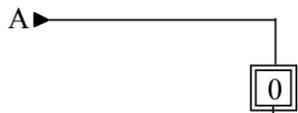
Comme cité ci dessus, les températures des substances des bains sont différentes, un déclencheur automatique sera placé au niveau de chacun des bains à chauffer car l'heure de leur déclenchement n'est pas la même.

X. Conclusion

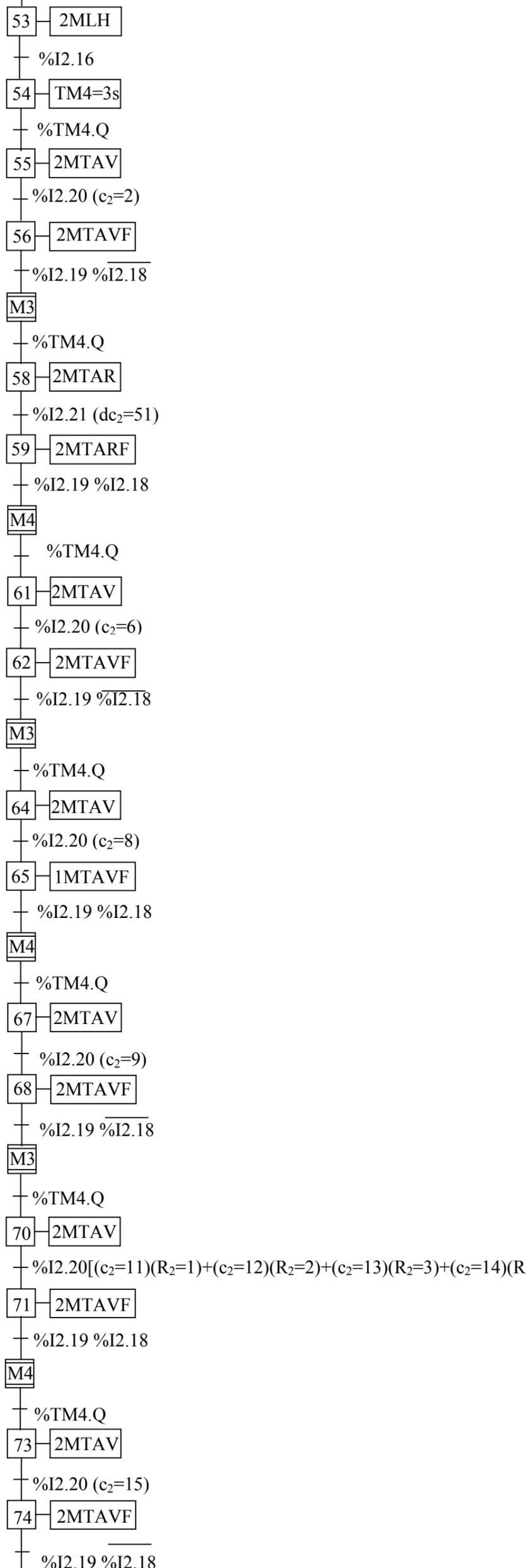
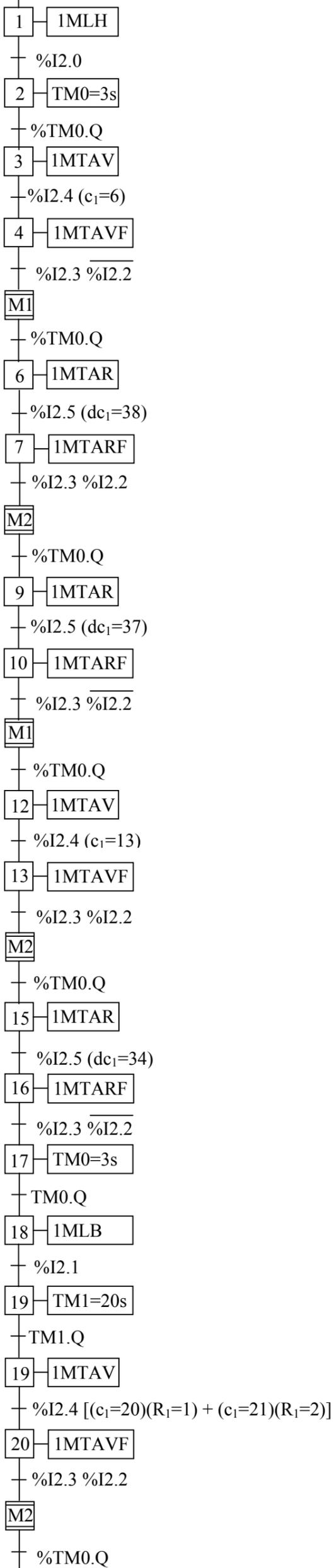
Dans ce présent chapitre nous avons de proposé quelques améliorations pour palier à certaines insuffisances que présente cette installation.

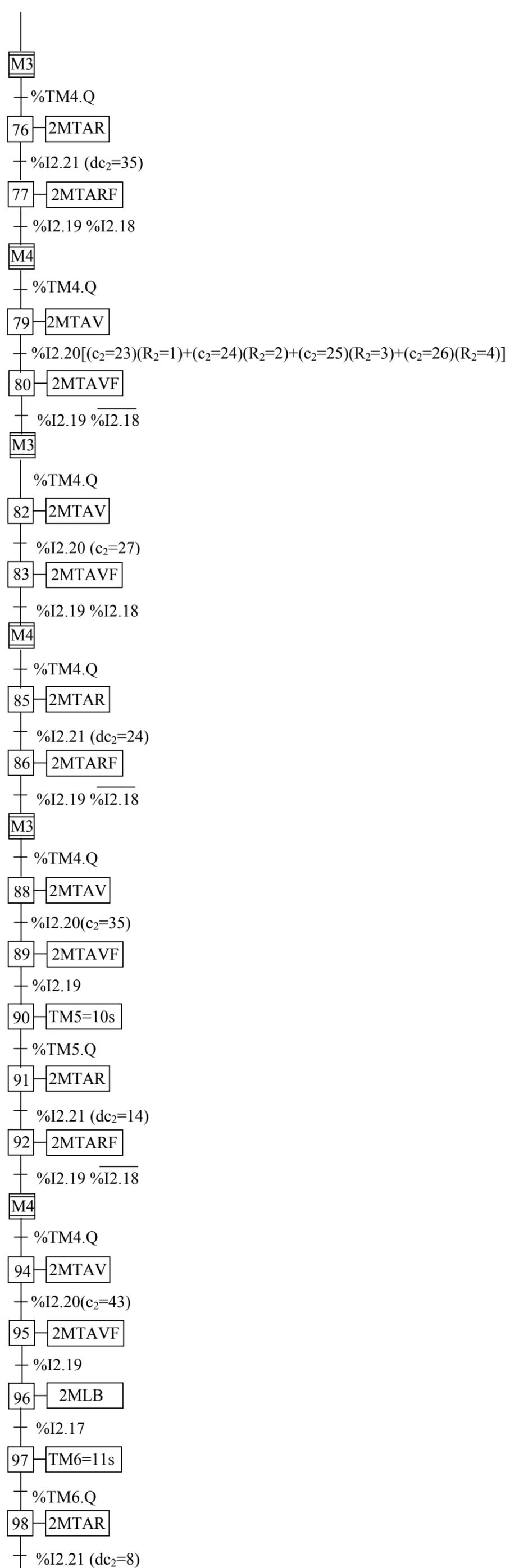
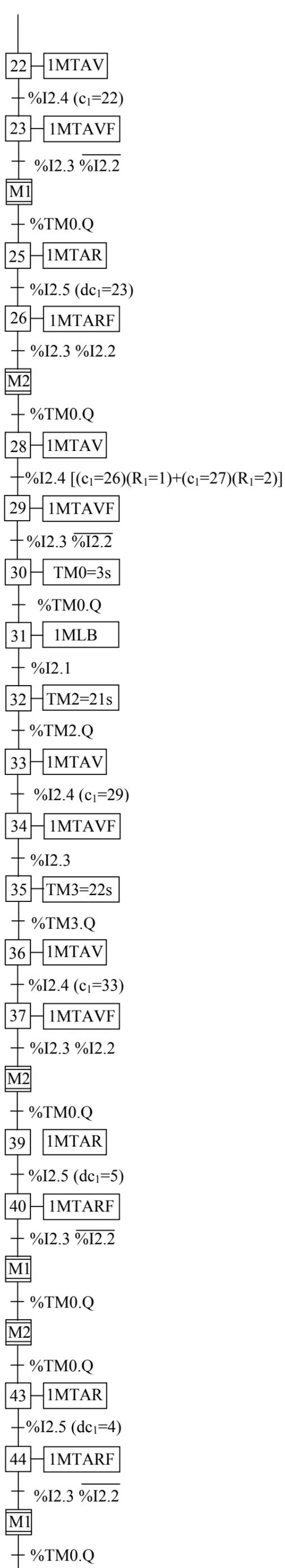
Ces améliorations ne restent que des propositions, mais leur mise en oeuvre et leur adaptation sont faciles et ne nécessite pas de grands moyens à part certains changements au niveau du câblage.

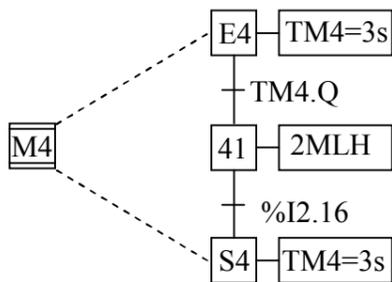
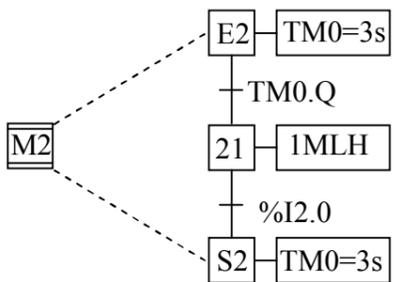
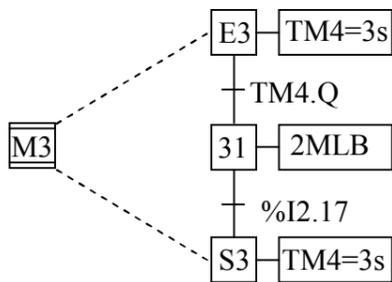
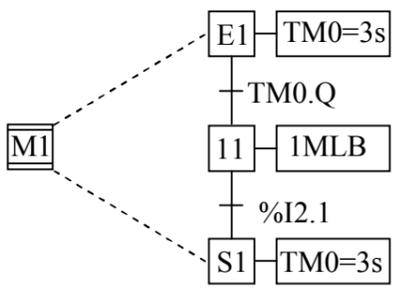
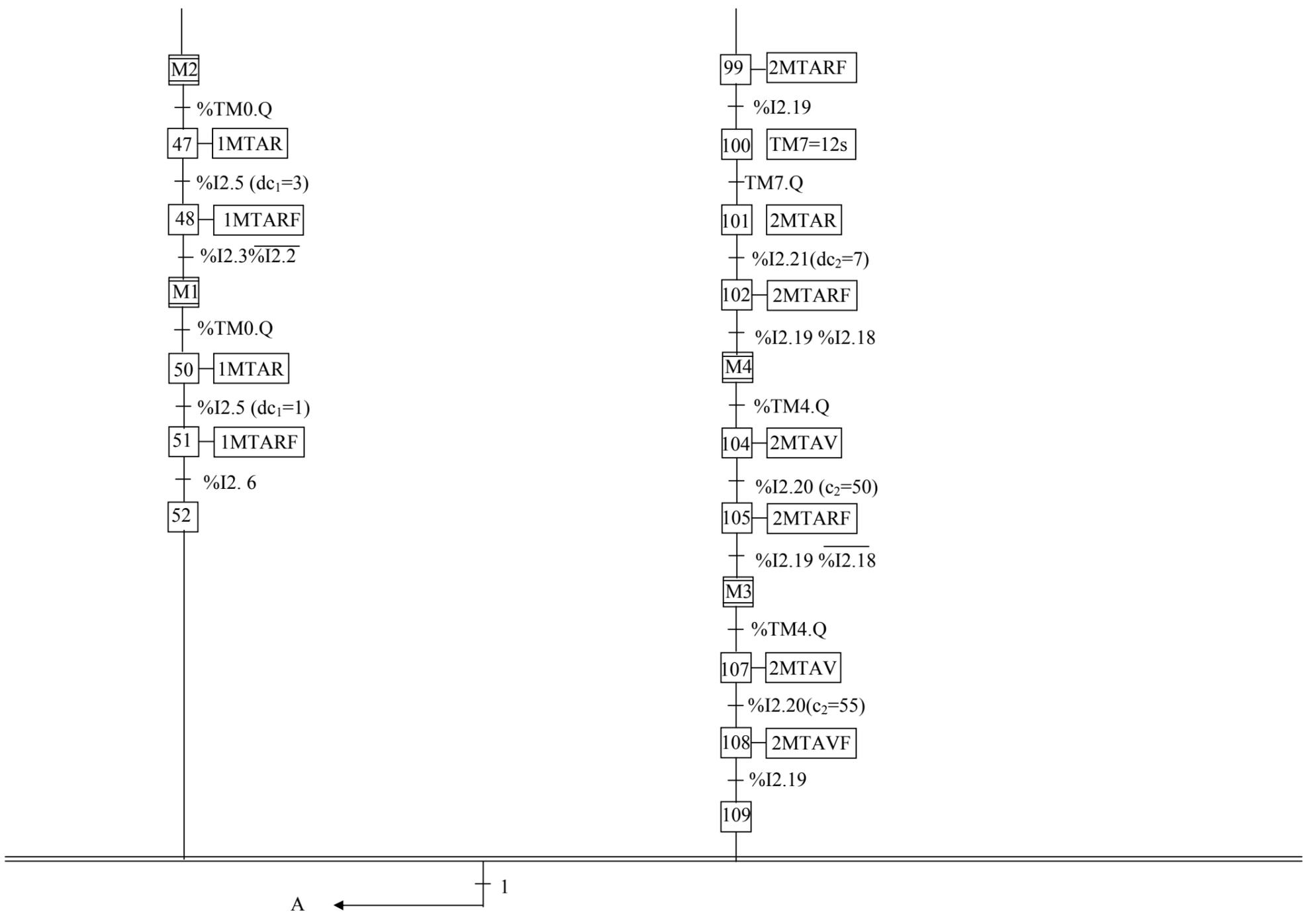
L'ajout des deux variateurs sera compensé du côté dépense par le gain de temps qui se traduira par une augmentation de la production, de plus les variateurs ne sont pas du combustible et ne nécessitent pas d'entretien.



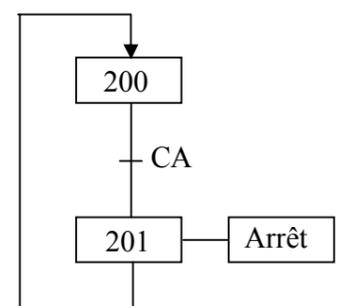
%I2.32 %I2.33 %I2.34 %I2.35 %I2.36 %I2.40





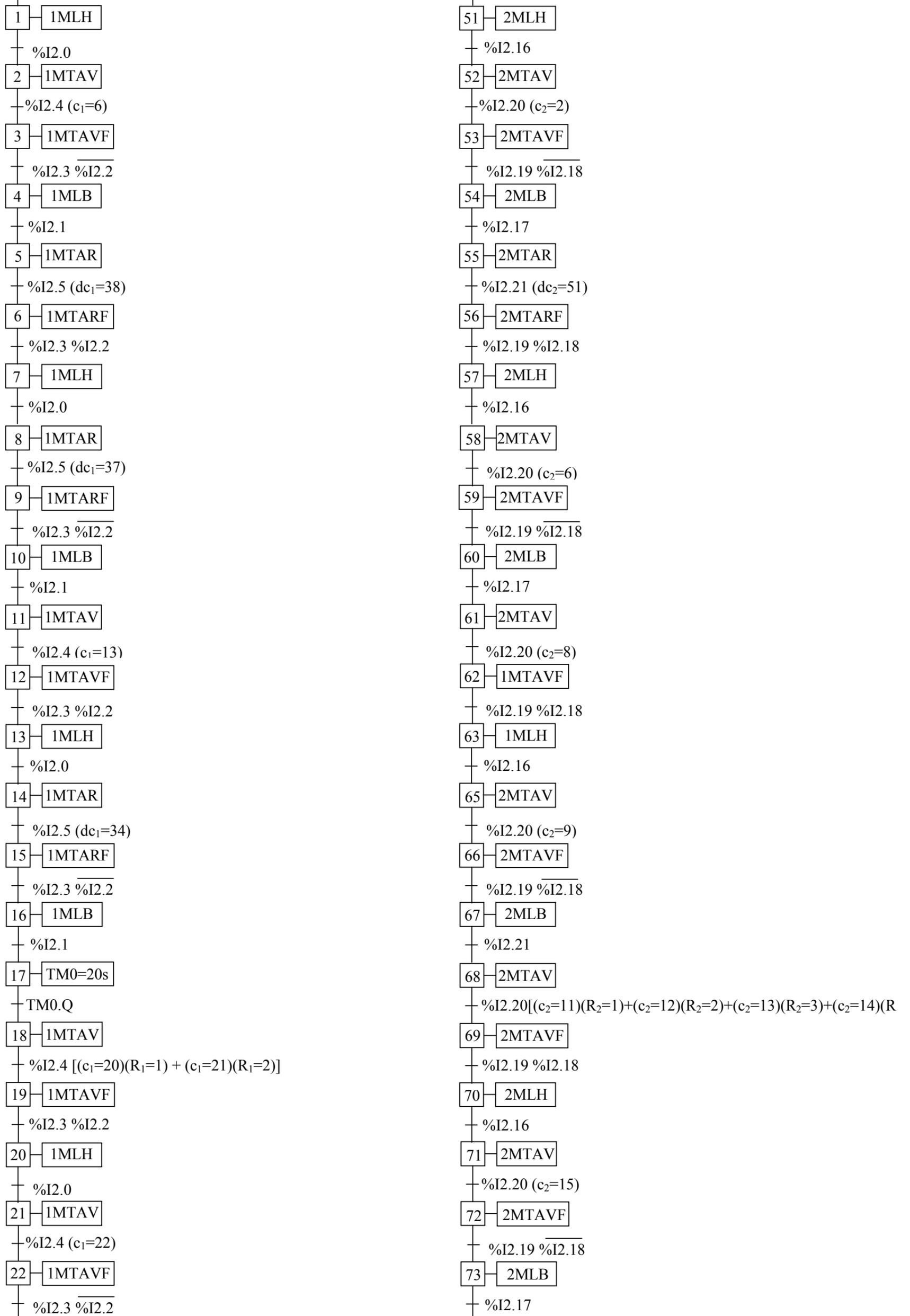


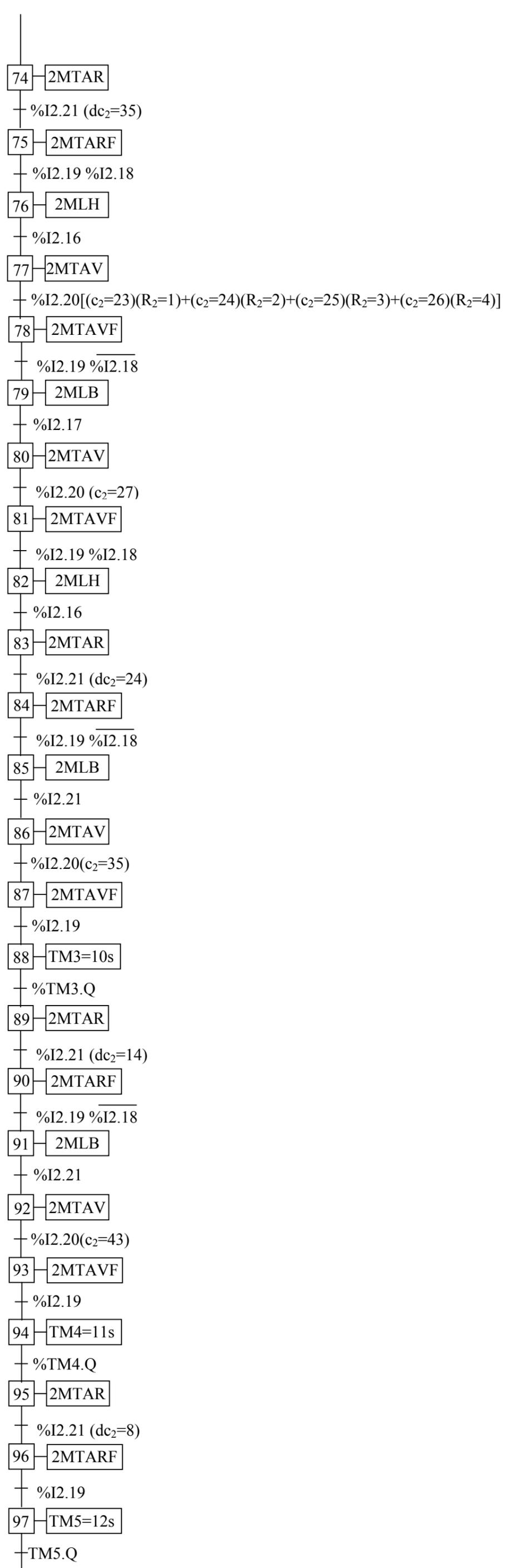
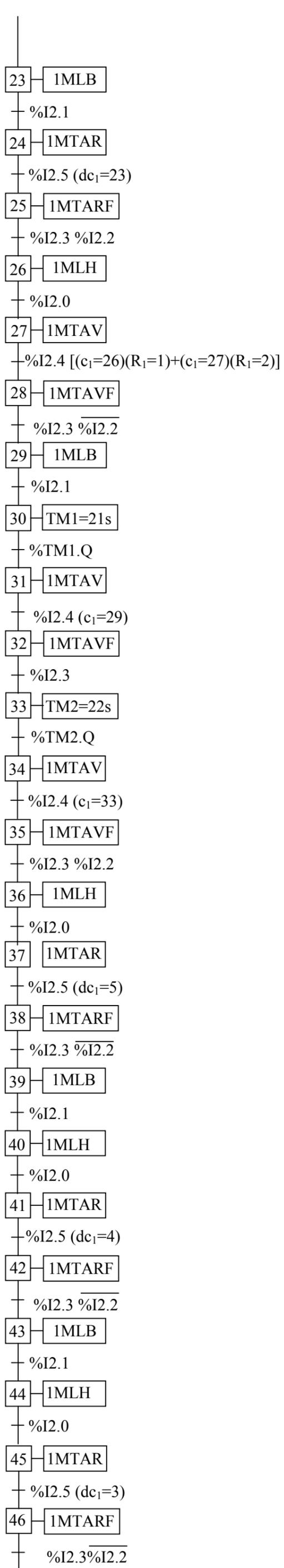
Grafctet de sécurité

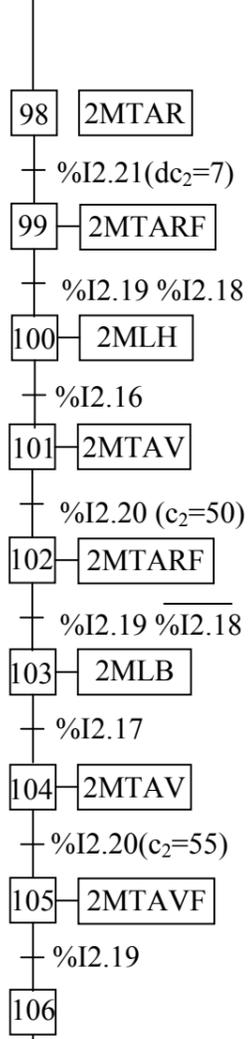
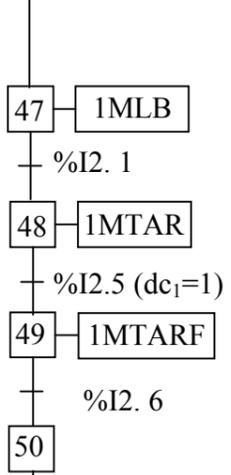




%I2.32 %I2.33 %I2.34 %I2.35 %I2.36 %I2.40







A ← 1

Conclusion générale

Notre travail de fin d'étude a été réalisé en grande partie au sein du complexe électroménager de l'ENIEM, dans le cadre d'un stage pratique de mise en situation professionnelle de trois mois.

En plus de l'étude que nous avons menée dans le cadre de notre projet, ce stage nous a permis de découvrir le monde industriel, d'enrichir nos connaissances sur le plan pratique et le domaine d'automatique et compléter ainsi notre formation théorique universitaire.

Dans ce mémoire, nous avons étudié l'installation de traitement de surface au niveau de l'unité cuisson de l'ENIEM et proposer quelques améliorations. Nous avons procédé pour cela par la compréhension du comportement de la station puis remarquer ses insuffisances et proposer des solutions.

Comme on peut le constater, notre mémoire est divisé en cinq chapitres décrivant l'aboutissement de notre travail.

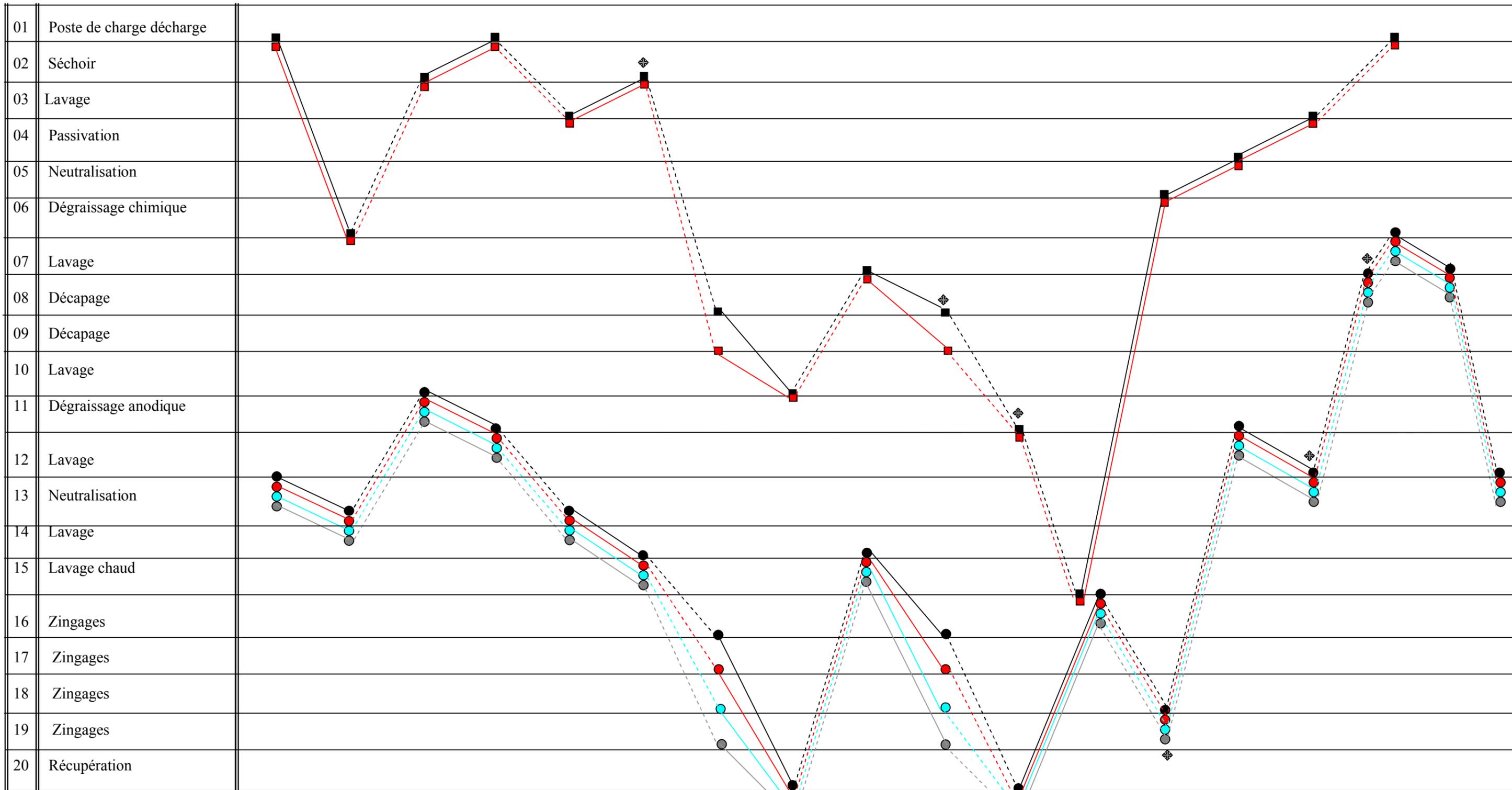
Après avoir donné une description générale sur le fonctionnement de la station et les systèmes automatisés, nous avons modélisé cette dernière en utilisant le Grafcet qui est une méthode d'analyse et de synthèse fiable, simple et permet le passage à la programmation en langage PL7.

Nous avons souhaité configurer et programmer les solutions proposées pour l'amélioration de l'installation, faute de temps et de moyen minime que l'ENIEM a mis notre disposition ceci ne nous a pas été possible.

Vu la complexité de l'installation et le manque de documentation sur les moyens utilisés pour son automatisation, on peut dire qu'il est plus facile de concevoir un nouveau système de commande selon un cahier de charge connu que d'étudier le système déjà conçu.

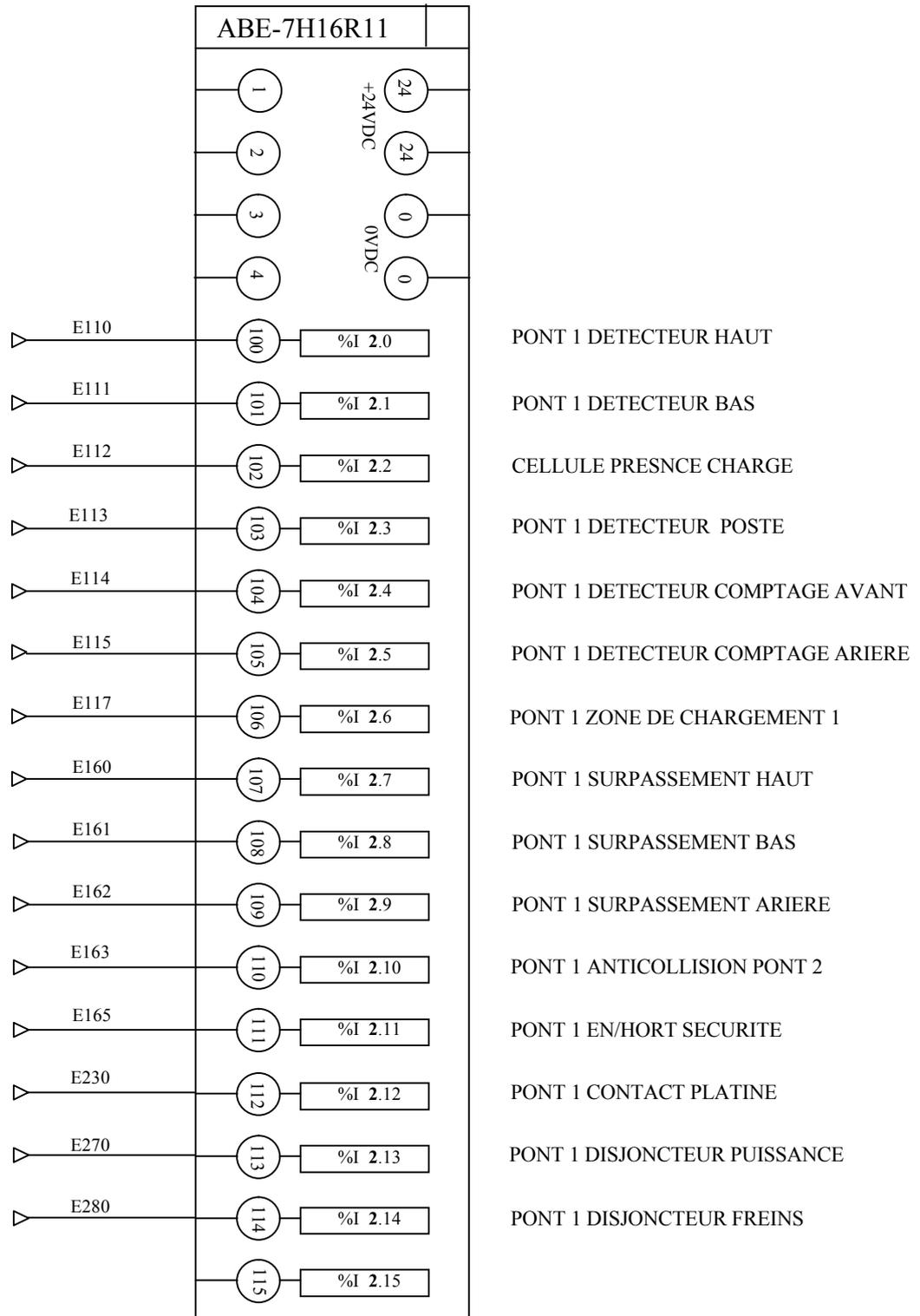
Comme il nous a été d'un grand honneur de conduire ce travail, nous espérons qu'il puisse servir de support aux promotions à venir autant d'avantages qu'il a été pour nous et pourquoi pas pousser plus loin cette étude.

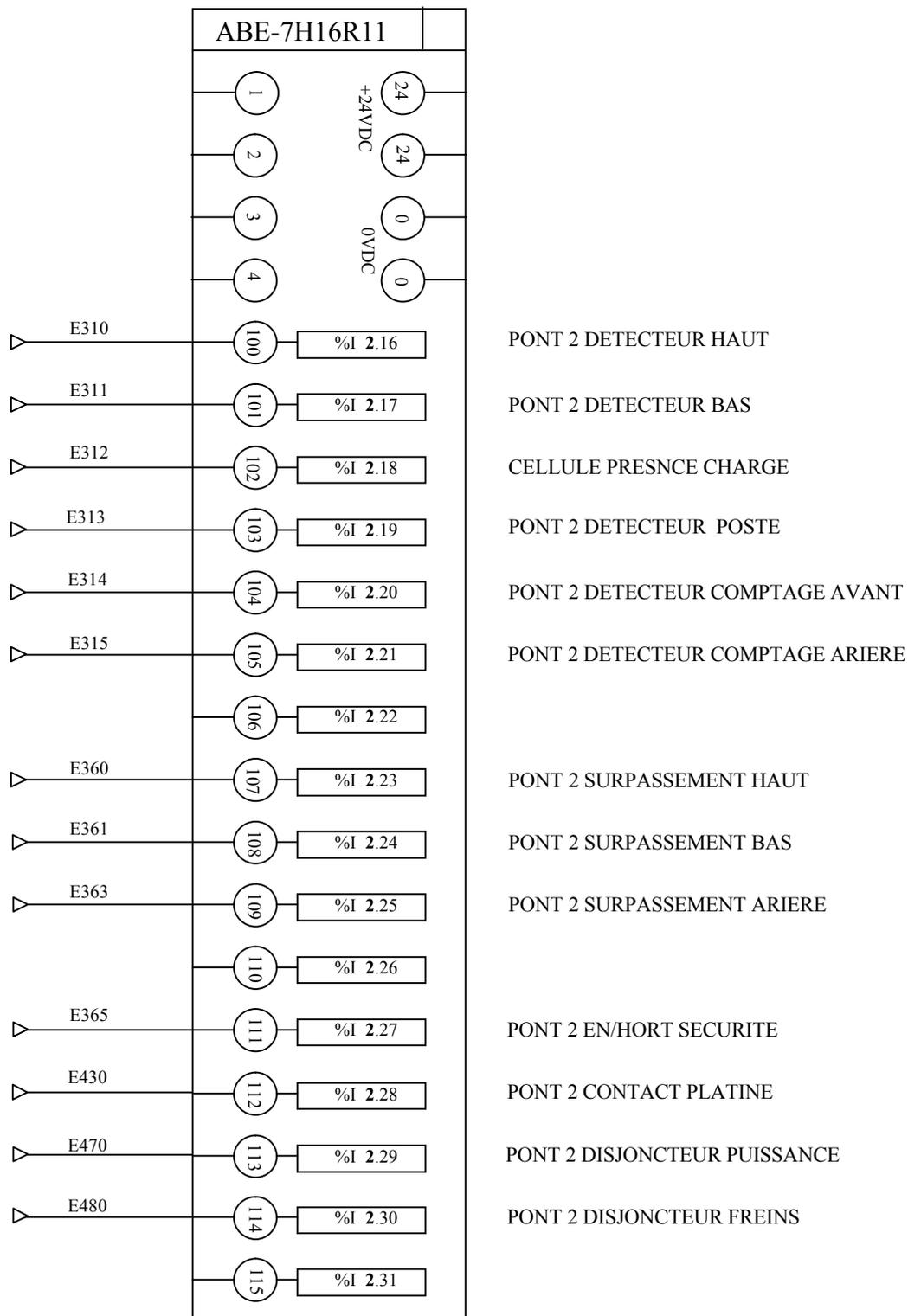
DIAGRAMME DE FONCTIONNEMENT DES DEUX PORTIQUES

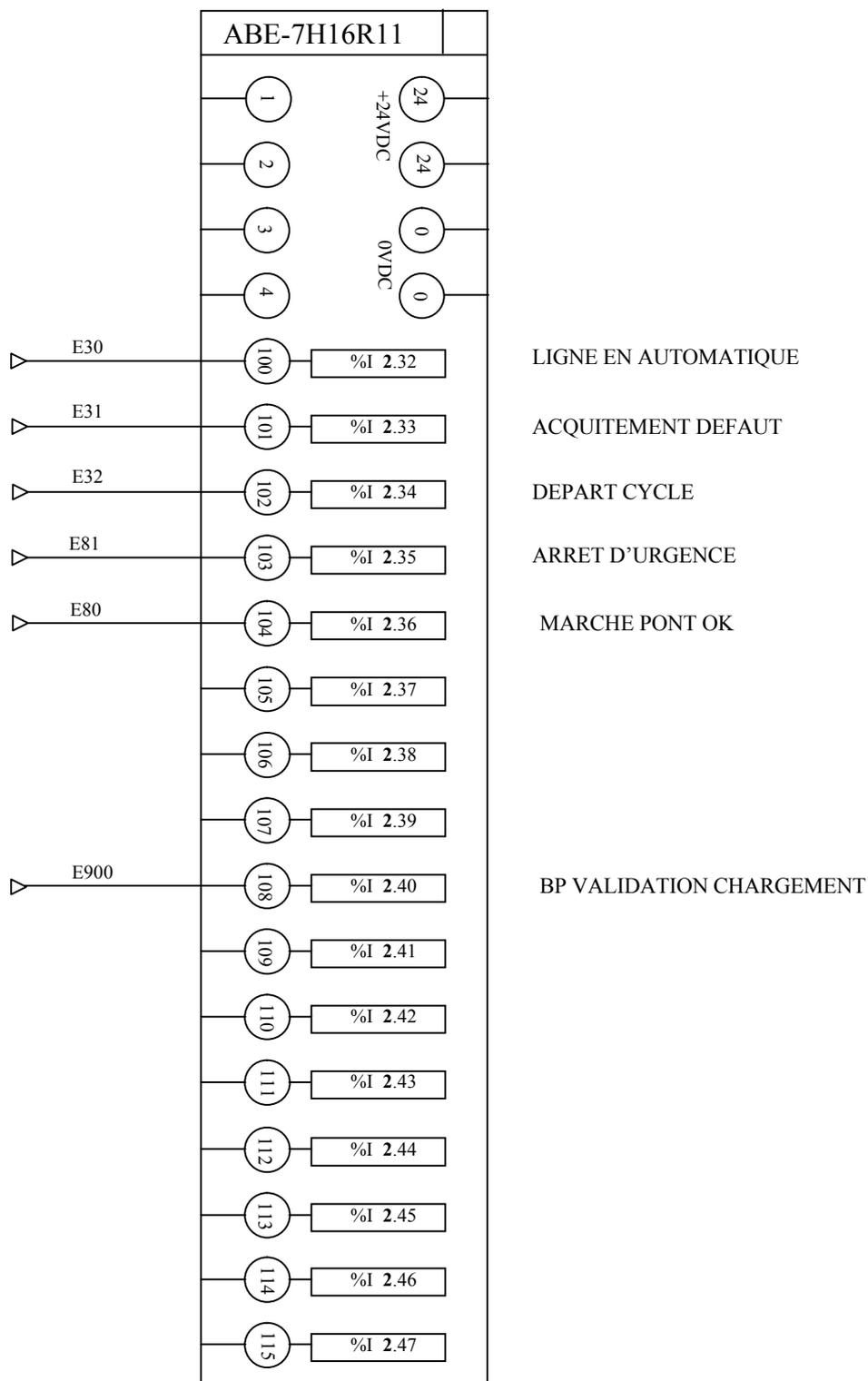


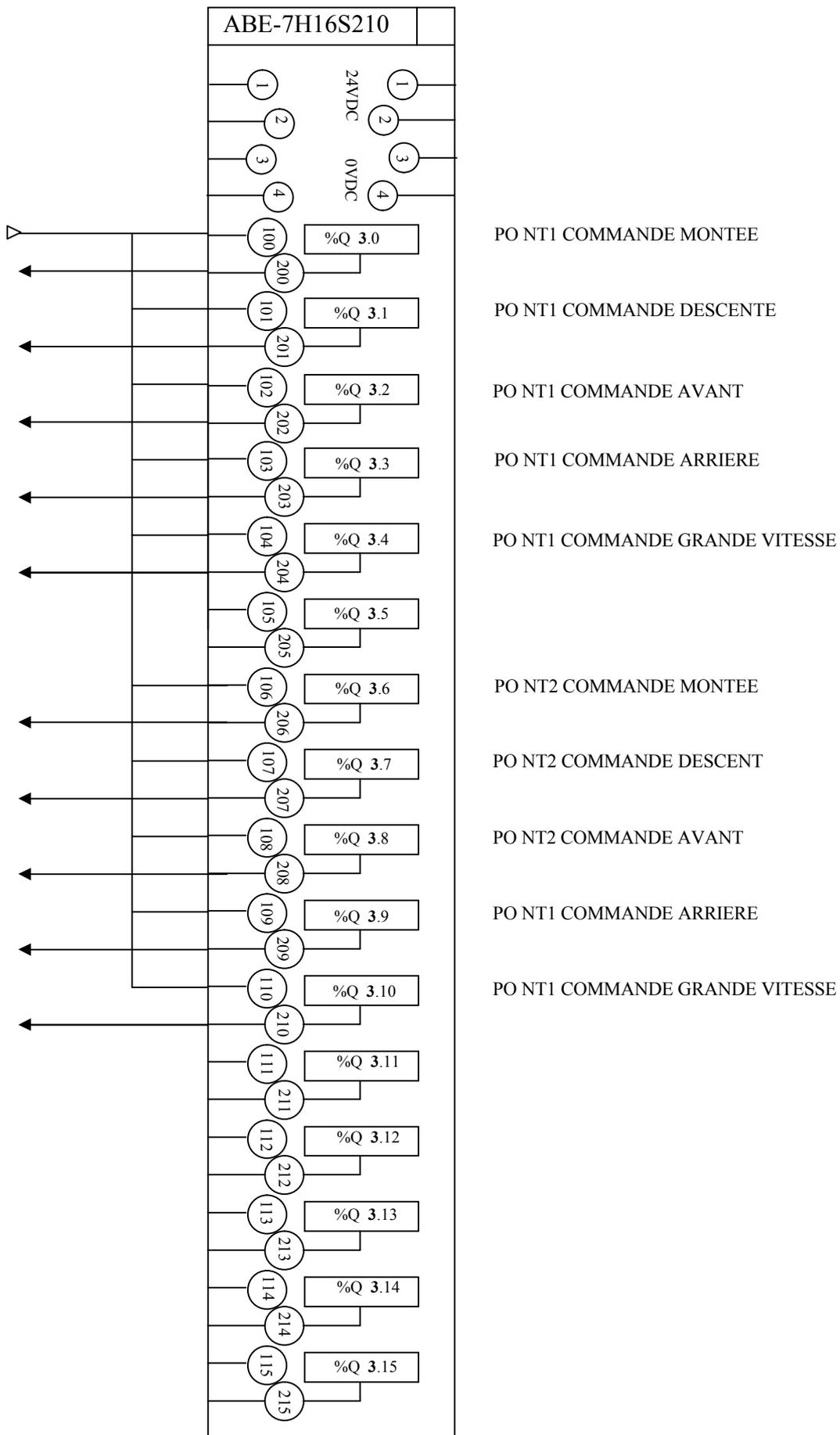
- : Portique I. — : Chargé.
- : Portique II. - - - - : Vide.
- ◆ : Attente.

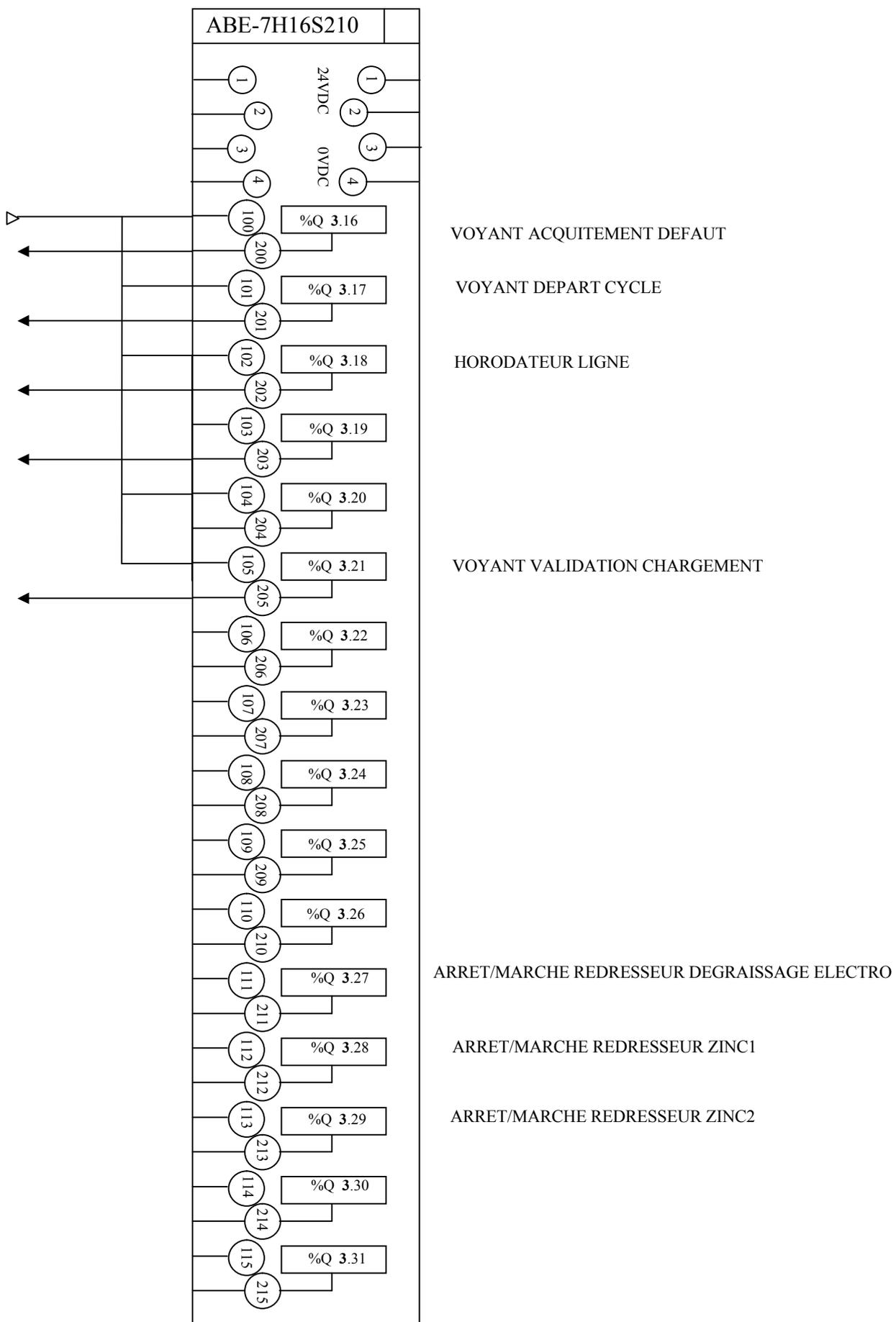
Embase des portées



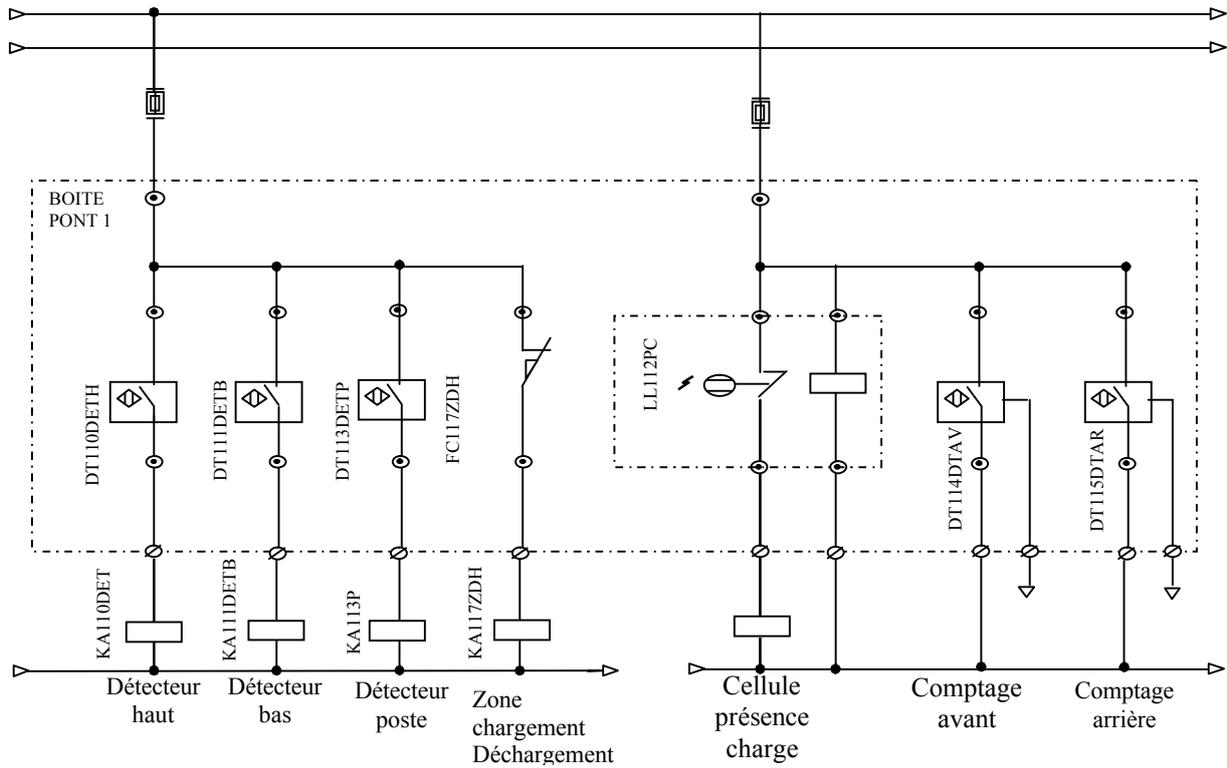




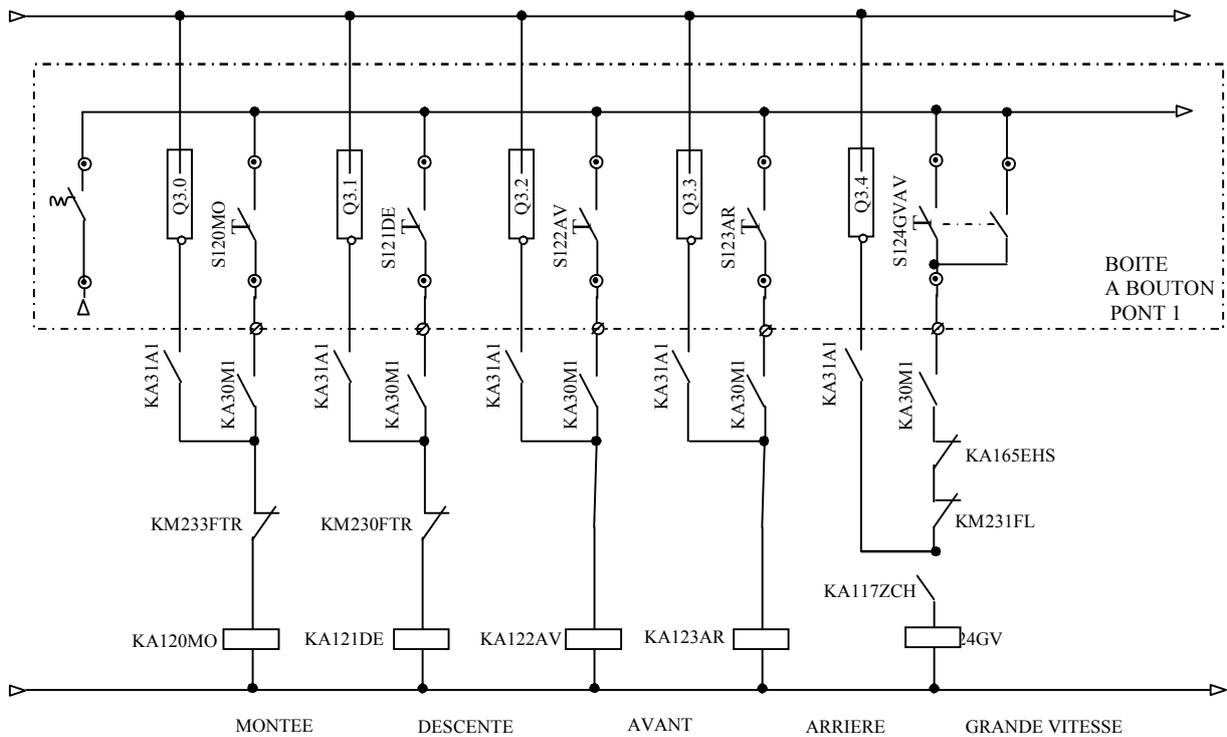




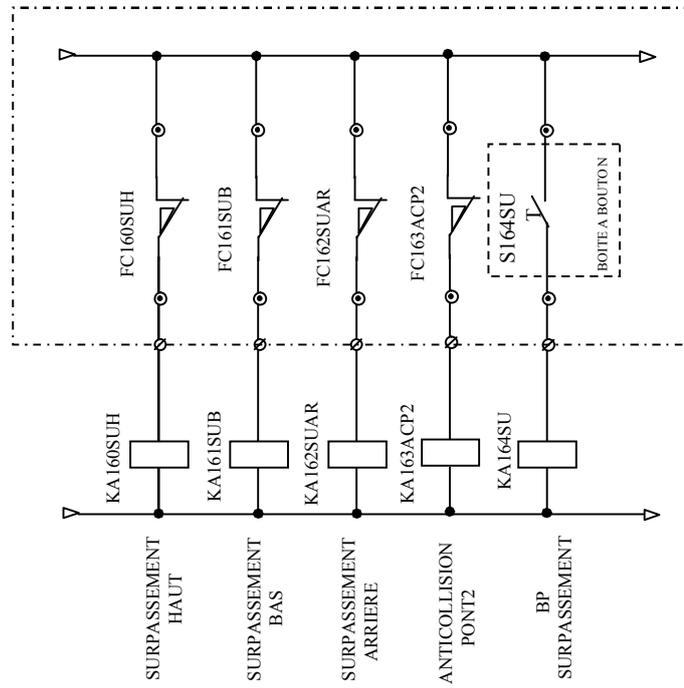
Détecteur de pont 1



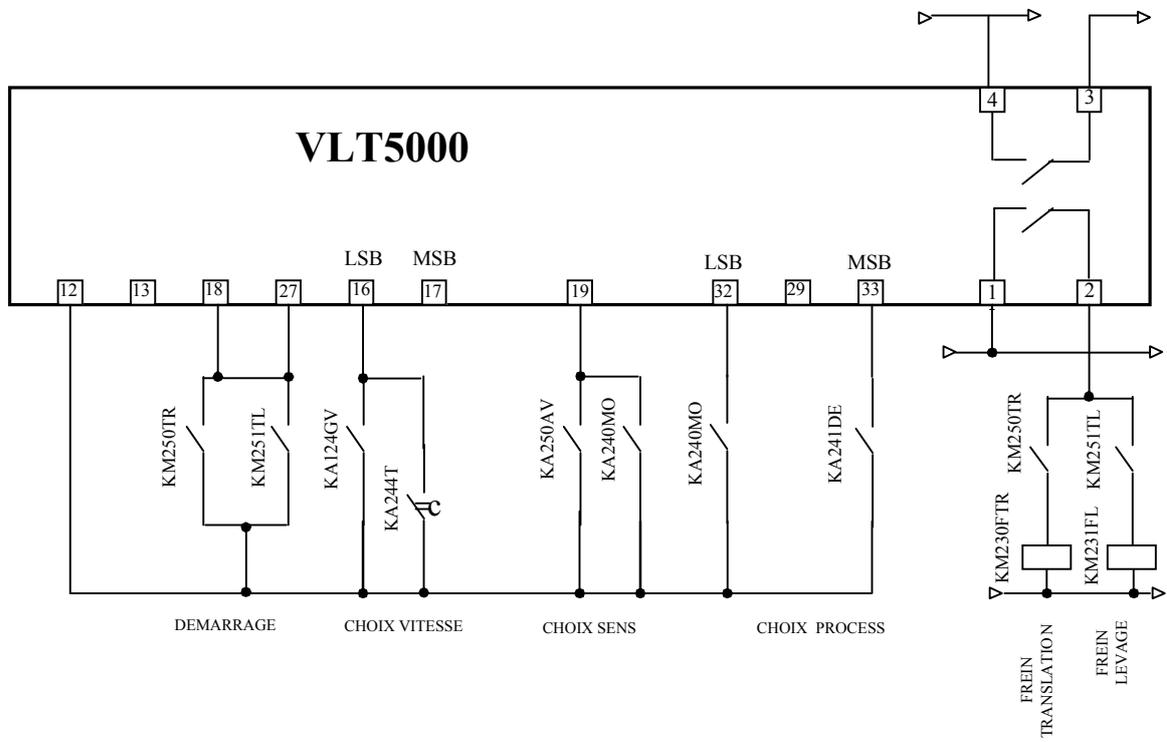
Commande pont 1



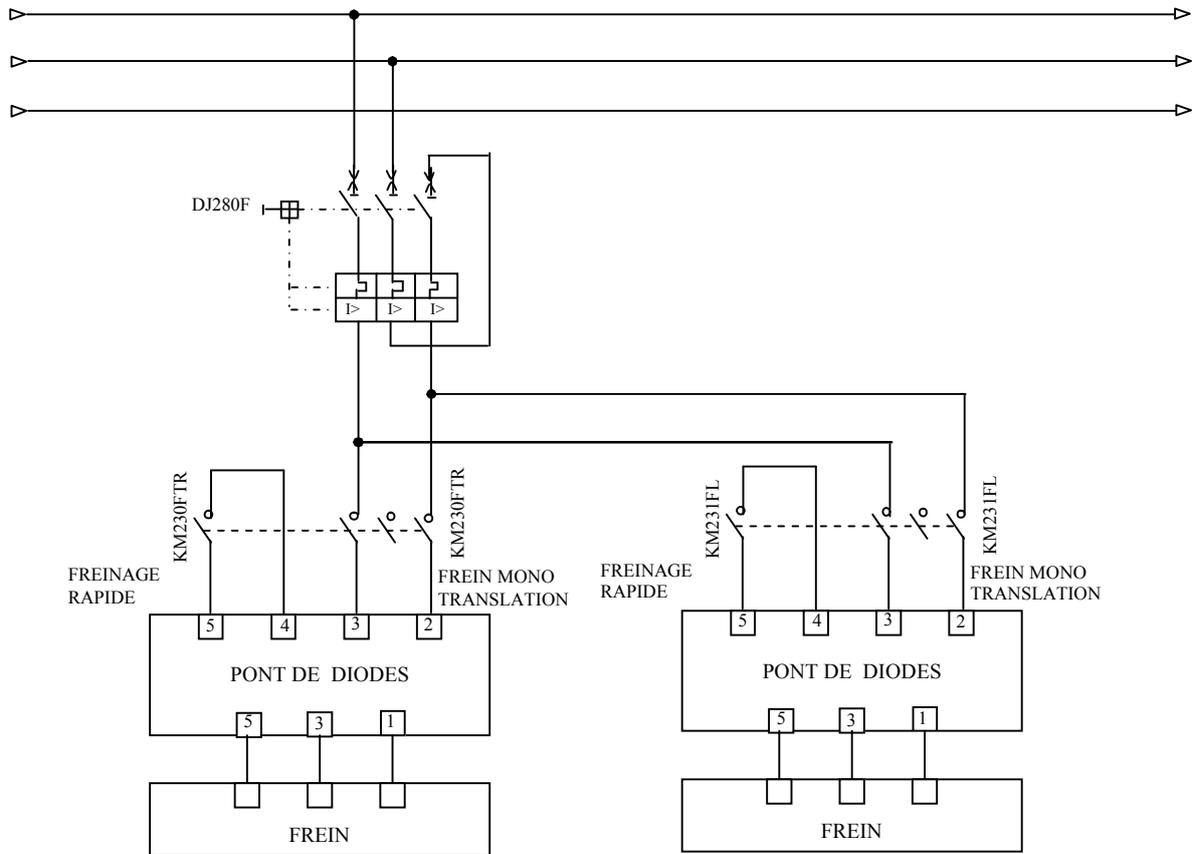
Surpassement pont 1



Variation de vitesse pont 1



Puissance pont 1 translation levage



- [1] **C.Bourbonne. J.Cojean**, « les systèmes automatisés, Tome1 » Edition foucher 1992.
- [2] **D.Blin ,J.Danic, R.Le Garrec, F.Trolez, J.C.séité** « automatique et informatique industrielle » Edition Casteilla 1999.
- [3] **J-M.Bleux, J-L.Fanchon**, « automatismes industriels » Edition Nathan1996.
- [4] **Télemécanique** « détecteurs électroniques » catalogue juillet 2000.
- [5] **Schneider Electric** « gamme magelis graphique XBT-F/TXBT-F » guide d'exploitation avril 2001.
- [6] **P.Trau** « le grafcet et sa mise en œuvre » ULP 1997.
- [7] **A.Juton** « automatismes industriels I » janvier 2007.
- [8] **Schneider Electric** «automates premium »septembre 2002.
- [9] **Schneider Electric** « démarreurs et variateurs de vitesse électroniques » cahier technique N° 208 année 2003.
- [10] **Schneider Electric** « variateur de vitesse pour moteurs asynchrones » Année 2003.
- [11] **Schneider Electric** « variateur de vitesse en coffret pour moteurS asynchrones » Année 2004.
- [12] **Melle OUMOUSA Kahina et OUHADA Fatiha** « développement d'une solution programmable et supervision de l'installation de traitement de surface » automatique 2007.
- [13] **Melle BEROUCHE Fazia, BIR Nassiama et BOUARABA Lila** « commande d'un variateur de vitesse MICROMASTER vector par PROFIBUS en communication avec un automate programmable S7-300 » automatique 2006.
- [14] **documentation de l'entreprise ENIEM.**
- [15] **logiciel PL7**