

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Mouloud MAMMARI de Tizi-Ouzou



Faculté de Génie Electrique et d'Informatique
Département d'Automatique

PROJET DE FIN D'ETUDES

En vue de l'obtention du diplôme

D'INGENIEUR D'ETAT EN AUTOMATIQUE

Thème

Etude en vue de la modernisation de la partie
commande d'une machine outil « SCEMM » de type
aléseuse.

Proposé par :

M^r M.GAOUAOUI

Présenté par :

M^{elle} MEKHTOUB Karima

Dirigé par:

M^r S. DJENNOUN

Soutenu le : 15 / 10 / 2009 Devant le jury d'examen composé de :

M^r Dirami

M^r Lekhlef

M^{eme} kharas

Promotion 2009

Remerciement

Avant tout je remercie dieu le tout puissant qui m'a donné la patience, la volonté, le courage d'affronter tous les obstacles afin d'aller jusqu'au bout de mon travail.

Je tiens à exprimer ma vive reconnaissance et profonde gratitude à mon promoteur Mr Djennoun pour son aide et son suivi théorique. A mon co-promoteur Mr Gaouaoui pour son suivi pratique et ses précieux conseils.

Il m'est particulièrement agréable de témoigner ma reconnaissance au personnel du service rénovation : Mr Hadj Ali, Mr Hargas, Mr Saïeh, Melle Benfreha, Da rezki, le petit Hargas et Oussama, Mr Doukhal et tous ceux qui m'ont aidé et qui ont été un appui et une source d'informations pendant toute la période de mon stage.

Je remercie aussi Mr Ouazar, Mr Mahfoufi, Mr Chahal, Mr Djebili, Mr Raouda, Mr Yahiaoui, Mr Sefroun, Mr Saidi.

Je remercie aussi l'équipe du transport de Boughni et de Tizi Ouzou (220A) pour leur aide.

Mes derniers remerciements vont aux membres de jury qui feront l'honneur de juger notre travail

Enfin, que toutes celles et tous ceux qui, de près ou de loin m'ont généreusement offert leurs aide à l'élaboration de ce travail, trouvent ici l'expression de ma profonde sympathie.

DÉDICACES

Je dédie ce modeste travail en témoignage de gratitude à :

A mes très chers parents, pour leur amour, leurs sacrifices et leur soutien moral, durant toute ma vie.

A mes chers frères et sœurs : Nana smina, Nora (khokha), Kamel, Djamel, Bélynda, Nadia, Karim, Iyes, djemaa, pour leurs encouragements, leur soutien moral et leurs aide durant mon parcours.

A mes beaux frères et belle sœurs : Da Ali, Chikh Ali, Taous, Rachida, Malik, Samir, Razika, Radia, hakima.

A tout mes neveux et nièces : Mahdi, Samah, Moumouh, Amina, Rayan, Léna, Djuba, Nabila, Yamina, Idir, Anis, Lisa, Rayan et Maria, et surtout la petite SARAH.

A mes chers amis de toujours : Said, Zakia, Djamila, Radia, Rima, Hocine, hakima, Yassine, Djamila R, Yahia.

A mes amis de fac : Karima, Malha, Kahina, Faiza.M, Zahoua, Malik, Khlifa, Farida, Chadia, Faiza, Saliha. Sans oublier mes camarades de l'automatique.

Et surtout à l'amour de ma vie Yehia, ainsi que toute sa famille.



Sommaire



Présentation de l'organisme d'accueil.....	01
Introduction Générale.....	02

Chapitre 1 : Présentation et étude de la machine

1-Introduction	04
2- Description et emploi des constituants de la machine.....	05
2-1- Partie mécanique.....	05
2-1-1-Principe de fonctionnement.....	06
2-2-Partie hydraulique.....	07
2-2-1-Pompe hydraulique.....	07
2-2-2-Pompe de graissage.....	07
2-2-3-Principe de fonctionnement à partir des schémas hydrauliques.....	09
a) Bridage de la pièce.....	09
b) Avance rapide.....	10
c) Avance lente.....	10
d) Blocage table et déblocage du transfert pièce.....	10
2-3-Partie électrique.....	10
2-3-1- Principe de fonctionnement à partir des schémas électriques.....	12
2-3-1-a) Préparation de la machine.....	12
2-3-1-b) Le départ cycle.....	13
🚦 En mode automatique.....	13
★ Postes A.....	13
★ Poste B.....	15
🚦 Le mode manuel.....	16
3-Conclusion.....	18

Chapitre 2 : Modélisation Graphique du Fonctionnement

1- Introduction.....	19
2- Proposition d'améliorations.....	19
2-1-Amélioration sur la sécurité.....	19
a) Le détecteur de niveau.....	19
b) Détecteur photoélectriques	19
2-2- Amélioration sur l'économie.....	20
3- Modélisation par GRAFCET	20
3-1- Définition du GRAFCET.....	20
3-2- Elément de base du GRAFCET.....	21
3-3- Les règles d'évolutions.....	21
3-4- Sélection de séquence et séquence simultanée.....	21
a- Sélection de séquences.....	22
b- Séquences simultanées.....	22
c- Saut d'étapes.....	22
d- Reprise de séquence.....	22
4- Mise en œuvre du GRAFCET.....	23
4-1Grafcet de sécurité niveau 1.....	23
4-2-Grafcet de sécurité niveau 2.....	23

4-3 Graficet de fonctionnement niveau 1.....	24
4-4 Graficet de fonctionnement niveau 2.....	26
5-Les équations d'activation et de désactivation des étapes.....	28
6-Application.....	28
a) Equation des transitions.....	28
c) Equation des étapes.....	29
7-Affectation des entrées/sorties.....	30
8-Conclusion	32

Chapitre 3 : Généralité sur les automates programmables

1- Introduction.....	33
2- Présentation d'un automate.....	33
2-1- Introduction.....	33
2-2- Définition d'un système automatisé.....	34
2-3- Structure générale d'un système automatisé.....	34
2-4- Système de commande.....	35
2-4-1- Logique câblée.....	35
2-4-2- Logique programmée.....	35
3- Définition d'un automate programmable industriel.....	35
4- Structure interne d'un automate programmable.....	35
4-1- L'unité centrale (UC).....	36
4-1-1- Le processeur.....	36
a. Registre interne.....	36
b. Compteur ordinal (pointeur).....	36
c. Registre d'instructions (code opératoire).....	36
d. Registre d'adresse (adresse opérande).....	36
e. Registre accumulateur.....	36
4-1-2- Mémoire.....	36
4-2- Bloc d'alimentation.....	37
4-3- Les modules enfichables.....	37
4-3-1- Les cartes mémoires (RAM, EPROM).....	37
4-3-2- Les coupleurs.....	37
4-3-3- Les cartes d'entrées / sorties.....	37
4-3-4- La console de programmation.....	37
5- Modules d'entrées/sorties.....	37
5-1- Les modules d'entrées tout ou rien (TOR).....	37
5-2- Les modules de sorties tout ou rien (TOR).....	37
6- Fonctionnement d'un automate programmable.....	37
6-1 détection.....	38
6-2 La commande d'action.....	38
6-3- Dialogue d'exploitation.....	38
6-4- Dialogue de supervision.....	39
6-5- Dialogue de programmation.....	39
7- Choix d'un automate programmable industriel.....	39

8- Conclusion.....	39
--------------------	----

Chapitre 4 : Etude et développement du logiciel de programmation en STEP7

1- Introduction.....	40
2- Logiciel STEP7.....	40
3-Gestionnaire de projets SIMATIC MANAGER.....	40
4-Editeur de mnémoniques.....	40
4-1- Définition d'un mnémorique.....	40
4-1- Adressage absolu.....	40
4-2- Adressage symbolique.....	40
5- Langage de programmation.....	41
5-1- En «Schéma à contact » symbolisé par : (CONT).....	41
5-2- En « liste d'instruction » symbolisé par : (LIST).....	41
5-3- En « logigramme » symbolisé par : (LOG).....	41
6- Programmation sous STEP7.....	42
6-1- Création d'un projet.....	42
6-2- Configuration du matériel (hardware configuration).....	43
7- Bloc dans le programme utilisateur.....	43
7-1-Définition.....	43
7-2-Bloc de code.....	44
a. Bloc d'organisation (OB).....	44
b. Bloc fonctionnel (FB).....	44
c. Bloc des données d'instance (SDB).....	44
d. Fonction (FC).....	44
e. Les SFB et SFC.....	44
7-3-Bloc des données.....	44
a. Bloc des données (BD).....	44
b. Bloc des données système (SDB).....	44
8-Traitement du programme.....	44
8-1-Traitement du programme cyclique.....	44
8-2- Traitement de programme déclenchant par événement.....	45
8-3- Programmation linéaire et programmation structurée.....	46
8-4- Appels de blocs.....	46
a- Appel du FB.....	47
b. Appel du FC.....	47
c. Opération nulle « NOP 0 » et « NOP 1 »	48
8-5- Temporisation et comptage.....	48
a- Définition.....	48
b- Opérations disponibles.....	48
8-6- Opérations de comptages.....	49
a- Les Opérations disponibles.....	49
9- Conclusion.....	50

Chapitre 5 : Chargement et test du programme

1- Introduction.....	51
2- Présentation de l'automate S7-300 de SIEMENS.....	51
3- Constitution de l'automate S7-300.....	51
3-1- Module d'alimentation (PS 307).....	51
3-2- Unité centrale (CPU 312 IFM).....	51
3-2-1- Un processeur.....	51
3-2-2- Des mémoires.....	52
3-2-3- Les modules d'entrées / sorties TOR (SM-300).....	52
3-2-4- Les modules d'entrées/sorties analogiques.....	52
3-2-5- Modules de simulation (SM 374).....	52
3-2-6- Le châssis (rack).....	53
3-2-7- Pupitre opérateur.....	53
3-2-8- Les modules enfichables.....	53
1. des coupleurs (IM)/S7-300.....	53
2. console de programmation (PG 700).....	53
4- Chargement et test du programme.....	53
4-1- Etablir la liaison en ligne.....	53
4-2- Chargement du programme.....	54
5- Le logiciel S7-PLCSIM- Simulateur des modules.....	57
5-1- Introduction générale sur le S7-PLCSIM.....	57
5-2- Etats de fonctionnement de la CPU.....	57
5-2-1- Etat de marche (RUN-P).....	57
5-2-2- Etat de marche (RUN).....	57
5-2-3- Etat d'arrêt (STOP).....	57
5-2-4- Effacement général de la mémoire de la CPU.....	58
5-3- Indicateurs de la CPU.....	58
5-4-1- Cycle unique.....	58
5-4- Sélection des options de mode d'exécution.....	58
5-4-2- Cycle continu.....	58
6- Test du programme.....	59
6-1- Test avec la fonction de visualisation.....	59
6-2- Test avec la table des variables.....	59
6-3- Test avec un programme de simulation (logiciel optionnel S7-PLCS1M).....	59
7- Conclusion.....	59

Conclusion générale

Conclusion générale	60
---------------------------	----

Annexes

- Annexe A : documentation techniques de la machine
- Annexe B : programme de la machine
- Annexe C : les capteurs



Présentatio de l'Organisme d'Accueil



Présentation de la SNVI & DVI :

La S.N.V.I (société nationale des véhicules industriels) est une usine de poids lourds construite en 1957 sous le nom « BERLIET ». En 1967 fut nationalisée et dénommé « SONACOM » (société national de construction mécanique). Elle a une superficie totale de 440 000m² et comporte 8000 travailleurs.

Parmi les plus importantes unités de la SNVI, on trouve l'unité DVI (division des véhicules industriels) ; entreprise située dans la zone industriel de ROUIBA à une trentaine de km à l'est d'ALGER.

Elle conçoit et réalise des engins roulant qui jouent un grand rôle dans l'économie national, sa superficie est égale a 746 980 m².

La DVI est composée des bâtiments suivants :

- Bâtiment mécanique a une superficie de 340 000 m².
- Bâtiment forge de 52 000 m².
- Bâtiment emboutissage de 43 000 m².
- Bâtiment de montage camions de 36 000 m².
- Bâtiment autobus- autocars de 34 000 m².
- Bâtiment carrosserie de 68 000 m².
- Ainsi que le bâtiment fonderie, centre de formation.

Bâtiment mécanique usine :

- 8 types de ponts
- 5 types essieux
- 6 types de boites de vitesses
- Carters de directions
- Pièces diverses (ferrures pour cadre châssis, pédaliers pour cabines, leviers de vitesses, tiges d'accélération, leviers de pédales d'embrayage, etc...).

Produits :

- Camions : K66 ; K120 ; M120 ; M230 ; B260 ; TB260 ; C260 ; TB350 ; B350.
- Cars: Safir (49L6); Minicars: 38L6.
- Bus: 100V8
- Minibus: 70L6

Parmi ces unités mon stage s'est déroulé à la DVI, dans le département mécanique.



Introduction Générale

Introduction Générale

Actuellement la demande en énergie est fortement accrue. En dépit de l'évolution rapide des nouvelles technologies et des solutions réelles proposées pour contourner les difficultés rencontrées dans le monde industriel, les entreprises se trouvent confrontées à des contraintes d'optimisation des consommations en énergie et de satisfaction des consommateurs.

La nécessité de l'amélioration de la production et de la productivité et l'augmentation des normes de sécurité du personnel et de matériels s'imposent d'elle même. Ceci implique l'obligation de disposer d'une chaîne de production souple et performante. Le recours à une automatisation programmée à l'aide d'un automate programmable s'avère comme une solution efficace. Pour ces différentes raisons, nous assistons actuellement dans notre pays, pour les entreprises industrielles nationales et privées, à une automatisation de grande envergure.

Ainsi, la Société Nationale des Véhicules Industriels (SNVI), s'est attelée depuis quelques années au remplacement et à la modernisation des équipements de contrôle et de conduite des machines dans différents postes de travail par des automates programmables. Dans la chaîne de production des véhicules, de nombreuses machines sont anciennes datant des années soixante dix. C'est le cas de l'aléuseuse SCEMM utilisée au niveau de la chaîne de fabrication des carters de boîte à vitesse. Cette aléuseuse est commandée par un système de commande de technologie câblée qui vétuste posant ainsi de sérieux problèmes d'approvisionnement en pièces de rechange. De plus l'entretien et la maintenance de ces équipements qui deviennent plus fréquents (y compris la boîte de vitesse).

Vue son importance dans la chaîne, il nous a été demandé de concevoir un système de commande par automate programmable pour envisager dans un futur proche au remplacement du système de commande à logique câblée.

L'introduction des automates dans les processus permet un gain du temps, une souplesse accrue dans la manipulation, une haute fiabilité, la localisation et l'élimination rapide des pannes.

Dans le cadre de notre projet, nous avons un stage pratique au sein de la société Nationale des Véhicules Industriels (SNVI), au niveau du département mécanique. Durant la première partie de ce stage, nous nous sommes attelés à étudier, sur le site, le fonctionnement de la machine afin de pouvoir le modéliser. La seconde partie porte sur la conception de la solution programmée qui a nécessité l'association des opérateurs intervenants sur cette machines.

Notre mémoire est organisé comme suit :

- En **avant propos**, nous exposons une brève description de l'organisme d'accueil qui est l'entreprise SNVI.
- Le **premier chapitre** est consacré à la présentation et à l'étude de la machine.
- Le **deuxième chapitre** porte sur la modélisation graphique du fonctionnement de la machine au moyen de l'outil GRAFCET.
- Dans le **troisième chapitre**, nous donnons une description des automates programmables ainsi que quelques généralités sur leur fonctionnement.
- Au **quatrième chapitre** nous proposons la solution programmée développée sous le logiciel STEP7.
- Dans le **cinquième chapitre**, nous procédons au chargement du programme réalisé sur micro ordinateur. Des tests de simulation sont effectués afin de valider la solution proposée.



Chapitre 1: Présentation et Etude de la Machine



1-Introduction :

La « **SCEMM** » représentée en « figure (1.2) » est une machine d'usinage mécanique. Elle a pour fonction l'alésage des carters de boîte à vitesse, utilisés dans les différents véhicules fabriqués par l'entreprise.

L'alésage est une opération d'usinage avec enlèvement de matière à l'aide d'outils spécifiques et durs. Il comporte deux opérations essentielles : l'ébauche et la finition.

- ❖ L'ébauche consiste à enlever une quantité importante de matière des lignes principales (post A) et intermédiaires (post B) des carters afin d'éliminer les pleines matières ou trous venus de fonderie, et obtenir un diamètre précis.
- ❖ La finition consiste à enlever une petite quantité de matière laissée par l'ébauche et l'obtention d'un bon état de surface interne du carter.

Le carter (figure (1.1)), qu'il soit en fonte (carter BR, ou BD), en aluminium (carter BB), ou mixte (fonte +aluminium : carter BX), permet de positionner les différents éléments de la boîte à vitesse afin de garantir un fonctionnement parfait. Comme il est une partie intégrante du châssis, il doit être rigide en torsion et en flexion car il doit encaisser les efforts aérodynamiques et de la suspension. L'alésage des lignes principales et intermédiaires doit être précis, afin de réduire les jeux en translation.



Figure (1.1): « Carter de boîte à vitesse »

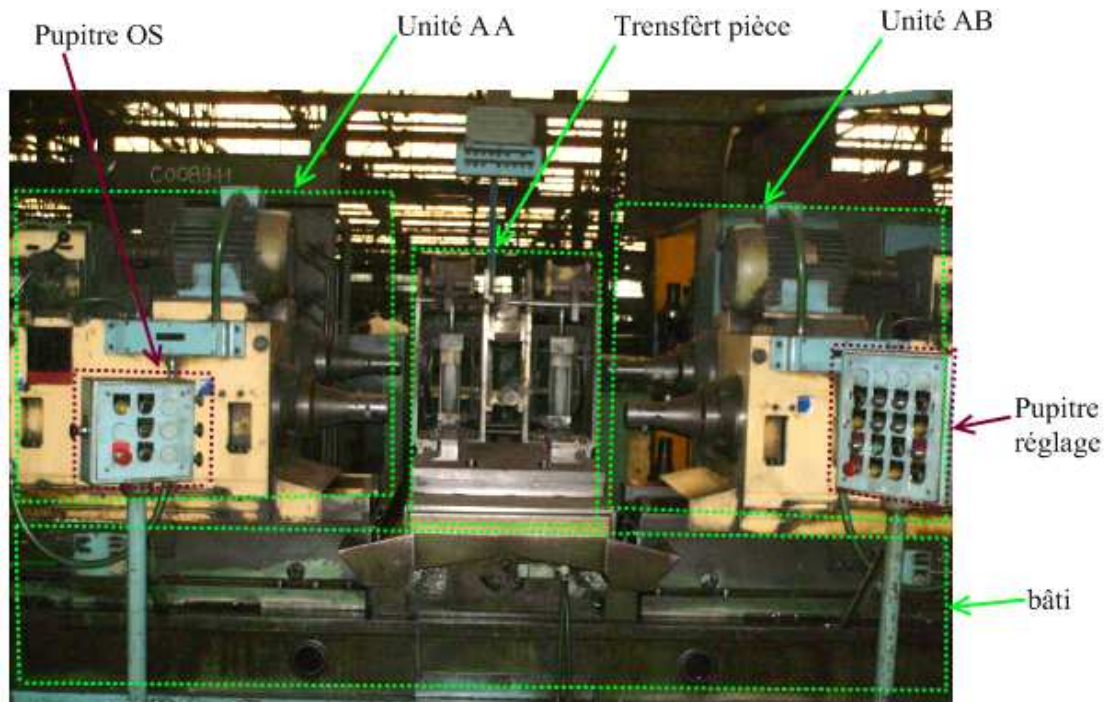


Figure (1.2) : « Schéma d'ensemble de l'aléuseuse SCEMM »

2- Description et emploi des constituants de la machine:

2-1- Partie mécanique :

Parmi les organes principaux qui la constituent on cite :

- Un bâti en forme de « T » qui supporte tous les organes de la machine.
- Un transfert pièce (table) entraîné par un vérin double effet sur lequel sont portés deux autres vérins doubles effets pour le bridage de la pièce.
- Des poupées portes broches : droite (unité AA), et gauche (unité AB) dont chacune est entraîné par un vérin hydraulique double effet (fixées sur les tiges) et guidé par des glissières « voire figure (1.3) ».
- Quatre broches d'alésage, dont celles pour ligne intermédiaires sont réglables.
- Deux volants qui déplacent les broches réglables.
- Une boîte à vitesse a base de pignons à 4 rapports.

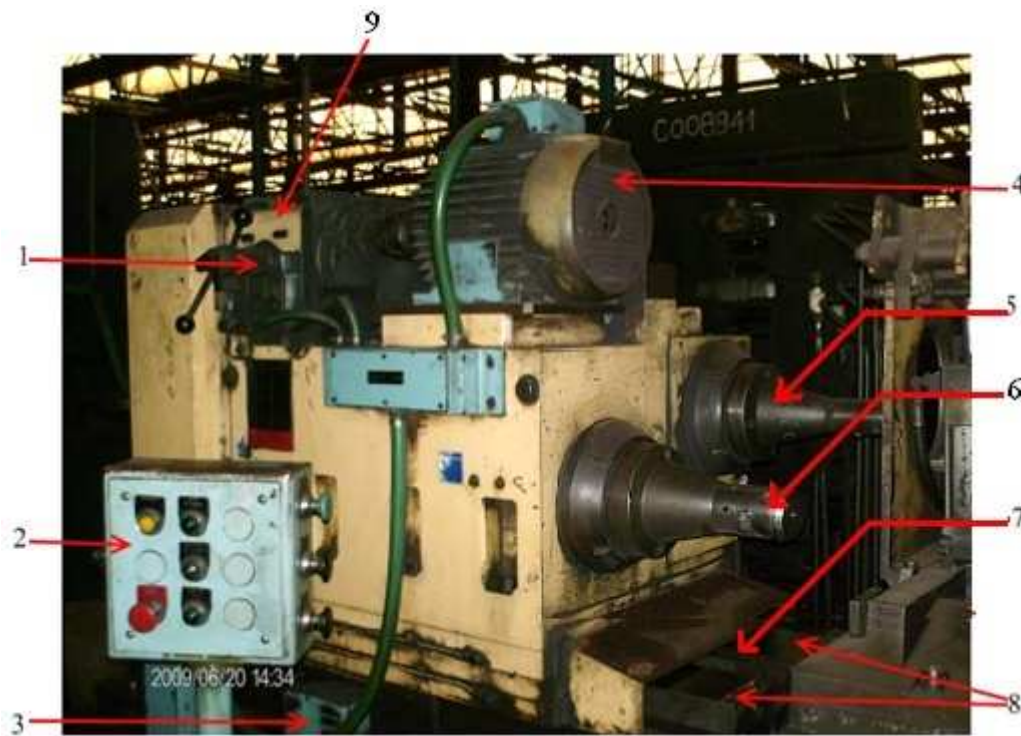


Figure (1.3) : « Schémas d'ensemble de la poupée porte broche {unité AA} »

1 : sélecteur de vitesse.

2 : pupitre OS (opérateur)

3 : les fins de course

4 : moteur d'entraînement des broches

5 : broche porte outils pour post A

6 : broche porte outils pour post B

7 : vérin hydraulique à double effet

8 : glissières

9 : boîte de vitesse

2-1-1-Principe de fonctionnement :

Les différents mouvements de la machine sont commandés de la façon suivante. Une translation horizontale des unités AA, AB et du transfert qui sont entraînées par des vérins hydrauliques à double effets. On distingue deux types d'avance réglée par des régulateurs de débit : rapide et lente (de travail). La longueur de la course est réglable par des cames, pour l'avance de travail, le réglage est lié à la gamme d'usinage.

La rotation de broche est obtenue par un moteur asynchrone; le moteur entraîne l'arbre d'entrée à travers l'accouplement, le mouvement est transmis à la boîte de vitesse qui a quatre rapports. Le changement de vitesse est assuré par deux fourchettes, chacune à deux positions entraînant les pignons sur l'arbre intermédiaire vers l'arbre de sortie.

L'arbre de sortie transmet le mouvement à l'arbre de la broche à travers une courroie crantée.

2-2-Partie hydraulique :

Le circuit hydraulique (annexe A) est constitué des différents éléments qui sont :

- Un groupe hydraulique situé du côté gauche derrière la machine, il est relié à celle-ci par des canalisations flexibles. Le groupe hydraulique comprend un bac, un moteur hydraulique, et une pompe hydraulique.
- Des vérins hydrauliques à double effet, deux sont fixés sur le bâti, pour la translation des unités AA, AB, deux autres sont fixés sur le transfert pour le bridage de la pièce.
- Des vérins hydrauliques à simple effet pour le blocage du transfert.
- Des constituants de protection, de régulation, des distributeurs portés sur une armoire fixée sur le bac hydraulique, ces constituants sont résumés dans le tableau (1.1).

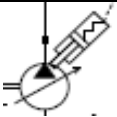
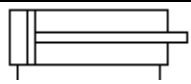
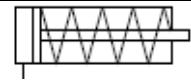

2-2-1-Pompe hydraulique :

C'est une pompe à engrenage à deux débits variables [PV1 (20 bars), PV2 (50 bars)] entraînée par un moteur asynchrone qui tourne à 1500tr/min.

Sa puissance maximale est de 100 bars. L'ensemble est fixé sur le bac hydraulique dont la capacité est de 300 L.

2-2-2-Pompe de graissage :

Le graissage est assuré par une pompe manuelle, d'une capacité de 2 L. l'action sur le levier crée un vide partiel à l'aspiration, ce qui permet, à la pression atmosphérique régnant dans le réservoir, d'obliger le liquide à remonter la canalisation jusqu'à la pompe. En second lieu, elle assure le transport du liquide jusqu'à l'orifice de refoulement.

Symbole	Désignation	Rôle
	Pompe hydraulique à 2 débits variables	Transformation de l'énergie mécanique en énergie hydraulique
	Vérins à double effet	Transformation de l'énergie hydraulique en énergie mécanique
	Vérins à simple effet	
	Clapet anti retour	Empêchement du passage d'écoulement

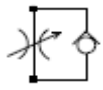
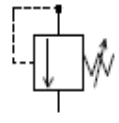
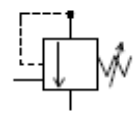
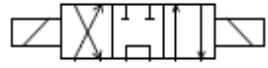



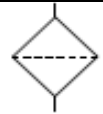

		dans le sens inverse.
	Régulateur de débit à tarage réglable.	Régulation du débit d'écoulement dans une partie du circuit
	Réducteur de pression réglable	Réduction de pression du service dans une partie du circuit.
	Réducteur de pression réglable piloté.	Réduction de pression du service dans une partie du circuit à condition d'avoir une pression qui autorise son fonctionnement.
	Distributeur 4/3 fermé à commande électrique.	Commande des vérins
	Distributeur 4/2 électromécanique.	
	Pressostat	Capteur de pression.
	Manomètre	Indication visuel de pression
	Filtre	Empêchement de passage des impuretés solides véhiculées par l'huile.
	Robinet d'isolement.	Evacuation de l'huile vers le bac en cas de dépassement de la pression la valeur limite.

Tableau (1.1) : désignations des éléments hydrauliques

Le tableau qui suit représente les caractéristiques générales des actionneurs hydrauliques (vérins hydrauliques).

Emplacement	Unité d'alésage droite	Unité d'alésage gauche	Transfert pièce	Serrage table	Serrage pièce
Nom du vérin	V1 _{AA}	V2 _{AB}	V1 _{FT}	- V2 _{FT} - V3 _{FT} - V4 _{FT} - V5 _{FT}	- V1 _{FA} - V2 _{FA}
Type	Double effet	Double effet	Double effet	Simple effet avec ressort	Double effet
nombre	1	1	1	4	2
Alésage piston	Ø 140	Ø 140	Ø 125		Ø 63
Alésage tige	Ø 63	Ø 63		Ø 63	Ø 22
Section	154 cm ² - 123 cm ²	154 cm ² - 123 cm ²	122.5cm ² - 91.5 cm ²	31.5cm ²	31.2 cm ² -27.4 cm ²
course	400 mm	400 mm	400 mm	165 mm	165 mm
Emplacement	Unité d'alésage droite	Unité d'alésage gauche	Transfert pièce	Serrage table	Serrage pièce
Nom du vérin	V1 _{AA}	V2 _{AB}	V1 _{FT}	- V2 _{FT} - V3 _{FT} - V4 _{FT} - V5 _{FT}	- V1 _{FA} - V2 _{FA}
Type	Double effet	Double effet	Double effet	Simple effet avec ressort	Double effet
nombre	1	1	1	4	2
Alésage piston	Ø 140	Ø 140	Ø 125		Ø 63
Alésage tige	Ø 63	Ø 63		Ø 63	Ø 22
Section	154 cm ² - 123 cm ²	154 cm ² - 123 cm ²	122.5cm ² - 91.5 cm ²	31.5cm ²	31.2 cm ² -27.4 cm ²
course	400 mm	400 mm	400 mm	165 mm	165 mm

Tableau (1.2) : « caractéristiques générale des vérins »

2-2-3-Principe de fonctionnement à partir des schémas hydrauliques(Voir annexe A):**a) Bridage de la pièce :**

Afin d'avoir une bonne précision, un bon état de surface, on bloque la pièce avant l'usinage, et cela à l'aide de deux vérins hydraulique V1_{FA}, V2_{FA}.

Toutes les avances et reculs des unités AA, AB et du transfert pièce se font par l'envoi d'énergie hydraulique à travers des distributeurs électriques 4 /3.

b) Avance rapide :

Elle se fait par l'envoi direct de l'huile dans les chambres arrière des vérins V1 et V2 à travers 2 distributeurs D1, D2 pour l'unité AA, et D1, D2 pour l'unité AB, ces vérins sont poussés en sortie rapide entraînant les poupées portes broches (droite, et gauche) fixées sur leurs tiges et guidées par des glissières.

Quand au transfert pièce l'envoi de l'huile dans la chambre négative du vérin V1_{FT} à travers les distributeurs D1 et D2 (FT) est conditionné par un régulateur de pression taré à 45 bars et piloté par le déblocage de la table, d'où V1_{FT} est poussé en rentré rapide.

c) Avance lente :

L'énergie hydraulique n'étant envoyée qu'à travers D2_{AA}, D2_{AB}, la pression diminue, ce qui entraîne une sortie lente des vérins, qui se traduit par l'avance lente des poupées portes broches droite et gauche.

La même chose pour le transfert pièce, le passage de l'huile par un seul distributeur D2_{FT} diminue la pression initiale, ce qui ralentit l'avance du vérin V1_{FT}, y compris l'avance de la table.

d) Blocage table et déblocage du transfert pièce :

Il s'effectue à l'aide de 4 vérins simple effet. L'huile est envoyée dans les chambres positives à travers le distributeur D3_{FT}, les tiges sortent, la table est dégagée.

La fermeture du distributeur, laisse échapper l'huile et grâce à la force de rappel des ressorts, les tiges reculent, ce qui bloque la table.

2-3-Partie électrique :

La machine contient différents organes électriques suivant dont les caractéristiques générales sont résumées dans le tableau (2):

➤ Une armoire électrique qui est constituée de.

Transformateurs, sectionneur, contacteurs, relais thermiques, temporisateurs.

➤ Des moteurs électriques

➤ Deux pupitres de commande (OS : opérateur, réglage), et un bloc de voyant.

➤ Des capteurs.

Ces différents éléments sont alimentés avec :

- 380V(~), 50Hz pour le circuit de puissance.
- 110V(~), 50Hz pour le circuit de commande.
- 24V(~) pour le circuit de signalisation.

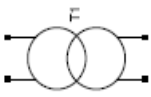
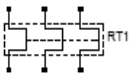

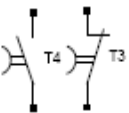
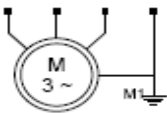
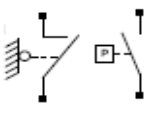
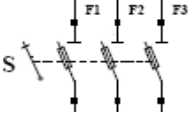
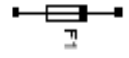
Symbole	Désignation	Rôle
	Transformateurs monophasés	Abaisse la tension pour alimenter le circuit de commande.
	Relais thermiques	Protection des moteurs électriques et hydrauliques contre les surcharges.
	Contacteurs de puissance et de commande	établissement ou interruption (pouvoir de coupure) du passage de l'énergie électrique.
	Contactes temporisés	établissement ou interruption du passage de l'énergie électrique pendant un certain temps.
	Moteurs	Entrainement de la pompe hydraulique, et des broches d'alésage.
	Capteurs (fin de course, pressostat)	Acquisition des informations qui seront transmises a la partie commande.
	Sectionneur porte fusible	Isolation du circuit électrique, et sa protection grâce aux fusibles.
	fusibles	Protection des

Tableau (1.3) : « Désignations des éléments électriques ».

Le tableau qui suit représente les caractéristiques générales des actionneurs électriques (moteurs électriques).

Moteur	Tension	Puissance	Vitesse	Courant max
Hydraulique (HA)	380V	5.5Kw	1500tr/mn	12A
Broche unité AA (ME1)	380V	7.5Kw	1450tr/mn	20A
Broche unité AB (ME1)	380V	7.5Kw	1450tr/mn	20A

Tableau (1.4) : « caractéristiques générales des moteurs »

2-3-1- Principe de fonctionnement à partir des schémas électriques :(voir annexe A)

2-3-1-a) Préparation de la machine :

➤ Mise sous tension :

On ferme le sectionneur S ; alimentation du transformateur T1, les lampes suivantes s'allument:

- L₀ (armoire sous tension).
- L_{4FT} (transfert en arrière).
- L_{1AA} (unité AA reculée).
- L_{1AB} (unité AB reculée).

Si L_{1AA} et L_{1AB} ne sont pas allumées, on se met en mode manuel pour remettre les unités AA, AB en position reculée.

On appuie sur le bouton poussoir BM1, si BA1 et BA ne sont pas actionnés et RTH_{HY} n'est pas déclenché; le contacteur CH s'excite, et ses contacts se ferment, cela implique :

- la mise en marche de l'hydraulique (le moteur ME1 en marche) ;
- L'auto-maintien du contacteur CH ;
- La mise sous tension des deux parties : puissance et commande (L₁ allumée)
- L'Alimentation du transformateur T2 ; possibilité de vérifier l'état de marche des voyants par l'appui sur le bouton poussoir BM2

➤ Sélection du mode de travail :

La machine a deux modes de travail dont la sélection se fait avec le commutateur I3 ; en automatique ; ou en manuel (réglage).

➤ Sélection du type de carter à usiner :

La sélection du type de carter se fait avec le commutateur I₁ :

Position BXSL_{FT} : pour le carter BX (aluminium+fonte) ; la lampe L_{3FT} s'allume.

Position NORMAL_{FT} : pour le carter NORMAL (aluminium, ou fonte) ; la lampe L_{2FT} s'allume.

➤ **Serrage de la pièce :**

Après le chargement de la pièce, l'appui maintenu sur les boutons poussoirs BM1 et BM2 ; excite RBP_{FT} ; fermeture de ses contacts ; CB_{FA} s'excite, ses contacts se ferment, ce qui entraîne son auto maintien ; et l'excitation de l'électrovanne EVB_{FA} .

Pour l'ébauche c'est le pressostat $MC2_{FA}$ qui s'excite, pour la finition c'est $MC1_{FA}$ qui s'excite; leurs contacts se ferment et $TMC1_{FA}$ s'excite, la lampe L_{3FA} s'allume (serrage pièce effectué).

2-3-1-b) Le départ cycle :

✚ **En mode automatique :**

Si le commutateur I2 est en :

- Position 1 : utilisation des boutons poussoirs (il faut appuyer sur BM1 et BM3 pour donner l'ordre de départ).
- Position 2 : non utilisation des boutons poussoirs (l'ordre de départ est donné automatiquement après le serrage de la pièce).

Le fin de course $m1_{FT}$ est actionné, les bobines $1T_{FT}$ et $1T1_{FT}$ sont excitées ; fermeture de leurs contacts $1T$ et $1T1$

Les fins de course $m1_{AA}$, et $m1_{AB}$ actionnés, les bobines $RA_{AA, AB}$ sont excitées, d'où l'excitation de RH_{FT} , fermeture de ses contacts RH ; excitation de RZA_{FT} ; et fermetures des contacts RZA , cela permet son auto-maintien ; la lampe L_{1FT} s'allume.

➤ **Auto-unité :**

RZ est appelée ; ses contacts se ferment, enclenchement de son auto-maintien.

★ **Postes A :**

➤ **Départ cycle:**

Au repos les contacts des contacteurs $VRK_{AA, AB}$ et $VRU_{AA, AB}$ sont fermés.

A la mise sous tension, le contacteur $Rg2$ est excité et auto-maintenu par la fermeture de ses contacts. RD_{FT} s'excite et auto-maintenu par son contacte temporisé RD .

On appuie sur BM1 et BM3; excitation de $DRK_{AA, AB}$ et $DRU_{AA, AB}$ (appel du déverrouillage des relais mémoires).

$VRK_{AA, AB}$ et $VRU_{AA, AB}$ se dés excitent, la fermeture de leurs contacts excite RE (ordre de départ des unités).

Les contacts RE_{FT} fermés, excitation de $RB_{AA, AB}$ (ce qui donne l'autorisation des avances).

➤ **Avance des unités AA et AB et rotation des deux broches :**

L'excitation des électrovannes qui commandent l'avance rapide et lente des unités AA et AB, ainsi que les moteurs qui entraînent les quatre broches se fait en même temps, car ils ont la même condition d'activation (excitation du contacteur RB).

➤ **Rotation broches :**

Les contacts $RB_{AA, AB}$ et $RZ_{AA, AB}$ sont fermés ; les contacteurs $RB_{AA, AB}$ sont auto-maintenus, d'où l'excitation de $CC_{AA, AB}$; leurs contacts se ferment ; enclenchement de leurs auto-maintien et mise en marche des moteurs broche (ME1AA et ME1AB).

➤ **Avance rapide des unités AA et AB :**

$CA_{AA, AB}$ s'excitent, leurs contacts CA se ferment ; d'où l'excitation des électrovannes $EVA_{AA, AB}$ et les tiges des vérins $V1_{AA, AB}$ avancent rapidement.

➤ **Avance lente des unités AA et AB :**

Les fins de course $m2_{AA, AB}$ pour le poste A, (respectivement $m3_{AA, AB}$ pour le poste B) sont actionnés ; $EVA_{AA, AB}$ se désexcitent, ce qui provoque l'arrêt de l'avance rapide, $EVC_{AA, AB}$ restent excités ; les tiges des vérins $V1_{AA, AB}$ continuent d'avancer lentement.

➤ **Opération de lamage :**

Les fins de course $m4_{AA, AB}$ sont actionnés ; excitation de $4T_{AA, AB}$, après 5s leurs contacts temporisés s'ouvrent ; désexcitation de $RB_{AA, AB}$, cela entraîne la désexcitation des électrovannes $EVC_{AA, AB}$ et l'arrêt de l'avance lente, mais la broche tourne toujours.

➤ **Recul rapide des unités AA et AB :**

La désexcitation de $RB_{AA, AB}$ entraîne l'excitation de $CB_{AA, AB}$ et la fermeture de ses contacts $CB_{AA, AB}$; excitation des électrovannes $EVB_{AA, AB}$ (recul rapide des vérins $V1_{AA}$ et $V2_{AB}$).

➤ **Fin du recul des unités AA et AB :**

Les fins de course $m1_{AA, AB}$ actionnés, cela entraîne :

L'excitation de $RA_{AA, AB}$, d'où l'excitation des contacteurs $VRK_{AA, AB}$, et excitation de la bobine $Rg1_{FT}$ (contrôle du 1^{er} travail des unités).

La désexcitation de $CC_{AA, AB}$, d'où l'arrêt des moteurs broche $ME1_{AA, AB}$.

La désexcitation de $CB_{AA, AB}$, et l'ouverture de ses contacts désexcite les électrovannes $EVB_{AA, AB}$; d'où l'arrêt du recul rapide.

➤ **Débloccage table :**

Les fins de course $m1_{AA, AB}$ sont actionnés ; $RA_{AA, AB}$ et RH_{FT} s'excitent à nouveau, la fermeture de leurs contacts excite REC_{FT} .

Les contacts REC_{FT} fermés, RB_{FT} s'excite, la fermeture de son contacte temporisé excite CE_{FT} , d'où l'excitation de l'électrovanne EVE_{FT} (la table est dégagée).

➤ **Avance de l'unité transfert :**

Avance rapide :

Une pression pilote arrive sur $RP1$ qui s'ouvre pour laisser passer l'huile, et comme CA_{FT} et CB_{FT} sont déjà excités, leur contacts fermés ; les électrovannes EVA_{FT} et EVB_{FT} s'excitent, ce qui enclenche l'avance rapide du transfert.

Avance lente :

Le fin de course $m3_{FT}$ actionné ; désexcitation de CA_{FT} ; d'où la désexcitation de l'électrovanne EVA_{FT} (arrêt de l'avance rapide), EVC_{FT} reste excitée le transfert continue en avance lente.

➤ **Blocage table :**

Le fin de course $m4_{FT}$ est actionné ; les bobines $4T_{FT}$ et $4T1_{FT}$ s'excitent.

La fermeture de leurs contacts désexcite REC_{FT} et CB_{FT} ; d'où la désexcitation de l'électrovanne EVB_{FT} (arrêt de l'avance lente), arrêt de l'électrovanne EVE_{FT} , La lampe L_{5FT} s'allume (transfert en avant)

★ **Poste B :**

Les opérations d'alésages qui s'effectuent sur la ligne principale (poste A) sont les mêmes que celles qui s'effectuent sur la ligne intermédiaire (poste B), la différence se trouve dans l'actionnement de certaines opération par des fins de courses ou des contacteurs différents.

➤ **Ordre de départ des unités**

Il s'effectue de deux manières, selon le type de carter.

La fermeture du contacte $4T$ excite la bobine REA_{FT} (ordre départ 2^{ème} usinage).

La fermeture de son contacte excite la bobine $RC_{AA, AB}$, qui excite a son tour la bobine $RB_{AA, AB}$ par la fermeture de ses contacts, ce qui donne l'autorisation des avances.

La fermeture des contacts RB entraine l'excitation de l'électrovanne $EVD_{AA, AB}$ (sélection des avances lentes), et des contacteurs $CC_{AA, AB}$, $CA_{AA, AB}$, l'ouverture de ses contacts désexcite les électrovannes $EVB_{AA, AB}$; d'où l'arrêt du recul rapide.

➤ **Avance des unités AA et AB et rotation des deux broches :**

Ces opération sont les même que pour le poste A

➤ **Ordre de retour de l'unité transfert :**

Le contacte 4T fermé, $VRU_{AA, AB}$ excités ; leurs contactes fermés ; donc Rg2 s'excite une deuxième fois, Rg2 auto-maintenue, ce qui entraîne l'excitation de REB_{FT} , son contacte se ferme ; excitation de RC_{FT} .

➤ **Déblocage table :**

Les fins de course $m1_{AA, AB}$ sont actionnés ; RA et RH sont excités de nouveau ; le contacte RC est fermé ; CE_{FT} excité, ses contacte se ferme et l'électrovanne EVE s'excite (la table est dégagée).

➤ **Retour du transfert (table) :**

Retour rapide

Une pression pilote arrive sur RP1 qui s'ouvre pour laisser passer l'huile, le fin de course $m2_{FT}$ n'étant pas actionné, son contacte est fermé ; CC_{FT} et CD_{FT} sont excités, leur contactes sont fermés ; les électrovannes EVC_{FT} et EVD_{FT} s'excitent, le recul rapide est enclenché.

Retour lent

Le fin de course $m2_{FT}$ est actionné ; son contacte s'ouvre, désexcite CC_{FT} ; et désexcite l'électrovanne EVC_{FT} (arrêt du recul rapide), EVD_{FT} reste excité le transfert recul lentement.

➤ **Blocage du transfert (table) :**

Le fin de course $m1_{FT}$ est actionné ; les bobines 1T et 1T1 sont excitées ; fermeture de leurs contactes et désexcitation de REC_{FT} , CD_{FT} , et l'électrovanne EVD_{FT} , d'où l'arrêt de l'avance lente.

➤ **Desserrage de la pièce**

RF_{FT} excité par la fermeture du contacte 4T, et auto-maintenue ; l'ouverture du contacte $1T_{FT}$ désexcite CB_{FT} , et désexcite l'électrovanne EVB_{FA} ; excitation de CA_{FA} ; excite l'électrovanne EVA_{FA} par son contacte fermé.

➤ **Fin de cycle**

A la fin de la temporisation du contacte CA, il s'ouvre et désexcite la bobine CA_{FA} .

🔧 **Le mode manuel :**

On le sélectionne avec le commutateur I_3 , le contacteur RZM_{FT} est excité, ses contactes se ferment permettant ainsi l'accès au mode manuel.

➤ **Départ cycle**

On met le commutateur I2 sur la position 2, et on appui sur les boutons poussoirs $BM1_{AA, AB}$ et $BM3$; excitation de $RE_{AA, AB}$ (ordre de départ des unités) ;

➤ **Avance rapide des unités AA et AB:**

On appui sur le bouton poussoir $BM2$; excitation de $RB_{AA, AB}$, les contacts $RB_{AA, AB}$ et $RA_{AA, AB}$ fermés ; auto-maintien de $RB_{AA, AB}$, $CA_{AA, AB}$ excité, ses contacts CA fermés ; d'où l'excitation des électrovannes $EVA_{AA, AB}$ et les tiges des vérins $V1_{AA, AB}$ avancent en rapide.

➤ **Rotation broche (BM2)**

Le contacteur $CC_{AA, AB}$ excité ; ses contacts se ferment ; enclenchement de l'auto-maintien et mise en marche des moteurs broche ($ME1AA$ et $ME1AB$).

➤ **Recul rapide des unités AA et AB**

L'appui sur le bouton poussoir $BM2_{AA, AB}$ désexcite $RB_{AA, AB}$ et excite $CB_{AA, AB}$, fermeture des contacts $CB_{AA, AB}$; enclenchement de son auto-maintien et excitation des électrovannes $EVB_{AA, AB}$ (recul rapide des tiges des vérins $V1_{AA}$ et $V2_{AB}$).

➤ **Arrêt des unités AA et AB**

On appui sur le bouton poussoir $BA1_{AA, AB}$; désexcitation des bobines $RB_{AA, AB}$, $CC_{AA, AB}$ (arrêt des moteurs broche $ME1_{AA, AB}$) ; $CA_{AA, AB}$ (arrêt des avances rapides) ; désexcitation de $CB_{AA, AB}$ (arrêt des reculs rapides).

➤ **Avance rapide de l'unité transfert**

Une pression pilote arrive sur $RP1$ qui s'ouvre pour laisser passer l'huile, et comme CE fermé ; on appui sur le bouton poussoir $BM5$; $CA_{AA, AB}$ et $CB_{AA, AB}$ sont excités ; les électrovannes EVA_{FT} et EVB_{FT} s'excitent, le transfert rapide est enclenché.

La lampe L_{5FT} s'allume (transfert en avant).

➤ **Retour rapide de l'unité transfert**

Une pression pilote arrive sur $RP1$ qui s'ouvre pour laisser passer l'huile, on appui sur le bouton poussoir $BM6$, le fin de course $m2_{FT}$ n'étant pas actionné, son contacte est fermé ; CC_{FT} et CD_{FT} sont excités, leur contacts fermés ; les électrovannes EVC_{FT} et EVD_{FT} s'excitent, le recul rapide est enclenché.

➤ **Desserrage de la pièce**


On appui sur le bouton poussoir $BM4_{FA}$, RJ_{FA} excité, désexcitation de CB_{FA} ; excitation de CA_{FA} , son contacte se ferme ; excitation de l'électrovanne EVA_{FA} .

3-Conclusion :

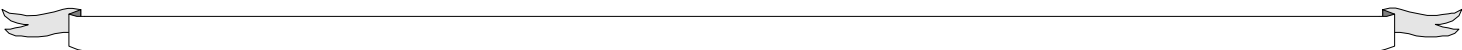
Au terme de l'étude menée sur le fonctionnement de la machine, nous avons pu cerner les manques dans le fonctionnement de la machine, causés par le fait que de nombreuses opérations se font manuellement dont :

- Un deuxième départ cycle pour l'alésage de la ligne intermédiaire (post B).
- Blocage des leviers lors du changement de vitesse (boîte de vitesse à base de pignons).
- Manque de capteurs indispensable au bon fonctionnement.
- Exposition au danger de l'opérateur à l'approche des unités pendant le cycle

Ce diagnostic nous permettra de concevoir au mieux une solution automatisée de la conduite de la machine.



Chapitre 2: Modélisation Graphique du Fonctionnement



1- Introduction :

Le cahier des charges étant la première étape de la conception d'un système automatisé, il est constitué d'une suite de phrases décrivant le fonctionnement désiré de ce système, des phrases qui ne sont pas toujours explicites, ou précises. Afin d'analyser et de valider le cahier des charges, nous le traduira en un formalisme qui ne permet aucune erreur d'interprétation. On parlera de modélisation. L'automatisme séquentiel requiert des outils graphiques (chronogramme, graphe de fluence, tableaux d'état, graphe d'état, graphe d'événement, GRAFCET, Réseaux de Pétri, etc.).

Afin de modéliser notre système industriel, tout en tenant compte des contraintes physiques et logiques du fonctionnement, nous avons choisi le GRAFCET, qui est un langage, facile à maîtriser plus proche de la phase de réalisation du système de commande que celle de l'analyse du procédé en général. Cependant des améliorations sont envisagées pour réduire l'intervention de l'opérateur afin d'assurer sa sécurité et celle du matériel, et évidemment améliorer le rendement de la machine.

2- Proposition d'améliorations :

2-1-Amélioration sur la sécurité :

a) Le détecteur de niveau :

Sur le groupe actuel, il n'existe aucune information sur le niveau d'huile dans le réservoir, ce qui représente un risque pour la pompe hydraulique, donc je propose de placer un détecteur de niveau.

Après une recherche sur les capteurs existants sur le marcher, les contrôleurs de niveau de type FSK qui assurent l'affichage et le contrôle de niveau des fluides hydraulique, semblent les mieux adapter pour cette application.

Le FSK "3 070 285" (voire annexe C) assure un contrôle automatique du niveau du fluide par un signal électrique. Le fluide pénètre dans l'appareil par l'orifice et agit sur le flotteur. Celui-ci est au niveau du fluide dans le réservoir. Lorsque le niveau de fluide baisse, le flotteur génère un signal de commutation.

b) Détecteur photoélectriques :

Afin de minimiser les accidents humains, surtout pendant le chargement (déchargement) de la pièce, une détection de la présence de l'opérateur dans la zone à risque est nécessaire, pour cela un détecteur photoélectrique (photocellule) de type barrage "XU1-P18PP340D" (voire annexe C) qui est composé d'un émetteur de lumière associé à un récepteur photosensible, permettra d'inhiber le déplacement des unités quand l'opérateur effectue le chargement ou le déchargement de la pièce.

2-2- Amélioration sur l'économie :

Détecteur de position :

Afin d'économiser l'énergie et éviter le serrage à vide, ou le déplacement à vide des unités, qui consomme une énergie électrique importante, nous avons jugé utile d'ajouter un détecteur mécanique de présence pièce "XCK-P110" qui sera fixé sur le transfert pièce. La détection de présence sera réalisée lorsque la pièce entrera en contact avec la tête de commande au niveau de son dispositif d'attaque. Le mouvement engendré sur la tête

d'attaque provoquera la fermeture du contact électrique situé dans le corps du capteur (voire annexe C).

3- Modélisation par GRAFCET :

3-1- Définition du GRAFCET :

Le GRAFCET est acronyme de « GRAPhe Fonctionnel de Commande Etape-Transition ». C'est un outil de représentation graphique du comportement de la partie commande (cahier des charges) d'un système automatisé, il a été proposé par l'ADEPA en 1977 et normalisé en 1982 (NF).

Il est représenté selon deux niveaux I, II

- GRAFCET niveau I (spécification fonctionnelles) décrit littérairement la séquence de fonctionnement de l'automate indépendamment de tout choix technologique.
- GRAFCET niveau II (spécification technologiques), prise en compte de la technologie des actionneurs et des capteurs, comme il décrit symboliquement les actions et les séquences de l'automatisme.

3-2- Élément de base du GRAFCET :

Ce modèle est défini par un ensemble constitué :

➤ D'étapes :

Elle correspond à une phase durant laquelle on effectue une ACTION (qui doit être stable) pendant une certaine DUREE (même faible mais jamais nulle), elle est soit active, soit inactive.

Chaque étape est représentée par un carré repéré numériquement.

L'étape initiale est activée au début du fonctionnement, et représenté par un double carré.

➤ D'actions :

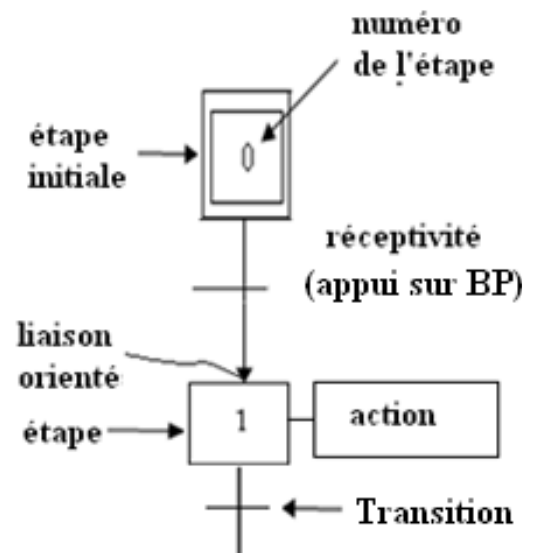
Elles sont associées à une étape, et exécutées lorsque celle-ci est active. Le nom ou le symbole de l'action s'écrit à l'intérieur d'un rectangle relié à l'étape par un tiret. Plusieurs actions peuvent être associées à la même étape, comme elles peuvent être de nature diverses (conditionné par des contraintes externes).

➤ De Transitions :

Elle marque la possibilité d'évolution entre une ou plusieurs étapes. Elle est représentée par un tiret perpendiculaire au trait de liaison.

➤ Réceptivité :

C'est la condition associée à la transition, elle ne peut avoir que deux états ; vrais ; ou faux. L'expression logique de la réceptivité s'écrit à droite de la barre de transition.



Figure(2.1) : schémas illustrant les éléments de base d'un grafcet

➤ **Les liaisons orientées :**

Elles relient les étapes et les transitions
(et vis-versa), et indiquent les voies suivant lesquelles se font les évolutions.

3-3- Les règles d'évolutions :

- ✚ **Règle 1 (situation initiale) :** elle caractérise le comportement initial de la PC vis-à-vis de la PO et correspond à l'ensemble des étapes actives autorisant le début de fonctionnement.
- ✚ **Règle 2 (le franchissement d'une transition) :** il ne peut se produire que si cette transition est validée (toutes les étapes immédiatement précédentes sont actives) et la réceptivité associée est vraie.
- ✚ **Règle 3 (évolution des étapes actives) :** le franchissement d'une transition entraîne l'activation de toutes les étapes immédiatement suivantes simultanément avec la désactivation de toutes les étapes immédiatement précédentes.
- ✚ **Règle 4 (évolution simultanée) :** plusieurs transitions simultanément franchissables sont simultanément franchies.
- ✚ **Règle 5 (activation et désactivation simultanée d'une étape) :** si une même étape doit être à la fois activée et désactivée, elle reste active.

3-4- Sélection de séquence et séquence simultanée

Le GRAFCET présente deux structures particulières :

- ✓ La sélection de séquences.
- ✓ Les séquences simultanées.

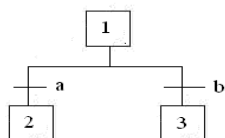
a- Sélection de séquences

- ✚ **Séquence :** On appelle séquence tout ensemble d'étapes successives où chaque étape est suivie d'une seule transition et chaque transition n'est validée que par une seule étape.
- ✚ **Sélection de séquence :** elle représente une alternative d'évolution vers plusieurs étapes.

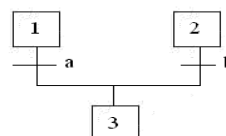
Les réceptivités associées aux transitions d'un aiguillage doivent être exclusives (leur ET logique doit être nul).

La sélection de séquence est aussi appelée divergent OU où divergence de sélection de séquences (figure 2.2-a).

La convergence de plusieurs séquences est appelée convergent OU où convergence de sélection de séquences (figure 2.2-b).



a- Divergence en OU.



b- Convergence en OU.

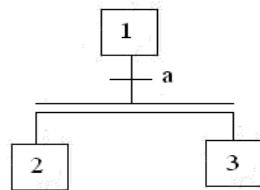
Figure (2.2) : Représentation graphique d'une sélection de séquences.

b- Séquences simultanées

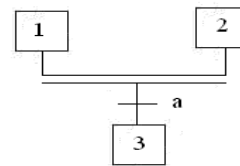
C'est un ensemble de séquences pouvant évoluer indépendamment, à partir du franchissement d'une transition activant simultanément plusieurs étapes.

Une transition qui possède plusieurs étapes de sortie représente l'exécution en parallèle de plusieurs séquences. On appelle cette structure divergent ET ou divergence de séquences simultanées Figure (2.3-a).

Une transition qui possède plusieurs étapes d'entrée représente la synchronisation de plusieurs séquences. On appelle cette structure convergent ET ou convergence de séquences simultanées Figure (2.3-b).



a- Divergence en ET



b- Convergence en ET

Figure (2.3) : Représentation graphique d'une séquence simultanée.

c-Saut d'étapes

Le saut d'une ou plusieurs étapes est réalisé lorsque les actions associées à ces étapes deviennent inutiles Figure (2-4)

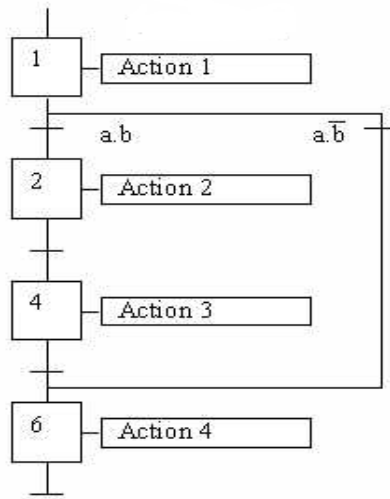


Figure (2-4): Saut d'étape.

d- Reprise de séquence

La reprise de séquence permet de répéter une même séquence jusqu'à ce que la réceptivité de fin de séquence soit vraie Figure (2-5).

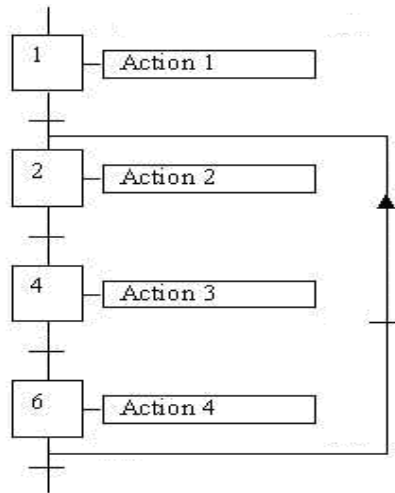
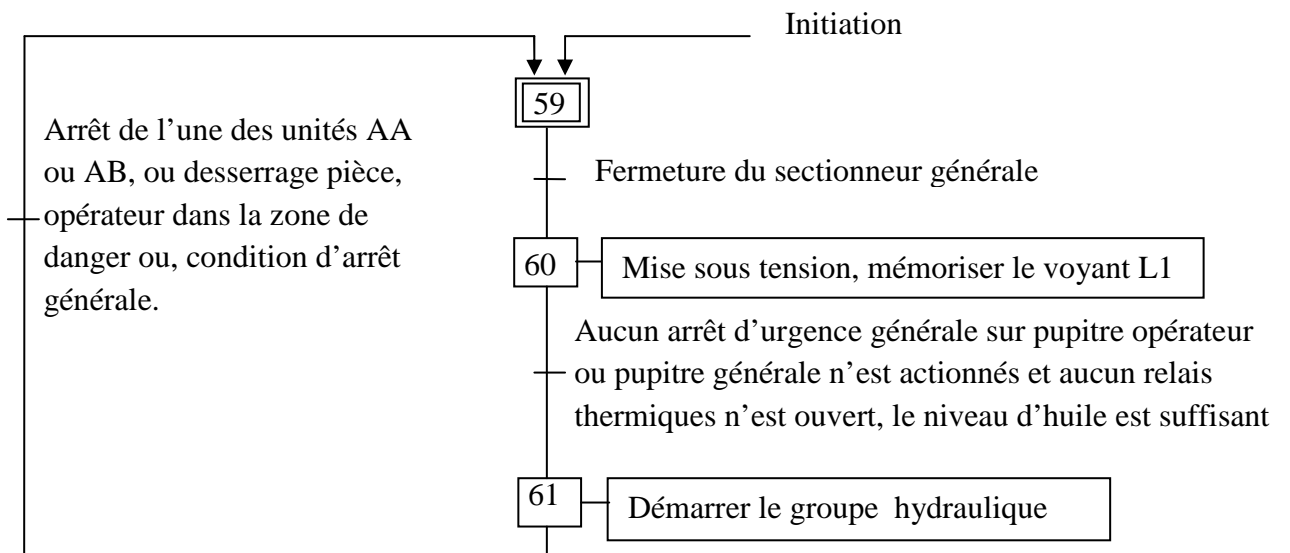


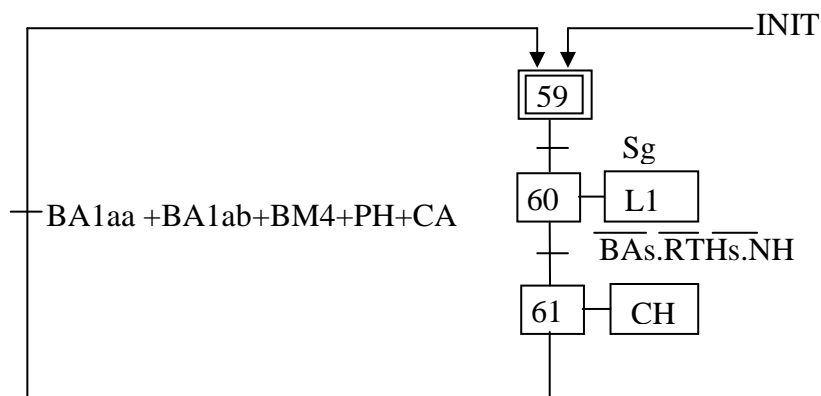
Figure (2-5) : Reprise de séquence.

4- Mise en œuvre du GRAFCET :

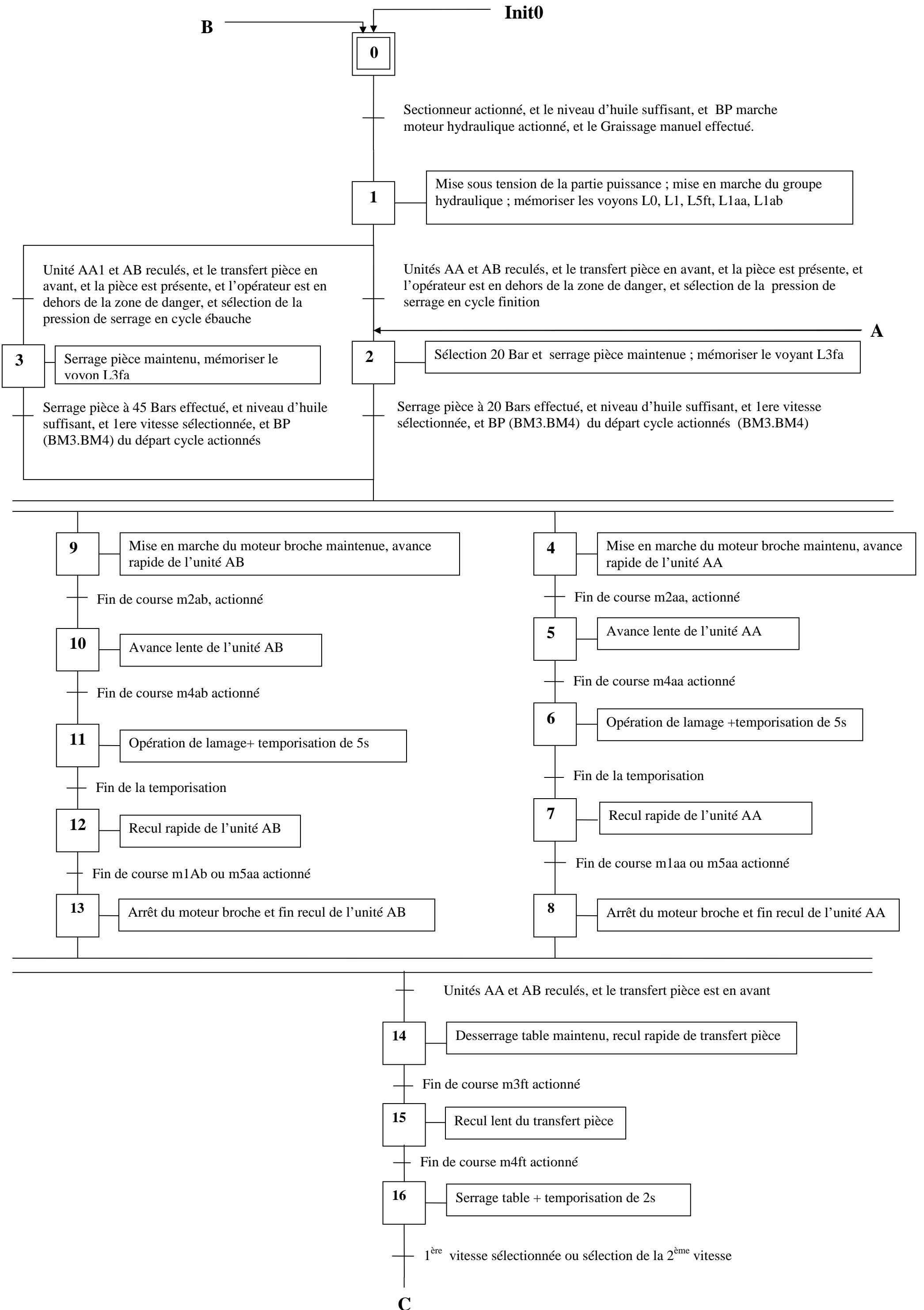
4-1 Grafcet de sécurité niveau 1



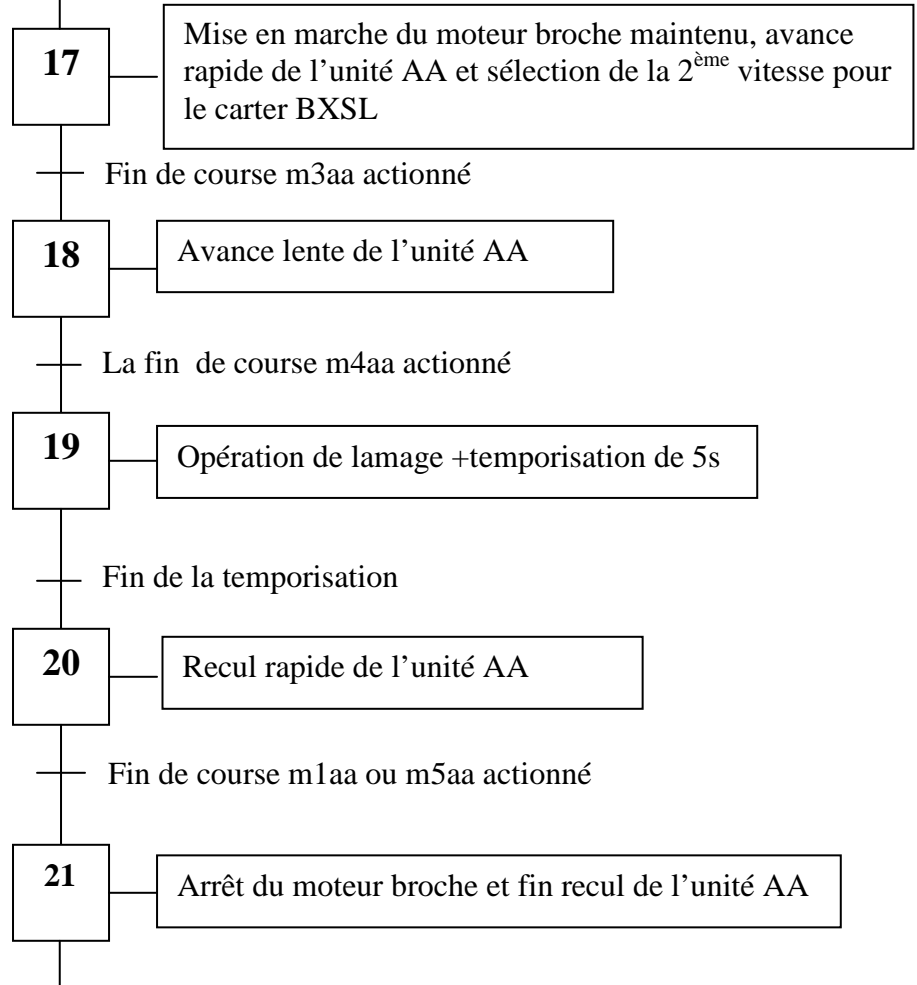
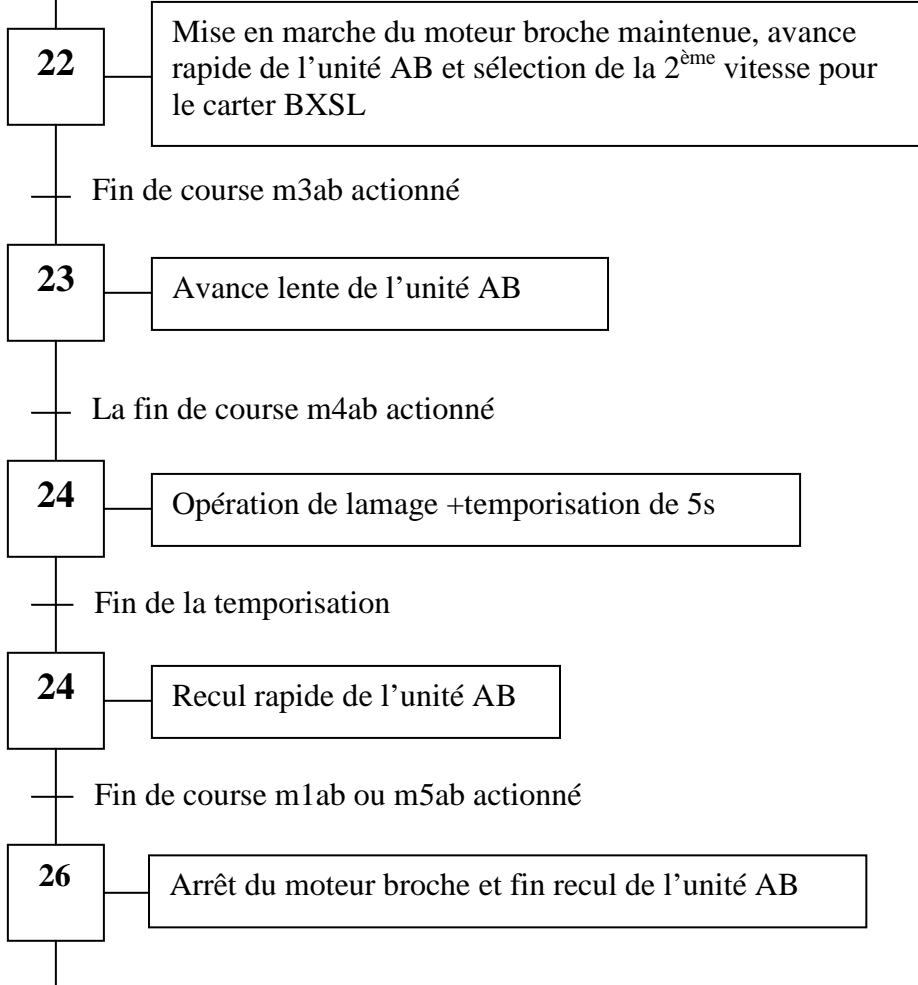
4-2-Grafcet de sécurité niveau 2



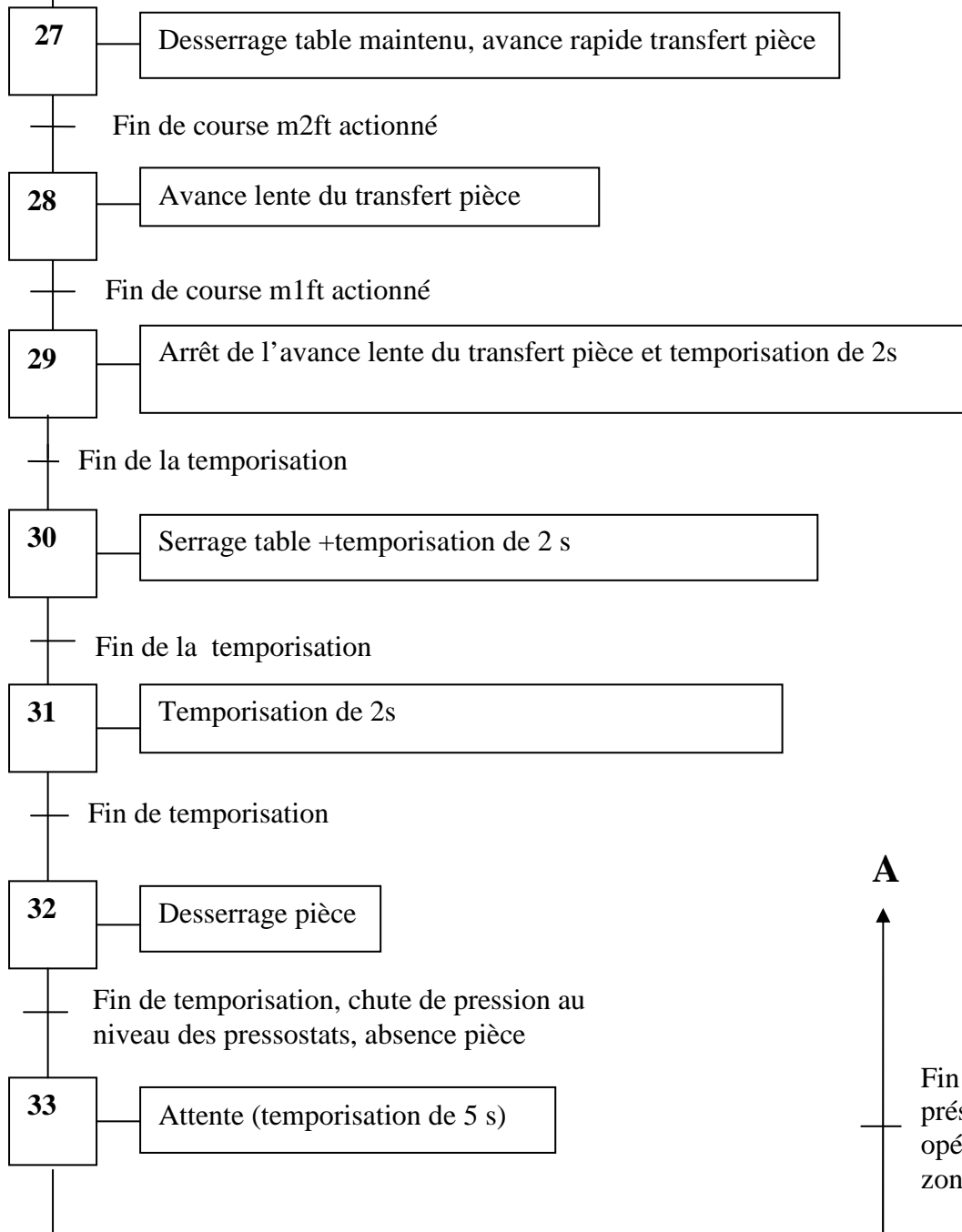
4-3-Grafcet de fonctionnement niveau 1



C



Unité AA, AB reculée et le transfert pièce en arrière.



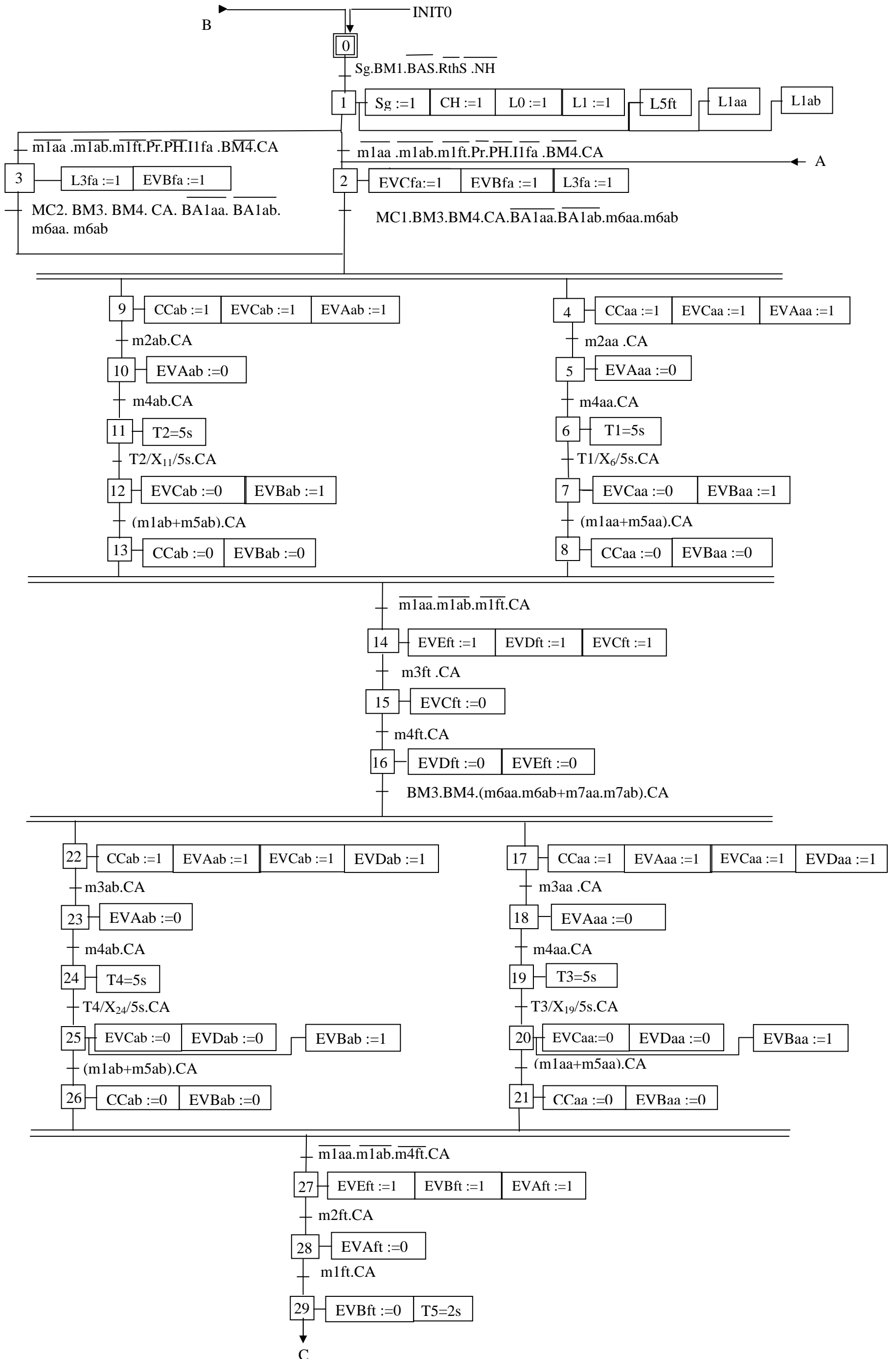
B

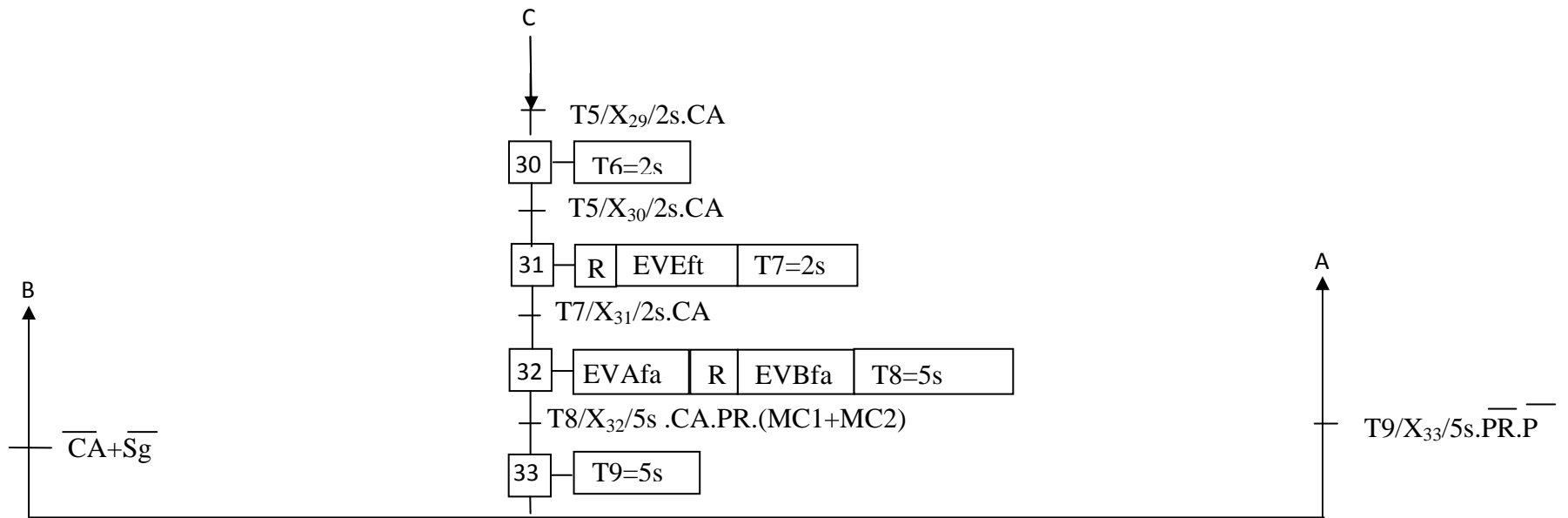
Condition d'arrêt ou mise hors tension

A

Fin de la temporisation, présence pièce, opérateur en dehors de la zone de danger

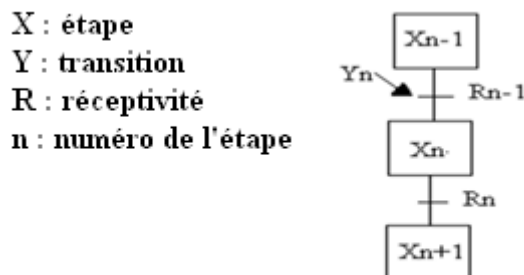
Grafcet Niveau 2





5-Les équations d'activation et de désactivation des étapes :

Pour convertir un GRAFCET en LADDER, il faut écrire les équations booléennes d'activations-désactivations des transitions et des étapes, pour cela on utilise la méthode suivante :



On commence par écrire et programmer les transitions (à partir de la 2^{ème} règle du GRAFCET):

Equation logique de la transition Yn :

$$Y_n = X_n * R_n$$

Elle donne un niveau logique 1 lorsque l'étape qui la précède Xn est active et lorsque la réceptivité Rn est vraie.

Equation logique de l'étape :

$$X_n = Y_{(n-1)} + X_n * /Y_n$$

L'étape Xn s'active si la transition précédente Yn-1 est franchie et se désactive si la transition

Yn suivante est franchie.

Equation logique de l'action A :

$$A = X_1 + X_2 + \dots + X_n + \dots$$

6-Application :

a) Equation des transitions :

$$RTHs = \overline{RTH} * \overline{RTCaa} * \overline{RTCab}$$

$$\overline{BA}s = \overline{BA1} * \overline{BA}$$

$$CA = \overline{BA1} * \overline{BA} * \overline{RTH} * \overline{RTCaa} * \overline{RTCab} * \overline{NH} * \overline{PR} * \overline{PH}$$

Grafcet de sécurité	
Transition « Yn »	Equation de transition associé : $Y_n = X_n * R_n$
Y59	$Sg * X59$
Y60	$\overline{BA}s * \overline{RTH}s * \overline{NH} * X60$
Y61	$(\overline{BA1aa} * \overline{BAab} * \overline{BM4} * \overline{PH} * \overline{CA})X61$
Grafcet de fonctionnement	
Y0	$Sg * \overline{BM1} * \overline{BA}s * \overline{RTH}s * \overline{NH} * X0$
Y1	$\overline{m1aa} * \overline{m1ab} * \overline{m1ft} * \overline{PR} * \overline{PH} * \overline{I1fa} * \overline{BM4} * \overline{CA} * X1$
Y2	$\overline{MC1} * \overline{BM3} * \overline{BM4} * \overline{BA1aa} * \overline{BA1ab} * \overline{m6aa} * \overline{m6ab} * X2$
Y3	$\overline{m1aa} * \overline{m1ab} * \overline{m1ft} * \overline{PR} * \overline{PH} * \overline{I1fa} * \overline{BM4} * \overline{CA} * X1$

Y4	$MC2 * BM3 * BM4 * \overline{BA1aa} * \overline{BA1ab} * m6aa * m6ab * X3$
Y5	$m2aa * CA * X4$
Y6	$m4aa * CA * X5$
Y7	$(T1/X6/5s) * CA * X6$
Y8	$m1aa + m5aa * CA * X7$
Y9	$m2ab * CA * X9$
Y10	$m4ab * CA * X10$
Y11	$(T2/X11/5s) * CA * X11$
Y12	$m1ab + m5ab * CA * X12$
Y13	$m1ab + m5aa * CA * X12$
Y14	$\overline{m1aa} * m1ab * m1ft * CA * X8 * X13$
Y15	$m3aa * CA * X14$
Y16	$m4ft * CA * X15$
Y17	$BM3 * BM4 * (m6aa * m6ab + m7aa * m7ab) * CA * X16$
Y18	$m3aa * CA * X17$
Y19	$m4aa * CA * X18$
Y20	$(T3/X19/5s) * CA * X19$
Y21	$m1aa + m5aa * CA * X20$
Y22	$m3ab * CA * X22$
Y23	$m4ab * CA * X23$
Y24	$T4/X24/5s * CA * X24$
Y25	$(m1aa + m5aa) * CA * X25$
Y26	$m1aa * m1ab * m4ft * CA * X21 * X26$
Y27	$m2ft * CA * X27$
Y28	$m1ft * CA * X28$
Y29	$T5/X29/2s * CA * X29$
Y30	$T6/X30/2s * CA * X30$
Y31	$T7/X31/2s * CA * X31$
Y32	$T8/X32/5s * CA * PR * (MC1 + MC2) * X32$
Y33	$T9/X33/5s * \overline{PR} * \overline{PH} * CA * X33$
Y34	$(\overline{CA} + \overline{Sg}) * X33$

c) Equation des étapes :

Grafcet de sécurité	
Etape Xn	Equation de l'étape associée : $Xn = Yn - 1 + Xn * \overline{Yn}$
X59	$(INIT + Y61) + X59 * \overline{Y59}$
X60	$Y59 + X60 * \overline{Y60}$
X61	$Y60 + X61 * \overline{Y61}$
Grafcet de fonctionnement	
X0	$(INIT0 + Y34) + (X0 * \overline{Y0})$

X1	$[Y0 + (X1 * (\overline{Y1} + \overline{Y3}))] * \overline{RTHs} * \overline{BA_s}$
X2	$[Y1 + (X2 * (\overline{Y2} + \overline{Y33}))] * \overline{RTHs} * \overline{BA_s}$
X3	$[Y3 + (X3 * \overline{Y4})] * \overline{RTHs} * \overline{BA_s}$
X4	$[(Y2 + Y4) + (X4 * \overline{Y5})] * \overline{RTHs} * \overline{BA_s}$
X5	$[Y5 + (X5 * \overline{Y6})] * \overline{RTHs} * \overline{BA_s}$
X6	$[Y6 + (X6 * \overline{Y7})] * \overline{RTHs} * \overline{BA_s}$
X7	$[Y7 + (X7 * \overline{Y8})] * \overline{RTHs} * \overline{BA_s}$
X8	$[Y8 + (X8 * \overline{Y12})] * \overline{RTHs} * \overline{BA_s}$
X9	$[(Y2 + Y4) + (X9 * \overline{Y9})] * \overline{RTHs} * \overline{BA_s}$
X10	$[Y9 + (X10 * \overline{Y10})] * \overline{RTHs} * \overline{BA_s}$
X11	$Y10 + X11 * \overline{Y11} * \overline{RTHs} * \overline{BA_s}$
X12	$Y11 + X12 * \overline{Y12} * \overline{RTHs} * \overline{BA_s}$
X13	$Y12 + X13 * \overline{Y13} * \overline{RTHs} * \overline{BA_s}$
X14	$Y13 + X14 * \overline{Y14} * \overline{RTHs} * \overline{BA_s}$
X15	$Y14 + X15 * \overline{Y15} * \overline{RTHs} * \overline{BA_s}$
X16	$Y15 + X16 * \overline{Y16} * \overline{RTHs} * \overline{BA_s}$
X17	$Y16 + X17 * \overline{Y17} * \overline{RTHs} * \overline{BA_s}$
X18	$Y17 + X18 * \overline{Y18} * \overline{RTHs} * \overline{BA_s}$
X19	$Y18 + X19 * \overline{Y19} * \overline{RTHs} * \overline{BA_s}$
X20	$Y19 + X20 * \overline{Y20} * \overline{RTHs} * \overline{BA_s}$
X21	$Y20 + X21 * \overline{Y25} * \overline{RTHs} * \overline{BA_s}$
X22	$Y16 + X22 * \overline{Y21} * \overline{RTHs} * \overline{BA_s}$
X23	$Y21 + X23 * \overline{Y22} * \overline{RTHs} * \overline{BA_s}$
X24	$Y22 + X24 * \overline{Y23} * \overline{RTHs} * \overline{BA_s}$
X25	$Y23 + X25 * \overline{Y24} * \overline{RTHs} * \overline{BA_s}$
X26	$Y24 + X26 * \overline{Y25} * \overline{RTHs} * \overline{BA_s}$
X27	$Y25 + X27 * \overline{Y26} * \overline{RTHs} * \overline{BA_s}$
X28	$Y26 + X28 * \overline{Y27} * \overline{RTHs} * \overline{BA_s}$
X29	$Y27 + X29 * \overline{Y28} * \overline{RTHs} * \overline{BA_s}$
X30	$Y28 + X30 * \overline{Y29} * \overline{RTHs} * \overline{BA_s}$
X31	$Y29 + X31 * \overline{Y30} * \overline{RTHs} * \overline{BA_s}$
X32	$Y30 + X32 * \overline{Y31} * \overline{RTHs} * \overline{BA_s}$
X33	$Y31 + X33 * (\overline{Y32} + \overline{Y33}) * \overline{RTHs} * \overline{BA_s}$

7-Affectation des entrées/sorties :

L'unité	Entrée	Type d'entrée	désignation
Transfert	BM1 _{FT}	Bouton poussoir	
	BM2 _{FT}		Serrage de la pièce
	BM3 _{FT}		Départ cycle 1 ^{er} usinage
	BM4 _{FT}		Départ cycle 2 ^{em} usinage
	BM5 _{FT}		Avance du transfert pièce
	BM6 _{FT}		Recul du transfert pièce


Unité AA	BM1		Avance unité AA
	BM2		Recul unité AA
	BA1		Arrêt unité AA
Unité AB	BM1		Avance unité AB
	BM2		Recul unité AB
	BA1		Arrêt unité AB
Unité FA	BM3		Démarrage
	I1 _{FA}	Commutateur deux Positions clé 455	Finition ou Ebauche
Transfert	I1 _{FT}	Commutateur deux Positions clé 455	Carter (B×5L-Normal)
	I2 _{FT}		Départ de cycles 1 ou 2
	I3 _{FT}		Auto ou manuel
Circuit général	BA	Bouton poussoir	Arrêt général
	BA1		Arrêt général
	BM2		Essai lampes

CAPTEURES	m1AA, m1AB m2AA, m2AB m3AA, m3AB m4AA, m4AB m5AA, m5AB m6AA, m6AB m7AA, m7AB m1FT m2FT m3FT m4FT pressostat : • MC1 • MC2	<ul style="list-style-type: none"> - Unités reculés - Arrêt de l'avance rapide poste A - Arrêt de l'avance rapide poste B - Arrêt de l'avance lente - Fin de recul extrême - Autorise départ unité au poste A - Autorise départ unité au poste B - transfert pièces en avant - décélération - décélération - transfert pièces en arrière - sélection pression de serrage - finition (20 bars) - débouche (45 bars)
	Type de sortie	
ELECTROVANES	D1aa • EVA • EVB	V1aa Avance rapide Recul rapide
	D2aa • EVC	Avance lente
	D3aa • EVD	Sélection des avances lentes
	D1ab • EVA • EVB	V2ab Avance rapide Recul rapide
	D2ab	

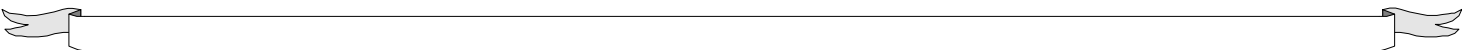
	<ul style="list-style-type: none"> • EVC D3ab <ul style="list-style-type: none"> • EVD D1ft <ul style="list-style-type: none"> • EVA • EVC D2ft <ul style="list-style-type: none"> • EVB • EVD D3ft <ul style="list-style-type: none"> • EVE D1fa <ul style="list-style-type: none"> • EVA • EVB D2fa <ul style="list-style-type: none"> • EVC 	Avance lente Sélection des avances lentes V1ft Avance rapide Recul rapide V2ft Avance lente Recul lente V2ft, V3ft, V4ft, V5ft Desserrage table V1fa, V2fa Serrage Desserrage V1fa, V2fa Sélection 20 bars
MOTEURS	ME1aa ME1ab ME1ha	

8-Conclusion

Le GRAFCET reste un outil graphique très puissant permettant de modéliser très facilement le fonctionnement de la machine. Cet outil permet donc d'élaborer un modèle de fonctionnement sur lequel peuvent être étudiées les propriétés et les performances du système de conduite de la machine. L'étape suivante consistera à programmer les équations du modèle obtenu en langage d'un automate. Avant d'entamer cette étape, il faudrait porter le choix sur le type d'automate à utiliser.



**Chapitre 3:
Généralités sur Les
Automates Programmables**

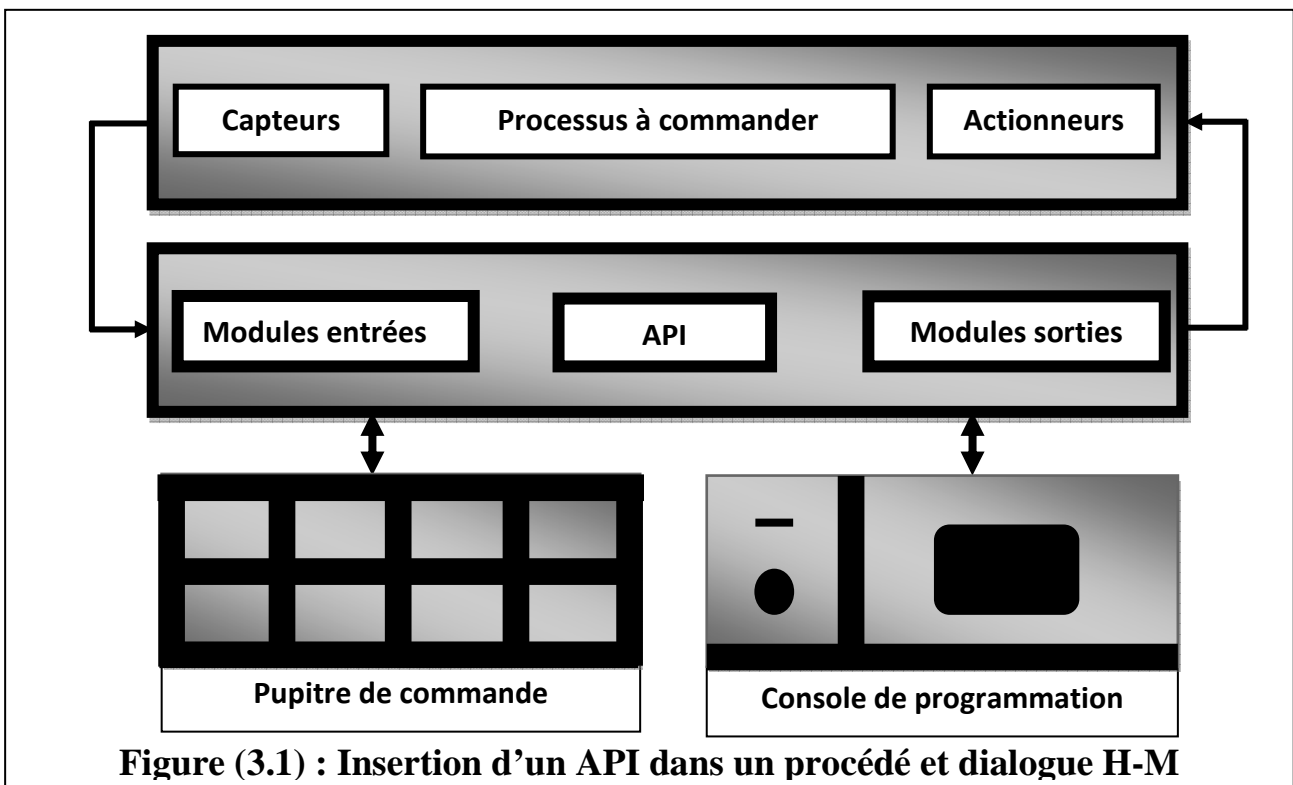


1- Introduction :

Un automate programmable industriel (API) est essentiellement un dispositif électronique possédant l'architecture d'un ordinateur mais destiné à être utilisé en milieu industriel. Cette vocation induit des caractéristiques distinctives dont les principales sont :

- ✚ grande capacité de communication avec un environnement industriel.
- ✚ grande fiabilité nécessitée par le caractère industriel de son utilisation.
- ✚ logiciels de programmation non standardisés.
- ✚ possibilités de dialogue homme-machine.

Dans la figure suivante Figure (3.1), on présente la commande par un API dans son contexte industriel, c'est-à-dire son insertion dans un procédé, telle qu'il est encore utilisé dans les applications de petites ou grandes dimensions.



2- Présentation d'un automate:

2-1- Introduction :

L'automate programmable que nous présentons est le S 7-300 de Siemens, c'est un automate de la famille S7, il est donc destiné à fonctionner également en réseau. C'est un automate de gamme intermédiaire, et comme les autres automates, le S7-300 est configurable selon les besoins de l'utilisateur, il est constitué d'une configuration minimale composée d'un module d'alimentation, d'une CPU et d'un module d'entrée ou de sortie comme le montre la Figure (3.2) qui suit.

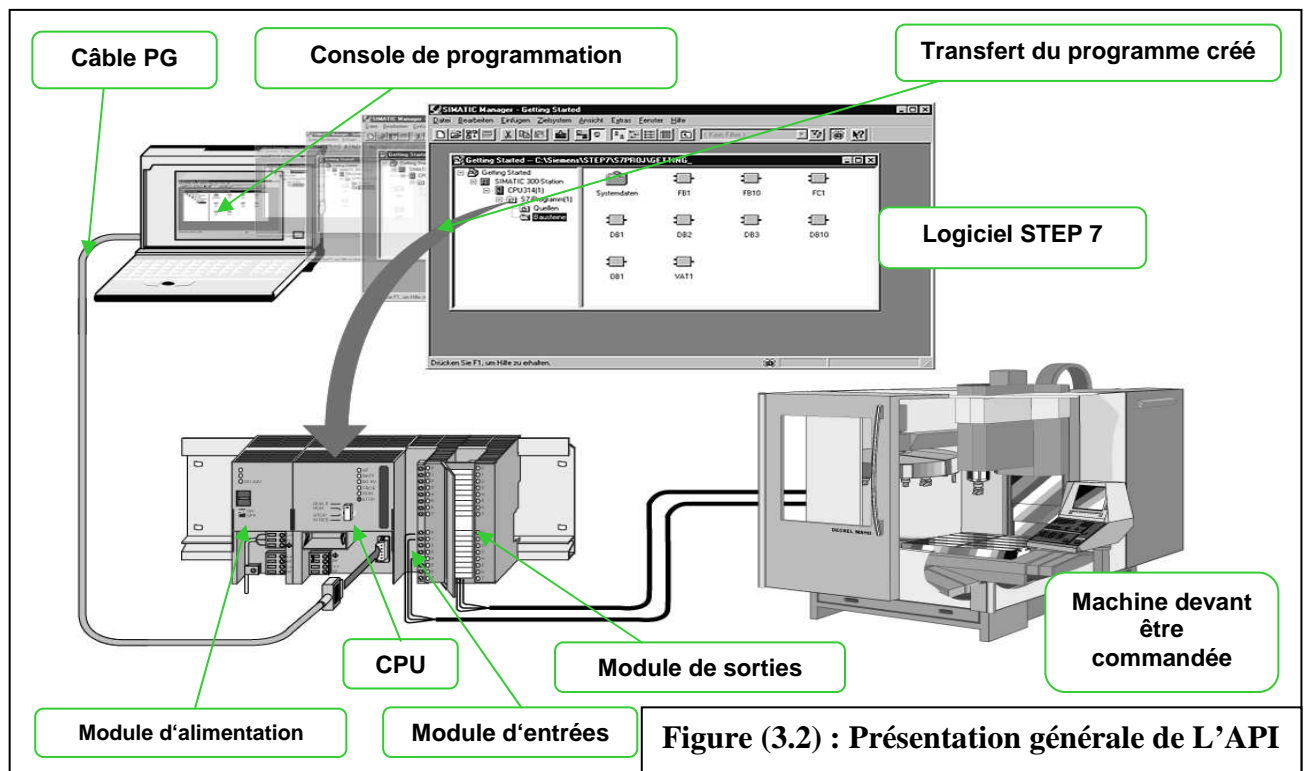


Figure (3.2) : Présentation générale de L'API

2-2- Définition d'un système automatisé :

Un système de production est dit automatisé, lorsqu'il peut gérer de manière autonome un cycle de travail préétabli qui se décompose en séquences ou étapes.

2-3- Structure générale d'un système automatisé :

Un système de production est dit automatisé, lorsqu'il peut gérer de manière autonome un cycle de travail préétabli qui se décompose en séquences ou étapes.

Les systèmes automatisés utilisés dans le secteur industriel possèdent une structure de base identique, voir Fig.3.3, à savoir :

- ✚ **Une partie opérative (PO) :** C'est la partie puissance, celle qui agit (le muscle). Elle regroupe tous les mécanismes et dispositifs opératifs qui assurent les différent mouvement, et l'acquisition des données à savoir des actionneurs, des pré-actionneurs et des capteurs (Tableau)
 - ✓ **Des pré-actionneurs :** reçoivent les ordres de la partie commande.
 - ✓ **Des actionneurs :** ont pour rôle d'exécuter ces ordres.
 - ✓ **Des détecteurs :** informent la partie commande de l'exécution du travail.
- ✚ **Une partie commande (PC) :** C'est la partie qui élabore les ordres nécessaires à l'exécution d'un processus (pour la partie opérative) et reçoit en retour des comptes rendus qui l'informent sur l'état des opérations effectuées (à partir des capteurs).
- ✚ **Une partie relation ou pupitre de dialogue (PR) :** C'est un bloc de commande contenant les boutons poussoirs, des commutateurs et des voyants, qui établi la communication entre l'opérateur et la partie opérative de la machine

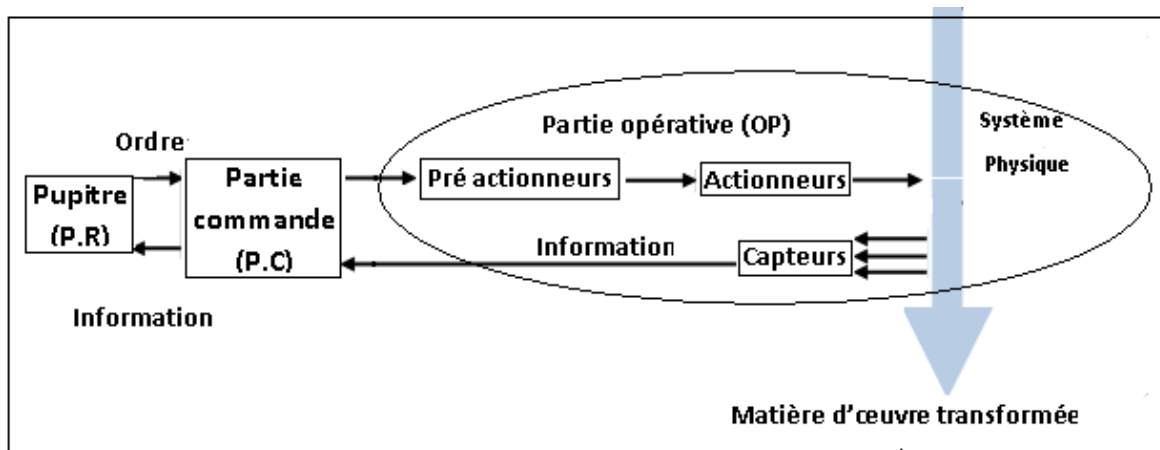


Figure (3.3) : Structure d'un système automatisé.

2-4- Système de commande :

L'automaticien dispose de nombreux outils technologiques pour réaliser l'organe de commande de son système. Ils sont classés habituellement dans l'une des commandes, à savoir :

2-4-1- Logique câblée :

C'est celle qui est utilisée au niveau de la SNVI. Sa mise en œuvre nécessite uniquement l'établissement de liaisons matérielles (câblage) selon un schéma établi à partir de la théorie ou de la pratique. Cette technologie est simple, mais toute modification dans le choix du fonctionnement de l'installation entraîne :

- ✚ une augmentation du nombre de relais et de fils de câblage (investissement à coût élevé).
- ✚ une intervention dont le câblage nécessite une main d'œuvre spécialisée, des auxiliaires de commandes. Par conséquent cela entraîne un encombrement dans les armoires électriques.

2-4-2- Logique programmée :

C'est la solution proposée. Elle utilisera un automate programmable industriel (API). L'encombrement se trouvera réduit et la recherche de pannes sera plus facile, ce qui nécessitera une main d'œuvre réduite lors du câblage. Par conséquent des modifications seront possibles sans intervention sur le câblage.

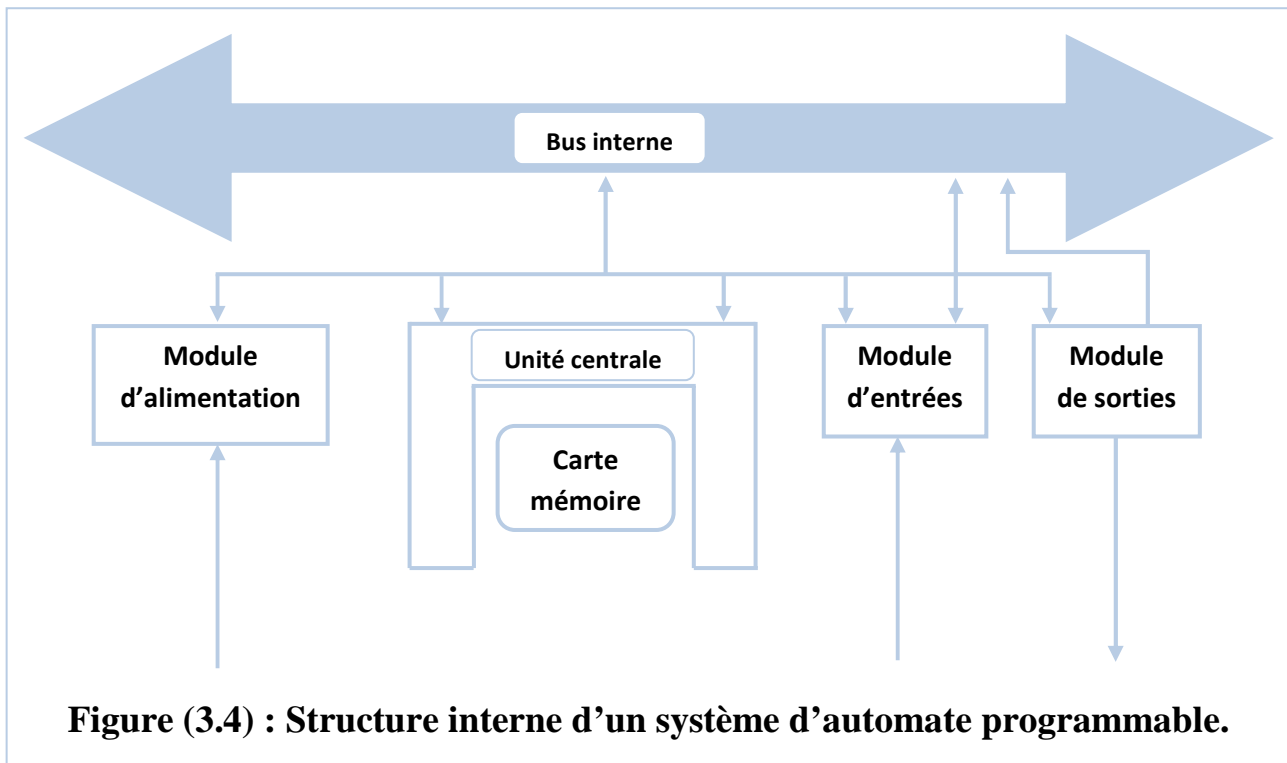
3- Définition d'un automate programmable industriel :

C'est un appareil électronique qui comporte une mémoire programmable, utilisé à l'aide d'un langage adapté. Il commande, mesure et contrôle au moyen de modules d'entrées et de sorties des processus.

4- Structure interne d'un automate programmable :

La structure d'un automate programmable (AP), ressemble à celle d'un micro-ordinateur, il comporte principalement les parties suivantes : l'unité centrale, le bloc d'alimentation, les modules d'entrées / sorties, les modules enfichables ajoutés en guise de périphérie.

Ces parties sont reliées entre elles par un bus, comme le montre la figure suivante (3.4) :



4-1- L'unité centrale (UC) :

C'est l'ensemble des dispositifs nécessaires au fonctionnement logique interne de l'API, elle comprend les parties suivantes :

4-1-1- Le processeur: c'est la partie intelligente de l'UC, appelé l'unité de traitement (UT). Il est chargé d'assurer le contrôle de l'ensemble de la machine et d'effectuer les traitements demandés par les instructions du programme. Il comprend différents registres:

- ❖ **Un registre:** est une mémoire rapide à semi-conducteurs, complétée de dispositifs logiques permettant la manipulation des informations qu'elle contient, ou de leurs combinaisons pour communiquer avec le milieu extérieur.

- a. **Registre interne:** accessible à la machine seulement, il assure les fonctions de gestion interne et de contrôle.

- b. **Compteur ordinal (pointeur):** contient en permanence l'adresse de l'instruction en cours d'exécution. Il décrit donc la zone mémoire où est rangé le programme actif.

- c. **Registre d'instructions (code opératoire) :** reçoit de la mémoire centrale le code opération (CO) de l'instruction à exécuter, désigné par le compteur ordinal. Il interprète ou décode le CO, positionne les entrées de validations des circuits logiques et des registres qui exécutent l'instruction.

- d. **Registre d'adresse (adresse opérande) :** reçoit parallèlement du registre d'instruction, la partie adresse opérande (AO) de l'instruction chargée depuis la mémoire centrale.

- e. **Registre accumulateur :** registre où s'effectuent les opérations du jeu d'instruction l'UC.

4-1-2- Mémoire : elle est conçue en vue de stocker toutes les informations du système; elle dialogue d'une part avec le processeur et d'autre part avec les organes d'entrées / sorties.

4-2- Bloc d'alimentation :

Il permet de fournir à l'automate l'énergie nécessaire à son fonctionnement, à partir d'une alimentation en 220 V (\approx) ; ce bloc délivre les tensions dont l'automate a besoin : 24, 12 ou 6 volts.

4-3- Les modules enfichables :

Le raccordement d'autres modules à l'automate est prévu pour l'obtention des fonctions supplémentaires ; ces modules sont:

4-3-1- Les cartes mémoires (RAM, EPROM) : ces cartes offrent un supplément de capacité mémoire pour les données et les programmes d'applications.

4-3-2- Les coupleurs : ce sont des cartes électriques qui assurent la communication et l'échange d'information entre l'UC et :

- ✚ les modules d'entrées / sorties par l'intermédiaire d'un bus interne.
- ✚ les périphériques de l'automate (console, imprimante, micro-ordinateur) par un bus interne.

4-3-3- Les cartes d'entrées / sorties:

Ces cartes sont équipées de connecteurs de raccordement permettant un montage et démontage rapide. Les automates programmables offrent une grande variété d'entrées / sorties tout ou rien (TOR) adaptées aux milieux auxquels ils sont soumis.

4-3-4- La console de programmation:

Elle est propre à l'automate, elle permet à l'opérateur le dialogue avec la machine par les échanges d'informations (modification, lecture et suivi de valeur). La console de programmation est conçue de manière compacte, tout particulièrement adapté à un environnement industriel.

5- Modules d'entrées/sorties :**5-1- Les modules d'entrées tout ou rien (TOR):**

Un module d'entrée doit permettre à l'unité centrale de l'automate d'effectuer une «lecture» de l'état logique des capteurs qui lui sont associés (module 4, 8, 16 ou 32 entrées). A chaque entrée correspond une voie qui traite le signal électrique pour élaborer une information binaire. Une diode électroluminescente « LED » située sur la carte donne l'état de chaque entrée.

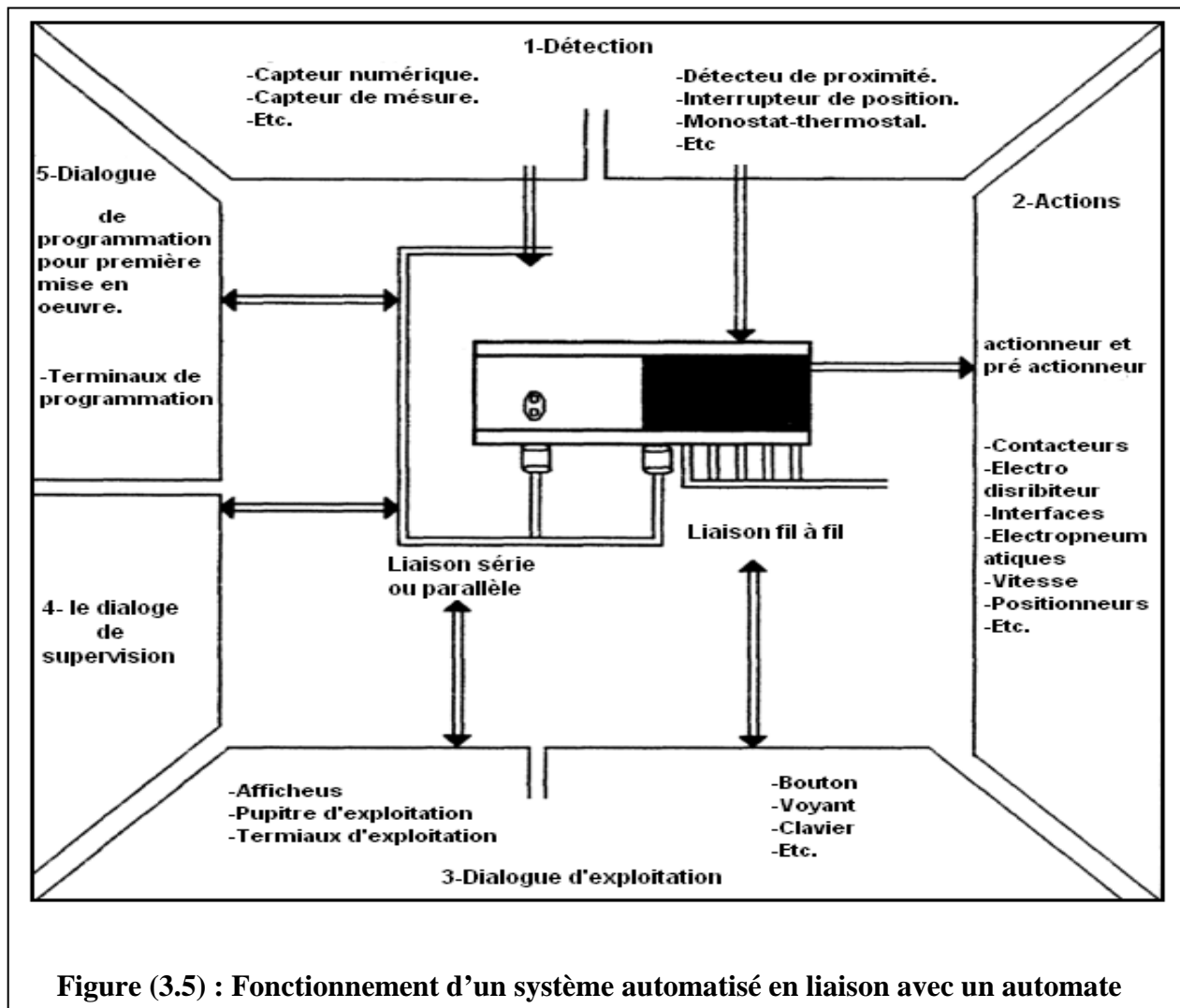
5-2- Les modules de sorties tout ou rien (TOR):

Un module de sortie permet à l'API d'agir sur les actionneurs. Il réalise la correspondance (état logique - signal électrique) et délivre l'information de l'automate vers le processus. L'état de chaque sortie est visualisé par une diode électroluminescente.

6- Fonctionnement d'un automate programmable :

Les fonctions susceptibles d'être amenées par un automate programmable sont:

Détection, dialogue d'exploitation, commande d'actions, dialogue de supervision, et dialogue de programmation (Voir Figure (3.5)).



6-1- Détection :

Les capteurs binaires ou tout ou rien «TOR» tels que : détecteur de proximité, interrupteur de position, thermostat, pressostat, sont raccordés à l'automate programmable par des liaisons fil à fil aboutissant aux modules d'entrées.

6-2 La commande d'action :

A chaque actionneur sur la machine (moteur, vérin) est associé un pré actionneur (contacteur, distributeur,...). L'actionneur TOR, est raccordé à l'API par des liaisons fil à fil depuis les modules de sorties.

6-3- Dialogue d'exploitation :

En période d'exploitation, des dialogues homme-machine sont nécessaires pour la conduite de la machine, afin de procéder à des réglages et à des dépannages.

- ✚ les constituants implantés tels que: boutons poussoirs automatiques, voyants, pupitres sont configurables.

- ✚ des terminaux d'exploitation amovibles, pour le réglage et dépannage.

6-4- Dialogue de supervision :

Les systèmes automatisés s'intègrent dans une production souvent gérée et supervisée d'une manière centrale. L'API est donc prévue pour dialoguer en liaison série avec des périphériques centraux.

6-5- Dialogue de programmation :

La première mise en œuvre de la machine ainsi que d'éventuelles évolutions exigent ce dialogue de programmation. La mise en œuvre des terminaux de programmation prévus à cet effet est simplifiée par l'utilisation de logiciels informatiques.

7- Choix d'un automate programmable industriel :

Les critères de choix d'un API sont:

- ✚ le nombre et la nature des entrées / sorties.
- ✚ la nature du traitement (temporisation, comptage,...).
- ✚ le dialogue (la console détermine le langage de programmation).
- ✚ les moyens de sauvegarde du programme (disquettes, cassettes,...).
- ✚ la fiabilité et la robustesse.
- ✚ l'immunité aux parasites.
- ✚ le service après vente et la durée de garantie.
- ✚ la formation.

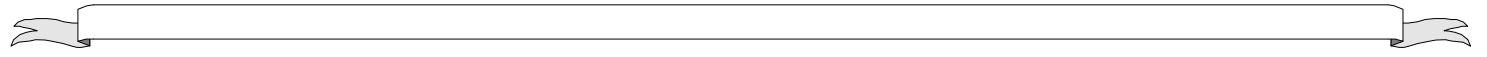
Un API disposant d'un langage GRAFCET est préférable à un autre dépourvu de cet attribut car il permet de bien structurer une application séquentielle

Sur la base de ces critères, j'ai opté pour un automate S7-300 de la firme SIEMENS, pour la commande de la machine. Cette technologie n'est pas étrangère à la SNVI puisqu'elle existe déjà au niveau du département mécanique.

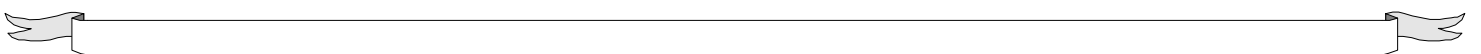
8- Conclusion :

L'automate programmable industriel (API), de part sa souplesse d'utilisation, son encombrement réduit et ses possibilités d'adaptation à tous les milieux industriels, est l'outil le plus privilégié de la commande des systèmes automatisés. C'est un bon produit facile à programmer, à connecter et à être adapté aux conditions industrielles. En plus de sa facilité à résoudre des problèmes, l'utilisation de l'API offre plusieurs avantages :

- ✚ le respect des règles d'installations.
- ✚ un bon choix du réseau de communication et de son protocole.
- ✚ un léger sur dimensionnement pour préserver des marges de modification.



Chapitre 4:
Etude et développement du
logiciel de programmation en
STEP7.






1- Introduction:

Pour pouvoir dialoguer avec une autre personne, il est impératif de posséder un moyen de communication que nous avons l'habitude d'appeler le langage.

En ce qui concerne l'automate programmable, il existe différents langages de programmation, qui peuvent être implémentés soit à partir d'une console correspondant à l'automate, soit à partir d'un logiciel propre à installer sur PC.

2- Logiciel STEP7 :

STEP7 est un logiciel de base pour la configuration et la programmation des systèmes d'automatisation SIMATIC. Il est formé d'un ensemble d'applications avec lesquelles nous pouvons aisément réaliser des tâches partielles comme:

-  la configuration et le paramétrage du matériel.
-  la création et le test de programme utilisateur.
-  la configuration de réseau et de liaison.

3-Gestionnaire de projets SIMATIC MANAGER :

Le gestionnaire de projets SIMATIC Manager gère toutes les données relatives à un projet d'automatisation, il démarre automatiquement les applications requises pour le traitement des données sélectionnées

4-Editeur de mnémoniques :

Il permet la gestion de toutes les variables globales. En effet, il définit des désignations symboliques et des commentaires pour les signaux du processus (Entrées/Sorties), les mémentos, les blocs de données, les temporisations et les compteurs.

4-1- Définition d'un mnémonique :

Dans un programme de STEP7, on utilise des opérandes comme des signaux des EIS, des mémentos, des capteurs, des temporisations, des blocs de données et des blocs fonctionnels.

On peut accéder deux opérandes de notre programme utilisateur via ces mnémoniques.

4-1- Adressage absolu :

Une adresse absolue est composée d'un identificateur d'opérandes et d'une adresse (par exemple : A4.0, E 1.1, M2.0).

4-2- Adressage symbolique :

Une adresse symbolique consiste à indiquer de manière symbolique l'opérande à traiter (au lieu d'indiquer son adresse). Un mnémonique est donc affecté à l'adresse.

5- Langage de programmation :

La programmation peut s'effectuer dans trois modes de langages dont leurs représentations sont désignées par (CONT), (LIST), (LOG).

5-1- En «Schéma à contact » symbolisé par : (CONT) :

C'est un langage de programmation graphique. La syntaxe des instructions fait penser aux schémas de contacts des circuits électriques.

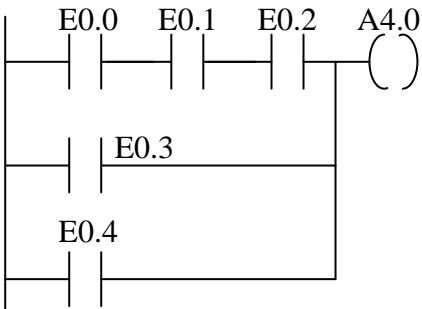
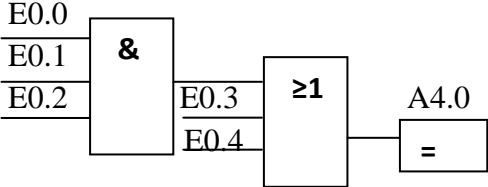
5-2- En « liste d'instruction » symbolisé par : (LIST) :

C'est un langage de programmation textuel de la machine.

5-3- En « logigramme » symbolisé par : (LOG) :

C'est un langage de programmation graphique qui utilise les boites de l'algèbre de Boole pour représenter les opérations logiques.

Le tableau ci dessous illustre les 3 modes de langage de programmation :

Schéma à contact	List d'instruction	Schéma logique
	<pre> U E0.0 UN E0.1 U E0.2 O E0.3 O E0.4 = A4.0 </pre>	
Contact	List	Logigramme

6- Programmation sous STEP7 :

Pour concevoir un projet avec STEP7, il existe 2 approches figure (4.1) :

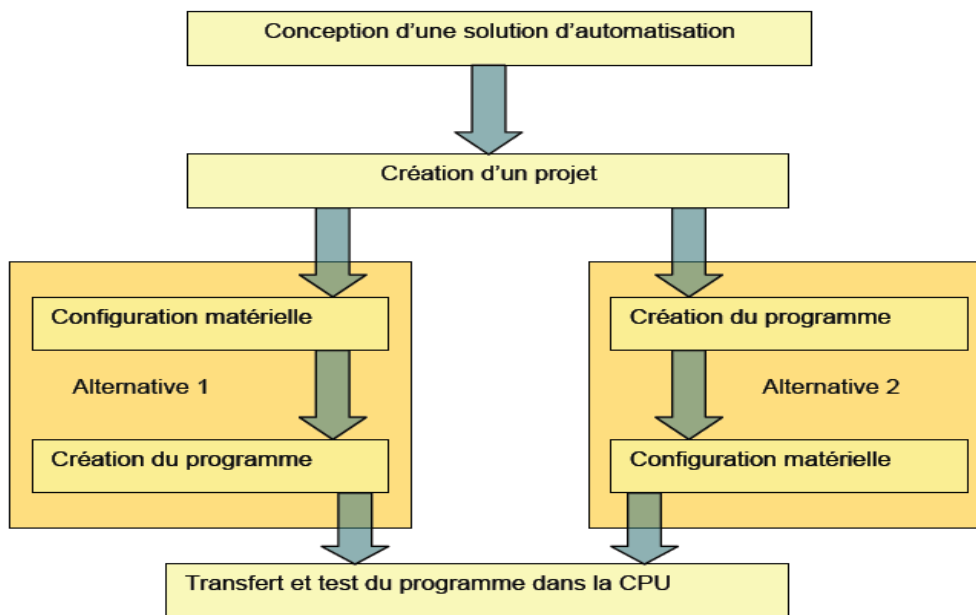


Figure 4-1 : différente méthode de création d'un projet

Il faut noter que pour un système contenant beaucoup de variables, la seconde alternative n'est pas très pratique.

Nous optons donc pour la première approche

6-1- Création d'un projet :

Pour créer un projet avec STEP 7, on peut lancer l'assistant de création de projet de STEP7, ou créer directement un projet que l'on configurera soi-même, cet assistant permet de créer un projet avec une interface simple. Les étapes à suivre sont :

Etape 1 :

- choisir la CPU à utiliser pour le projet (312 IFM),
- Choisir l'adresse MP (l'adresse MPI est réglée par défaut à la valeur 2).

Etape 2 :

- Choisir le bloc d'organisation OB,
- Choisir un langage de programmation (CONT).

Après avoir créé un projet, on passe à la configuration et le paramétrage du matériel.

6-2- Configuration du matériel (hardware configuration) :

L'appareillage réel (PS, CPU, etc), le profilé support des modules et de cartouche interface, est représenté par une table de configuration dans une fenêtre de station, dans laquelle on peut enficher un nombre définis de modules, qu'on amène à partir de la fenêtre "catalogue de matériel". L'enfichage des modules dans la table ; se fait comme dans le profilés support (châssis) réels; c'est-à-dire le PS dans la 1^{ère} ligne, la CPU dans la 2^{ème}, le IM dans la 3^{ème}, puis les SMS (ou FM, CP, ..) viennent s'enficher à partir de la 4^{ème} ligne (figure 4-2).

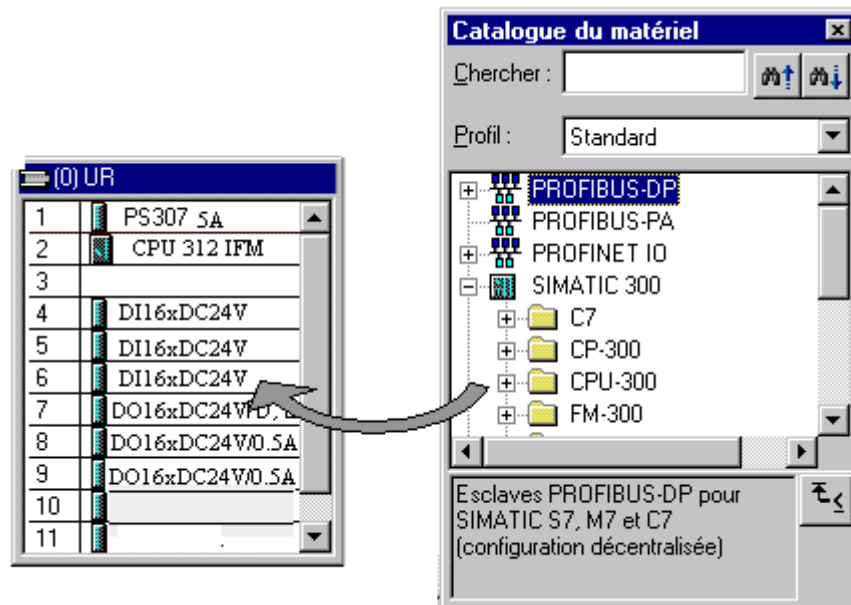


Figure 4-2 : Enfichage des Modules dans la fenêtre de station.

7- Bloc dans le programme utilisateur :

7-1-Définition :

Un bloc correspond à une partie du programme utilisateur, délimité par sa fonction, sa structure et par son application.

Dans STEP7 il existe d'une part :

- ✚ des blocs des codes (FB, FC, OB, SFB, SFC).
- ✚ des blocs des données (DB, SDB).
- ✚ et d'autres types de données utilisateurs (UDT).

Les blocs sont rangés dans le dossier « blocs » sous le programme S7.

7-2-Bloc de code : Il existe les blocs de code suivants:

a. Bloc d'organisation (OB): Il constitue l'interface entre le système d'exploitation du CPU et le programme utilisateur, c'est dans ce bloc que l'on définit l'ordre de traitement du programme utilisateur. Cela revient à dire que l'exécution d'un OB peut être interrompue par l'appel d'un autre OB plus prioritaire.

b. Bloc fonctionnel (FB): un FB permet de transmettre des paramètres dans le programme utilisateur. Il contient un programme qui est exécuté quand ce bloc qui est appelé par autre bloc de code. Le bloc FB facilite la programmation des fonctions complexes souvent utilisées. Un FB possède une mémoire (bloc de donnée d'instance).

c. Bloc des données d'instance (SDB): un DB d'instance mémorise les paramètres formels et les données statiques de blocs fonctionnels. Il peut être associé à un appel de FB ou à une hiérarchie d'appel de bloc fonctionnel.

d. Fonction (FC) : une fonction est un bloc sans mémoire. Les données sont perdues à l'achèvement de la fonction. Les FC peuvent faire appel à des blocs de données globaux pour la sauvegarde des données.

e. Les SFB et SFC : les blocs fonctionnels système (SFB) et fonction système sont intégrés au système d'exploitation de la CPU-S7 pour créer des messages.

7-3-Bloc des données : Il constitue les blocs suivants :

a. Bloc des données (BD) : Il s'agit d'une zone dans le programme utilisateur qui contient des données utilisateurs.

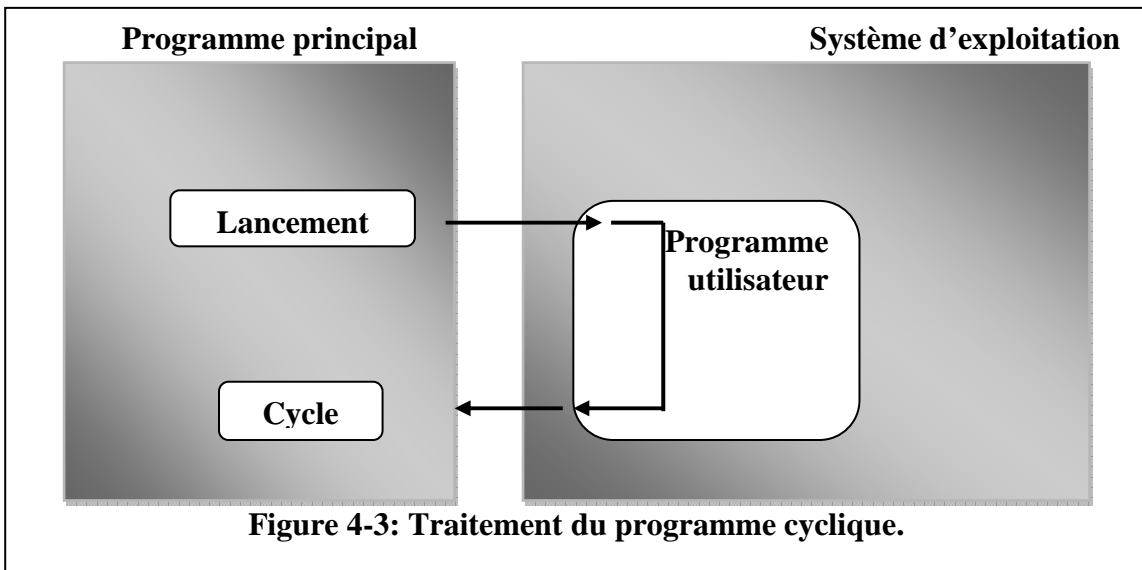
Il y a d'une part les blocs de données globaux auxquels tout bloc de code peut accéder, et d'autre part les blocs de codes de données d'instances. Les blocs de données ne contiennent aucune instruction.

b. Bloc des données système (SDB) : C'est une zone de mémoire dans la CPU contenant des paramètres et des paramètres de bloc.

8-Traitement du programme :

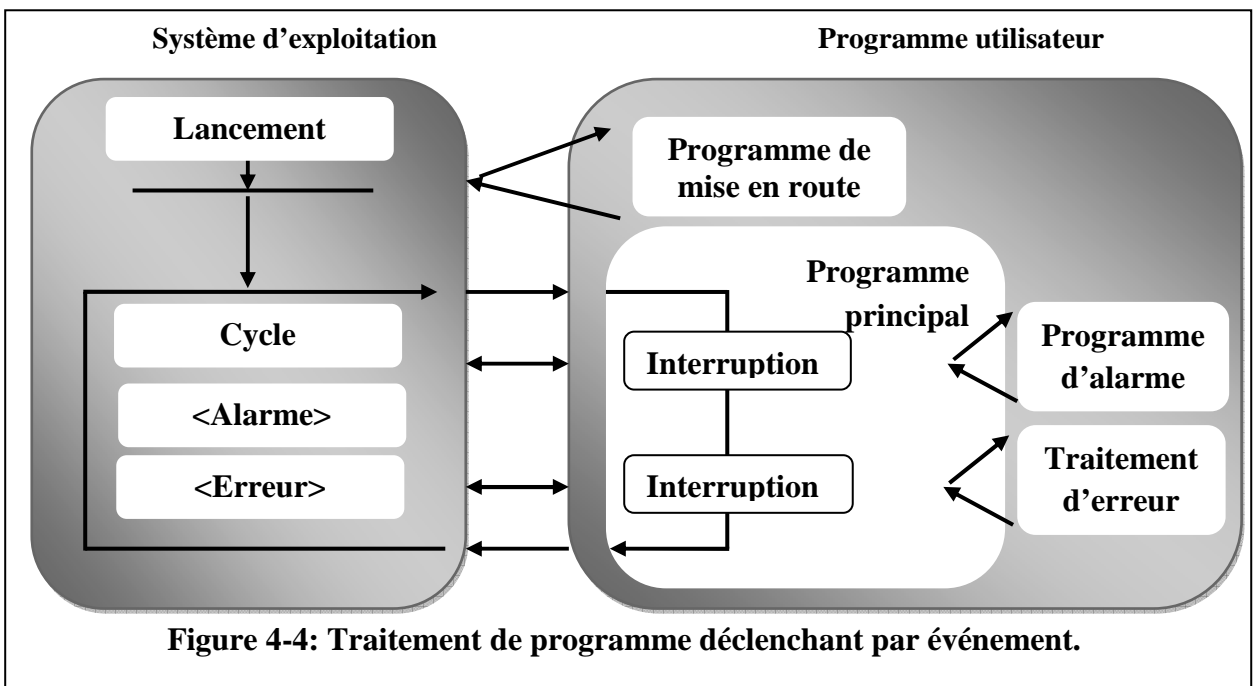
8-1-Traitement du programme cyclique :

Le programme constitue le traitement normal pour les API. Cela signifie que le système d'exploitation parcourt une boucle de programme (le cycle) et appelle le bloc d'organisation OB1 dans le programme principal une fois par boucle. Le programme dans l'OB1 est donc exécuté cycliquement, voir Figure4-3



8-2- Traitement de programme déclenchant par événement :

Le traitement de programme cyclique peut être interrompu par des événements déclencheurs précis, les alarmes () En présence d'un tel événement, le bloc en cours d'exécution est interrompu en limite d'instruction et le bloc d'organisation associé à l'événement déclencheur est traité. Le traitement du programme cyclique reprend ensuite au point d'interruption.



8-3- Programmation linéaire et programmation structurée :

On peut écrire notre programme utilisateur complet dans l'OB1 (programme linéaire), figure 4-5, Cela n'est toutefois recommandé que pour le programme simple. Pour que les automatismes complexes soient mieux traités, on subdivise le programme en parties qui correspondent aux fonctions technologiques pouvant être utilisées plusieurs fois.

Dans le programme utilisateur, ces tâches partielles sont représentées par des parties de programme des blocs de programmation structurées.

Cette structuration offre les avantages suivants :

- ✚ programmation simple et claire.
- ✚ organisation simple du programme.
- ✚ facilite de la modification.

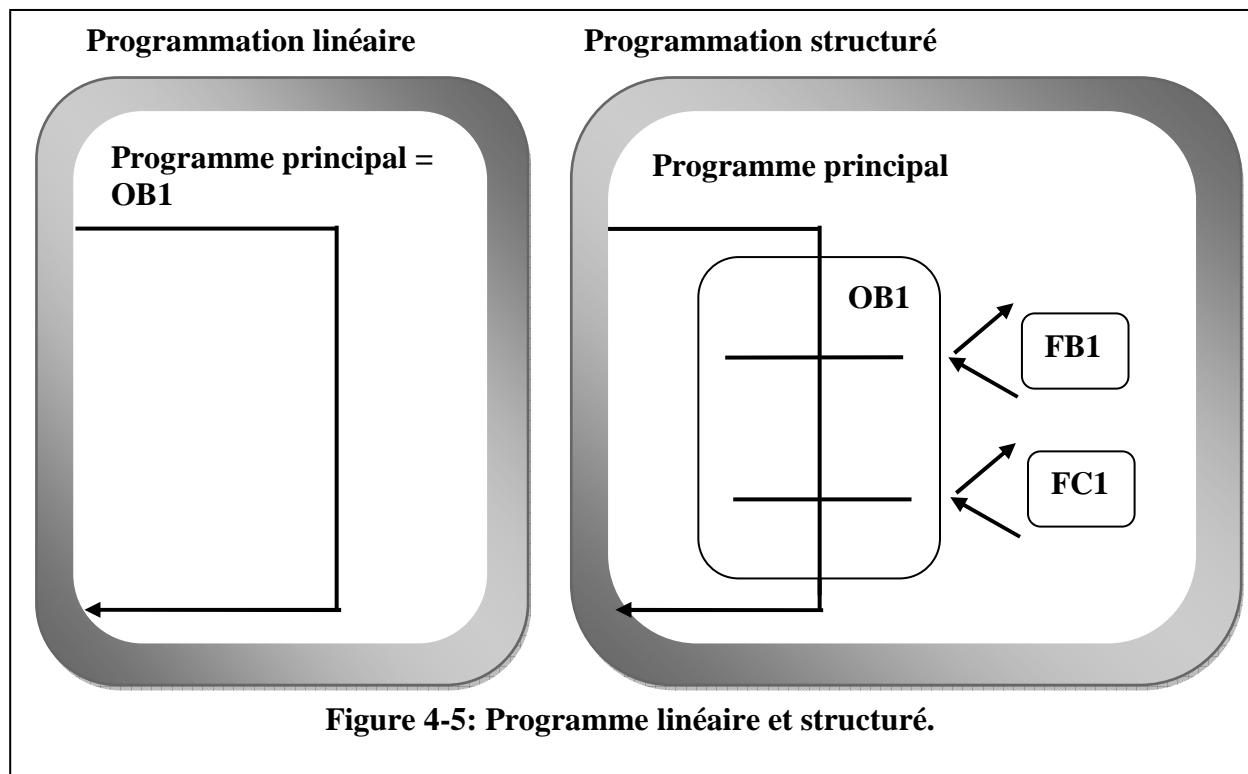


Figure 4-5: Programme linéaire et structuré.

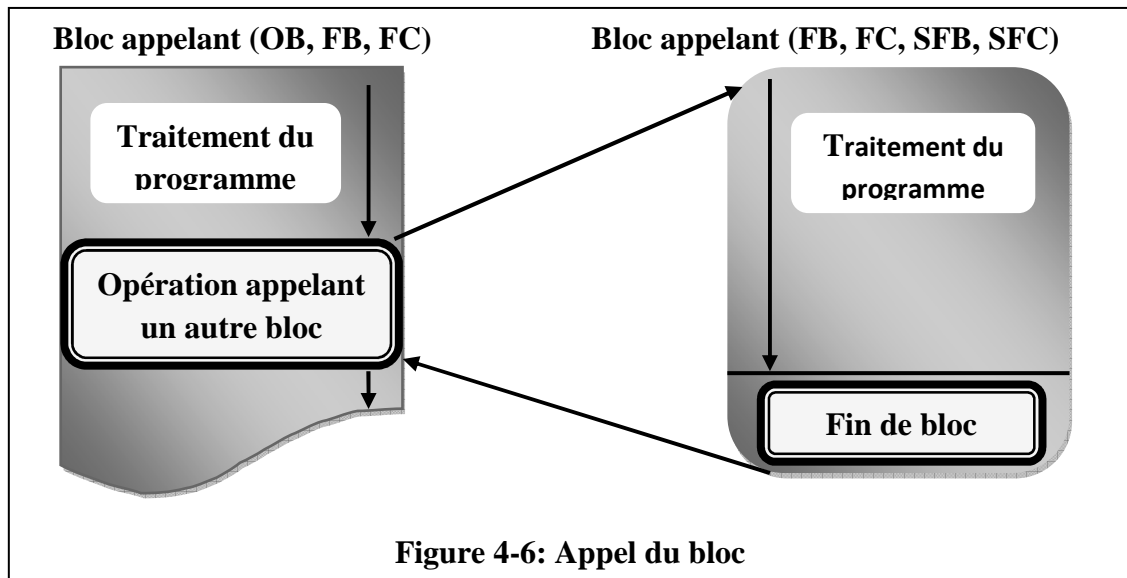
8-4- Appels de blocs :

Le programme utilisateur appelle le deuxième bloc dont les opérations sont alors traitées

dans leurs intégrations.

Une fois le bloc appelé achevé, le traitement se poursuit avec l'opération suivant l'appel de bloc dans le bloc appelant.

Pour l'appel on utilise l'opération « CALL », figure 4-6



a- Appel du FB :

• **Format : CALL FB1, DB2:**

• **Description de l'opération :**

- ✚ cette opération permet l'appel des blocs fonctionnels (FB) que vous avez programmés.
- ✚ si vous appelez un FB à l'aide de CALL, vous devez préciser un bloc de données d'instance. Une fois le bloc appelé traité, le programme se poursuit dans le bloc appelant. Vous pouvez indiquer l'identificateur de bloc sous forme absolue ou symbolique.
- ✚ lors de l'appel de FB, vous devez uniquement indiquer les paramètres effectifs devant changer par rapport au dernier appel, car les paramètres effectifs sont sauvegardés dans le DB d'instance après le traitement du FB. Si le paramètre effectif est un DB, il faut toujours indiquer l'adresse absolue entière.

b. Appel du FC :

• **Format : CALL FC1**

• **Description de l'opération :**

- ✚ cette opération permet l'appel de fonctions (FC). Une fois le bloc appelé traité, le programme se poursuit dans le bloc appelant. Vous pouvez indiquer l'identificateur de bloc sous forme absolue ou symbolique.

- ✚ vous devez, lors de l'appel de FC, affecter des paramètres effectifs du bloc de code appelant aux paramètres formels.

c. Opération nulle « NOP 0 » et « NOP 1 » :

- ✚ NOP 0 (Opération NOP avec l'opérande O).
- ✚ NOP 1 (Opération NOP avec l'opérande 1).

Les deux opérations n'exécutent aucune fonction et n'influent pas sur les bits du mot d'état. Le code d'opération contient un profil binaire de 16 bits. L'opération ne revêt d'importance que pour la console de programmation lorsqu'un programme est affiché.

8-5- Temporisation et comptage :

a- Définition :

Une temporisation est un élément fonctionnel du logiciel de programmation STEP 7 qui met en œuvre et contrôle des séquences temporisées.

Elle permet :

- ✚ d'autoriser des temps d'attente : après une procédure de moulage par injection par exemple, le module doit rester fermé pendant deux secondes. Votre programme assure que deux secondes s'écoulent avant que la pièce ne soit retirée du module.
- ✚ d'autoriser des temps de contrôle : le programme contrôle, par exemple, la vitesse d'un moteur pendant 30 secondes une fois que vous avez appuyé sur le bouton de démarrage.
- ✚ de générer des impulsions : le programme fournit, par exemple, des impulsions qui font clignoter une lampe.
- ✚ de mesurer le temps : votre programme peut par exemple déterminer le temps qu'il faut pour remplir un récipient.

b- Opérations disponibles :

Le langage de programmation LIST du logiciel de programmation STEP 7 met les opérations de temporisation suivantes à votre disposition :

- ✚ démarrer une temporisation :
 - SI** : sous forme d'impulsion.
 - SV** : sous forme d'impulsion prolongée.
 - SE** : sous forme de retard à la montée.
 - SS** : sous forme de retard à la mémoriser.
 - SA** : sous forme de retard à la retombée.
 - R** : remettre à zéro une temporisation.
 - FR** : valider une temporisation.
- ✚ charger une temporisation

L : comme nombre entier.

LC : comme nombre décimal codé binaire.

U, UN, O, ON, X, XN : interroger l'état de signal d'une temporisation.

Voire figure 4-7

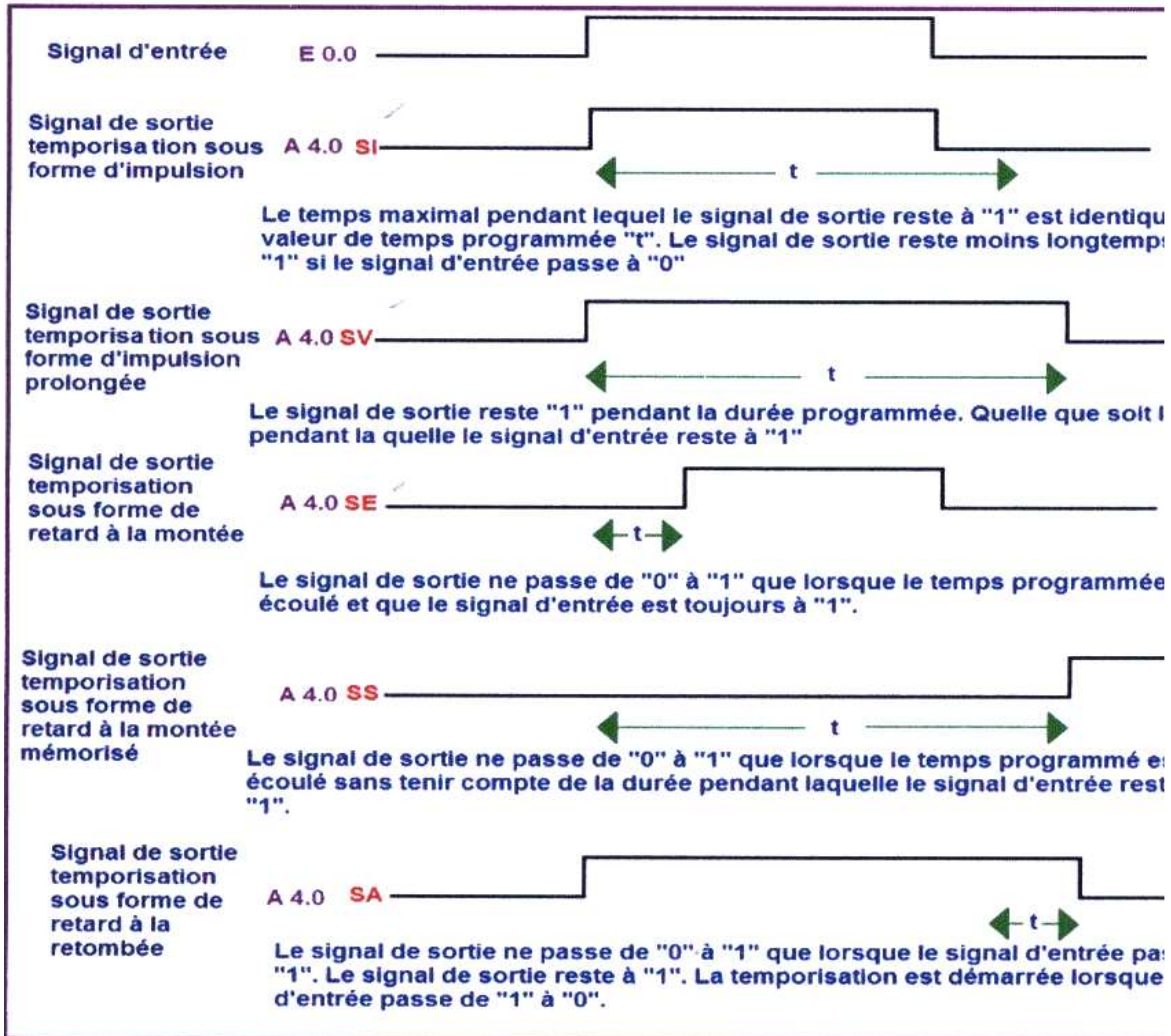


Figure 4-7: Temporisations et comptage.

8-6- Opérations de comptages :

Une zone est réservée au compteur, cette zone de mémoire réserve un mot de 16 bits à chaque compteur. La programmation avec 'LIST' prend en charge 256 compteurs.

a- Les Opérations disponibles :

- Opération « S » (mettre à 1) :

L'opération 'S' permet de mettre à '1' l'état de signal d'un bit. Elle permet également de remettre une temporisation ou un compteur à '1'.

• **Opération « R » (mettre à 0) :**

L'opération 'R' permet de mettre à '0' l'état de signal d'un bit. Elle permet également de remettre une temporisation ou un compteur à '0'.

ZV : incrémentation.

FR : validation un compteur.

ZR : décrémentation.

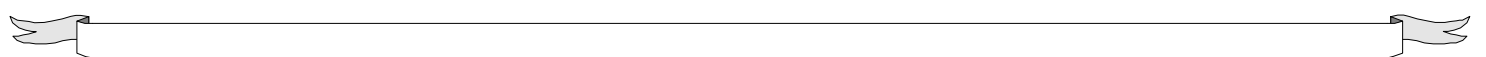
9- Conclusion :

Le STEP7 permet d'écrire un programme pour mettre l'automate en marche. Ce programme est écrit dans un langage spécifique à l'automate.

Il est constitué principalement d'une table mnémonique (voir l'annexe) qui englobe toutes les données et commentaires des entrées de l'automate. Il est constitué aussi des blocs où on doit écrire le programme sous des réseaux.



Chapitre 5:
Chargement et Test du programme



1- Introduction :

Après avoir élaboré le programme de conduite de la machine, nous allons maintenant passer à la phase de son implémentation sur l'automate et à la phase de tests de validité réalisés par simulation sur un PC à l'aide du logiciel de simulation S7-PLCS1M. Ces tests ont porté sur quelques aspects de fonctionnement de la machine.

2- Présentation de l'automate S7-300 de SIEMENS:

C'est l'automate destiné à des tâches d'automatisation moyennes et supérieures. Il peut supporter jusqu'à «512 entrées / sorties » tout ou rien (TOR) et «64 entrées / sorties» analogiques, comme il peut être configuré avec un maximum de «32 modules » de signaux pouvant être répartis sur un châssis de base et trois châssis d'extension.

Le SIMATIC S7-300 est un automate universel avec ce qui suit :

- ✚ Aptitude élevée à l'environnement industriel humide, aux perturbations électromagnétiques et à la résistance élevée aux chocs des secousses.
- ✚ Un système destiné à travailler pour des gammes de température d'étendue allant de «- 25 °C» à «+ 60 °C » et résistant aux plus fortes sollicitations d'humidité.

3- Constitution de l'automate S7-300:

Il se compose essentiellement de:

3-1- Module d'alimentation (PS 307):

La tension de secteur «120 / 230» volts est ramenée à la tension de service nécessaire de l'alimentation de l'API en 24 V.

Cette dernière convient à l'alimentation des circuits internes de l'automate, de même qu'à l'alimentation des circuits de capteurs et des actionneurs.

3-2- Unité centrale (CPU 312 IFM) :

Le S7-300 (CPU 312 IFM) est destinée aux tâches moyennes qui revendiquent un traitement rapide et une périphérie de configuration moyenne. Elle consiste en :

3-2-1- Un processeur :

Pour l'exécution et le codage des instructions STEP7. Il atteint un temps d'exécution de 0.6 µs pour l'instruction sur bit.

3-2-2- Des mémoires :

Elle dispose d'une mémoire de chargement qui garde le programme et une mémoire de travail (de 20 Ko équivalent de 6 Ko d'instructions) pour l'exécution rapide du programme. Elles

offrent une place suffisante pour le stockage des programmes utilisateurs et d'autres mémoires nécessaires pour le stockage des «64 temporisations », «32 compteurs », «1024 mémentos » et des mémoires d'images «d'entrées- sorties ».

- ✚ la sauvegarde du programme en absence de tension est assurée par les cartes mémoires EPROM de capacité maximal 20 ko.
- ✚ le temps d'exécution pour opération sur bit est de «0.6 μ s» à «1.2 μ s », pour une temporisation, «15 μ s » pour un comptage, et « 150 ms » pour chien de garde (temps de cycle).

3-2-3- Les modules d'entrées / sorties TOR (SM-300) :

- ✚ les modules d'entrées TOR(SM321) ramènent le niveau des signaux binaires des matériels au niveau du signal interne de l'automate.
- ✚ les modules de sorties TOR(SM322) convertissent le niveau du signal interne de l'automate au niveau externe requis par le matériel.
- ✚ on n'est pas obligé de respecter les réglages d'emplacement et ce tout en sachant que les adresses des entrées, sont liées à l'emplacement prédéfini par l'utilisateur.

3-2-4- Les modules d'entrées/sorties analogiques:

Ces modules permettent de raccorder à l'automate des capteurs et actionneurs analogiques.

1. **Le module SM 331** : réalise la conversion des signaux analogiques en signaux numériques.
2. **Le module SM 332** : réalise la conversion des signaux numériques en signaux analogiques.
3. **Le module SM 334** : réalise les deux fonctions.

3-2-5- Modules de simulation (SM 374) :

Le module de simulation est un module spécial qui est installé à la demande de l'utilisateur. Dans le S7-300, ce module se monte à la place d'un module d'entrée ou de sortie TOR. Il assure plusieurs fonctions telles que :

- ✚ la simulation des signaux de capteurs aux moyens d'interrupteurs.
- ✚ la signalisation d'état des signaux de sorties par des LED.

3-2-6- Le châssis (rack) :

Les châssis constituent des éléments mécaniques de base du S7-300. Ils remplissent les fonctions suivantes :

- ✚ assemblage mécanique des modules.

- ✚ distribution de la tension d'alimentation des modules.
- ✚ acheminement du bus de fond de panier aux différents modules.

3-2-7- Pupitre opérateur :

Le pupitre opérateur offre des actions prioritaires, il met à l'aise l'opérateur pour intervenir à n'importe quel moment opportun. C'est la commande forcée par l'opérateur.

3-2-8- Les modules enfichables :

Ils se composent de:

1. des coupleurs (IM)/S7-300 :

Ils sont utilisés en cas de besoin. Il y a plus de 8 modules de signaux de communication pour réaliser une application d'automatisation. Ils sont utilisés pour interconnecter le châssis de base (ZG) et le châssis d'extension (EG) dans la configuration à plusieurs rangées.

2. console de programmation (PG 700) :

C'est une console à usage multifonctions pour toutes les tâches de conception et de programmation. Son utilisation est possible en ligne et ou hors ligne pour les automates. Sa grande puissance, sa flexibilité de configuration et sa riche dotation fonctionnelle en font un outil idéal pour toutes les applications d'automatisation.

4- Chargement et test du programme :

4-1- Etablir la liaison en ligne :

Le montage suivant montre comment charger le programme via une console de programmation PG, dans le système d'automatisation (AP) et ce pour le tester. Voir Figure (4.1)

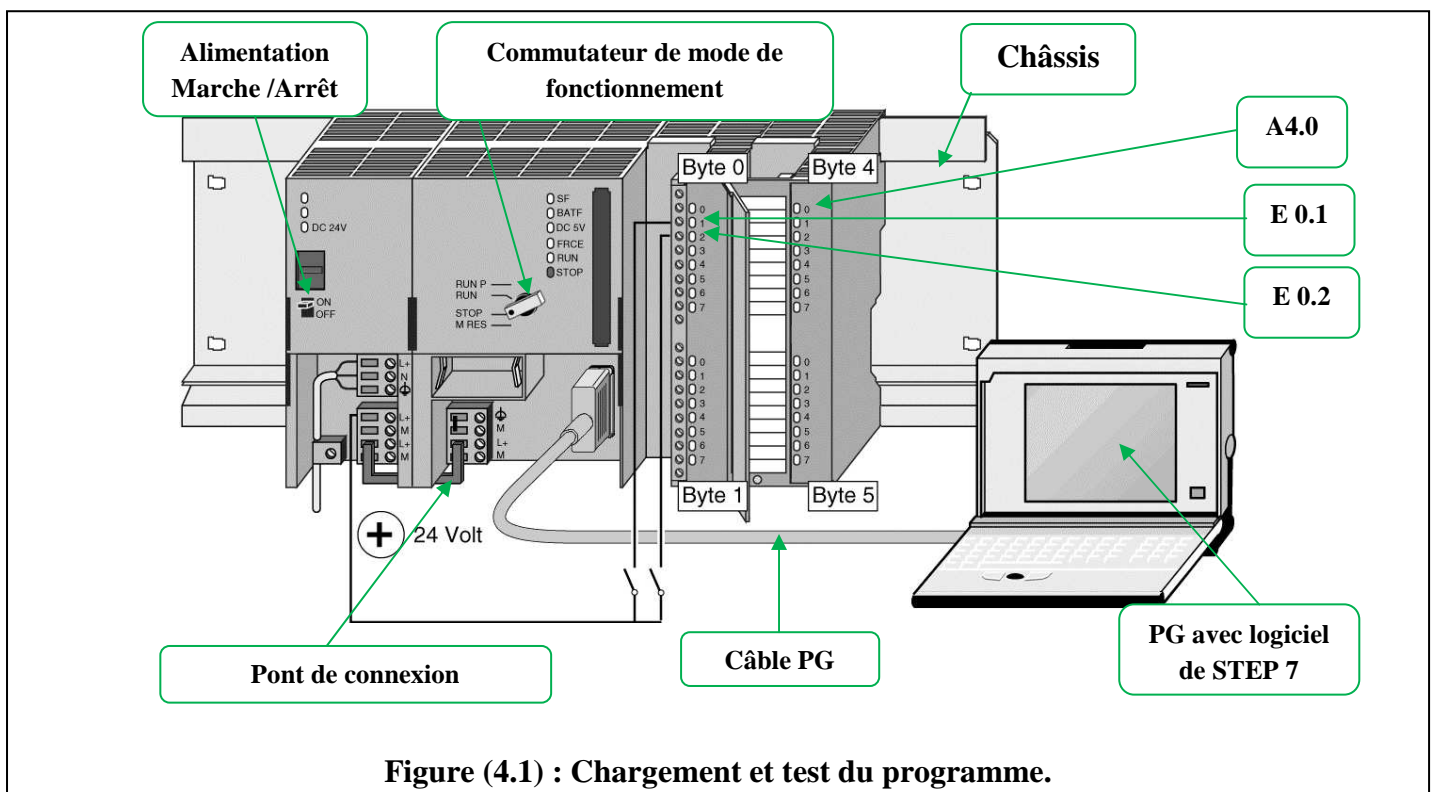
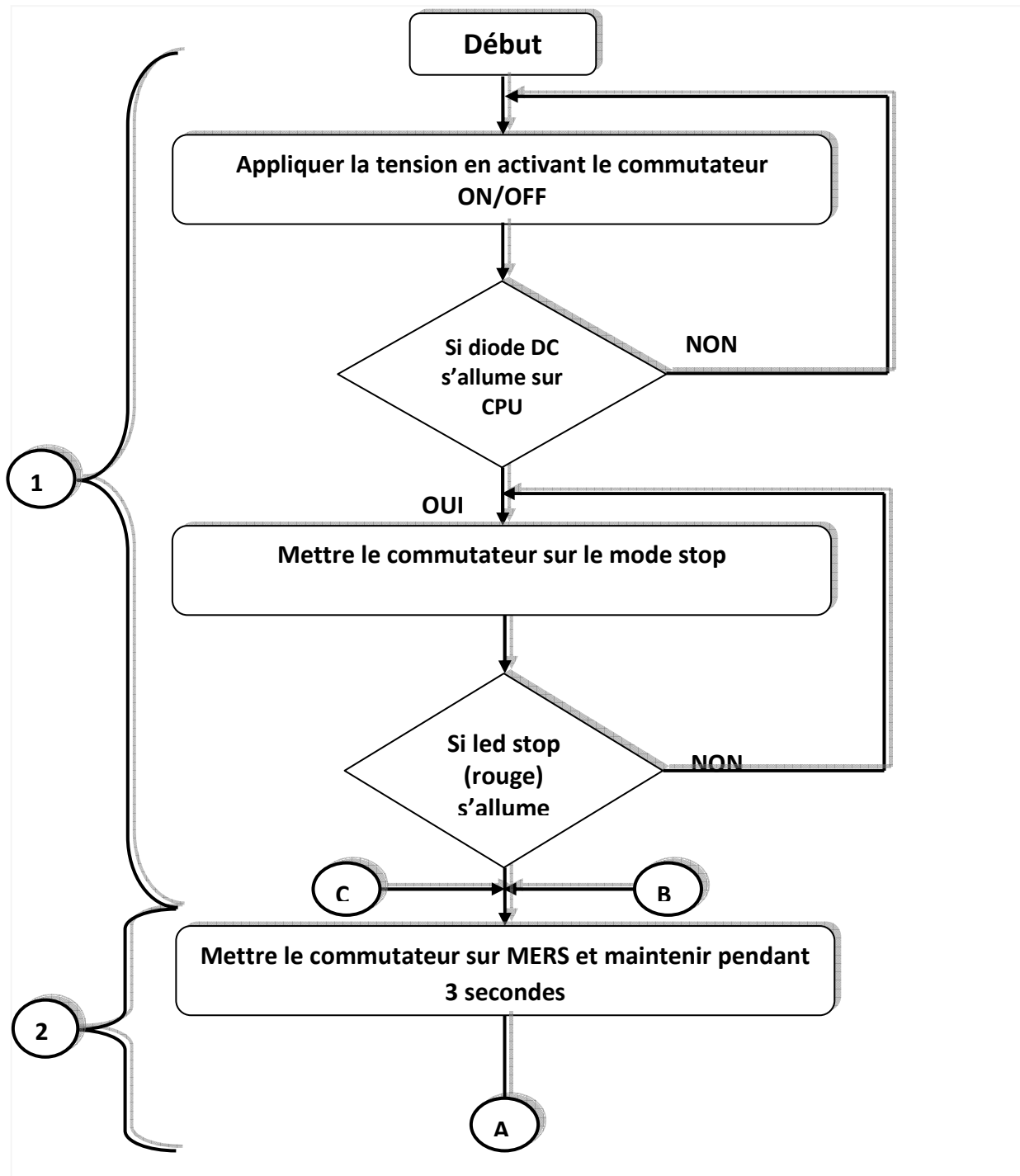


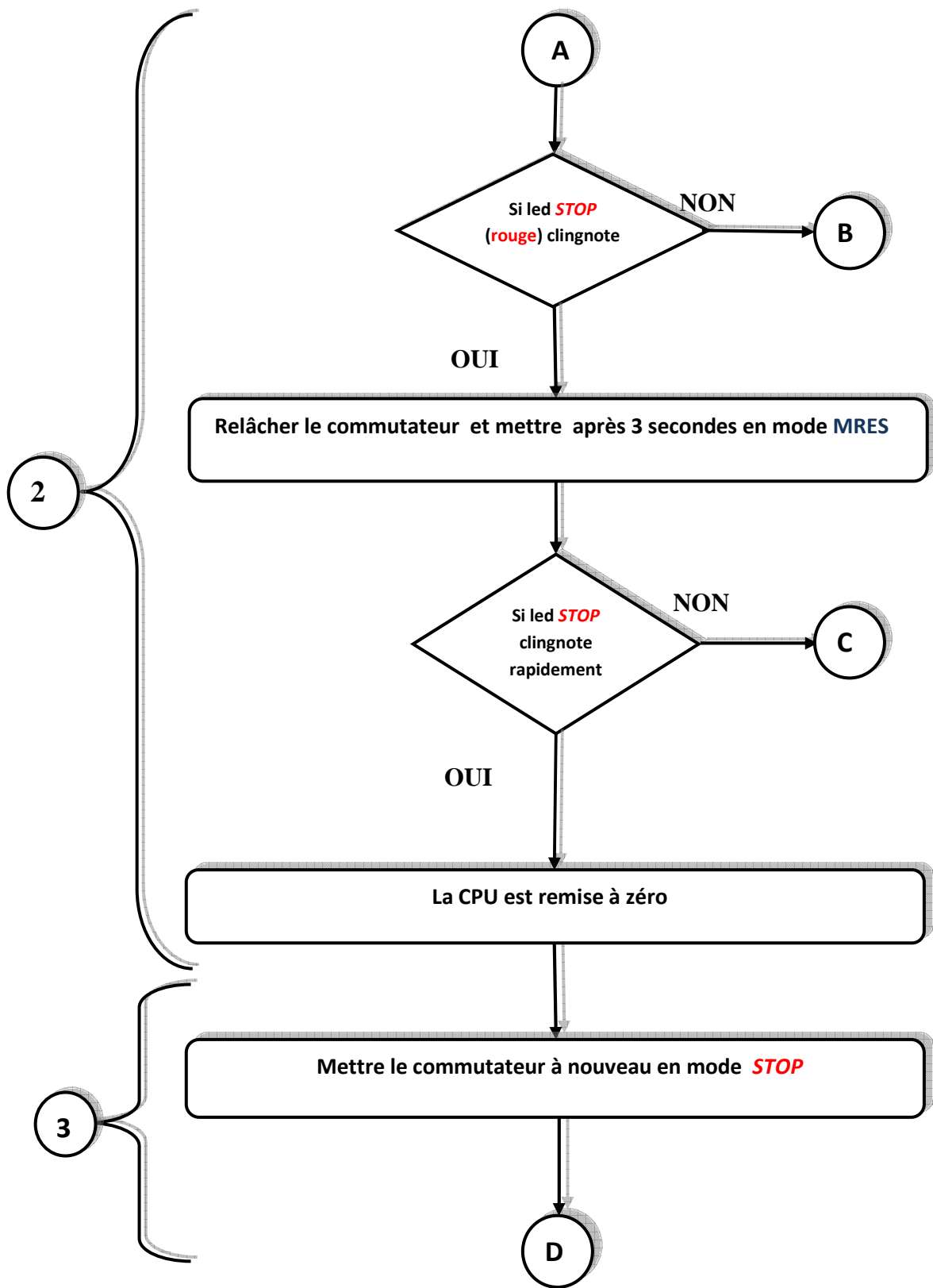
Figure (4.1) : Chargement et test du programme.

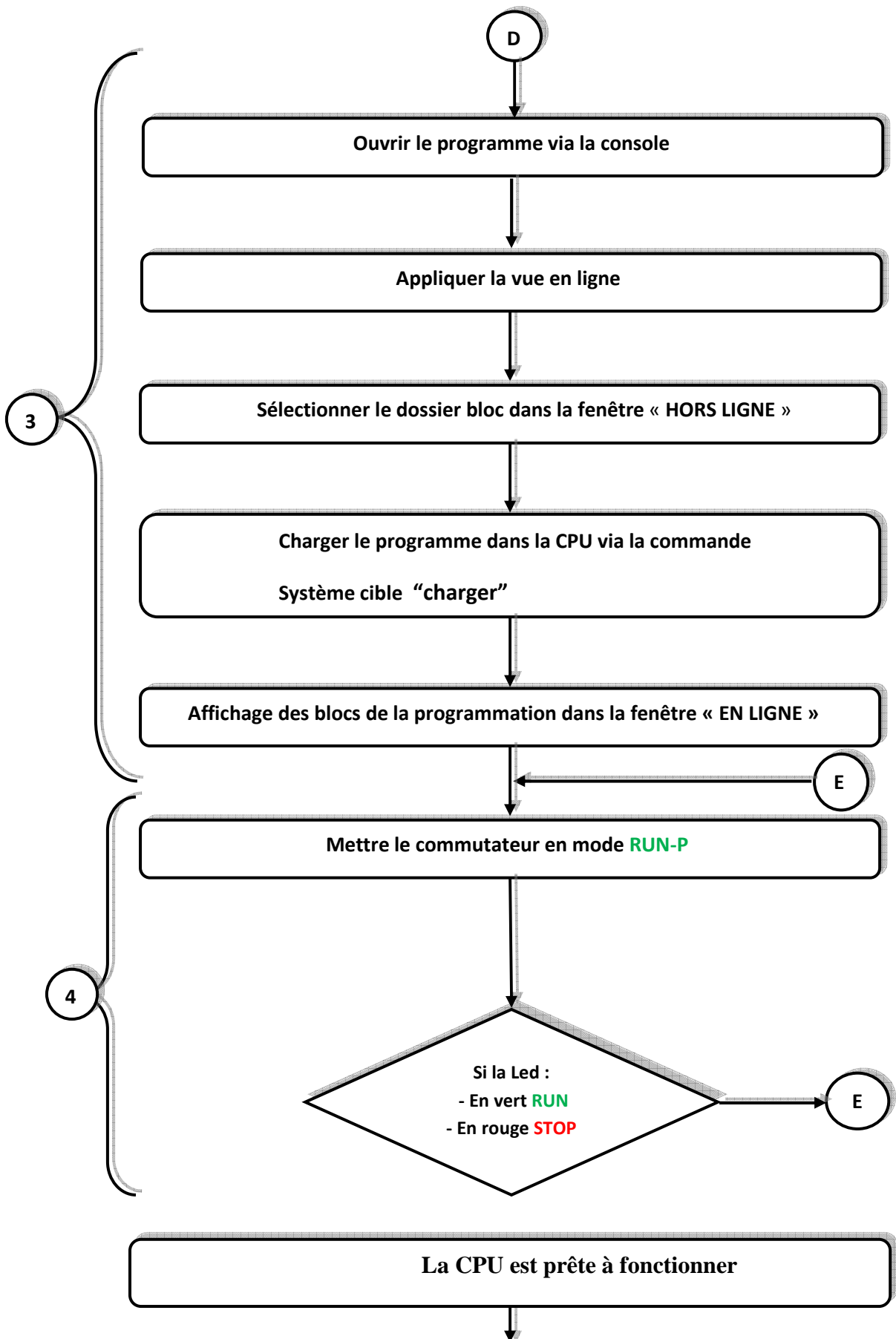
4-2- Chargement du programme :

Le chargement du programme dans le système automatisé comporte quatre étapes, à savoir:

- ✚ application de la tension.
- ✚ effacement général de la CPU.
- ✚ chargement de programme.
- ✚ mise en marche de la CPU.







- 1: Application de la fonction.**
- 2: Effacement général de la CPU.**
- 3: Chargement du programme.**
- 4: Mettre la CPU en marche.**

5- Le logiciel S7-PLCSIM- Simulateur des modules :

5-1- Introduction générale sur le S7-PLCSIM :

L'application de simulation de modules S7-PLCSIM nous permet d'exécuter et de tester notre programme dans un automate programmable (API) que nous simulons dans un ordinateur ou dans la console de programmation.

5-2- Etats de fonctionnement de la CPU :

5-2-1- Etat de marche (RUN-P) :

La CPU exécute le programme tout en nous permettant de le modifier, de même que ses paramètres. Afin de pouvoir utiliser les applications de STEP 7 pour forcer un paramètre quelconque du programme durant son exécution, nous devons mettre la CPU à l'état RUN-P.

Nous avons toujours la possibilité d'utiliser les "fenêtres" créées dans l'application de simulation de modules S7-PLCSIM pour modifier une donnée quelconque utilisée par le programme.

5-2-2- Etat de marche (RUN) :

La CPU exécute le programme en lisant les entrées, exécutant le programme, puis en actualisant les sorties. Lorsque la CPU se trouve à l'état de marche (RUN), nous ne pouvons charger aucun programme, ni utiliser les applications de STEP 7 pour forcer un paramètre quelconque (comme les valeurs d'entrée). Nous pouvons utiliser uniquement les fenêtres créées dans l'application de la simulation des modules S7-PLCSIM, pour modifier une donnée quelconque utilisée par le programme.

5-2-3- Etat d'arrêt (STOP) :

La CPU n'exécute pas le programme. Contrairement à l'état d'arrêt (STOP) de CPU réel, les sorties ne prennent pas de valeurs (de "sécurité") prédéfinies, mais conservent l'état dans lequel elles étaient lorsque la CPU est passée à l'état d'arrêt (STOP).

Nous pouvons charger des programmes dans la CPU lorsqu'elle est à l'arrêt. Le passage de l'état d'arrêt (STOP) à celui de marche (RUN) démarre l'exécution du programme à partir de la première opération.

Les états de fonctionnement de la CPU, les indicateurs de la CPU et le bouton MRES (effacement général) sont tous représentés dans la fenêtre "CPU". Nous pouvons définir l'état de fonctionnement de la CPU avec la commande position du commutateur à clé. Nous pouvons

suspendre l'exécution du programme de l'API de simulation lorsque la CPU est à l'état RUN ou RUN-P.

5-2-4- Effacement général de la mémoire de la CPU :

Pour effacer la mémoire de la CPU de simulation, sélectionnez-la commande « CPU » « effacement général » ou cliquez sur le bouton MRES dans la fenêtre "CPU". Les zones de mémoire sont alors réinitialisées et les blocs de code ainsi que la configuration matérielle de l'API de simulation sont effacés.

La CPU passe automatiquement à l'état d'arrêt (STOP) lorsque nous effectuons un effacement général.

5-3- Indicateurs de la CPU :

La fenêtre CPU dispose d'une série d'indicateurs qui correspondent aux voyants de signalisation sur une CPU réelle:

- ✚ **SF** (erreur système) nous averti que la CPU a détecté une erreur système, entraînant un changement d'état de fonctionnement.
- ✚ **DP** (périphérie décentralisée ou E/S éloignées) indique l'état de la communication avec les entrées/sorties décentralisées (éloignées).
- ✚ **DC** (alimentation) indique si la CPU se trouve sous ou hors tension.
- ✚ **RUN** indique que la CPU se trouve à l'état de marche.
- ✚ **STOP** indique que la CPU se trouve à l'état d'arrêt.

5-4- Sélection des options de mode d'exécution :

L'application de simulation de modules S7-PLCSIM met à votre disposition des options d'exécution du programme de simulation.

5-4-1- Cycle unique :

La CPU exécute un cycle, puis attend que nous lui indiquions d'exécuter le cycle suivant. A chaque cycle, la CPU lit la périphérie d'entrée (PE). Elle exécute le programme puis inscrit les résultats dans la périphérie de sortie (PA). Elle attend ensuite que nous lui demandions d'exécuter le cycle suivant avec la commande Exécution> Cycle suivant.

5-4-2- Cycle continu :

La CPU exécute un cycle complet puis en démarre un autre. A chaque cycle, la CPU lit la périphérie d'entrée (PE), exécute le programme puis inscrit les résultats dans la périphérie de sortie (PA).

6- Test du programme :**6-1- Test avec la fonction de visualisation :**

La fonction de visualisation permet de tester le bloc d'un programme pour cela.

- ✚ une liaison en ligne devra être établie avec la CPU.
- ✚ la CPU doit être en mode de fonctionnement.
- ✚ le programme devait être chargé dans la CPU.
- ✚ il faut ouvrir le bloc OB1 dans la fenêtre en ligne de projet.
- ✚ il faut activer la fonction TEST <VISUALISER.

6-2- Test avec la table des variables :

La marche à suivre pour la création de la table des variables sont :

- ✚ on teste les variables du programme en les visualisant et en les forçant à partir de la table des variables, il faut pour cela:
 - ✚ que la liaison en ligne à la CPU existe.
 - ✚ que la CPU se trouve en mode de fonctionnement.
 - ✚ que le programme soit chargé.

6-3- Test avec un programme de simulation (logiciel optionnel S7-PLCS1M) :

Une des fonctions les plus importantes, est la possibilité de simuler le programme sur PC sans être relié à aucun système cible et donc d'effectuer la mise au point du programme sans être sur site.

7- Conclusion :

Le STEP7 permet une simulation complète au sein du logiciel, il n'est pas nécessaire qu'on soit connecté à un matériel S7 quelconque (CPU ou modules de signaux). Le test du programme permet de remédier à d'éventuelles erreurs.



Conclusion Générale



Conclusion générale

L'évolution remarquable de la technologie cautionne l'amélioration des méthodes de résolution des problèmes d'automatisations séquentiels, en utilisant moins de composants, dans le but d'améliorer la production et d'augmenter la sécurité.

Dans le cadre du développement de mon mémoire, il m'a été demandé de moderniser et réaliser la commande séquentielle par automate programmable de l'aléuseuse « SCEMM » à l'aide d'un API à la société nationale SNVI (zone industrielle ROUIBA), En guise de : remplacement de la commande en logique câblée existante, dont les aléas sont mentionnés précédemment, mon option c'est portée dans le choix de l'outil de l'automatisation, à savoir l'API, ceci permet d'avoir un produit fiable et dont l'application ne nécessite pas un très grand investissement.

Pour atteindre cet objectif, j'ai opté pour un automate "S7-300" de SIMENS. Cet automate peut être programmé à partir d'un micro-ordinateur, en installant dans celui-ci le logiciel STEP7. Par la suite il faut transférer le programme dans l'automate et/ou à l'aide d'une console de programmation à des fins de son exécution.

Du fait je n'ai pas eu les moyens de tester l'installation avec un automate réel, j'ai utilisé une autre méthode pour le test du programme par le logiciel S7-PLCSIM. Il m'a été permis de faire une simulation, pour assurer le bon fonctionnement du programme.

Par suite de la simulation du programme, je considère que je suis arrivée à l'objectif que je me suis fixée dans l'introduction générale de cette étude.

Ce projet m'a permis de me familiariser avec les systèmes automatisés et d'approfondir mes connaissances des API et à compléter mes connaissances théoriques par le contact avec le milieu industriel.

Je souhaite, que les promotions a venir vont compléter les parties : conception d'un bras pour le chargement et déchargement de la pièce, et remplacement de la boite à vitesse existante par un variateur de vitesse.

Cependant, j'espère que mon travail verra sa concrétisation sur le plan pratique et que mon mémoire servira pour tous ceux qui s'intéressent au contrôle et à la commande des installations industrielles par un automate programmable SIMATIC S7.



Bibliographie

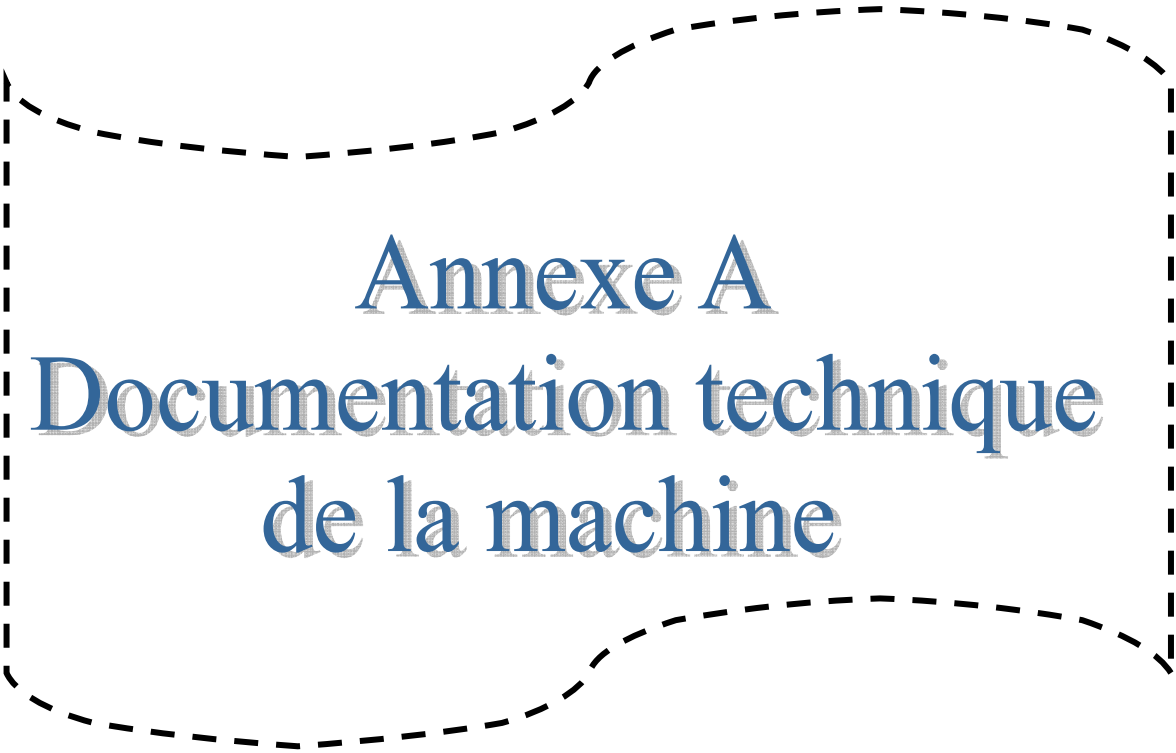


- [1] Documentation interne de l'usine, « documentation technique de la machine »
- [2] S.MORENO, E. PEULOT, initialisation au grafcet, Casteilla, paris, 1999
- [3] S.MORENO, E. PEULOT, le grafcet, conception-implantation dans les automates programmables industriels, Casteilla, paris, 1999
- [4] Documentation SIEMENS, « **Configuration matérielle dans STEP 7** », édition 01/2004.
- [5] Documentation SIEMENS, « **Programmation avec STEP 7** », édition 01/2004.
- [6] Site internet : www.siemens.com
- [7] site internet : www.Mesure et détection des niveaux.com

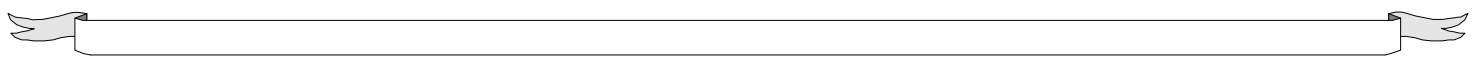


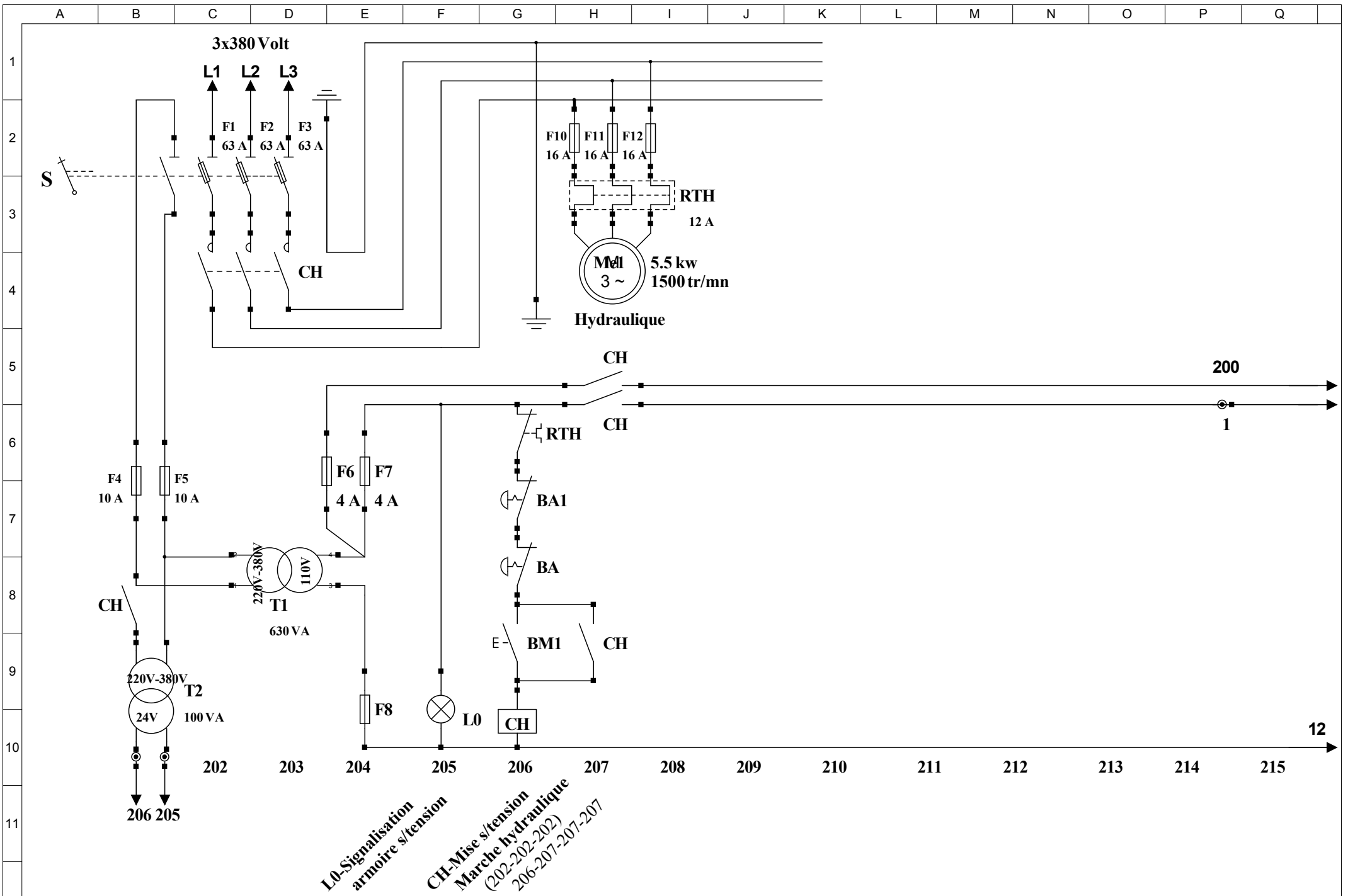
Annexes

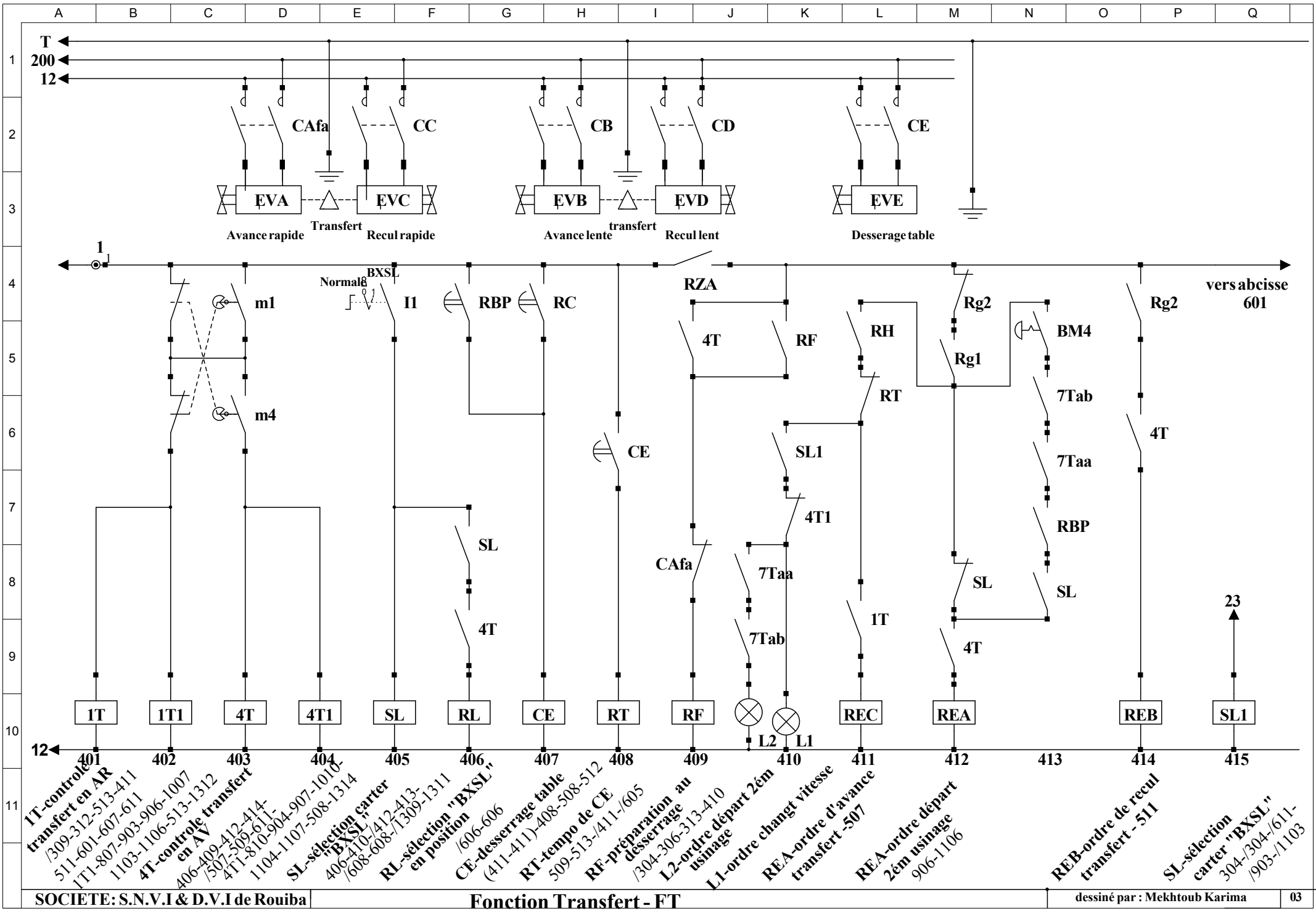


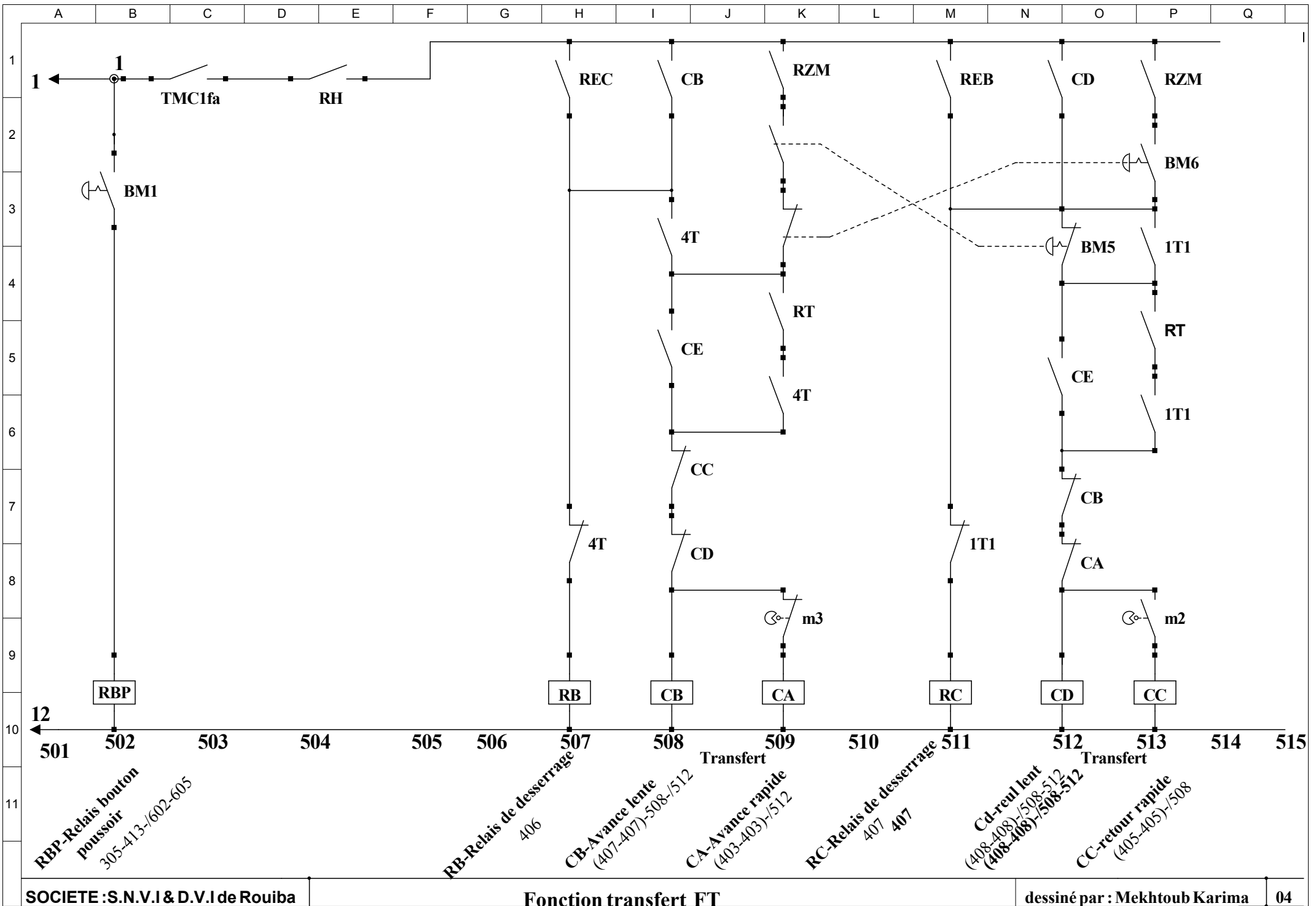


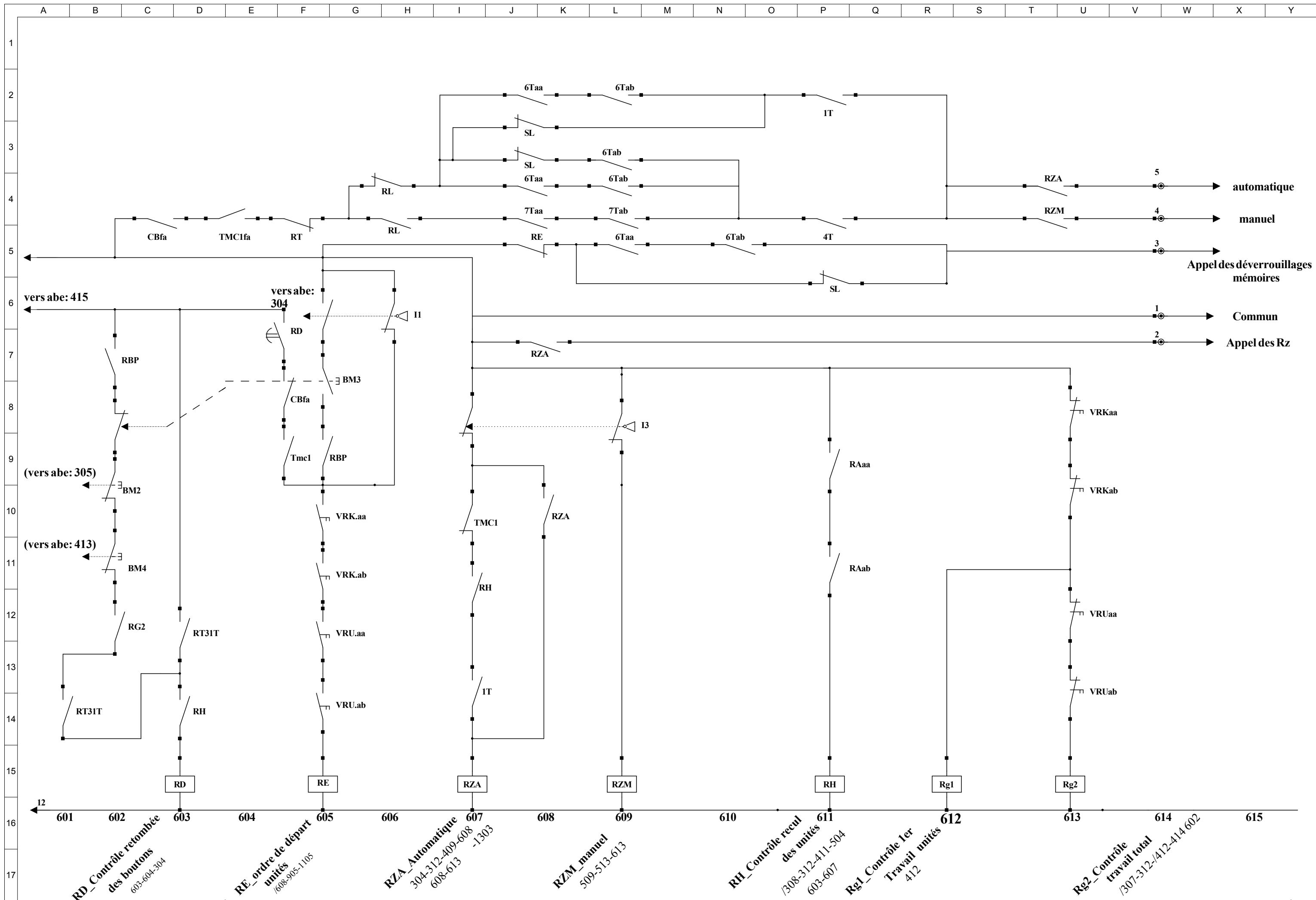
Annexe A
Documentation technique
de la machine

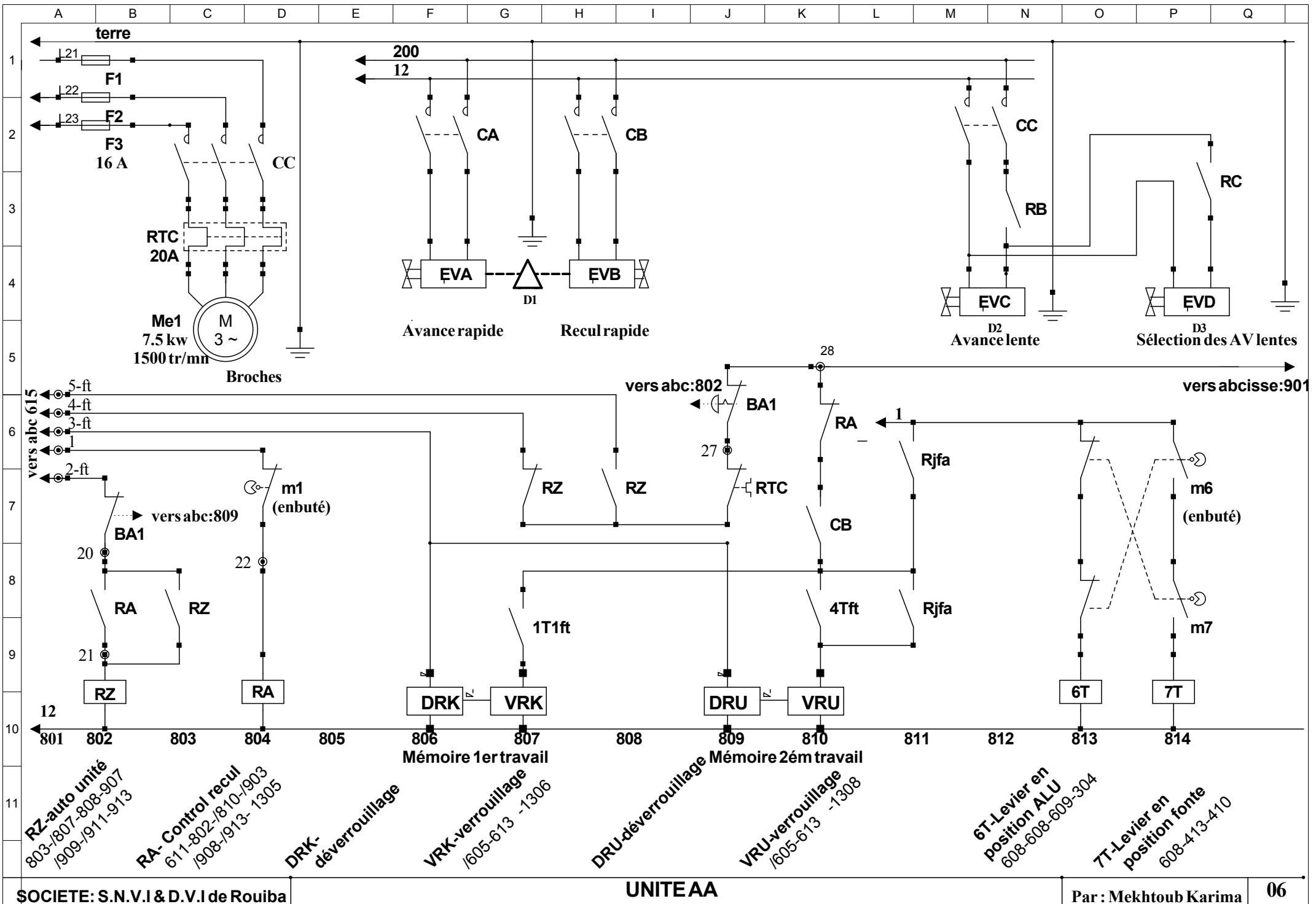


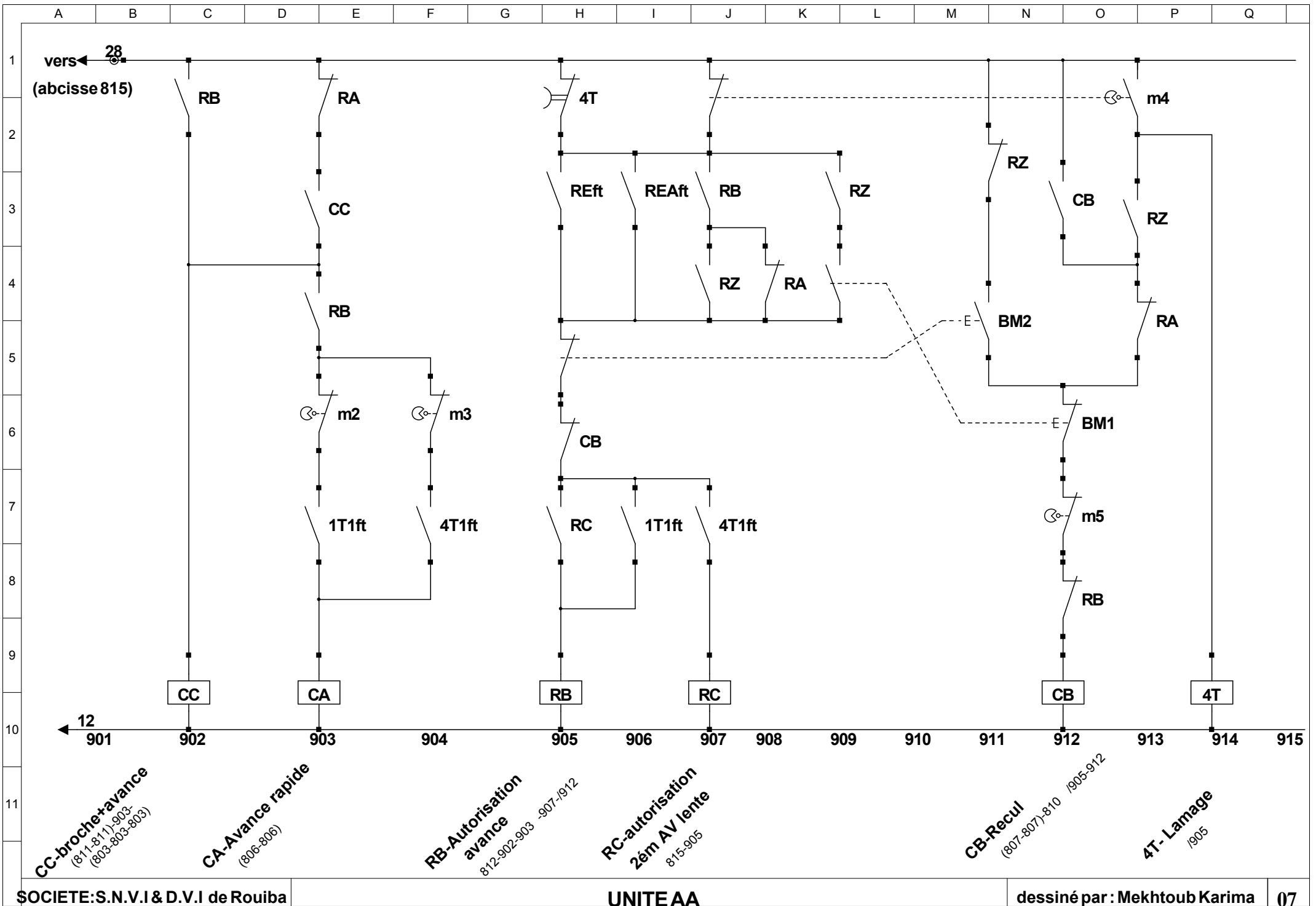


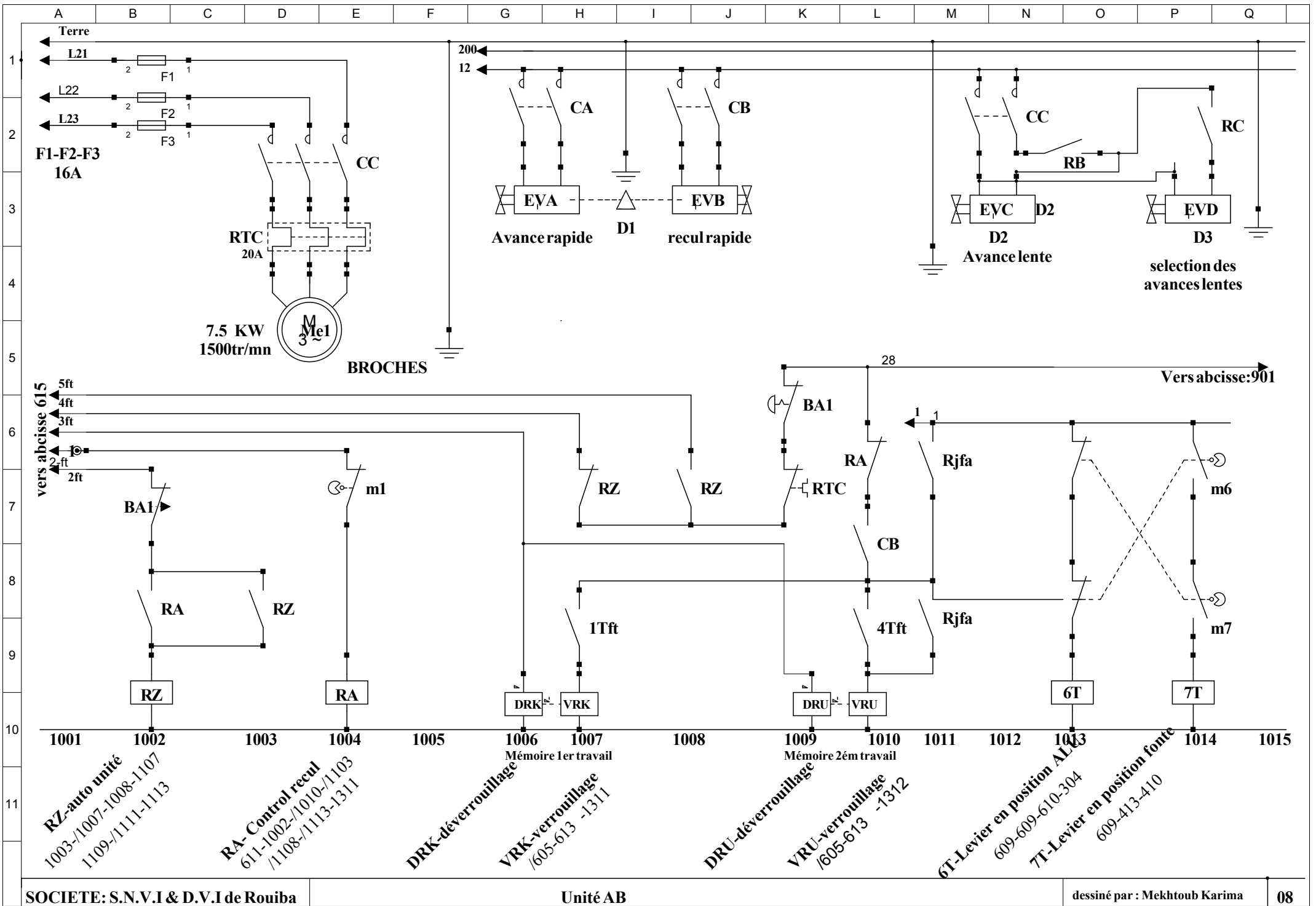


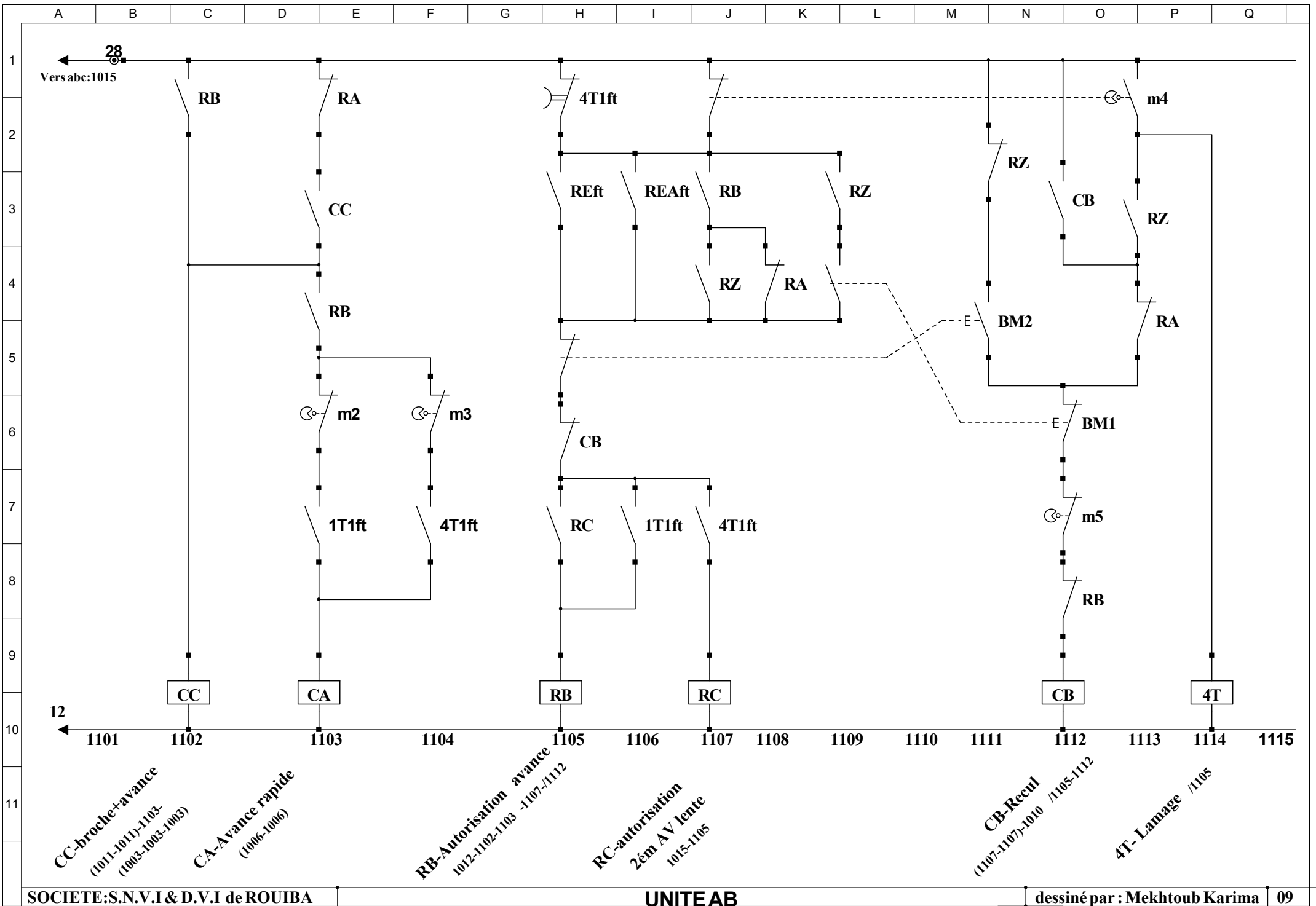


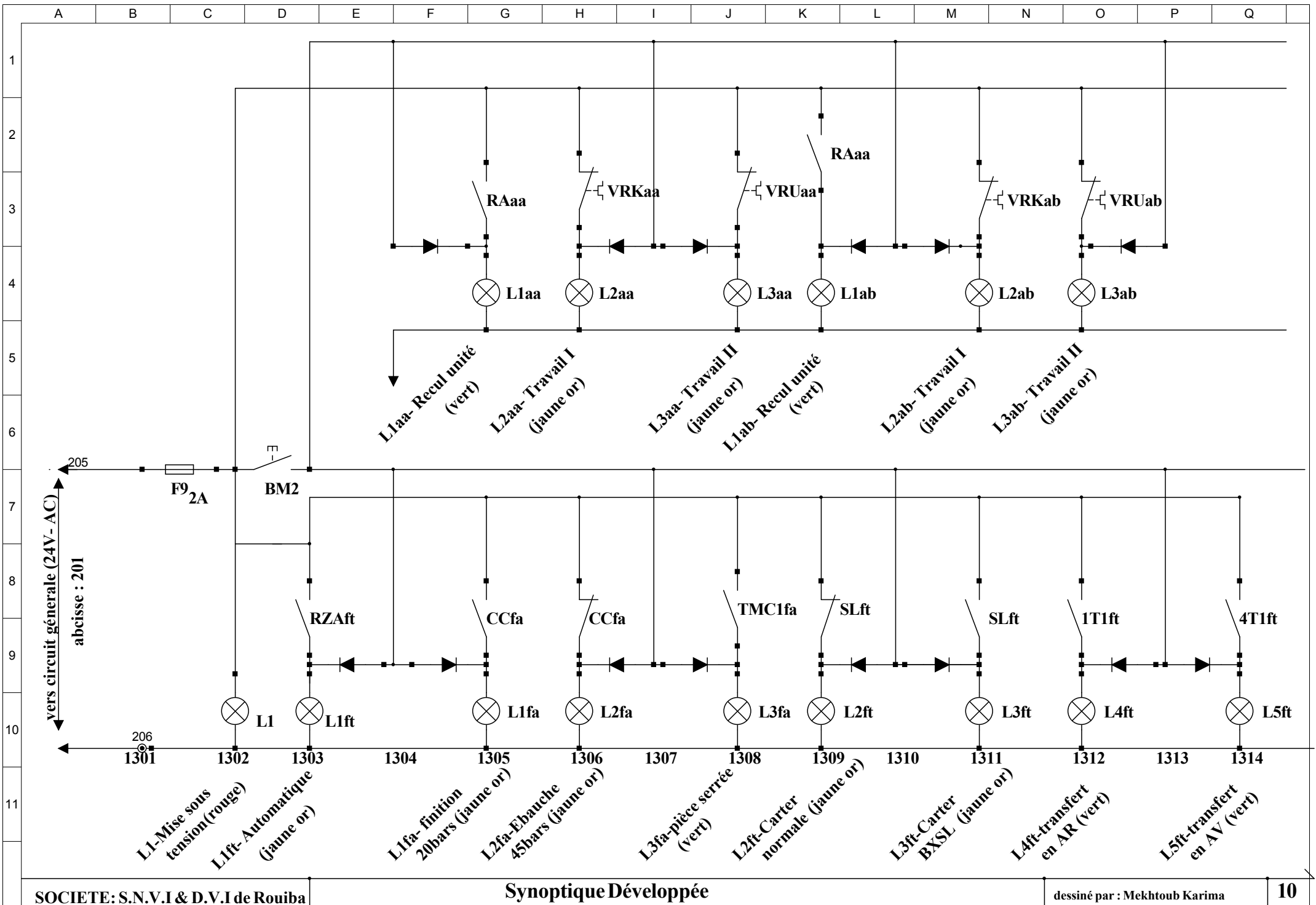


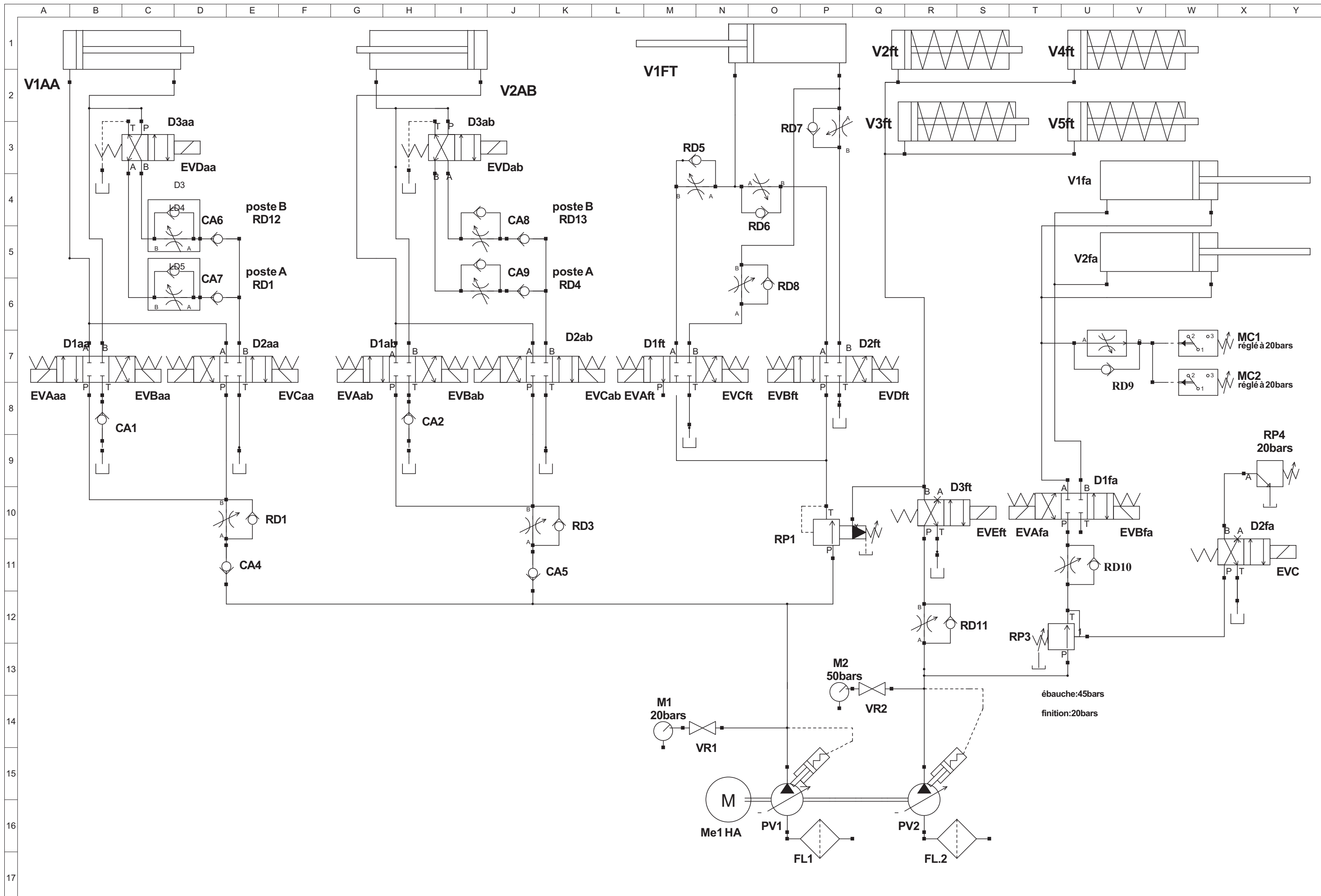


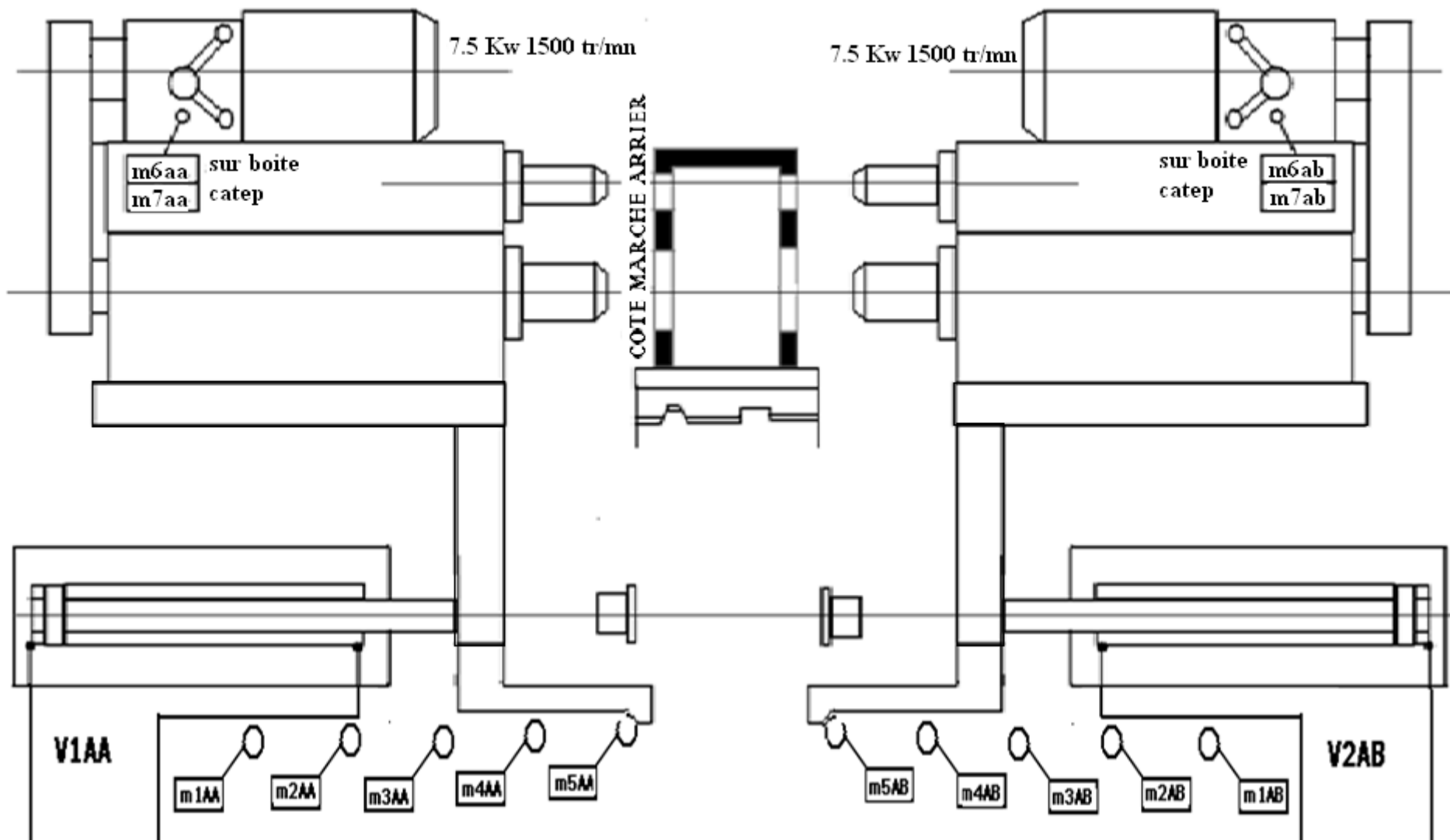




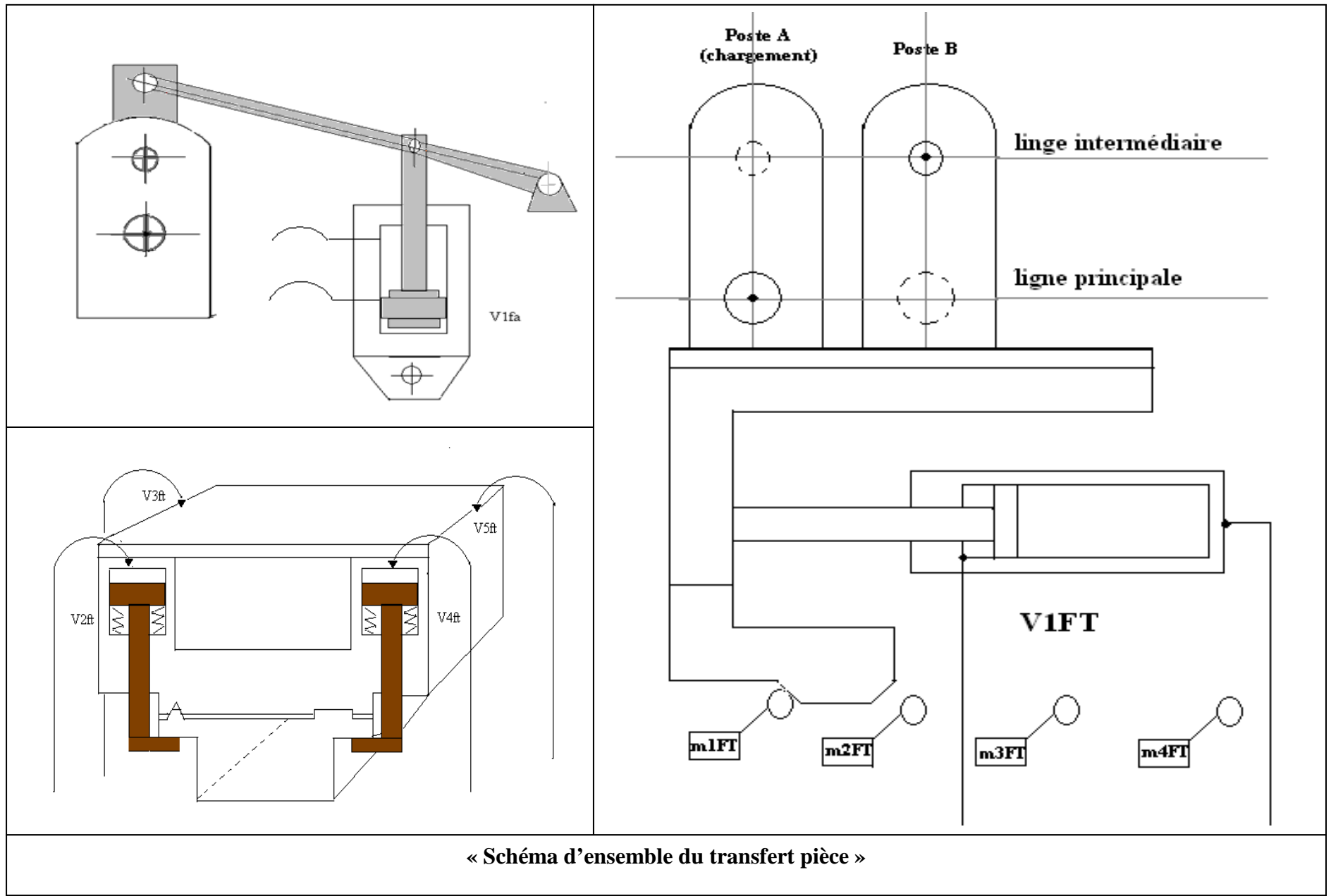


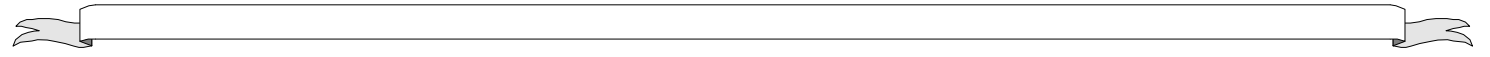






« Schéma d'ensemble de la machine »





Annexe B
Programme de la machine



Propriétés de la table des mnémoniques

Nom : Mnémoniques
Auteur :
Commentaire :
Date de création : 02/10/2009 17:21:36
Dernière modification : 20/10/2009 11:05:58
Dernier filtre sélectionné : Mnémoniques univoques
Nombre de mnémoniques : 1/177
Dernier tri : Mnémonique ordre croissant

Etat	Mnémonique	Opérande	Type de données	Commentaire
	cycle execution	OB 1	OB 1	

OB1 - <offline>

"cycle execution"

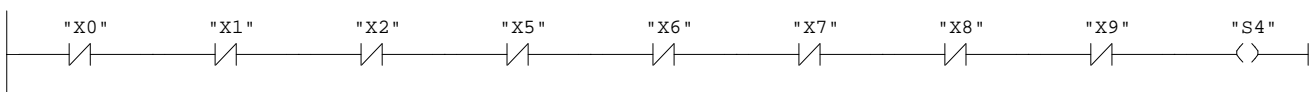
Nom :
Auteur :
Horodatage Code :
Interface :
Longueur (bloc/code /données locales) : 01960 01528 00020

Famille :
Version : 0.1
Version de bloc : 2

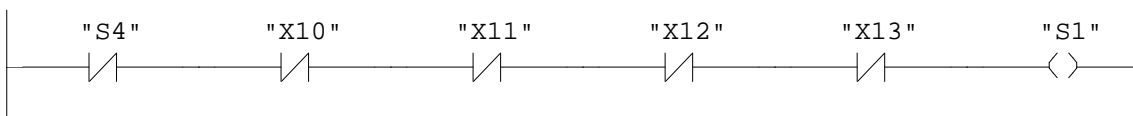
Nom	Type de données	Adresse	Commentaire
TEMP		0.0	
OB1_EV_CLASS	Byte	0.0	Bits 0-3 = 1 (Coming event), Bits 4-7 = 1 (Event class 1)
OB1_SCAN_1	Byte	1.0	1 (Cold restart scan 1 of OB 1), 3 (Scan 2-n of OB 1)
OB1_PRIORITY	Byte	2.0	Priority of OB Execution
OB1_OB_NUMBR	Byte	3.0	1 (Organization block 1, OB1)
OB1_RESERVED_1	Byte	4.0	Reserved for system
OB1_RESERVED_2	Byte	5.0	Reserved for system
OB1_PREV_CYCLE	Int	6.0	Cycle time of previous OB1 scan (milliseconds)
OB1_MIN_CYCLE	Int	8.0	Minimum cycle time of OB1 (milliseconds)
OB1_MAX_CYCLE	Int	10.0	Maximum cycle time of OB1 (milliseconds)
OB1_DATE_TIME	Date_And_Time	12.0	Date and time OB1 started

Bloc : OB1 "Main Program Sweep (Cycle)"

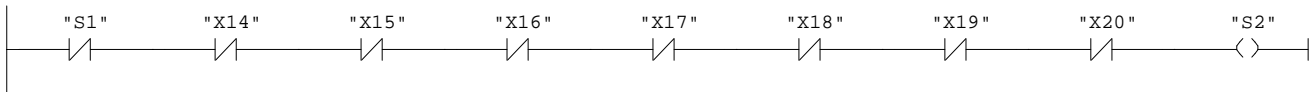
Réseau : 1



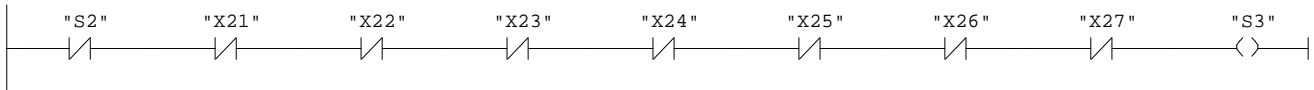
Réseau : 2



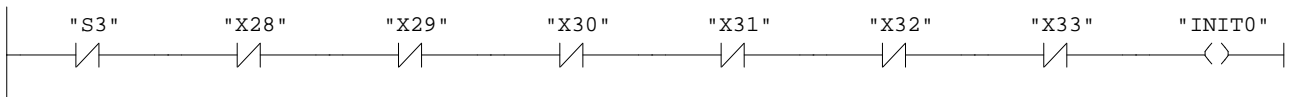
Réseau : 3



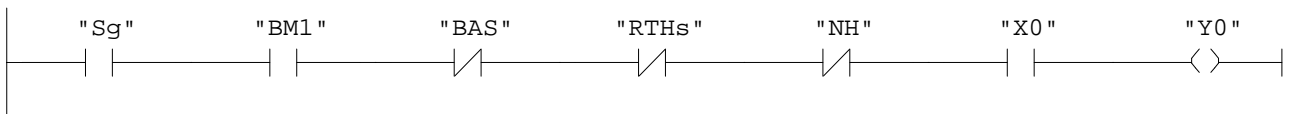
Réseau : 4



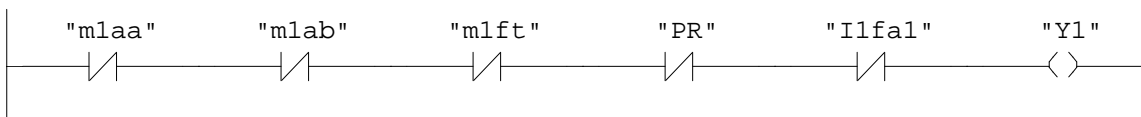
Réseau : 5 Initiation de l'étape 0



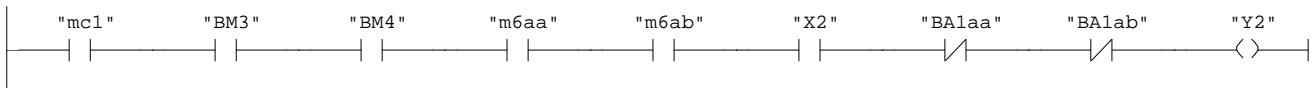
Réseau : 6 Transition "Y0"



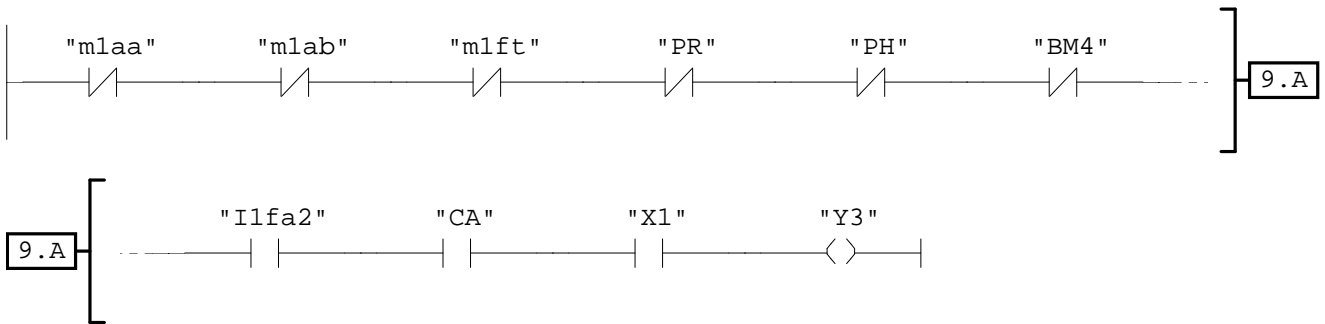
Réseau : 7 transition "Y1"



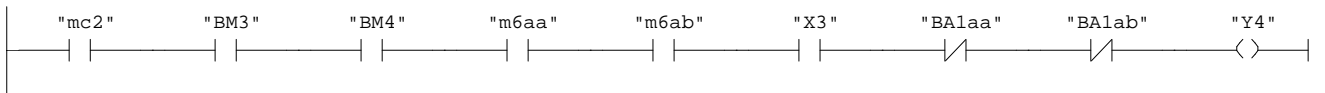
Réseau : 8 transition "Y2"



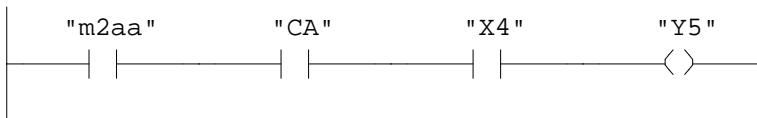
Réseau : 9 transition "Y3"



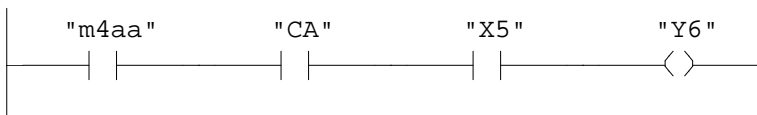
Réseau : 10 transition "Y4"



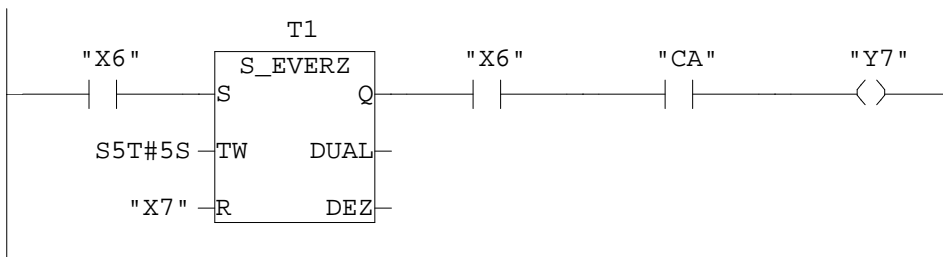
Réseau : 11 transition "Y5"



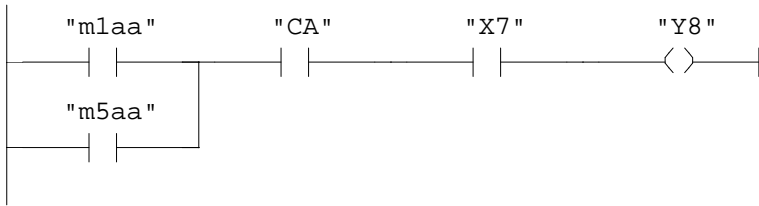
Réseau : 12 transition "Y6"



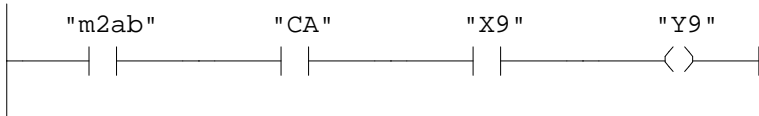
Réseau : 13 Transition "Y7"



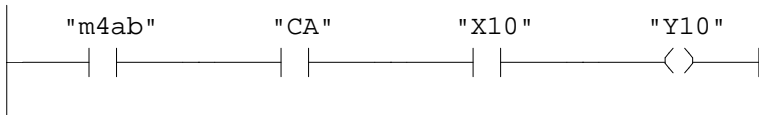
Réseau : 14 transition "Y8"



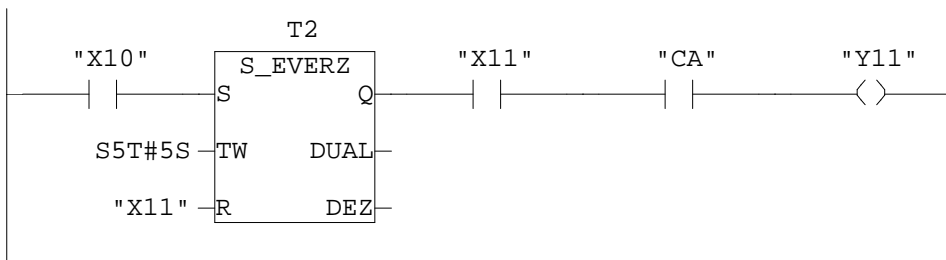
Réseau : 15 transition "Y9"



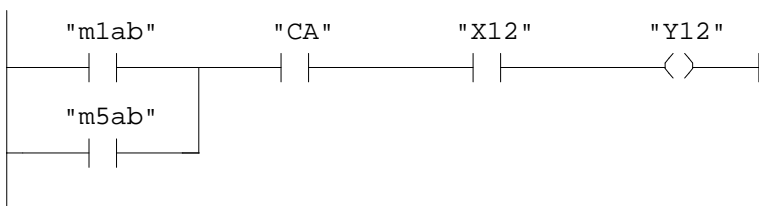
Réseau : 16 transition "Y10"



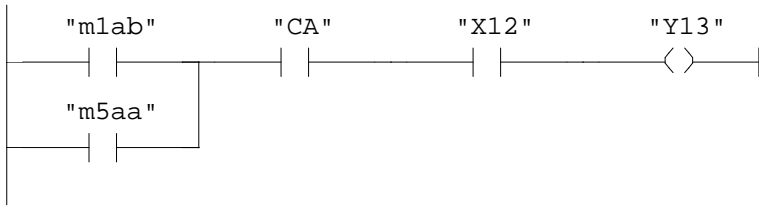
Réseau : 17 Transition "Y11"



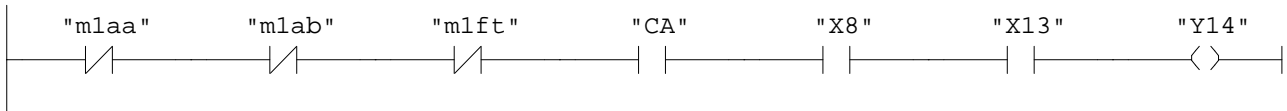
Réseau : 18 transition "Y12"



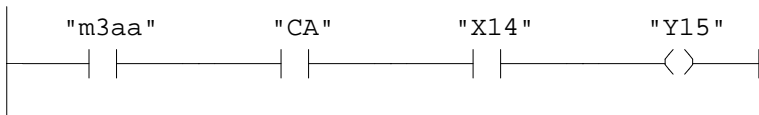
Réseau : 19 transition "Y13"



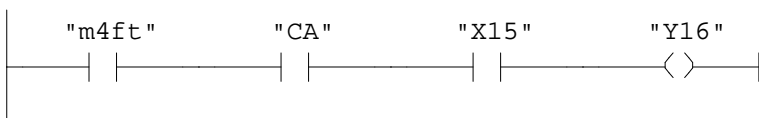
Réseau : 20 transition "Y14"



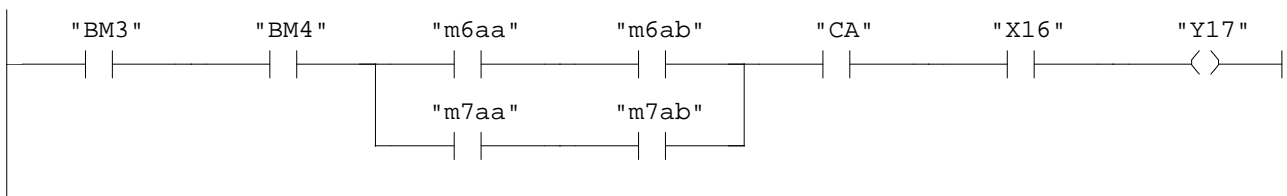
Réseau : 21 transition "Y15"



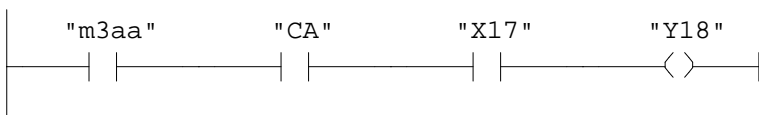
Réseau : 22 transition "Y16"



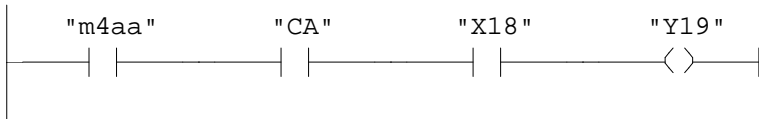
Réseau : 23 transition "Y17"



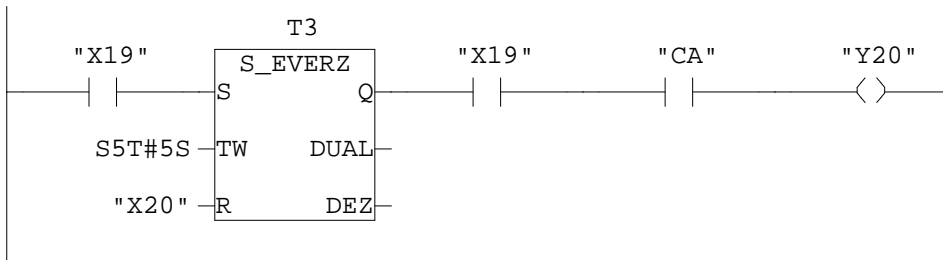
Réseau : 24 transition "Y18"



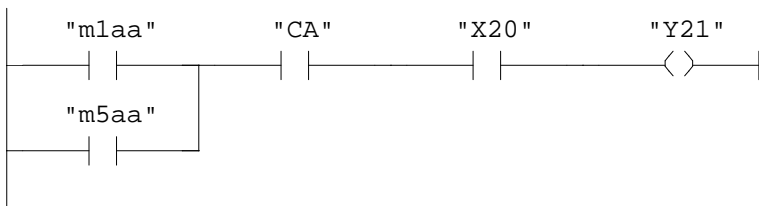
Réseau : 25 transition "Y19"



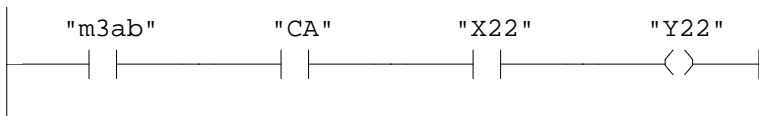
Réseau : 26 Transition "Y20"



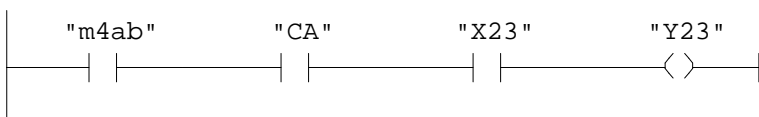
Réseau : 27 transition "Y21"



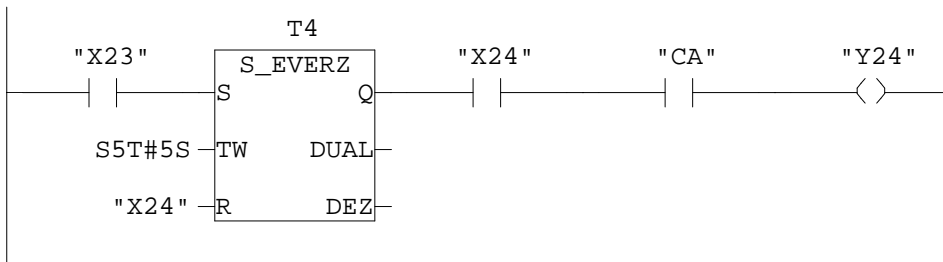
Réseau : 28 transition "Y22"



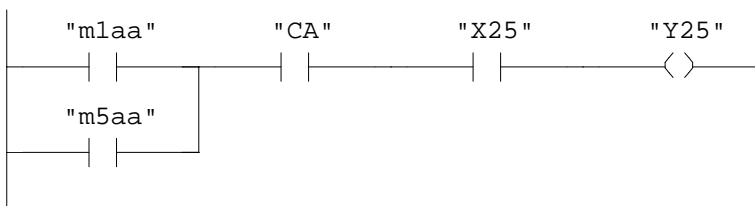
Réseau : 29 transition "Y23"



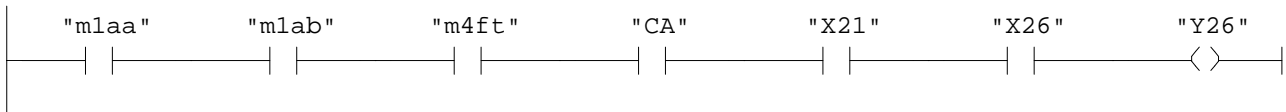
Réseau : 30 Transition "Y24"



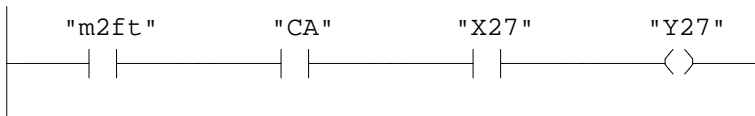
Réseau : 31 transition "Y25"



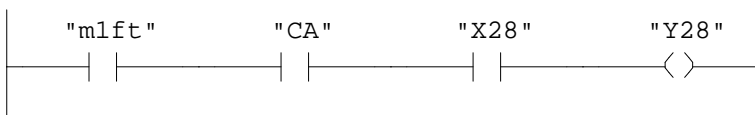
Réseau : 32 transition "Y26"



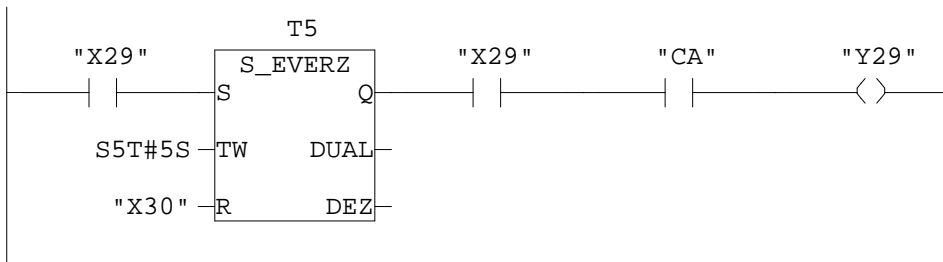
Réseau : 33 Transitin "Y27"



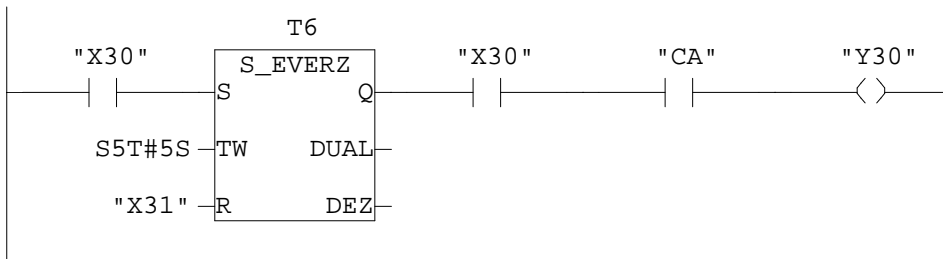
Réseau : 34 transition "Y28"



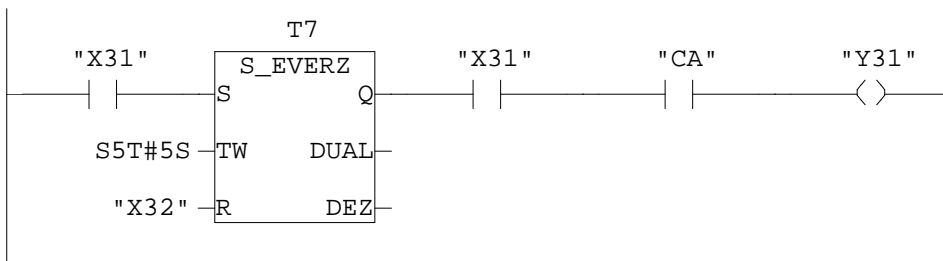
Réseau : 35 Transition "Y29"



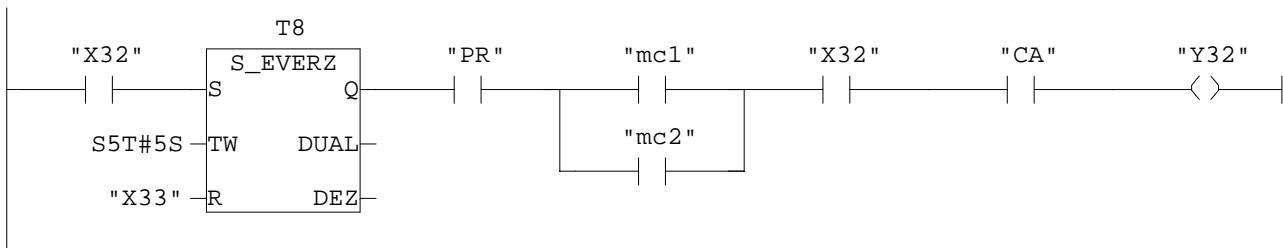
Réseau : 36 Transition "Y30"



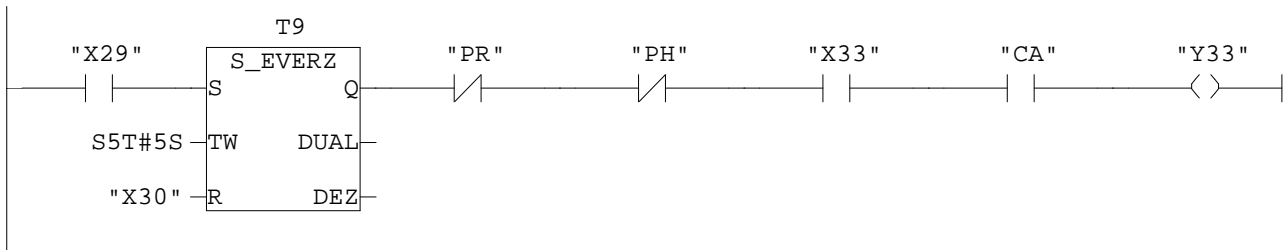
Réseau : 37 Transition "Y31"



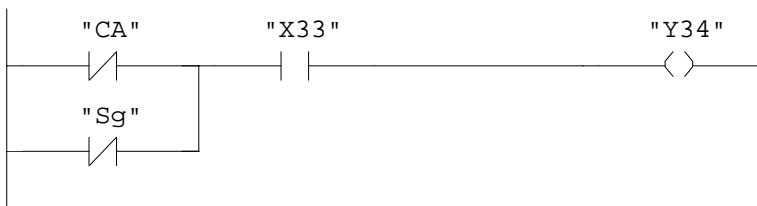
Réseau : 38 Transition "Y32"



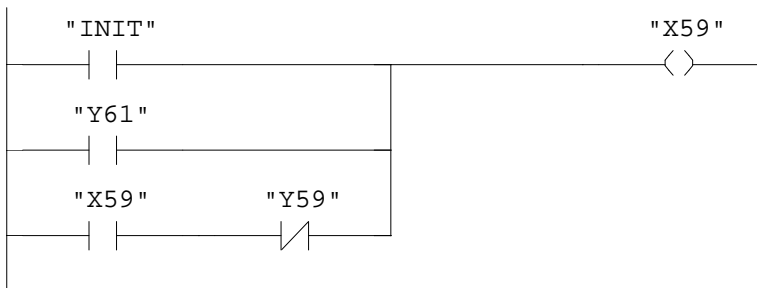
Réseau : 39 Transition "Y33"



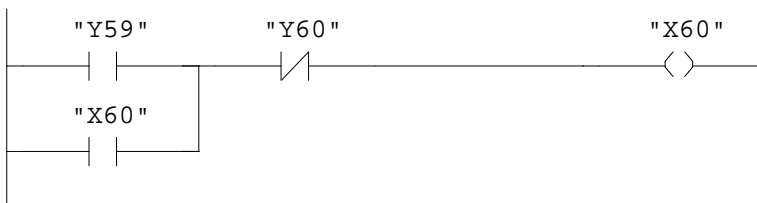
Réseau : 40 transition "Y34"



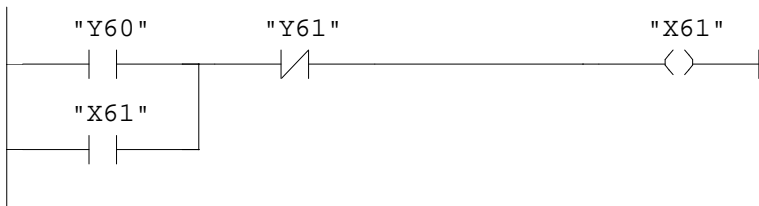
Réseau : 41 etape "X59"



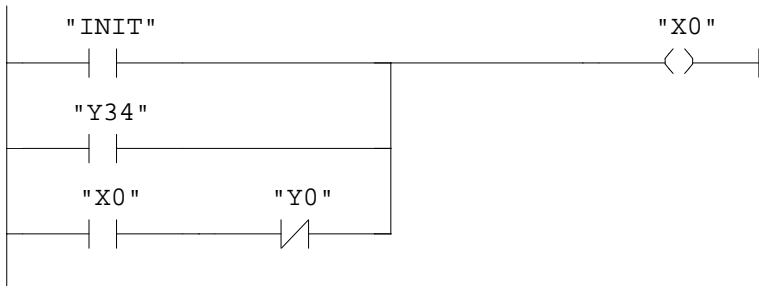
Réseau : 42 etape "X60"



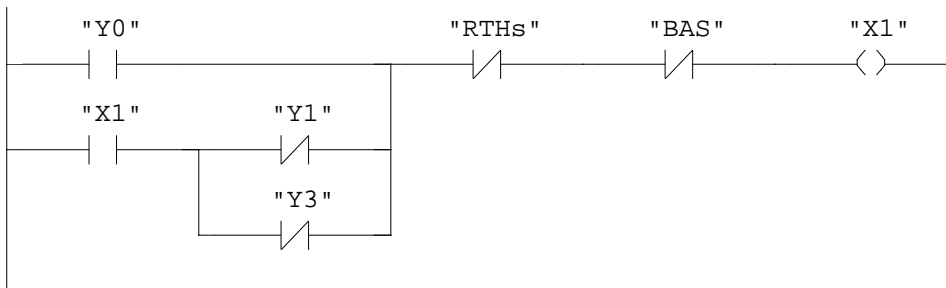
Réseau : 43 etape "X61"



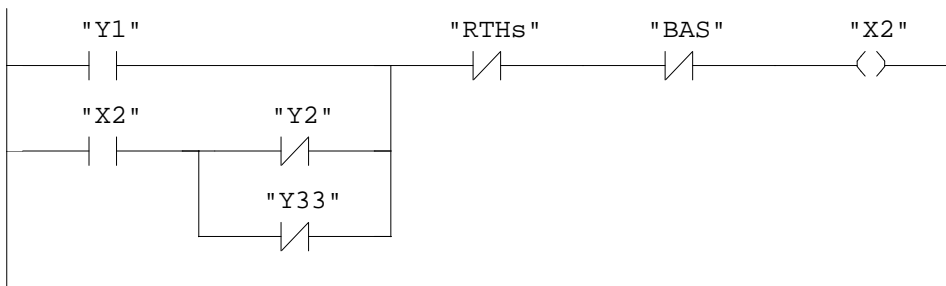
Réseau : 44 etape "X0"



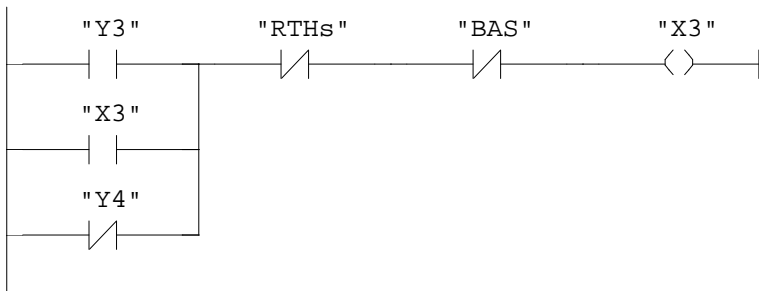
Réseau : 45 etape "X1"



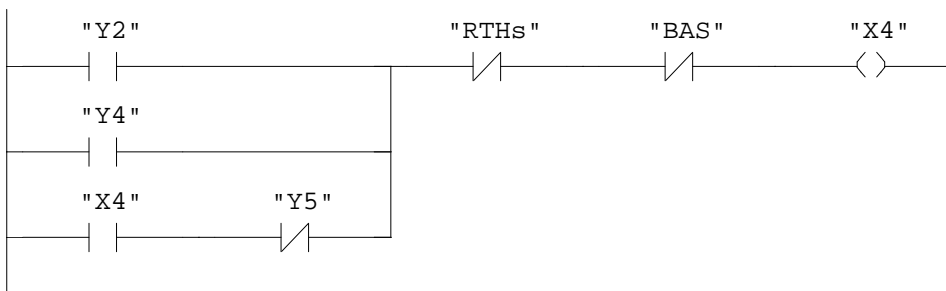
Réseau : 46 etape "X2"



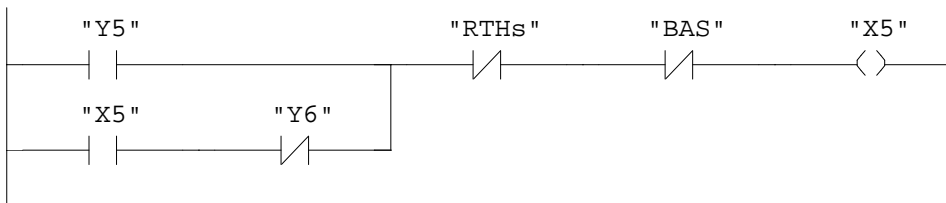
Réseau : 47 etape "X3"



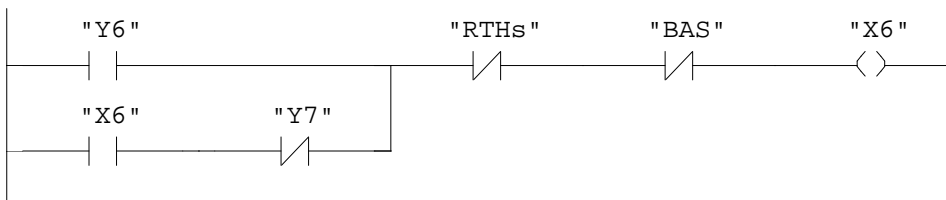
Réseau : 48 etape "X4"



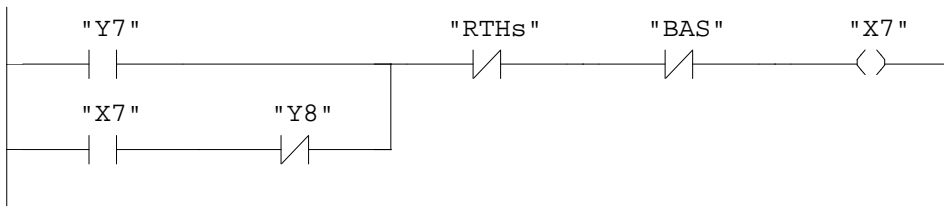
Réseau : 49 etape "X5"



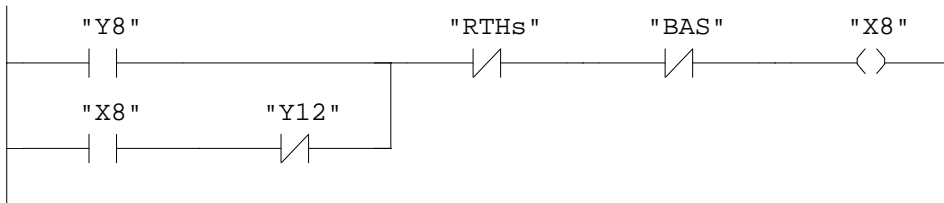
Réseau : 50 etape "X6"



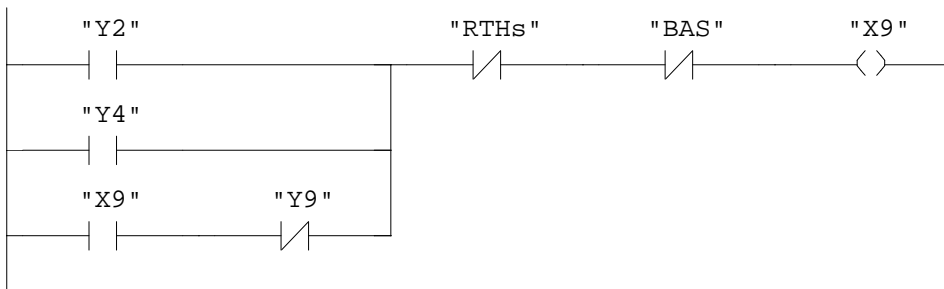
Réseau : 51 etape "X7"



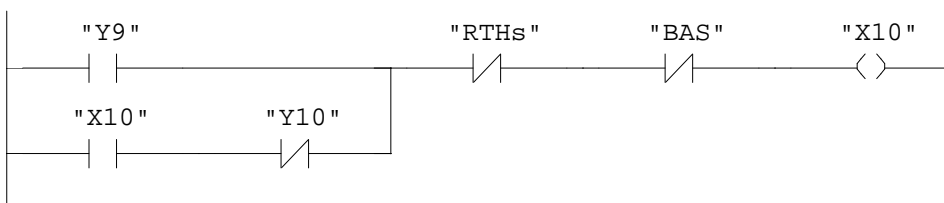
Réseau : 52 etape "X8"



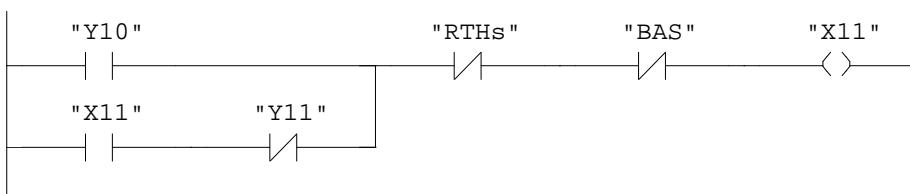
Réseau : 53 etape "X9"



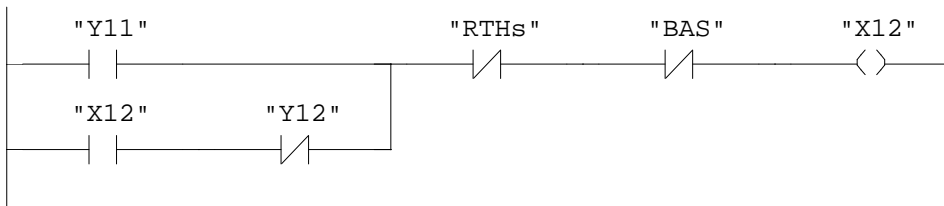
Réseau : 54 etape "X10"



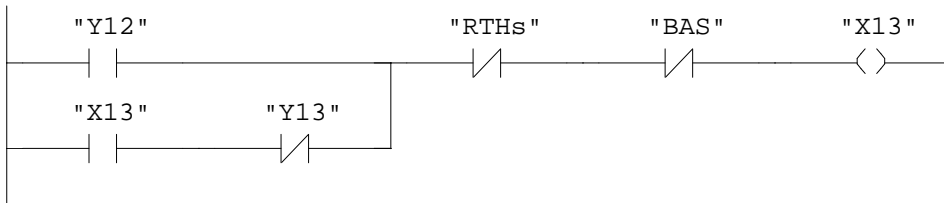
Réseau : 55 etape "X11"



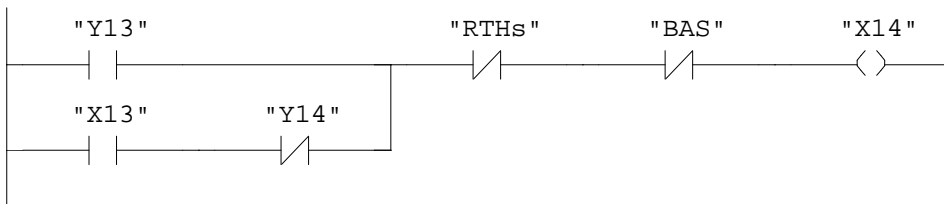
Réseau : 56 etape "X12"



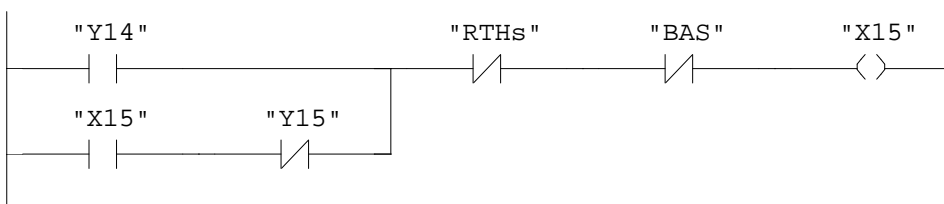
Réseau : 57 etape "X13"



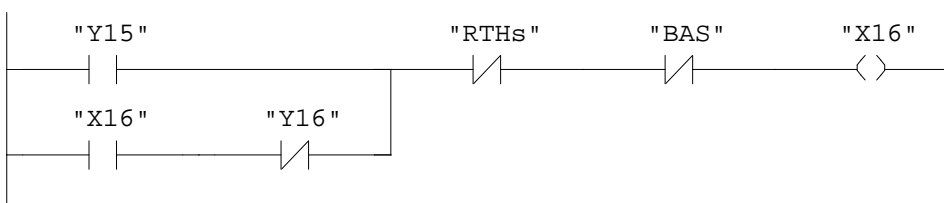
Réseau : 58 etape "X14"



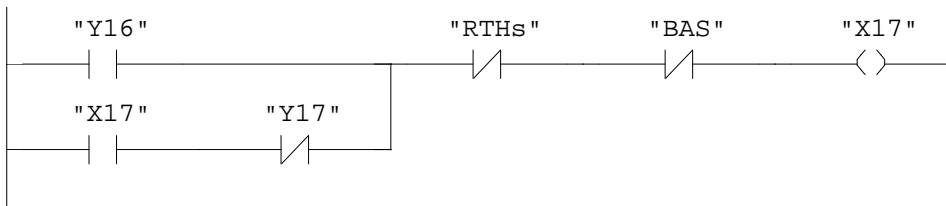
Réseau : 59 etape "X15"



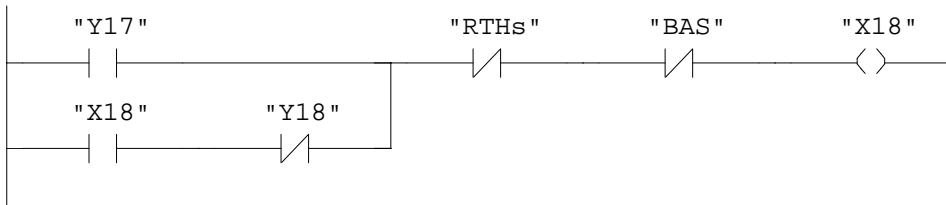
Réseau : 60 etape "X16"



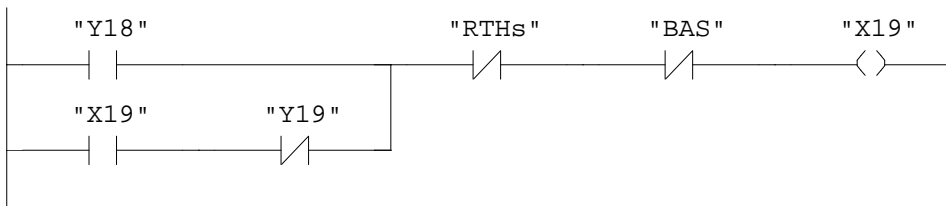
Réseau : 61 etape "X17"



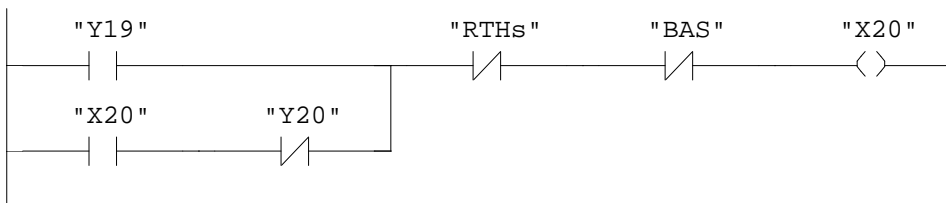
Réseau : 62 etape "X18"



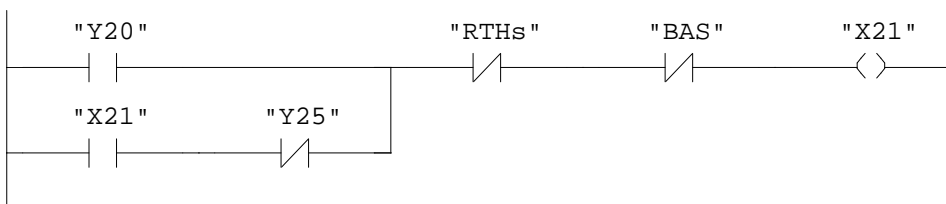
Réseau : 63 etape "X19"



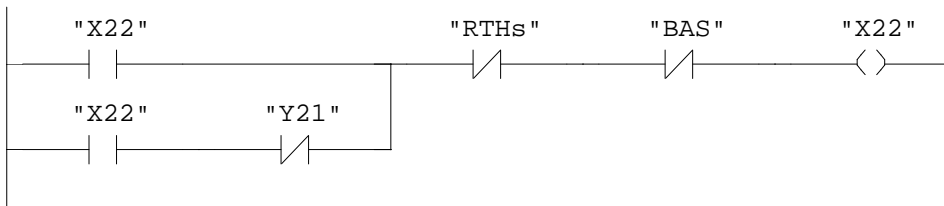
Réseau : 64 etape "X20"



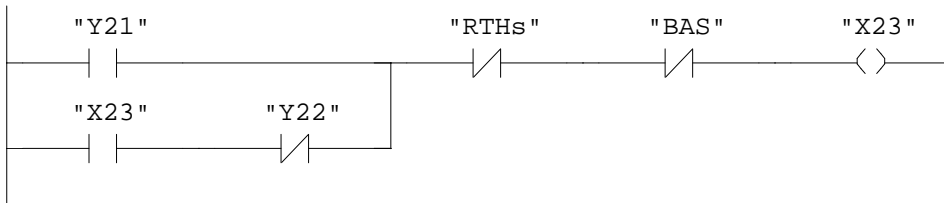
Réseau : 65 etape "X21"



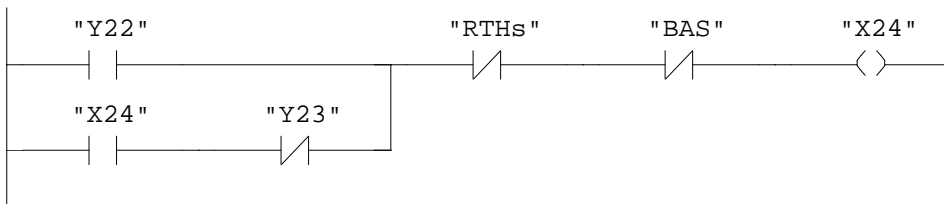
Réseau : 66 etape "X22"



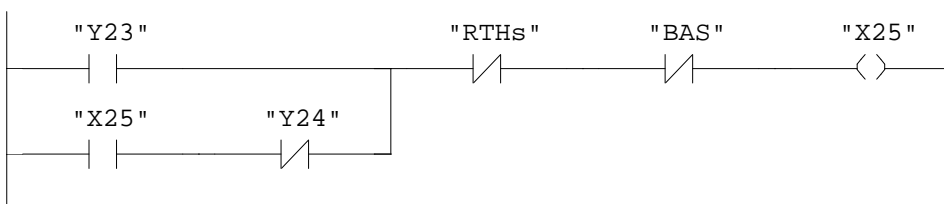
Réseau : 67 etape "X23"



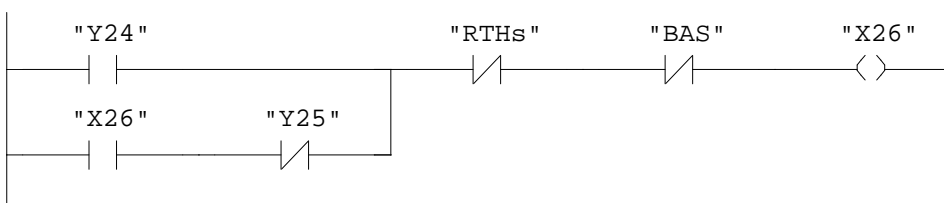
Réseau : 68 etape "X24"



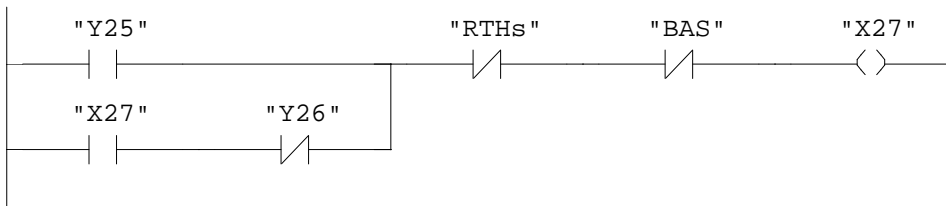
Réseau : 69 etape "X25"



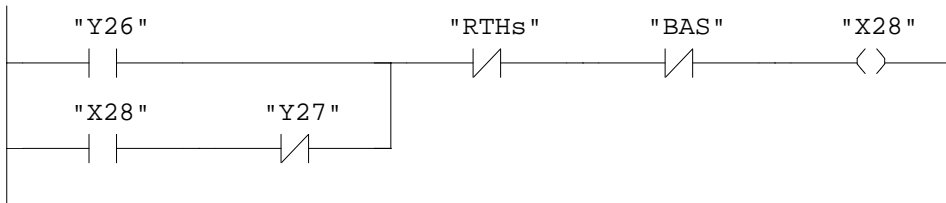
Réseau : 70 etape "X26"



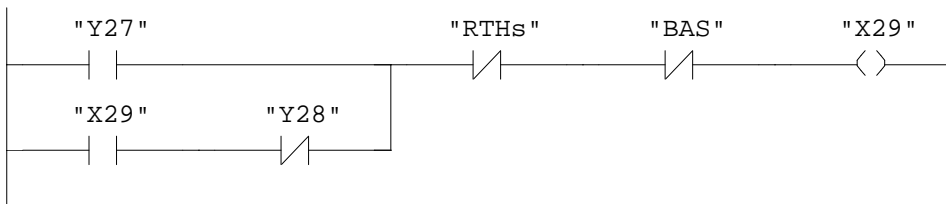
Réseau : 71 etape "X27"



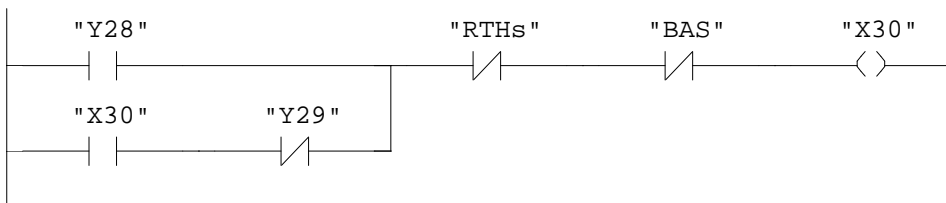
Réseau : 72 etape "X28"



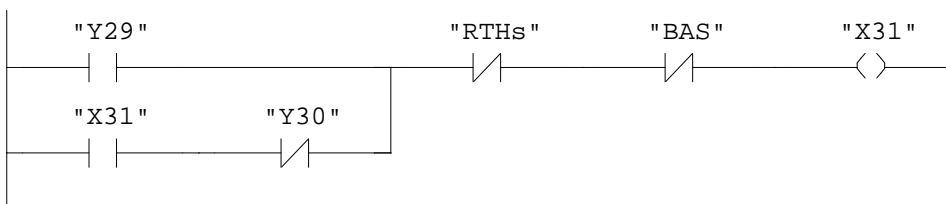
Réseau : 73 etape "X29"



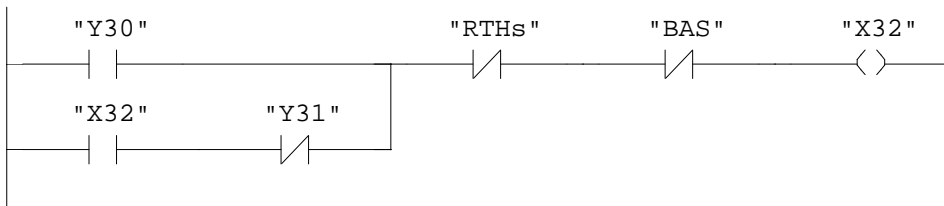
Réseau : 74 etape "X30"



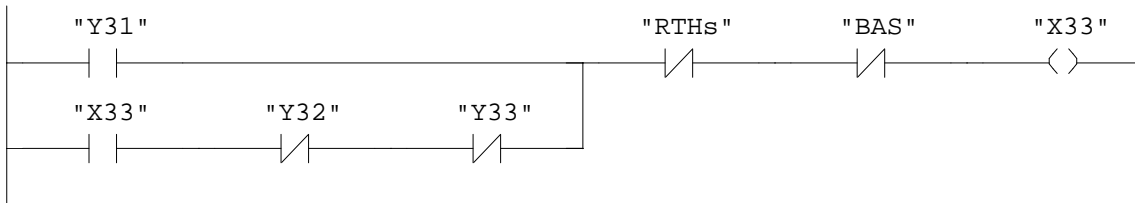
Réseau : 75 etape "X31"



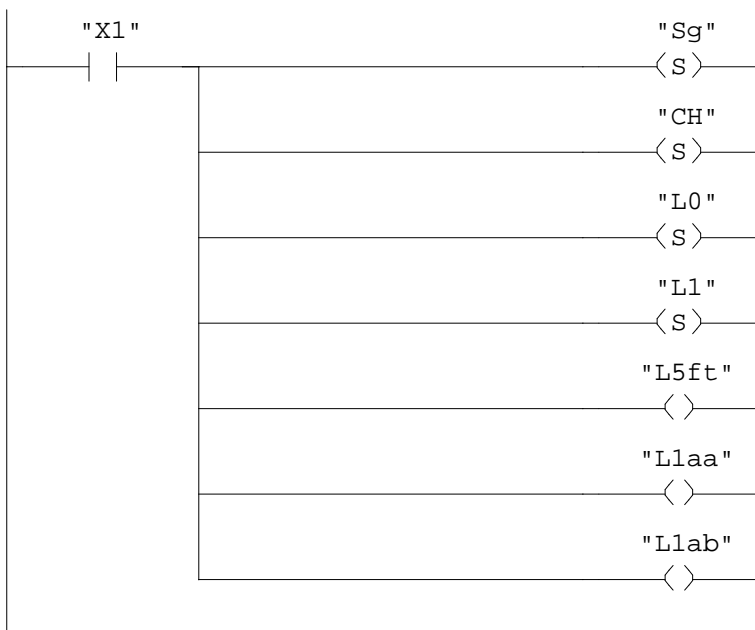
Réseau : 76 etape "X32"



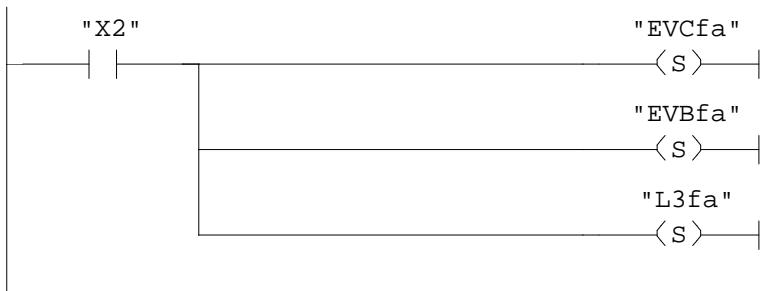
Réseau : 77 etape "X33"



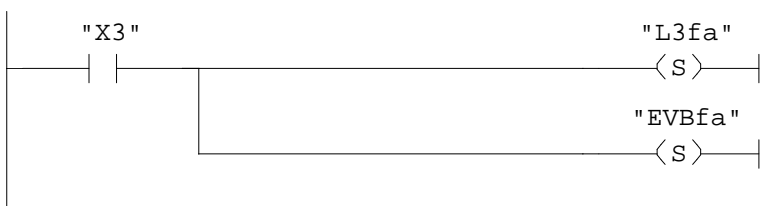
Réseau : 78 ACTION "X1"



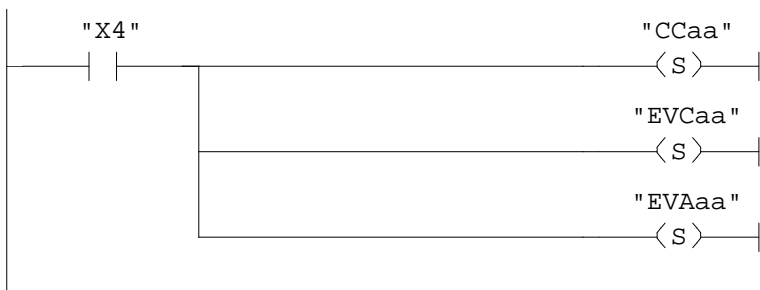
Réseau : 79 Electrovanne ''sélection 20 bars''



Réseau : 80



Réseau : 81 contacteur ''moteur broches unité AA''



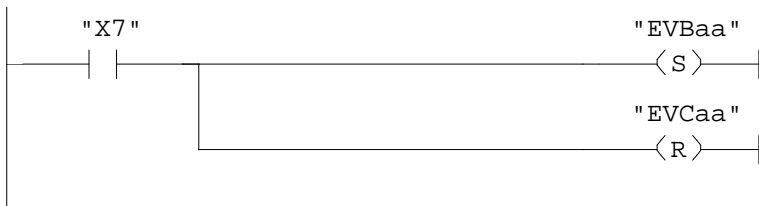
Réseau : 82 Electrovanne ''avance rapide unité AA''



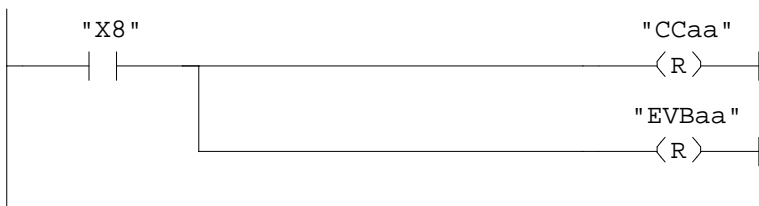
Réseau : 83



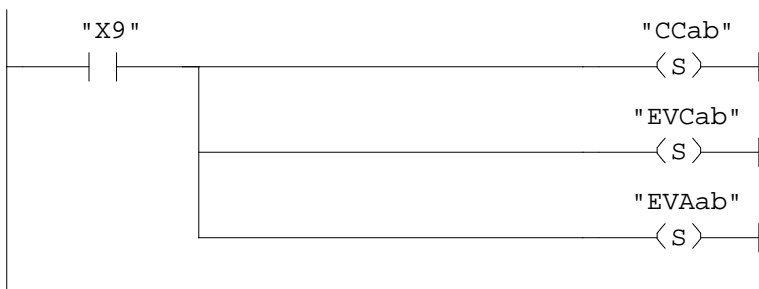
Réseau : 84 Electrovanne ''recul rapide unité AA''



Réseau : 85 contacteur ''moteur broches unité AA''



Réseau : 86



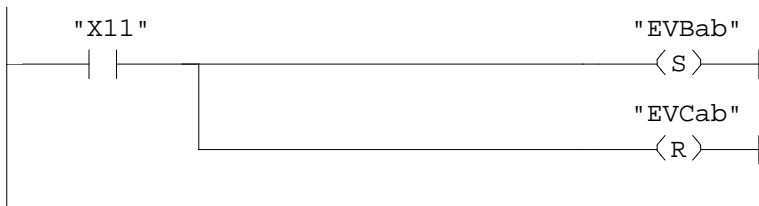
Réseau : 87 Electrovanne ''avance rapide unité AB''



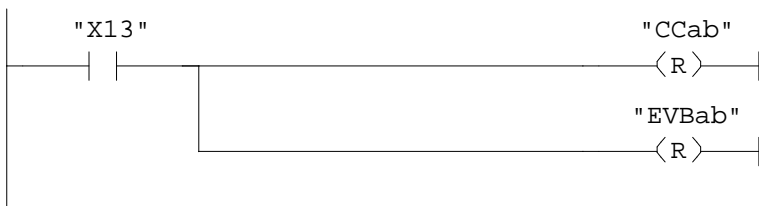
Réseau : 88 temporisation T2\X11\5S



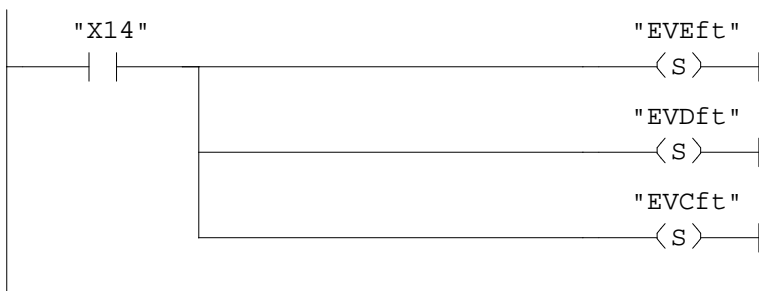
Réseau : 89 Electrovanne ''recul rapide unité AA''



Réseau : 90 contacteur ''moteur broches unité AA''



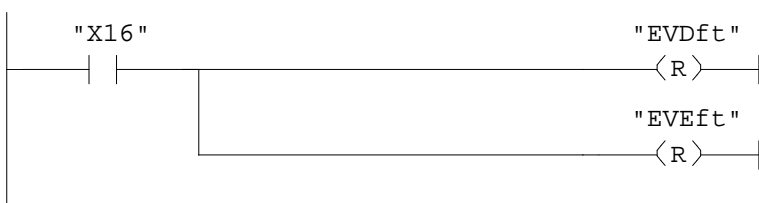
Réseau : 91 Electrovanne ''dessaillage table''



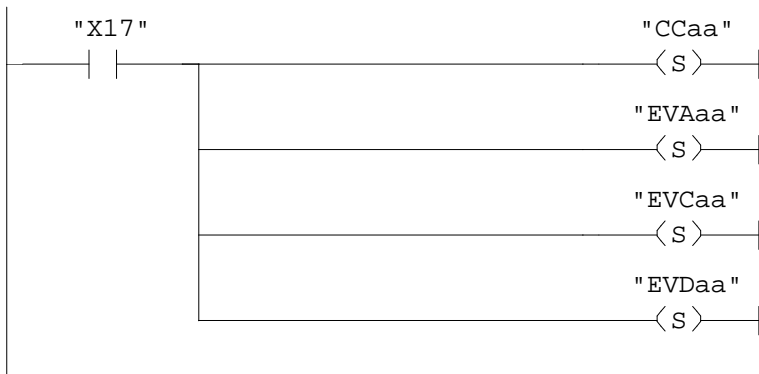
Réseau : 92 Electrovanne ''recul rapide transfert''



Réseau : 93 Electrovanne ''avance rapide unité AB''



Réseau : 94 contacteur ''moteur broches unité AA''



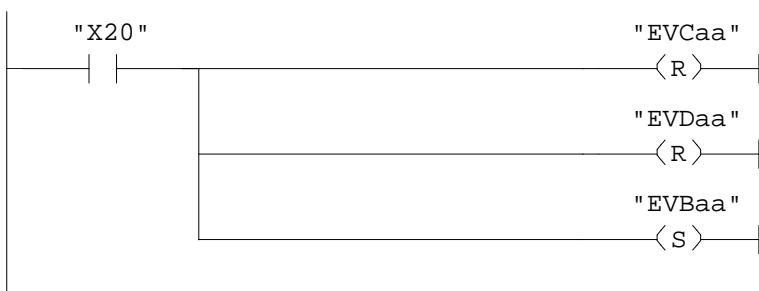
Réseau : 95 Electrovanne ''avance rapide unité AA''



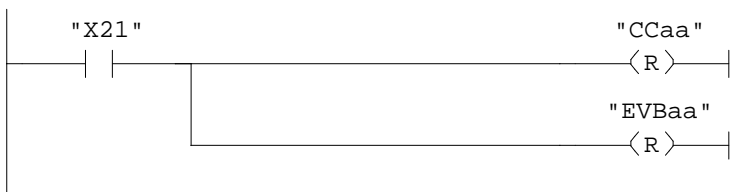
Réseau : 96 temporisation T3\X19\5S



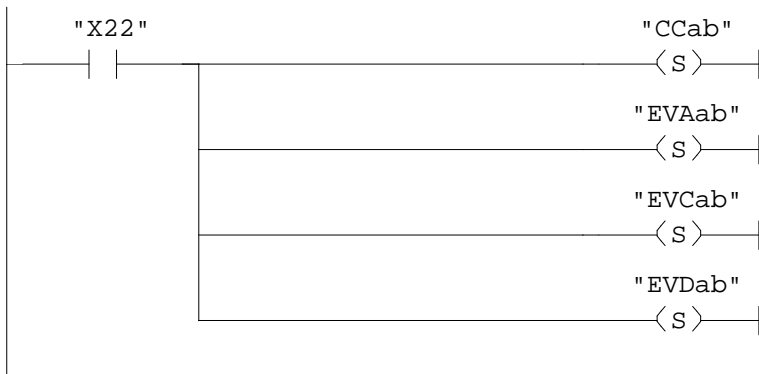
Réseau : 97 Electrovanne ''avance rapide unité AA''



Réseau : 98 contacteur ''moteur broches unité AA''



Réseau : 99 contacteur ''moteur broches unité AA''



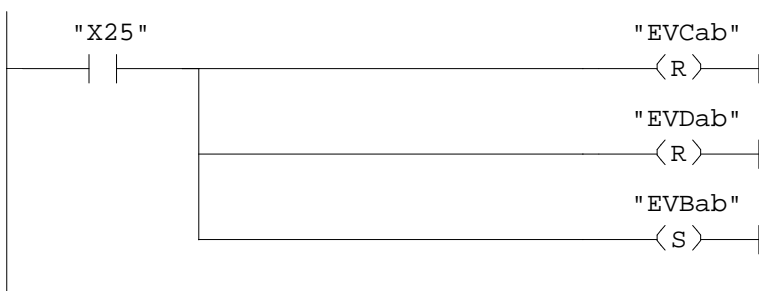
Réseau : 100 Electrovanne ''avance rapide unité AA''



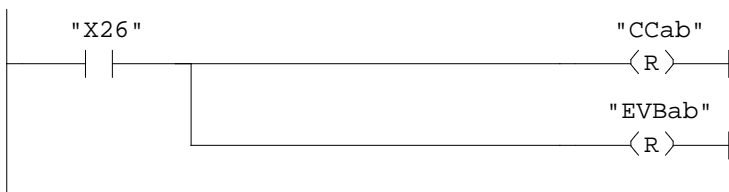
Réseau : 101 temporisation T3\X19\5S



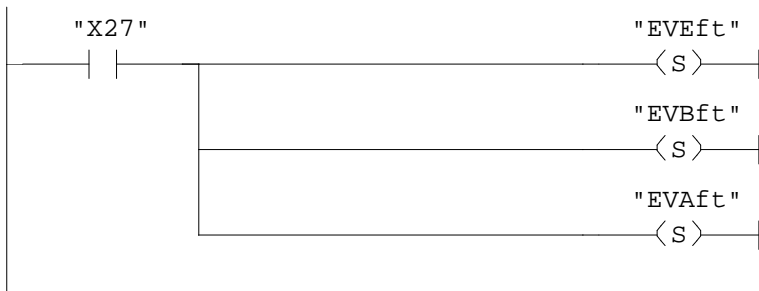
Réseau : 102 contacteur ''moteur broches unité AA''



Réseau : 103 Contacteur ''moteur broches unité AB''



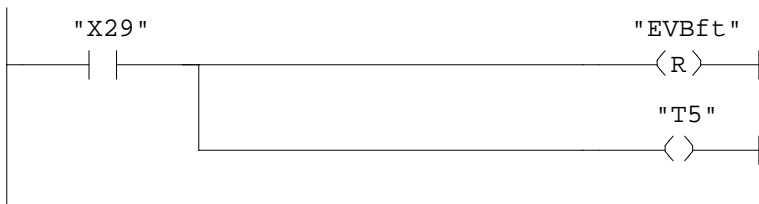
Réseau : 104 contacteur ''moteur broches unité AA''



Réseau : 105 Electrovanne ''avance rapide transfert''



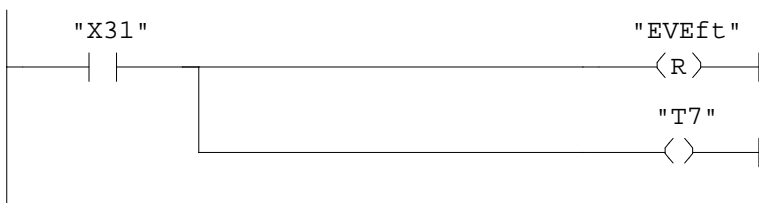
Réseau : 106 Electrovanne '' avance lente transfert''



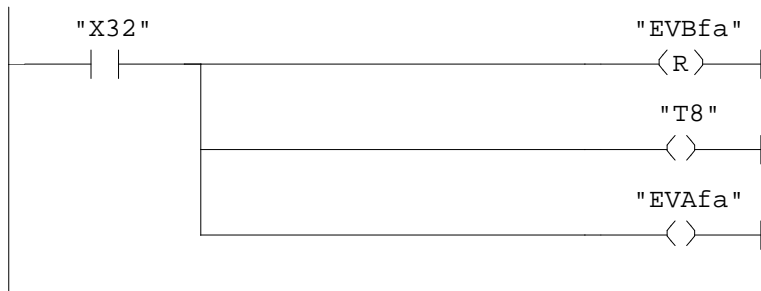
Réseau : 107 temporisation T6\X30\2S



Réseau : 108 Electrovanne '' desserage table''



Réseau : 109 Electrovanne ''bridage pièce''



Réseau : 110 temporisation T9\X33\5S





Annexe C

Les capteurs



1- Les Capteurs :

Les capteurs sont des composants de la chaîne d'acquisition ; ils prélèvent une information sur le comportement de la partie opérative et la transforment en une information exploitable par la partie commande. Pour pouvoir être traitée, cette information sera portée par un support physique (énergie), on parlera alors de signal (qui est généralement de nature électrique ou pneumatique).

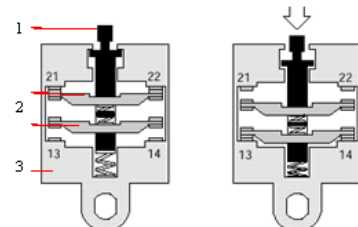
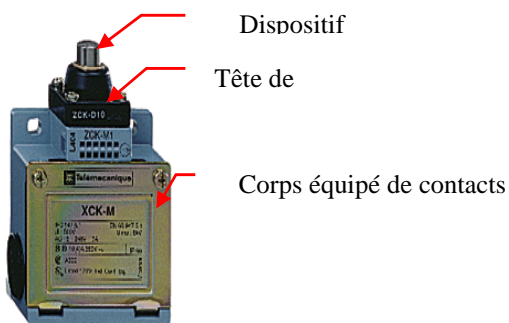
1-1- Détecteur de présence :

a) Définition :

Un détecteur de présence est un capteur mécanique, il est utilisé pour détecter la présence ou le passage de matériaux rigides, en butée, fin de course, présence pièce sur support d'usinage

Les interrupteurs de position sont constitués de trois éléments de base suivants:

- Un contact électrique (1)
- Un corps (2)
- Une tête de commande avec son dispositif d'attaque (3).



b) Caractéristiques générales:

Symbole:

Portée nominale: Contact direct.

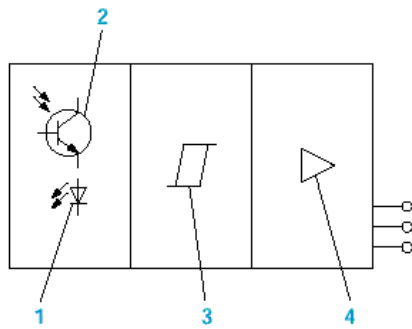
Tension d'alimentation: 240 V AC ; 250 V DC max.

1-2- Les détecteurs photoélectriques

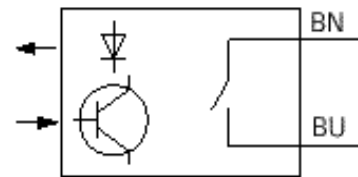
a) Définition :

Un détecteur photoélectrique se compose essentiellement d'un émetteur de lumière (diode électroluminescente) associée à un récepteur sensible à la lumière reçue (phototransistor).

Une diode électroluminescente émet de la lumière lorsqu'il est traversé par un courant électrique. Il y a détection quand la cible pénètre dans le faisceau lumineux émis par le détecteur et modifie la lumière reçue par le récepteur pour provoquer un changement d'état de la sortie.



- 1 Emetteur de lumière
- 2 Récepteur de lumière
- 3 Etage de mise en forme
- 4 Etage de sortie



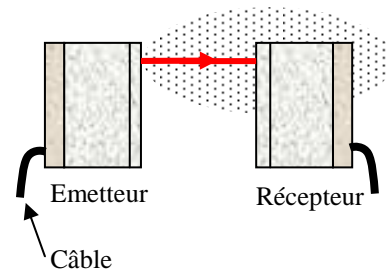
b) Caractéristiques générales:

Portée nominale: jusqu'à 50 m pour le type barrage.
Tension d'alimentation: 20 à 264 VAC; 10 à 30 VDC.

c) Détecteurs de type barrage

Principe de fonctionnement:

Un détecteur de type barrage est composé d'un émetteur de lumière associé à un récepteur photosensible. Dans le cas du système barrage, les deux composants sont indépendants et placés l'un en face de l'autre. La présence d'un objet dans le champ du capteur interrompt le faisceau lumineux et le récepteur délivre alors un signal.



d) Caractéristiques générales:

Portée nominale: jusqu'à 50 m (100 m : laser).
Tension d'alimentation: 20 à 264 VAC; 10 à 30 VDC.

c) Applications:

Barrages et réflex: Détection du passage de pièces ou de personnes. Application: barrière détection grands magasins ou des portes de l'ascenseur.

Proximités: Ils sont susceptibles de détecter tous les matériaux non noirs (pas de réflexion). Application: ouverture des portes automatiques des grands magasins.

1-3- Les contrôleurs de niveau :

a) Définition :

Les contrôleurs de niveau FSK HYDAC sont des éléments assurant l'affichage et le contrôle de niveau des fluides hydrauliques.

Le FSK est un thermo contacteur avec au choix un contact ouvrant (type O), fermant (type C) ou inverseur (type W). Sonde thermométrique pour la surveillance de la température de l'huile dans le réservoir hydraulique, avec affichage en °C. Affichage en °C et °F sur demande.

b) Fonctionnement du FSK

