

137  
ORSAY

Série A

N°

229

YELLECT  
CNRS

N°

THESE

présentée

A L'UNIVERSITE DE PARIS SUD  
CENTRE D'ORSAY

pour obtenir

LE TITRE DE DOCTEUR-INGENIEUR

spécialité :

ELECTRONIQUE  
TRAITEMENT DE L'INFORMATION ET AUTOMATIQUE

par

MANSOUR HORMOZ

Sujet :

*SYSTEME DE TEMPS PARTAGE PERMETTANT  
LA COMPILATION INTERACTIVE*

Soutenue le 7 juillet 1977 devant la Commission d'examen

MM. P. DEBRAINE      Président

L. DONADIEU

MME F. MADAULE

M. G. CHARBONNEAU

ORSAY  
Série  
N°

CNRS  
N°

# THESE

présentée

A L'UNIVERSITE DE PARIS SUD  
CENTRE D'ORSAY

pour obtenir

LE TITRE DE DOCTEUR-INGENIEUR

spécialité :

ELECTRONIQUE  
TRAITEMENT DE L'INFORMATION ET AUTOMATIQUE

par

MANSOUR HORMOZ

Sujet :

*SYSTEME DE TEMPS PARTAGE PERMETTANT  
LA COMPILATION INTERACTIVE*

*ST4/3920.*

Soutenue le 7 juillet 1977 devant la Commission d'examen

MM. P. DEBRAINE      Président

L. DONADIEU

MME F. MADAULE

M. G. CHARBONNEAU

**A mon Père,**

**A mon frère Iraj,**

- Avant propos -

Avant de présenter ce mémoire, je voudrais exprimer ma profonde gratitude à Monsieur le professeur P. Debraine pour sa bienveillance et l'excellence de ses conseils, ainsi que pour m'avoir fait l'honneur de présider le jury de thèse.

Ce travail a pu être mené à bien grâce à Monsieur le professeur L. Donadiou qui m'a accepté dans son laboratoire de l'Université d'ORSAY, je lui en suis très reconnaissant.

Je remercie très sincèrement Madame le professeur F. Madaule, professeur à l'institut de programmation de l'Université de PARIS VI, qui malgré ses nombreuses occupations a bien voulu participer au jury.

A mon ami G. Charbonneau qui tout au long de mon séjour au laboratoire a été un véritable secours aux moments de "crises" et qui a participé au jury de cette thèse comme rapporteur, j'adresse mes plus vifs remerciements.

A Charalambos Krontiris, mon collègue, qui a traversé avec moi des périodes difficiles au laboratoire, j'exprime mes sentiments les plus amicaux.

Que C. Chenu, qui lors de son passage au laboratoire a su créer un climat si amical, soit assuré de toute mon amitié.

## P L A N

### INTRODUCTION.....

### I. PREMIERE PARTIE - DESCRIPTION DU SYSTEME.....

#### I.1. FONCTIONNEMENT STATIQUE DU SYSTEME, GESTION DE LA MEMOIRE CENTRALE ET DE LA MEMOIRE DISQUE.....

I.1.1. Gestion de la mémoire centrale.....

I.1.2. Gestion de la mémoire annexe, disque.....

I.1.3. Gestion des tampons, zone des tampons....

#### I.2. GESTION DYNAMIQUE DU SYSTEME.....

I.2.1. Le rôle du répartiteur et de l'interruption d'horloge.....

I.2.2. Un module important "SWAPR" responsable de l'ensemble de mise en garde.....

I.2.3. Subroutine d'insertion d'un utilisateur dans la file d'attente.....

I.2.4. Structure arborescente du système.....

#### I.3. COMMUNICATIONS.....

I.3.1. Communication de la mémoire centrale avec la mémoire disque.....

I.3.2. Communications liées au multiplexeur.....

I.3.3. Communication du pupitre avec le système.

### II. DEUXIEME PARTIE - UTILISATION DU SYSTEME.....

#### II.1. LANCEMENT ET GENERATION, LES OPERATIONS DE DEBUT ET DE FIN DE TRAVAIL.....

II.1.1. Lancement du système.....

II.1.2. Génération du système.....

II.1.3. Opérations de début de travail.....

II.1.4. Opérations de fin de travail.....

II.1.5. Sauvegardes du système.....



## Errata

Page 29	ligne 3	il faut lire	nous venons
Page 33	ligne 1	"	nouvel utilisateur
" 36	" 25	"	SCH5
" 36	" 26	"	SCH8
" 39	dernière ligne	"	FEED
" 64	ligne 14	"	102011
" 67	" 26	"	SCH7
" 75	" 25	"	CODE ASCII
" 76	" 3	"	SSA, RSS
" 80	dans le dernier rectangle à gauche	"	SCH 3
" 92	" 26 et 27	"	la commande est ignorée

## REMARQUES GENERALES

Il importe de préciser les points suivants :

1. Toutes les adresses mémoire utilisées, soit dans le texte soit dans les figures, sont exprimées en octal.

Au cas où, pour une plus grande clarté, nous employons des valeurs décimales, cela sera clairement indiqué par le nombre 10 situé à droite de l'adresse en question.

2. Ci-dessous nous donnons la correspondance entre français et anglais de certains termes.

Nous avons utilisé ces termes anglais parce qu'ils ont été consacrés par l'usage.

En outre, nous avons conservé les noms (en anglais évidemment) des modules ou sous-modules du système initial, afin de faciliter la tâche d'un programmeur éventuel désireux de faire d'autres transformations.

Pilote

Driver

Mémoire tampon

Buffer

"Mise en garde". Le fait de copier des informations de la mémoire centrale sur la mémoire disque et vice versa

Swapping

Le chargeur utilitaire

Loader Utility

Initialisation du disque

Formattage



Sous-programme de gestion  
de la Bibliothèque des sous-  
programmes système

Library Subroutine

Chargeur des bandes binaires  
absolues, résidant en mémoire  
centrale avec une protection  
du matériel

Basic Binary Loader  
(B.B.L.)

"Répartiteur". Nom du module  
assurant l'ordonnancement et  
l'enchaînement des tâches qui  
se présentent en parallèle

Scheduler

Saut de ligne, le saut à la  
ligne suivante

Line Feed

3. Le corps des 4 annexes que nous avons ajoutées à la fin de l'exposé peut paraître de prime abord assez long par rapport à l'ensemble.

Ceci est dû au souci que nous avons eu de ne pas interrompre le texte par des explications de détail, (quoique fort importants) qui auraient pu nuire à la clarté de l'ensemble.

Ainsi, tout ce qui pouvait constituer des digressions gênantes, a été rassemblé dans ces annexes.

## I N T R O D U C T I O N

Le temps-partagé est une méthode particulière d'utilisation des ordinateurs, qui n'est pas spécifique d'un type d'ordinateurs.

Depuis le projet "MAC", mis au point au "Massachusetts Institute of Technology" sur un ordinateur IBM 7094 et mis en service en 1961, le temps-partagé est défini de la manière suivante :

Une technique d'exploitation des ordinateurs qui permet à plusieurs utilisateurs d'avoir accès direct et en apparence simultanément au système, ainsi que la possibilité d'ajouter ou de modifier des fichiers de façon sélective.

Dans le cadre de cette thèse, nous nous sommes proposés de perfectionner un système d'exploitation en temps-partagé sur un mini ordinateur (HP 2100) de manière à le rendre adaptable à toute compilation conversationnelle, alors qu'à l'origine ce système était limité à un seul langage : le "BASIC".

Le nouveau système (que nous avons appelé "TSM") a été mis au point à partir du système de temps-partagé "T.S.B.", en éliminant ce qui était spécifique du langage "BASIC" tout en conservant le noyau proprement dit du système ; il s'y ajoute un module spécialisé qui joue un rôle essentiel dans le fonctionnement du "T.S.M."

Le "T.S.M." est donc constitué par un ensemble de modules et de sous-modules issus du système initial, la liaison entre ces modules étant spécifique du nouveau système.

D'un point de vue matériel, nous avons extrait environ onze mille instructions des vingt cinq mille instructions du sys-

tème "T.S.B.", auxquelles s'ajoutaient les cinq mille instructions du "LOADER UTILITY".

Nous décrivons l'architecture de ce nouveau système ("T.S.M.") en 2 parties et 4 annexes.

1°) La première partie est consacrée à la description du système ; nous verrons d'abord "le fonctionnement statique du système" qui décrit les détails de la mise en place du système temps-partagé et du sous-système "LOADER UTILITY", dans la mémoire centrale, ainsi que l'étude des tampons et la gestion de la mémoire disque.

Ensuite nous verrons l'étude de "la gestion dynamique du système", qui permettra de comprendre le fonctionnement du système temps-partagé en mettant en évidence le rôle extrêmement important du module "SCHEDULER" ainsi que celui du module "LIBRARY SUBROUTINE".

Ces 2 modules gèrent la partie du système qui réside sur disque et notamment les modules qui sont à implanter en mémoire centrale dans une zone de recouvrement.

Une structure arborescente, qui suit immédiatement la gestion dynamique, permet ensuite de bien saisir la connexion et l'interaction des différents modules du système.

Le problème des communications est traité à la fin de cette partie.

2°) La deuxième partie a pour objet l'utilisation du système.

Tout au début nous verrons les opérations de génération, d'initialisation et de lancement ainsi que les opérations de fin de travail. Cela implique l'étude des interactions du système temps-partagé et du sous-système "LOADER UTILITY".

Ensuite nous entreprendrons l'étude détaillée du jumelage du système temps-partagé avec un compilateur conversationnel quelconque ; toutes les précisions nécessaires au remplacement d'un compilateur par un autre ainsi que les précautions à prendre seront décrites.

A la fin de cette deuxième partie l'utilisation du système est illustrée par un exemple.

L'Annexe 1 donne la configuration du système matériel et ensuite celle du système logiciel.

L'Annexe 2 contient la liste totale des commandes avec le mode d'exploitation de chacune d'elles.

L'Annexe 3 fournit l'ensemble des tâches que le sous-système "LOADER UTILITY" peut exécuter.

L'Annexe 4 contient les 3 modes de chargement du système temps-partagé.



# I. PREMIERE PARTIE

## DESCRIPTION DU SYSTEME

Le système temps-partagé global une fois en service se caractérise, en réalité, par l'interaction du système temps-partagé et du sous-système "LOADER UTILITY" qui est initialement chargé à l'aide du "BASIC BINARY LOADER" ("B.B.L").

C'est le sous-système "LOADER UTILITY" qui charge le système de temps-partagé, fait le chaînage des modules et charge la partie en recouvrement du système temps-partagé sur disque.

C'est toujours le "LOADER UTILITY" qui à la fin du travail est appelé par la commande "SLEEP" à partir du disque et qui assure la sauvegarde du système temps-partagé, soit sur la bande magnétique, soit sur le disque.

## I.1. FONCTIONNEMENT STATIQUE DU SYSTEME, GESTION DE LA MEMOIRE CENTRALE ET DE LA MEMOIRE DISQUE

### I.1.1. GESTION DE LA MEMOIRE CENTRALE

Il existe deux gestions différentes de la mémoire centrale, l'une correspond au sous-système "LOADER UTILITY" et l'autre correspond au système de temps-partagé proprement dit.

#### Gestion du sous-système "LOADER UTILITY"

Le "LOADER UTILITY" une fois chargé, à l'aide du "B.E.L.", occupe dans la page de base une vingtaine de mots, pour la table d'équipements, ensuite il couvre la mémoire centrale de l'adresse 2000 jusqu'à l'adresse 13322, (cf. figure N° 1).

Rappelons que les ordinateurs de la série HP 2100 utilisent une division de la mémoire en pages de 2000 mots. Les seuls mots machine directement adressables sont ceux contenus dans la page courante ou dans la page de base (adresses 0 à 1777). La page de base sert notamment à établir les liens qui sont nécessaires pour accéder aux autres pages.

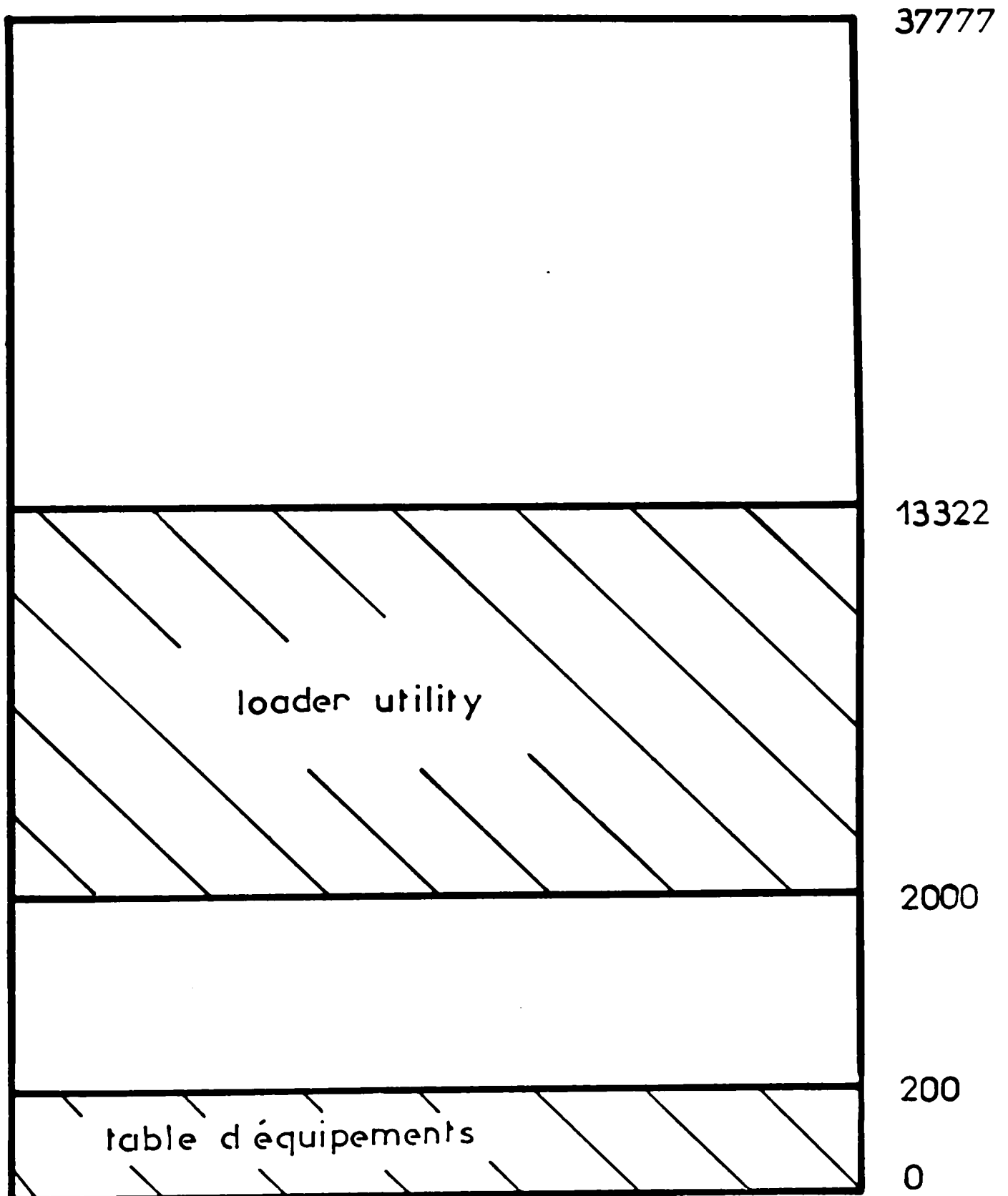
#### Gestion du système temps-partagé

Le système temps-partagé, chargé lui-même à l'aide du "LOADER UTILITY", occupe, au moment de son chargement d'abord toute la page de base des adresses 0000 à 1777.

Ensuite et à partir de 2000 il existe toute une zone non occupée par le système qui est destinée à l'utilisateur pour l'exécution dans sa phase opérationnelle (cf. I.2) et cette zone d'exécution de l'utilisateur s'étend jusqu'à 12253.

schéma de la mémoire  
sous système loader utility

adresses (octales)





A l'intérieur de cette zone, il existe, de l'adresse 4001 à l'adresse 4024,  $20_{10}$  mots qui sont des informations de chaînage fournies au "LOADER UTILITY" par le système temps-partagé au moment de son chargement.

Il est à remarquer qu'une fois le travail de chaînage terminé, au moment où le système de temps-partagé en entier est opérationnel, ces  $20_{10}$  mots seront écrasés par le premier utilisateur qui fera exécuter son programme.

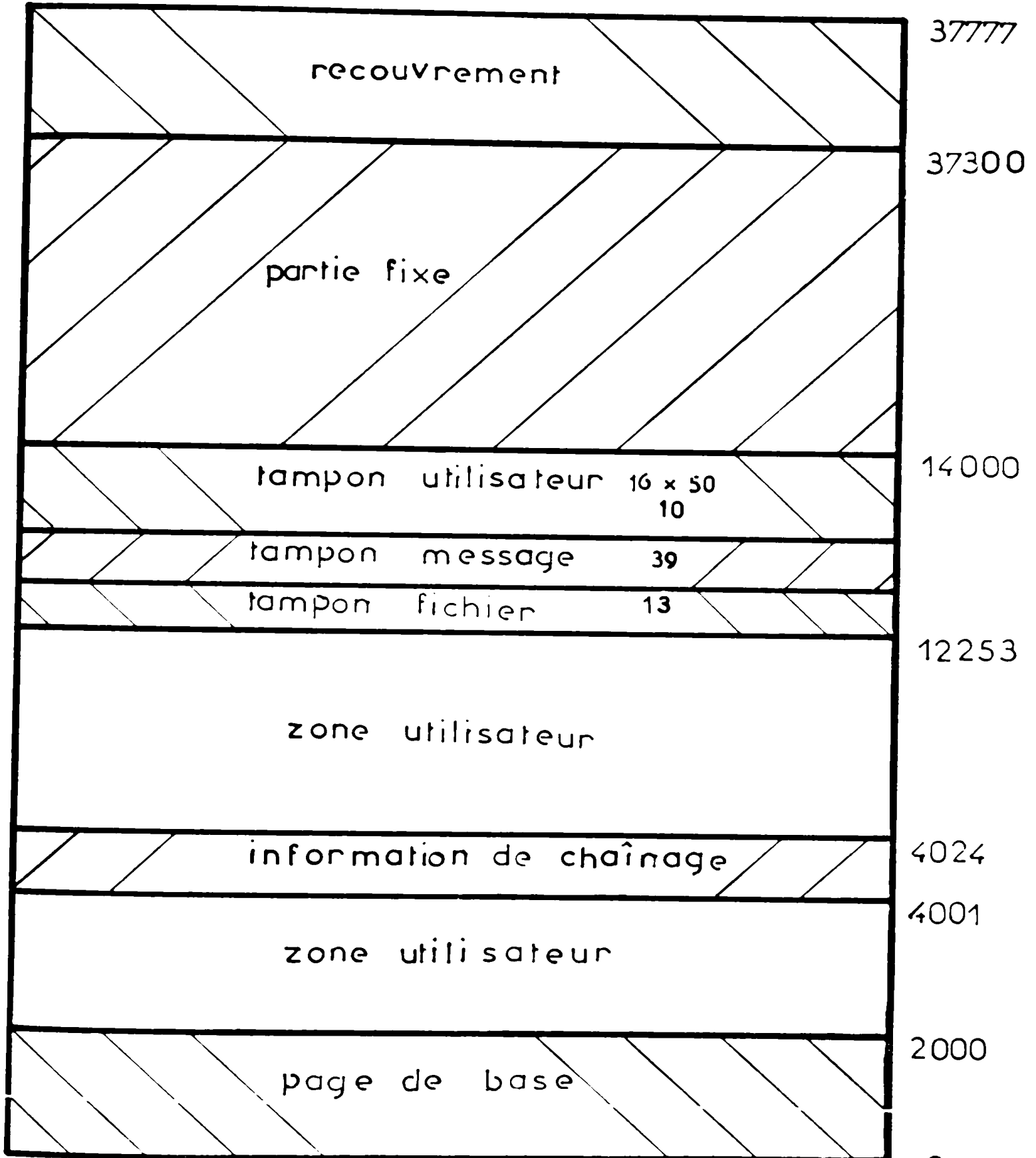
Ensuite la partie du système résidente en permanence dans la mémoire, est implantée de l'adresse 14000 à l'adresse 37300 (cf. Figure N° 2).

Enfin à l'adresse 37300 commence la zone de recouvrement : le système comprend une soixantaine de modules dont le tableau "LIBRARY SIZE" (cf. Figure N° 3) précise les tailles respectives. Ces modules viennent constamment et selon les nécessités se superposer à partir de 37300. Ce tableau permet au chargeur de réserver le nombre minimum de secteurs sur disque pour chacun des modules. Il est écrit sur disque comme un premier module en recouvrement.

schéma de mémoire

systeme temps partagé

adresses (octales)



PAGE 0008 #01

```

4179** DEBUT DES MODULES EN RECOUVREMENT
4180*****
4181**
4182 37300 ORG LIBRA
4183 37300 177707 ABS LIBRA-#SLPL
4184 37301 177600 DEC -128
4185 37302 177406 ABS LIBRA-#FLB
4186 37303 177441 ABS LIBRA-#CHN
4187 37304 177401 ABS LIBRA-#SAV
4188 37305 177406 ABS LIBRA-#SAV
4189 37306 177562 ABS LIBRA-#GET
4190 37307 177453 ABS LIBRA-#APP
4191 37310 177404 ABS LIBRA-#HEL
4192 37311 177640 ABS LIBRA-#HEL
4193 37312 177554 ABS LIBRA-#BYE
4194 37313 177412 ABS LIBRA-#KIL
4195 37314 177402 ABS LIBRA-#REN
4196 37315 177652 ABS LIBRA-#NAM
4197 37316 177535 ABS LIBRA-#CAT
4198 37317 177535 ABS LIBRA-#CAT
4199 37320 177650 ABS LIBRA-#DEL
4200 37321 177640 ABS LIBRA-#TIM
4201 37322 177662 ABS LIBRA-#UDIS
4202 37323 177666 ABS LIBRA-#MESS
4203 37324 177663 ABS LIBRA-#PRO
4204 37325 177663 ABS LIBRA-#PRO
4205 37326 177400 ABS LIBRA-#OPE
4206 37327 177670 ABS LIBRA-#OPE
4207 37330 177716 ABS LIBRA-#LEN
4208 37331 177745 ABS LIBRA-#ECH
4209 37332 177523 ABS LIBRA-#REP
4210 37333 177400 ABS LIBRA-#DIR
4211 37334 177675 ABS LIBRA-#ROS
4212 37335 177613 ABS LIBRA-#RES
4213 37336 177567 ABS LIBRA-#SPE
4214 37337 177676 ABS LIBRA-#POR
4215 37340 177626 ABS LIBRA-#CHA
4216**
4217 37341 177605 ABS LIBRA-#ANN
4218 37342 177400 ABS LIBRA-#SLE
4219 37343 177422 ABS LIBRA-#SLE
4220 37344 177466 ABS LIBRA-#SLE
4221 37345 177516 ABS LIBRA-#NEW
4222 37346 177400 ABS LIBRA-#KID
4223 37347 177421 ABS LIBRA-#KID
4224 37350 177402 ABS LIBRA-#MOV
4225 37351 177401 ABS LIBRA-#MOV
4226 37352 177641 ABS LIBRA-#MOV
4227 37353 177400 ABS LIBRA-#PUR
4228 37354 177525 ABS LIBRA-#PUR
4229**
4230 37355 177746 ABS LIBRA-#PHO
4231 37356 177400 ABS LIBRA-#IN
4232 37357 177443 ABS LIBRA-#UP
4233 37360 177675 ABS LIBRA-#TSB
4234 37361 177517 ABS LIBRA-#PFRS

```

PAGE 0009 #01

```

4235 37362 177400 ABS LIBRA-#PFSW
4236 REP 6
4237 37363 177400 DEC -256
4237 37364 177400 DEC -256
4237 37365 177400 DEC -256
4237 37366 177400 DEC -256
4237 37367 177400 DEC -256
4237 37370 177400 DEC -256
4238 37371 $SLFL EQU *
4239*****
4240** FIN DES MODULES EN RECOUVREMENT
4241*****
** NO ERRORS*

```

### I.1.2. GESTION DU DISQUE

Le disque qui joue le rôle de la mémoire annexe comprend trois zones.

#### Zone binaire système

C'est la zone qui contient, sous forme binaire, l'ensemble des modules qui traitent les commandes du "TSM". Ces modules viennent s'implanter dans la zone de recouvrement.

Comme nous l'avons déjà indiqué dans la gestion mémoire, la partie du système en recouvrement "SYSTEM LIBRARY" est prise en charge par le "LOADER UTILITY" qui après avoir calculé l'emplacement et le minimum de place en secteurs disque écrit, ces commandes sur disque. Et comme nous le verrons en gestion dynamique de la mémoire, c'est au moment opportun, c'est-à-dire au moment où une commande a été tapée, soit au pupitre, soit sur une console multiplexée, que le module "SCHEDULER" amène le module nécessaire en mémoire à l'adresse 37300.

#### Zone de compilation partielle

A la fin de chaque ligne déterminée par le caractère "retour chariot" le programme source est partiellement compilé.

Cette compilation partielle comprend l'analyse syntaxique, l'affectation des opérandes à une table des symboles et le remplacement des étiquettes du programme par des pointeurs absolus.

Les résultats de ces compilations partielles sont ensuite rangés sur disque dans la zone partitionnée pour des utilisateurs et constituent la zone de compilation partielle (cf. Figure N° 5).

### Zone A.S.C.I.I.

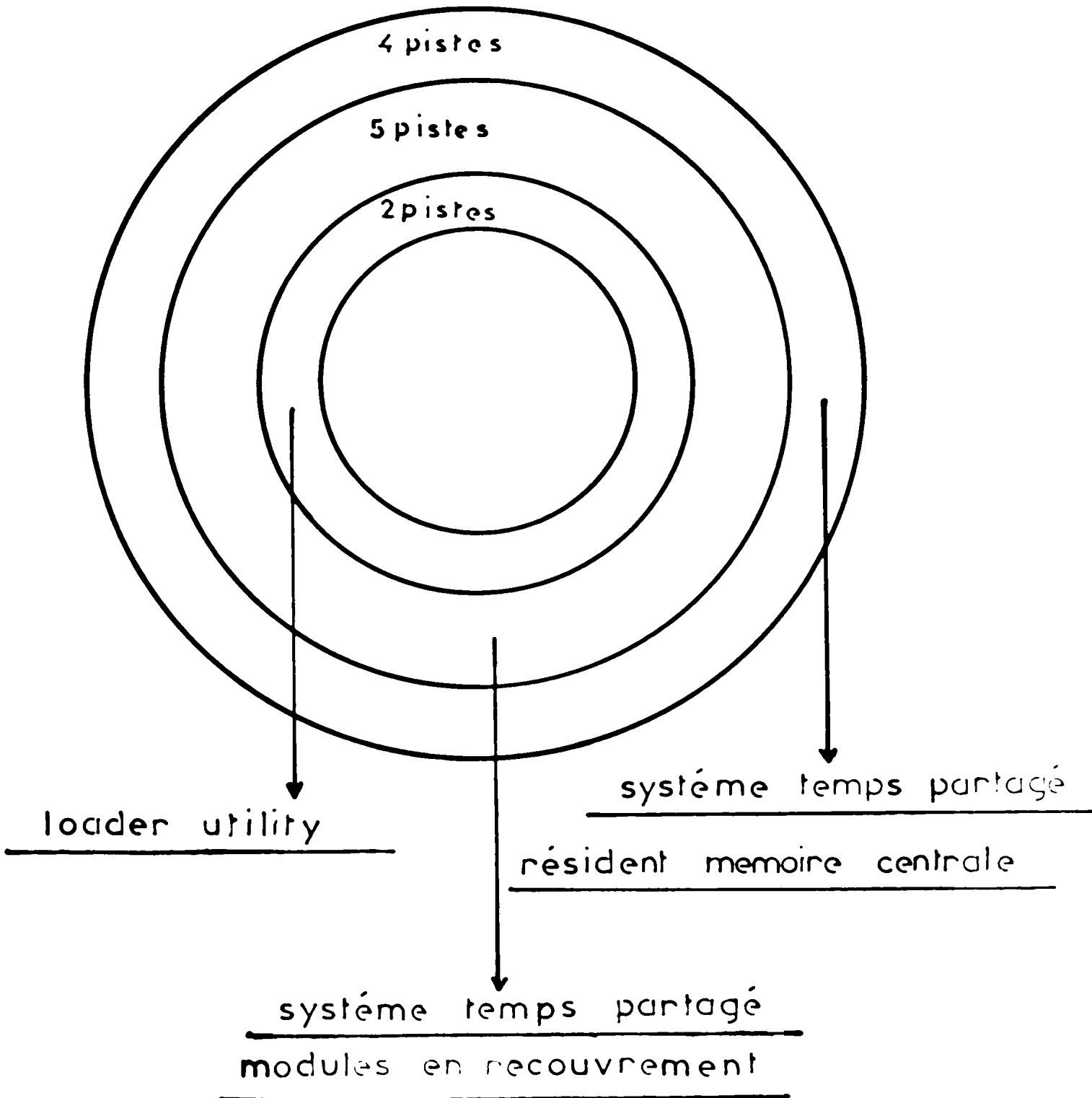
Pour sauvegarder un programme nous lui assignons un nom par la commande "NAME" et nous utilisons la commande "SAVE".

C'est au moment où l'on fait cette sauvegarde que l'ensemble du programme qui avait été précédemment compilé "partiellement" et rangé dans la zone de compilation partielle du disque, se décompile et se range sous forme A.S.C.I.I. dans une autre zone du disque.

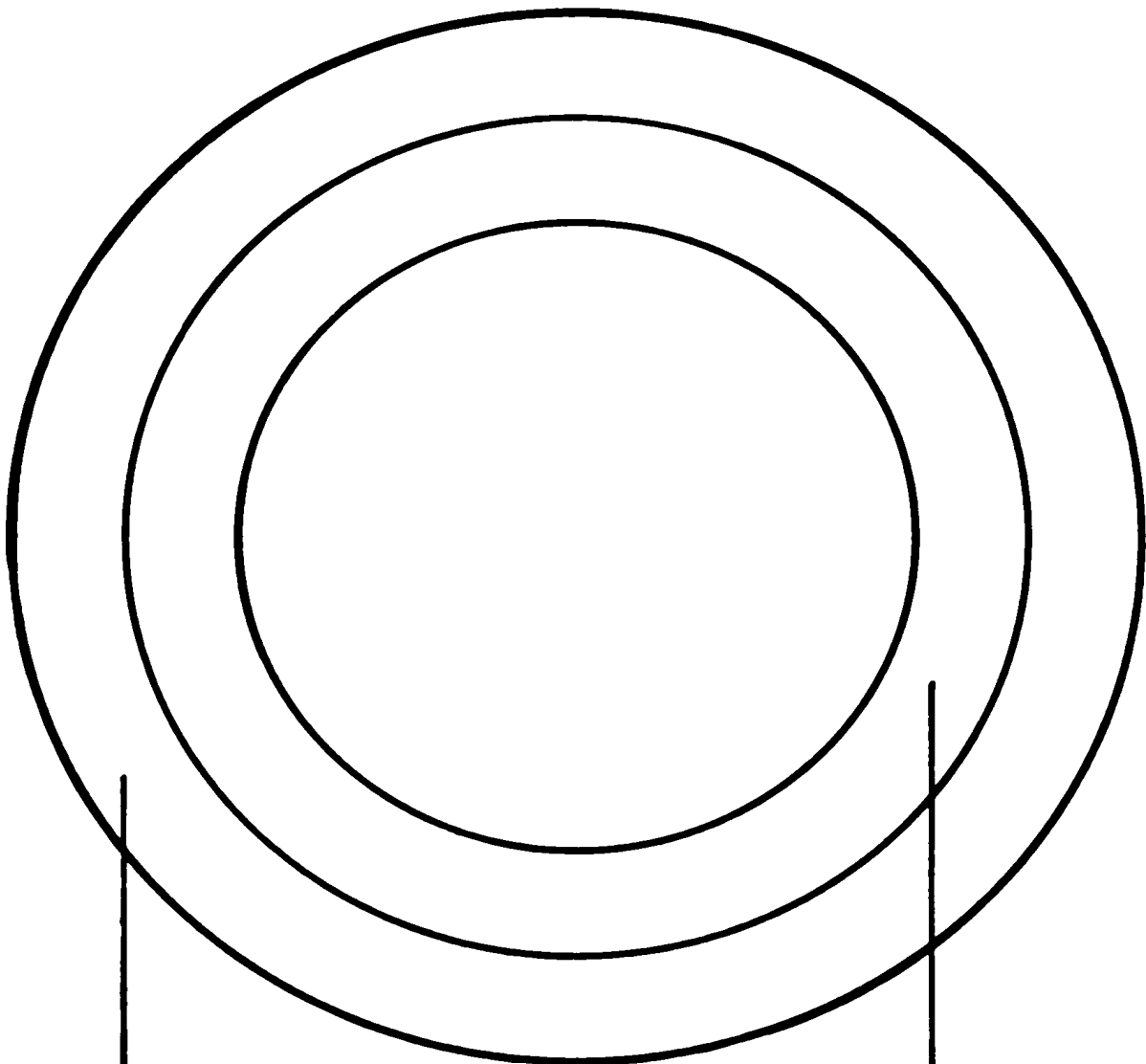
Dans la figure N° 4, on a matérialisé le disque fixe, avec les deux parties fixe et en recouvrement du système temps-partagé ainsi que le "LOADER UTILITY" sous forme binaire.

La figure N° 5 montre le disque amovible avec, d'une part, la zone réservée pour les programmes partiellement compilés, et d'autre part la zone des programmes sauvegardés sous forme A.S.C.I.I.

disque fixe



disque amovible



zone partitionnée  
entre les utilisateurs

zone A·S·C·I·I de nombre  
de pistes variable

### I.1.3. GESTION DES TAMPONS, ZONE DES TAMPONS

A partir de l'adresse 12253 commence ce que nous avons appelé la zone des tampons, nous y distinguerons 3 parties

#### - Zone du tampon fichier FILBF (file buffer)

De longueur 13<sub>10</sub> mots, c'est le tampon de gestion de fichiers. Ce tampon est utilisé par le module "%FLB" qui est en recouvrement et correspond à la gestion des fichiers, c'est-à-dire qu'il procure au compilateur des informations concernant l'adresse disque ainsi que la longueur du fichier.

#### - Zone du tampon message MSQHD

De longueur 39<sub>10</sub> mots, ce tampon reçoit les messages d'une des consoles multiplexées au pupitre.

Ce tampon est pris essentiellement en charge par le module "%MESS" qui est en recouvrement et qui est responsable du traitement des messages.

#### - Zone des tampons utilisateurs

A chaque utilisateur correspond un tampon dont la taille est de 62 mots, de sorte que pour le nombre total d'utilisateurs (16<sub>10</sub>), il existe une zone de 1440 mots.



- Zone du tampon pupitre

Avant de décrire cette zone nous précisons que :

. L'instruction assembleur "JSB M"(jump to subroutine) provoque un branchement inconditionnel au module "M".

. L'étiquette "TTY35" définit l'adresse du module qui traite le driver du pupitre.

La variable "T35F1" employée dans ce driver peut prendre les 2 valeurs 0 ou 1, pour indiquer, qu'il s'agit respectivement soit d'une entrée, soit d'une sortie du pupitre.

Après ces 3 précisions, nous abordons la zone du tampon pupitre.

C'est une zone variable qui sera remplie avant de sortir un message quelconque sur le pupitre, c'est-à-dire avant de faire "JSB TTY35, I" pour faire appel au driver télétype et de positionner T35F1 = 1 pour la sortie d'un message registre B (cf. Annexe 1) pointera sur le mot de ce tampon à partir duquel le prochain caractère sera pris pour être sorti.

Ainsi à tout moment le module "TTY35" peut repérer les points d'entrée des différents messages à sortir, dans n'importe quelle zone de la mémoire (l'adresse ayant été préalablement mise dans le registre B).

Il n'y a donc pas de zone fixe pour les messages télétype, c'est pourquoi nous l'avons appelée zone tampon variable.

La figure N° 6 montre l'adresse mémoire des 3 zones de tampons fichier "FILBF", message "MSQHD" et le tampon utilisateur.

```
1130 12253          ORG 14000B-853
1131*
1132**  DEBUT DE LA ZONE DES BUFFERS
1133*
1134*
1135 12253 000000  FILBF BSS 13
1136*
1137 12270 000000  MSQHD BSS 39
1138*
1139 12337 000000  BUF00 BSS BUFLN
1140 12421 000000  BUF01 BSS BUFLN
1141 12503 000000  BUF02 BSS BUFLN
1142 12585 000000  BUF03 BSS BUFLN
1143 12647 000000  BUF04 BSS BUFLN
1144 12731 000000  BUF05 BSS BUFLN
1145 13013 000000  BUF06 BSS BUFLN
1146 13075 000000  BUF07 BSS BUFLN
1147 13157 000000  BUF10 BSS BUFLN
1148 13241 000000  BUF11 BSS BUFLN
1149 13323 000000  BUF12 BSS BUFLN
1150 13405 000000  BUF13 BSS BUFLN
1151 13467 000000  BUF14 BSS BUFLN
1152 13551 000000  BUF15 BSS BUFLN
1153 13633 000000  BUF16 BSS BUFLN
1154 13715 000000  BUF17 BSS BUFLN
1155 13777          BSS 0
1156*
1157**  FIN DE BUFFERS
```

## I.2. GESTION DYNAMIQUE DU SYSTEME

Par gestion dynamique nous entendons la description du système après sa mise en marche, c'est-à-dire le déroulement des évènements qui ont lieu quand le système a été lancé.

Pour cela nous aurons à étudier le rôle extrêmement important du "SCHEDULER", la notion d'affectation du temps, le déroulement des mises en garde des modules et des commandes qui sont en recouvrement et enfin la gestion de la file d'attente.

Avant d'entrer dans ces détails de la gestion dynamique, nous procédons à la classification de l'ensemble des commandes et de leurs extensions (cf. conclusion de la thèse). L'organisation de ces commandes est indiquée sur la figure N° 7.

Pour une plus grande facilité de compréhension la table des commandes est divisée en 2 parties ; la première partie est elle-même partagée en trois sections.

- les commandes de la première section sont celles directement exécutables sans nécessité d'accès au disque.

- les commandes de la deuxième section sont appelées commandes du compilateur. Ces commandes se trouvent dans la mémoire centrale mais pour être exécutées elles ont besoin de programmes qui doivent être chargés à partir de la mémoire disque.

- les commandes de la troisième section sont les commandes système. Se trouvant en permanence dans la mémoire disque leur usage nécessite à chaque fois un appel disque.

La deuxième partie de la table des commandes contient d'une part l'adresse de début des commandes directement exécutables et résidant en mémoire centrale. Elle contient d'autre

```

1166                                LST
1167**
1168** TABLE DES COMMANDES
1169**
1170*  PREMIERE PARTIE
1171*
1172*  SECTION 1
1173*
1174  35746                                ORG 35746B
1175  35746 044121  COM1  OCT 44121
1176  35746                                SCR  EQU COM1
1177  35747 046017  OCT 46017
1178  35750 024230  OCT 24230
1179**
1180**
1181*  SECTION 2
1182*
1183  35751 043215  COM2  OCT 43215
1184  35752 026422  LIS   OCT 26422
1185  35753 037215  FUN   OCT 37215
1186  35754 056764  XFUN  OCT 56764
1187  35755 177777  CTAPR OCT -1
1188**
1189**
1190*  SECTION 3
1191*
1192  35756 177777  COM3  OCT -1
1193  35757 177777  OCT -1
1194  35760 177777  OCT -1
1195  35761 177777  COMCH OCT -1
1196  35762 044025  SAVE  OCT 44025
1197  35763 177777  SAVO  OCT -1
1198  35764 014223  OCT 14223
1199  35765 000757  OCT 757
1200  35766 016213  HELLO OCT 16213
1201  35767 177777  OCT -1
1202  35770 003404  BYE   OCT 3404
1203  35771 024413  OCT 24413
1204  35772 042215  OCT 42215
1205  35773 032014  OCT 32014
1206  35774 004023  CAT   OCT 4023
1207  35775 026401  LIBRY OCT 26401
1208  35776 006213  OCT 6213
1209  35777 046414  OCT 46414
1210  36000 006422  OCT 6422
1211  36001 030222  OCT 30222
1212  36002 037056  OCT 37056
1213  36003 050657  OCT 50657
1214  36004 034744  OCT 34744
1215  36005 177777  OCT -1
1216  36006 026215  OCT 26215
1217  36007 010107  OCT 10107
1218  36010 142217  COM4  OCT 142217
1219  36011 106421  OCT 106421
1220  36012 142722  ROS   OCT 142722
1221  36013 142222  OCT 142222

```

PAGE 0006 #01

1222	36014	144744		OCT	144744
1223	36015	136721		OCT	136721
1224	36016	104340		OCT	104340
1225	36017	100655		OCT	100655
1226	36020	144544		OCT	144544
1227	36021	177777		OCT	-1
1228	36022	177777		OCT	-1
1229	36023	132226		OCT	132226
1230	36024	124413		OCT	124413,-1
1231	36026	130725		OCT	130725
1232	36027	177777		OCT	-1
1233	36030	177777		OCT	-1
1234	36031	137221		OCT	137221,-1
1235	36033	136356		OCT	136356
1236	36034	106422	DISCC	OCT	106422,-1
1237**					
1238	36036	177777	START	OCT	-1
1239	36037	177777		OCT	-1
1240	36040	177777		OCT	-1
1241**					
1242**					
1243*	DEUXIEME PARTIE				
1244*					
1245	36041	035176	COM5	DEF	#SCR
1246	36042	035213		DEF	#TAP
1247	36043	035026		DEF	SCH20
1248	36044	014215		DEF	CMFLE
1249	36045	014216		DEF	LIST
1250	36046	014217		DEF	PUNCH
1251	36047	014220		DEF	XPUNCH
1252	36050	014005		DEF	TAPER
1253**					
1254**					
1255	36051	000000	COM6	BSS	COM5-COM3
1256	36053		FILIB	EQU	COM6+2
1257	36054		CHLIB	EQU	COM6-COM3+COMCH
1258	36131		TSBLB	EQU	COM6-COM3+START
1259	36132		RSLIB	EQU	COM6-COM3+START+1
1260	36133		SWLIB	EQU	COM6-COM3+START+2
1261	36134	000000	DSERA	BSS	6
1262	36142	000000	LOGGR	BSS	32
1263*					
1264**	FIN DE TABLE DE COMMANDES				

part, l'adresse de début des commandes, dites commandes du compilateur.

Enfin elle contient l'adresse disque pour les commandes qui résident sur le disque.

### I.2.1. LE ROLE DU REPARTITEUR ET L'INTERRUPTION D'HORLOGE

Le driver d'horloge est le point central du contrôle. En effet l'initialisation de cette horloge est faite de telle sorte que toutes les 100 ms (un temps réel et scalaire) l'horloge provoque une interruption et à ce moment là le contrôle est passé au répartiteur (SCHEDULER).

Le travail principal du répartiteur de ce système de temps-partagé consiste à voir quel est le prochain programme qui doit être exécuté et ceci toutes les 100 ms.

Il est à remarquer qu'outre le fait de passer le contrôle au répartiteur le driver de l'horloge a aussi pour tâche la mise à jour de la date.

Ainsi toutes les 100 ms et après une interruption faite par l'horloge le répartiteur procède à l'exécution d'un autre programme qui avait été précédemment mis en file d'attente.

Dans le cas où un programme dépasse le temps maximum de 100 ms, l'exécution sera interrompue et le programme sera remis au bout de la file d'attente (cf. I.2.3), donc la priorité la moins élevée lui sera affectée et sa place précédente dans la file d'attente sera effacée.

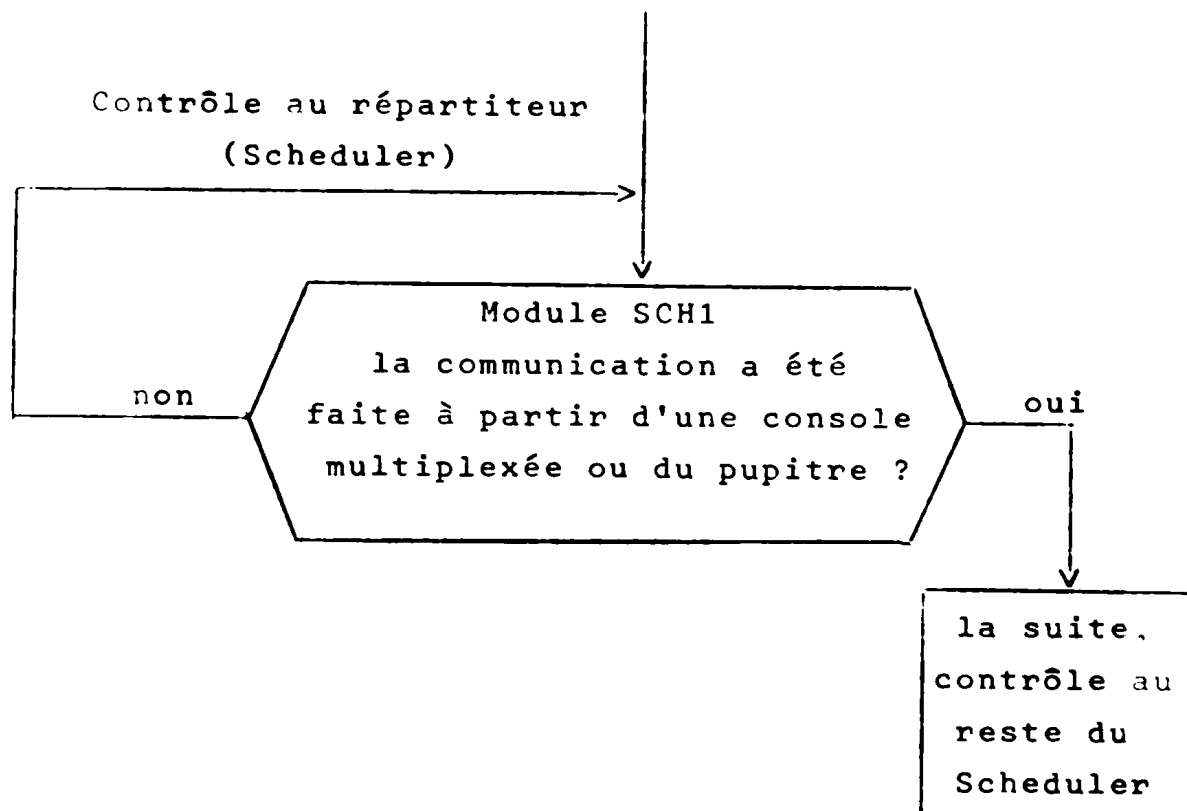
## Le mécanisme dynamique du système

En I.1 (fonctionnement statique du système), nous avons décrit la mise en place soit dans la mémoire centrale, soit dans la mémoire disque, de la partie fixe et de la partie de recouvrement du système ainsi que la gestion du fichier des utilisateurs.

Dans la suite nous reparlerons de toutes les parties détaillant le système de temps-partagé à l'état dynamique.

Une fois le système de temps-partagé lancé (cf. II.1), le contrôle est entièrement laissé au répartiteur qui boucle en permanence sur une partie de lui-même et cici, tant qu'il n'y a rien à faire ; c'est-à-dire tant qu'aucune communication n'a lieu soit entre une console multiplexée et le système, soit entre le pupitre et le système.

Le petit organigramme qui suit, schématise bien ceci



Dès qu'une communication a lieu, dans la suite de "SCHEDULER", le module "SCH54" effectue des tests pour s'assurer que cette communication a lieu à partir d'une console ou du pupitre.

Si c'est un message en provenance du multiplexeur le module "SCH5" procède à une étude détaillée pour vérifier la nature de la communication, ainsi ce module étudie tous les cas tels que ...avortement, commande...etc, (cf. II.2).

Dans le cas où la communication consiste en une ligne syntaxiquement correcte et que de plus ce n'est pas une commande, le module "SCH9" l'analyse et la traite (cf. II.2.3, remarque importante).

Il faut noter que dans le cas d'un jumelage du système avec le compilateur d'un autre langage que le "BASIC", il convient de procéder à des modifications de cette partie du "SCHEDULER" (-SCH9). En effet le module "SCH9" vérifie que la ligne commence par des chiffres (toutes les instructions sont étiquetées en "BASIC"), ce qui n'est pas une obligation pour d'autres langages.

Une fois terminés l'analyse et le traitement par le module "SCH9", le sous-programme "INSEQ" prend le contrôle, affecte la priorité nécessaire et place le résultat dans la file d'attente (cf. I.2.3).

A cet instant du fonctionnement le contrôle est passé au module "SWAPR" qui fait la mise en garde du résultat sur le disque et, à nouveau, le module "SCH1" du "SCHEDULER" reprend le contrôle du système et refait les mêmes tests qu'au tout début pour voir s'il y a quelque chose à faire.

Comme nous le verrons dans l'organigramme (cf. Figure N° 8) et comme nous l'avons d'ailleurs indiqué plus haut, au moment où une communication a lieu avec le système, un test important est effectué pour déterminer sa provenance, soit d'une console multiplexée, soit du pupitre.



Nous détaillons ci-dessous le mécanisme dans ce cas.

Tout d'abord un test a lieu pour déterminer si la communication est un message.

Si c'est le cas, c'est le module "SCH15" qui prend contrôle et par le module "SWAPR" fait la mise en garde de la commande "MESSAGE" qui comme les autres commandes est en permanence et réside sur disque ; la commande une fois amenée à la mémoire centrale et exécutée, le contrôle repasse au module fondamental "SCH1" du répartiteur.

Si la communication n'est pas un message une étude détaillée a lieu pour déterminer la nature de la ligne entrée au pupitre (cf. II.2.2) ; si l'étude montre que cette ligne est une commande correcte c'est le module "SCH15" qui prend à nouveau le contrôle en main et par un branchement approprié (cf. Figure N° 8) fait amener par le module "SWAPR" la commande en question dans la mémoire centrale (cf. I.2, classification des commandes).

La commande une fois exécutée, le contrôle passe au module "SCH1" (début du contrôle de "SCHEDULER").

Dans le cas où la ligne entrée au pupitre est incompréhensible, c'est-à-dire si les 2 premières lettres ne correspondent pas à la liste de la table des commandes, un message d'erreur sous forme de 3 interrogations est sorti, et à nouveau le contrôle passe au module "SCH1" du répartiteur pour voir s'il y a quelque chose à faire, soit au niveau d'autres utilisateurs (existence d'une entrée sortie à partir d'autres consoles) soit au niveau de l'exécution d'un autre programme.

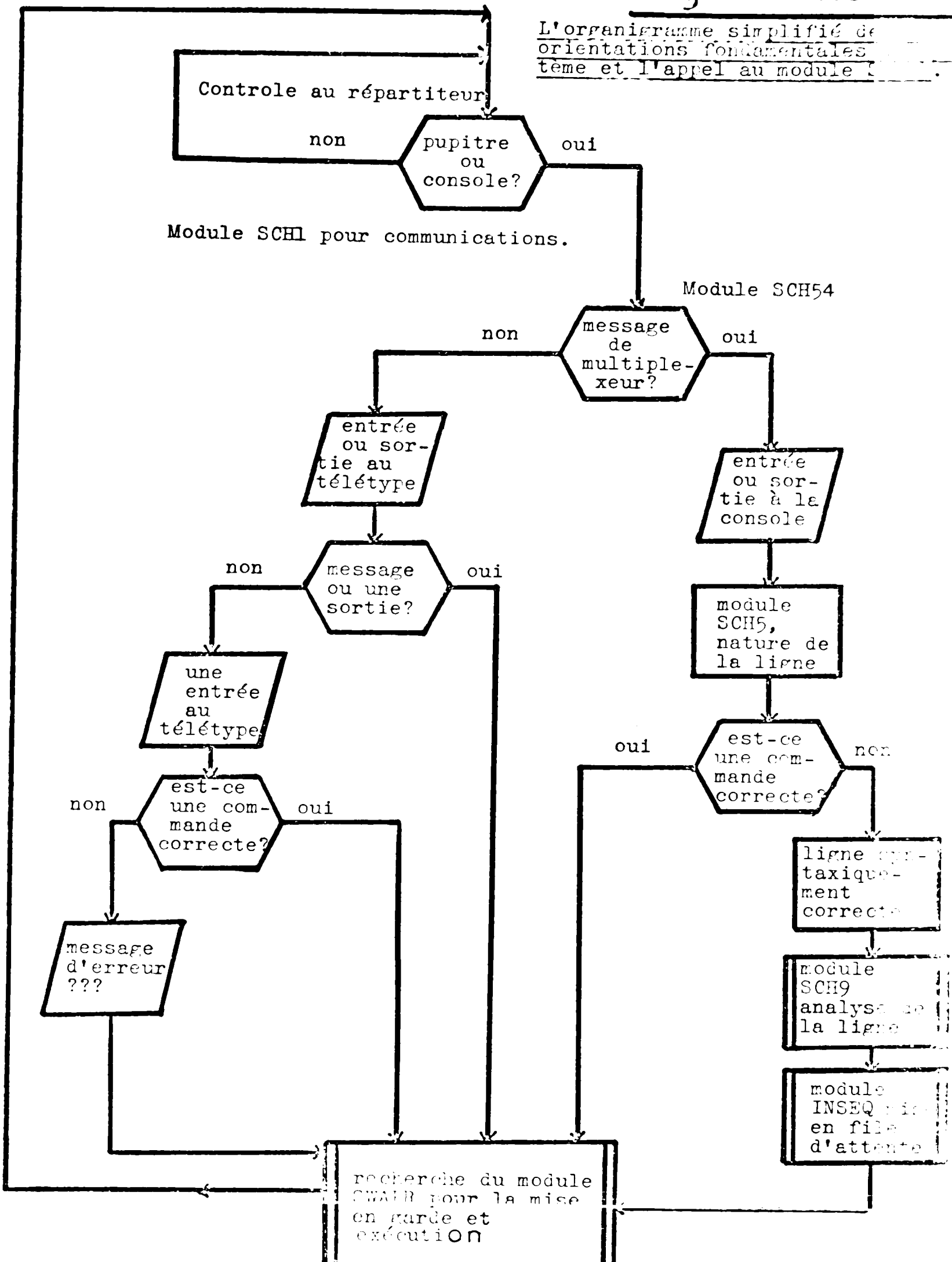
L'organigramme (Figure N° 8) ci-dessous résume l'ensemble de l'étude précédente.

Signalons que le système est doté d'une protection contre les baisses accidentelles de tension.

Après qu'une telle variation de tension ait eu lieu, le système ayant sauvegardé pendant la chute de tension les informations indispensables au redémarrage, l'exploitation peut reprendre normalement, toutes les informations ayant été sauvegardées à l'exception de celles qui étaient en cours de transmission au moment de la panne.

figure no 8

L'organigramme simplifié de orientations fondamentales de l'ensemble et l'appel au module SCH1.



Controle au répartiteur

pupitre ou console?

Module SCH1 pour communications.

Module SCH54

message de multiple-xeur?

entrée ou sortie au télétype

entrée ou sortie à la console

message ou une sortie?

une entrée au télétype

Module SCH5, nature de la ligne

est-ce une commande correcte?

est-ce une commande correcte?

message d'erreur ???

ligne syntaxiquement correcte

recherche du module SWAIB pour la mise en garde et exécution

Module SCH9 analyse de la ligne

Module INSEQ mis en file d'attente

### I.2.2. ETUDE D'UN MODULE IMPORTANT "SWAPR" RESPONSABLE DE L'ENSEMBLE DE MISE EN GARDE

Nous venons de voir la façon dont nous faisons appel au module "SWAPR" dans différentes situations au niveau du répartiteur.

Ce module qui est le noyau de l'ensemble des travaux de mise en garde au sein du répartiteur est détaillé par l'organigramme (cf. Figure N° 10) et les explications suivantes.

Dès son appel (cf. Figure N° 9), un premier test a lieu (à l'intérieur du module "SWAPR") pour voir si un transfert a lieu entre la mémoire centrale et le disque.

Si oui, le nouveau transfert n'est pas possible et le retour a lieu vers le module "SCH1", et comme nous l'avons déjà vu à plusieurs reprises, ce retour a lieu pour l'étude d'une éventuelle entrée-sortie à partir d'une autre console.

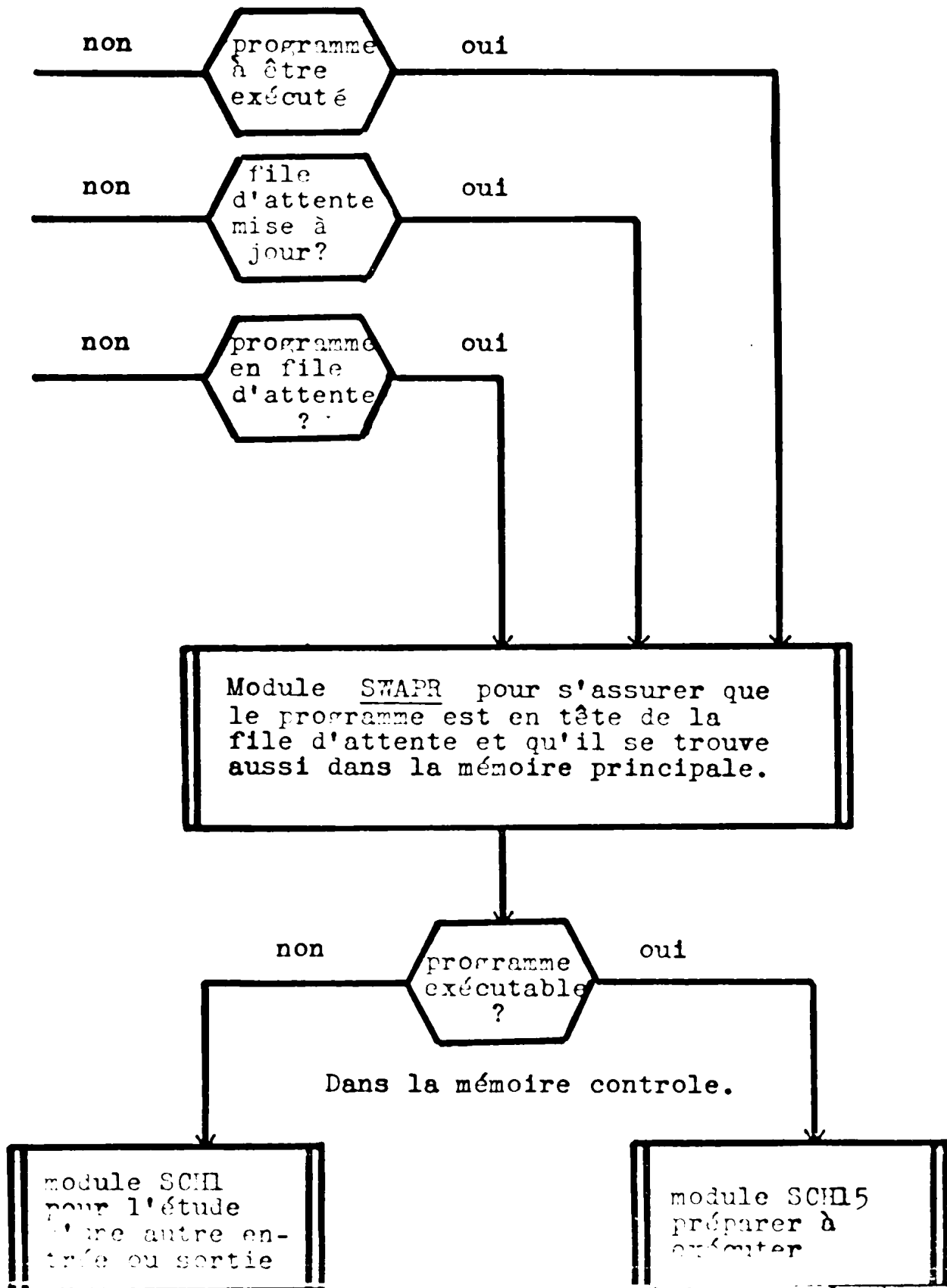
Par contre, si aucun transfert n'a lieu, à ce moment-là, une étude de la file d'attente est entreprise pour voir quel est le prochain utilisateur à traiter (cf. I.2.3).

Si la file est vide, pour les mêmes raisons que celle que nous venons de décrire, le retour a lieu vers le module "SCH1".

La file n'étant pas vide, un nouveau test a lieu pour déterminer s'il s'agit d'une commande. Si ce n'est pas le cas c'est qu'une ligne d'instruction a été traitée et le résultat se matérialise par une sortie sur une console ou sur le pupitre.

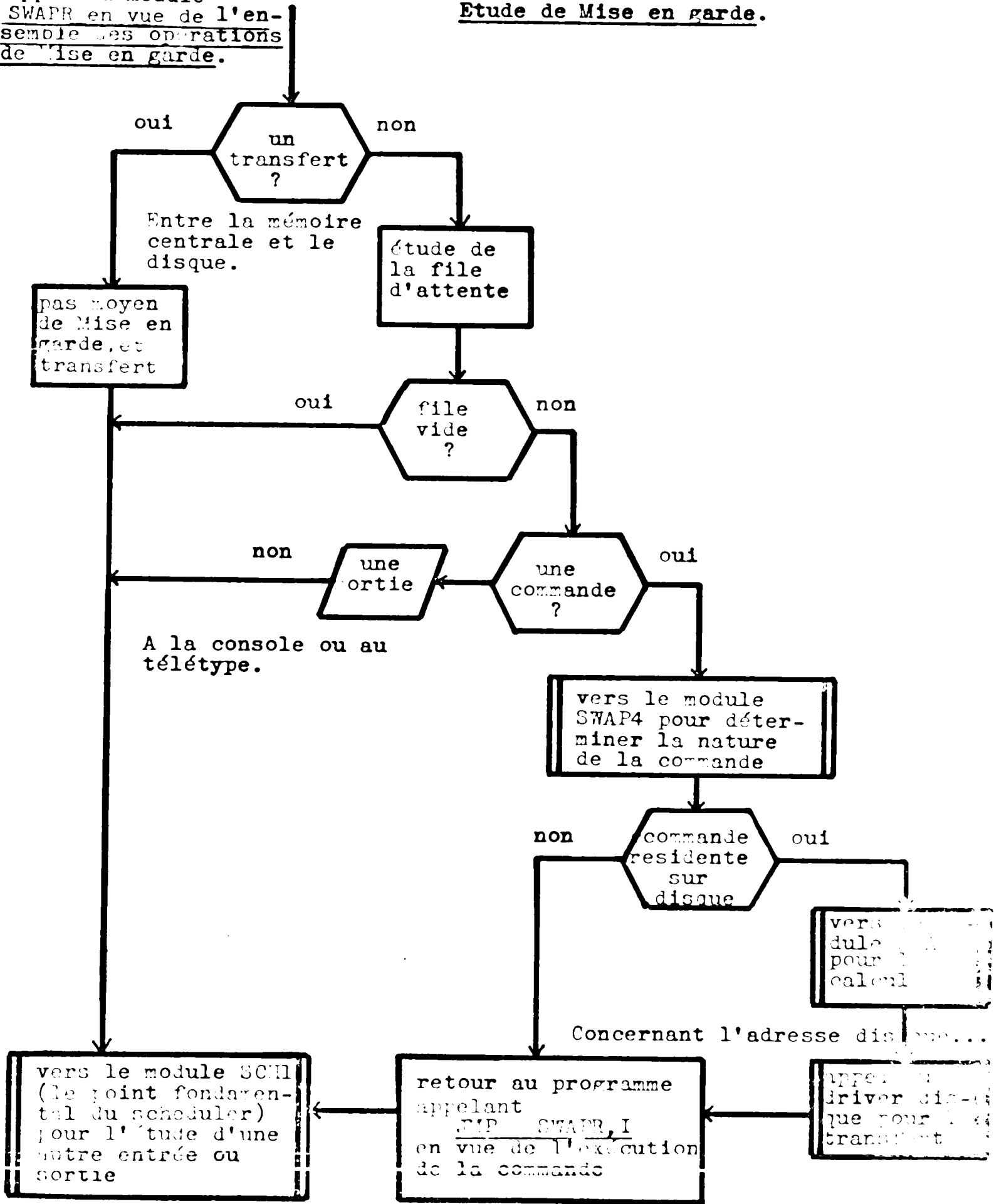
S'il s'agit d'une commande, un branchement a lieu vers le module "SWAP4" qui détermine la classe de la commande (cf. I.2).

Les différentes situations créant la nécessité de faire appel à SWAPR.



Appel au module SWAPR en vue de l'ensemble des opérations de Mise en garde.

Etude de Mise en garde.



Si nous sommes en présence d'une commande résidant en mémoire centrale, le module "SWAP4" retourne au programme appelant qui exécute immédiatement la commande en question et retour a ensuite lieu vers le module "SCH1".

Si la commande réside sur le disque, c'est le module "SWAP3" qui calcule l'adresse et la longueur sur disque de la commande, et le contrôle est passé ensuite au module "DISC" (le driver disque) qui transfère la commande de disque vers mémoire centrale ; le retour vers le programme appelant a ensuite lieu, une fois la commande exécutée, comme le montre l'organigramme (cf. Figure N° 10) ; le contrôle final est de nouveau au module "SCH1".

### I.2.3. MODULE D'INSERTION D'UN UTILISATEUR DANS LA FILE D'ATTENTE

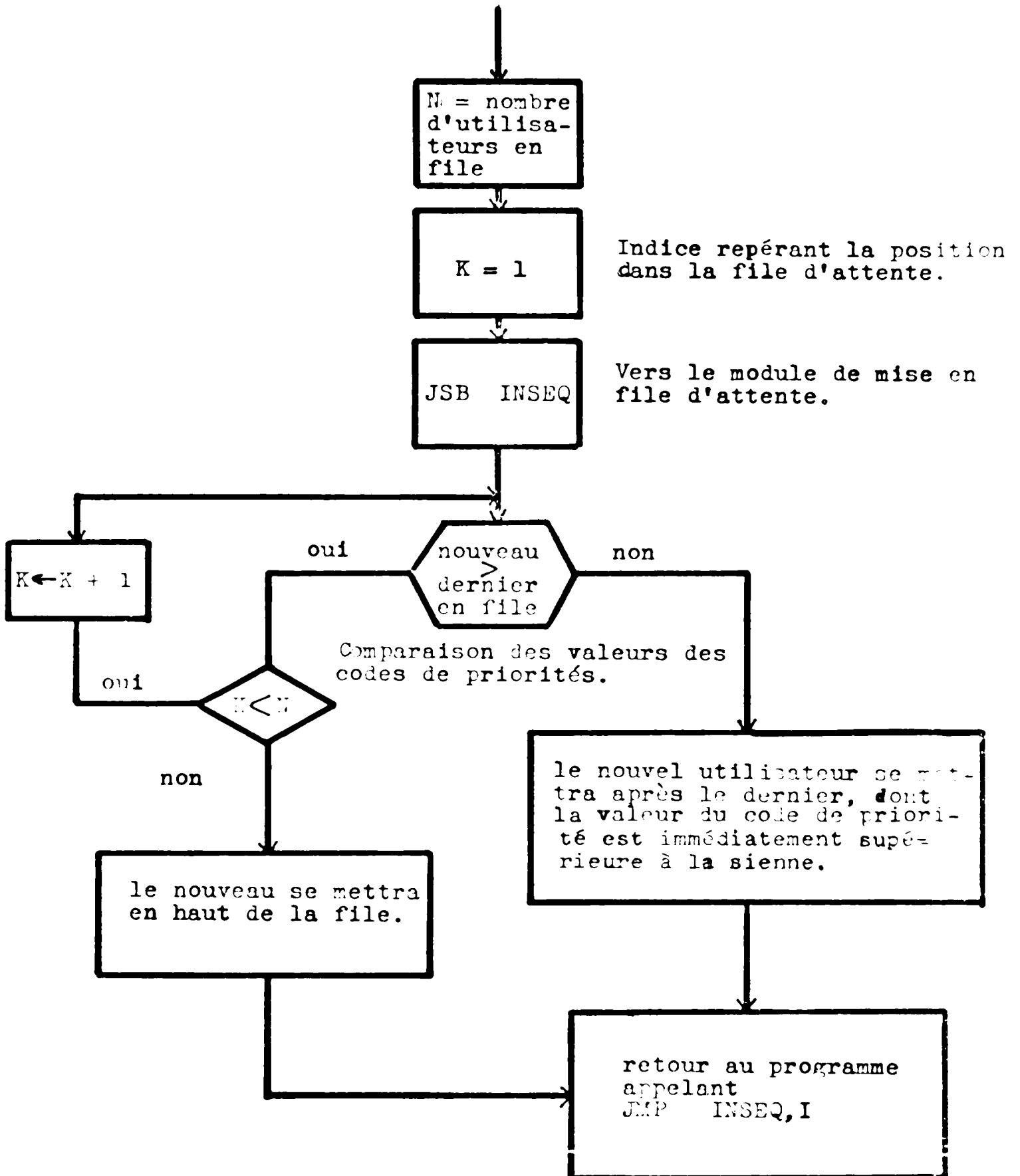
Ce module que nous détaillons par l'organigramme (cf. Figure N° 11), définit la manière dont une tâche se positionne dans la file d'attente.

Les notations k et n employées dans l'organigramme ne figurent pas de façon explicite dans le module correspondant ainsi que dans le système, mais, c'est par un souci de clarté de l'exposé que nous les avons adoptées.

Dès l'arrivée d'un nouvel utilisateur, son code de priorité est comparé avec le code le moins prioritaire dans la file d'attente.

Si le nouveau est moins prioritaire le chaînage se fera de sorte qu'il sera positionné après le dernier en attente, et le retour du système a lieu au programme appelant.

Subroutine d'insertion d'un nouvel utilisation dans la file d'attente.





S'il est plus prioritaire le nouveau remonte, par le chaînage approprié, dans la file jusqu'à ce qu'il se situe correctement dans la hiérarchie des codes prioritaires, la file d'attente étant ainsi mise à jour, le retour a lieu vers le programme appelant.

#### I.2.4. STRUCTURE ARBORESCENTE DU SYSTEME

Notre but, à la fin de cette première partie, consiste à représenter par une structure arborescente le système temps-partagé et ceci pour une meilleure compréhension du fonctionnement de l'ensemble.

Une partie très importante des modules et des sous-modules ne sont mentionnés ni sur les figures ni dans le texte, car leurs interactions avec les autres sont secondaires par rapport aux actions fondamentales des modules indiqués qui, par leurs cheminements arborescents ou bouclés décrivent le fonctionnement global du système.

Sur la figure N° 12, nous avons matérialisé le module "SCH1", car c'est à partir de ce module que le véritable mécanisme dynamique du système commence (cf. I.2.1).

A partir de ce module, un branchement direct a lieu vers le module "SCH54" qui aura pour objet la vérification de la provenance de l'information (du pupitre ou de la console multiplexée).

Un autre branchement a lieu vers le module "SPL", qui a pour principale tâche la mise à jour de la table du télétype.

Le module "SPL" s'oriente vers 4 branchements possibles. D'abord vers le sous-module "SPL1" qui positionne un bit de contrôle quand un utilisateur est activé.



A partir de "SPL1" il existe à nouveau 2 branchements : "SPL5" pour le cas où le programme de la bibliothèque est utilisé et "SPL2" pour le cas où un nouvel utilisateur est activé.

Dans ce dernier cas le contrôle passe au module "DECUI" qui insère l'utilisateur dans la file d'attente.

Deux autres branches, issues de SPL, c'est-à-dire "SPL3" et "SPL4", s'occupent respectivement de la mise au point des modules de contrôle pour le temps utilisé ; ainsi que de la sortie du signal sonore de contrôle.

Le dernier sous-module attaqué par "SPL" ou par "SPL2" est "SPLN". "SPL3", "SPL4" et "SPL5" et "SPLN" a le rôle important de procéder à la mise à jour des pointeurs ainsi que de la table pour l'utilisateur suivant.

Après avoir ajusté l'ensemble des pointeurs et la table des utilisateurs "SPLN" passe le contrôle au module "SCH54".

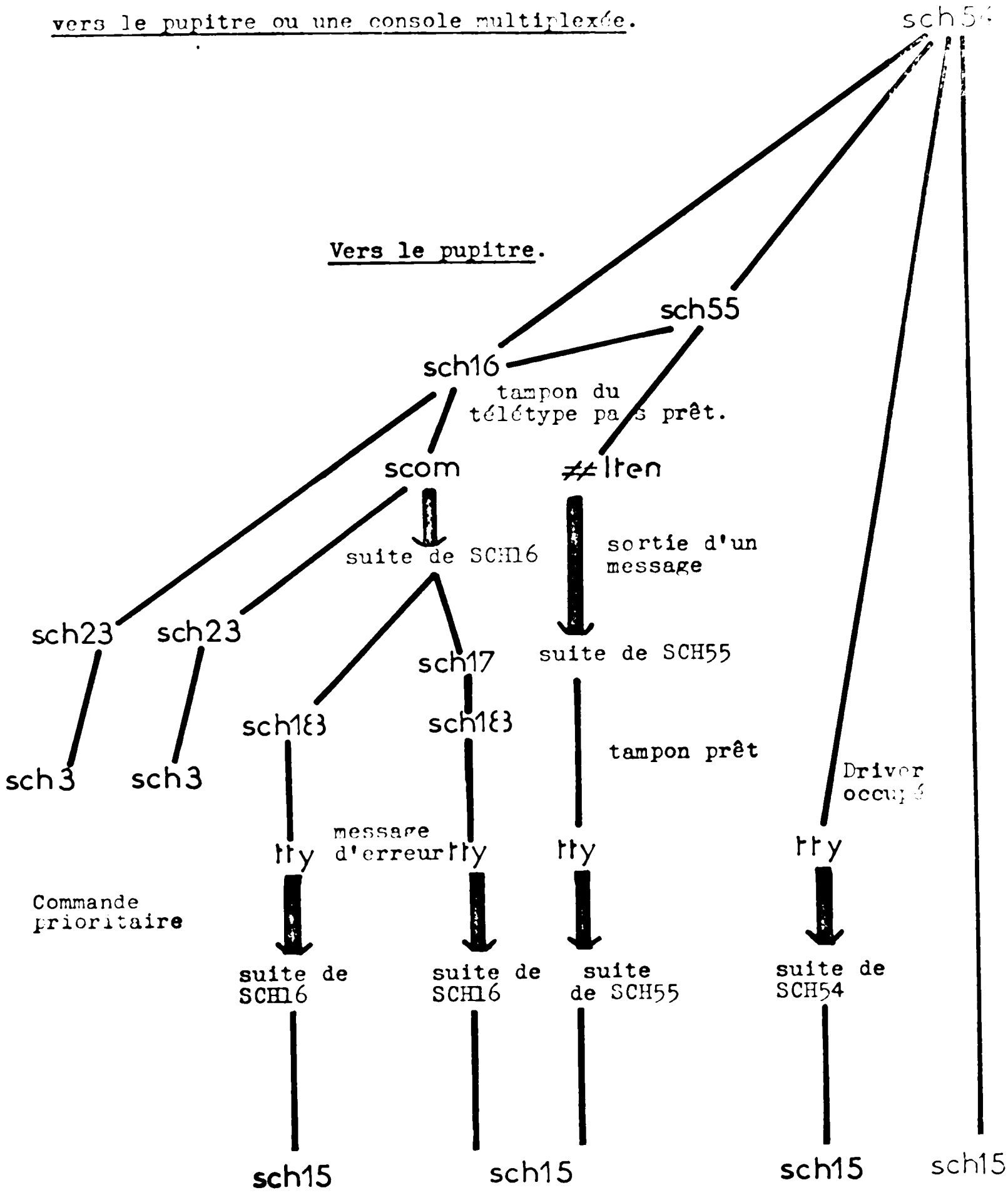
A partir du module "SCH54 (figure N° 13) et une fois que le test a eu lieu pour déterminer la provenance de l'information, 2 branches fondamentales se forment, l'une vers les modules qui traitent le cas des consoles multiplexées et l'autre vers ceux traitant le pupitre.

### Les modules traitant les consoles multiplexées (cf. II.2.1)

Issu du module "SCH54", c'est le module "SCH5" qui constitue la première branche.

A partir de "SCH8", il existe 3 nouvelles orientations (figure N° 13), la première "SCHB", qui par l'intermédiaire de "SCH61" et "SCH14" traite les différents cas de "déconnexion spéciale" (cf. II.2.1) pour passer en dernier lieu le contrôle au module "SCH3" (figure N° 18) pour l'ensemble des travaux de la mise en file d'attente et de la mise en garde, "SWAPPING".

Structure arborescente du système: orientation modulaire  
vers le pupitre ou une console multiplexée.





Remarque importante

Dans les 2 branches fondamentales du système temps-partagé, vers les modules des consoles ou du pupitre (Figure N° 13), nous rencontrons souvent 3 modules importants ; avec des structures à la fois arborescentes et bouclées. C'est le cas du module "SCOM" concernant le traitement des commandes (Figure N° 16), du module "SCH3" pour la mise en file d'attente des travaux de mise en garde (Figure N° 18). C'est également le cas du module "TYPE" pour l'extraction des caractères à partir des tampons.

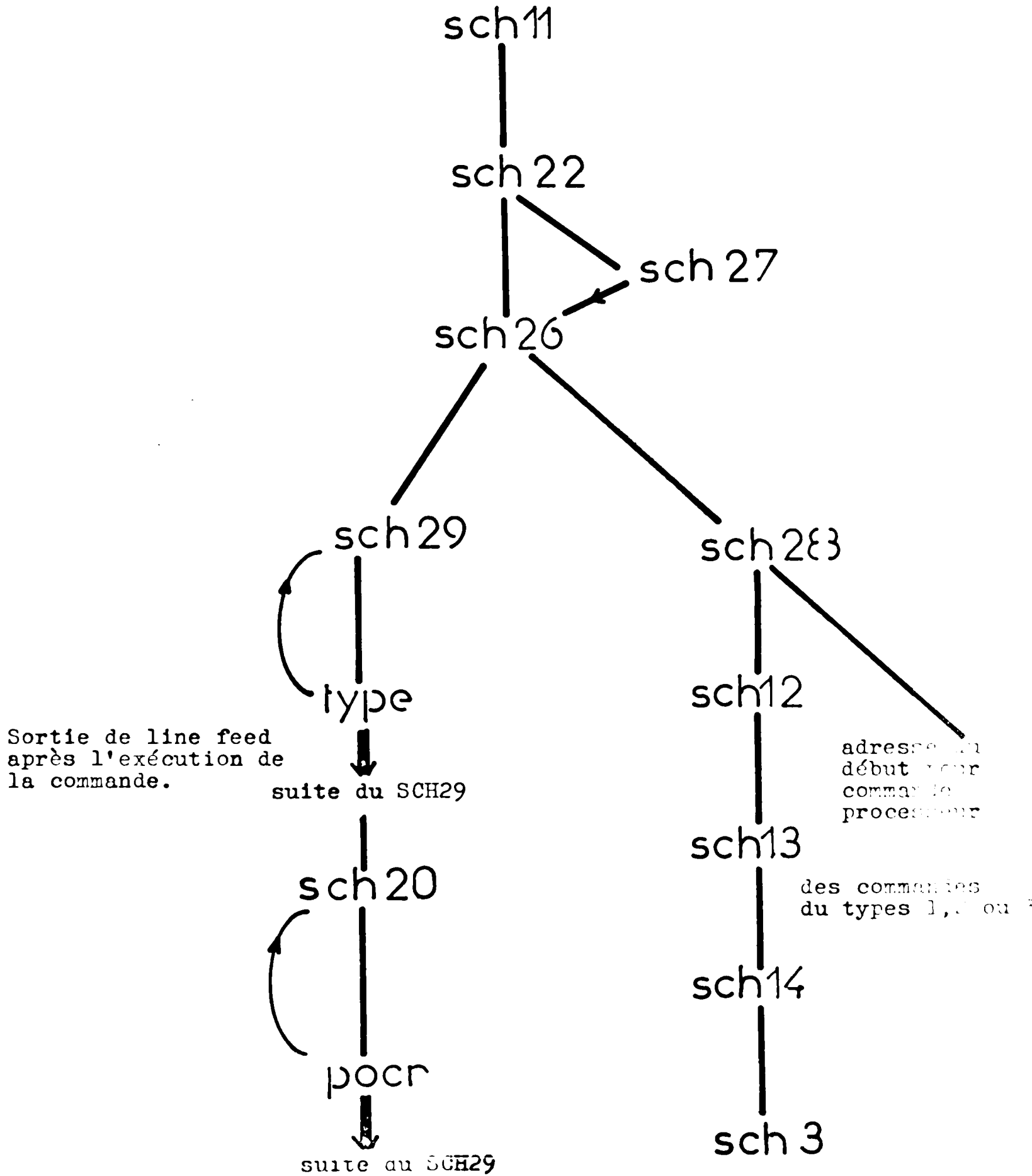
Pour une plus grande clarté du texte nous avons rassemblé les explications concernant ces 3 modules à la fin des traitements des branches des consoles et du pupitre.

Une 2<sup>e</sup> branche à partir du module "SCH5" est constituée par le module "SCH7" qui traite les cas d'abandons. Ainsi, après avoir ramené l'utilisateur de la file d'attente, par le module "DEQUE", et après avoir pris en charge le contenu des tampons par le module "TYPE" (Figure N° 17), un test a lieu pour vérifier si la commande "HELLO" était déjà entrée, et ceci par le module "HTEST". Si ce n'est pas le cas, l'information entrée par l'utilisateur sera écrasée dans la mémoire centrale, par le module "SCRAT", et le contrôle sera repassé au module "SCH5".

La dernière branche, issue du module "SCH5", est le module "SCH6" qui en passant par les modules "SCH24" et "SCOM" traite les commandes. Le branchement a lieu de façon systématique au module "SCH30" pour une entrée de "LINE FEED" ; au module "SCH9" si le 1<sup>er</sup> caractère de l'instruction est un chiffre, au module "EHERR" si c'est une commande invalide et enfin au module "SCH11" dans le cas où il s'agit de l'entrée d'une commande correcte (Figure N° 14).

A partir du module "SCH30", (après l'entrée d'une "LINE FEED") 2 branchements sont possibles pour déterminer si l'utilisateur

L'entrée d'une commande correcte.



a eu lieu ou non par bande perforée. Si ce n'est pas le cas, par les modules "SCH20" et "POCR" un caractère "LINE FEED" est envoyé à la console. Si c'est le cas, (l'entrée a bien eu lieu par la bande perforée) par le module "SCH24", les mémoires temps des autres utilisateurs seront scrutées ; et le même traitement qui vient d'être décrit dans la même branche issue de "SCH6", recommence.

A partir du module "SCH9" (le 1<sup>er</sup> caractère de l'installation est un chiffre), 2 branches se produisent, l'une des branches correspond au cas où la commande "HELLO" n'a pas encore été entrée. Dans ce cas, les modules "SCH25" et "TAPE", prennent en charge la suite, et c'est à la fin de cette branche que le message "PLEASE LOG IN" sortira sur la console concernée.

L'autre branche, issue du module "SCH9", s'oriente vers le module "SCH3" (figure N° 18) qui a pour rôle la mise en file d'attente ainsi que la mise en garde des informations.

Une 3<sup>e</sup> branche à partir de "SCH24" va vers le module "EHERR" qui s'occupe du cas des commandes invalides et par l'intermédiaire du module "TAPE" fera sortir le message "???".

La dernière branche issue de "SCH24" concerne le traitement des commandes qui sont entrées de façon correcte - syntaxiquement et faisant aussi partie de la table des commandes - le traitement commence par le module "SCH11" (Figure N° 14).

Au cours du traitement de ces commandes, en passant par "SCH22" et "SCH27" et à partir de "SCH26", 2 branches distinctes se présentent.

Une 1<sup>ère</sup> branche allant vers les modules "SCH29", "TAPE" et "POCR" a pour but la sortie d'un "LINE FEED", une fois que la commande est exécutée. La 2<sup>ème</sup> branche, à partir de "SCH26", détermine le type de la commande (cf. I.2).

Ainsi les modules "SCH28", "SCH12", "SCH13" et "SCH14" traitent les types 1, 2, ou 3 des commandes.



### Les modules traitant une entrée par le pupitre

A partir du module "SCH54" et avant que l'on aborde le cas du pupitre proprement dit, (Figure N° 13), un test nous permet de vérifier si le driver du pupitre est occupé. Si c'est le cas, c'est qu'un transfert a lieu entre la mémoire centrale et le pupitre ; dans ce cas le branchement direct a lieu au module "SCH15".

Les explications concernant le module "SCH15" ont lieu sur la figure N° 15. A partir de ce module, l'ensemble des travaux de mise en garde et l'exécution des modules, dans la mémoire centrale, ont bien lieu. Ensuite par l'intermédiaire des sous-modules "CLKIN", "CLC1", "SCHED" et "CLKED", une nouvelle interruption d'horloge est prise en compte et en dernier lieu, le contrôle repasse au module "SCH1", qui, comme on l'a déjà vu, (chap. I.2.1) est le début de tous les travaux.

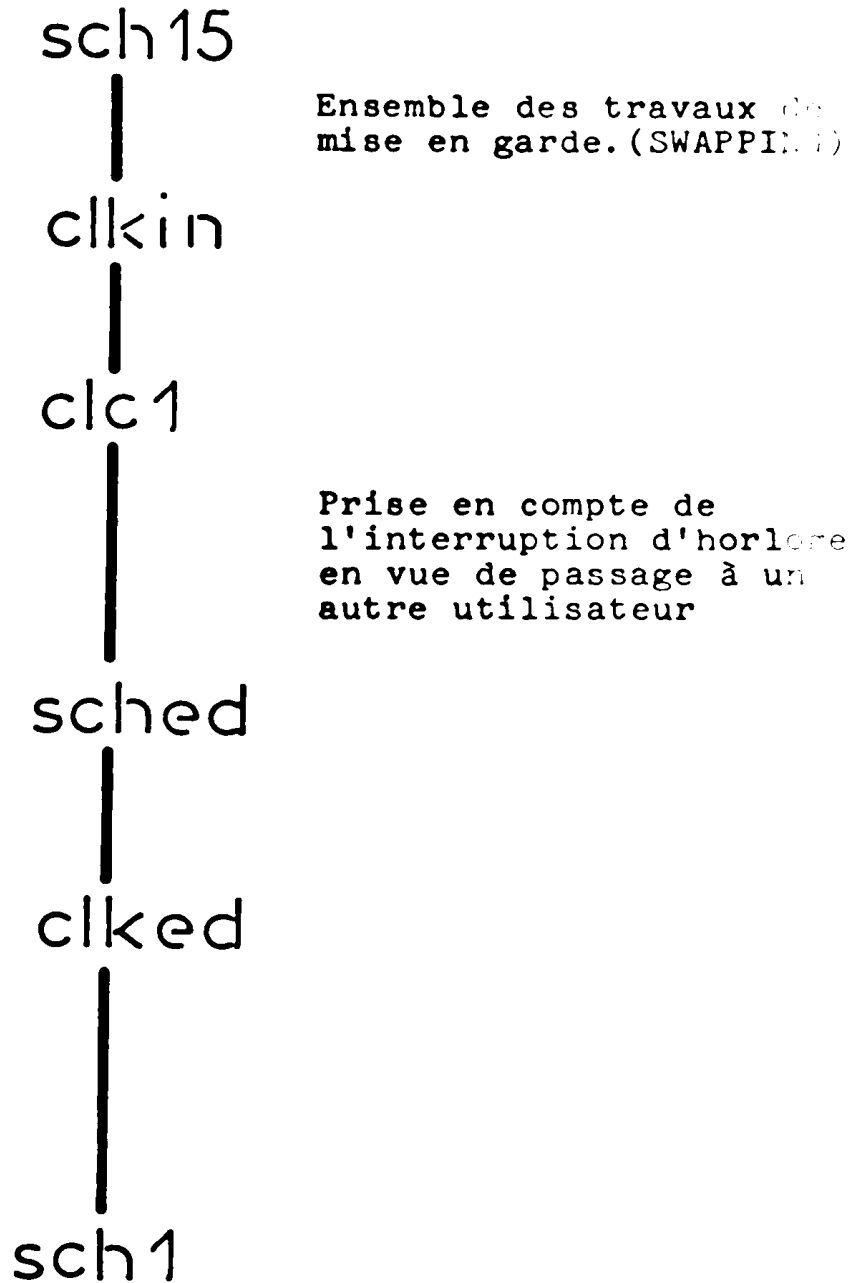
Un autre branchement, à partir du module "SCH54" a lieu vers le module "SCH55" qui, par l'intermédiaire d'autres sous-modules, en particulier  $\neq$  "LTEN" et "TTY" permet de sortir un message au télétype.

Le véritable branchement pour le pupitre commence à partir du module "SCH16", qui d'une part s'oriente vers les modules "SCH23" et "SCH3" (Figure N° 18), en vue d'une mise en file d'attente, d'une commande immédiatement prioritaire, et d'autre part vers le module "SCOM", (Figure N° 16).

A partir du module "SCOM", nous avons le même traitement pour une mise en file d'attente, plus prioritaire, c'est-à-dire vers les modules "SCH23" et "SCH3".

Comme nous le verrons dans le cas particulier de "SCOM", le branchement à partir de ce module a lieu vers les sous-modules "SCH17" et "SCH18" qui traitent le cas de "LINE FEED" et des messages d'erreurs.

Le passage des travaux de mise en garde vers une  
nouvelle interruption d'horloge



Retour à tout au début du système pour voir s'il y a des choses à faire.

Le contrôle est ensuite passé au module "SCH15" (Figure N° 15) qui s'occupera de la mise en garde des commandes en r...  
couvrement, se trouvant sur le disque.

### Module "SCOM"

Pour le traitement des commandes, comme indiqué sur la figure N° 13, c'est le module "SCOM" qui joue un rôle prépondérant.

La structure de ce module étant assez particulière, nous y distinguons 2 formes à la fois arborescente et bouclée.

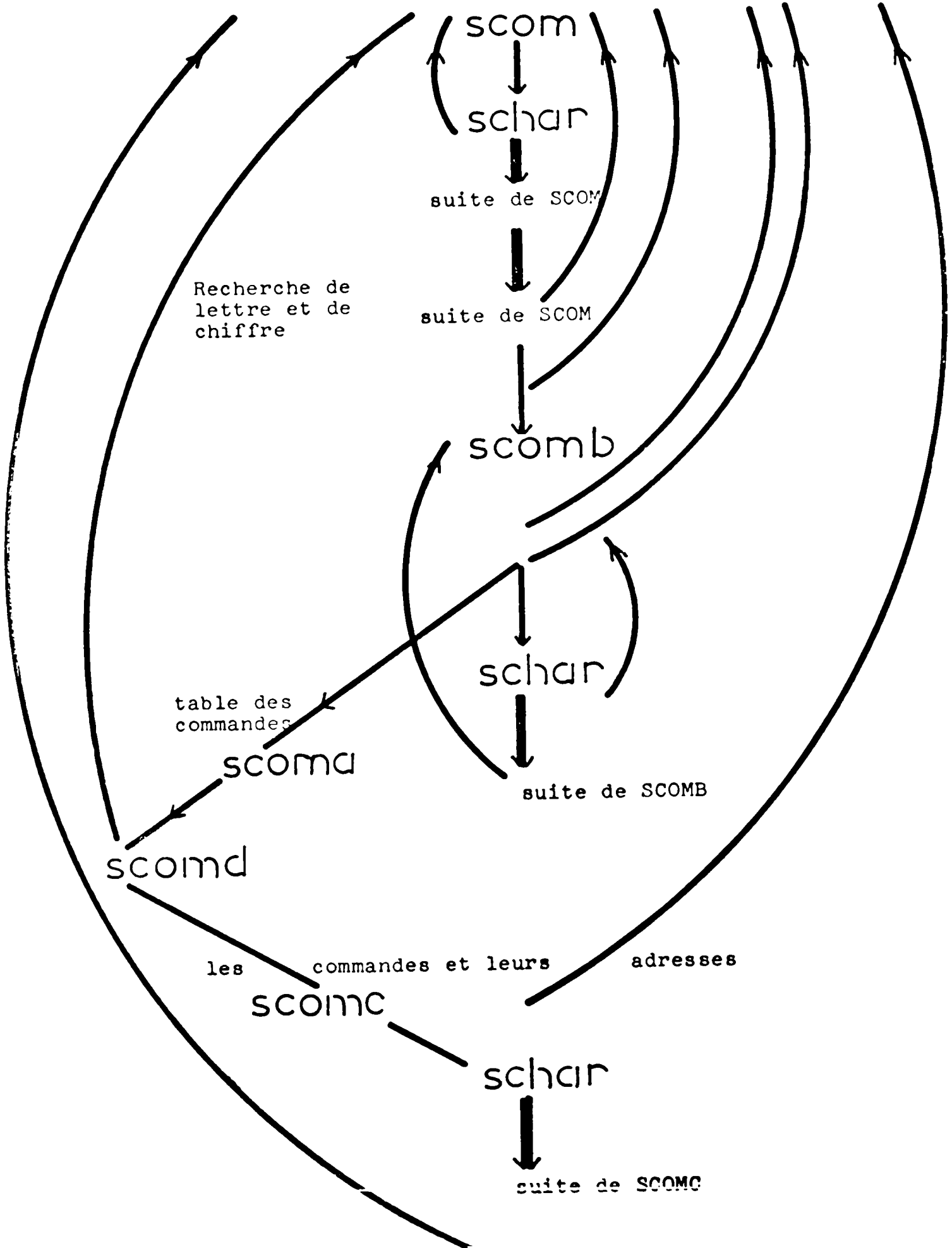
A l'intérieur de ce module un premier appel a lieu vers un sous-module, "SCHAR", qui est responsable de la prise en compte d'un caractère dans un tampon. A l'intérieur de ce sous-module, un test détermine s'il s'agit d'un "LINE FEED" ; si c'est le cas, le retour a lieu au programme appelant, sinon la suite du traitement continue à l'intérieur du module "SCOM". Les 2 tests qui suivent ont pour objet la vérification de l'existence d'un chiffre ou d'une lettre, dans chacun des 2 cas le retour bouclé se produit vers le programme appelant.

Ainsi comme nous pouvons l'observer sur la figure N° 13, la suite du traitement s'effectue en passant par "SCOMA" et "SCOMB", pour l'étude de 3 lettres consécutives (les initiales d'une commande).

Si l'analyse ne rencontre pas 3 lettres, un retour au programme appelant a lieu qui comme nous l'avons déjà décrit dans la structure globale, fera sortir des messages appropriés.

Les branchements vers "SCOMD" et "SCOMC" traitent le cas des commandes trouvées ainsi que la recherche appropriée de leurs adresses.

Traitement des commandes.



### Module "TYPE"

Ce module, figure N° 17, avec ceux qu'il entraîne est le 2<sup>e</sup> module important, avec une structure à la fois arborescente et bouclée.

Ce module a pour tâche, l'envoi d'une chaîne de caractères à une console et ceci en passant par le sous-module "TY" et le module "POC" qui a pour objet le positionnement des caractères dans les tampons des utilisateurs.

Le passage a ensuite lieu au sous-module "OINT", "OINT<sub>1</sub>" et "OINT<sub>2</sub>" qui procèdent à l'initialisation et à la sortie de caractères, (en faisant intervenir le module "MUXR", le drive du multiplexeur).

La sortie une fois terminée, le retour a lieu au module "POCR" qui dans la suite des opérations, traite le cas où le tampon est plein, et ceci par les modules "OBF" et ensuite "SUSP" (responsable de la suspension du travail). Si ce n'est pas le cas, c'est qu'après avoir initialisé la sortie, et avoir envoyé la chaîne des caractères, le retour a lieu au programme qui avait fait appel au module "TYPE".

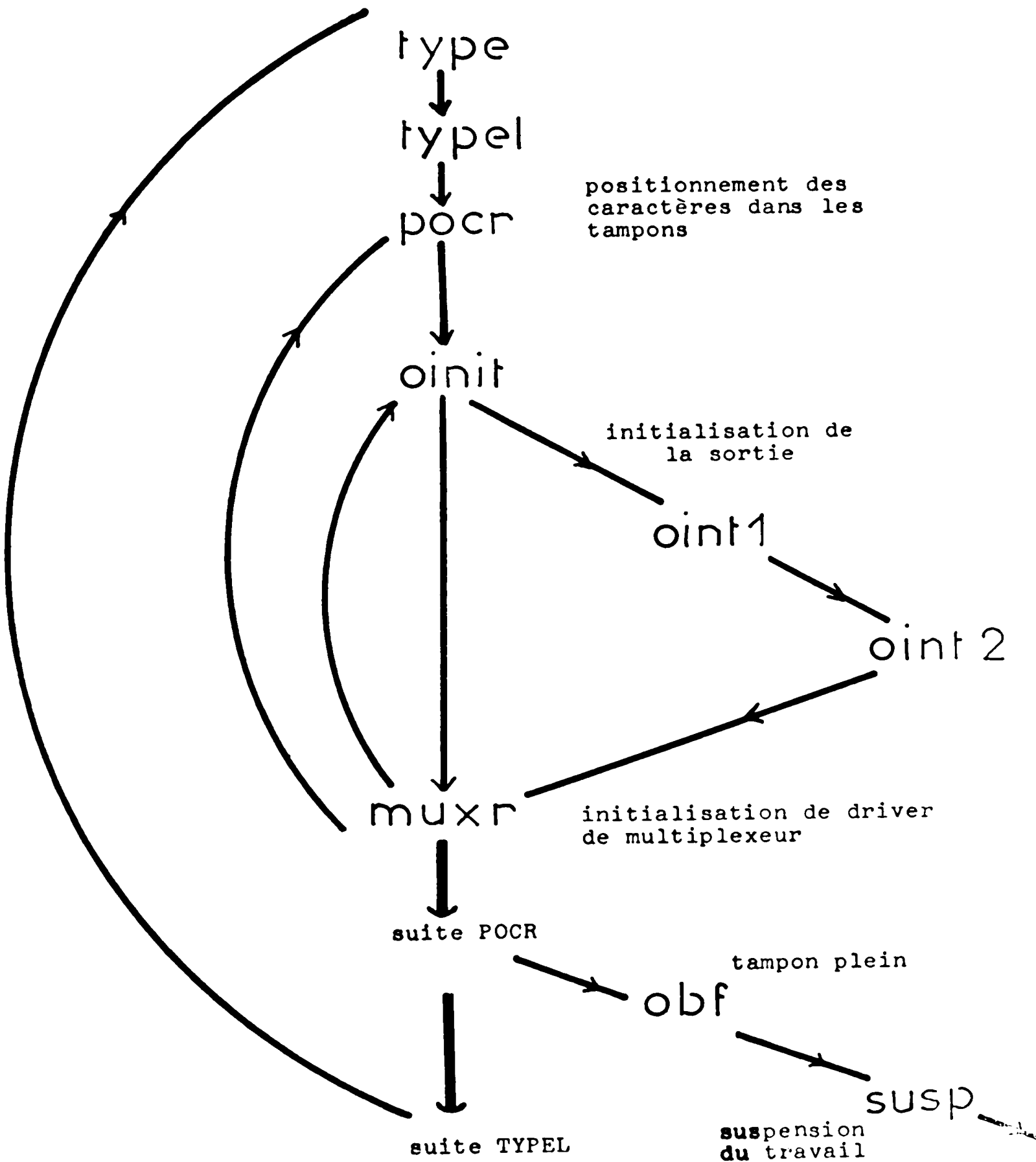
### Module "SCH3"

A partir de ce module (Figure N° 18), s'effectue le travail de la mise en file d'attente et de la mise en garde.

Le branchement a lieu d'abord au module "INSEQ", qui positionne l'utilisation dans la file d'attente. Le retour ayant ensuite lieu à "SCH3" qui se branche au module "SWAPR", chargé de l'ensemble des travaux de mise en garde.

Comme nous pouvons l'observer sur la figure, il existe une relation non arborescente entre "SWAP1", "SWAP2", "SWAP3"

Extraction des caractères,  
à partir des tampons.



et "SWAP4" qui sont des sous-modules issus de "SWAPR", chargés d'initialisation, du transfert entre la mémoire centrale et disque ainsi que de l'exécution des commandes se trouvant déjà dans la mémoire centrale.

C'est à partir de ces sous-modules qu'a lieu l'appel au module "DISC", qui, en tant que driver disque fait le transfert nécessaire entre la mémoire centrale et le disque.

Le transfert terminé, le retour s'effectue au même sous-module. Quel que soit ce sous-module, le branchement se fait au dernier lieu vers le module "SCH1", figure N° 12.

Il est à remarquer que dans le cas du sous-module "SWAP4" et au cas où un programme se trouve déjà dans la mémoire centrale, le retour a lieu au programme appelant.

#### Image simplifiée du déroulement des commandes

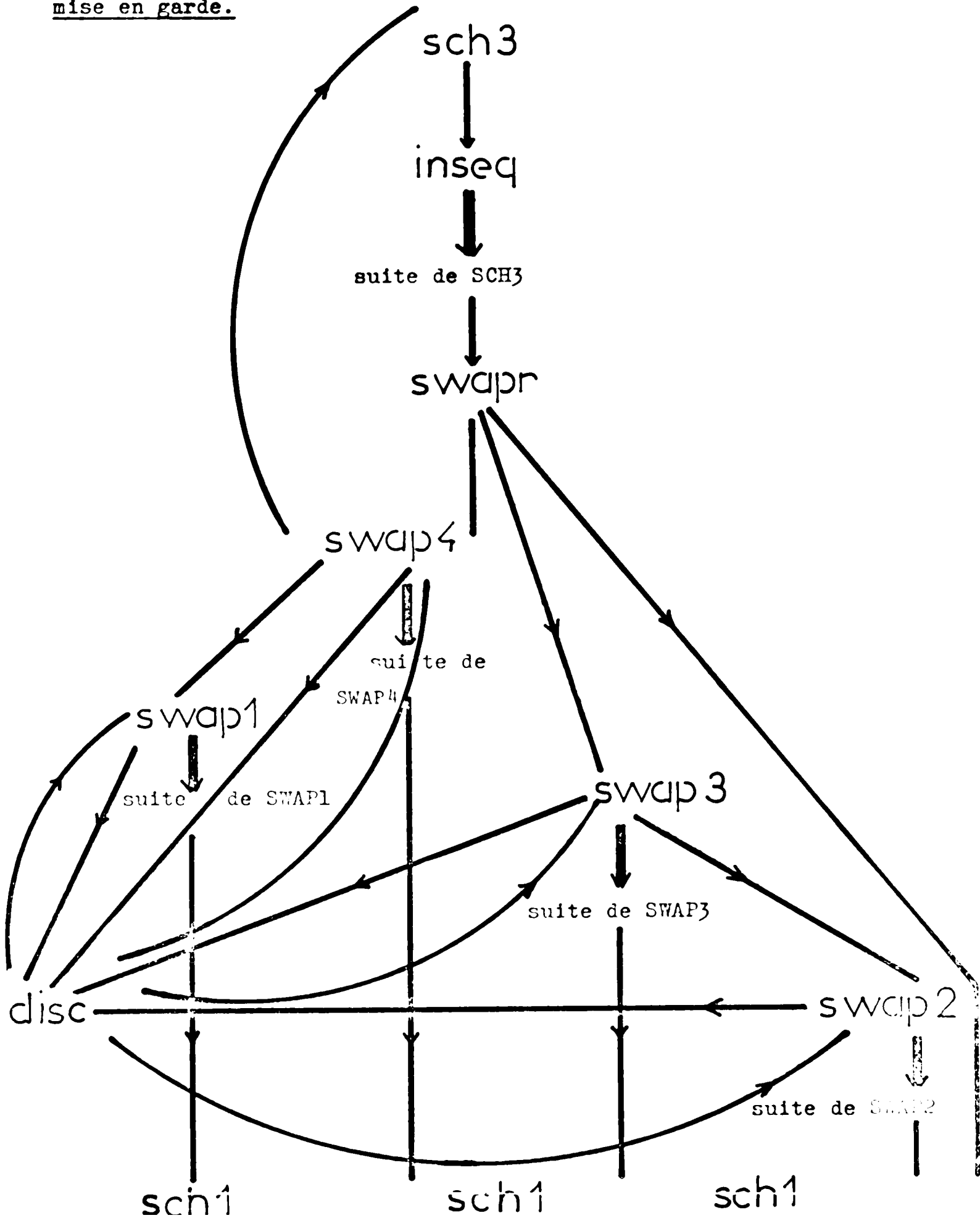
Il nous paraît important, voire indispensable, de décrire de façon succincte, le déroulement d'une commande, une fois qu'elle est correctement frappée.

Nous ne décrivons pas les relations entre les modules et sous-modules, constituant chacune des 32 commandes du système que nous avons exploitées et mises au point. Ces relations sont spécifiques de chaque commande, et n'ont que peu d'intérêt au niveau de la conception du système.

Par contre, par la figure N° 19, et des explications suivantes, le comportement global du système, dès l'instant qu'une commande est correctement frappée sera clairement compris.

Dès l'instant que la commande, en vue de son analyse, est prise en charge, le module "SCH54" oriente le système vers les 2 principaux modules "SCH16" et "SCH5", suivant sa provenance, du pupitre ou de la console multiplexée, respectivement.

File d'attente et la  
mise en garde.

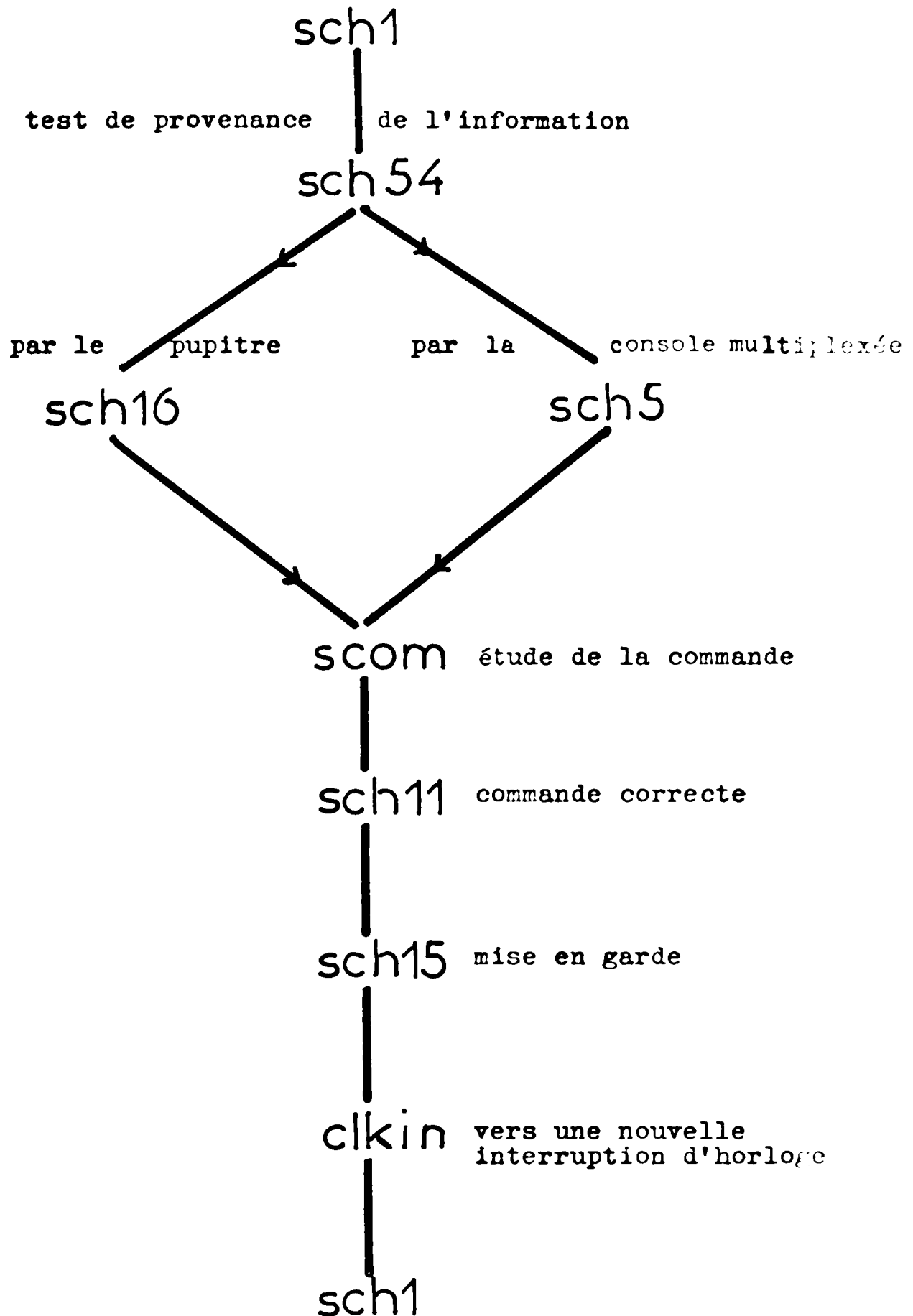




Le module "SCOM", déterminant s'il s'agit d'une commande se trouvant dans la table des commandes (par l'intermédiaire du module "SCH11" qui traite ce cas), fait appel au module "SC" qui, soit par un travail de mise en garde ramène en mémoire centrale la commande se trouvant sur le disque et l'exécute en soit procède à l'exécution de la commande se trouvant déjà la mémoire centrale.

C'est le module "CLKIN" qui détermine ensuite en fonction de l'interruption d'horloge, la prise en charge d'une éventuelle nouvelle commande.

Le cheminement d'une commande, correctement frappée.



### 1.3. COMMUNICATIONS

Nous expliquons de façon brève, les différentes méthodes employées pour les transmissions et les communications entre l'unité centrale et le disque, puis celles liées au multiplexeur et en dernier lieu celles du pupitre avec le système temps-partagé.

#### 1.3.1. COMMUNICATION ENTRE L'UNITE CENTRALE ET LA MEMOIRE DISQUE, DRIVER DISQUE

La transmission se fait par les 2 canaux de "DIRECT MEMORY ACCESS", "DMA", permettant une vitesse de transmission maximale (cf. Annexe 1).

Le canal 11 est destiné à la transmission des données.

Le canal 12 est utilisé à la transmission des commandes.

La communication avec l'unité de disque implique un positionnement de la tête de lecture/écriture qui se fait par boucle d'attente.

En revanche, le transfert proprement dit de l'information se fait par interruption (une fois que l'adresse secteur a été transmise au canal 12).

### I.3.2. COMMUNICATION MULTIPLEXEUR CONSOLES

L'ensemble des communications tant pour les entrées-sorties des consoles d'utilisateurs par la voie du multiplexeur que pour le système temps-partagé se fait par boucle d'attente.

### I.3.3. COMMUNICATION DU PUPITRE AVEC LE SYSTEME TEMPS-PARTAGE

Ici la communication se fait par les 2 méthodes de l'interruption et de la boucle d'attente.

L'entrée au pupitre se fait par interruption et la sortie par boucle d'attente.



## II. DEUXIEME PARTIE

### UTILISATION DU SYSTEME

## II.1. LANCEMENT ET GENERATION DES OPERATIONS DE DEBUT ET DE FIN DE TRAVAIL

Nous avons expliqué, au début de la première partie façon succincte, que le système temps-partagé pour être mis en état de marche devait tenir compte d'abord du système temps-partagé proprement dit, ensuite d'un sous-système "LOADER UTILITY" qui intervient dans toutes les opérations de lancement, génération, début et fin de travail.

Dans cette deuxième partie, nous allons voir les détails de toutes ces opérations.

### Quelques mots sur le sous-système "LOADER UTILITY"

Le "LOADER UTILITY" une fois chargé, (cf. I première partie), est lancé à l'adresse 2000 et s'étend jusqu'à l'adresse 13322, (cf. I.1.1).

Le "LOADER UTILITY" est capable d'accomplir les tâches suivantes : (l'explication exhaustive des différents rôles de "LOADER UTILITY" est fournie en Annexe 3).

- Formattage, pour vérifier notamment s'il existe éventuellement de mauvaises pistes.

- Chargement du système temps-partagé qui aurait été préalablement mis sur bande magnétique.

- Chargement sélectif, (pour faire le chargement d'un fichier sélectif) de bande magnétique sur disque.

- Dump sélectif, pour envoyer l'image d'un fichier sélectionné du disque sur bande magnétique, pour impression ultérieure.

- Copie, permet la copie du disque d'un canal sur le disque d'un autre canal.

Et enfin, une dernière possibilité, permet l'élimination des pistes qui auraient été libérées par la commande "KILL" (cf. Annexe 2 la liste des commandes).

Ceci ne constitue que l'une des différentes tâches établies par le "LOADER UTILITY".

Pour la forme exacte des commandes ainsi que pour des explications détaillées, voir Annexe 3.

#### II.1.1. LANCEMENT DU SYSTEME

Le "LOADER UTILITY" une fois chargé est lancé à l'adresse 2000 ; la réponse au pupitre est alors

2000 E UTILITY PROGRAM

\*

L'astérisque veut dire que le sous-système "LOADER UTILITY" est prêt pour recevoir des instructions.

A ce moment là, nous initialisons le disque utilisé par la commande propre au "LOADER UTILITY" qui a la forme suivante

FORMAT, 1

(cf. Annexe 3)

si le disque est bon c'est-à-dire si toutes les pistes sauf au maximum sont en bon état nous pouvons continuer sinon il faudra changer de disque.

Pour obtenir la taille du module (ce qui est nécessaire pour les opérations de chargement sur disque ou en mémoire centrale) il suffit d'utiliser alors une instruction du type

ABS LIBRA - # NOM

Ainsi la taille de chaque module est déterminée au moment du chargement du système, ce qui évite de procéder à son calcul pendant le fonctionnement du système.

L'exemple suivant résume les opérations précédentes

Exemple :

37300		ORG	LIBRA
37672	#	FLB	EQU *

ABS LIBRA-# FLB détermine la valeur  $37672 - 37300 + 1 = 373$  donne la longueur d'un des modules en recouvrement (le module "FLB").

Le "LOADER UTILITY" nous permet d'ailleurs l'option suivante.

En positionnant à 1 le bit 15 du registre d'entrée S, (cf. Annexe 1), nous faisons arrêter le système au moment où un module en recouvrement est chargé dans la mémoire centrale, ceci avant son chargement sur le disque.

Cette option permet de vérifier et d'apporter tout changement qu'on estimerait nécessaire à ce module au moment de son passage dans la mémoire centrale.

Le système temps-partagé une fois chargé en entier, le "LOADER UTILITY" prend le contrôle et demande

DATE ?



La réponse étant fournie, il demande à nouveau

TIME :

voir, à ce niveau, le détail des commandes ainsi que les messages qu'on leur donne dans l'annexe 4, concernant les 3 modules de chargement.

Nous fournissons si on le désire l'heure.

A cet instant-là, le "LOADER UTILITY" passe le contrôle au module "SYSTEM START UP" qui est en recouvrement et qui se trouve dans la mémoire centrale.

Ce module est ramené dans la mémoire centrale, une fois que le système est entièrement chargé.

Il procède à l'initialisation de l'ensemble des consoles multiplexeurs, et sort le message

READY

A ce niveau-là, nous devons procéder à la configuration du système matériel et ceci par deux commandes que nous frappons au pupitre.

La première commande est

DISC - UP, 1

qui nous permet d'ajouter au système le disque utilisateur.

La deuxième commande de la configuration du système matériel est

$$\text{SPE} - 47, 2, \left\{ \begin{array}{l} \text{All} \\ n_1, n_2, \dots, n_i \end{array} \right.$$

qui détermine la vitesse de transmission du système et des consoles repérées par leurs numéros  $n_1, \dots, n_i$ .

Ici nous avons indiqué deux formes particulières de ces deux commandes ; la forme générale ainsi que l'explication détaillée de chacune d'elle font l'objet de l'Annexe 2.

### II.1.2. GENERATION DU SYSTEME

La génération du système veut dire : affecter aux utilisateurs un code ID, un mot de passe, un temps d'utilisation ainsi qu'un certain nombre de secteurs disque, pour cela, nous utilisons la commande suivante de la façon indiquée :

NEWID - code ID, mot de passe, le temps autorisé,  
le nombre de secteurs disque disponibles

De cette façon chaque utilisateur sera connu du système par un certain nombre de paramètres que nous indiquons à l'Annexe 2.

Si l'on ne procède pas à la génération du système et si l'on désire travailler avec une console quelconque, le système refuse et répond

ILLEGAL ACCESS

### II.1.3. EXECUTION DE DEBUT DE TRAVAIL

Le système ayant été initialisé et lancé comme il a été décrit précédemment (fourniture des codes ID, mot de passe, temps et place disque disponible), avant de commencer à travailler, l'utilisateur doit procéder à partir de sa console à un certain nombre de travaux préliminaires qui se matérialisent essentiellement par l'entrée de la commande "HELLO" sous la forme suivante :

HELLO - ID code, mot de passe

(cf. Annexe 2)

A cette commande de l'utilisateur, si le code, ainsi que le mot de passe, sont identifiés par le système, la réponse du système sera indiquée à la console par

READY

dès cet instant, l'utilisateur pourra frapper ligne par ligne son programme qui en transitant par le tampon correspondant sera stocké sous forme de "COMPILATION PARTIELLE" sur disque.

Toutes les instructions du programme ayant été fournies, au moment de l'exécution, les codes générés binaires reviennent et occupent la mémoire centrale de l'adresse 2000 jusqu'à l'adresse 12253 début des zones de tampons (cf. I.1.1).

Le "LOADER UTILITY" qui occupait pratiquement toute cette zone de mémoire sera entièrement détruit à l'arrivée et pendant l'exécution du premier programme.

#### II.1.4. EXECUTION DE FIN DE TRAVAIL

Le travail de l'utilisateur étant terminé il frappe à la console la commande

BYE

Cette commande fait sortir l'utilisateur du système et permet à ce dernier d'annoncer le reste du temps disponible à la console d'utilisateur (cf. Annexe 2).

L'ensemble des utilisateurs ayant fini de travailler, au pupitre nous commandons la déconnexion du disque utilisateur par la commande de configuration du matériel

DISC - DN, 1

(cf. Annexe 2)

Le disque utilisateur une fois mis hors système, nous employons la commande

SLEEP

qui produira les effets suivants.

Une chaîne de caractères sous forme de message est transmise à toutes les consoles encore actives et tous les utilisateurs actifs sont immédiatement déconnectés du système.

### II.1.5. SAUVEGARDE DU SYSTEME TEMPS-PARTAGE

#### - Sauvegarde du système sur la bande magnétique

Au moment où nous frappons au pupitre la commande "SLEEP" (cf. II.1.4) la réponse du système est :

MAG TAPE SLEEP ?

Par ce message, le système demande si la sauvegarde doit avoir lieu sur la bande magnétique.

Si c'est le cas nous frappons au pupitre :

YES

A cet instant-là, le module "SLEEP" appelle le "LOADER UTILITY" qui se trouve sur le disque et lui passe le contrôle.

Le "LOADER UTILITY", ayant besoin du numéro de canal de la bande magnétique pour faire la sauvegarde demande :

MAG TAPE ?

Dans le cas de la configuration de notre système, c'est le numéro 20 qui correspond au canal "DONNEES" du dérouleur de bande magnétique (le dérouleur est relié à l'ordinateur par 2 canaux, un canal "DONNEES" et un canal "COMMANDES").

Si la bande magnétique est prête, l'ensemble du contenu du disque système est recopié sur la bande magnétique.

Si l'unité de bande magnétique n'était pas en état de marche, un diagnostic d'erreur serait fourni par le système (HLT 11 B).

Dans ce cas, il faudrait préparer l'unité de bande magnétique et appuyer sur le bouton "RUN".

- Sauvegarde du système sur le disque

Si l'on ne dispose pas d'unité de bande magnétique, à la question de "LOADER UTILITY", (après avoir frappé la commande "SLEEP") qui était

MAG TAPE SLEEP ?

nous répondons NON au télétype, ce qui veut dire implicitement, que nous désirons la sauvegarde du système sur le disque. A ce moment-là le système dit :

INSERT CARTRIDGE FOR SYSTEM DISC DUMP  
PRESS "RUN" WHEN DISC READY

Après avoir sorti ce message, le système se met sur la position "HLT" avec 1020H dans le registre "MEMORY DATA".

Ce qui veut dire que nous devons changer la cartouche de disque pour mettre celle sur laquelle nous désirons mettre le système temps-partagé.

Quand le changement a eu lieu et que l'équipement est prêt nous appuyons sur le bouton "RUN".

A cet instant, le contenu du disque système, canal 0 est recopié sur le disque canal 1, le système envoyant ensuite le message suivant :

SYSTEM SLEEP COMPLETE

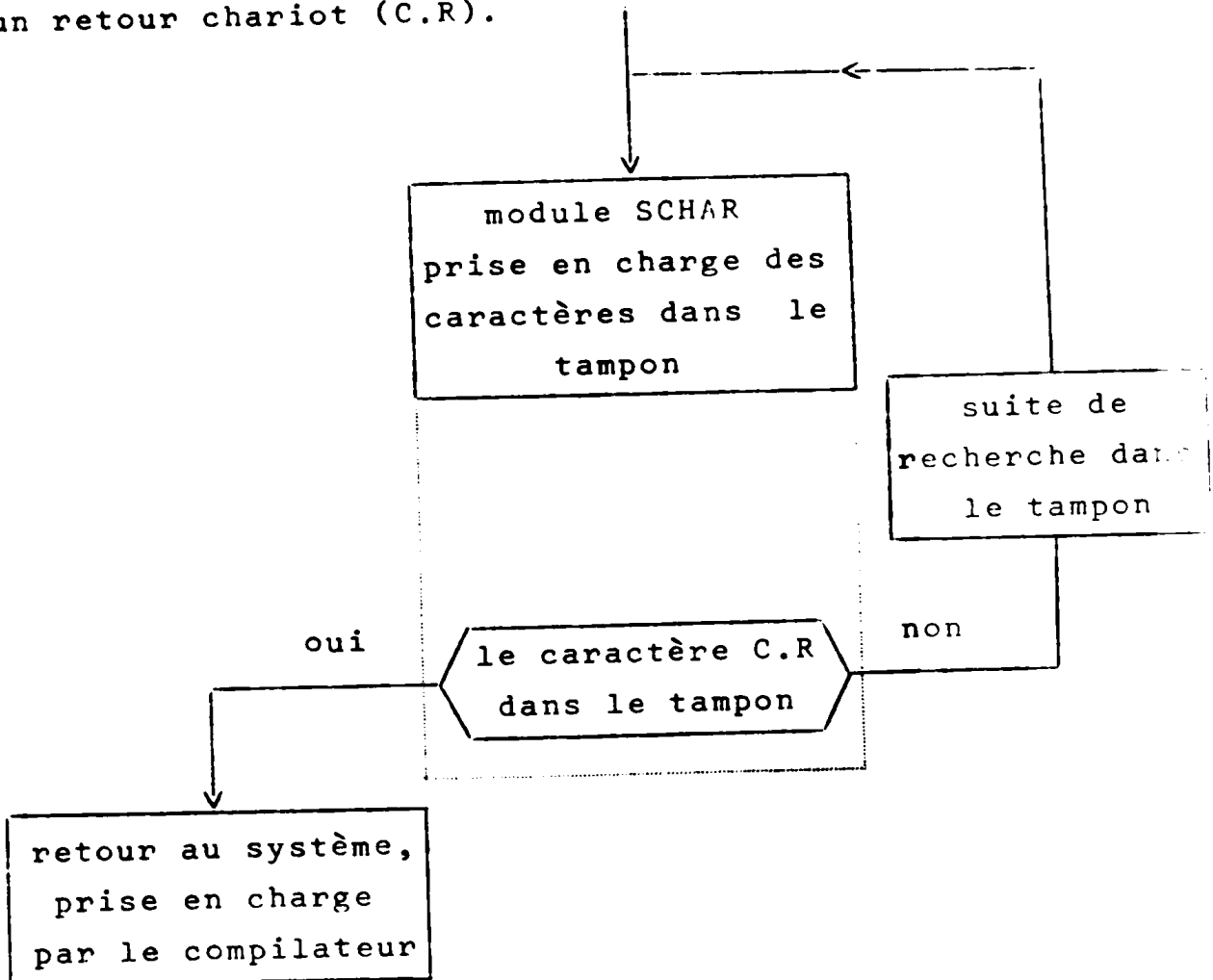
se met en "HLT" avec 102077 dans le registre "MEMORY DATA".

*Remarque* : si avant l'entrée de la commande "SLEEP" nous ne déconnectons pas le disque utilisateur par commande de la configuration matérielle "DISC-DN,1", le système répond "REMOVE DISC SUBCHANNEL 1".

Dans ce cas, la commande "SLEEP" est entièrement ignorée et doit être entrée à nouveau, après la déconnexion du disque utilisateur.

## II.2. JUMELAGE ET ZONE DE COMMUNICATION AVEC LE COMPILATEUR

Le rôle du compilateur, et par suite les problèmes liés au jumelage de celui-ci avec le système interviennent dès l'instant qu'une ligne frappée à partir soit d'une des 16 consoles multiplexées, soit du pupitre a été terminée dans chacun des cas par un retour chariot (C.R).



Ainsi le contenu des tampons étant constamment scruté par le module "SCHAR", tant qu'un C.R n'est pas rencontré la recherche continue. Dès l'instant que le module "SCHAR" dans son analyse rencontre un C.R, immédiatement, le contenu du tampon est pris en charge par le compilateur et le reste des analyses est du ressort de ce dernier dont nous verrons ci-dessous le mécanisme.



### II.2.1. MARCHE A SUIVRE DANS LE CAS OU UNE INFORMATIQUE PROVIENT D'UNE DES CONSOLES MULTIPLEXEES

Nous avons vu en I.2, (gestion dynamique du système), au moment où l'on décrivait les différentes étapes du répartiteur que le module "SCH1" jouait un rôle important, que c'était toujours "SCH1" qui étudiait la liste des tâches à exécuter.

En d'autres termes, celui-ci demandait, (pour tester et voir), si une entrée ou une communication avait eu lieu à partir du pupitre ou d'une des consoles multiplexées, (cf. Figure N° 19).

Comme nous l'observons sur les organigrammes, (cf. Figure N° 20 et 21), après le premier test pour nous informer sur la provenance de la communication, il y a une prise en charge du contenu du tampon et c'est ensuite le module "SCH5" qui étudie l'ensemble des possibilités qui existent.

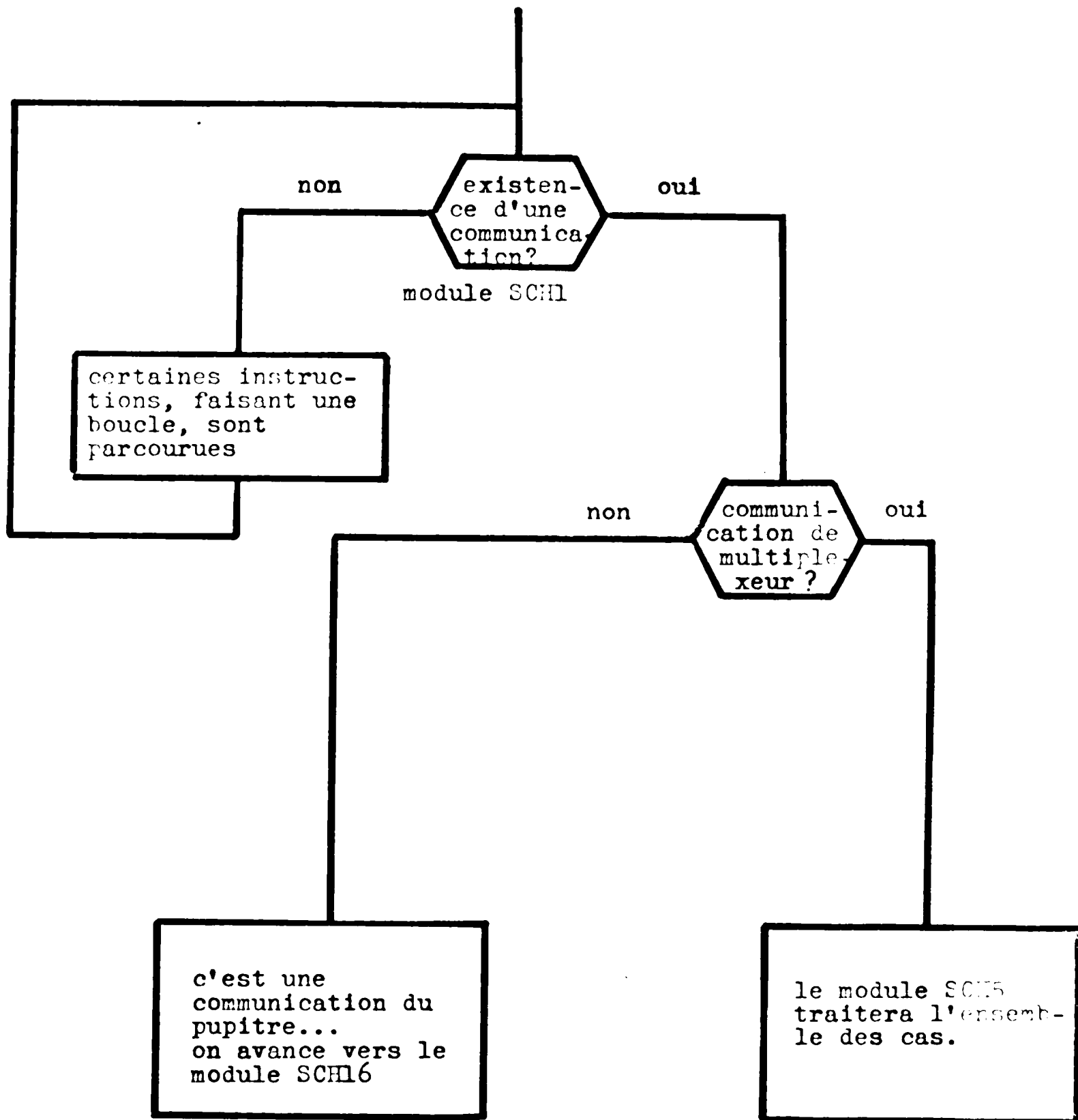
Un premier test est effectué pour déterminer si l'instruction frappée ne correspond pas à ce que nous appellerons une "déconnexion spéciale".

Par ce terme, nous désignons une information qui exige une mise en file d'attente prioritaire. Dans ce cas, le module "SCH8" qui assure le traitement de ces déconnexions spéciales, place la commande en tête de la file d'attente.

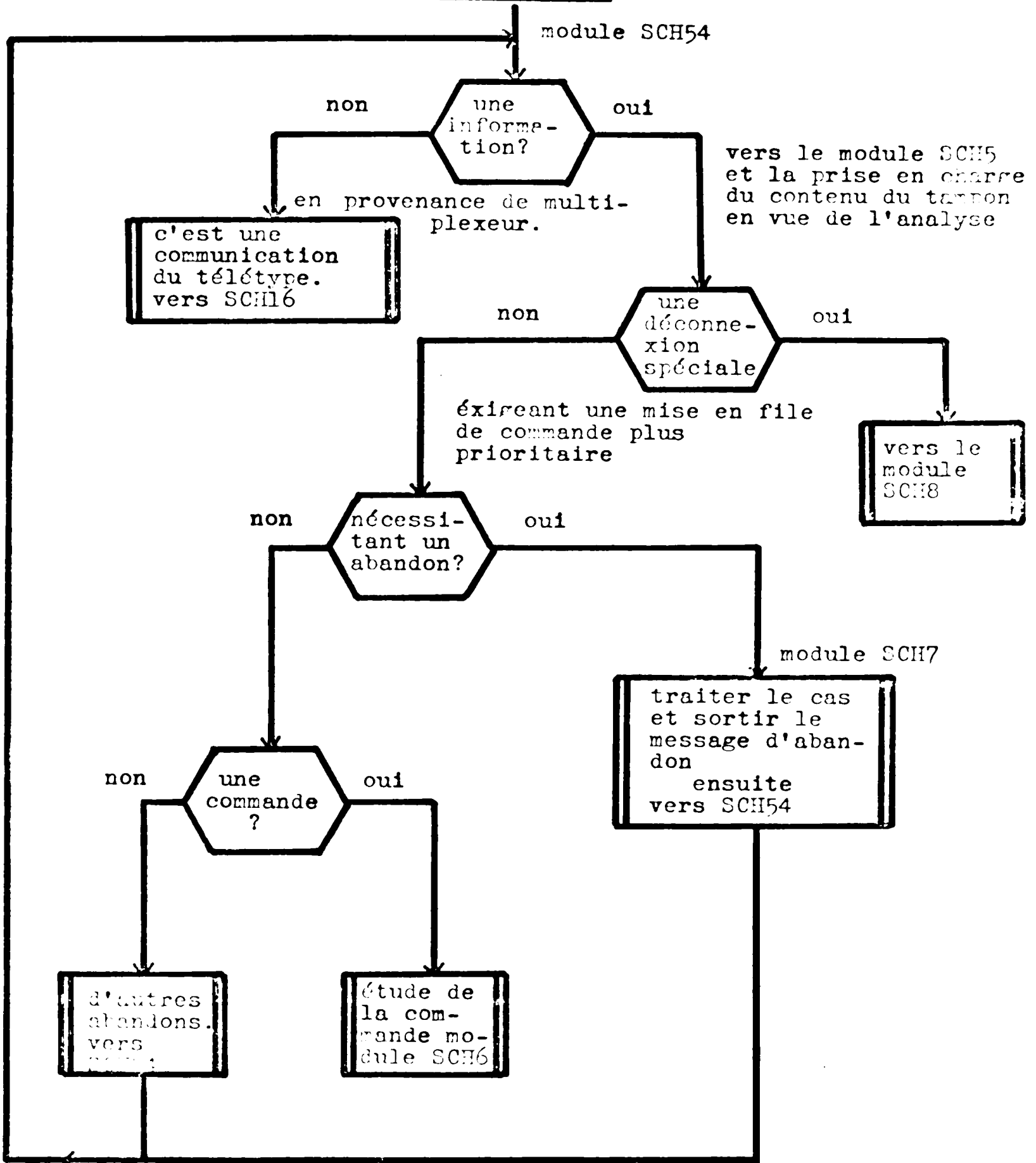
Si nous ne sommes pas en présence d'une "déconnexion spéciale", un test a lieu pour l'étude d'un éventuel abandon, par exemple : s'il s'agissait de l'entrée d'une instruction avant que la commande "HELLO" (cf. Annexe 2) soit frappée.

Dans le cas où cette nécessité existe, le module "SCH1" s'en occupe, envoie un message d'abandon et le retour a lieu vers le module "SCH54" pour une nouvelle étude de la provenance de la communication.

figure no 20



Marche à suivre pour l'organigramme de communication  
à partir d'une console multiplexée.



S'il n'existe pas de nécessité d'abandon, c'est qu'une étude détaillée pour une éventuelle commande a lieu, à partir du module "SCH6", (voir la suite du même chapitre et plus particulièrement II.2.3).

Si nous ne sommes pas en présence d'une commande connue, où si la forme de la commande n'est pas complète, une nouvelle étude d'abandon aura lieu et le retour final se fera au module "SCH54".

#### II.2.2. MARCHE A SUIVRE DANS LE CAS OU UNE INFORMATION PROVIENT DU PUPITRE

C'est dans le module "SCH16" que ce cas est traité ; tout d'abord les pointeurs de début et de fin du tampon du pupitre sont positionnés.

Ensuite on fait appel au module "SCOM" qui traite le cas des commandes.

Il est à remarquer que tout au contraire de ce que nous avons décrit au § II.2.1, qui était la marche à suivre dans le cas d'une console utilisateur, ici, et puisque nous sommes au pupitre, ce qui nous intéresse est uniquement le cas des commandes.

La première analyse du module "SCOM" consiste à voir si nous sommes en présence d'un caractère "LINE FEED" (passage à la ligne suivante) ; si c'est le cas, le module "SCH17" se charge de faire sortir au télétype système un caractère "LINE FEED"; le contrôle est donné ensuite au module "SCH15" qui fera appel au module "SWAPR", responsable de l'ensemble de "SWAPPING", (cf. I.2.1) et (cf. I.2.2).

Nous attirons l'attention sur le fait suivant : si la console travaille en "mode local" (non branchée au système),

le fait de frapper le caractère "LINE FEED", fera apparaître immédiatement à l'écran le caractère en question.

Par contre, si on travaille en "MODE LIGNE" (branchée au système), le fait de frapper le caractère "LINE FEED" ne fera plus apparaître automatiquement à l'écran un "LINE FEED" et c'est le système logiciel et la manière dont il est conçu qui déterminent la réponse. D'où la nécessité de l'étude ci-dessus.

Si le premier test, à l'intérieur du module "SCOM" décide que nous ne sommes pas en présence d'un "LINE FEED", un autre test est fait pour déterminer si le premier caractère de l'instruction est un chiffre (cf. II.2.3 le point fondamental du jumelage). Dans ce cas les modules "SCH18" et "SCH21" prennent successivement le contrôle et font sortir au pupitre les 3 caractères "???".

Ceci est dû au fait, qu'au pupitre, nous ne pouvons entrer que des commandes. Une ligne d'instruction, quel que soit le langage conversationnel choisi, (cf. II.2.3), est prohibée et la réponse du système serait "???" qui est une sorte de message d'erreur.

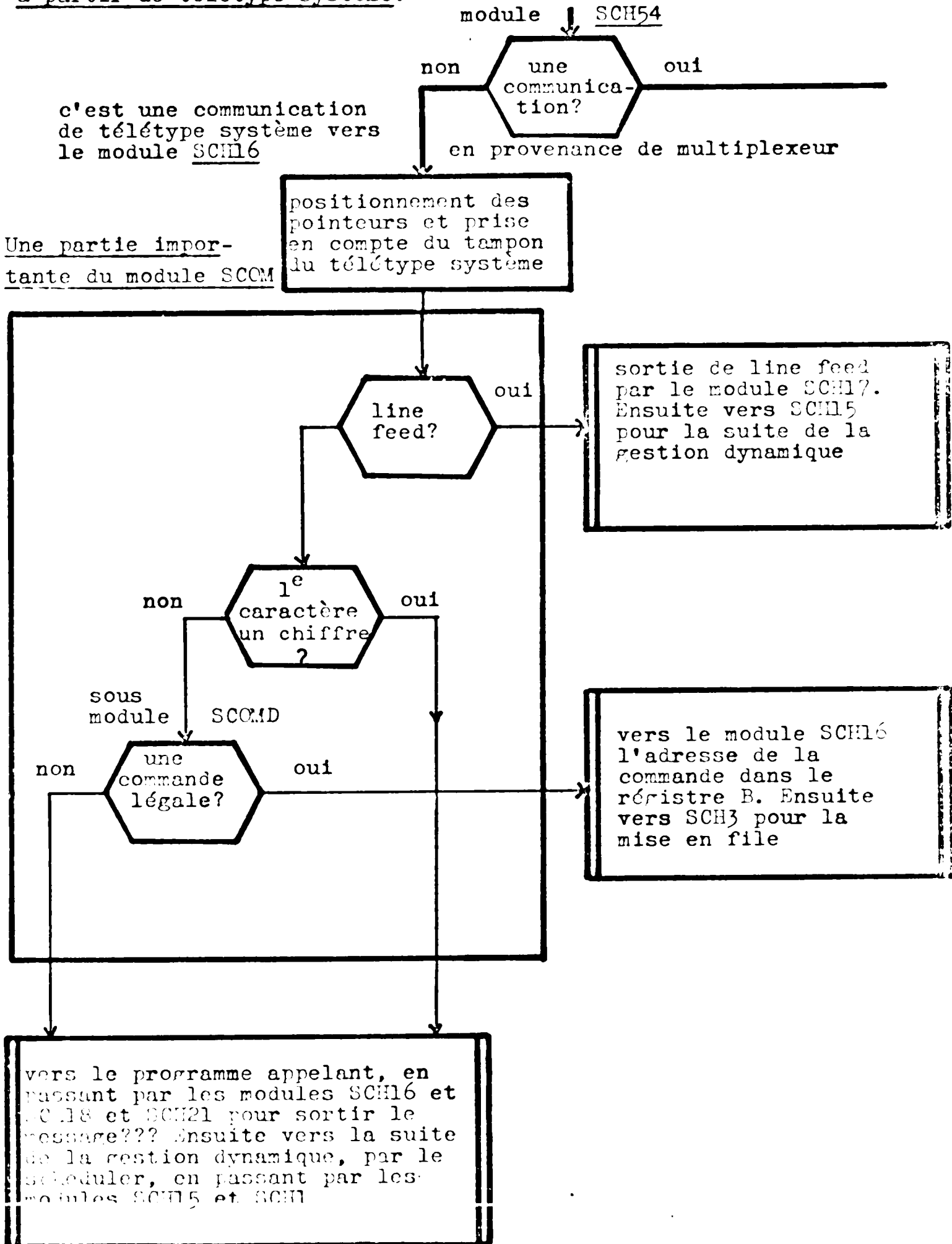
Si le premier caractère de la ligne n'est pas un chiffre, toujours à l'intérieur du module "SCOM" les sous-modules "SCOME" et "SCOMC" prennent le contrôle et font les tests nécessaires pour déterminer si nous sommes en présence d'une commande, et si c'est le cas (3 lettres consécutives), s'il s'agit d'une commande légale ?

Par commande légale, nous comprenons des commandes système accessibles uniquement du pupitre.

Si le système reconnaît une commande légale, l'adresse de la commande est sauvegardée dans le registre B et le retour a lieu vers le programme appelant, le module "SCH16", qui à son tour passe le contrôle au module "SCH3" pour la mise en file d'attente de la commande.

Marche à suivre et organigramme de la communication

à partir de télétype système.



S'il ne s'agit pas d'une commande légale, un message d'erreur est sorti au pupitre, par les modules "SCH18", et "SCH21".

En dernier lieu le contrôle est passé au module "SCH1" pour une éventuelle mise en garde (cf. I.2.2). S'il n'y a pas de nécessité, c'est le module "SCH1" qui prend le contrôle pour voir, au niveau du module "SCHEDULER" s'il existe quelque chose à faire (cf. I.2.1).

L'organigramme qui suit (cf. Figure N° 22) résume la discussion ci-dessus.

### II.2.3. POINT FONDAMENTAL DU JUMELAGE DU SYSTEME TEMPS-PARTAGE AVEC UN COMPILATEUR CONVERSATIONNEL

Nous venons de voir, en II.2.1 et en II.2.2 le déroulement dynamique du système, une fois qu'une ligne d'instruction ou de commande a été frappée soit sur une console multiplexée, soit au pupitre.

Nous allons faire maintenant la synthèse des 2 cas et ceci pour tenter de montrer le plus clairement possible le niveau du jumelage et faire en sorte que le déroulement du fonctionnement soit le plus compréhensible possible.

Nous espérons ainsi que l'utilisation et le jumelage de ce système temps-partagé à tout langage conversationnel apparaîtra dans ses aspects généraux et structurels.

Ici, nous faisons la synthèse des 2 organigrammes précédents (cf. Figure N° 21), (cf. Figure N° 22), dont nous ne répèterons pas les détails.

Un premier test a lieu au module "SCH54" pour voir la provenance de la communication.

Qu'il s'agisse de la console multiplexée ou du pupitre, l'analyse de toutes les possibilités a lieu systématiquement ; pour la console, le travail est mené par le module "SCH5" et pour le pupitre par le module "SCH16".

Dans le cas des 2 modules, nous faisons appel à un autre module "SCOM", qui sert à vérifier s'il s'agit d'un "LINE FEED" ou si le 1<sup>er</sup> caractère d'une instruction est un chiffre ou s'il s'agit effectivement d'une commande.

Le retour à partir du module "SCOM" aux 2 autres a lieu par l'instruction "JMP SCOM, I" ("JMP" : branchement).

Bien qu'il s'agisse de 2 modules indépendants, "SCH16" et "SCH5", qui font appel à ce même module "SCOM", le retour peut s'effectuer correctement ainsi : chacun des 2 modules "SCH16" et "SCH5", pour se brancher aux différents endroits, emploie un certain nombre de branchements ("JUMP") qui se suivent de la façon suivante :

JSB	SCOM
JMP	....
JMP	....
JMP	....

A l'intérieur du module "SCOM" un label "SCOM" s'incrémente à chaque fois qu'un test a lieu, et comme les possibilités communes du pupitre et de la console utilisateur sont étudiées de façon que l'ordre des différents "JUMP" qui se suivent leur corresponde respectivement, le retour a correctement lieu et ainsi l'appel aux différents endroits se fera convenablement dès le retour à partir du module "SCOM".

Comme nous l'observons dans l'organigramme de synthèse des 2 cas, (cf. Figure N° 23), le module "SCOM", dans son deuxième test, et dès l'instant qu'il s'aperçoit que l'instruction



commence par un chiffre, passe l'analyse à un autre module qui est "SCH9".

Ceci ne concerne évidemment que le cas d'une console multiplexée, car comme nous l'avons déjà vu plus haut, et comme les organigrammes le montrent bien, (cf. Figure N° 22 et Figure N° 23), si l'instruction en provenance du pupitre commence par un chiffre, le système répondra par un message d'erreur.

Dans le module "SCOM" figure au début, une séquence d'instruction qui analyse le premier caractère d'une information (issue du pupitre, ou d'une console multiplexée). Cette séquence constitue le point fondamental du jumelage avec le compilateur, car elle détermine l'orientation vers le module qui assurera le traitement de l'information.

Or, il est évident que le nombre de branches de l'orientation dépend directement du type de compilateur que l'on se propose d'implanter. Dans le cas du "BASIC", comme on peut le constater sur la séquence reproduite ci-dessous, l'analyse du premier caractère a pour but de déterminer simplement si c'est un chiffre, auquel cas l'information de la ligne est transmise au module "SCH9" pour être analysée (syntaxiquement et compilée, ou si c'est une lettre, ce qui indique qu'il s'agit d'une commande.

ISZ	SCOM	
ADA	M60B	
SSA		le caractère est
ISZ	SCOM	un code ASCII
ADA	.-10	correspondant à
SSA		un chiffre ?
JMP	SCOM,I	

ISZ	SCOM	
ADA	M+1B	
SSA,	RSS	le caractère est
JMP	SCOM,I	un code ASCII
ADA	.+32B	correspondant à
SSA		une lettre ?
JMP	SCOM,I	

Si le compilateur à implanter n'exige pas que toutes les instructions soient numérotées, la séquence précédente doit être transformée complètement pour assurer une orientation qui peut être beaucoup plus compliquée.

Dans le cas du "BASIC", l'orientation est évidemment simple. Si l'on se trouve en présence d'une lettre, il suffit de s'assurer qu'avec les deux caractères qui suivent, cela forme une commande valide.

S'il s'agit d'un chiffre, le compteur correspondant "SCOM" dans le module "SCOM" est incrementé en sorte que le retour aux programmes appelant, (module "SCH5" pour la console multiplexée et module "SCH16" pour le pupitre), "JMP SCOM,I" corresponde à "JMP SCH9" pour le début de l'analyse syntaxique.

Donc le véritable point crucial du jumelage à tout autre langage conversationnel est ici-même.

Car tous les autres modules qui seront des branchements et des retours à partir du module "SCOM", comme par exemple dans le cas d'une entrée vide, d'une commande correcte, d'une commande fausse...etc, resteront entièrement valides.

Dans le cas du pupitre, ces divers renvois ("JMP SCOM,I") au programme appelant, le module "SCH16", correspondent à l'ensemble des branchements :

	JMP	SCH30
	JMP	SCH9
	JMP	EHERR
	JMP	SCH11

qui veulent dire respectivement :

- la ligne est vide ("LINE FEED")
- le premier caractère d'une ligne est un chiffre
- la commande n'est pas valide
- la commande est correcte. (cf. II.2.2)

Bien entendu, toute cette structure reste entièrement valable pour toute compilation conversationnelle.

Dans le cas de la console multiplexée les divers renvois ("JMP SCOM,I") au programme appelant, le module "SCH5", correspondent à l'ensemble des branchements :

	JMP	SCH8
	JMP	SCH7
	JMP	SCH6
	JMP	SCH54

qui veulent dire respectivement :

Déconnexion spéciale (commande plus prioritaire qu'une instruction)

Abandon

Etude de la commande

Autres abandons. (cf. II.2.1)

Comme dans le cas du pupitre, toute cette structure reste entièrement valable pour toute compilation conversationnelle.

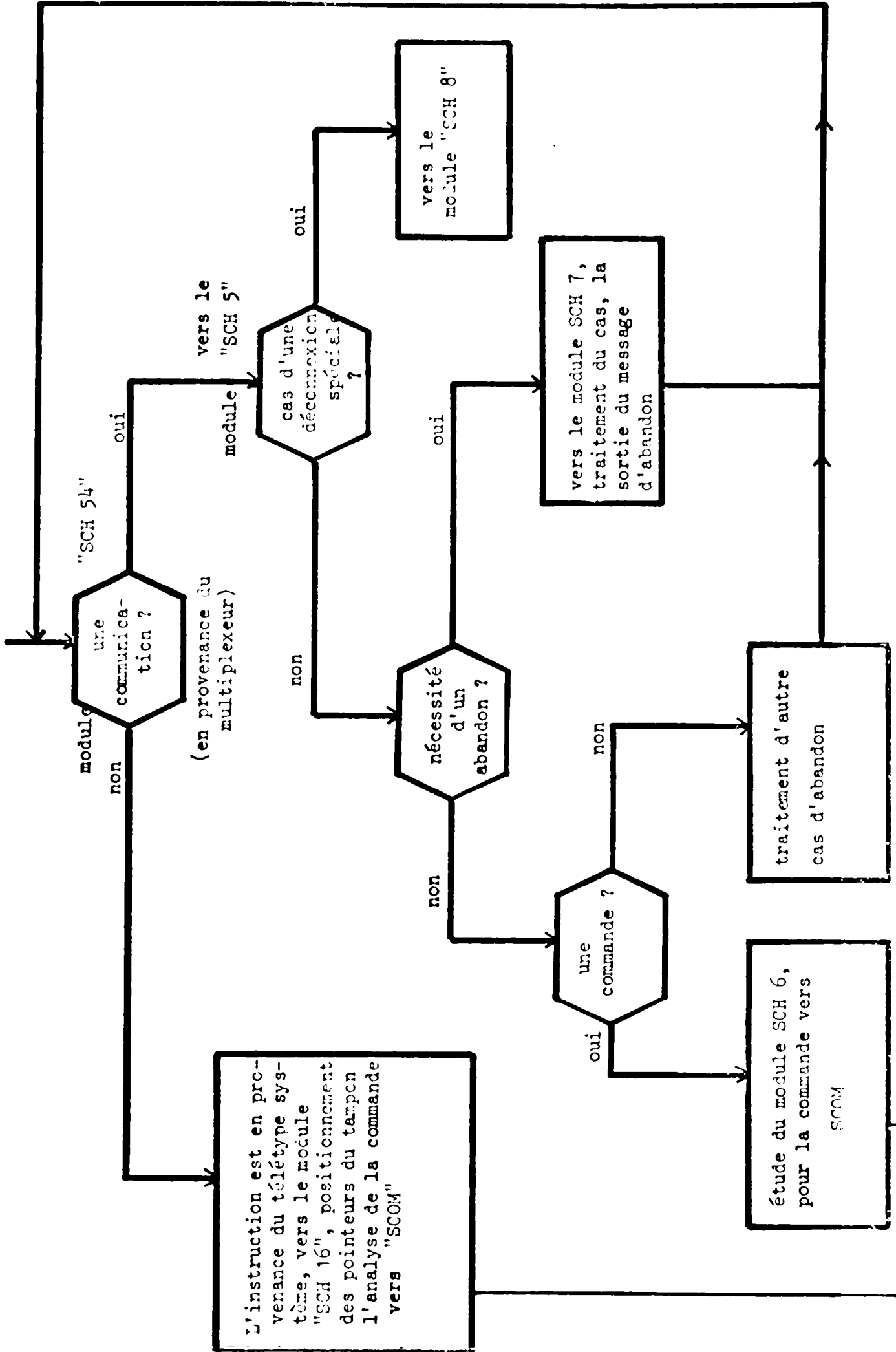
Les paragraphes qui précèdent nous montrent, qu'en fonction du langage conversationnel choisi et des contraintes imposées par le début d'une ligne d'instruction, il faudra réécouter les quelques instructions du module "SCOM", qui ont fait l'objet d'une analyse approfondie, en sorte que l'envoi, (quelles qu'aient été les contraintes syntaxiques du début d'instruction), ait lieu à nouveau au module "SCH9".

C'est à partir de ce dernier module que commence le début de l'analyse syntaxique et le branchement vers les différentes étapes de la compilation au moyen de l'instruction "LDA SYNTA" ("SYNTA" est l'adresse du début de l'analyse syntaxique).

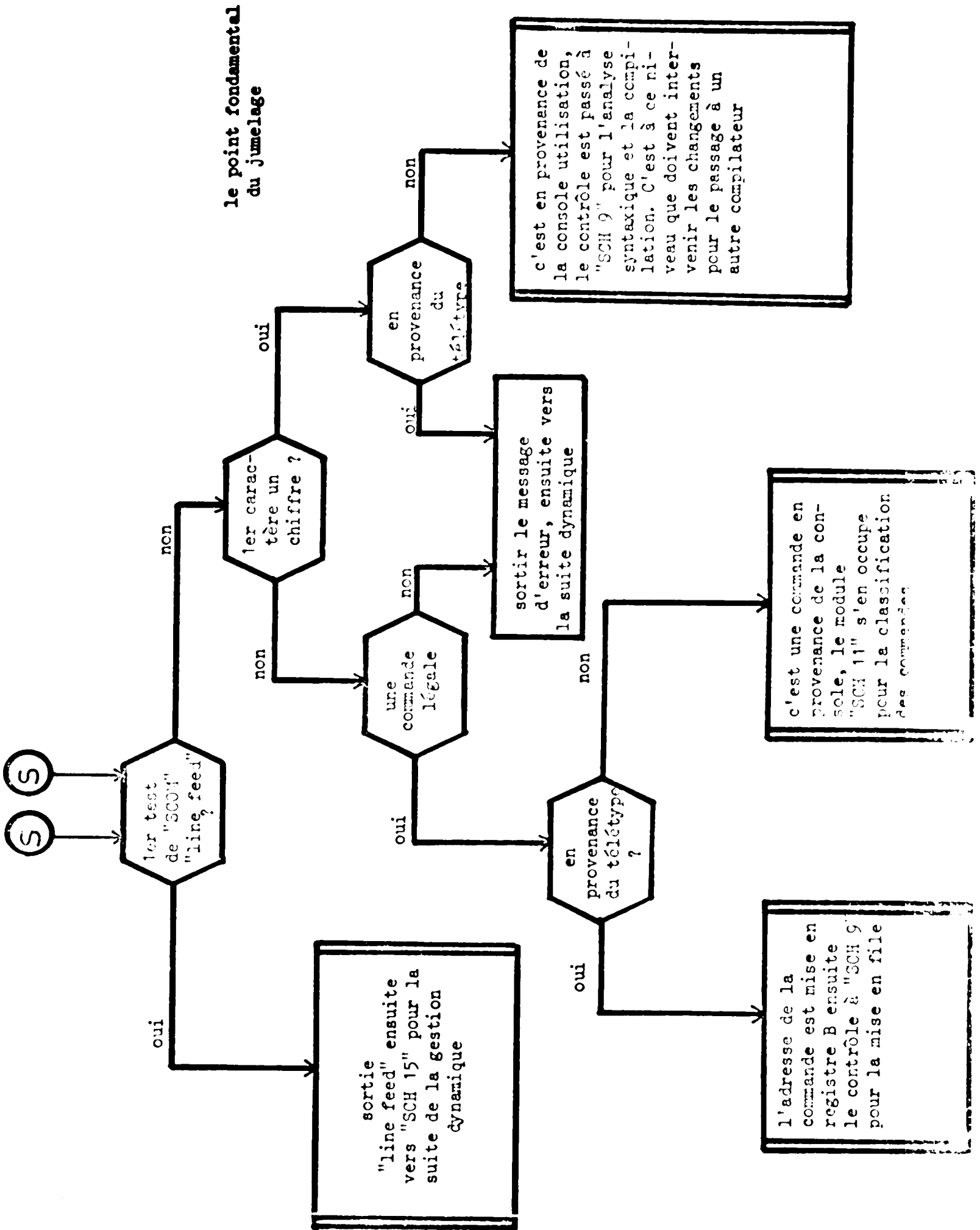
L'organigramme, (cf. Figure N° 23) fournit le traitement global de la communication en provenance du pupitre ou de la console multiplexée.

LE POINT FONDAMENTAL DU JUVELAGE, LA SYNTHÈSE DU TRAITEMENT DE LA COMMUNICATION EN PROVENANCE DE LA CONSOLE

MULTIPLIÉ PAR UN SYSTÈME SCHEM



le point fondamental du jumelage



sortie "line feed" ensuite vers "SCH 15" pour la suite de la gestion dynamique

sortir le message d'erreur, ensuite vers la suite dynamique

c'est en provenance de la console utilisation, le contrôle est passé à "SCH 9" pour l'analyse syntaxique et la compilation. C'est à ce niveau que doivent intervenir les changements pour le passage à un autre compilateur

c'est une commande en provenance de la console, le module "SCH 11" s'en occupe pour la classification des commandes

l'adresse de la commande est mise en registre B ensuite le contrôle à "SCH 9" pour la mise en file

### II.3. UN EXEMPLE D'UTILISATION

L'ensemble des unités périphériques et l'unité centrale étant mis en état de marche nous procédons au lancement du système temps-partagé par une des méthodes décrites dans l'Annexe 4.

Pour des raisons de simplicité et de rapidité nous préférons le lancement du système temps-partagé par la bande magnétique.

Pour la génération, configuration, lancement et l'arrêt du système nous employons un certain nombre de commandes ; les détails de toutes ces commandes sont fournis dans l'Annexe 3, ici nous ne décrivons qu'un cas d'utilisation.

Le système temps-partagé étant chargé nous recevons au pupitre le message

READY

Nous commençons par l'activation du disque utilisateur en tapant la commande utilisateur

DIS - UP,1

Ensuite nous activons toutes les consoles en donnant à toutes une vitesse égale de 30 caractères par seconde par la commande

SPE - 47, 2, ALL

Nous autorisons tous les utilisateurs à travailler avec "IDCODE C555" et le mot de passe "BOOM", en accordant 1000 minutes de temps de travail et 240 secteurs d'espace sur le disque.

NEWID - C555, BOOM, 1000, 240

A ce moment-là, le travail de l'opérateur système étant terminé les utilisateurs peuvent commencer à travailler.

Un utilisateur doit d'abord annoncer son entrée dans le système en introduisant à sa console

HEL - C555, BOOM

la réponse du système est

READY

Ensuite l'utilisateur peut donner un nom à son programme par exemple "ZIPOF"

NAM - ZIPOF

Nous entrons notre programme soit ligne par ligne en mode "KEY" soit le tout en mode "TAPE".

Les corrections du programme étant terminées nous l'exécutons par la commande "RUN".

Notre travail ainsi que celui de toutes les autres consoles une fois terminé, nous sauvegardons nos programmes en entrant (dans notre cas particulier)

SAV - ZIPOF

Bien sûr, cette sauvegarde est facultative.

En dernier lieu nous donnons l'ordre de déconnecter les consoles du système en entrant la commande "BYE".

A cet instant, le système nous fournit l'information concernant le temps employé depuis le début, c'est-à-dire, depuis l'entrée de l'utilisateur, à partir de sa console.



Le rôle des utilisateurs étant terminé, c'est l'opérateur du système qui intervient.

D'abord il déconnecte le disque utilisateur en tapant la commande.

DIS - DN , 1

Ensuite et à partir du pupitre il entre la commande "SLEEP" et ceci pour lui permettre de sauvegarder le système.

La sauvegarde du système étant terminée, on débranche l'ensemble des consoles et on disjoncte l'alimentation électrique.

o o  
o

## C O N C L U S I O N

Notre système temps-partagé étant très perfectionné, avec les 32 commandes que nous avons élaborées et mises au point, (cf. Annexe 2), nous disposons d'une gamme étendue de possibilités très variées.

Ce système pourra être encore plus développé, en exploitant une quinzaine de commandes, faciles désormais à mettre au point.

La souplesse du système est telle, que la place en recouvrement ainsi que l'espace disque, sont bien prévus pour ces futures possibilités du système.

Actuellement la place disponible, pour le compilateur conversationnel, dans la mémoire centrale est de 6 K mots.

Comme ce système est initialement prévu pour une mémoire de 16 K mots, et que nous disposons actuellement de 32 K mots mémoire, une prochaine étape du développement du système consistera à faire les modifications nécessaires dans le "LOADER UTILITY" pour utiliser toute la place mémoire disponible.

Ceci doit se faire au niveau de l'adresse de "LIBRA", (77300 au lieu de 37300), pour gagner ces 16 K mots.

Ainsi disposant d'une possibilité de 22 K mots, nous pouvons imaginer, la conception et le jumelage de compilateurs conversationnels plus volumineux (cela fait partie des projets futurs).

ANNEXE 1  
CONFIGURATIONS

SYSTEME MATERIEL

Nous décrivons ci-dessous, des particularités importantes des divers éléments constituant le système de temps-partagé.

UNITE CENTRALE

Elle est constituée par un ordinateur "Hewlett-Packard" de la série 2100A d'une mémoire de 32 K mots de 16 bits, avec un registre d'entrée appelé S et 2 registres accumulateurs pour l'unité de calcul, (appelés registre A et B), addition et multiplication en virgule flottante câblée.

Elle est dotée de 2 canaux d'accès direct à la mémoire permettant une vitesse de transfert maximale de 1 million de mots par seconde.

Compte tenu de l'extension, HP 2155A, il est possible de connecter 52 périphériques en même temps.

MEMOIRE DE MASSE

La première mémoire de masse est constituée par une unité de disque (HP 7900A) à tête mobile, celle-ci comprend 2 disques,

l'un fixe, l'autre mobile. Chaque disque compte 200 pistes disponibles sur chaque face, une piste étant composée de 24 secteurs de 128 mots.

La capacité totale de stockage est de 2,5 millions de mots. La vitesse de transfert moyenne des données est de 312000 octets par seconde.

Une 2<sup>ème</sup> mémoire de masse est constituée par un dérouleur de bande magnétique (HP 7970 E) à neuf pistes compatibles IBM.

La densité d'enregistrement est de 1600 bpi et la vitesse de défilement de la bande est de 45 pouces/sec ( $\approx 1.15$  m/s), autorisant ainsi une vitesse de transfert maximale de 72000 caractères de 8 bits par seconde.

Cette vitesse n'est jamais atteinte car elle correspond à un transfert d'un bloc de très grande longueur.

#### AUTRES UNITES PERIPHERIQUES

Un lecteur de ruban perforé (HP 2748A) lisant jusqu'à 500 caractères par seconde.

Un perforateur de ruban (HP 8100) perforant jusqu'à 75 caractères par seconde.

Une imprimante série (LX 180) sortant environ 150 caractères par seconde.

Une unité de visualisation (HP 1300A) qui est dotée de 2 convertisseurs Digital-Analogique de 8 bits ce qui autorise une définition d'image de 1/256 en abscisse et en ordonnée.

Une console "SINTRA" servant de pupitre.

16 consoles "SINTRA" pour le multiplexage et le partage de temps.

Les terminaux "TTE" "SINTRA" employés pour le pupitre ainsi que pour les consoles multiplexées ont une vitesse de transmission (pouvant atteindre) de 50 à 9600 bits par seconde.

Elles sont dotées d'un écran de visualisation à rayons cathodiques autorisant 80 caractères par ligne, le nombre maximal de lignes pour l'écran étant de 24.

Les consoles ont la possibilité de travail en mode "local" ou en mode "ligne".

L'envoi d'information peut se faire soit en "HALF-DUPLEX", soit en "FULL-DUPLEX".

Le terminal "SINTRA" ne vérifie pas le bit de parité dans les caractères reçus.

### SYSTEME LOGICIEL

Le logiciel employé est tout d'abord, celui du système temps-partagé "BASIC", qui comporte quelques 25000 instructions (décimales).

Il y a en outre celui du sous-système "LOADER UTILITY" qui compte environ 5000 instructions (décimales).

Ainsi nous avons en tout quelques 30000 instructions (décimales) qui nous ont servi comme logiciel de base.

## ANNEXE 2

### LES COMMANDES EMPLOYEES PAR LE SYSTEME TEMPS-PARTAGE

Nous distinguons 2 sortes de commandes :

Commande opérateur, ou commande de pupitre.

Commande utilisateur, ou commande d'une console multiple-xée.

#### COMMANDE OPERATEUR OU COMMANDE DU PUPITRE

Nous les énumérons d'abord, l'explication détaillée pour chacune d'elle étant fournie ensuite.

- DISC Informe le système de l'addition d'un disque utilisateur, employé au moment du démarrage ou de l'arrêt du système.
- SPEED Informe le système d'une nouvelle configuration d'une ou de l'ensemble des consoles, (en ce qui concerne la vitesse de transmission et le bit stop).
- NEWID Fait entrer un nouveau "IDCODE" (code d'identification) dans le système.
- ANNOUNCE Transmet un message de l'opérateur à un utilisateur ou à l'ensemble des utilisateurs.
- RESET Mise à zéro du temps d'un ou de l'ensemble des "IDCODES".

- CHANGE Change, le mot de passe, le temps limite ou l'espace disque limite d'un "IDCODE".
- ROSTER Fournit une liste des consoles et des "IDCODES" actifs.
- PORT Fournit une liste de la configuration des consoles.
- DIRECTORY Fournit une liste des programmes de la bibliothèque et des fichiers.
- PURGE Détruit les programmes et les fichiers qui n'ont pas été utilisés depuis une date spécifiée.
- SLEEP Pour l'arrêt du système temps-partagé et sa sauvegarde.

Nous décrivons ci-dessous le détail de ces commandes, la manière dont il faut les employer ainsi que les messages système fournis à ce sujet.

$DISC - \left\{ \begin{array}{l} UP \\ DN \end{array} \right\},$  numéro de canal

C'est le format employé pour l'addition ou la mise hors service d'un disque.

- UP Indique au système qu'un disque est ajouté au canal spécifié.
- DN Indique au système que le disque du canal est déconnecté (du point de vue logiciel).
- NUMERO  
DU CANAL Numéro de disque, 1-3, auquel un nouveau disque peut être ajouté ou déconnecté.

Avant d'entrer la commande, il faut que l'unité de disque soit en marche, et que le disque utilisateur soit initialisé, (cf. Annexe 3).

Le numéro de canal 0 est affecté au disque système et ne devrait pas être employé, si on l'emploie, le message suivant est donné à l'opérateur.

CAN NOT SET SYSTEM DISC DOWN

D'autre part, si le numéro de canal dépasse la limite 3, le message suivant sera imprimé au pupitre

SUBCHANNEL ERROR

Si le disque indiqué n'est pas un disque utilisateur le message suivant apparaît au pupitre

NOT USER DISK

et le disque n'est pas ajouté au système.

Si le disque indiqué a un code d'identification qui diffère de celui du code d'identification du système, le message suivant apparaît au pupitre.

DISC ID CODE NOT SAME AS SYSTEM ID CODE  
WARNING ONLY

et le disque est ajouté au système.

Si des fichiers sur le disque indiqué sont en train d'être utilisés le message suivant sort au pupitre

BUSY FILES

et le disque n'est pas déconnecté du système.



SPEED { vitesse de transmission, bit stop,  
numéro de console (numéro de console;...) }  
All

Initialement toutes les consoles sont configurées à la vitesse de 1300 bauds, avec le bit stop 2.

La configuration des consoles pour ces caractéristiques pourrait être modifiée par la commande speed.

VITESSE DE TRANSMISSION      Vitesse de transfert des données de la console indiquée, suivant le code qui est venu.

BIT STOP                      Nombre total de bits stop inclus dans la composition en bits des caractères. Il peut être 1 ou 2.

NUMERO DE CANAL              Valeur décimale entre 0 et 15, représentant la console particulière à configurer. On peut configurer plus d'une console.

ALL                              Si cette opération est choisie, toutes les consoles seront configurées avec les paramètres indiqués.

Si le numéro de la console est supérieur au nombre de consoles autorisées, la commande est ignorée et le message suivant est envoyé au pupitre

#### ILLEGAL FORMAT

Si une console référencée par un numéro de console est branchée, une fois que la commande "SPEED" a été entrée, la commande est ignorée et il faudra entrer la commande "SPEED" de nouveau.

**Exemples.**

SPE - 47, 1, 0

Configure la console 0, avec la vitesse de transfert de 30 caractères par seconde.

SPE - 47, 1, ALL

Configure toutes les consoles à la vitesse de 30 caractères par seconde.

o o  
o

**NEWID-IDCODE** {mot de passe, temps, disque}

Cette commande est employée pour entrer un nouveau "IDCODE" dans le système. La commande "NEWID" établit un "IDCODE" unique avec un mot de passe pour chaque utilisateur, de plus fixe la limite maximum du temps et de l'espace disque alloué.

**IDCODE** Identification de l'utilisateur, consiste en une lettre suivie de 3 chiffres décimaux.

**MOT DE PASSE** Le mot de passe de l'utilisateur associé avec "IDCODE", consiste en 1 à 6 caractères alphanumériques. A l'exception de Null, X<sup>C</sup>, Rubout, Comma, space, return, linefeed, X-off.

**TEMPS** Le nombre maximum de minutes qu'un utilisateur est autorisé à employer ; le temps ne peut pas dépasser 65535 minutes, soit plus de 1000 heures.

**DISQUE** Le nombre maximum de secteurs (décimal) qu'un utilisateur est autorisé à employer pour ses programmes et fichiers. Ce nombre ne peut pas dépasser 65535 secteurs.

Le mot de passe peut être spécifié sans visualisation des caractères, en appuyant sur la clé "CONTROL" quand on entre un caractère. Cette combinaison est symbolisée dans le texte en ajoutant un "C" en haut du caractère comme X<sup>C</sup>. En employant cette option, l'utilisateur emploie un mot de passe secret qui n'apparaît pas à la console d'utilisateur.

Si le "IDCODE" spécifié existe déjà dans le système le message est envoyé au pupitre

#### DUPLICATE ENTRY

Si la table ID du système ne peut pas accepter un nouveau "IDCODE", l'opérateur reçoit le message suivant

#### ID TABLE FULL

Si le disque ne dispose pas du nombre de secteur suffisant pour faire face à la demande de l'utilisateur, le message suivant est envoyé à sa console.

#### FILE SPACE FULL

Si l'utilisateur dépasse son temps limite un drapeau est mis à 1 et l'utilisateur peut quand même continuer le travail en cours. Mais une fois le travail terminé, s'il essaie à nouveau, il reçoit le message suivant

#### NO TIME LEFT

ANNOUNCE - { numéro de console } , chaîne de caractères  
All

La commande "ANNOUNCE" permet à l'opérateur d'envoyer une ligne de message à la console des utilisateurs actifs.

NUMERO DE CONSOLE Valeur décimale de 0-15, désignant la console choisie.

ALL Si cette option est choisie la chaîne de caractères est envoyée à toutes les consoles actives.

CHAINE DE CARACTERES Message de 66 caractères envoyé à un utilisateur spécifique ou à tous.

La chaîne de caractères est imprimée de façon littérale, y compris les blancs, à la console indiquée. Et toute sortie qui était en train de se faire est interrompue.

La commande "ANNOUNCE" doit être utilisée avec précaution, car elle fait sortir des textes inattendus à la console d'utilisateur et peut gêner le travail d'un utilisateur.

o o  
o

RESET - { IDCODE } { , time }  
All

Cette commande permet à l'opérateur de changer la valeur du temps total d'un "IDCODE".

IDCODE Code identificateur d'un utilisateur spécifique.  
Seul le temps final de cet "IDCODE" est changé par la valeur du paramètre "TIME".

**ALL** Dans ce cas le temps final de tous les utilisateurs est changé.

**TIME** Valeur optionnelle de temps en minutes, qui ne peut pas excéder 65535. Si elle est omise, le système suppose une valeur nulle.

Si le "IDCODE" ne peut pas être trouvé le message suivant est envoyé à l'opérateur.

NO SUCH ID

Exemple :

RES- Q123, 20

Change la valeur limite du temps alloué à l'utilisateur de "IDCODE Q123" à 20 minutes.

o o  
o

CHANGE-IDCODE, {password}, {time}, {disc}

Cette commande est employée pour changer un paramètre quelconque d'un "IDCODE" existant.

**IDCODE** Code d'identification d'un utilisateur spécifique.

**PASSWORD** Si précisé, il devient le nouveau mot de passe associé à l'"IDCODE". L'ancien n'est plus valable à partir de ce moment.

**TIME** Si indiqué, il devient le nouveau temps, en minutes, affecté à "IDCODE". Il ne peut pas excéder 65535.

**DISC** Si précisé, il devient le nouveau espace disque, en secteurs, affecté à "IDCODE" (il ne peut pas dépasser 65535).

Au moins un paramètre, autre que "IDCODE", doit être précisé, et les paramètres doivent être entrés dans l'ordre indiqué. Si un paramètre manque, il faut le remplacer par une virgule.

Si l'"IDCODE" n'existe pas ou s'il n'est pas reconnu par le système, le message suivant sera visualisé :

NO SUCH ID

Exemple :

CHA - Q123, BASIC

Change le mot de passe de "IDCODE Q123" à "BASIC", le temps et l'espace disque affectés restent inchangés.

o o  
o

## ROSTER

Permet d'avoir une liste d'"IDCODES" actifs, la liste apparaît sur 2 lignes de 8 colonnes, l'élément de la première ligne et de la première colonne correspond à la console numéro 0 et le dernier élément correspond à la console numéro 15.

Une console active est précisée par l'"IDCODE" de son utilisateur. Une console inactive est précisée par 4 points successifs.

Le même "IDCODE" peut être actif pour plusieurs consoles.

## PORT

L'entrée de cette commande permet d'avoir la configuration de toutes les consoles.

La liste envoyée à l'opérateur montre le bit stop suivi de la vitesse de la communication pour laquelle la console est configurée.

La liste apparaît sur 2 lignes d'éléments, chaque élément de la ligne 1 correspond à une console (commençant par la console 0 et se terminant par la console 7). Les éléments de la 2<sup>e</sup> ligne correspondent à la console numérotée 8 et se terminant par celle numérotée 15.

Les éléments de la liste sont affichés sous la forme suivante :

	S - BBB
S	BIT STOP
BBB	VITESSE DE COMMUNICATION

o o  
o

DIRECTORY / - { IDCODE  
SUBCHANNEL } /

**IDCODE** Code d'identification d'un utilisateur spécifique. S'il est choisi, seuls les programmes et les fichiers de cet "IDCODE" sont fournis à l'opérateur.

**SUBCHANNEL** Entier compris entre 0 et 3, indiquant un disque spécifique, s'il est choisi ; les programmes et les fichiers de ce disque seront fournis à l'opérateur.

Si aucun des paramètres n'est précisé, une liste est imprimée commençant par le disque 0, le code utilisateur "IDCODE A000", en continuant par tout le répertoire de tous les disques.

Les informations suivantes sont fournies avec chaque programme et fichier.

ID	IDCODE d'utilisateur
NAME	Nom du programme ou du fichier (1 à 6 caractères)
DATE	Date de la dernière utilisation de l'élément (jour de l'année/année)
SUB	Numéro de canal du disque contenant le programme ou le fichier
TR/SEC	Adresse disque indiquant la piste et le secteur
LEN	Longueur du programme ou du fichier en secteurs (en décimal)
FP	Si F apparaît, l'élément est un fichier, sinon c'est un programme ; P indique que le programme ou le fichier est protégé contre un accès non autorisé.

o o  
o

### PURGE - DDD/YY

L'opérateur emploie la commande "PURGE" pour détruire dans la bibliothèque les fichiers ou les programmes de l'utilisateur qui n'ont pas été utilisés depuis une date précisée.

L'espace disque qui se trouve ainsi libéré est ajouté au système.



- DDD Valeur décimale de 1 à 366 représentant un jour de l'année. Elle doit être inférieure ou égale au jour actuel de l'année en cours.
- / Nécessaire pour séparer les éléments de ce paramètre.
- YY Valeur décimale représentant l'année dans le siècle (par exemple 72 pour l'année 1972). Elle doit être inférieure ou égale à l'année en cours.

Si le jour de l'année ou l'année ne sont pas référencés, le message suivant est envoyé à l'opérateur

#### ILLEGAL PARAMETER

Si le fichier indiqué pour être purgé est en cours d'utilisation, la commande "PURGE" n'est pas admise, et il y aura le message suivant

#### BUSY FILES

Le système n'informe pas l'opérateur du fichier ou programme qui est purgé, c'est la commande "DIRECTORY" qui doit être employée avant et après "PURGE" pour de telles vérifications.

Exemple :

PUR - 153/72

Purge tous les programmes et les fichiers qui n'avaient pas été utilisés depuis le premier juin 1972.

o o  
o

## **SLEEP (-character string)**

La commande "SLEEP" est employée pour l'arrêt du système temps-partagé. Cette commande assure une procédure d'arrêt systématique qui permettra un démarrage ultérieur plus aisé.

Avant d'entrer la commande "SLEEP", il faudrait déconnecter le disque utilisateur, et pour cela, il faut employer la commande "DISC".

**CHARACTER-STRING**      Message de 68 caractères au maximum qui, une fois spécifié est visualisé sur la console de tous les utilisateurs.

La commande "SLEEP" a les conséquences suivantes :

1. La chaîne de caractères est transmise à toutes les consoles actives.
2. Les consoles actives sont immédiatement déconnectées du système.
3. La bibliothèque du disque système se compacte pour éliminer des espaces morts ou non affectés.
4. Le système demande :

MAG TAPE SLEEP ?

Si le système doit être sauvegardé sur la bande magnétique, il faut entrer

YES

5. Si le système doit être sauvegardé sur le disque, entrer  
NO le système répond

INSERT CARTRIDGE FOR SYSTEM DISC DUMP  
PRESS "RUN" WHEN DISC READY

6. Le système se met en Halt avec 102011 dans le registre "MEMORY DATA". Il faut alors remplacer le disque mobile utilisateur du canal 1 par celui dont on désire qu'il contienne le système temps-partagé, mettre en marche l'équipement et presser sur "RUN".
7. Le contenu du disque système, canal 0 est copié sur le disque du canal 1. A la fin le système présente

SYSTEM SLEEP COMPLETE

Un Halt apparaît avec 102077 affiché dans le registre "MEMORY DATA".

Si le disque utilisateur n'était pas déconnecté par la commande "DISC" avant l'entrée de la commande "SLEEP", le système indique :

REMOVE DISC SUBCHANNEL N

N étant le numéro de canal (1-3), dans ce cas la commande "SLEEP" est ignorée.

o o  
o

COMMANDE UTILISATEUR, OU COMMANDE DE LA CONSOLE MULTIPLEXEE

Nous énumérons les commandes utilisateurs l'une après l'autre, avec des explications qui les suivent immédiatement.

HELLO-IDCODE, PASSWORD, TERMINAL TYPE CODE

Cette commande permet de faire entrer un utilisateur dans le système.

IDCODE et PASSWORD sont affectés par l'opérateur système.

Plusieurs utilisateurs peuvent être acceptés avec le même IDCODE sur des consoles différentes.

Le "TERMINAL TYPE CODE" informe le système de quel terminal il s'agit.

Une entrée de 0 à 5 comme indiqué ci-dessous détermine le genre de travail.

0	HP 2600 A, HP 2749 A, ASR-33, ASR-35.
1	EXECUPORT 300
2	ASR-37
3	TERMINET-300
4	MEMOREX 1240
5	UNIVAC DCT 500

Un message "ILLEGAL FORMAT" est imprimé si le code est inférieur à 0 et supérieur à 5.

Si ce code est omis, c'est qu'il s'agit de 0.

BYE

Cette commande permet de sortir l'utilisateur du système.  
Elle imprime le total du temps utilisé par l'utilisateur.

o o  
o

MESSAGE - CHARACTER STRING

Envoie une chaîne de caractères à l'opérateur du système,  
précédée du numéro de console de l'utilisateur.

Si la zone message du système est pleine l'avertissement

CONSOLE BUSY

sera présenté sur la console d'utilisateur, indiquant que le  
message n'a pas été transmis et doit à nouveau être entré.

o o  
o

TIM

Cette commande demande le temps utilisé depuis l'entrée  
ainsi que le total du temps utilisé.

Le temps employé pour chaque "IDCODE" est enregistré  
automatiquement par le système temps-partagé. C'est l'opérateur  
du système qui contrôle le temps total de tous.

o o  
o

## DISC

Cette commande indique la somme totale de l'espace disque affecté, ainsi que l'espace déjà utilisé par le système.

L'espace disque utilisé par un ID est enregistré automatiquement chaque fois qu'une commande "SAVE", "OPEN" ou "KILL" est employée. La somme totale de l'espace disque est affectée et contrôlée par l'opérateur système.

o o  
o

## ECHO-ON

## ECHO-OFF

Cette commande permet d'utiliser la console en mode "HALF-DUPLEX" ou "FULL-DUPLEX".

Un utilisateur en mode "HALF-DUPLEX" doit d'abord entrer dans le système. Ensuite frapper "ECHO-OFF", à ce moment là, l'entrée-sortie devient lisible.

"ECHO-ON" ramène l'utilisateur au mode "FULL-DUPLEX".

o o  
o

## NAME - Programm name de 1 à 6 caractères

Cette commande affecte un nom au programme qui est actuellement en cours de frappe à la console.

Le premier caractère du nom d'un programme utilisateur ne doit pas être un \$, ce signe étant réservé aux programmes du système.

o o  
o



APPEND - nom du programme

APPEND § nom du programme de la bibliothèque système

L'objet de cette commande est de chercher le programme spécifié par le nom et de l'ajouter au programme actuellement en cours.

"RUN ONLY" définit des programmes qui ne peuvent pas être listés, perforés ou sauvegardés.

Si un programme de la bibliothèque publique est ajouté (commande "APPEND") à un programme en cours et si le programme de la bibliothèque était "RUN ONLY" tout le programme prend le même statut.

o o  
o

PROTECT - nom

UNPROTECT - nom

L'utilisateur avec "IDCODE A000" peut sauvegarder des programmes et des fichiers qui sont accessibles à chaque utilisateur du système. De plus, les 2 commandes ci-dessus ne sont disponibles que pour des utilisateurs de "IDCODE "A000".

Des programmes protégés pourront être exécutés par tout utilisateur, mais ils ne pourront être perforés, listés, modifiés ou sauvegardés que par l'utilisateur de "IDCODE A000".

Les fichiers protégés ne sont accessibles qu'avec le code "A000".

Avec le code "A000", un programme déclaré "UNPROTECT" devient entièrement accessible sur tous les plans à tout utilisateur.

De même, un fichier qui devient "UNPROTECT" devient accessible à tout utilisateur.

o o  
o



## CATALOG

Cette commande produit le listing, dans l'ordre alphabétique de tous les programmes et fichiers sauvegardés dans le système.

La longueur de chacun est indiquée par deux mots de 2 caractères, celle du fichier est précédée d'une lettre F.

Exemple :

```
PROG1 0024    PROG2 2348    PROG3 1489
```

```
o  o  
o
```

## LIBRARY

Cette commande produit le listing, dans l'ordre alphabétique, de tous les programmes et fichiers système sauvegardés.

La longueur de chacun est indiquée par deux mots de 2 caractères, celle du fichier est précédée d'une lettre F.

```
o  o  
o
```

## LENGTH

Cette commande imprime le nombre de mots (chaque mot a 2 caractères) constituant le programme. C'est la somme de l'espace d'enregistrement nécessaire pour sauvegarder le programme.

Chaque programme dispose d'un espace approximatif de 4000 mots. La commande "LENGTH" est nécessaire pour vérifier la longueur quand il s'agit d'un long programme.

```
o  o  
o
```

## SCRATCH

Cette commande écrase dans la mémoire centrale le programme qui vient d'être chargé.

o o  
o

## TAPE

Prévient la machine que l'entrée suivante se fera à partir de la bande perforée.

La réponse du système temps-partagé à la commande "TAPE" est un "LINE FEED".

o o  
o

## KEY

Cette commande informe le système que l'entrée se fera au clavier "KEYBOARD".

Il ne faut employer cette commande qu'une fois que l'exécution de la commande "TAPE" est terminée.

o o  
o

## RUN

C'est la commande qui permet l'exécution d'un programme dès le début.

o o  
o

## BACKSPACE (+)

La touche de la flèche inverse, annule le caractère qui précède immédiatement la flèche.

o o  
o

## ANNEXE 3

### LES DIFFERENTES TACHES DU SOUS SYSTEME "LOADER UTILITY"

Le "LOADER UTILITY" de HP 2000 E, réside sur le disque du système temps-partagé.

Le minimum de l'équipement matériel nécessaire est d'abord une "UNITE" de bande magnétique et ensuite une "UNITE" de disque.

#### DETAIL DES OPERATIONS DU "LOADER UTILITY"

#### SAUVEGARDE ET CHARGEMENT DU SYSTEME TEMPS-PARTAGE

##### a) Sauvegarde sur bande magnétique

1. Quand la commande "SLEEP" est entrée, (cf. Annexe 2) le système demande :

MAG TAPE SLEEP ?

On répond                      YES

2. La commande "SLEEP" appelle le "LOADER UTILITY" et lui passe le contrôle :

3. Le "LOADER UTILITY" demande

MAG TAPE ?

A ce moment là, on entre le numéro du canal des données de l'unité "BANDE MAGNETIQUE".

4. Pour la suite voir la commande "SLEEP" (cf. Annexe 2), pour le chargement du "LOADER UTILITY" et le contrôle de ce dernier, voir II.1, initialisation et lancement.

b) Chargement du système temps-partagé à partir de la bande magnétique, voir Annexe 4

Copie disque à disque

1. A l'astérisque (cf. Annexe 4) du "LOADER UTILITY" on répond :

COPY, SUBCHANNEL 1, SUBCHANNEL 2

Le subchannel 1 est un entier (0-3) représentant le disque d'origine.

Le subchannel 2 est un entier (0-3), représentant le disque de destination.

Le subchannel 1 et le subchannel 2 ne peuvent pas être identiques.

2. Le contenu du disque spécifié par le subchannel 1 est copié sur le disque du subchannel 2.

Si un mauvais numéro de subchannel est entrée ou si le disque n'est pas prêt un diagnostic est imprimé et la commande "COPY" doit à nouveau être entrée.

Chargement sélectif à partir de la bande magnétique

1. En réponse à l'astérisque on entre

SLOAD, SUBCHANNEL, SELECT-CODE, (FILE)

Subchannel est un entier, 0-3 représentant le disque destinataire.

Select-code est le plus petit code du canal d'origine de la bande magnétique.

File représente le fichier choisi pour être chargé, si File n'est pas spécifié, le premier fichier sera chargé.

2. Le contenu de la bande magnétique est écrit sur le disque. Si le fichier est précisé, la bande magnétique est avancée jusqu'au fichier indiqué avant que le chargement ait lieu.

Si la bande magnétique ou le disque n'est pas prêt un diagnostic est imprimé et la commande "SLOAD" doit être tapée à nouveau.

### Copie sélective sur la bande magnétique

1. En réponse à l'astérisque on entre :

SDUMP, SUBCHANNEL, SELECT-CODE, (FILE)

Subchannel est un entier (0-3) représentant le disque d'origine.

Sélect-Code est le code correspondant au canal de données de l'unité de la bande magnétique destinataire.

File, représente un fichier choisi dans la bande magnétique. S'il est omis, le disque est recopié comme premier fichier sur la bande magnétique.

2. Le contenu du disque est copié sur la bande magnétique. Si le fichier est précisé la bande magnétique est avancée jusqu'au fichier indiqué, avant que la copie ait lieu.

Si un mauvais subchannel ou select-code est entré ou si l'unité de bande magnétique ou de disque n'est pas prête, un diagnostic est imprimé ; dans ce cas, la commande "SDUMP" est ignorée et doit être entrée à nouveau.

### Initialisation du disque utilisateur

1. En réponse à l'astérisque nous entrons :

FORMAT, SUBCHANNEL

Le subchannel est un entier (1-3) représentant le disque utilisateur destiné à être initialisé. Le subchannel 0 n'est pas permis car c'est celui du disque système.

2. Le disque est vérifié (recherche des mauvaises pistes). Ensuite le "LOADER UTILITY" demande

LABEL ?

il faut entrer alors une étiquette de 1 à 6 caractères.

3. Le programme répond par le nombre de mauvaises pistes qu'il a rencontrées.

Exemple : Si les pistes 0, 1 ou 2 s'avèrent mauvaises, le message suivant est imprimé

TRACK 0, 1 OR 2 BAD  
can not use this disc

La procédure d'initialisation est terminée.

### Compacter le disque utilisateur

1. En réponse à l'astérisque du "LOADER UTILITY" nous entrons

PACK, SUBCHANNEL

Subchannel est un entier, (1-3) représentant le disque utilisateur à compacter.

Subchannel 0 n'est pas permis car c'est celui du disque système.

2. Chaque piste du disque spécifié est compactée pour éliminer des espaces morts causés par le déplacement des programmes et des fichiers par la commande "KIL".

## ANNEXE 4

### LES DIFFERENTS MODES DE CHARGEMENT DU SYSTEME TEMPS-PARTAGE

#### CHARGEMENT PAR LA BANDE PERFOREE

1. Appuyer sur "HALT", "INTERNAL PRESET" et "EXTERNAL PRESET" de la machine.
2. Placer le "LOADER UTILITY" du système temps-partagé dans le lecteur rapide.
3. Positionner l'adresse de démarrage du "BASIC BINARY LOADER" (B.B.L) à 77700.
4. Mise à 0 du registre "SWITCH" et appuyer sur "LOADER ENABLE".
5. Appuyer sur "RUN" ; le chargeur du système temps-partagé sera lu par le lecteur rapide.
6. Après le "HALT" de la machine, positionner l'adresse du démarrage du "LOADER UTILITY", (Registre P à 2000).
7. Appuyer sur "RUN", le "LOADER UTILITY" répond

2000 E UTILITY PROGRAM

\*

L'astérisque est le message du "LOADER UTILITY" qui annonce que ce dernier est prêt.

On entre la commande "LOAD"

8. Maintenant la procédure du chargement devient un dialogue entre le "LOADER UTILITY" et l'opérateur du système. Le "LOADER UTILITY" pose d'abord une question au sujet de la bibliothèque disque.

LIBRARY ?

Comme il s'agit d'un nouveau système il n'existe pas de bibliothèque et on entre "NO".

9. Le "LOADER UTILITY" demande :

SYSTEM IDCODE ?

Entrer une valeur décimale entre 0 et 65532 représentant le code identification du système.

10. Le "LOADER UTILITY" demande :

NUMBER OF PORTS ?

Entrer une valeur décimale de 1 à 16 représentant le nombre de consoles permises.

Si la réponse de l'opérateur est seulement un retour, le système le compte pour 16.

11. A ce point là, le système temps-partagé est lu à partir de la bande perforée, quand la lecture est terminée le "LOADER UTILITY" demande :

DATE ?

On entre la date sous la forme suivante :

DDD/YY

DDD est le jour actuel de l'année (1-366).

YY les 2 derniers chiffres de l'année en cours.

12. Le "LOADER UTILITY" demande :

TIME ?

L'entrée du temps a lieu sous la forme :

hh mm



hh heure de l'opération.  
mm minute de l'opération.

### 13. Le système imprime

READY le chargement est terminé.

#### Chargement à partir du disque

Si le système temps-partagé 2000 E a été sauvegardé précédemment sur le disque (cf. commande "SLEEP"), le chargement à partir du disque peut être effectué par le "LOADER" (chargeur) "BOOTSTRAP" de 2000 E.

Au début on charge le chargeur "BOOTSTRAP" par "BASIC BINARY LOADER" (B.B.L).

Ensuite on positionne l'adresse du démarrage (Registre P à 2), et on appuie sur "RUN".

1. Le "BOOTSTRAP" répond :

IS SYSTEM ON SUBCHANNEL 0 ?

2. Si le système réside dans le canal 0 et le disque utilisateur dans le canal 1, entrer "YES" et aller à l'étape 5.
3. Si le système réside dans le disque cartouche entrer "NO".
4. Le système est copié sur le disque fixe (canal 0) et on aura un "HALT" 102033.

Remplacer le disque cartouche par le disque utilisateur. Ensuite préparer l'unité de disque ; appuyer sur "RUN".

5. Le système est chargé à partir de disque. En complément du chargement de système le "BOOTSTRAP" demande :

DATE ?

Entrer la date de la forme DDD/YY

DDD le jour actuel de l'année.

YY les 2 derniers chiffres de l'année en cours.

6. Le "BOOTSTRAP" demande :

TIME ?

Entrer l'heure sous forme hh mm

hh l'heure de l'opération

mm minute de l'opération.

7. Le système répond

READY

Le chargement est complet.

Si le disque système, issu de la commande "SLEEP", n'est pas celui utilisé pour le chargement par disque, le message suivant est fourni :

INVALID DISC

Il faudrait placer le disque correct et retourner à l'étape 1.

Chargement du système temps-partagé à partir de la bande magnétique

Pour charger le système temps-partagé 2000 E à partir de la bande magnétique, le "LOADER UTILITY" doit être chargé d'abord à l'adresse 77700 en employant le B.B.L.

Ensuite on positionne l'adresse du démarrage du "LOADER UTILITY" à 2000.

Puis on appuie sur "INTERNAL PRESET", "EXTERNAL PRESET" et "RUN".

Le "LOADER UTILITY" répond

2000 E UTILITY PROGRAM

\*

L'astérisque est le signe qui annonce la présence du "LOADER UTILITY".

1. Pour charger le système par la bande magnétique entrer :

LOAD, SELECT-CODE

Select-code est le numéro du canal de données de l'unité de bande magnétique.

Dans notre cas c'est 20.

2. Le "LOADER UTILITY" charge le contenu de la bande magnétique sur le disque, qui pourrait ensuite être chargé dans la machine et être initialisé.
3. Si l'unité de bande magnétique ou de disque n'est pas prête, un diagnostic est imprimé, préparer l'ensemble et appuyer sur "RUN".

o o  
o

## BIBLIOGRAPHIE

Les ouvrages référencés ci-après concernent exclusivement l'organisation et l'utilisation des systèmes temps-partagé.

### LIVRES, RAPPORTS, THESES, en français

- Exploitation partagée des calculateurs.  
J.Bertin, M.Ritout, J.C.Rougier  
Dunod, Paris (1967)
- Emploi partagé des calculateurs.  
J.Arsac, M.Dupuy, M.F.Leroch  
Institut de programmation, rapport convention DRME (1970)
- Gestion des tampons dans le système à temps-partagé.  
Girault  
Doctorat d'Etat ès Sciences, Université de Paris VI (1972)
- Une analyse de système.  
Lesoudier  
Doctorat de 3<sup>e</sup> cycle, Université de Grenoble (1973)
- Elaboration d'un système conversationnel pour l'enseignement assisté par ordinateur.  
J.M.Adamo  
Doctorat de 3<sup>e</sup> cycle, Université Claude Bernard Lyon (1973)
- Initiation à l'informatique et Méthodes pédagogiques actives, étude et réalisation par des débutants, du système ENS LOT destiné à l'initiation à l'informatique.  
J.L.Leanhardt  
Doctorat de 3<sup>e</sup> cycle, Université de Lyon (1974)

- Le système de fichier d'un système de multiprogrammation en temps-partagé. Pagination multiple.  
G.Brenaud  
Doctorat de 3<sup>e</sup> cycle, Paris VI (1974)
- Système à temps-partagé réalisé sur un petit ordinateur.  
Y.Chambaret  
Doctorat de 3<sup>e</sup> cycle, Informatique, Université de Lyon (1973)
- Système d'exploitation en temps-partagé. Etude d'un système de pagination programmé.  
C.Barbe  
Doctorat-Ingénieur, Orsay (1971)
- Mécanismes de base dans les systèmes superviseurs. Conception et réalisation d'un système à accès multiples.  
J.Bellino  
Doctorat d'Etat ès Sciences appliquées. Grenoble (1973)
- Un support de terminaux lourds sous un système à temps-partagé.  
Z.Papacristodoulou  
Doctorat-Ingénieur, Grenoble (1974)
- Exécutif d'un système de multiprogrammation en temps-partagé.  
G.Fourmentaux  
Doctorat de 3<sup>e</sup> cycle Informatique, Paris VI (1974)
- Vérification de programmes utilisant des pointeurs.  
D.Bert  
Université de Grenoble (1976)
- Outils informatiques d'aide à l'enseignement de l'algorithmique. Analyse, Prospective.  
P.C.School, M.Cabric  
Université de Grenoble (1976)
- Validité des simulations de files d'attente.  
M.Becker  
Doctorat d'Etat ès Sciences, Université Paris VI (1976)

LIVRES, RAPPORTS, THESES, en anglais

- "Multilevel Processor - Sharing Queueing Models for Time-shared models".  
Kleinrock L., and Muntz R.R.  
Proc.Sixth International Teletraffic Congress, (1970)
- Tight Bounds on Average response time for Processor-Sharing Models of Time-shared Computer systems.  
Kleinrock L., Muntz R.R., and Hsu J.  
Proc.International Federation for Information Processing Congress (1971)
- A selected menu of analytical results for time-shared computer systems.  
Kleinrock L.  
Germany (1972)
- Analysis of a continuum of processor-sharing models for time-shared computer systems.  
Hsu J.  
School of engineering and applied science, Univ.of California, Los Angeles (1971)
- Analysis of a time-shared processor.  
Kleinrock L.  
Naval Research Logistics Quarterly (1964)
- "Time-shared systems : a theoretical treatment".  
Kleinrock L.  
JACM (1967)
- A continuum of time-sharing scheduling algorithms.  
Kleinrock L.  
AFIPS Conference Proc., spring joint computer Conference (1970)
- Program and addressing structure in a time-sharing environment.  
Arden B.W., Galler B.A., O'Brien T.C., Westernell F.H.  
Journal of the Association for Computing Machinery (J.A.C.M.)  
january (1966)

- Analysis of drum input/output queue under scheduler operation in a paged computer systems.  
Coffman E.G. (J.A.C.M. january 1969)
- Analysis and optimisation of disk storage devices for time-sharing systems.  
J.A.C.M. october (1969)
- User program measurement in a time-shared environment.  
Nemeth A.G. and Rouner P.D.  
C.A.C.M. october (1971)
- Stochastic models of multiple and time-shared computer operations.  
Coffman E.G.  
Report N° 66-38, Dept. of Engineering, University of California at Los Angeles (1966)
- Waiting time distribution for processor-sharing systems.  
Coffman E.G. Jr., Muntz R.R. and Trotter  
J.A.C.M. (1970)
- Measures, Models and Measurements for time-shared computer utilities, Proc. of 22<sup>nd</sup> National Conference of the Association for Computing Machinery.  
Estrin G. and Kleinrock L.  
Washington, D.C. (1967)
- Distribution of attained and residual service in general queueing systems.  
O'Donovan T.M.  
Operations Research (1974)
- Analysis of computer architectures for information retrieval.  
Hurley (Bernard John)  
University of Illinois (1976)
- Time-sharing system design concepts.  
Watson R.W.  
Department of electrical engineering and computer science,  
University of California Berkeley, Mc Graw Hill U.S.A. (1976)

- Time-sharing computer systems  
Wilkes M.V.  
Mc Donald and Co, London (1968)
- An analysis of time-shared computer systems.  
Allan Lee Scherr  
Research Monograph n° 36, the M.I.T. Press, Cambridge,  
Massachusetts (1967)
- Time-shared data processing systems  
Ziegler J.R.  
Prentice-Hall, Inc U.S.A. (1967)
- Conversational Computers  
William D. orr  
John Wiley and Sons, U.S.A. (1968)
- 2000 E Time-shared Basic systems operators guide.  
Hewlett Packard, U.S.A.
- 2000 E A guid to time-shared basic.  
Hewlett Packard, U.S.A.
- The simulation of time-sharing systems  
J.A.C.M. July (1967)
- Performance criteria and measurement for a time-sharing system.  
Bard Y.  
I.B.M. systems journal N° 3 (1971)
- A consistent and complete detective system for the verification  
of parallel programs.  
Owicki S.  
Cornell University Ithaca, New-York (1976)
- Processor multiplexing in a layered operating system.  
Reed D.P.  
M.I.T. (1976)



- Analysis and synthesis of time-shared computer systems.  
Nilsson A.  
Lund Institute of Technology, Sweden (1976)
  
- A computational algorithm for queue distributions via Poly.  
Theory of enumeration  
Kobayashi H.  
I.B.M. Research division (1976)
  
- Modular multiplexing computer interface system.  
Ayers G.L., Horn D.E.  
I.B.M. Research division (1976)

