

## IX . EXERCICES DE SYNTHESE

### 1 - BACS DE TREMPE

#### 1 - 1. Présentation du sujet

Le poste de traitements de surface étudié est composé de 4 bacs de trempe, d'un dispositif automatique d'accrochage munis d'un système de reconnaissance de type de plateau, d'un dispositif automatique de décrochage et d'un portique sur lequel évolue un chariot.

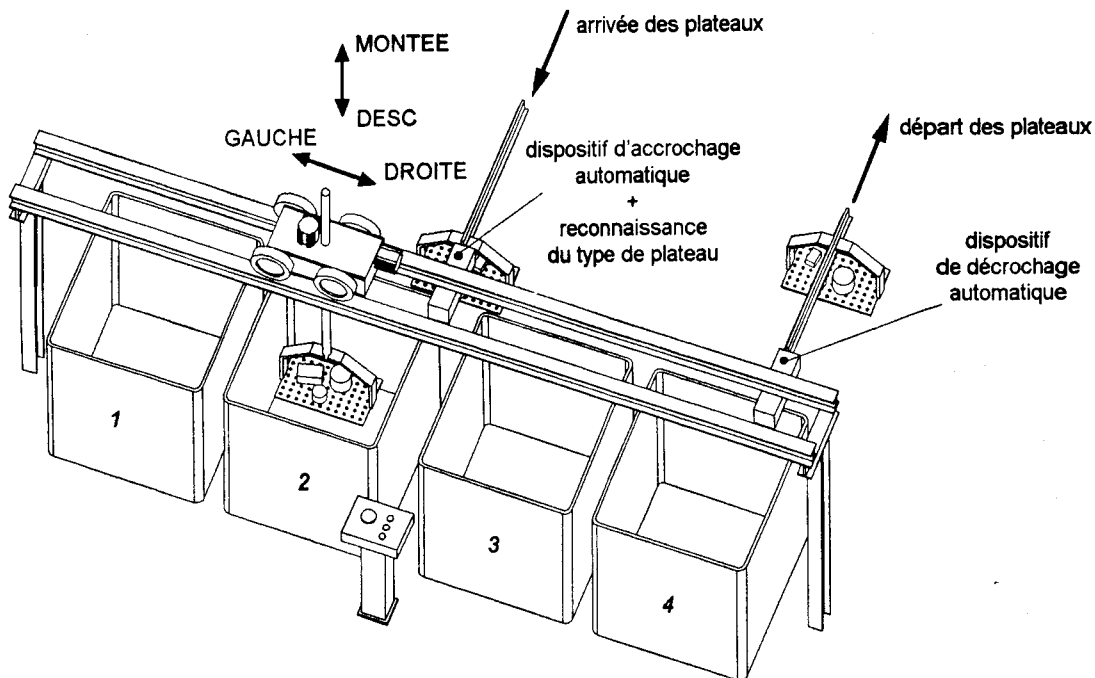


figure D-1

Les plateaux proviennent d'un atelier d'usinage et se présentent dans le désordre à l'entrée de ce poste. Chaque type de plateau nécessite un cycle de trempe spécifique, appelé *gamme*. En voici quelques exemples :

bac 2 → bac 3 → bac 3

bac 1

bac 4 → bac 2 → bac 3 → bac 4 → bac 1

Le nombre total de gammes réalisables dépend du nombre de bacs et du nombre maximum d'opérations. Pour 4 bacs et 8 opérations par exemple, la sélection de la gamme se fait parmi 87380 possibilités ( $4^1+4^2+4^3+4^4+4^5+4^6+4^7+4^8$ ).

*On souhaite :*

- pouvoir affecter à chaque type de palette l'une de ces gammes en la décrivant simplement,
- modifier la gamme alors que le processus est en mode exploitation,
- choisir la durée de trempe individuellement pour chaque opération.

## 1 - 2. Fonctionnement du système

A condition qu'un plateau soit présent à l'entrée du poste, un cycle est lancé :

- le plateau est identifié afin de déterminer la gamme associée,
- le plateau est pris en charge par le chariot,
- le chariot se déplace vers le premier bac demandé,
- le plateau plonge dans le bac,
- une temporisation est lancée,
- le délai écoulé, le plateau remonte et le chariot se positionne au-dessus du bac suivant, le cas échéant, pour effectuer une nouvelle trempe,
- lorsque toutes les opérations sont terminées, le plateau est décroché et quitte le poste,
- le chariot se présente immédiatement au dispositif d'accrochage,
- un nouveau cycle démarre si un plateau est présent à l'entrée du poste.

## 1 - 3. Enoncé du sujet

Les questions ne concernent que le mode de fonctionnement normal : la gestion des modes de marches et d'arrêts ainsi que la surveillance sont gérés à des niveaux hiérarchiques supérieurs.

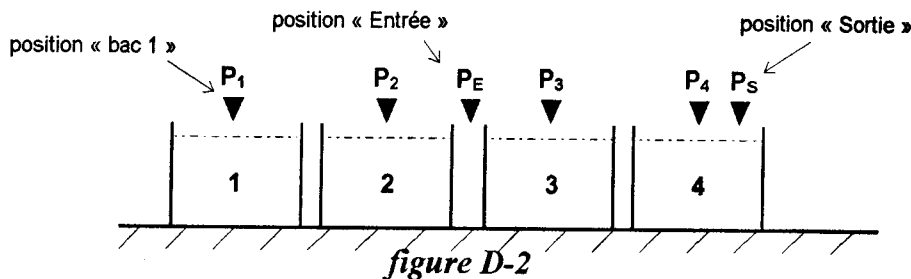


figure D-2

### a) Question 1 : analyse des fonctions

Faire l'inventaire des fonctions et les détailler. Une décomposition en plusieurs niveaux est nécessaire : on rappelle que l'analyse est suffisamment fine lorsque les *fonctions* peuvent être transcrites en *actions*.

Note : les dispositifs d'accrochage et de décrochage ne sont pas définis pour l'instant. L'étude détaillée de leur automatisme n'est pas demandée.

### b) Question 2 : épure du grafcet

Rechercher les séquences obligées et tracer une épure du grafcet de commande. Ne préciser aucune réceptivité pour l'instant. Note : le trop grand nombre de gammes possibles ne permet évidemment pas de les décrire toutes par autant de séquences de grafcet.

### c) Question 3 : structure de données

On souhaite mémoriser les cycles pour 12 types de plateaux. Les bacs sont au nombre de 4. Le nombre maximum d'opérations de trempe est limité à 8. Tracer la structure de données qui permet d'archiver les gammes. Toutes les données sont *résidentes dans l'automate* : celui-ci ne dispose pas de fonction intégrée de type recette, aussi faut-il développer entièrement la structure de données.

### d) Question 4 : interface utilisateur

L'accès direct à la structure de données par un personnel non autorisé est une source de danger. En effet, la maîtrise du procédé est indispensable pour paramétrer correctement le système. Proposer une interface utilisateur sûre et facile à mettre en oeuvre sachant qu'aucune connexion informatique ne lie le poste de traitement à une autre machine.

**e) Question 5 : réceptivités développées**

Ecrire les expressions des réceptivités qui permettent de gérer les déplacements du chariot entre 2 opérations consécutives : utiliser les noms des détecteurs de la figure D-2 et définir des symboles pour les mémoires utiles.

Peut-on simplifier ces expressions de manière algébrique ?

**f) Question 6 : gestion des données et équations simplifiées**

On constate qu'au lancement d'un cycle de trempe, seules les données relatives au type de plateau traité sont nécessaires. Par ailleurs, seules certaines données sont utiles à chaque opération de trempe. Tracer la *mémoire de travail* qui découle de cette constatation.

Reconsidérer les expressions des réceptivités et compléter le grafcet par des étapes qui commandent les différentes fonctions de gestion des données.

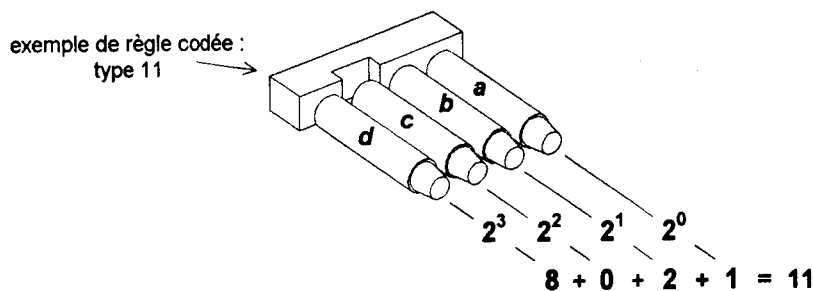
Tracer ensuite le grafcet complet.

**g) Programmation**

Si le lecteur ne dispose pas du manuel de référence du constructeur, il se reportera directement au corrigé.

**- Reconnaissance du type de plateau**

Les plateaux sont identifiés grâce à un jeu de 4 détecteurs (solidaires du dispositif d'accrochage) combiné à des règles codées (solidaires des plateaux) :



Les détecteurs sont reliés

- aux entrées E2.2, E2.3, E2.4 et E2.5 si l'automate est de marque Siemens, série S5,
- ou aux entrées %I2.2, %I2.3, %I2.4 et %I2.5 s'il s'agit d'un automate Schneider-Télé-mécanique TSX37/57.

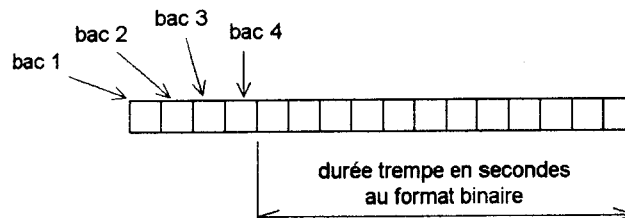
Proposer des solutions pour mémoriser le type de plateau en fonction des états des détecteurs.

**- Paramétrage de la durée de trempe**

Cette question concerne la contrainte suivante, formulée dans le cahier des charges initial :

*La durée de trempe doit pouvoir être spécifiée individuellement pour chaque opération.*

On précise que les données relatives à une opération de trempe sont transmises à l'automate sous le format suivant :



Proposer des solutions pour gérer la durée des trempes.

## 1 - 4. Corrigé et commentaires

### a ) Analyse des fonctions (Q.1)

Un premier niveau d'analyse permet de dégager les fonctions inventoriées à la figure D-3 : cette façon de les représenter évite de suggérer un quelconque aspect séquentiel dès ce stade de l'étude.

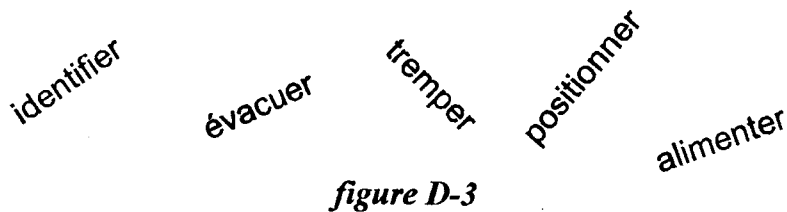


figure D-3

#### **Fonction : identifier**

Il s'agit de déterminer le type de plateau en fonction de la morphologie de la règle codée.

#### **Fonction : évacuer**

Quel que soit l'endroit où se trouve le chariot, il se positionne devant le dispositif de décrochage. Celui-ci a lieu. Le chariot se positionne ensuite devant le dispositif d'accrochage.

#### **Fonction : tremper**

Quel que soit le bac où se trouve le chariot, le plateau descend. Dès qu'il est arrivé en position basse, une temporisation est lancée : sa durée dépend de la valeur prévue pour la trempe en cours. Une fois le délai écoulé, le plateau remonte. Arrivé en position haute, la fonction de trempe est terminée.

#### **Fonction : positionner**

Quel que soit le bac où se trouve le chariot, il se positionne au-dessus du bac requis pour la trempe de l'opération suivante. Si le prochain bac est le même que celui où une trempe vient d'avoir lieu, aucun déplacement n'est associé à cette fonction.

#### **Fonction : alimenter**

Cette fonction consiste seulement à accrocher le plateau au bras du chariot.

#### **- Décomposition supplémentaire**

- La fonction *identifier* est étudiée en détail à la question 7.
- Les fonctions *évacuer*, *tremper* et *alimenter* feront l'objet d'un grafcet de tâche chacune. Il ne subsiste aucune difficulté pour mettre en place les actions.
- Par contre, aucune action ne peut être associée à la fonction *positionner* au niveau d'analyse actuel. En effet, l'exécution de cette fonction nécessite 3 comportements différents :
  - le chariot se déplace vers la droite,
  - le chariot se déplace vers la gauche,
  - le chariot ne se déplace pas.

L'inventaire des fonctions à un niveau d'analyse supplémentaire est donc le suivant :

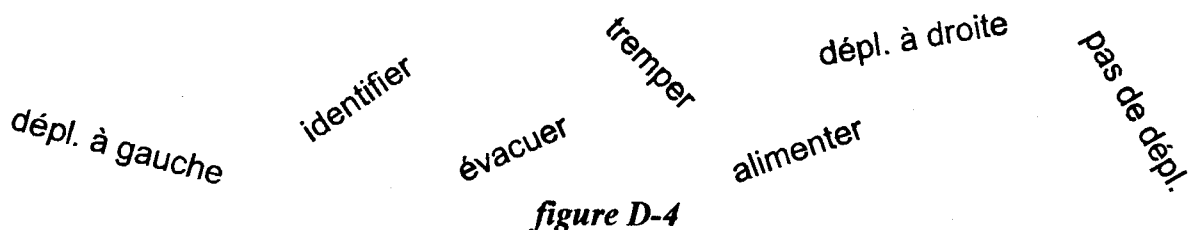


figure D-4

### b ) Epure du grafcet (Q.2)

L'organisation chronologique des fonctions peut être étudiée très librement dans un premier temps en s'aidant d'un diagramme :

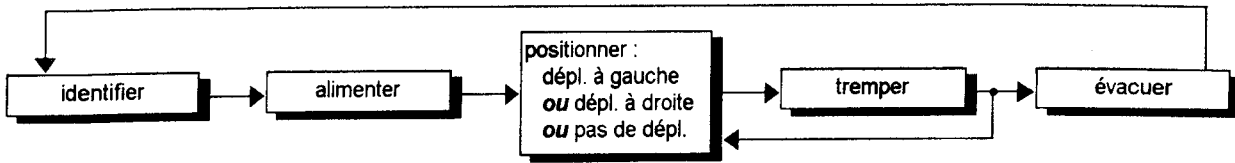


figure D-5

Un grand nombre d'opérations peut être géré grâce à cette structure générale par itérations. L'épure du graficet en est déduite aisément (figure D-6).

**c ) Structure de données (Q.3)**

La structure de données initiales doit être conçue afin qu'elle puisse mémoriser toutes les informations nécessaires à la gestion des plateaux. On recense l'ensemble des paramètres :

- on souhaite traiter 12 types de plateaux,
- un traitement nécessite un maximum de 8 opérations, chacune pouvant s'effectuer dans l'un des 4 bacs.

L'affectation d'un bac à chaque opération peut être réalisée de deux manières, soit en utilisant un quartet dont chaque bit est l'image d'un bac, soit un octet (ou un mot) qui contient le numéro de bac. La figure D-7 propose la structure sous les deux représentations.

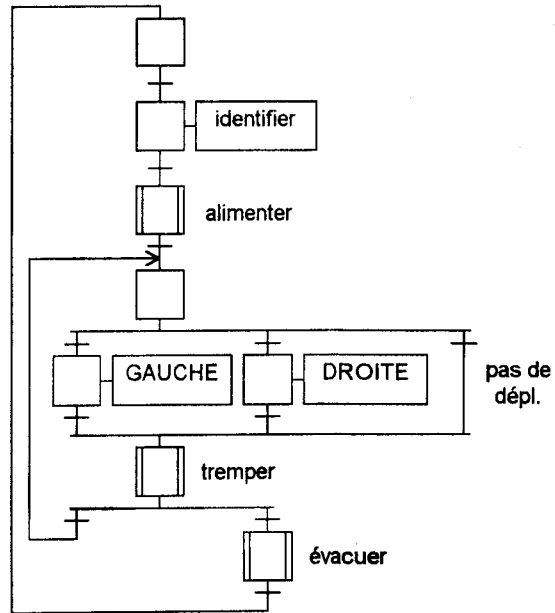


figure D-6

**utilisation de 4 bits pour chaque opération**

**utilisation d'1 octet pour chaque opération**

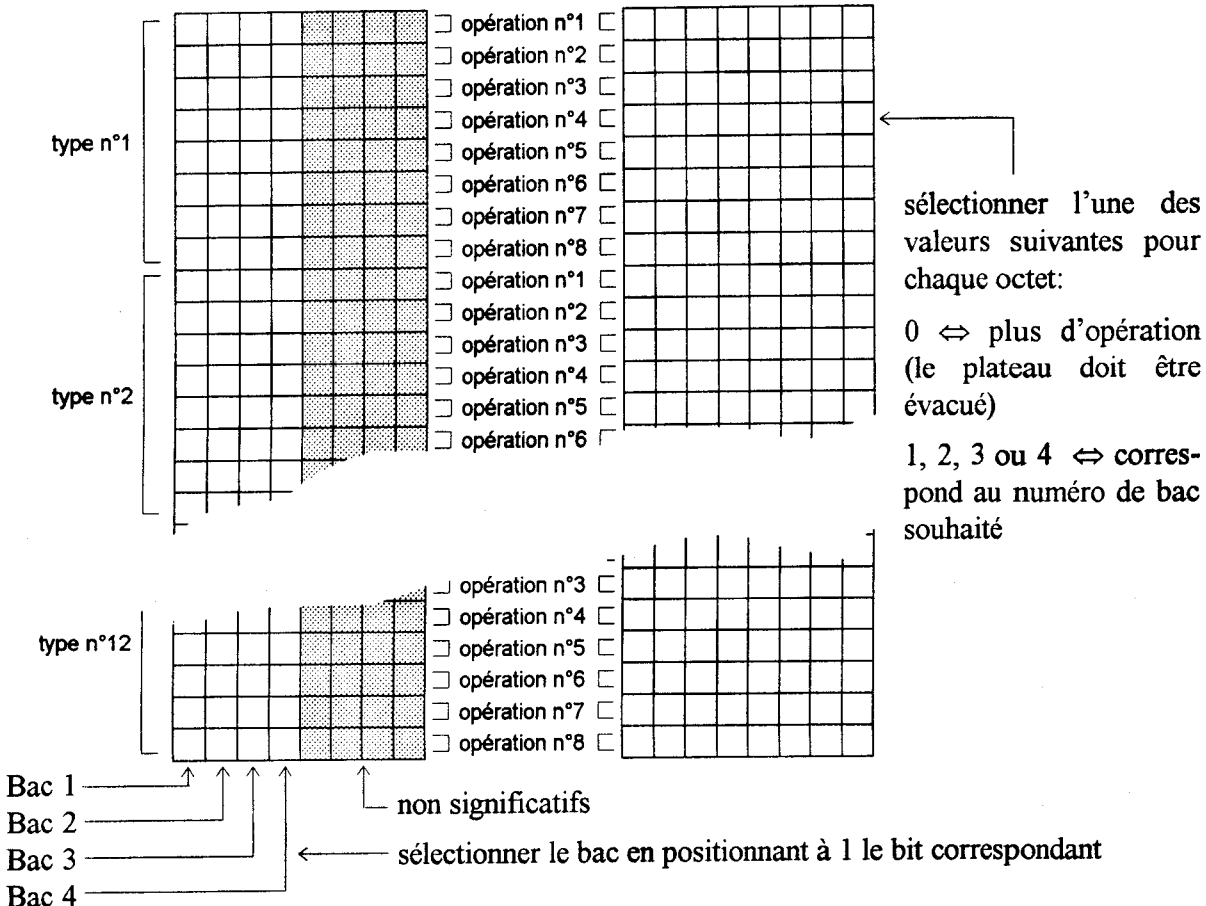


figure D-7

La figure D-8 montre l'exemple de la gamme : bac 3 → bac 4 → bac 4 → bac 1 → bac 2

**d ) Interface utilisateur (Q.4)**

Le personnel de conduite de ce poste n'est pas forcément qualifié pour configurer les données sous la forme présentée : une interface plus conviviale est nécessaire. Il est judicieux de prévoir un mot de passe pour réserver l'accès de cette configuration au seul personnel compétent. Le poste de traitement n'étant pas connecté à une plus vaste structure, un terminal opérateur est indiqué pour la saisie des gammes. Il est également chargé de la mise en forme automatique des données et de leur mise à jour dans la mémoire de l'automate. L'illustration ci-contre reprend le principe du terminal déjà décrit au chapitre VIII (page 137, figure C-24).

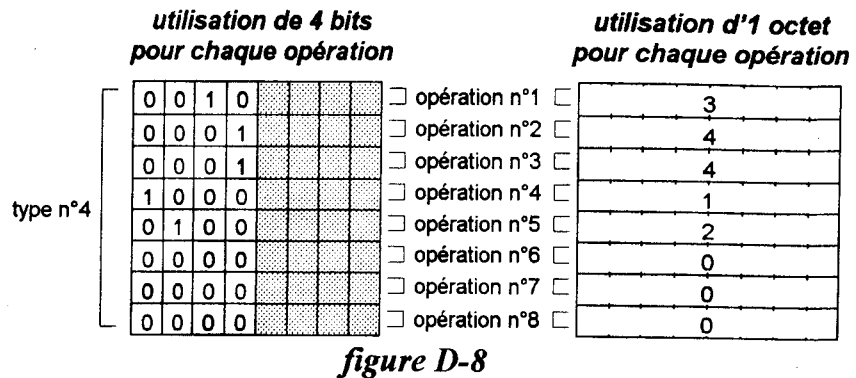


figure D-8

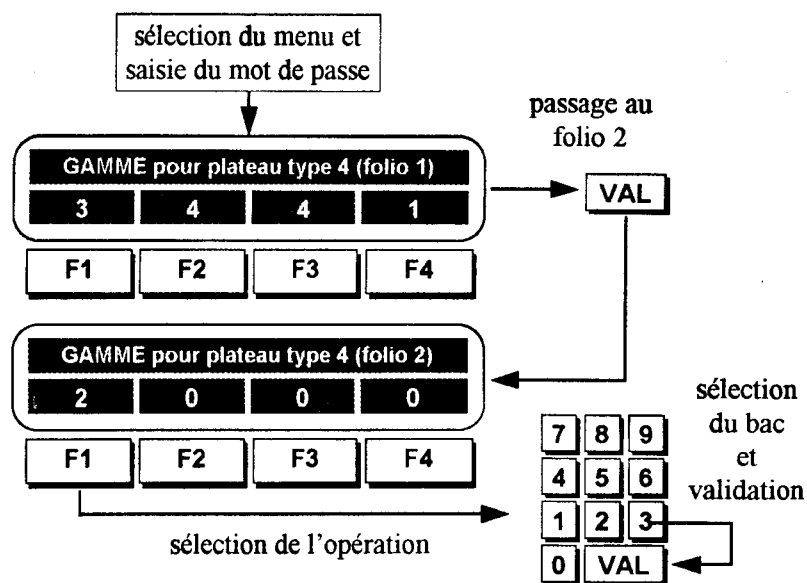


figure D-9

Soit à nouveau la gamme de la figure D-8 : on souhaite modifier la cinquième opération en remplaçant le bac 2 par le bac 3. La figure D-9 indique la manipulation à effectuer pour répondre à ce besoin. Si c'est l'une des quatre premières opérations qui doit être modifiée, l'appui sur la touche de fonction correspondante permet la sélection du bac lorsque le folio 1 est affiché.

**e ) Réceptivités développées (Q.5)**

Le grafctet de la figure D-6 ne décrit que les possibilités d'évolution du système mais ne peut pas, dans son état actuel, gérer de cycle complet relativement à une gamme. Les réceptivités jouent le rôle de règles qui permettront la réalisation effective d'un tel cycle.

On renvoie le lecteur à la page 80. Les réceptivités sont déterminées en réponse aux interrogations suivantes :

- ⇒ quelles informations permettent le départ des tâches,
- ⇒ quelles informations permettent leur arrêt.

En premier lieu, un certain nombre de réceptivités sont d'écriture immédiate :

- la fonction *identifier* démarre lorsqu'un plateau est présent et s'arrête dès que le type de plateau est enregistré, ce qui provoque la tâche d'alimentation,
- en aval des étapes qui font appel à des tâches, les réceptivités comportent l'étape de compte-rendu de fin de tâche.

Il reste à définir les conditions qui permettent de sélectionner le sens de déplacement (ou l'absence de déplacement) du chariot. La sélection en aval de l'étape qui appelle la tâche de *trempe* se fait selon le numéro de bac de la prochaine opération : si ce numéro est nul, le plateau doit être évacué.

Le sens de déplacement dépend de deux paramètres :

- l'endroit où doit se rendre le chariot pour effectuer l'opération suivante  
☞ *celui-ci est matérialisé par la donnée correspondante*
- l'endroit où se trouve le chariot à l'instant considéré  
☞ *celui-ci est matérialisé par le détecteur correspondant*

Le recensement de tous les cas de figure permet l'écriture de l'expression générique des réceptivités.

Plusieurs possibilités sont à l'origine du déplacement vers la gauche :

- si la prochaine opération a lieu dans le bac 1  
et que le chariot se trouve au-dessus du bac 2, ou du bac 3, ou du bac 4, ou à l'entrée du poste
- ou si la prochaine opération a lieu dans le bac 2  
et que le chariot se trouve au-dessus du bac 3, ou du bac 4, ou à l'entrée du poste
- ou si la prochaine opération a lieu dans le bac 3  
et que le chariot se trouve au-dessus du bac 4

Plusieurs possibilités sont à l'origine du déplacement vers la droite :

- si la prochaine opération a lieu dans le bac 2  
et que le chariot se trouve au-dessus du bac 1
- ou si la prochaine opération a lieu dans le bac 3  
et que le chariot se trouve au-dessus du bac 1, ou du bac 2, ou à l'entrée du poste
- ou si la prochaine opération a lieu dans le bac 4  
et que le chariot se trouve au-dessus du bac 1, ou du bac 2, ou du bac 3, ou à l'entrée du poste

Il n'y a pas de déplacement lorsque :

- la prochaine opération a lieu dans le bac (n)  
et que le chariot se trouve au-dessus du bac (n)

On choisit la structure de données par bits pour la suite de l'étude. Compte tenu des différents type de plateaux ainsi que des opérations successives, la réceptivité en amont de  $X_{\text{GAUCHE}}$  s'écrit donc :

$$\begin{aligned}
 R_{\text{GAUCHE}} = & \{ \text{type}=1 \cdot \text{op}=1 \cdot [M_{111} \cdot (P_2+P_3+P_4+P_E)] + [M_{211} \cdot (P_3+P_4+P_E)] + [M_{311} \cdot P_4] \} \\
 & + \{ \text{type}=1 \cdot \text{op}=2 \cdot [M_{112} \cdot (P_2+P_3+P_4+P_E)] + [M_{212} \cdot (P_3+P_4+P_E)] + [M_{313} \cdot P_4] \} \\
 & \text{etc...} \\
 & + \{ \text{type}=1 \cdot \text{op}=8 \cdot [M_{118} \cdot (P_2+P_3+P_4+P_E)] + [M_{218} \cdot (P_3+P_4+P_E)] + [M_{318} \cdot P_4] \} \\
 & \quad + \{ \text{type}=2 \cdot \text{op}=1 \cdot [M_{121} \cdot (P_2+P_3+P_4+P_E)] + [M_{221} \cdot (P_3+P_4+P_E)] + [M_{321} \cdot P_4] \} \\
 & \quad + \{ \text{type}=2 \cdot \text{op}=2 \cdot [M_{122} \cdot (P_2+P_3+P_4+P_E)] + [M_{222} \cdot (P_3+P_4+P_E)] + [M_{322} \cdot P_4] \} \\
 & \quad \text{etc...} \\
 & \quad + \{ \text{type}=2 \cdot \text{op}=8 \cdot [M_{128} \cdot (P_2+P_3+P_4+P_E)] + [M_{228} \cdot (P_3+P_4+P_E)] + [M_{328} \cdot P_4] \} \\
 & \quad + \{ \text{type}=3 \dots
 \end{aligned}$$

ainsi de suite pour les 12 types de plateaux

Avec les mémoires  $M_{ijk}$  :

$i$  varie de 1 à 4 : nom du bac

$j$  varie de 1 à 12 : type de plateau

$k$  varie de 1 à 8 : numéro d'opération

extrait de la structure de données  
pour les deux premières opérations du type 1

$M_{111}$	$M_{211}$	$M_{311}$	$M_{411}$				
$M_{112}$	$M_{212}$	$M_{312}$	$M_{412}$				

Le choix de la structure de données par octets conduit à une expression légèrement différente, mais tout aussi dense. Il n'est pas possible de simplifier algébriquement cette expression car toutes les mémoires sont indépendantes.

### f) Gestion des données et réceptivités simplifiées (Q.6)

A chaque opération, seuls sont nécessaires les 4 bits qui appartiennent à la fois au plateau en cours de traitement et à l'opération à réaliser :

- les plateaux se présentent dans un ordre aléatoire
  - ☞ *la zone mémoire à isoler dépend du type de plateau à chaque cycle complet*
- les opérations sont systématiquement réalisées dans l'ordre 1 à 8
  - ☞ *une file d'attente permet de dérouler les opérations au fur et à mesure*

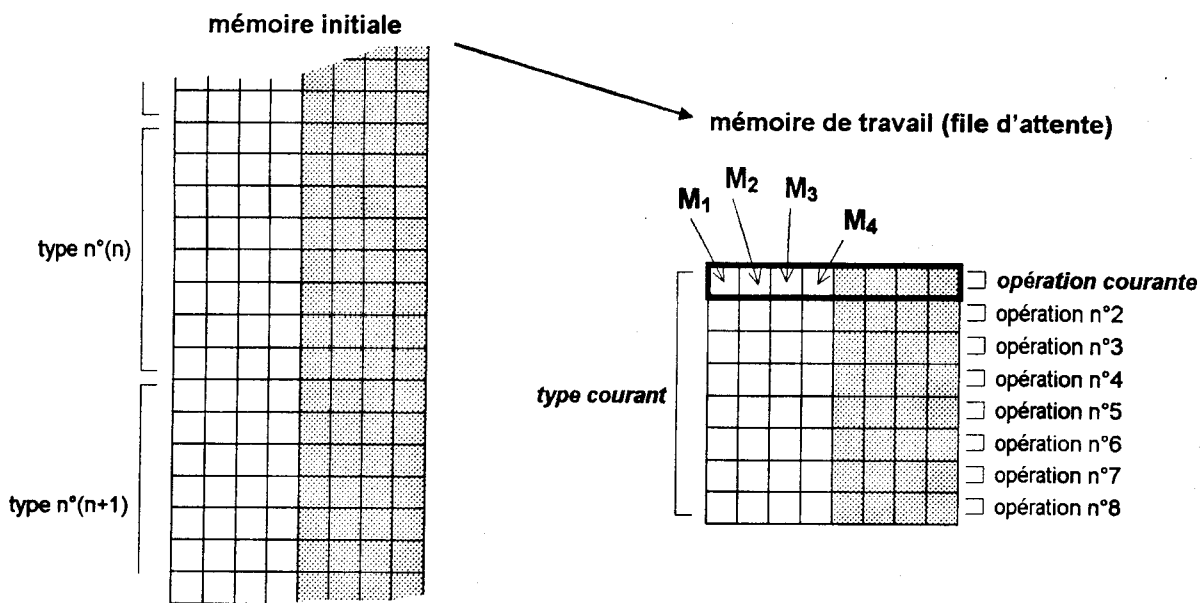


figure D-10

Le grafacet est complété par les fonctions suivantes :

- dupliquer les données relatives au type de plateau immédiatement après sa reconnaissance,
- à chaque nouvelle opération, décaler la file d'attente d'un pas vers le haut et enregistrer la valeur 0 dans l'octet du bas (pour purger les données pour le cas où 8 opérations sont demandées).

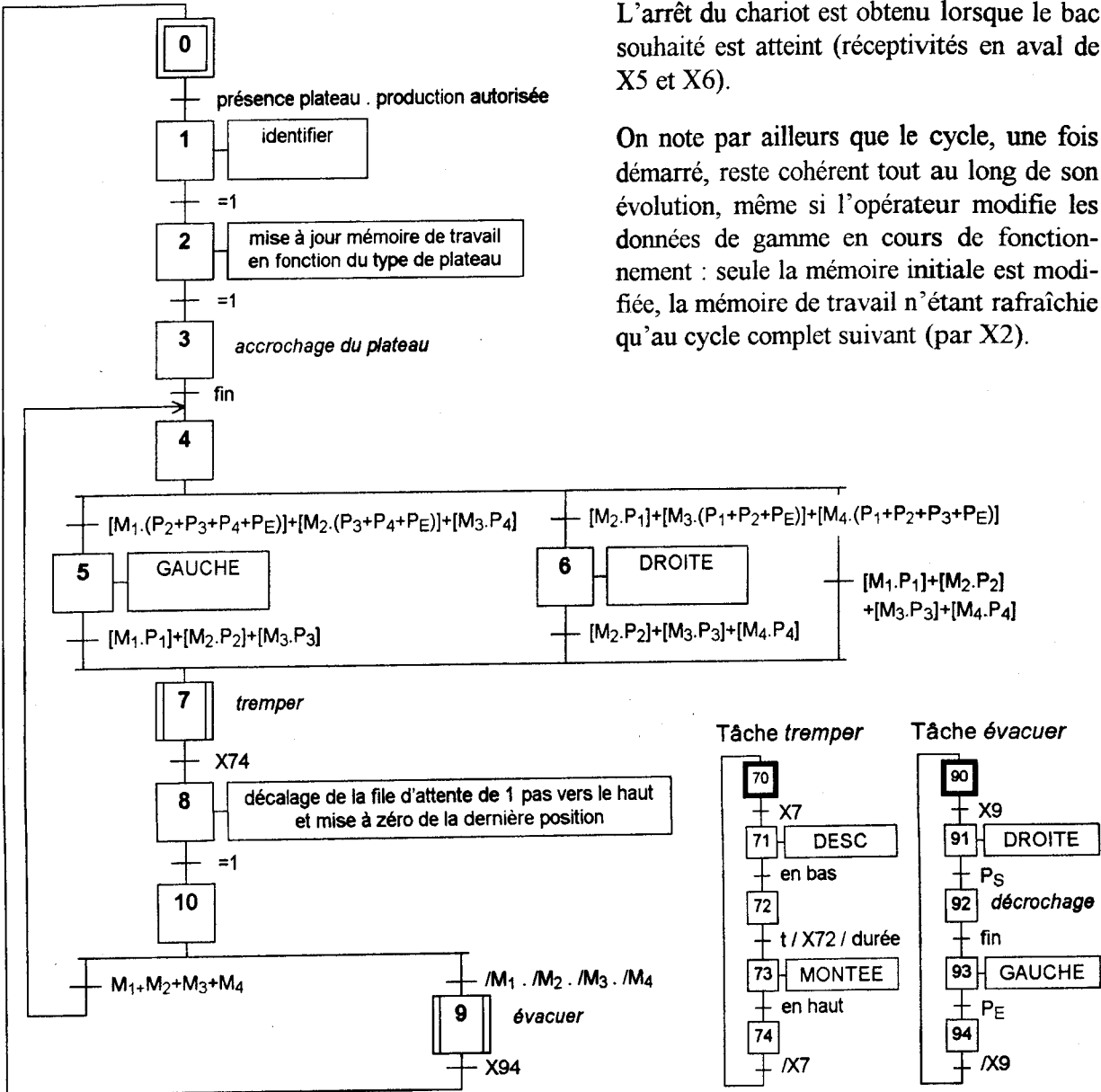
La réduction du nombre de mémoires à tester simplifie considérablement les réceptivités. Leur développement complet conduit aux expressions suivantes :

$$R_{\text{GAUCHE}} = [M_1 \cdot (P_2 + P_3 + P_4 + P_E)] + [M_2 \cdot (P_3 + P_4 + P_E)] + [M_3 \cdot P_4]$$

$$R_{\text{DROITE}} = [M_2 \cdot P_1] + [M_3 \cdot (P_1 + P_2 + P_E)] + [M_4 \cdot (P_1 + P_2 + P_3 + P_E)]$$

$$R_{\text{pas de dépl.}} = [M_1 \cdot P_1] + [M_2 \cdot P_2] + [M_3 \cdot P_3] + [M_4 \cdot P_4]$$





L'arrêt du chariot est obtenu lorsque le bac souhaité est atteint (réceptivités en aval de X5 et X6).

On note par ailleurs que le cycle, une fois démarré, reste cohérent tout au long de son évolution, même si l'opérateur modifie les données de gamme en cours de fonctionnement : seule la mémoire initiale est modifiée, la mémoire de travail n'étant rafraîchie qu'au cycle complet suivant (par X2).

figure D-11

g) Programmation

- Reconnaissance du type de plateau

L'utilisation d'instructions de base n'est pas très élégante : pour les 12 types de plateaux, il faut en effet tester quasiment toutes les combinaisons des 4 entrées qui correspondent aux détecteurs. L'organigramme général est représenté ci-dessous :

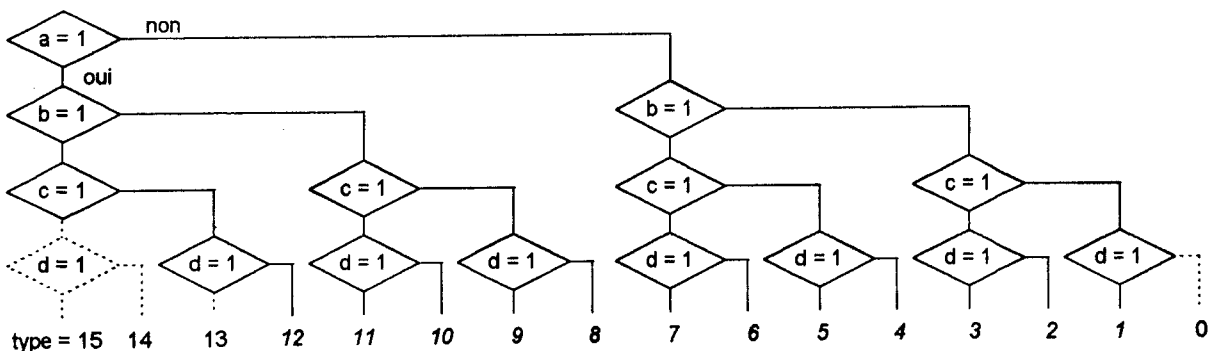


figure D-12

Le programme peut avantageusement être simplifié en utilisant des *instructions avancées*. Le principe consiste à lire le module d'entrées complet sur lequel sont reliés les détecteurs et de déduire directement le type de plateau à partir du profil binaire obtenu.

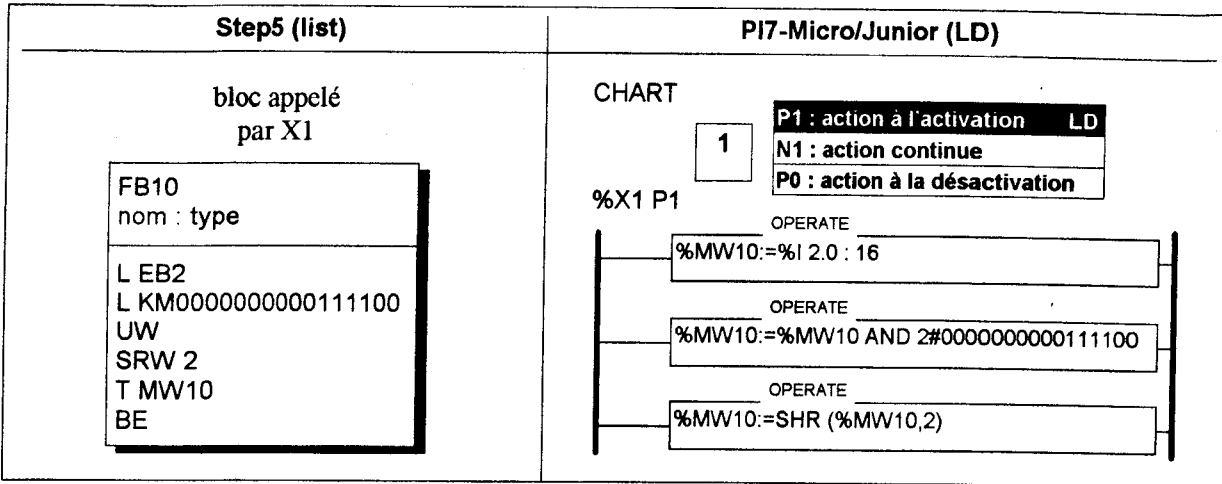


figure D-13

La procédure est identique pour les deux exemples de programmes ci-dessus. Elle est la suivante :

- 1) lecture de l'octet ou mot associé au module d'entrées : EB2 ou les 16 bits consécutifs à %I2.0
- 2) masquage des informations indésirables et filtrage des informations significatives (1 pour les données significatives, 0 pour les données non significatives) réalisés par la fonction ET SUR MOT (fonction logique effectuée bit par bit) : UW ou AND
- 3) décalage de 2 bits vers la droite pour faire correspondre les poids des bits : SRW ou SHR
- 4) et mémorisation dans le mot d'indice 10 (MW10 ou %MW10)

En fonction de l'état réel des capteurs (détecteurs de type de plateau et autres capteurs liés au processus) le module d'entrées est par exemple dans l'état de EB2 ci-dessous au moment de la réalisation de cette opération (exemple en Step5) :

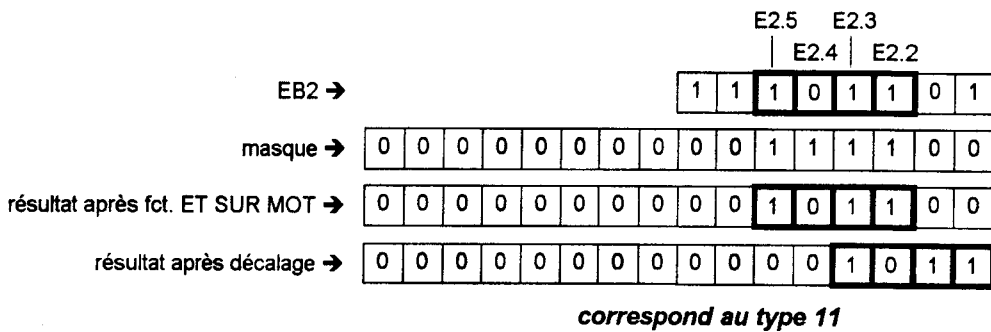
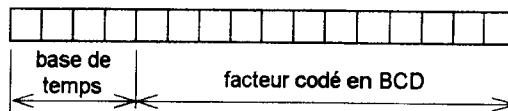


figure D-14

**- Paramétrage de la durée de tremp (en Step 5)**

En Step5, le format de temporisation est le suivant :



- la base de temps est codée sur les 4 bits de gauche :
  - 0000 = 1/100° sec.
  - 0001 = 1/10° sec.
  - 0010 = 1 sec.
  - 0011 = 10 sec.
- le facteur est codé en BCD sur 3 quartets, sa valeur maximale est donc 999
- la durée réelle est : facteur x base de temps (la précision de la tempo équivaut à la base de temps)

Différentes manipulations sont donc nécessaires :

- 1) lecture du mot associé à l'opération de trempé à réaliser
- 2) isolement et transcodage du facteur de temps
  
- 3) ajout de la base de temps
  
- 4) départ de la temporisation sur activation de X72

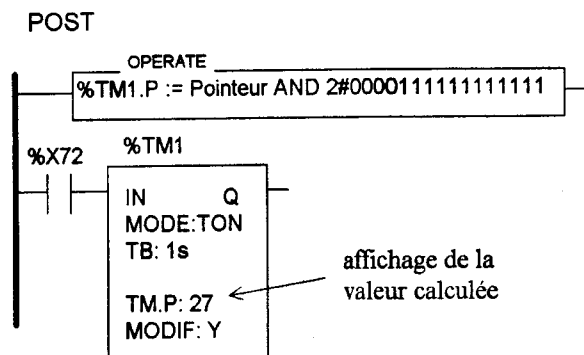
```

L -pointeur
L KM0000111111111111
UW
T MW50
SPA FB241
L MW50
L KM0010000000000000
OW
T MW50
U M7.2
L MW50
SE T1
  
```

Le premier masque, en combinaison avec la fonction ET SUR MOT, permet d'isoler le facteur de temps contenu dans le pointeur. FB241 est un bloc fonctionnel intégré qui permet le transcodage de valeurs décimales en codes BCD : il suffit de paramétrer le mot d'entrée et ensuite un mot + un octet de sortie lors de la saisie du programme (MW50 est le mot d'entrée dans cet exemple et étant donné que la valeur est limitée à 999, seulement le mot de sortie est significatif : on choisit MW50). Le deuxième masque, en combinaison avec la fonction OU SUR MOT, permet d'ajouter la base de temps (1 sec. dans cet exemple).

#### - Paramétrage de la durée de trempé (en PI-7)

La temporisation est réalisée par un bloc fonction, par exemple %TM1, qu'il suffit de paramétrer : TB est la base de temps, directement exprimée dans sa valeur, TM.P est le facteur exprimé en décimal. Celui-ci peut être modifié par configuration (avec la console) ou par programme (ce qui est l'objectif souhaité). Il faut également isoler au préalable le facteur de temps contenu dans le pointeur. On utilisera ensuite %TM1.Q dans la réceptivité X72 → X73.



Dans les deux cas, tous les octets des mémoires initiales et de travail sont remplacés par des mots.

## 2 - PREPARATION DE MEDICAMENTS

### 2 - 1. Présentation du sujet

Le pharmacien d'un hôpital est responsable de la préparation des boîtes de médicaments prêtes à l'emploi qui répondent aux traitements prescrits par les médecins. Le patient est guidé jour par jour :

- la première rangée de la boîte contient les médicaments du premier jour et ainsi de suite,
- les 7 rangées d'une boîte correspondent à une semaine,
- aucun traitement n'excède 3 semaines,
- un lot comporte toujours 3 boîtes au cas où la deuxième semaine par exemple ne nécessite aucun médicament (exemple du traitement n°2 ci-dessous).

Un certain nombre de traitements standards sont fréquents. Des traitements spécifiques peuvent être créés au fur et à mesure du besoin. Pour garantir la traçabilité des interventions médicales, il est indispensable de mémoriser le traitement prescrit à chaque patient.

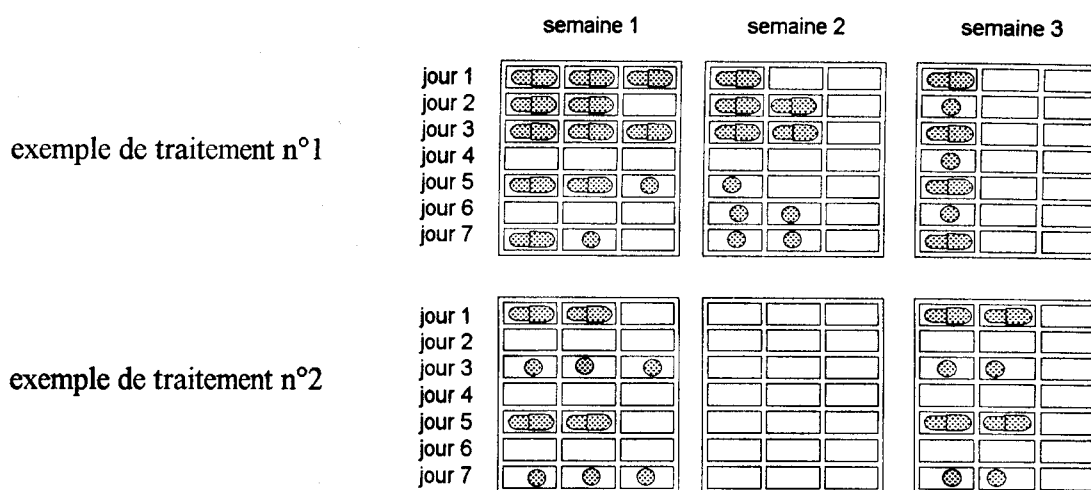


figure D-15

#### - Reconditionnement préalable

Jusqu'à 120 références, limitées à des gellules, pillules et comprimés, interviennent dans les traitements. Pour que la préparation des boîtes puisse être réalisée par un système automatique, il est nécessaire que les médicaments soient au préalable reconditionnés dans des emballages standards (ce poste de reconditionnement n'est pas étudié). Ces nouveaux emballages se présentent sous forme de rouleaux marqués d'un code-barre qui sont ensuite installés sur un magasin. Chaque numéro de rouleau a sa place réservée dans le magasin.

### 2 - 2. Fonctionnement du système (figure D-16)

Les médicaments sont prélevés individuellement par le dispensateur au fur et à mesure du besoin. A chaque fois, le code-barre imprimé sur le rouleau est comparé au numéro de médicament demandé : en cas d'erreur, l'opération est interrompue et une alarme est déclenchée (si le rouleau est placé au mauvais emplacement par exemple).

*Selon le traitement à préparer, le bras manipulateur saisit un à un les médicaments dans le dispensateur et les dépose dans les boîtes aux emplacements adéquats.*

Le bouchage hermétique des boîtes et la pose des étiquettes d'identification se font ensuite sur un poste indépendant (non étudié).

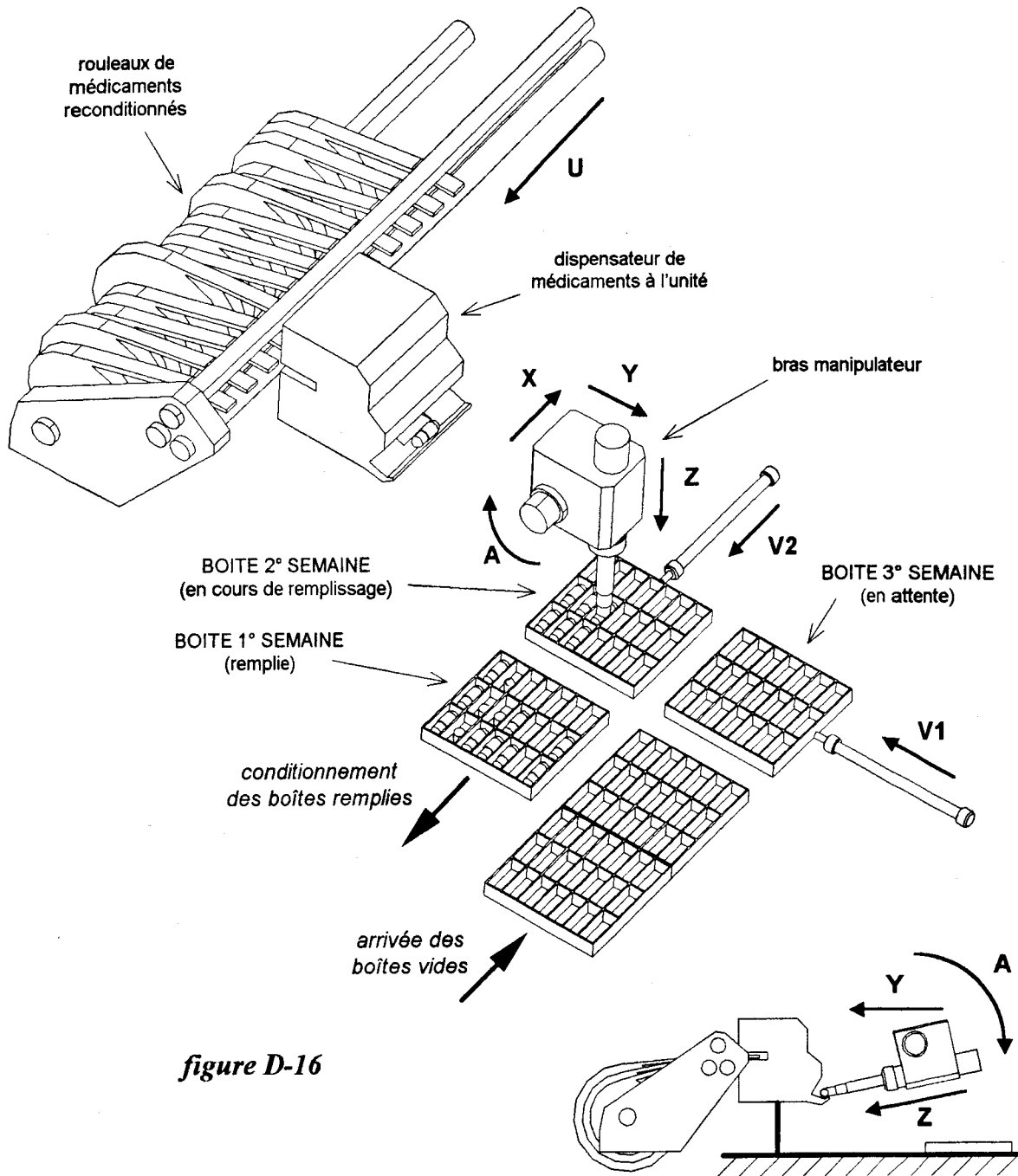


figure D-16

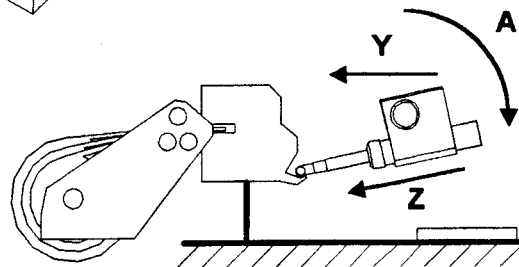


figure D-17

La figure D-17 montre le bras manipulateur vu de profil entrain de saisir un médicament dans le dispensateur. Un cycle complet pour un lot de boîtes est le suivant :

- amenée d'une boîte vide à l'aire de remplissage (vérin V1)
- positionnement X et Y du bras manipulateur et rotation A pour l'amener dans l'axe du médicament à saisir, Z étant en position arrière
- pendant ce temps : préparation du magasin de rouleaux à la bonne position et prélevement d'un médicament par le dispensateur
- lorsque ces deux opérations sont terminées : avancée suivant Z et saisie d'un médicament (l'organe de préhension n'est pas étudié)
- retour de Z, puis de A
- positionnement X et Y en fonction de l'endroit où déposer le médicament
- avancée de Z, dépose, retour de Z
- mise en place de tous les médicaments de la boîte
- évacuation de la boîte par le vérin V2
- préparation des deux boîtes suivantes

## 2 - 3. Énoncé du sujet

### a ) Question 1 : premier niveau d'analyse

- Faire l'inventaire des fonctions à un premier niveau d'analyse et les détailler.
- Rechercher les séquences et tracer l'épure du grafctet de commande.

### b ) Question 2 : structures de données et grafctet de coordination

On prévoit de mémoriser jusqu'à un million de prescriptions puisque la nature exacte du traitement administré à chaque patient doit être conservée.

- Proposer la structure matérielle du système de commande ainsi que l'organisation générale des mémoires. Justifier.
- Proposer une interface utilisateur.
- Tracer la structure de données.
- Prévoir la gestion des données.
- Tracer le grafctet de coordination complet.

### c ) Question 3 : analyse détaillée des tâches

On note X+ l'avance du bras suivant cet axe et X- son recul. L'axe est positif dans le sens de la flèche sur la figure D-16. On note tous les autres mouvements selon le même principe. Les fins de course de V1 sont notés v11 en position avancée et v10 en position reculée. On note de même les fins de course des mouvements V2, A et Z. Les axes U, X et Y sont contrôlés par codeurs incrémentaux.

*Magasin* : le pas entre les rouleaux est de 30 mm. Il correspond à 100 points du codeur incrémental qui contrôle l'axe du magasin. L'origine de l'axe U se situe à 30 mm à gauche du premier rouleau de gauche. Les numéros des rouleaux correspondent aux numéros des médicaments et sont placés dans l'ordre croissant dans le magasin.

*Bras manipulateur* : les empreintes des boîtes ont un entre-axe de 20 mm suivant X et de 15 mm suivant Y. Le dispensateur se trouve dans le prolongement des empreintes du milieu. Suivant Y, le déplacement nécessaire entre le dispensateur et la rangée du premier *jour* est de 40 mm. Les codeurs suivant X et Y délivrent 10 points par mm de déplacement. Leur origine est la suivante :

- suivant X : 20 mm à gauche de la colonne d'empreintes de gauche
  - suivant Y : en position de prise de médicament dans le dispensateur
- Traiter l'alarme.
  - Choisir des symboles pour les variables significatives et les définir.
  - Tracer les grafctets de tâches au niveau pré-actionneur.

## 2 - 4. Corrigé et commentaires

### a ) Premier niveau d'analyse (Q.1)

- *Inventaire et définition des fonctions*

palettiser

déposer

saisir

prélever

figure D-18

**Fonction : palettiser**

Cette fonction consiste à évacuer la boîte qui vient d'être remplie et ensuite à présenter une boîte vide à l'aire de remplissage.

**Fonction : déposer**

Le bras manipulateur se positionne au-dessus de l'emplacement où il doit déposer le médicament. Le bras descend, le médicament est déposé et le bras remonte.

**Fonction : saisir**

Quelle que soit la position du bras, il se positionne en face du dispensateur. Le bras avance et le médicament est saisi s'il est disponible. Puis le bras recule. Il bascule ensuite en position verticale.

**Fonction : prélever**

En fonction de la référence de médicament demandée, le magasin se positionne pour que le rouleau adéquat soit engagé dans le dispensateur. Le code du rouleau est vérifié. S'il est correct, un médicament est prélevé et mis à disposition du bras manipulateur. Si le code n'est pas correct, l'alarme est déclenchée et le mode de production est interrompu. Un opérateur change les rouleaux mal placés et retire les boîtes éventuellement remplies ainsi que la boîte qui était en cours de remplissage. Après acquittement du défaut, le cycle complet est relancé.

**- Recherche des séquences**

Les fonctions telles qu'elles ont été définies, conduisent à la structure suivante :

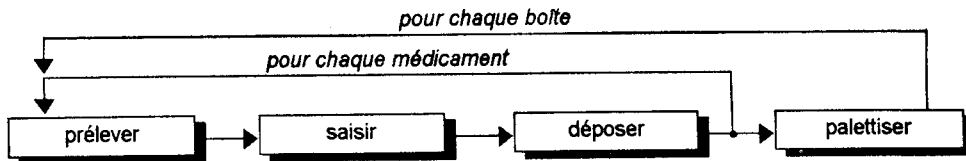


figure D-19

Une analyse plus fine montre que la cadence peut être augmentée en scindant la fonction *saisir* en deux parties dont la première (notée *préparer le bras*) peut être réalisée en même temps que la fonction *prélever*. Le grafcet au niveau *fonction* de la figure D-20 traduit le cycle complet.

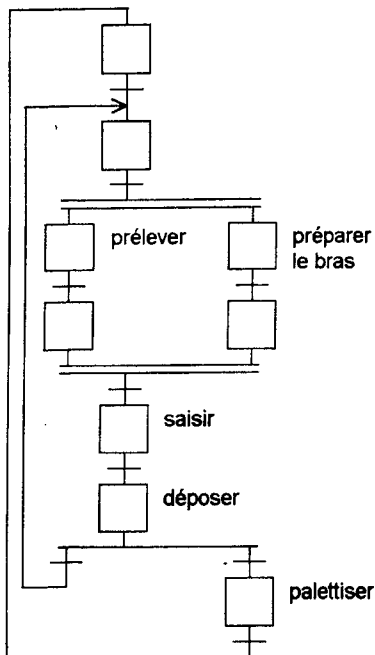


figure D-20

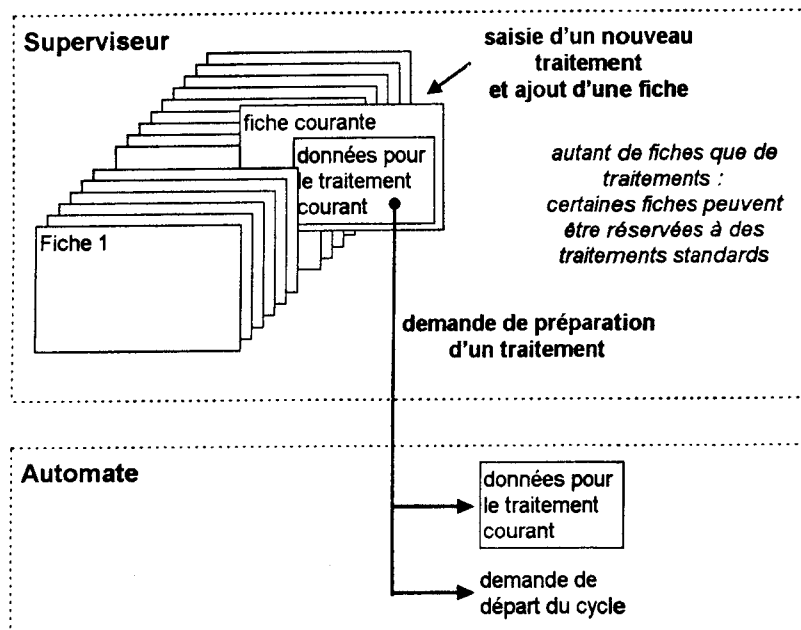


figure D-21

### b) Structure de données et graficet de coordination (Q.2)

La quantité d'informations à archiver n'est pas compatible avec la taille mémoire d'un automate. Les données sont gérées dans ce cas par un superviseur en ayant recours à une recette. Chaque fiche de cette recette est réservée à un patient et comporte toutes les informations : nature exacte du traitement, identité complète du patient...

La mise en forme des données est faite par le superviseur en fonction de la nature du traitement définie par le médecin et les données d'identification du patient. Seules les informations relatives au traitement à préparer sont transmises à l'automate au début de chaque cycle. Voir la figure D-21.

Un fragment de l'interface utilisateur est présenté figure D-22. La configuration des boîtes est représentée à l'écran et pour chaque emplacement, deux champs de saisie sont disponibles : l'un pour le numéro du médicament, l'autre pour son nom en clair. Chaque fiche est identifiée par son numéro auquel est associée l'identité du patient.

The image shows a graphical user interface for a medical system. It features several input fields and buttons. At the top, there are two main sections: 'numéro patient' and 'identité patient'. Under 'numéro patient', the value '001.356' is displayed, with two radio buttons labeled 'appeler' and 'enregistrer'. Below this, there is a section for 'numéro traitement prééglé' with the value '000.000' and similar 'appeler' and 'enregistrer' radio buttons. Further down, it says 'semaine 1'. The bottom part of the interface shows a grid for medication scheduling. It has two columns for 'jour 1' and 'jour 2'. For 'jour 1', the value '083' is shown in two boxes, with 'trevielos 1000' written below each. For 'jour 2', the value '049' is shown in a box, with 'carbone' written below it.

figure D-22

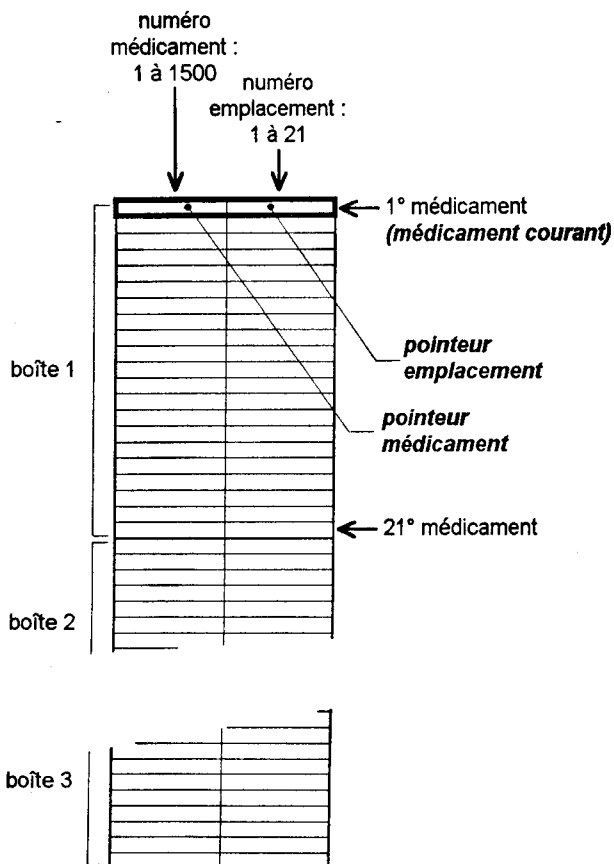


figure D-23

La structure de données détaillée est la même pour chaque fiche de la recette du côté du superviseur et est également reproduite du côté de l'automate (figure D-23).

Toutes les lignes de la mémoire peuvent être scrutées systématiquement les unes après les autres :

- si les pointeurs affichent des valeurs non nulles, ils servent à désigner le médicament à saisir et l'endroit de sa dépose,
- si les pointeurs sont nuls, aucune saisie de médicament n'a lieu,
- après 21 de ces opérations, la tâche de palettisation échange les boîtes,
- au bout de 3 palettisations, le cycle complet est terminé : le superviseur doit en être informé.

Les fonctions liées à la gestion des données sont les suivantes :

- enregistrement des données issues du superviseur en début de cycle,
- décalage de la file d'attente systématiquement après chaque opération de saisie-dépose,

- incrémentation d'un compteur à chaque opération (noté Cop),
- incrémentation d'un second compteur à chaque nouvelle boîte (noté Cb).

Les compteurs sont initialisés en début de cycle. Le compteur Cop est de plus réinitialisé à chaque nouvelle boîte.



**- Grafcet de coordination complet**

Le cycle démarre sur demande du superviseur. Celui-ci transmet en même temps les données relatives au traitement à préparer : une première étape consiste à les enregistrer.

Les pointeurs correspondent alors au premier médicament à déposer. Les 21 décalages successifs permettent le remplissage de la première boîte.

Le compteur des opérations est réinitialisé et le compteur de boîtes est incrémenté.

Lorsque les 3 boîtes ont été remplies, le cycle est terminé et le superviseur en est informé.

Le saut d'étapes X2 → X10 a lieu lorsqu'une opération doit rester sans effet. De ce fait, toutes les 63 positions de la file d'attente sont scrutées.

*Remarque :* le nombre de décalages étant géré par compteurs, il n'est pas utile de forcer la dernière position à zéro.

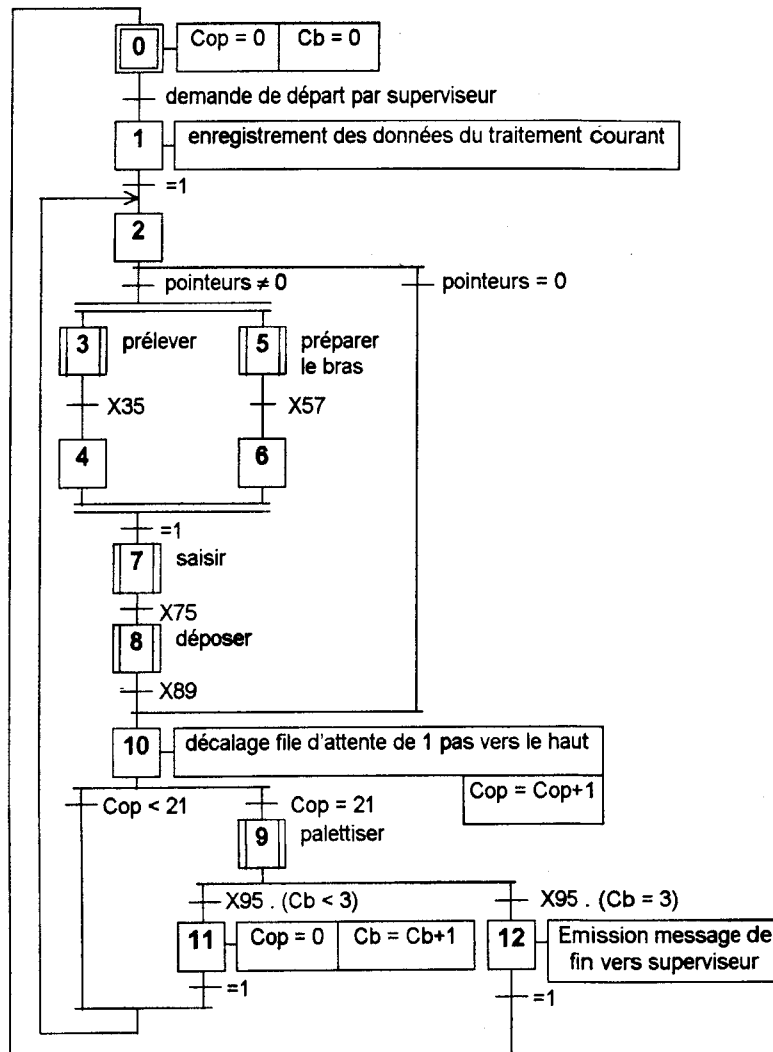


figure D-24

**c) Analyse détaillée des tâches (Q.3)**

**- Alarme**

L'alarme est générée par l'étape X33 de la tâche *prélever* en cas d'erreur de code (figure D-28). Le grafcet de surveillance X90 est de niveau hiérarchique le plus élevé et commande l'interruption du cycle grâce au forçage de tous les autres grafcets dans la situation vide. Après intervention de l'opérateur (changement de place des rouleaux par exemple et enlèvement des boîtes), le cycle complet doit reprendre. Tous les grafcets de tâche sont réinitialisés et le grafcet de coordination du mode de production doit démarrer à condition d'en refaire la demande à partir du superviseur. Le grafcet de surveillance complet se limite aux étapes de la figure D-25.

L'architecture hiérarchisée est résumée par le diagramme suivant :

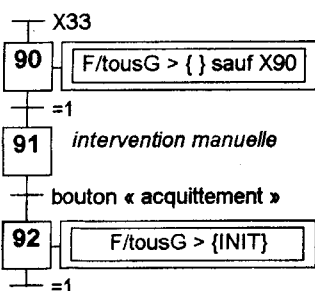


figure D-25

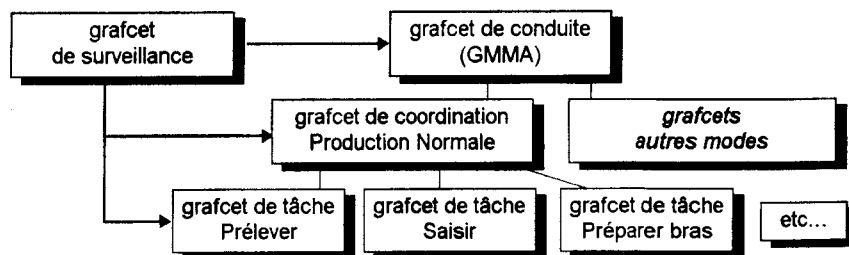


figure D-26

**- Définition des variables**

Les variables numériques sont de deux natures :

- les compteurs qui évoluent avec les codeurs incrémentaux, ils sont notés  $C_u$ ,  $C_x$  et  $C_y$ .
- les valeurs utiles de la structure de données, il s'agit :
  - du numéro de médicament, noté  $N_m$  : la position réelle à atteindre suivant l'axe U est :  

$$\{N_m * \text{pas des rouleaux}\}$$
  - de la position de dépose du médicament. Elle est déduite des numéros de colonnes et de lignes, notées  $P_{mx}$  et  $P_{my}$ . Ces numéros sont calculés par le superviseur en fonction de la place occupée par chaque médicament : le pointeur *emplacement* est donc composé de deux variables.

Selon X, la coordonnée réelle est :

$$\{P_{mx} * \text{pas suivant X}\}$$

Selon Y, elle est :

$$(40 \text{ mm} + \{P_{my} * \text{pas suivant Y}\})$$

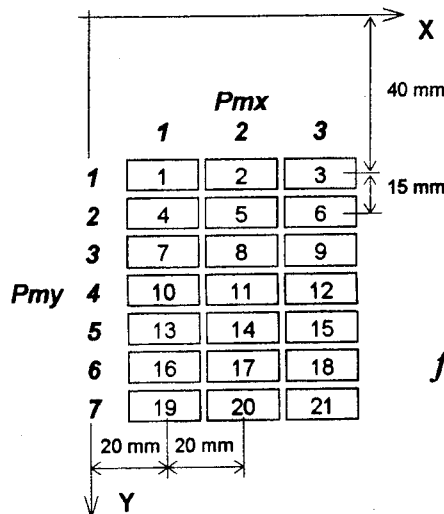


figure D-27

**- Graficets de tâches**

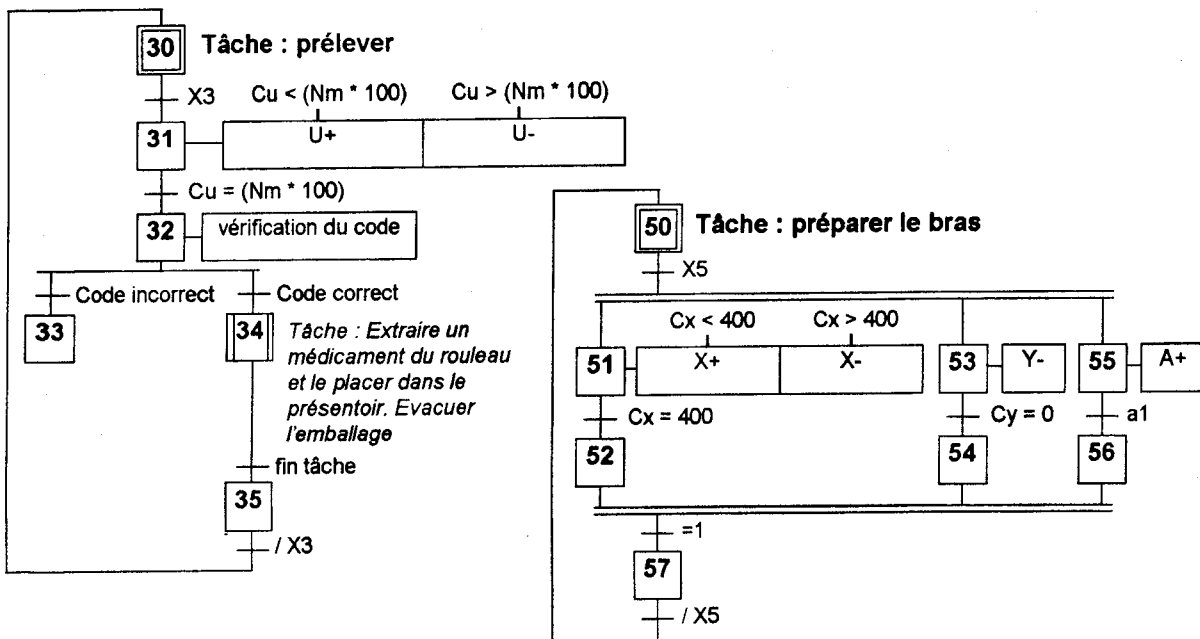


figure D-28

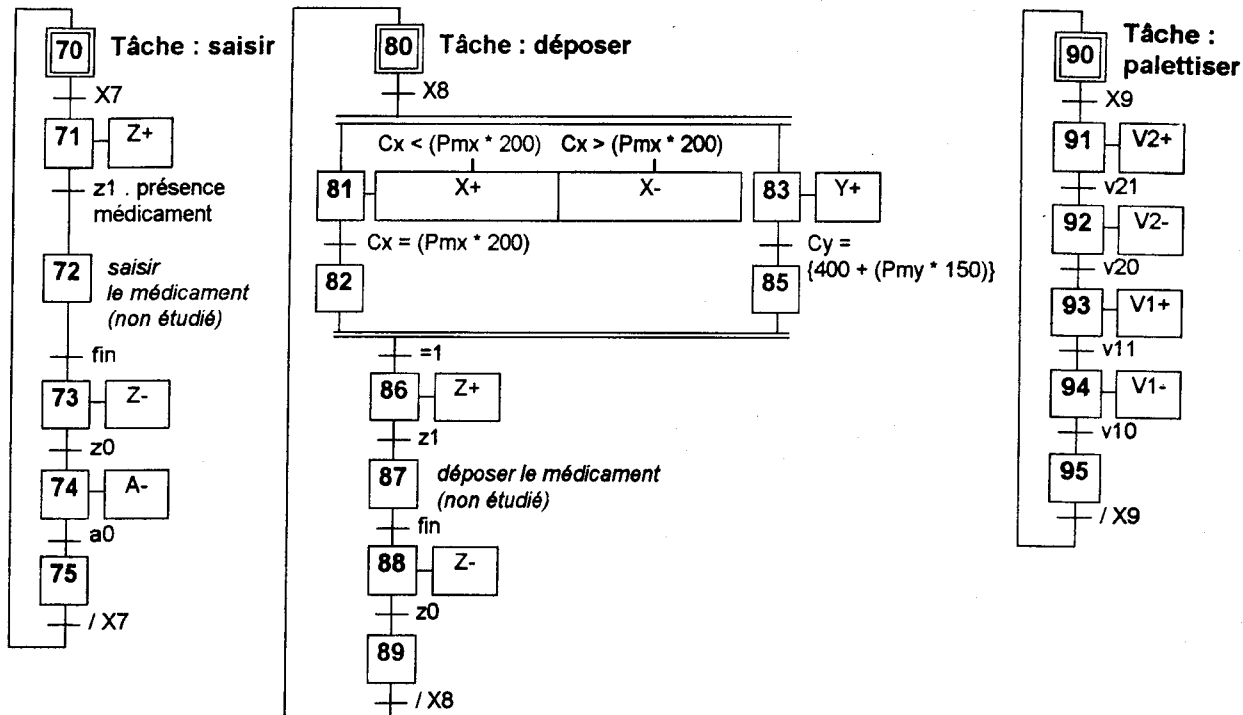


figure D-29

*Remarque 1 :* la discrimination du sens de marche des axes numériques peut également se faire par sélection de séquences. Cette solution avait été proposée en corrigé de l'exercice précédent (bacs de trempe).

*Remarque 2 :* l'écriture des grafquets au niveau *automate* mérite quelques précautions, illustrées par l'exemple suivant.

Supposons qu'un médicament doit être saisi et que le bras se trouve au-dessus de l'empreinte de gauche. Un déplacement suivant X+ positionne le bras en face du dispensateur. Le passage de l'étape X51 vers l'étape X52 (qui provoque l'arrêt du bras) se fait lorsque la coordonnée X400 est atteinte. Or l'inertie mécanique du bras et le temps interne de traitement de l'automate ne permettent pas l'arrêt *immédiat* du bras : un léger dépassement a lieu. Si le médicament doit ensuite être déposé dans la deuxième rangée (alignée avec le dispensateur), Pmx est = 2. L'étape X81 provoquera ainsi un léger retour suivant l'axe X. Ce mouvement n'est pas nécessaire et réduira la durée de vie du moteur. Pour éviter cela, on peut adapter la réceptivité en aval de X81 comme suit :

$$R_{81-82} = [Cx \geq \{(Pmx * 200) - 10\}] \cdot [Cx < \{(Pmx * 200) + 10\}]$$

si l'on considère que le dépassement est proche de 1 mm sans dépasser cette valeur.

De surcroît, le positionnement sera de meilleure précision car l'arrêt est commandé 10 points avant la coordonnée à atteindre, ce qui compense le défaut de dépassement.

### 3 - CELLULE DE POINÇONNAGE-CONTROLE

#### 3 - 1. Présentation du sujet

La cellule comporte 3 poinçonneuses à commande numérique et un poste de contrôle automatisé. Un robot est chargé de dispatcher des pièces en tôle en fonction de leur type. Un code-barre imprimé sur chaque tôle permet leur reconnaissance à l'entrée de la cellule (en position de saisie par le robot) et également au niveau des trois poinçonneuses.

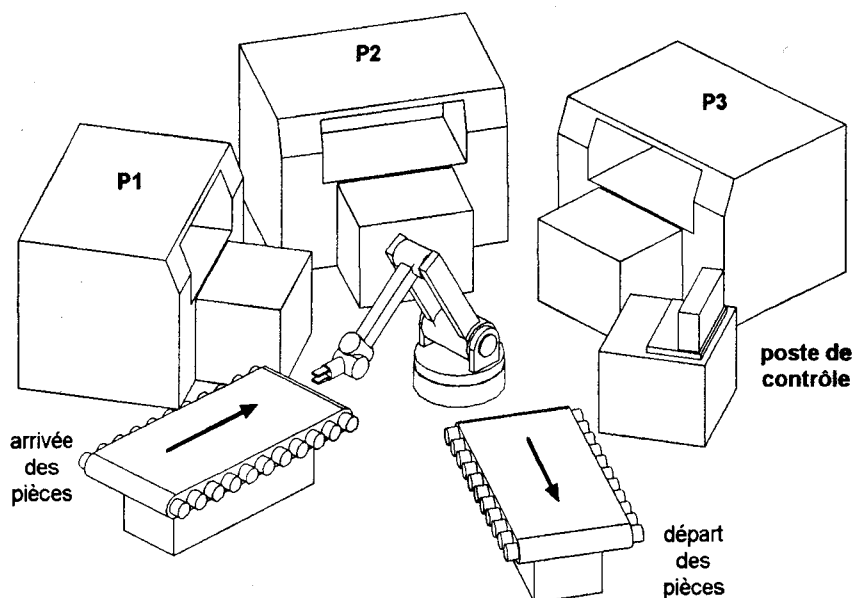


figure D-30

#### 3 - 2. Fonctionnement du système

Selon le type de tôle, le Bureau des Méthodes définit si la gamme nécessite une ou deux opérations :

- l'opération (a) est obligatoire  $\Leftrightarrow$  passage sur l'une des trois poinçonneuses
- l'opération (b) n'est pas obligatoire  $\Leftrightarrow$  passage sur le poste de contrôle

Pour l'opération (a), le Bureau des Méthodes définit les poinçonneuses qui sont autorisées à recevoir les différents types de pièces. Les choix possibles sont :

- le passage est autorisé sur la machine 1
- le passage est autorisé sur la machine 2
- le passage est autorisé sur la machine 3
- le passage est autorisé sur la machine 1 ou sur la machine 2
- le passage est autorisé sur la machine 1 ou sur la machine 3
- le passage est autorisé sur la machine 2 ou sur la machine 3
- le passage est autorisé sur l'une des trois machines

Pendant le fonctionnement, lorsque plusieurs machines sont permises, la sélection définitive se fait selon leur disponibilité. Cette condition étant évolutive, son résultat varie en temps réel.

Un numéro de programme doit en outre être affecté à chaque opération de chaque type de pièce. Ce numéro sera pris en compte par la commande de la machine en question lors des opérations de poinçonnage et de contrôle.

On souhaite pouvoir modifier les gammes sans devoir provoquer l'interruption de la production. Le taux d'occupation de la cellule doit être le plus élevé possible afin d'assurer une productivité maximale.

### 3 - 3. Architecture du système

Les commandes du robot, des 3 poinçonneuses et du poste de contrôle sont toutes reliées directement aux entrées-sorties d'un automate qui gère l'ensemble de la cellule. Par ailleurs, les commandes sont reliées à un serveur de programmes par l'intermédiaire d'un réseau (figure D-31).

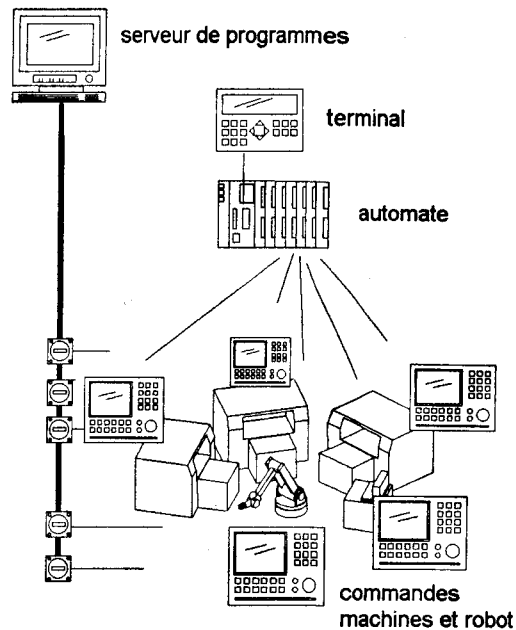


figure D-31

### 3 - 4. Enoncé du sujet

#### a) Question 1 : étude dynamique

Soit la situation suivante :

- arrivée de la pièce 1 (de type 9, seule la machine 1 est autorisée, le contrôle n'est pas demandé),
- arrivée de la pièce 2 (de type 3, toutes les machines sont autorisées, le contrôle est demandé),
- arrivée de la pièce 3 (de type 9).

Envisager plusieurs évolutions différentes en faisant varier la fréquence d'arrivée des pièces et tracer un graphique pour deux d'entre-elles. On précise que les temps des opérations sont différents selon le programme exécuté. Conclure sur la prévisibilité des cycles.

#### b) Question 2 : analyse

- Faire l'inventaire des fonctions et les détailler. Le niveau d'analyse doit être suffisamment précis pour pouvoir mettre en place les mouvements du robot.
- Rechercher les séquences et tracer l'épure des grafjets.

#### c) Question 3 : structures de données et grafjets

Les pièces sont regroupées en 20 familles. On souhaite pouvoir affecter un millier de programmes différents aux opérations de poinçonnage et autant au contrôle.

- Tracer la structure des données initiales.
- Prévoir la gestion des données.
- Tracer les grafjets complets.

#### d) Programmation

*Si le lecteur ne dispose pas du manuel de référence du constructeur, il se reportera directement au corrigé.*

On envisage de décentraliser la commande en affectant un micro-automate à chaque constituant de la cellule (figure D-32). Un réseau local véhicule alors les signaux de synchronisation.

A titre d'exemple, traiter la demande de départ cycle de la machine 1 (le téléchargement des programmes de poinçonnage n'entre pas dans le cadre de cette étude). Proposer une solution pour les réseaux Sinec-L1 et Fipway.

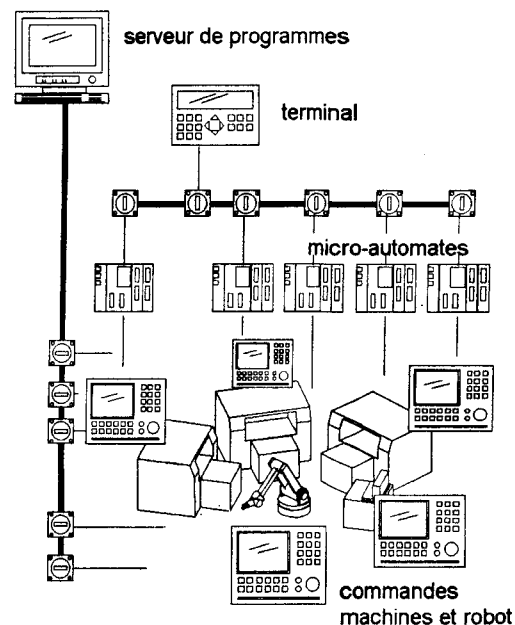


figure D-32

### 3 - 5. Corrigé et commentaires

#### a ) Etude dynamique (Q.1)

L'axe du temps des diagrammes de la figure D-33 évolue de haut en bas. En situation initiale, aucune pièce n'est présente à l'entrée de la cellule. L'arrivée de la première pièce déclenche le fonctionnement. Les deux simulations diffèrent en raison de la fréquence d'arrivée des pièces : un rythme lent pour la première simulation, un rythme plus soutenu pour la seconde.

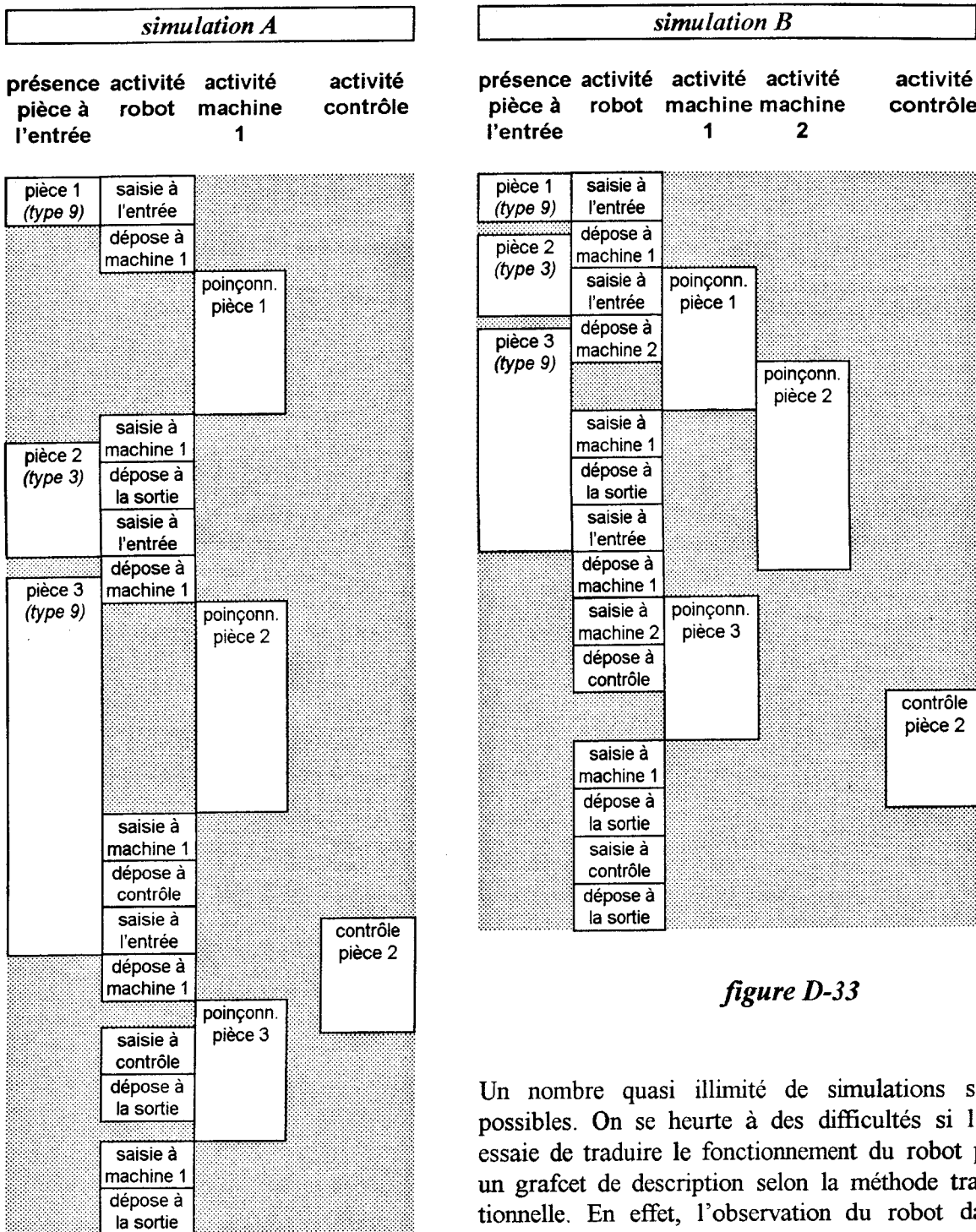


figure D-33

Un nombre quasi illimité de simulations sont possibles. On se heurte à des difficultés si l'on essaie de traduire le fonctionnement du robot par un grafcet de description selon la méthode traditionnelle. En effet, l'observation du robot dans différentes simulations ne permet pas de dégager de séquence *a priori*.

On constate également que l'activité des machines ne suit pas la même évolution dans les deux cas. Par ailleurs, la pièce n°2, de type 9, n'est pas traitée par la même poinçonneuse. Dans d'autres

simulations, elle pourrait également être traitée par la machine 3. Toutes les combinaisons sont susceptibles de se produire à un moment où à un autre si la cellule fonctionne suffisamment longtemps.

Même s'il est de type déterministe (les mêmes effets provoquant les mêmes causes), le système n'a pas un comportement prévisible. La fréquence d'arrivée des pièces, leur ordre d'apparition, les durées d'exécution des programmes, la disponibilité au fil du temps des machines, du poste de contrôle et du robot sont autant de paramètres qui influencent la progression des activités.

*Remarque* : du point de vue de l'ordonnancement, seul un *logiciel de simulation* permet d'étudier avec précision le comportement futur de la cellule. Ces logiciels sont d'utilisation assez complexes (à cause de la quantité et de la pertinence des informations à saisir) et restent d'un prix très élevé, de l'ordre de 300 kF.

### b) Analyse (Q.2)

Malgré le caractère aléatoire du fonctionnement, il faut construire le programme de l'automate qui commande la cellule. Procédons selon la méthode proposée.

#### - Décomposition fonctionnelle

Un premier niveau de décomposition permet d'isoler le robot et chaque poste de travail puisque ces cinq constituants sont indépendants et ne sont pas synchronisés de manière immuable. On trace un grafcet autonome pour chacun d'eux.

#### Les 4 postes

Chaque poste peut avoir 3 états différents :

- il est *libre* (aucune pièce ne se trouve sur le poste, le poste est à l'arrêt),
- il est en *travail* (un cycle de poinçonnage ou de contrôle est en cours),
- il est en *attente* de déchargement (le cycle de travail est terminé mais la pièce est encore présente sur le poste).

Ces 3 états s'enchaînent toujours de la même façon. Le tracé des grafquets correspondants est donc immédiat :

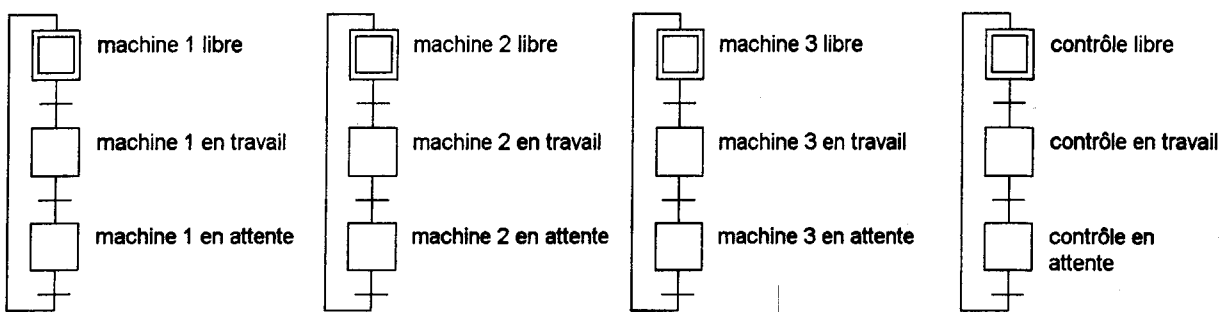


figure D-34

#### Le robot

De manière similaire à l'étude de cas du chapitre VI (cellule double), deux phases se succèdent invariablement : la saisie d'une tôle puis la dépose de la tôle, puis à nouveau une saisie. Le grafcet de commande du robot est ainsi construit très globalement en deux phases :

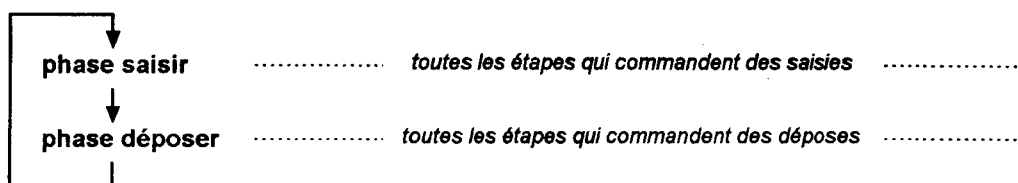


figure D-35

Les fonctions liées au robot, de niveau de précision suffisant pour sa gestion, sont les suivantes :

PHASE SAISIR		PHASE DEPOSER	
SE	saisir à l'entrée de la cellule	DP1	déposer au poste 1
SP1	saisir au poste 1	DP2	déposer au poste 2
SP2	saisir au poste 2	DP3	déposer au poste 3
SP3	saisir au poste 3	DC	déposer au poste de contrôle
SC	saisir au poste de contrôle	DS	déposer à la sortie de la cellule

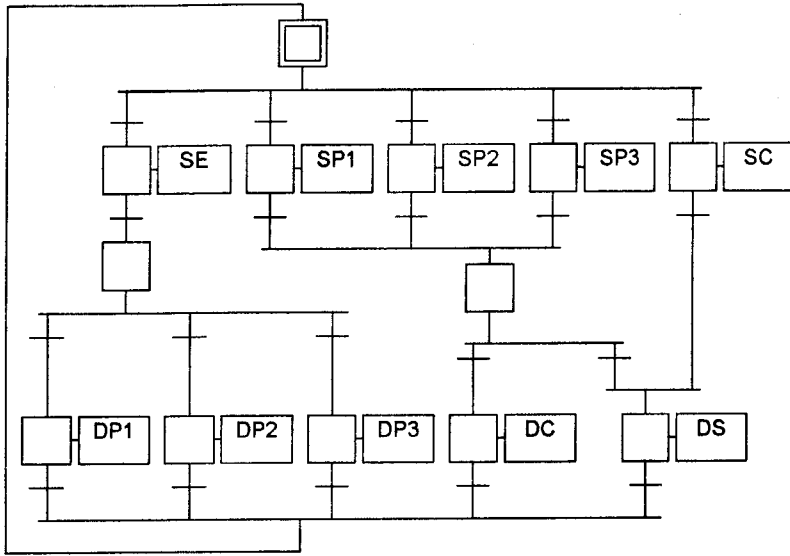


figure D-36

La recherche des séquences obligées permet de tracer simplement une première épure du grafcet de commande du robot. Toutes les saisies sont autorisées lorsque le robot est disponible. Lorsqu'une saisie a eu lieu à l'entrée de la cellule, la dépose se fait obligatoirement sur l'une des poinçonneuses. Après saisie sur une poinçonneuse, la tôle peut être déposée au poste de contrôle ou être évacuée directement. Après saisie sur le poste de contrôle elle est obligatoirement évacuée. Aucune contrainte ne conduit à exécuter une saisie particulière après une dépose.

c ) Structures de données et grafkets (Q.3)

- Mémoire initiale

Chacune des 20 familles de pièces nécessite jusqu'à 2 opérations. La première opération est paramétrée et la deuxième est facultative. Un numéro de programme est affecté individuellement à chaque opération. On propose la mémoire initiale suivante :

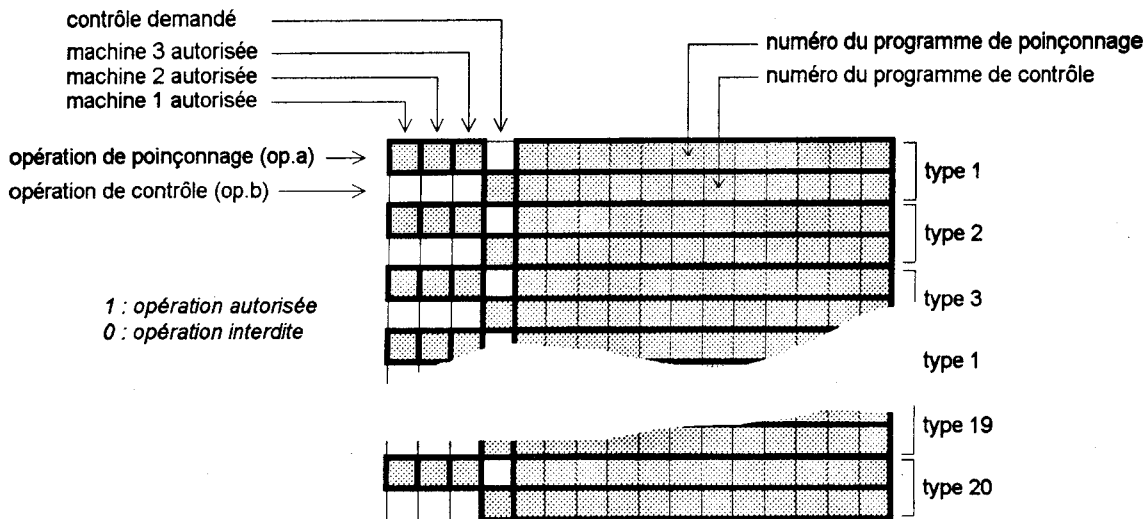


figure D-37

Les 4 bits de poids fort permettent de spécifier la nature des opérations. Pour les opérations de poinçonnage, on a soit un seul, soit 2, soit 3 bits à un. Les numéros de programmes sont codés sur les 12 bits de poids faible.



### - Gestion des données

L'utilisation d'une mémoire de travail facilite l'écriture des réceptivités en réduisant le nombre de paramètres à tester. La mise à jour d'une seule opération est suffisante, la signification des 4 bits de poids le plus fort étant fonction du type d'opération en cours.

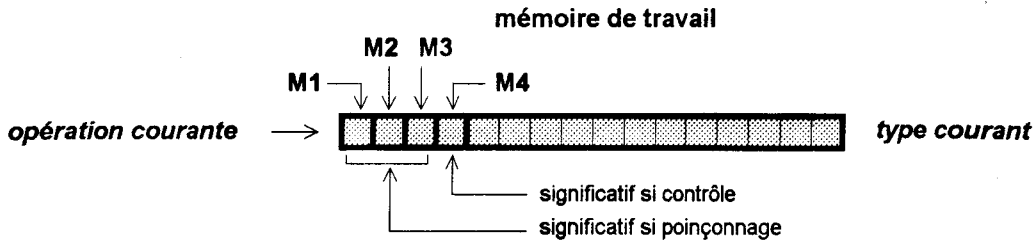


figure D-38

La sélection des informations à dupliquer dépend du type de tôle et du type d'opération à réaliser.

### - Grafecets complétés

#### Synchronisation entre le robot et les postes

Lorsque le robot a déposé une tôle sur un poste, celui-ci doit démarrer en utilisant le numéro de programme indiqué par la mémoire de travail. A cet effet, des étapes en aval de celles qui commandent les déposes sont ajoutées au grafecet du robot (il s'agit de X74, X76, X78 et X80).

Lorsque le programme est terminé, et quelle que soit sa durée, l'étape de compte-rendu du poste est activée (X12, X22, X32 ou X42). Elle génère une demande d'évacuation, interprétée par le grafecet de commande du robot au niveau des réceptivités en amont de X54, X56, X58 et X64. Lorsque la pièce est enlevée, le grafecet du poste est réinitialisé pour indiquer qu'il est de nouveau libre.

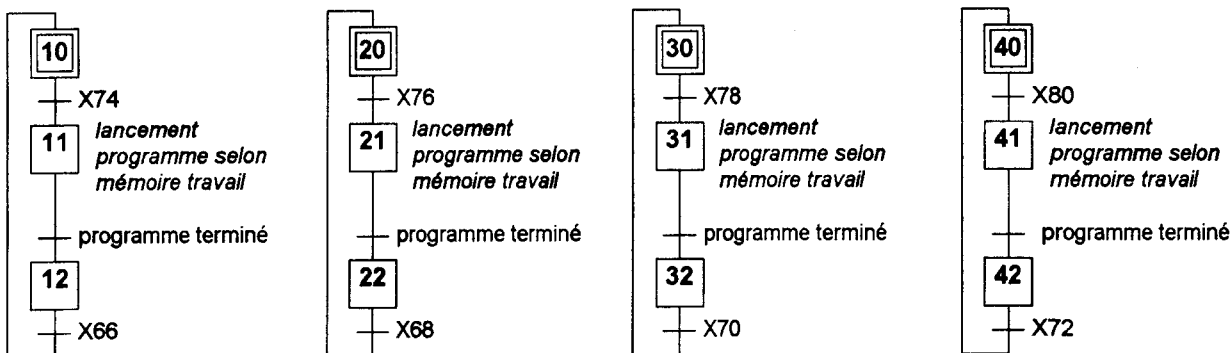


figure D-39

#### Fonctionnement du robot

Le fonctionnement normal de la cellule est autorisé ou non par un grafecet de niveau hiérarchique supérieur.

Pour éviter de bloquer la cellule, on retient le principe qui interdit au robot de saisir une pièce tant que sa dépose ne peut pas être garantie :

- dans le cas où une saisie est envisagée à l'entrée de la cellule, il est donc nécessaire d'identifier le type de pièce (étape X52), de consulter quels sont les postes de poinçonnage autorisés et vérifier la liberté de ces derniers (réceptivités en aval de l'étape X53),
- si une saisie est envisagée sur une poinçonneuse, il faut vérifier si, pour le type de pièce en question, l'opération de contrôle est demandée, et, le cas échéant, vérifier la disponibilité de ce poste (réceptivités en aval des étapes X55, X57 et X59).

En cas d'échec, aucune action n'a lieu : le grafecet est immédiatement réinitialisé. Au niveau automate, il faut veiller à ce que le grafecet puisse quitter la boucle [X50-51-52-53] afin de permettre l'évolution vers les séquences X54, X56, X58 ou X64.

Afin de ne pas surcharger la figure, la gestion des priorités n'est pas représentée. On prévoira par exemple de favoriser l'entrée d'une pièce dans la cellule plutôt qu'une évacuation pour maximiser le taux d'occupation des machines et donc la cadence générale.

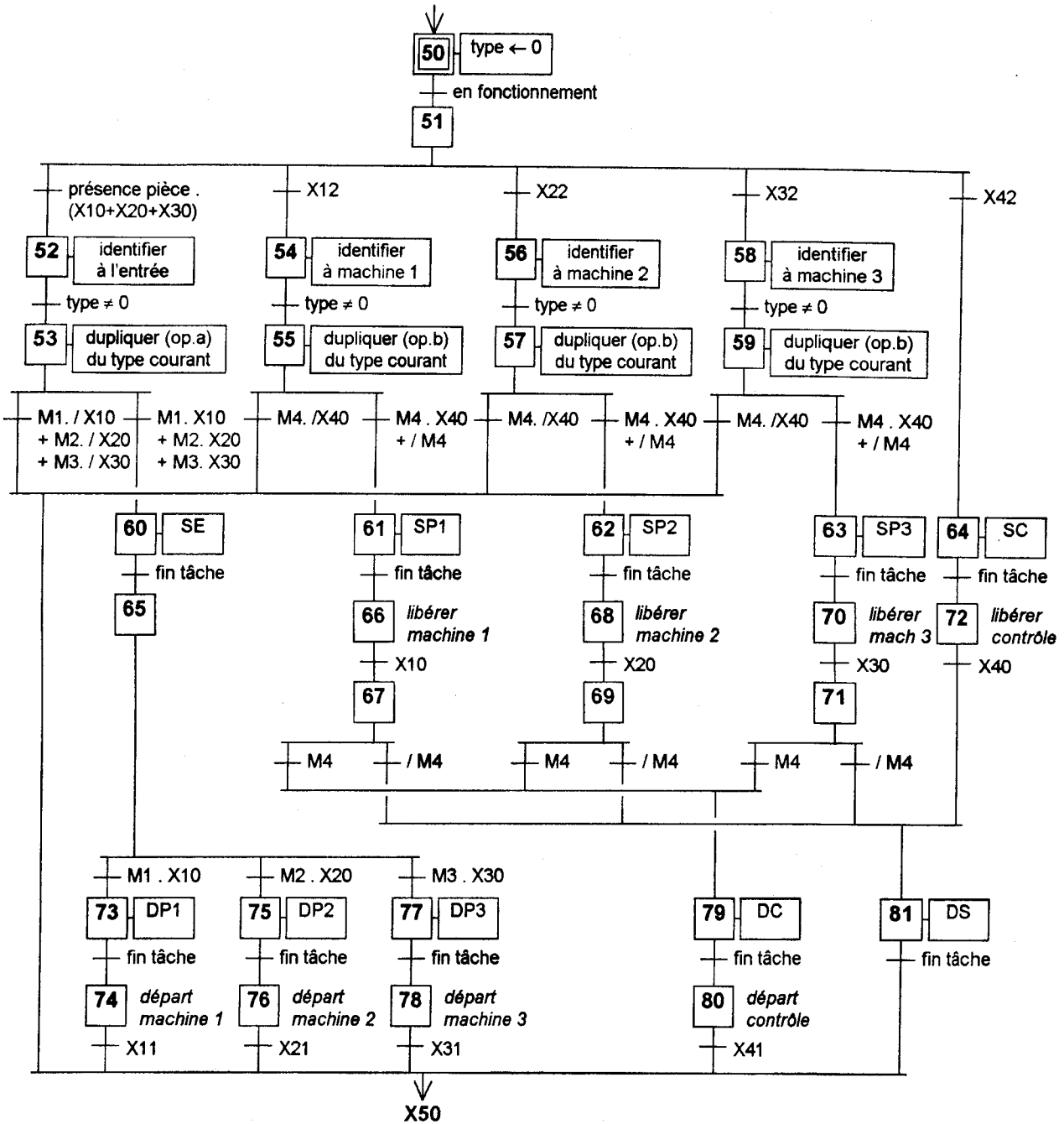


figure D-40

**d) Programmation**

Dans la nouvelle configuration de la figure D-32, les graficets ci-dessus sont tous implantés dans des automates différents. On demande de traiter le départ cycle de la machine 1 :

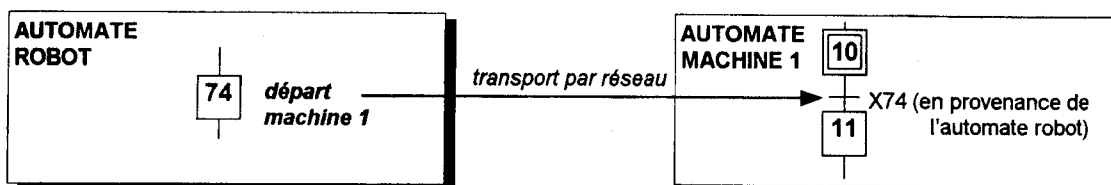


figure D-41

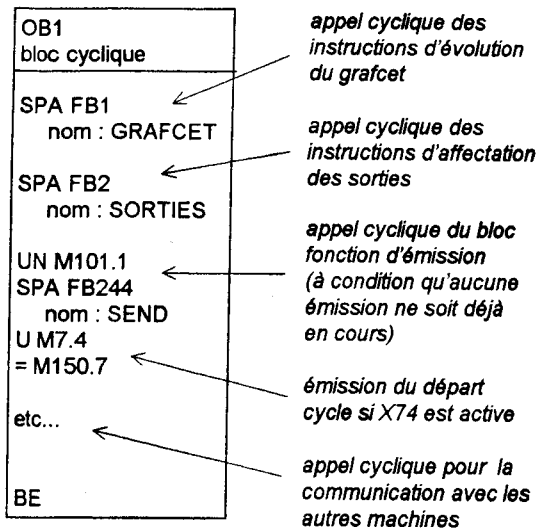
### - Réseau local Sinec-L1 (Step5)

Deux types d'interfaces, qui nécessitent des techniques de programmation différentes, sont mis à disposition de l'utilisateur :

- un coupleur spécialisé,
- la prise console de l'automate.

On propose de connecter l'automate du robot sur le réseau par l'intermédiaire d'un coupleur, et l'automate de la machine 1 par sa prise console.

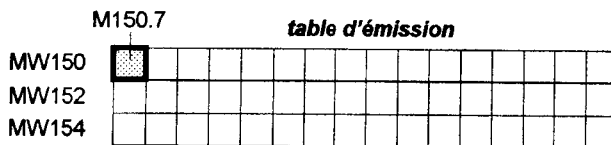
#### Emission de l'ordre de départ (connexion par le coupleur)



Le bloc FB244 est réservé au paramétrage de la table d'émission :

lors de la saisie du programme, on précise le numéro d'interface, le numéro de l'automate destinataire, l'adresse du mot indicateur, l'adresse de début de la table d'émission et sa taille, et l'adresse de l'octet d'indication d'erreur.

Le mot indicateur reflète l'état de traitement de la fonction communication. L'un des bits permet de déterminer si une émission est en cours. Dans l'exemple ci-contre, le mot indicateur MW100 a été configuré : le bit d'activité à tester est donc M101.1



L'adresse de début de table a été configurée à MW150. La taille est de 3 mots. Le bit qui est l'image du départ cycle est choisit librement : par exemple celui en haut à gauche de cette table, à savoir M150.7

figure D-42

#### Réception de l'ordre de départ (connexion par la prise console)

La structure de la table de réception doit être identique à celle de la table d'émission. Il n'est pas utile par contre que les adresses soient les mêmes. Ainsi, dans l'exemple, le départ cycle est représenté par le bit M130.7 du côté de l'automate qui commande la machine.

Dans le cas d'une connexion par la prise console, le paramétrage de la table se fait à partir du bloc de démarrage de l'automate en chargeant les valeurs adéquates dans des mots systèmes réservés : on précise le numéro de l'automate, les adresses des octets de coordination (émission et réception) et les adresses et tailles des tables (émission et réception)

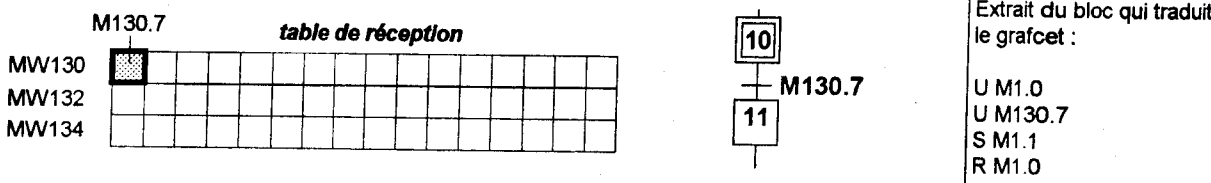


figure D-43

Alors que l'utilisation d'un coupleur permet de définir autant de tables que souhaité (une table d'émission et une table de réception en relation avec chaque automate du réseau par exemple), on ne peut définir qu'une seule table d'émission et une seule table de réception lorsque les échanges se font par la prise console. Dans ce cas, si l'automate doit dialoguer avec plusieurs stations, il faut organiser un indexage par le programme utilisateur en rafraîchissant les tables finales à tour de rôle.

**- Réseau local Fipway**

Il existe 3 manières de programmer des échanges d'informations entre des équipements raccordés à un réseau Fipway :

- les instructions Uni-Telway (READ\_VAR pour lire un objet, WRITE\_VAR pour positionner un objet...),
- les mots communs (chaque station dispose de 4 mots %NW qui sont diffusés sur tout le réseau et donc exploitables par toutes les autres stations),
- la table partagée.

L'utilisation de la table partagée permet une programmation transparente, comme si les données manipulées se trouvaient toutes dans chaque automate. On configure dans chaque automate une table qui débute toujours à la même adresse et qui est toujours de même longueur. Cette table est accessible en lecture par tous les automates. Par contre, chaque automate ne dispose que d'une plage limitée spécifique pour l'accès en écriture. La configuration de l'exemple pourrait être la suivante :

Automate robot	Automate machine 1	Automate machine 2	Automate machine 3	Automate contrôlé
%MW200 ↓ %MW209	%MW200	%MW200	%MW200	%MW200
	%MW210 ↓ %MW219			
		%MW220 ↓ %MW229		
			%MW230 ↓ %MW239	
<b>%MW249</b>	<b>%MW249</b>	<b>%MW249</b>	<b>%MW249</b>	%MW240 ↓ <b>%MW249</b>

Chaque automate a accès en lecture aux adresses %MW200 à %MW249. Les zones mémoires en gras indiquent les adresses d'écriture pour chaque automate. Ainsi, les ordres de départ des machines que doit émettre l'automate du robot sont inscrits dans la zone %MW200 à %MW209. Le dialogue demandé peut être programmé ainsi :

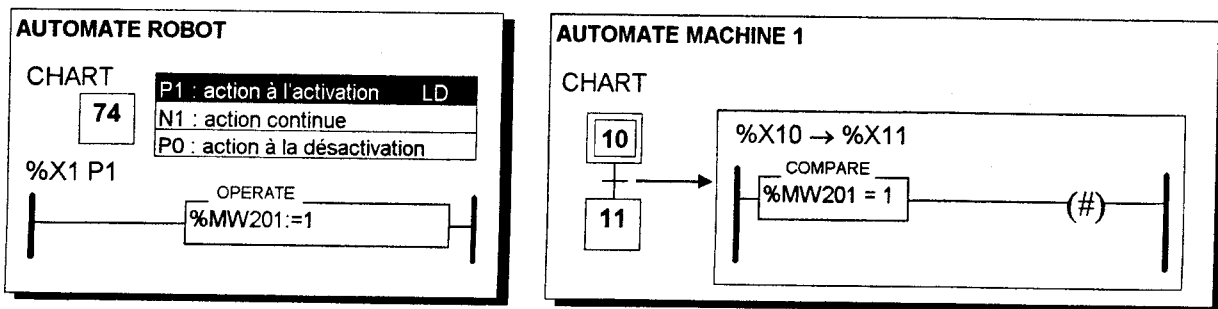


figure D-44

Dans l'automate robot, il faut repositionner %MW201 à Zéro après désactivation de X74. On utilisera %MW202 à %MW204 respectivement pour les machines 2 et 3 et le poste de contrôle.