



Ressource de formation individualisée



MI IV 333 B

CODER UNE SEQUENCE DE GRAFCET

**Maintenance industrielle
Niveau IV**

*Préparer l'approvisionnement***Ordre de travail**

A partir du dossier MI IV 22 B, programmer sur PC et avec le logiciel PL7 MICRO le fonctionnement en mode automatique du système PREHENSICC.

Compétence

S'INFORMER : Sur la procédure de programmation d'un API
COMMUNIQUER : Utiliser le langage PL7 MICRO

Objectif

Suivre une procédure de programmation
Intégrer une démarche de programmation dans le but de faire fonctionner en mode automatique un système automatisé.

Pré Requis

Maîtriser La lecture du GRAFCET

- Point de vue OPERATIVE
- Point de vue COMMANDE
- Point de vue AUTOMATE

Matériel nécessaire

1 dossier technique PREHENSICC
1 PC avec le logiciel PL7 MICRO
1 Dossier MI IV 22 B

Conditions de sécurité

Si intervention sur système, respecter les consignes de sécurité.

Durée

2h

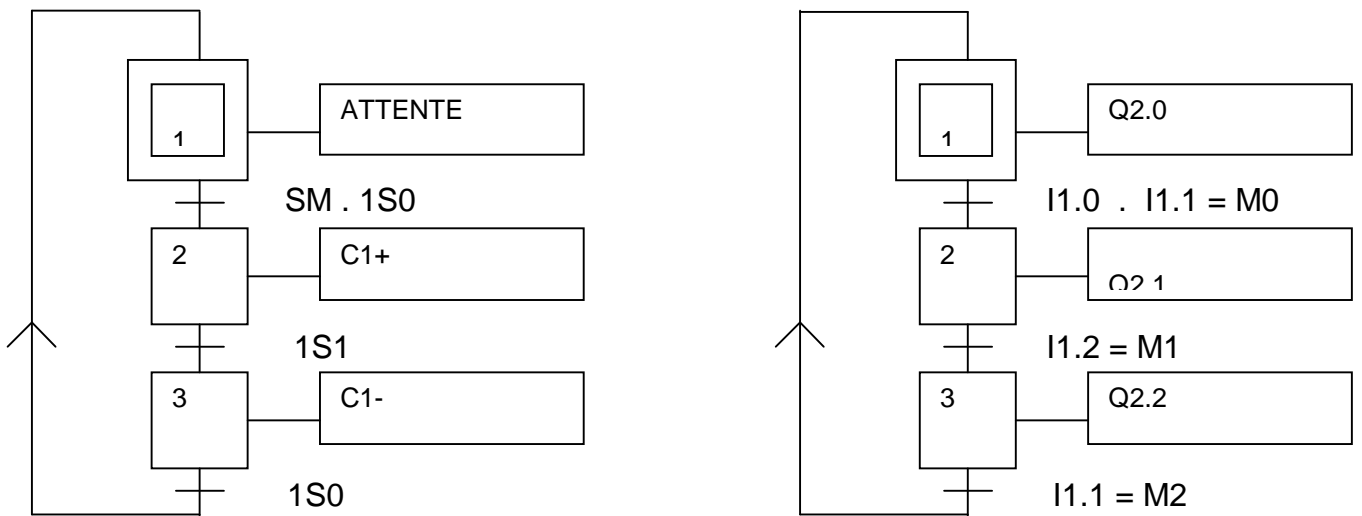


**Ce signe indique que cette activité nécessite une évaluation. Lorsqu'il est présent, faites appel à votre formateur
Lorsqu'un choix est demandé ..oui ..non, entourer la réponse convenable**

Dans l'industrie, il existe de nombreux fabricants d'automates programmables (Allan Bradley, Siemens, Télémécanique ,etc...) et chaque fabricant développe un grand nombre de produits (exemple pour Télémécanique : TSX 17, 27, 37, 47, MICRO, NANO, etc...)

Pour programmer tous ces automates, chaque constructeur a développé son propre mode de « langage ». On se propose aujourd'hui d'étudier un des langages utilisés par le groupe Télémécanique, à savoir le PL7 MICRO.

Pour « étudier » ce langage, prenons l'exemple suivant, dont on donne le GRAFCET point de vue COMMANDE et le GRAFCET point de vue API.



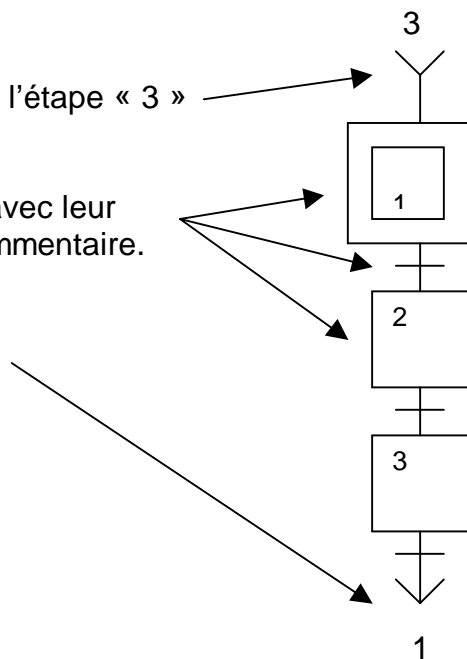
A) LE PROGRAMME :

1) La première étape, consiste à transformer le GRAFCET de manière à ce qu'il soit compréhensible par l'automate. En langage automate on appelle cette étape le « CHART ». Notre GRAFCET deviendra :

* cette flèche (inversée) indique que l'on vient de l'étape « 3 »

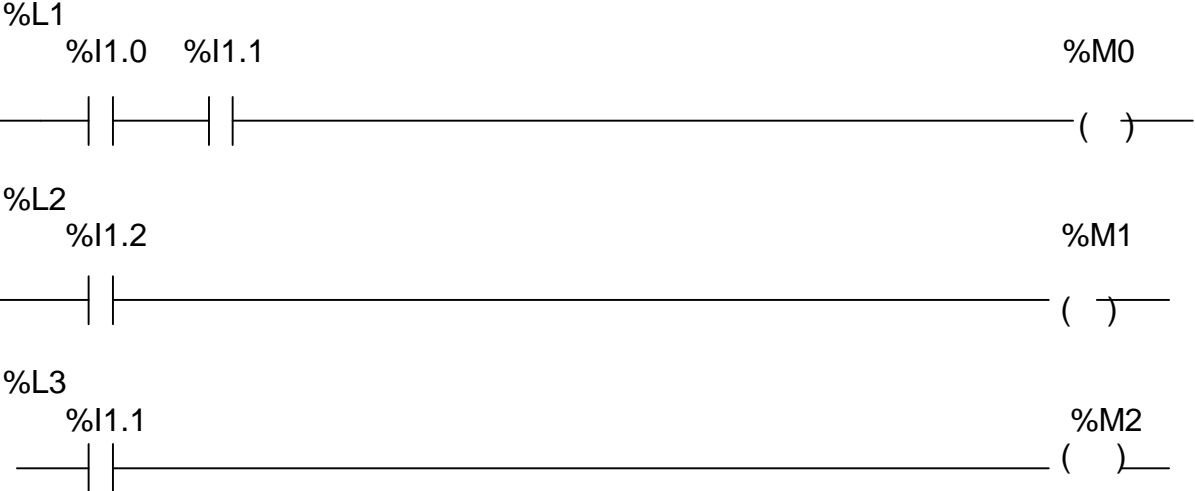
* A noter que l'on ne représente que les étapes avec leur numéro et les transitions. Le tout sans aucun commentaire.

* Cette flèche indique que l'on va à l'étape « 1 ».



2) La deuxième étape consiste à traiter les transitions. On appelle cette partie PRELIMINAIRES. Nous pouvons constater sur notre GRAFCET point de vue API (page précédente) qu'à chaque transition est associé une mémoire « M ». Cette mémoire est appelée BIT INTERNE.

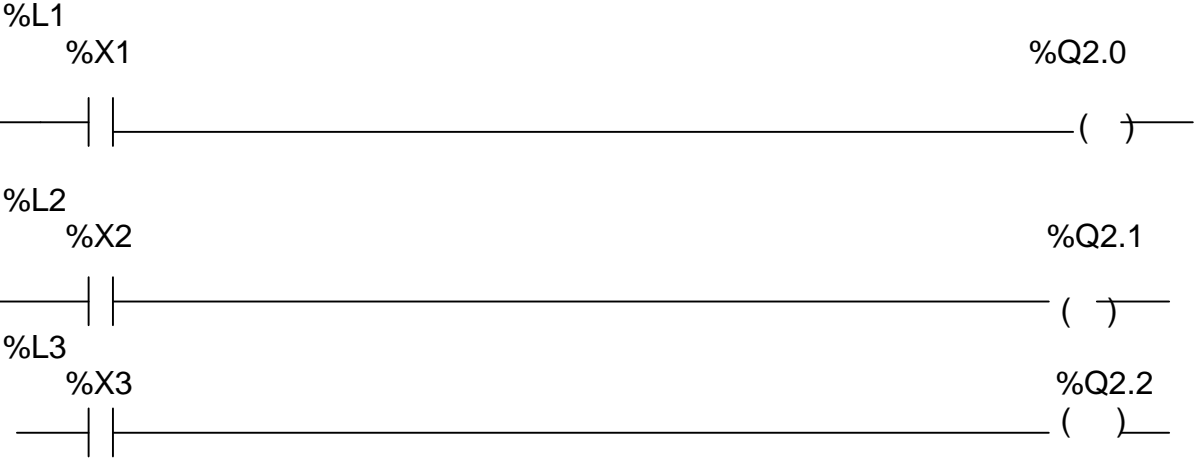
Lorsque nos « BITS INTERNES » sont affectés, le programme devient :



L'automate va lire son programme de la façon suivante:

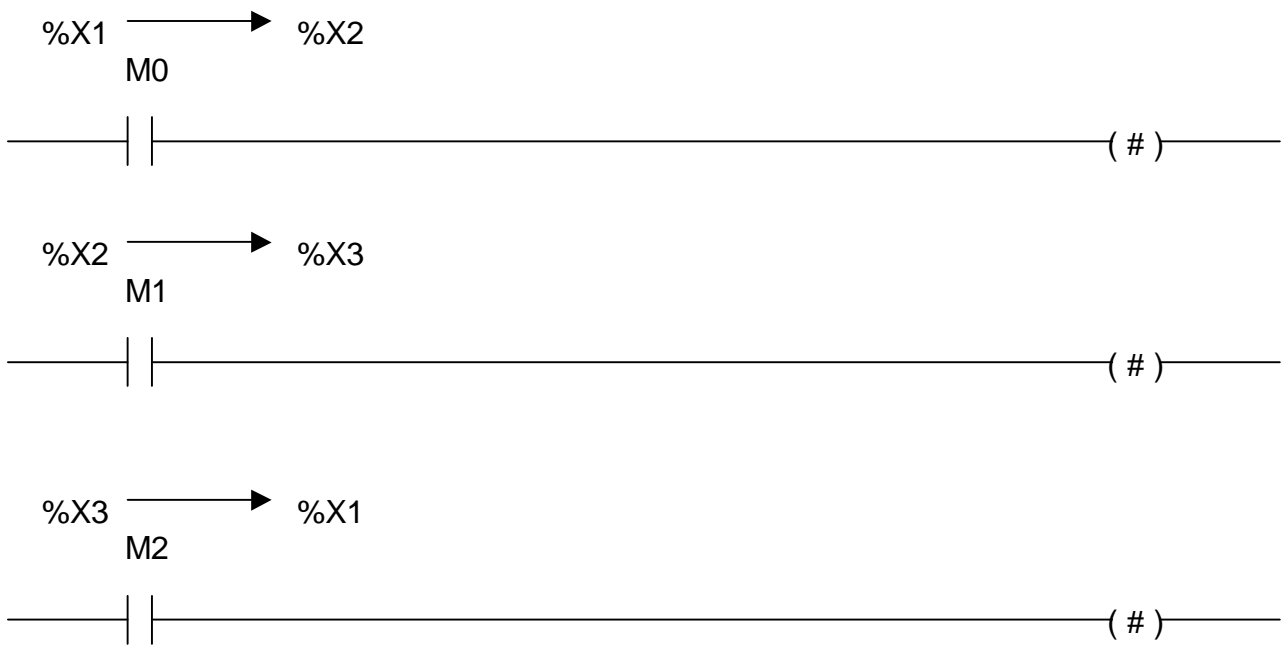
- Pour L1 :
Si j'ai l'entrée I1.0 ET si j'ai l'entrée I1.1, alors j'affecte le BIT INTERNE M0
- Pour L2 :
Si j'ai l'entrée I1. ,alors j'affecte le BIT INTERNE M
- Pour L3 :
Si j'ai l'entrée I1. , alors j'affecte le BIT INTERNE M

3) La troisième et dernière étape consiste à affecter les sortie correspondantes aux étapes de notre GRAFCET . Cette partie est appelée POSTERIEUR. (cf GRAFCET point de vue API). Le programme devient :



- L'automate va lire son programme de la façon suivante:
- Pour L1 :
Si j'ai l'étape X1, alors j'affecte la sortie Q2.0
 - Pour L2 :
Si j'ai l'étape ,alors j'affecte la sortie
 - Pour L3 :
Si j'ai l'étape , alors j'affecte la sortie

4) La quatrième et dernière étape consiste à faire « dérouler » le GRAFCET. En effet, il faut indiquer à l'automate quand passer d'une étape à l'autre. Le programme deviendra :



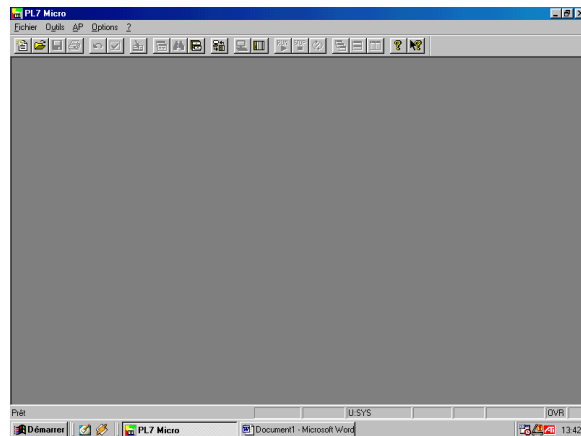
La « lecture » de l'automate pourrait de déchiffrer ainsi :

- pour passer de l'étape 1 à l'étape 2, il faut le BIT INTERNE M0.
- pour passer de l'étape à l'étape , il faut le BIT INTERNE
- pour passer de l'étape à l'étape , il faut le BIT INTERNE

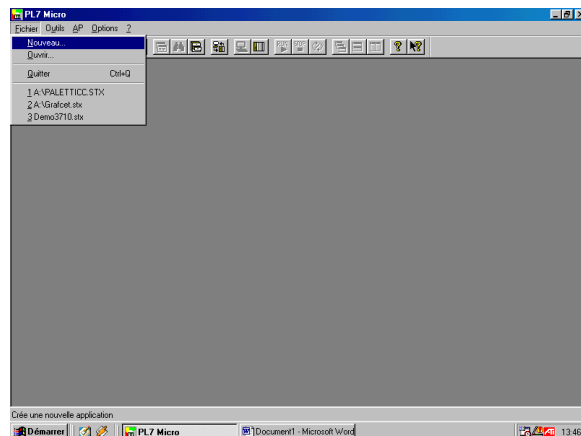
B) LE PROGRAMME

B1 : Allumer l'écran puis l'unité centrale du PC.

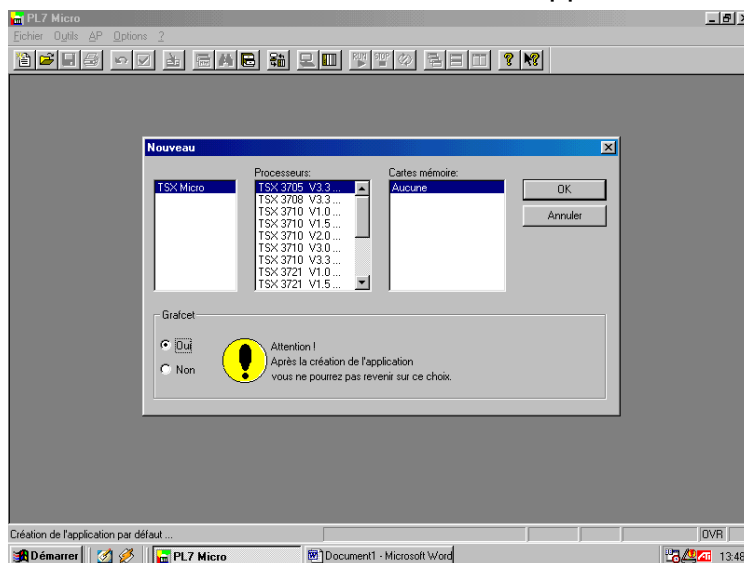
B2 : Double clic sur l'icône PL7 MICRO. L'écran suivant apparaît.



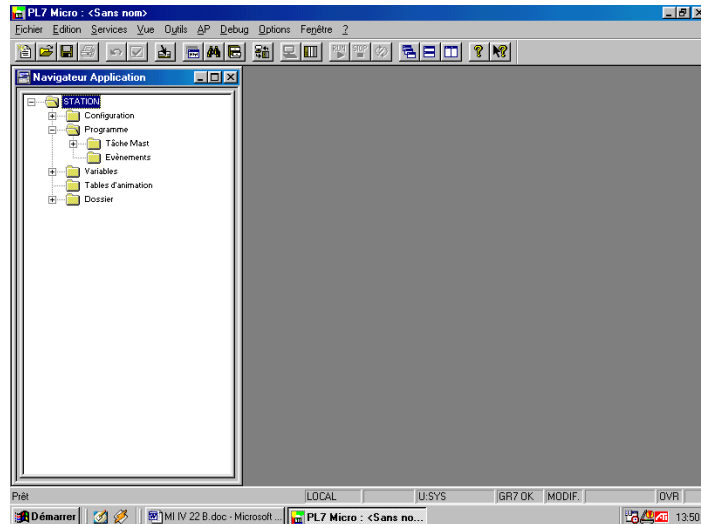
B3 : Cliquer sur « Fichier ». L'écran suivant apparaît.



B4 : Cliquer sur « nouveau ». L'écran suivant apparaît.



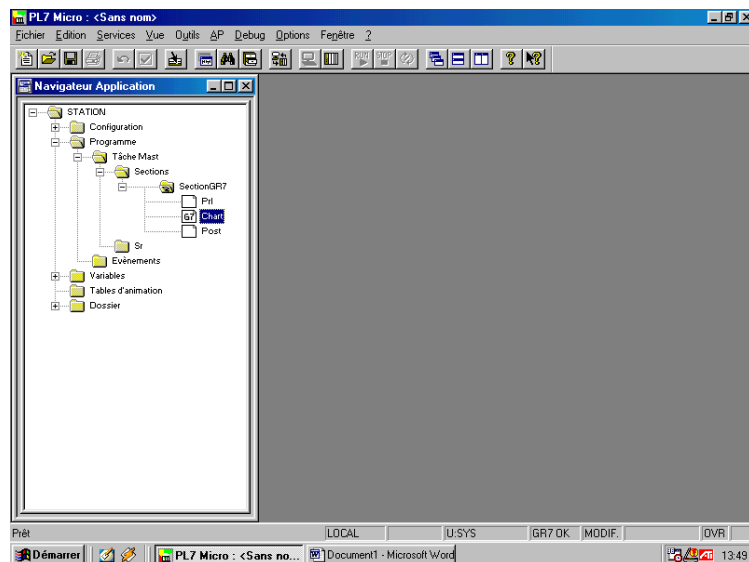
B5 : Sélectionner l'API correspondant au votre (EX : TSX 3705 V3.3)
 B6 : En dessous de GRAFCET, cliquer OUI. L'écran suivant apparaît.



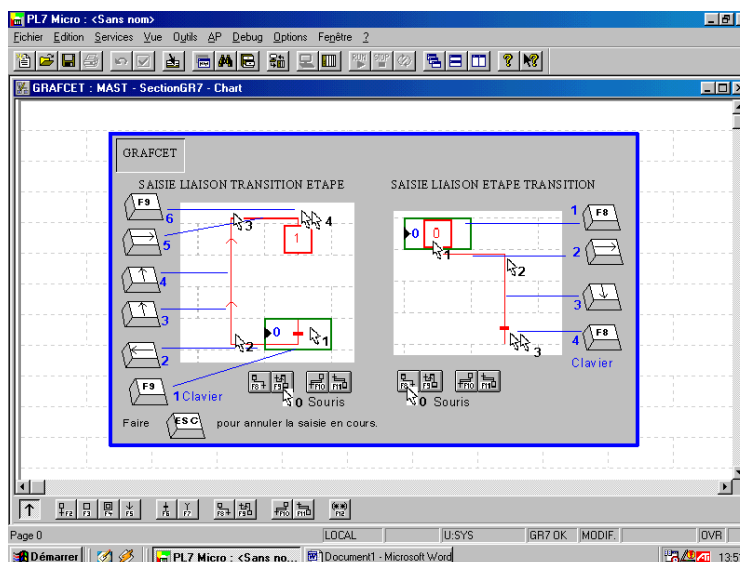
B7 : Double clic sur « Tache Mast »

B8 : Double clic sur « Sections »

B9 : Double clic sur « Section GR7 ». L'écran suivant apparaît.

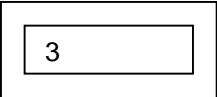


B10 : Double clic sur « Chart ». L'écran suivant apparaît.

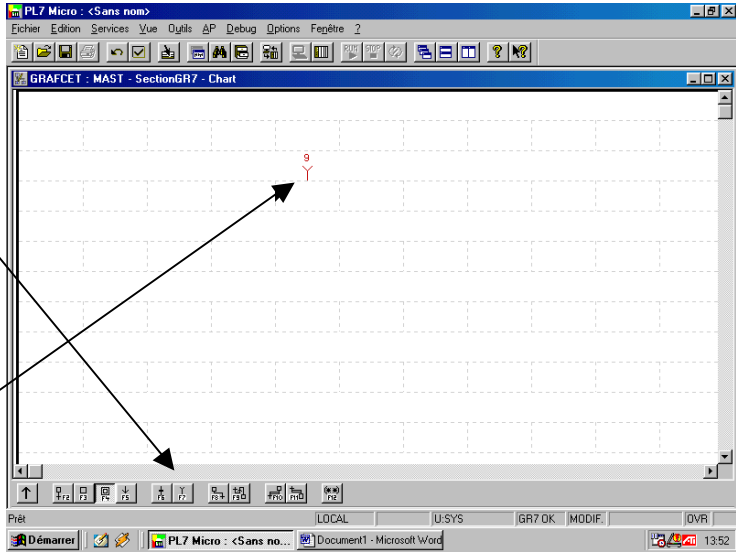


B11 :Etudiez bien cette fenêtre, elle vous donne les touches à actionner si vous n'utilisez pas la souris. Puis cliquer sur n'importe quelle touche. L'écran suivant apparaît.

- 1) Cliquer sur l'élément que l'on désire placer
- 2) Implanter cet élément à l'emplacement désiré. Une mini fenêtre apparaît et « demande » le numéro de l'organe. Dans le cas de notre exemple, vue que l'on vient de l'étape 3, alors taper 3 et valider :

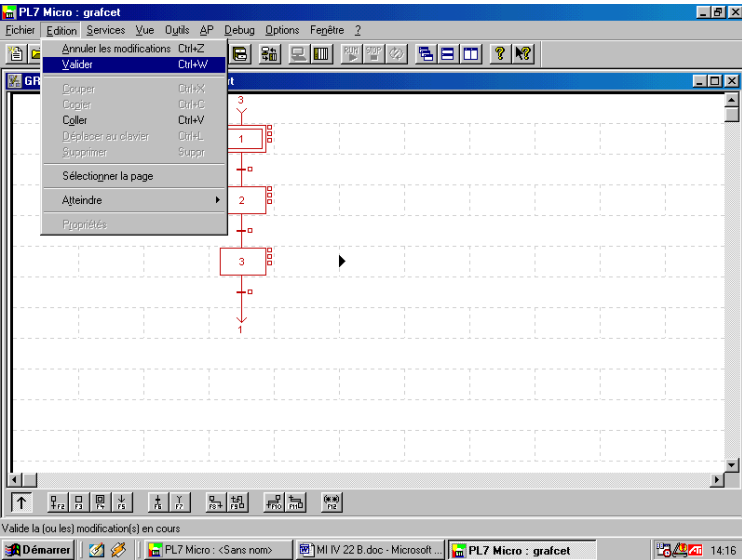


puis entrée



3) L'écran devient :

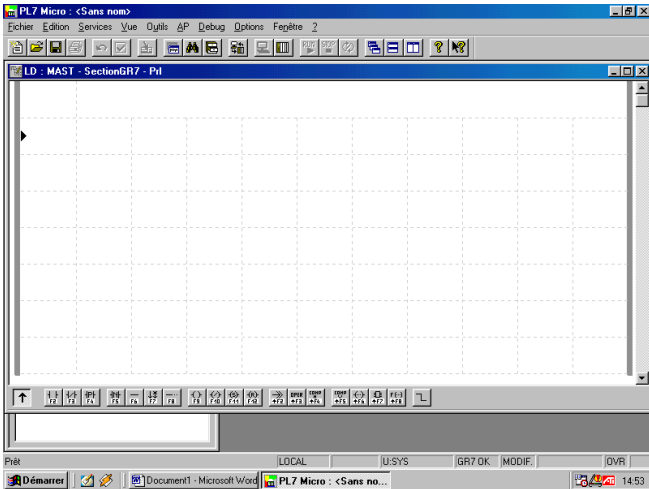
B12 : Lorsque votre GRAFCET est réalisé, il vous faut le valider en allant dans « édition »



vous pouvez alors fermer cette fenêtre par : X

B13 : Automatiquement, vous revenez à la figure B9.

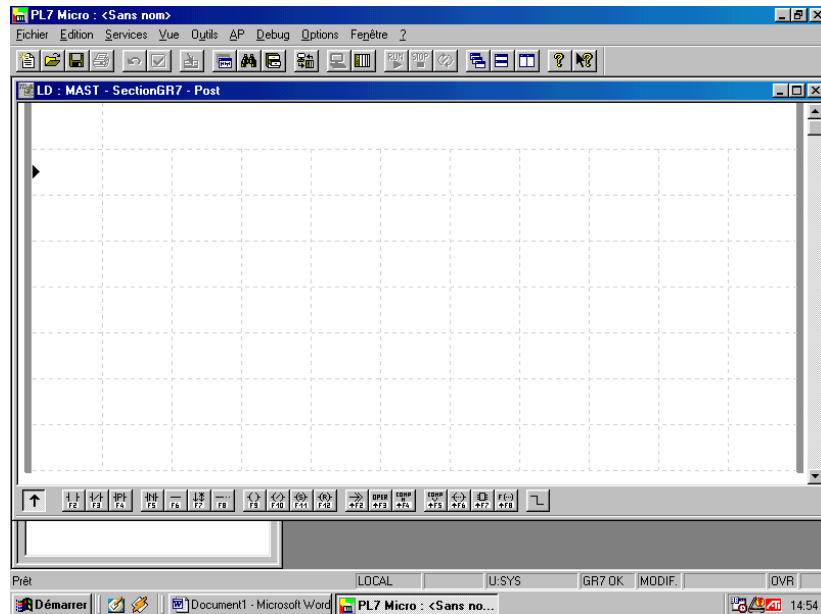
Un double clic sur « Prel » fera apparaître un écran de propriétés des préliminaires, sur lequel il suffit de valider en tapant « entrée » pour arriver à l'écran de programmation des préliminaires.



Réaliser votre programmation de la même façon que le « Chart », penser à valider à la fin de la programmation et sortez de votre écran.

B14 : Automatiquement vous revenez à la figure B9.

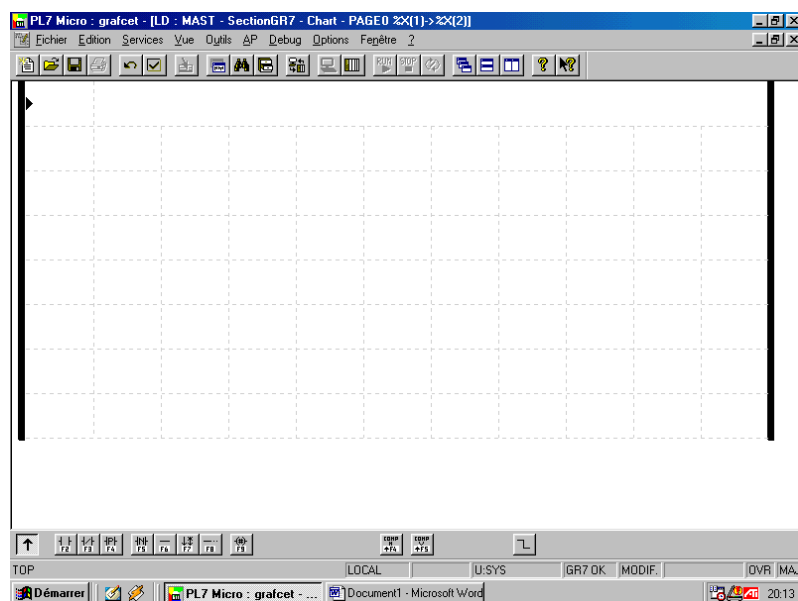
Un double clic sur « post » fera apparaître l'écran des propriétés du postérieur, sur lequel il suffit de valider en tapant entrée » pour arriver à l'écran de programmation du postérieur.



Réaliser votre programmation de la même façon que le « Chart », penser à valider à la fin de la programmation et sortez de votre écran.

B15 : La dernière étape de la programmation consiste à faire évoluer le GRAFCET. Pour se faire, il faut être en fenêtre B9. Double cliquer sur « Chart » et la fenêtre B12 apparaîtra.

Double cliquer sur la première transition (celle entre X1 et X2), alors l'écran suivant va s'afficher :



Réaliser votre programmation de la même façon que le « Chart », penser à valider à la fin de la programmation et sortez de votre écran. Renouveler l'opération pour chaque transition sans oublier de valider chaque page.

C) EXERCICE D'APPLICATION

A partir des GRAFCETS point de vue SYSTEME (annexe 1), mais surtout API (annexe 2)et des Entrées/Sorties API (annexe 3) on vous demande d'écrire sur papier libre le programme du PREHENSICC

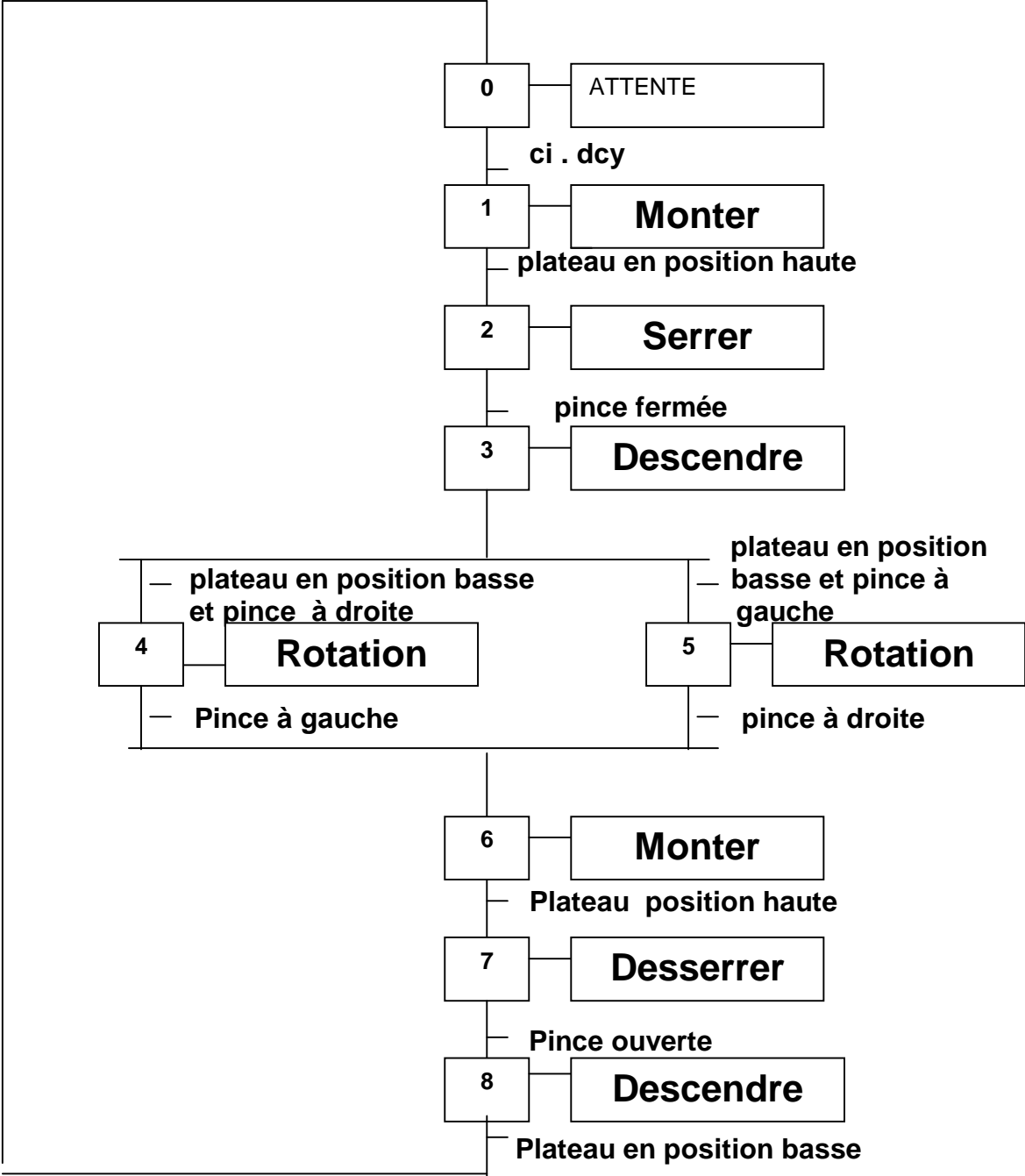


En respectant la procédure, réaliser la programmation sur PC et imprimer le programme. Pour se faire, il suffit dans le « Chart », « Prel » et « Post » d'ouvrir le fichier dans la barre d'état et de cliquer sur imprimer.



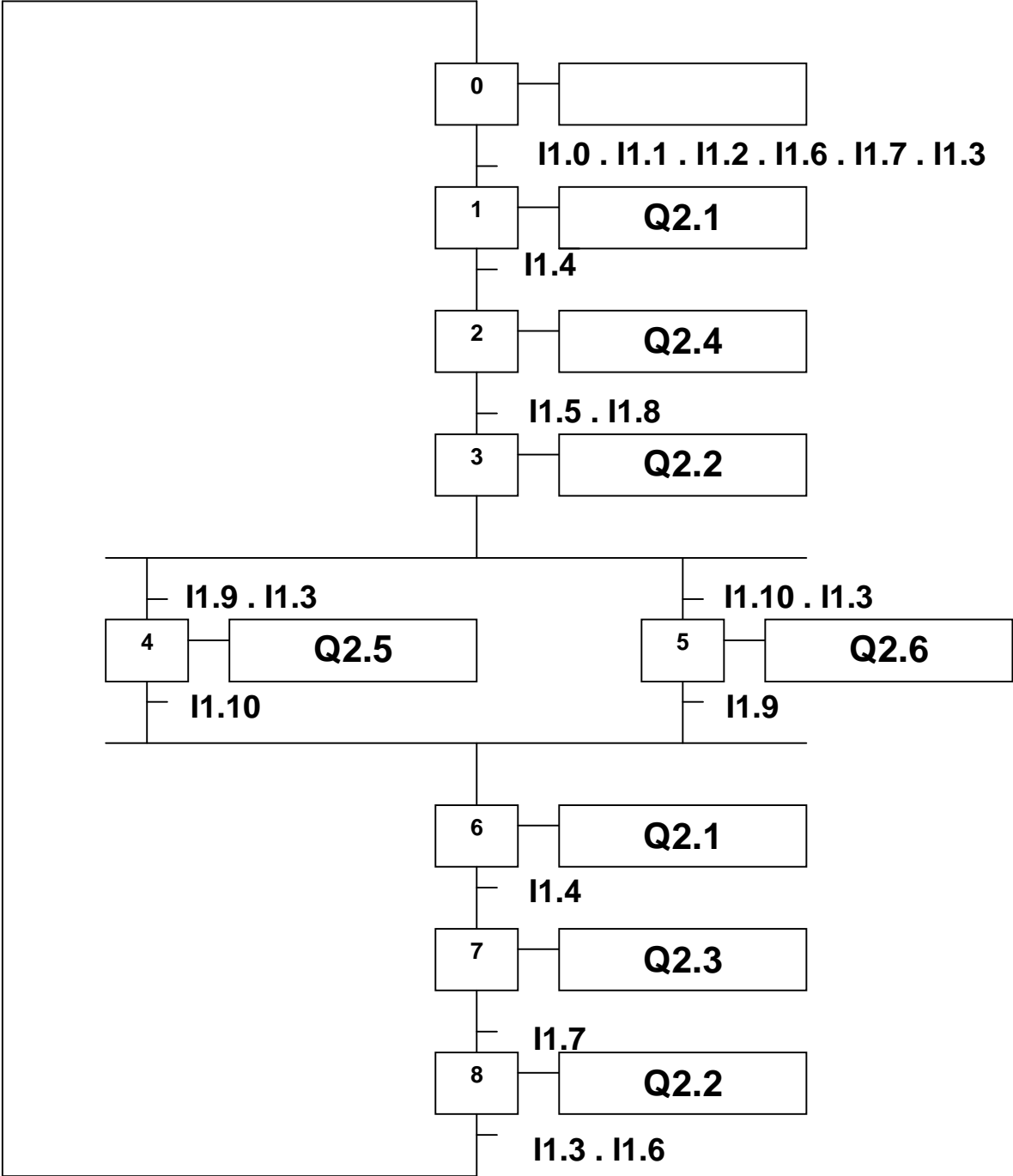
ANNEXE 1

GRAFCET POINT DE VUE SYSTEME



ANNEXE 2

GRAFNET POINT DE VUE A.P.I



ANNEXE 3**ENTREES A.P.I**

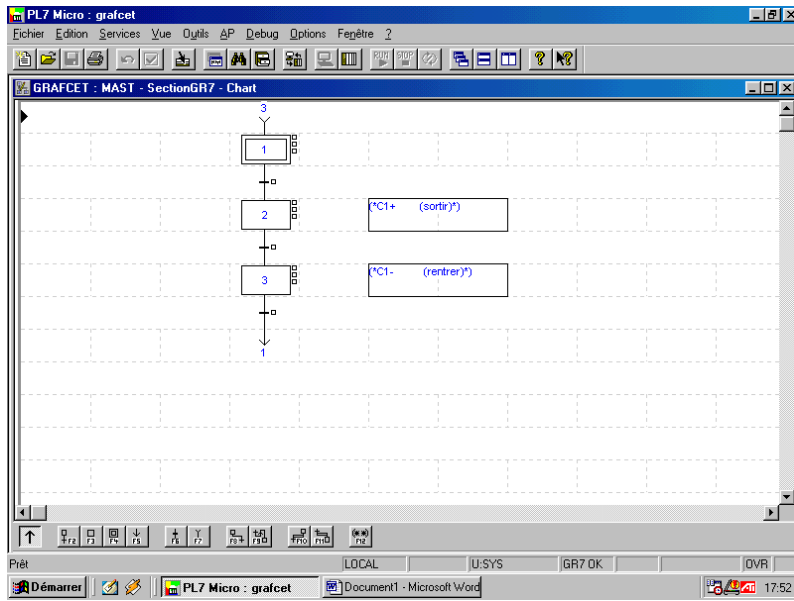
MNEMONIQUES	DESIGNATION	AFFECTATIONS A.P.I
Bp Run/stop	Marche arrêt a.p.i	I1.0
Dcy	Départ cycle	I1.1
S Auto	Automatique	I1.2
D12	Position basse	I1.3
Fc3	Accostage	I1.4
Fc9	présence carton dans pince	I1.5
D13	présence carton sur plateau	I1.6
D1	pince ouverte	I1.7
D2	pince fermée	I1.8
D3	0°	I1.9
D4	90°	I1.10
Init	initialisation	I1.11

SORTIES A.P.I

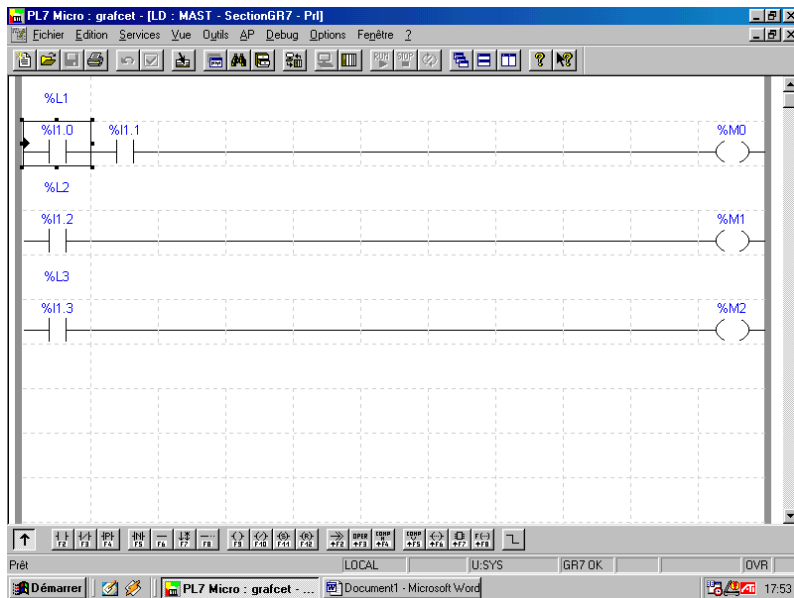
MNEMONIQUES	DESIGNATION	AFFECTATIONS A.P.I
Km2	montée plateau (M1)	Q2.1
Km1	descendre plateau (M2)	Q2.2
Ev2a	ouverture pince (C1+)	Q2.3
Ev2b	fermeture pince (C1-)	Q2.4
Ev1b	rotation gauche (C2+)	Q2.5
Ev1a	rotation droite (C2-)	Q2.6

CORRECTION

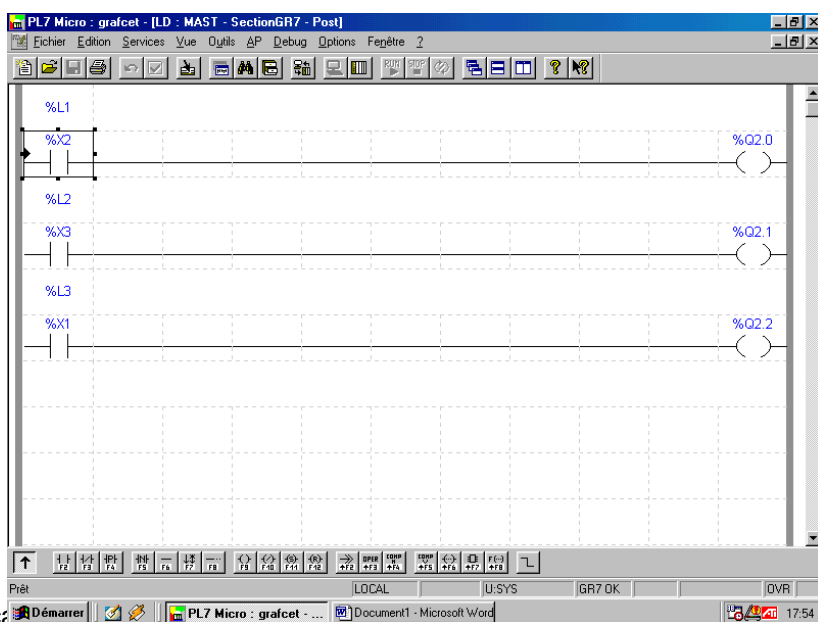
B12



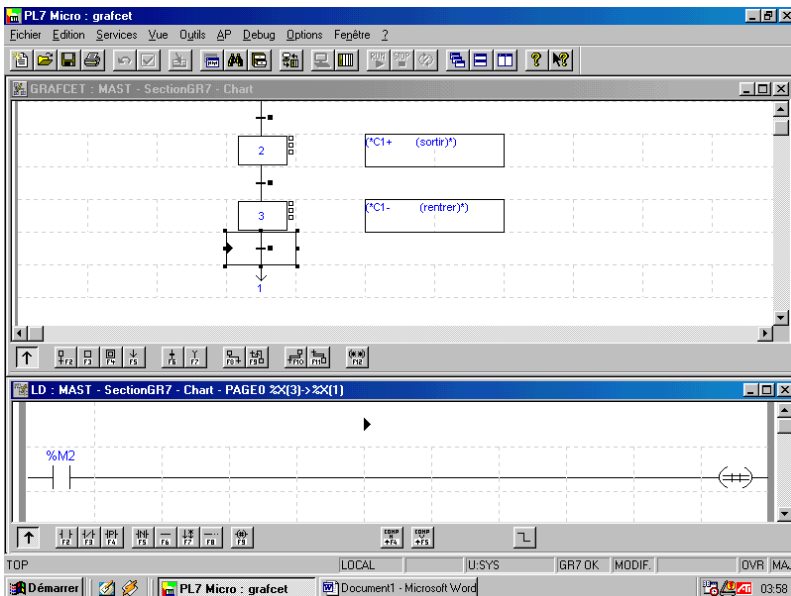
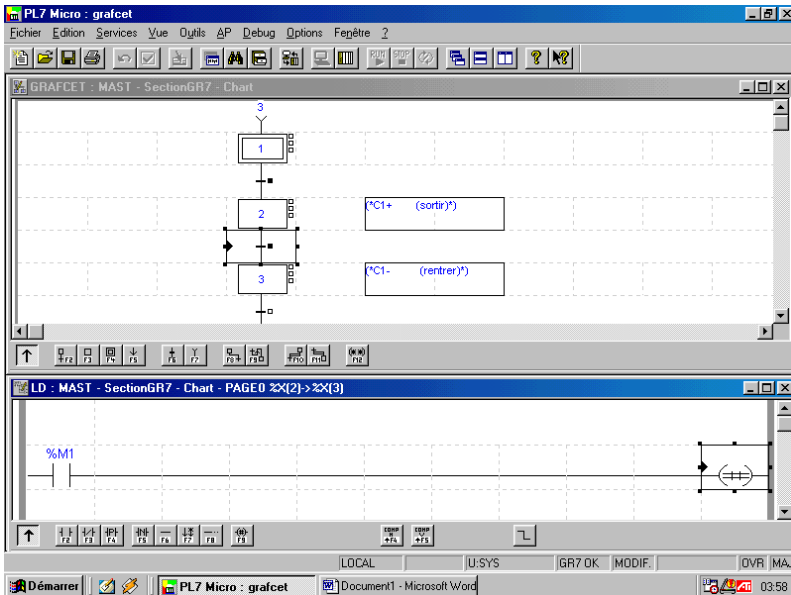
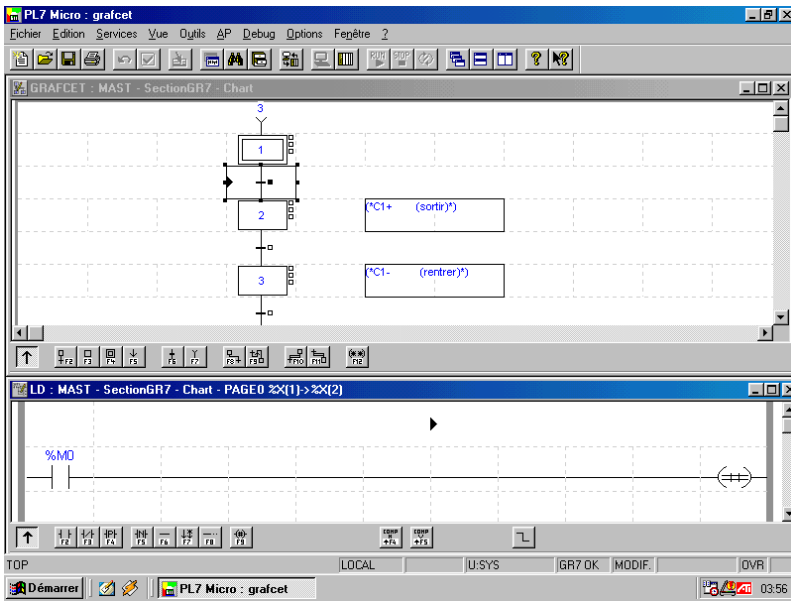
B13



B14

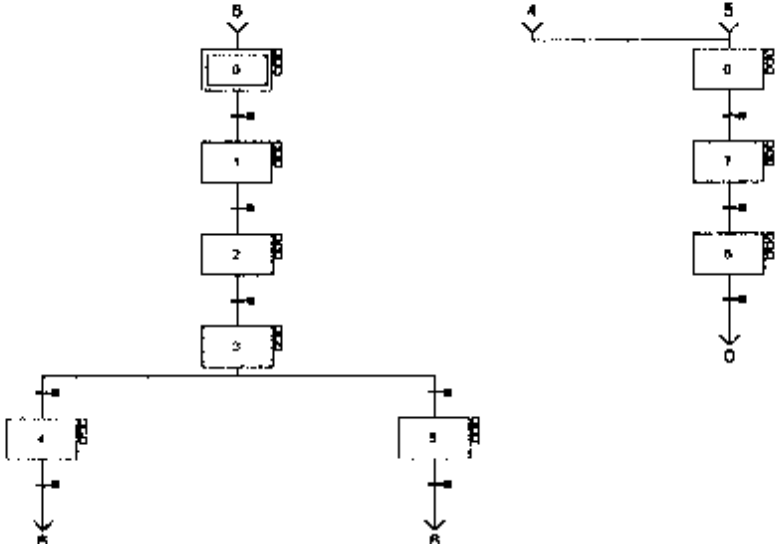


CORRECTION
B15

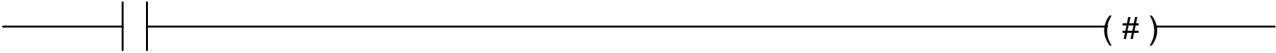


CORRECTION

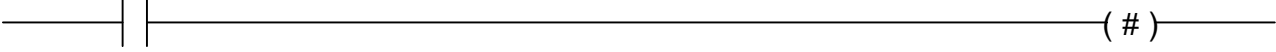
CHART



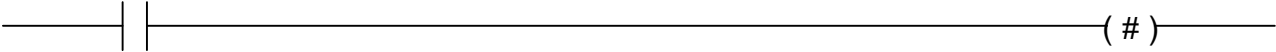
%X0 → %X1
M0



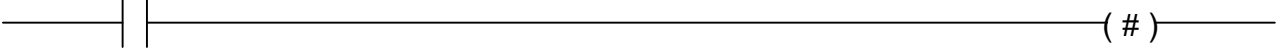
%X1 → %X2
M1



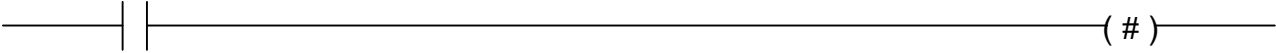
%X2 → %X3
M2



%X3 → %X4
M3

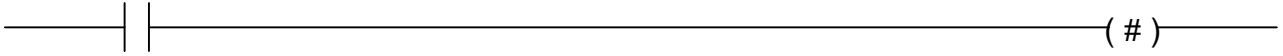


%X3 → %X5
M4

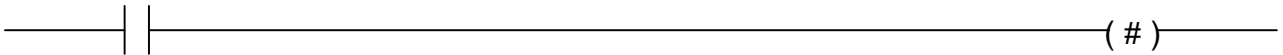


CORRECTION

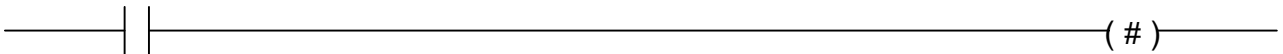
%X4 → %X6
M5



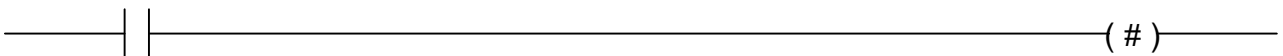
%X5 → %X6
M6



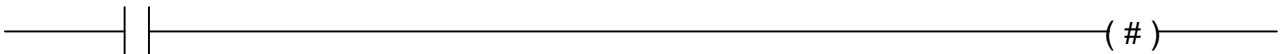
%X6 → %X7
M7



%X7 → %X8
M8

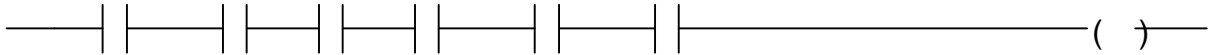


%X8 → %X0
M9

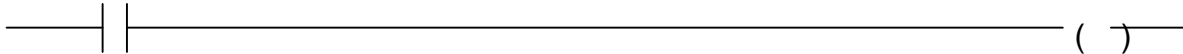


PRELIMINAIRES

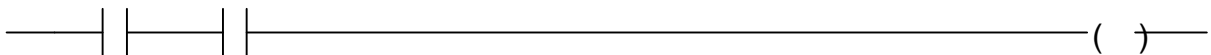
%L1
%l1.0 %l1.1 %l1.2 %l1.3 %l1.6 %l1.7 %M0



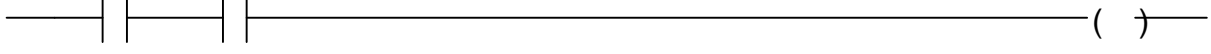
%L2
%l1.4 %M1



%L3
%l1.5 %l1.8 %M2



%L4
%l1.9 %l1.3 %M3



CORRECTION



POSTERIEUR



CORRECTION

