



division petits ordinateurs  
et applications systèmes

mitra 15

manuel d'utilisation

COMPAGNIE INTERNATIONALE POUR L'INFORMATIQUE

## Génération de système

**Gamme** : MITRA 15

**Systèmes** : MTR, MTRD, MMT

**Objet** : Cette mise à jour corrige quelques anomalies de la première version du manuel.

**Remarques** :

**Nombre de pages** : 10

**Date d'édition** : SEPTEMBRE 1975

Pour commander ce document,  
envoyez votre demande à l'adresse  
ci-contre en reproduisant intégrale-  
ment la "référence document".

Compagnie Internationale pour l'Informatique  
CIDOC 68, route de Versailles 78430. LOUVECIENNES

Référence document  
4255 U/FR REV.

1977-1978



division petits ordinateurs  
et applications systèmes

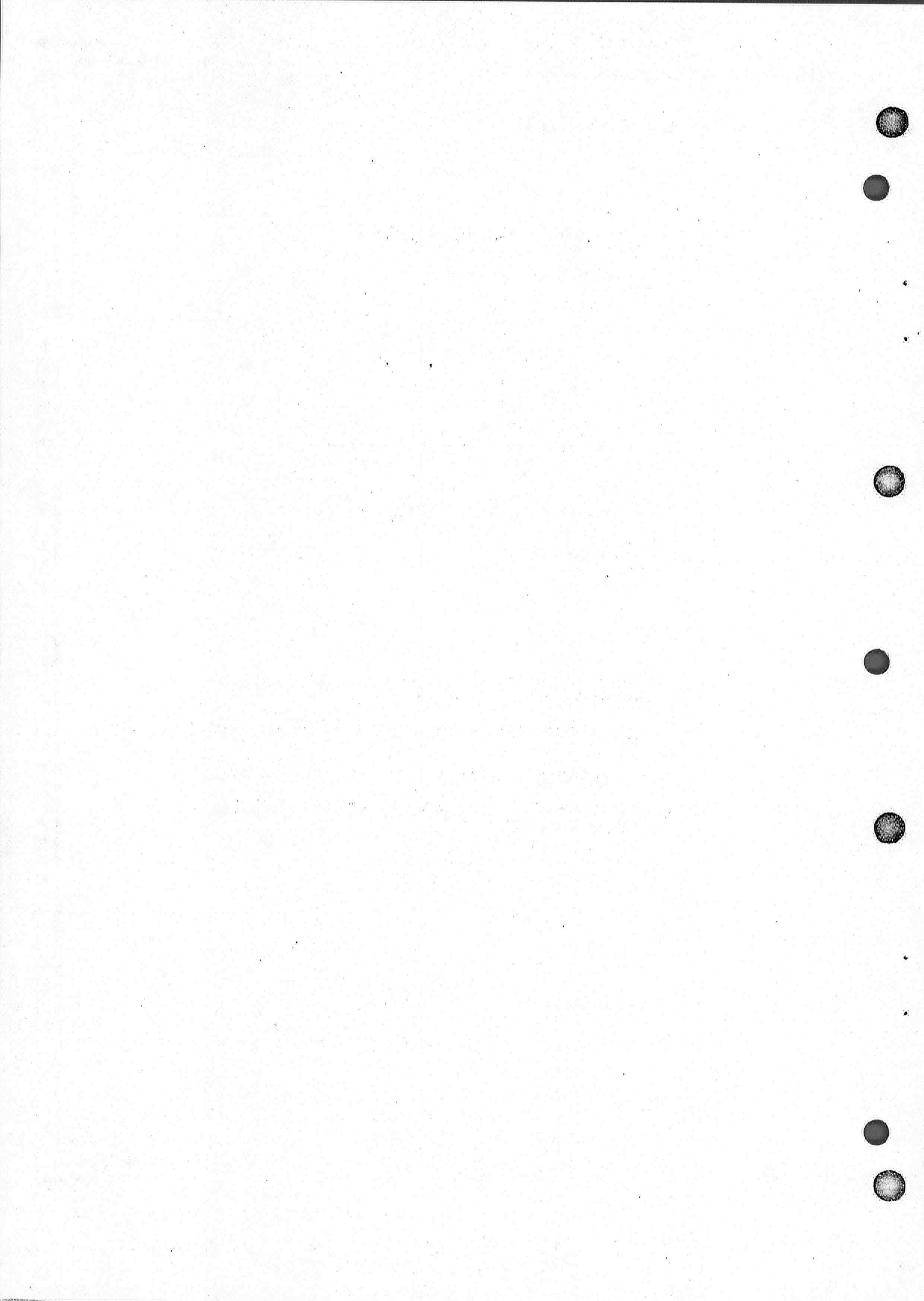
mitra 15

manuel d'utilisation

COMPAGNIE INTERNATIONALE POUR L'INFORMATIQUE

## Génération de système

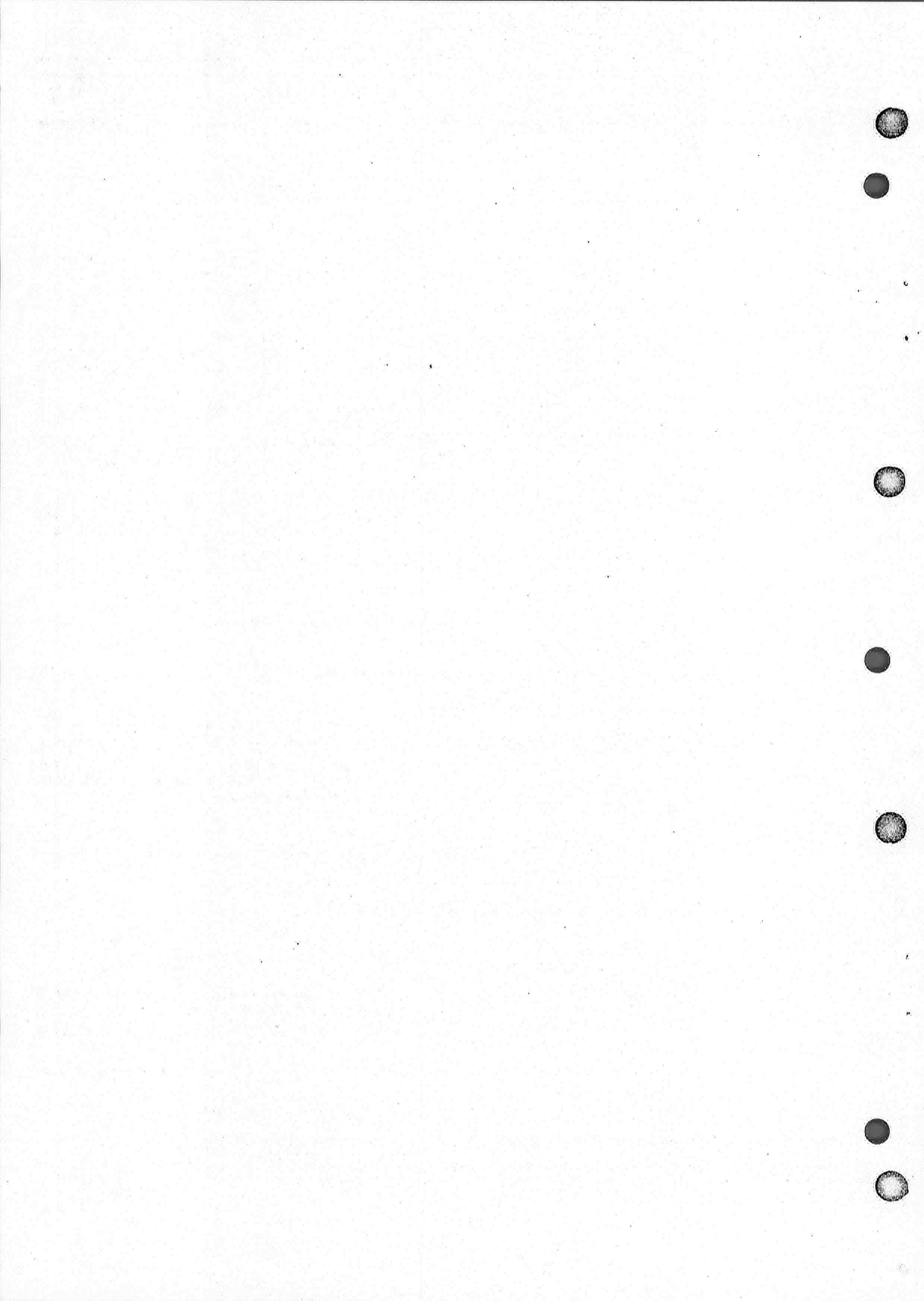
Gamme	:	MITRA 15
Systèmes	:	MTR, MTRD, MMT
Objet	:	<p>Ce manuel décrit les différents processeurs utilisés pour la génération et la sauvegarde d'un ou de plusieurs moniteurs.</p> <p>Les processeurs concernés sont les suivants : GSYSS, SYOUTE, DISAVE, GENIOT.</p> <p>On y trouve également des informations sur les modules de chargement.</p>
Remarques	:	<p>Ces processeurs sont utilisables avec les moniteurs MTR, MTRD et MMT dont l'indice est au moins égal à 05.</p>
Nombre de pages	:	37.
Date d'édition	:	SEPTEMBRE 1975



# Génération de système

## Sommaire

<b>1. GSYSS : Générateur de système</b>	1-1
Introduction	1-1
Mise en œuvre	1-1
Commandes de génération	1-2
Messages d'erreur du générateur	1-6
Mnémoniques des types de périphériques	1-8
Listes de génération	1-9
Organisation logique du format IMA sur disque	1-10
Organisation logique du format IMA sur support physique	1-10
Tableau synoptique montrant les interfaces d'entrée/sortie	1-11
Exemple de commandes de génération (sous BATCH)	1-12
<b>2. SYOUTE : Générateur de ruban/cartes système</b>	2-1
Introduction	2-1
Mise en œuvre	2-1
Format des commandes	2-1
Messages d'erreur de SYOUTE	2-3
Exemple de génération	2-3
<b>3. DISAVE : Sauvegarde d'un système disque sur support externe</b>	3-1
Introduction	3-1
Mise en œuvre	3-1
Format des commandes	3-1
Messages d'erreur de DISAVE	3-3
Exemples de sauvegarde de système	3-3
<b>4. Modules de chargement</b>	4-1
Introduction	4-1
Organisation des modules	4-1
Commande de chargement	4-2
Mise en œuvre	4-3
Messages d'erreur	4-3
Modification au chargement d'un système	4-4
Exemples de chargement	4-4
<b>5. GENIOT : Générateur des tables d'entrée/sortie</b>	5-1
Introduction	5-1
Mise en œuvre	5-1
Description des tables d'entrée/sortie	5-1
Mode d'utilisation	5-6
Exemple	5-7
Messages d'erreurs	5-9



# 1. GSYSS : Générateur de système

## 1-1. Introduction

Le programme "GSYSS" est un processeur MITRA 15 destiné à produire un système au format Image Mémoire Absolue (I.M.A.), chargeable, à partir d'éléments au format Image Mémoire Translatable (moniteur, handlers, tâches immédiates...)

Ce processeur fonctionne en deux passes.

A chaque passage, les différents IMT constitutifs du système seront analysés depuis leur support respectif : support physique (ruban perforé, cartes perforées, bande magnétique...) au support disque (bibliothèque des IMT 'EP'; bibliothèque de manœuvre des processeurs 'GI-GO').

Avant la première passe le processeur traite des commandes qui déclarent les IMT à introduire dans le système.

La production d'IMA se fait au cours de la seconde passe sur le support indiqué (physique : ruban, carte, bande magnétique... ou disque : zone de manœuvre 'GO').

### Remarque :

Dans le cas d'une sortie sur support physique (ruban, cartes, bande magnétique...) pour que le système soit auto-chargeable, il est impératif de le faire précéder des modules d'auto-chargement BOOTPR (ruban), ou BOOTCR (cartes) ou BOOT9T (bandes magnétiques) et PRECHA et CHARG, et ce, à l'aide du processeur SYOUTE.

## 1-2. Mise en œuvre

### 1-2.1. Sans enchaînement

"GSYSS" est chargé et lancé au moyen des commandes :

- %LOAD
- %RUN

- Le processeur édite alors le message : '%GSYSS/' à la suite duquel l'opérateur frappe les options de génération en terminant par les 2 caractères 'line-feed' 'retour-chariot'

- Après analyse de ces options, le générateur lit la succession des commandes jusqu'au %EOD, à la suite duquel il entame l'examen en deux passes des modules déclarés. Les commandes sont lues sur l'étiquette opérationnelle M:CI.

### 1-2.2. Avec enchaînement

#### Par le moniteur d'enchaînement : BATCH

L'enchaînement se fait par le moniteur BATCH dans un JOB par la commande : %CALL/GSYSS/ + options de génération.  
(le processeur lisant les commandes placées derrière %CALL)

#### Par le moniteur à l'aide de la commande :

%C/GSYSS/ + options.

### 1-2.3. Assignations des organes d'entrée/sortie

Dans le cas du lancement par %L, %R, le titre est édité sur l'ASR33 de service "à travers" M:OC; les options sont entrées au clavier depuis M:OC.

Dans le cas du lancement par le BATCH la commande et les options sont entrées par le BATCH sur M:CI.

Dans le cas d'un lancement par %CALL les options sont traitées depuis M:OC (clavier ASR33).

- L'entrée des commandes de génération se fait depuis M:CI
- Les supports d'entrées sont assignés à :  
M:BI (support non disque)  
M:GI ou M:EP (support disque)

- Les supports de sorties sont assignés à :  
M:BO (non disque)  
M:GO (support disque : zone de manœuvre des processeurs)

**Pas d'option présente** : sortie de l'IMA sur M:GO.

- Les listes sont éditées sur M:LL
- Les diagnostics d'erreurs sont éditées sur :  
M:LL (fonctionnement BATCH)  
M:OC (fonctionnement non BATCH)

Dans le cas d'un fonctionnement sous moniteur BATCH, tout diagnostic d'erreur (mineure ou majeure) provoque l'abandon de la génération.

Les erreurs mineures sont réparties en 3 groupes :

- erreur de syntaxe dans l'analyse d'une commande
- erreur d'entrée/sortie sur périphérique non opérationnel
- débordement du nombre de handlers permis.

Les erreurs majeures sont réparties en 3 groupes :

- erreurs sur les modules IMT
- erreurs irrécupérables d'entrée/sortie
- erreurs de dépassement de capacité mémoire.

Les erreurs majeures entraînent toujours l'abandon de la génération quelque soit le mode d'enchaînement.

### 1-3. Commandes de génération

#### 1-3.1. Options de génération

$$\% \left[ \left\{ \begin{array}{c} C \\ CALL \end{array} \right\} / \right] \text{GSYSS} / \left[ \left\{ \begin{array}{c} BO \\ GO \end{array} \right\} \right] [,\text{SEQ:} [ \& ] \text{xxxx} ] , (\text{nom} [ , [ \& ] \text{vv} ] )$$

**BO** (Binary Output) le module IMA généré se trouvera sur un support physique (ruban - carte - bande...) à travers l'étiquette opérationnelle M:BO.

**GO** Le module IMA généré se trouve sur disque dans le fichier de manœuvre GI-GO des processeurs.  
L'option GO est implicite.

**SEQ:xxxx** précise la valeur du pas de séquençement apparent à contrôler. Si cette option est absente, il n'y a pas de contrôle du séquençement apparent (octets 104 à 119 des blocs IMT sur support externe [ex: sur carte, séquençement des colonnes 73 à 80])

**(nom)** donne le nom du système à générer.  
(1 à 6 caractères alphanumériques dont le premier au moins est une lettre).

**vv** précise le numéro de version du système à générer, avec  $0 < vv < 9$

Ce nom et ce numéro de version sont normalement éditées par le module de chargement "INIT" lors de l'introduction du système.

#### 1-3.2. Définition de module résidant

$$\text{SV} / (\text{nom} [ , [ \& ] \text{vv} ] ) \left[ \left\{ \begin{array}{c} BI \\ GI \\ EP \end{array} \right\} \right]$$

**SV** déclaration de module Superviseur.  
Elles est essentiellement composée de modules communs exécutant des services, appelables par "CSV"

**BI** le module moniteur est sur support externe au format IMT (blocs de 120 octets dont 4 octets de contrôle.)

**GI** le module se trouve dans la zone de travail des processeurs

**EP** le module est situé dans la bibliothèque des programmes exécutables EP.

**(nom)** c'est le nom (1 à 6 caractères) du module IMT superviseur donné par l'éditeur de liens.

**vv** c'est le numéro de version du module IMT.



Cette déclaration doit précéder toute autre déclaration de handler, de tâche immédiate ou de module commun.

### 1-3.3. Définition de module programme commun

Il existe à cet effet 3 commandes :

$$SC / (\text{nom} [ \cdot [ \& ] \text{vv} ] ) \left[ \left\{ \begin{array}{c} BI \\ GI \\ EP \end{array} \right\} \right] [ , N [ \& ] \text{nnn} ] [ \left\{ \begin{array}{c} i \\ R \end{array} \right\} [ \& ] \text{xxx} ]$$

Définit un module IMT de sous-programme constitué d'au moins une section programme LPS, à insérer ou à remplacer dans le module superviseur, handler ou tâche immédiate déclaré précédemment.

— Sous-programme ou sous-programme commun.

BI le module IMT est sur support non disque.

GI le module est dans la zone de travail des processeurs GI-GO.

EP le module est la bibliothèque des programmes exécutables.

(nom) nom du module commun IMT donné à l'édition des liens.  
(1 à 6 caractères alphanumériques).

vv précise le numéro de version du module  
 $0 \leq \text{vv} \leq 9$

Nnnn nombre de sections LPS du module IMT  
N = 1 implicite.

Ixxx précise le numéro de section du module SV à partir duquel le générateur insérera la ou les sections données par le module.

Rxxx précise le numéro de section du noyau SV à partir duquel le générateur substituera la ou les sections données par le module.

$$SP / (\text{nom} [ \cdot [ \& ] \text{vv} ] ) \left[ \left\{ \begin{array}{c} BI \\ GI \\ EP \end{array} \right\} \right] [ , N [ \& ] \text{nnn} ]$$

Définit un module IMT de sous-programme constitué d'au moins une section programme LPS, à insérer devant la dernière section du module superviseur déclaré précédemment.

BI le module IMT est sur support physique.

GI le module IMT se trouve dans la zone de travail des processeurs GI-GO.

EP le module IMT est en bibliothèque EP des programmes exécutables.

(nom) nom du module commun à l'édition des liens  
(1 à 6 caractères alphanumériques).

vv numéro de version du module.

Nnnn nombre de sections LPS du module IMT.  
N = 1 implicite.

Cette commande permet l'insertion automatique des modules IMT devant la section M:INIT laquelle doit être la dernière (section de lancement du superviseur).

$$DR / (\text{nom} [ \cdot [ \& ] \text{vv} ] ) \left[ \left\{ \begin{array}{c} BI \\ GI \\ EP \end{array} \right\} \right] [ , N [ \& ] \text{nn} ]$$

Définit le module IMT de traitement des dérivations. Ce module, constitué d'au moins 1 section programme LPS sera inséré en tête de tous les modules CSV du superviseur dans le PRTS et il prendra automatiquement le numéro 0. Physiquement, il sera placé après la section de lancement M:INIT.

BI Le module IMT est sur support physique.

GI le module est dans la zone GI-GO du disque.

- EP le module est en bibliothèque des programmes exécutables.
- (nom) c'est le nom du module IMT donné à l'édition des liens (1 à 6 caractères alpha-numériques).
- vv c'est le numéro de version du module.
- Nnn nombre de sections du module (LPS) de traitement des dérivements.  
N=1 implicite.

Cette commande permet l'insertion automatique du module de traitement des dérivement M:TRAP, sans spécification de niveau d'insertion.

**Nota :**

Pour les commandes d'insertion ou remplacement de sous-programmes SC, SP, DR, le module d'insertion présenté doit posséder une base locale propre : le module IMT peut comporter de 1 à n sections programmes LPS et au moins une section de données LDS. (ce qui est le cas pour le module de dérivement M:TRAP..)

**Exemple :**

- 1 - SV/(MTRDE.4), GI
- 2 - DR/(TRAPO3.2)
- 3 - SC/(M:TRAC.3), N3, I23, EP
- 4 - SP/(M:ANAC.6)

- 1 - le module superviseur 'MTRDE' version 4 est dans GI; il comporte 47 sections.
- 2 - le générateur insérera le module de traitement des dérivements 'TRAPO3' version 2 dans le noyau SV en première section (numéro 0). Ce module DR est issue d'une LDS et d'une LPS éditées dans EP.
- 3 - le générateur insérera 3 sections (mise au point dynamique) de nom 'M:TRAC' version 3 dans le noyau SV à partir des sections 23. Le module est situé dans EP.
- 4 - le générateur insérera la section n° 47 du noyau (devant la dernière) section 'M:INIT'.

### 1-3.4. Définition de module de données

DA / (nom [ . [&] w ])  $\left[ \begin{array}{c} BI \\ GI \\ EP \end{array} \right]$

Définit un module IMT de type 'DATA' constitué d'au moins une section de données 'LDS' à insérer dans le noyau superviseur, handler ou tâche immédiate déclaré précédemment. Cette commande permet en particulier d'insérer une ou plusieurs tables : tables d'E/S du type OLT, DCT, ou tables particulières dans le noyau SV ou dans un handler pour un sous-handler...)

- (nom) nom du module IMT donné à l'édition des liens (1 à 6 caractères). C'est un IMT de type 'DA'  
[donc N=3 du bloc en-tête IMT]
- vv précise le numéro de version de l'IMT.
- BI le module est sur support physique
- GI le module est dans la zone GI-GO
- EP le module est en bibliothèque de programmes EP.

### 1-3.5. Définition de référence

RF/@ [&]xxxx:  $\left\{ \begin{array}{c} L \\ P \\ D \end{array} \right\} \left[ \begin{array}{c} + \\ - \end{array} \right] \left[ \begin{array}{c} + \\ - \end{array} \right] \left[ \begin{array}{c} + \\ - \end{array} \right] [&] dddd \left[ \begin{array}{c} + \\ - \end{array} \right] [,S [&] sss]$

Permet de satisfaire les pointeurs de LDS ou LPS d'un module 'données' ou 'programme' inséré dans le module superviseur. Le générateur complète le relais en page 0 du moniteur.

**Exemple :**

L:TRAC DATA N:TRAC RELAIS LDS MISE AU POINT  
P:WAIT DATA R:WAIT RELAIS M:WAIT INTERNE

où L:TRAC et P:WAIT sont les adresses en page 0 (adresse absolue) respectivement des relais de la LDS des modules de mise au point M:TRAC, M:HALT, M:SNAP, et de la LPS de M:WAIT.

@[&]xxxx adresse absolue en page 0 du relais.  
 L signifie que le relais contiendra une valeur relative à la base locale L de la section insérée.  
 P signifie que le relais pointera sur une instruction de LPS de la section spécifiée.  
 D signifie que le relais contiendra une valeur relative au début des segments de données précédemment inséré.  
 ±dddd précise le déplacement par rapport au début de LDS ou LPS.  
 Ssss donne le numéro de section de référence.

### 1-3.6. Définition de modifications

MO/ [S [&] sss,] [P,] @ [&] xxxx: [&] yyyy<sub>1</sub> [, [&] yyyy<sub>2</sub> [, ... ]]  
 [, @ [&] xxxx: [&] yyyy<sub>1</sub> [, [&] yyyy<sub>2</sub> [, ... ]]] [, @ ... ]

Permet d'apporter des modifications à l'intérieur d'un module IMT en cours de génération (modifications de données LDS ou d'instructions de LPS).

Les commandes MO, RF se rapportent toujours à la définition de module immédiatement précédente : SV - DV - TK - SC - DR - SR ou DA.

S[&]sss précise la section de référence des modifications par son numéro.  
 $1 \leq sss \leq \text{nb de sections de l'IMT}$

P précise que les modifications sont relatives à un segment programme LPS. En l'absence de Ssss, les modifications sont relatives à la base générale du programme ou à la base des DA qui précèdent. Avec Ssss, et P absent les modifications se font par rapport à la section de données LDS.

@xxxx donne l'adresse à partir de laquelle les modifications yyyy vont être apportées.

yyyy modification courante.

### 1-3.7. Définition de handler

DV / (nom [ . [&] w ] )  $\left[ \begin{array}{l} \text{BI} \\ \text{GI} \\ \text{EP} \end{array} \right] \left[ \begin{array}{l} t_1 \quad t_2 \\ \text{[&] g: [&] ii} \end{array} \right] [ , N [ & ] nn ] ]$

Définit un module IMT comme tâche immédiate liée aux transferts d'E/S sur périphérique.

Le générateur insère l'IMT du handler, génère la configuration de contrôle d'IT associée ainsi que le contexte de lancement de l'interruption de fin de transfert. Il place en outre dans une table d'E/S standard "T:IOT1" la base locale du handler au rang correspondant.

(nom) nom du module IMT donné à l'édition des liens. (1 à 6 caractères)

wv numéro de version du module IMT.

BI le module doit être traité depuis un support physique

GI le module est dans la zone de travail des processeurs

EP le module est dans la bibliothèque des programmes

t<sub>1</sub>t<sub>2</sub> mnémonique d'un type de périphérique standard (voir tableau correspondant). Ce type de mnémonique permet d'associer la configuration de contrôle d'IT au handler.

g:ii définit un coupleur par son numéro d'IT 'ii' dans un groupe 'g' avec  $0 \leq g \leq 3$  et  $1 \leq ii \leq 11$  (voir paragraphe 1-5)  
 Cette option est intéressante dans le cas d'un périphérique non standard.

Nnn précise le numéro d'IT ( $1 \leq nn \leq 31$ ) software sur lequel on désire que l'interruption fin de transfert soit traitée en dehors des niveaux standards (voir tableau des niveaux d'IT standard)

#### Remarque :

La base L d'un handler est toujours :  $L = (G) + 32$

### 1-3.8. Définition de tâche immédiate

TK/(nom [,&] w) [ { BI } [,&] g: [,&] ii] [N [,&] nn]

Définit un IMT comme tâche immédiate software.

Le générateur engendre la configuration de contrôle d'IT (C.I.T.) associée ainsi que le contexte de lancement de la tâche.

(nom) nom du module IMT de la tâche (1 à 6 caractères)  
 w numéro de version du module.  
 BI le module est sur support physique.  
 GI le module est dans la zone de travail disque des processeurs.  
 EP le module est en bibliothèque des programmes exécutables.

g:ii définit le coupleur par son numéro ii dans le groupe g avec  $0 \leq g \leq 3$  et  $1 \leq ii \leq 11$ .

Nnn précise le numéro d'IT software non standard sur lequel la tâche sera connectée.

Pour la coupure et le retour secteur, seul le niveau d'interruption (30 et 31) est indiqué car la configuration d'interruption est ineffective.

### 1-3.9 Définition de la taille zone commune

ZC/[&] xxxx

Définit la longueur en octets de la zone commune.

Implicitement cette taille est fixée à 330 octets (plus petite zone permettant le 'run' des processeurs standards : MITRAS 2...)

xxxx taille en octets de la Z.C.

### 1-3.10. Annulation des commandes

XX/

Annule toutes les commandes précédentes et réinitialise le générateur en son début pour l'édition de : "%GSYSS/"

Cette commande n'a de sens qu'en fonctionnement sans enchaînement donc en entrée clavier.

### 1-3.11. Fin d'entrée des commandes

%EOD

Signale la fin des commandes de génération et initialise le travail de la première passe.

### 1-3.12. Remarques

1. La suite des 2 caractères "+" et "RC" annule la commande en cours. Le générateur redonne le contrôle pour la frappe d'une nouvelle commande.
2. La fin d'une commande est caractérisée par la succession de la frappe des 2 caractères "LF" et "RC".
3. La commande SV/ est toujours la première des commandes.
4. Une suite de commandes MO/RF est relative à la déclaration de module immédiatement précédente. (SV, DV, TK, SC, SP, DR ou DA)
5. Le caractère [&] indique que la valeur qui suit est hexadécimale si non elle est considérée comme décimale.

## 1-4. Messages d'erreurs du générateur

### Erreurs mineures

#### Erreurs d'E/S

Les erreurs dûes aux périphériques non opérationnels donnent lieu à l'édition d'un message :

"%%DV" ou %%DVxx où xx est le mnémonique du périphérique incriminé.

#### Erreur de nombre de handlers

La table des bases L des handlers (IOT1) est sous-dimensionnée; les dernières valeurs sont perdues.

"NOMBRE DE HANDLERS MAX : XX"

**. Précision des options du système****OPTIONS**

DA:OUI, NBUL:15, OLD:OUI, OLF:OUI ;

↑ étiquettes opérationnelles foreground présentes  
 ↑ étiquettes opérationnelles disque présentes  
 ↑ nombre d'étiquettes opérationnelles utilisateurs (U:XX)  
 ↑ gestion de la zone DA présente (GOTO de compilation : G:DA=1)

**. Constitution de la table : LIGNES****LIGNES**

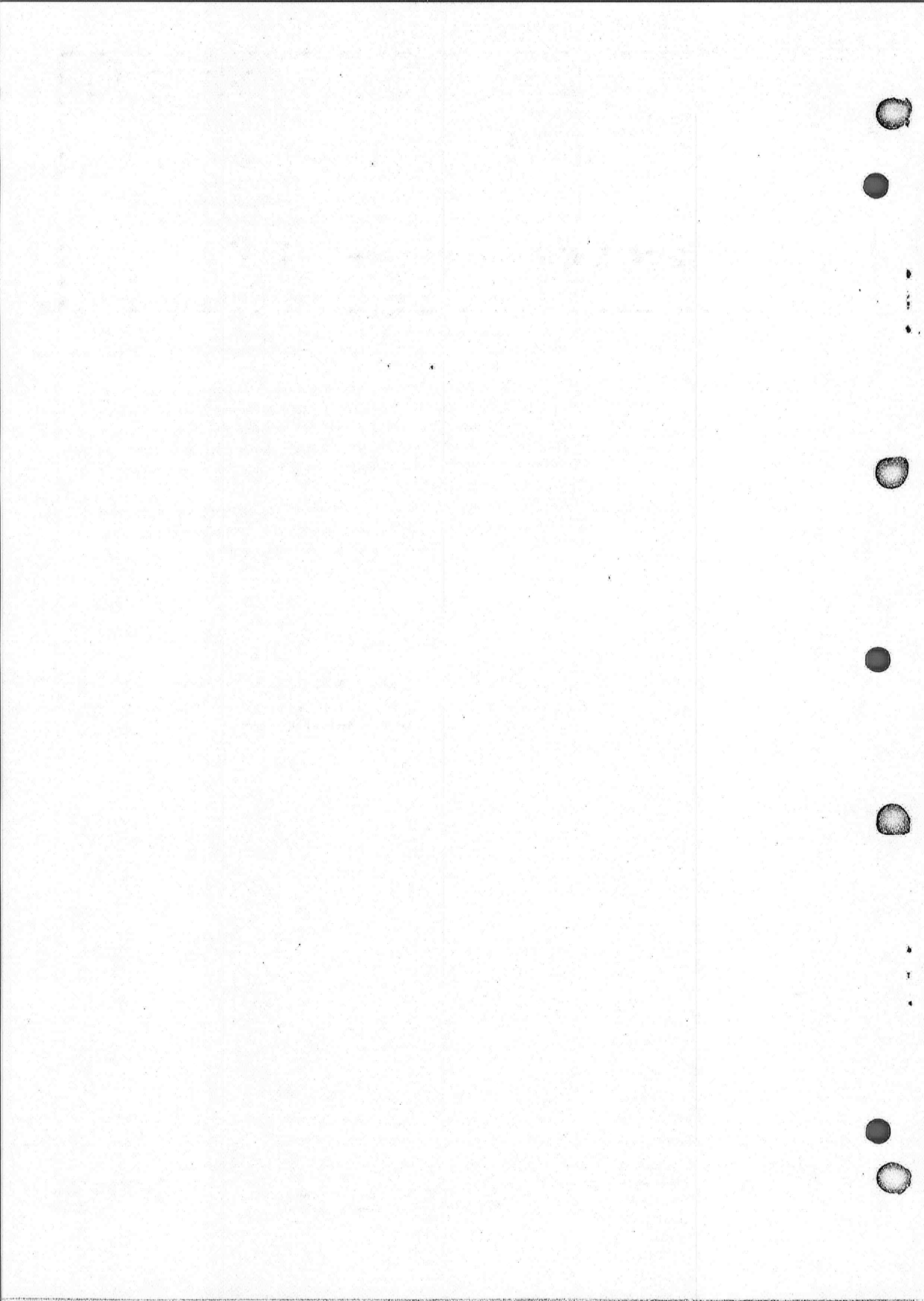
LAS:0 LS:0; ← les lignes asynchrones et synchrones connectées à l'unité de traitement : 0.

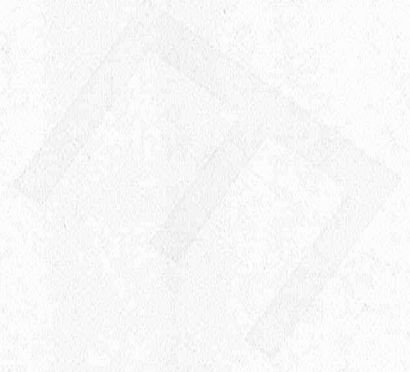
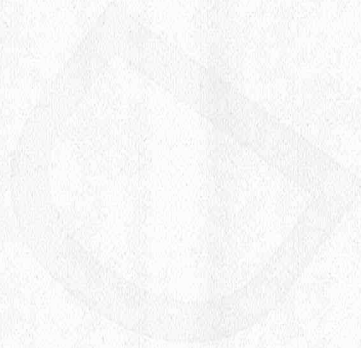
FIN ← indique la fin de la macrogénération.  
 < SWITCH SI > pour la lecture des macroinstructions MAG15 exploitant les données.

**5-6. Messages d'erreurs**

Les messages d'erreurs sont sortis sur l'étiquette opérationnelle : M:LL.

- \*\*\* OL NON CORRECTE : XX \*\*\* : étiquette opérationnelle XX incorrecte.
- \*\*\* PERIPHERIQUE NON CORRECT : XX \*\*\* : mnémonique du périphérique incorrect.
- \*\*\* NUMERO D'UEM NON CORRECT : X \*\*\* : le numéro d'UEM doit être inférieur à 4.
- \*\*\* NUMERO D'UNITE NON CORRECT : X \*\*\* : numéro d'unité incorrect
- \*\*\* ERREUR DE SYNTAXE \*\*\*
- \*\*\* ERREUR DE NIVEAU \*\*\* : XX : un niveau d'interruption est compris entre 0 et 31.
- \*\*\* ERREUR DE CIT \*\*\* : XXX : configuration d'interruption non conforme.
- \*\*\* PERIF INCONNU : XX \*\*\* : nom du périphérique inconnu.







COMPAGNIE INTERNATIONALE POUR L'INFORMATIQUE

Société anonyme  
au capital de 143 700 00 F  
INSEE 297.78.350.0005  
R.C. Versailles 66 B 576

Centre de Vélizy  
10-12 avenue de l'Europe  
78-Vélizy  
tél. 946.96.70

Siège social  
et Division des Ordinateurs  
et des systèmes Informatiques :  
68 route de Versailles  
78-Louveciennes  
tél. 951.86.00

Centre des Clayes-sous-Bois :  
Avenue Jean-Jaurès  
78-Les Clayes-sous-Bois  
tél. 460.14.00

Usine de Toulouse :  
Avenue du Général-Eisenhower  
31-Toulouse 03  
tél. 40.44.88

DÉLÉGATIONS RÉGIONALES

**Rhône-Alpes**

18 avenue Félix-Faure  
69-Lyon  
tél. 69.14.36

15 bd du Général-Leclerc  
38-Grenoble  
tél. 44.99.22

**Centre-Ouest**

9 place Rouget-de-Lisle  
37-Tours  
tél. 53.22.09

**Midi-Pyrénées**

avenue du Général-Eisenhower  
31-Toulouse - Le Mirail  
tél. 40.44.88

**Sud-Est**

433 rue Paradis  
13-Marseille  
tél. 77.09.94

**Sud-Ouest**

353 bd du Président Wilson  
33-Bordeaux  
tél. 44.62.15

**Est**

25 avenue Robert-Schuman  
57-Metz  
tél. 68.49.21

9 rue des Francs-Bourgeois  
67-Strasbourg

**Nord**

13 boulevard de la Liberté  
59-Lille