

CONTRIBUTION A LA RECHERCHE AUTOMATIQUE
DES TRACES ET DES EVENEMENTS INSCRITS DANS L'EMULSION IONOGRAPHIQUE

par

JEAN-PIERRE MATTE

Thèse présentée en vue de l'obtention
du grade de Maître ès Sciences Physiques

Département de Physique,
Faculté des Sciences et de Génie,
Université d'Ottawa
Ottawa, Canada

1972

SOMMAIRE

Un algorithme a été conçu pour analyser une image digitisée de 76 x 96 unités (120 x 150 microns environ). L'algorithme a d'abord été développé en utilisant le langage FORTRAN (IBM 360/65) et, comme données, des images d'événements cosmiques inscrits dans l'émulsion ionographique, codées à la main au moyen de l'écran digitisé d'un microscope à faisceau balayeur ont été utilisées.

L'image, sous forme de points noirs sur la grille de 76 x 96 est résumée en une liste de traces. L'analyse procède par la recherche de courts bâtonnets qui sont réunis pour former les traces.

Le programme a été traduit et optimisé en langage d'assemblée pour un mini-ordinateur PDP 8/E (4,096 mots de mémoire de 12 bits).

Une interface a été construite pour transmettre les images codées du microscope à faisceau balayeur au mini-ordinateur.

L'utilisation de cet appareil et de cet algorithme pour l'exploration automatique de plaques d'émulsion ionographique est discutée.

REMERCIEMENTS

Mes remerciements vont d'abord au professeur Hébert qui m'a guidé et encouragé tout au long de ce projet.

Je remercie aussi:

M. Daniel Rollin pour l'aide active apportée au travail de programmation et pour l'idée du filtre;

M. François Roy pour sa collaboration à la programmation, pour les essais expérimentaux, et pour avoir mis au propre la majorité des organigrammes;

Mm. Paul St-Arnaud* et William Davidson qui ont conçu et construit l'interface selon nos spécifications;

M. Bruce Wiggins qui a construit le microscope à faisceau balayeur, pour des échanges d'idées très profitables;

Mlle Marie Diorio, pour l'exploration d'un centimètre carré de la plaque d'émulsion avec un microscope optique;

Le Centre de Calcul de l'Université d'Ottawa et Mm. R. Weygand et T. O'Neil de la société Digital Equipment of Canada pour leur très utile collaboration;

M. A. Buser pour son travail technique;

Mm. P. Sima et Ben-Tchatchavadze pour les photographies;

Ma soeur Lorraine qui a dactylographié ce texte;

Et toutes les autres personnes qui ont collaboré à ce travail.

Enfin, je remercie le Ministère de l'Education de la province d'Ontario pour une bourse d'études supérieures.

* Décédé

TABLE DES MATIERES

Sommaire.....	
Remerciements.....	
Table des matières.....	
Liste des figures.....	
Liste des organigrammes.....	
Liste des tableaux.....	
Chapitre I. INTRODUCTION.....	1
Chapitre II. L'APPAREIL.....	8
A. Le microscope à faisceau balayeur.....	8
B. Le discriminateur et l'interface.....	12
C. Le mini-ordinateur PDP 8/E.....	17
Chapitre III. LE PROGRAMME - GENERALITES.....	19
A. L'image initiale.....	25
B. Liste des traces.....	29
C. Quelques définitions.....	33
D. Mise au point du programme FORTRAN.....	35
E. Mise au point du programme du PDP 8/E...	37
(1) Examen des éléments de la matrice TRACEF.....	38
(2) Examen séquentiel de points dans l'image.....	39
Chapitre IV. LE FILTRAGE.....	42
A. Elimination et mise en réserve des points isolés.....	42
B. Résultats du filtre.....	50
C. Autres méthodes de filtrage.....	55

Chapitre V.	RECHERCHE DES TRACES.....	56
A.	Orientations et interpolation.....	56
B.	Recherche des segments - Sous-routines CHSEG.....	68
C.	Recherche des traces	
Chapitre VI.	PROLONGEMENT ET ASSEMBLAGE DES TRACES...	95
A.	Prolongement des traces.....	95
B.	Assemblage des traces	108
Chapitre VII.	RECHERCHE DES ETOILES.....	129
Chapitre VIII.	RESULTATS EXPERIMENTAUX.....	138
Chapitre IX.	CONCLUSION.....	141
	Bibliographie.....	144

APPENDICES

Appendice A INTERPOLATION SELON "D"

1.	Porgramme FORTRAN.....	A-1
2.	Table d'interpolation.....	A-2
3.	Matrices "NECV" et "NECH".....	A-3
4.	Matrice "NEC".....	A-4
5.	Sous-routine "SUIVRE".....	A-5
6.	Sous-routine "MODE 1" et "MODE 2"....	A-6
7.	Sous-routine "AVANT" et "ARRIERE"....	A-7

Table des matières - (suite)

Appendice B - Programmes FORTRAN

1. Programmes de dessin.....	B-1
2. Filtre et remise dans l'image des points enlevés par le filtre.....	B-2
3. Recherche, prolongement et assemblage des traces..	B-3
4. Recherche des traces	B-4
5. Mode d'emploi	B-5

Table des matières - (suite)

Appendice C - Programmation du mini-ordinateur PDP 8/E

Généralités.....	C-1
1. Instructions de référence à la mémoire.....	C-2
2. Instructions d'entrée - sortie	C-5
3. Instructions d'opération.....	C-6
4. Quelques exemples de programmation.....	C-9

Table des matières - (suite)

Appendice D - Sous-programmes utilitaires

1. Sous-routine MINIMAX (minimum et maximum).....	D-1
2. Sous-routine MULTS (multiplication).....	D-2
3. Sous-routine DIVIS (division).....	D-3

Table des matières - (suite)

Appendice E - Transmission de l'image

1. Accès direct à la mémoire ("Data Break").....	E-1
2. Compteur de ligne	E-1
3. Lignes transmises.....	E-2
4. Action de l'interface.....	E-3
5. Programme pour le "Data Break" (organigramme).....	E-4
6. Étapes de transfert des données.....	E-5
7. Programme pour le "Data Break" (texte).....	E-9
8. Note - Comment éviter de sauter une ligne sur deux...	E-11

Table des matières - (suite)

Appendice F - Programmes en langage d'assemblage

1. Page 0.....	F-1
2. Dessin de l'image.....	F-2
3. Dessin des traces.....	F-3
4. Sous-routines de conversion.....	F-4
5. Filtrage.....	F-5
6. Sous-routine CHSEG.....	F-6
7. Recherche des traces.....	F-7
8. Sous-routines DEPH et DEPV.....	F-8
9. Sous-routine EVPROX.....	F-9
10. Prolongement des traces.....	F-10
11. Assemblage des traces.....	F-11
12. Recherche des étoiles.....	F-12
13. Mode d'emploi.....	F-13

LISTE DES FIGURES

Figure

1.	Schéma du système.....	6
2.	Vue d'ensemble du montage.....	7
3.	Système optique.....	7
4.	Evénement cosmique inscrit dans l'émul- sion ionographique, vu sur l'écran moni- teur.....	10
5.	Le même événement, image renversée.....	10
6.	Le même événement, image inversée et dis- criminée.....	11
7.	Effet du discriminateur et de l'inter- face.....	14
8.	L'événement cosmique (fig. 4, 5, 6) tel que transmis au PDP 8/E.....	15
9.	1 ^{ère} étoile digitisée manuellement.....	21
10.	1 ^{ère} étoile dans la mémoire du PDP 8/E....	22
11.	2 ^e étoile digitisée manuellement.....	23
12.	3 ^e étoile digitisée manuellement.....	24
13.	Evénement cosmique, après filtrage.....	51
14.	Etoile no. 1 après filtrage.....	52
15.	Etoile no. 2 après filtrage.....	53
16.	Etoile no. 3 après filtrage.....	54
17.	Les 41 orientations "D".....	57
18.	Etoile no. 1 après l'étape de recherche des traces. (CRITER = 6 et 5).....	94
19.	Etoile no. 1 après l'étape de prolonge- ment des traces. (CRITER = 6 et 5 à l'étape 3).....	107
20.	Etoile no. 1 à la fin de l'analyse (CRITER = 6 et 5 à l'étape 3).....	120

21.	Etoile no. 1 (CRITER = 5 et 4 à l'é- tape 3).....	121
22.	Etoile no. 2 (CRITER = 6 et 5 à l'é- tape 3).....	122
23.	Etoile no. 2 (CRITER = 5 et 4 à l'é- tape 3).....	123
24.	Etoile no. 3 (CRITER = 6 et 5 à l'é- tape 3).....	124
25.	Etoile no. 3 (CRITER = 5 et 4 à l'é- tape 3).....	125
26.	Événement cosmique (analysé <u>sans</u> filtrage, avec CRITER = 7 et <u>6</u> à l'étape 3).....	126
27.	Événement cosmique (analysé <u>sans</u> filtrage, avec CRITER = 5 et <u>4</u> à l'étape 3).....	127
28.	Événement cosmique (analysé <u>sans</u> filtrage, avec CRITER = 6 et <u>5</u> à l'étape 3).....	127
29.	Événement cosmique (analysé <u>avec</u> filtrage, avec CRITER = 5 et <u>4</u> à l'étape 3).....	128
30.	Événement cosmique (analysé <u>avec</u> filtrage, avec CRITER = 6 et <u>5</u> à l'étape 3).....	128

LISTE DES ORGANIGRAMMES

Organigramme

no.

1.	Organigramme général.....	20
2.	Filtrage.....	45
3.	Filtrage - Version "ASSEMBLEUR".....	46
4.	Restauration des points enlevés.....	49
5.	S.R. SUIVRE.....	62
6.	SOUS-ROUTINE DMOYEN.....	66
7.	SOUS-ROUTINE DMOYEN--VERSION "ASSEMBLEUR".....	67
8.	SOUS-ROUTINE CHSEG.....	70
9.	Sous-routine CHSEG--Version "ASSEMBLEUR".....	72
10.	Structure du programme de recherche des traces.....	83
11.	3-RECHERCHE DES TRACES (ANCIENNE VERSION).....	84
12.	3-RECHERCHE DES TRACES.....	85
13.	Recherche des traces - Version ASSEMBLEUR.....	86
14.	SOUS-ROUTINE EVPROX.....	89
15.	SOUS-ROUTINE DEPV.....	90
16.	SOUS-ROUTINE DEPH.....	92
17.	SOUS-ROUTINES DERNPT et EFF.....	93
18.	PROLONGEMENT DES TRACES.....	98
19.	Prolongement des traces - Version "ASSEMBLEUR".....-.....	101
20.	ASSEMBLAGE DES TRACES.....	110
21.	Assemblage des traces - Version "ASSEMBLEUR".....	114
22.	RECHERCHE DES ETOILES.....	131
23.	Recherche des étoiles - "ASSEMBLEUR".....	134

ORGANIGRAMMES DES APPENDICES

Sous-routine MINIMAX (minimum et maximum).....	D-1/1
Sous-routine MULTS (multiplication)	D-2/1
Sous-routine DIVIS (division).....	D-3/1
Action de l'interface	E-3
Programme pour le "Data Break"	E-4

xv

LISTE DES TABLEAUX

Tableau

1. Correspondance position dans l'image -
position dans la mémoire..... 27
2. Information sur les traces dans le pro-
gramme FORTRAN..... 30
3. Information sur les traces dans la mé-
moire du PDP 8/E..... 32

I- INTRODUCTION

Depuis la mise au point de l'appareil HPD ("Hough Powell Digitiser") en 1960⁽¹⁾, l'automatisation des mesures, puis du dépouillement, des clichés de chambre à bulles et à étincelles a été l'objet d'un effort de recherche très intense.⁽²⁻⁸⁾ Cet appareil à faisceau balayeur* s'est répandu dans de nombreux laboratoires.

Le HPD fut d'abord utilisé au CERN, et l'est encore dans plusieurs laboratoires, pour mesurer avec précision les événements préalablement choisis par un opérateur, et que reconstituent, géométriquement et cinématiquement, les célèbres programmes "REAP", "THRESH" et "GRIND".⁽²⁾

Depuis, R.B. Marr, G. Rabinowitz et al. ont développé des algorithmes capables, (à partir des positions des bulles sur le film, transmises par le HPD) de reconnaître les événements puis de les mesurer. Ainsi, à l'Université Columbia on analyse les clichés de chambre à bulles de façon presque entièrement automatique, la présence d'un opérateur n'étant requise que dans les cas particulièrement difficiles. Toutefois, ce système exige des moyens de calcul très importants: il accapare le quart du temps de l'ordinateur IBM 360/91 du centre de calcul de l'Université Columbia.⁽⁹⁾

*Le spot lumineux qui balaie le film est produit par un système opto-mécanique de fentes rotatives. C'est pourquoi le HPD est plus lent que les systèmes avec tube à rayons cathodiques qui lui ont succédé.

D'autres systèmes ont été conçus afin de réduire le fardeau imposé à l'ordinateur. Une façon est d'avoir un tube à rayons cathodiques dont la direction de balayage est contrôlable par la calculatrice comme, par exemple, le système POLLY. Allison et al. du Laboratoire Argonne ont réalisé le système "POLLY II", très automatique, ne nécessitant qu'un ordinateur SIGMA 7 (48K). (10)

Le "COCCINELLE" de B. Equer et al., au Collège de France, confie une bonne partie du travail de reconnaissance de forme à des circuits spécialement conçus à cet effet, ce qui réduit d'autant le travail imposé à l'ordinateur (CII 9010). (11)

Pour le film de chambre à étincelles, Goldwasser et al. du CEN Saclay ont réalisé l'appareil "ARIANE" auquel un mini-ordinateur CDC 8090 (8K) suffit. (12)

Pour l'émulsion ionographique, l'effort de mécanisation est plus ancien, mais n'a jamais mobilisé autant de ressources humaines et surtout matérielles. (13)

On conçût d'abord des systèmes pour faciliter les mesures aux opérateurs humains. Des appareils furent construits pour enregistrer (sur cartes ou ruban perforé) les coordonnées de pointés faits par un observateur humain pour mesurer la diffusion multiple. (14) Des systèmes à fente et photomultiplicateur ont été créés pour mesurer la largeur d'une trace, ou effectuer le comptage des grains ou des lacunes. Mais, c'était l'opérateur qui devait centrer la trace sous la fente. (15)

Puis, on réalisa des appareils capables, une fois alignés sur une trace, de la suivre automatiquement (grâce à un microscope avec téléviseur et un système de direction à asservissement). (16)

Ceci permet, non seulement d'effectuer automatiquement les mesures mentionnées ci-dessus, mais aussi d'effectuer le dépouillement, puisque la fin d'une trace est souvent le lieu d'une réaction nucléaire.

Certains laboratoires de spectroscopie nucléaire utilisent des plaques d'émulsion ionographique comme détecteur. Comme les traces sont toutes parallèles et de même largeur, leur comptage se prête particulièrement bien à l'automatisation. Hough à l'Université du Michigan, puis Erskine et Vonderohe à Argonne ont réalisé des appareils comportant un système de télévision en circuit fermé monté sur un microscope. Le premier utilise des compteurs tandis que le second est couplé à un mini-ordinateur PDP-9 (8K) pour compter les traces. (17)

En 1970, Brettel et al. du Max Planck Institut für Physik und Astrophysik de Munich ont rapporté la réalisation d'un système avec caméra Orthikon monté sur un microscope, et téléviseur. L'appareil sera couplé à l'ordinateur IBM 360/91 de l'institut et pourra faire le dépouillement, suivre les traces etc. une fois terminée la liaison avec l'ordinateur.* (18)

Le microscope à faisceau balayeur a été conçu en 1951 par Young et Roberts pour des applications biologiques. (19)

En 1959, McEwen et Hébert en proposaient l'utilisation pour la mesure et le dépouillement des événements inscrits dans l'émulsion nucléaire. (20) Ils construisirent un premier exemplaire au Laboratoire de Physique des Particules Élémentaires

*Malgré des demandes répétées, il a été impossible d'obtenir des renseignements détaillés sur ce système.

de l'Université d'Ottawa en 1960. A ce même laboratoire, Wiggins mit au point une version améliorée, avec résolution de 1 micron et écran digitisé, en 1966. (21) Peu après, des modifications supplémentaires donnaient à l'appareil sa configuration actuelle. En plaçant le tube à rayons cathodiques horizontalement et en utilisant un miroir plan incliné, il devenait possible de changer sa position (et partant, le grossissement). De plus, la direction du balayage du tube à rayons cathodiques et l'orientation de la platine ont été rendues modifiables à volonté.

Dans les expériences de Physique des Hautes Energies effectuées avec des émulsions ionographiques, l'un des principales difficultés est l'étape de l'exploration, ou recherche des événements.

Afin d'automatiser cette recherche, il est essentiel de pouvoir reconnaître les événements automatiquement après qu'ils ont été transmis au mini-ordinateur sous forme d'image digitisée en noir et blanc. L'algorithme fut d'abord développé en FORTRAN (pour être exécuté sur l'ordinateur IBM 360/65 du centre de calcul avec des images codées à la main) parce qu'il est plus facile de travailler dans ce langage. Néanmoins, le programme FORTRAN ne fut utilisé que dans le but de développer le programme qui servirait, finalement, dans le mini-ordinateur. C'est pourquoi il fallait toujours garder à l'esprit les possibilités et contraintes du mini-ordinateur: manipulation de bits individuels, mais, par contre, nécessité d'éviter tout calcul en virgule flottante.

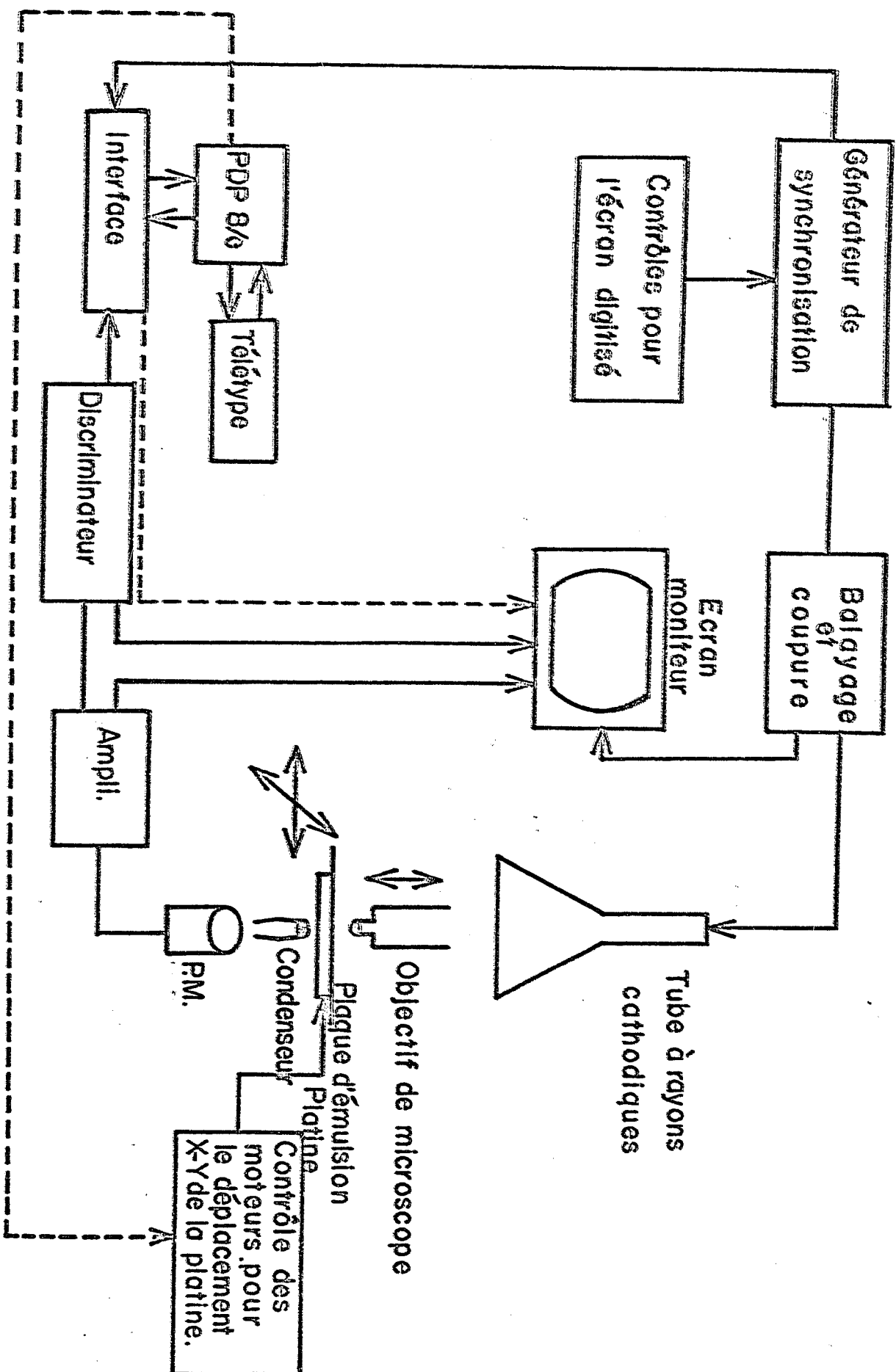
L'algorithme fut, par la suite traduit en langage d'assemblée pour être exécuté par le mini-ordinateur PDP 8/E. Pendant ce temps, deux étudiants en génie électrique MM. Paul St-Arnaud[†] et Bill Davidson ont conçu et construit l'interface.

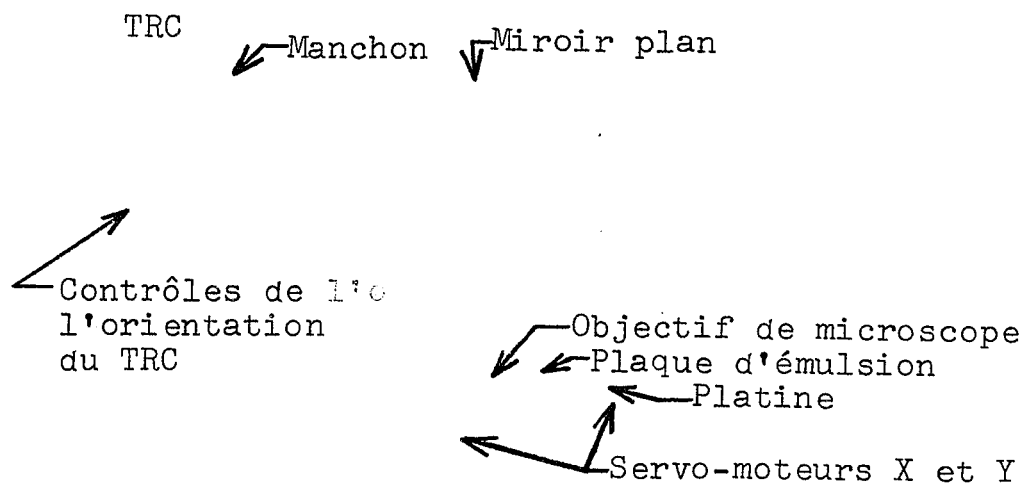
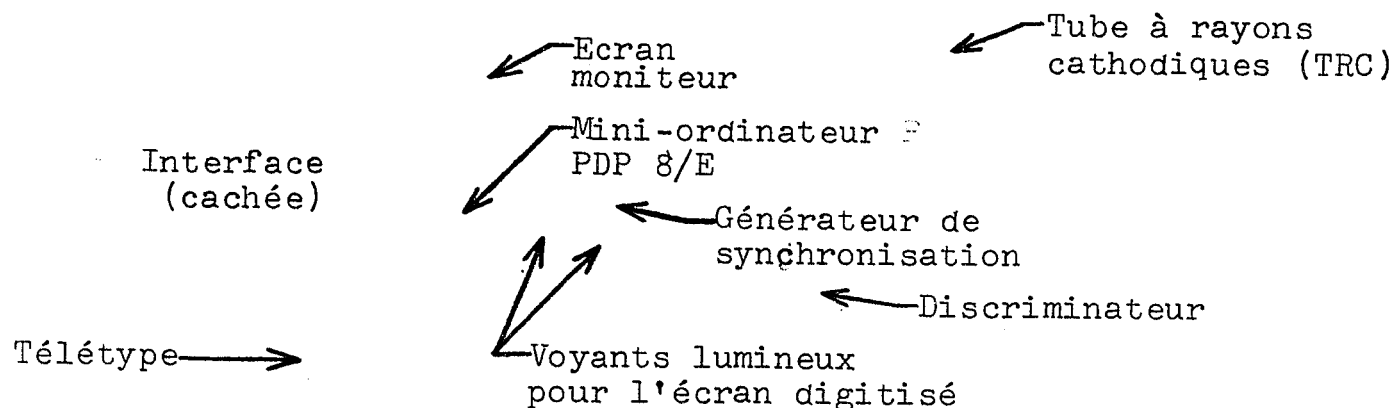
Maintenant, l'appareil transmet les images au mini-ordinateur qui les analyse immédiatement pour décider s'il y a évènement ou pas.

Pour être utile à l'exploration des plaques, il fallait que le temps d'exécution de l'algorithme soit très court. C'est pourquoi des efforts particuliers ont été consentis afin de minimiser le temps nécessaire aux calculs.

[†] Décédé.

SCHEMA DU SYSTEME





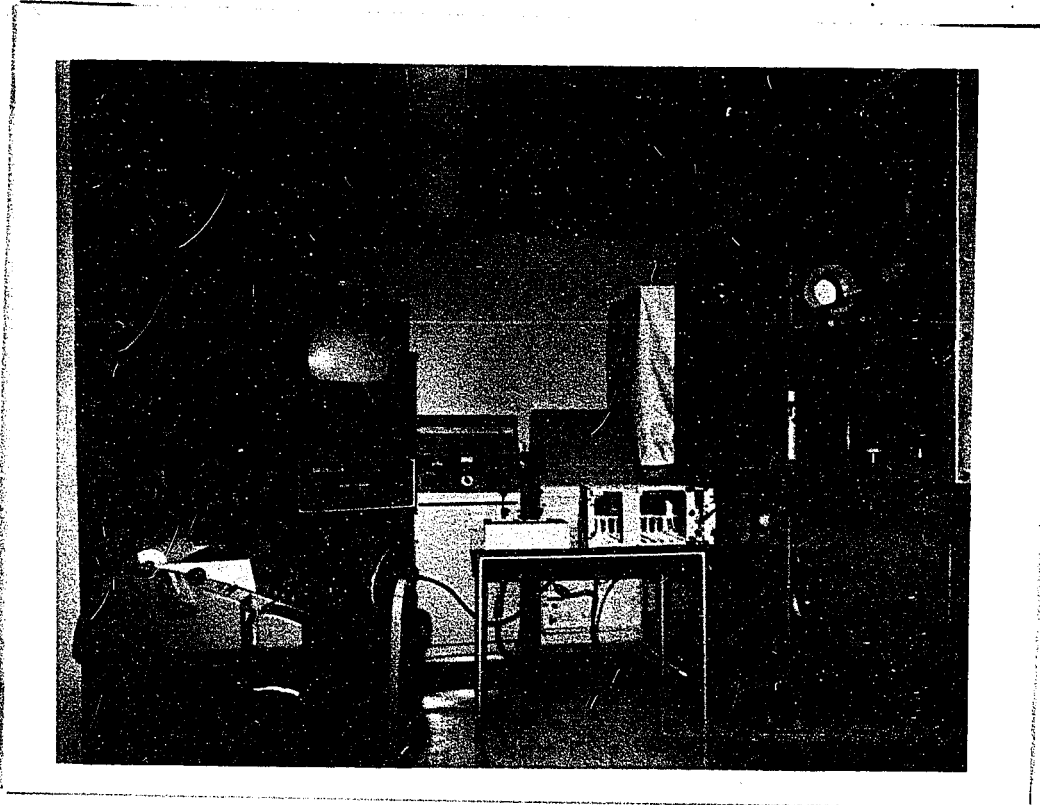


Figure 2-Vue d'ensemble du montage

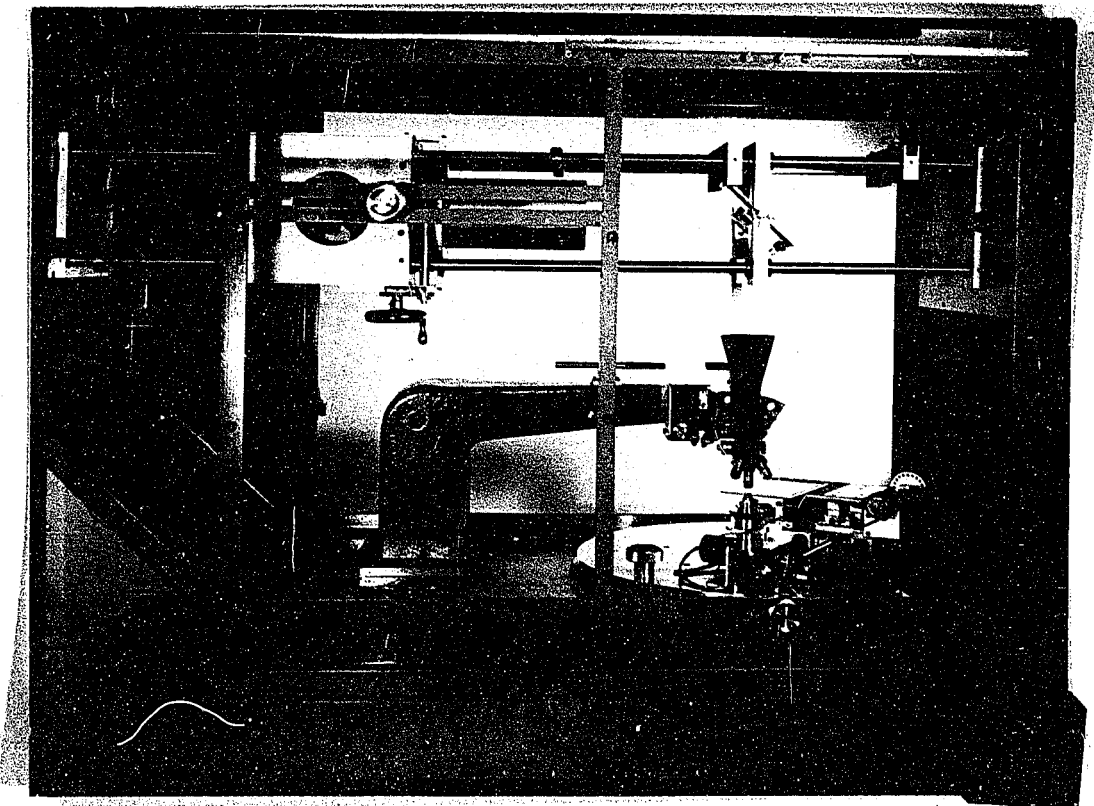


Figure 3-Système optique

II. L'appareil.

La figure 1 est un schéma des différentes composantes de l'appareil et des relations entre elles. Les traits pointillés indiquent des liens à venir: contrôle du mouvement de la platine et de la mise au point du microscope par le mini-ordinateur, et transmission de l'image du mini-ordinateur à l'écran, afin de faciliter la vérification du programme.

A) Le microscope à faisceau balayeur.

Le fonctionnement du microscope à faisceau balayeur a été décrit en détail par d'autres auteurs (19-21). Nous le rappellerons brièvement ici, tout en traitant plus longuement de la synchronisation du balayage à cause de son importance pour la transmission de l'image.

Un tube à rayons cathodiques produit un spot lumineux de quelques 100 microns qui balaie une surface d'environ 6 x 6 cm.* Le spot et son mouvement de va et vient sont démagnifiés par l'objectif du microscope de sorte que c'est un spot d'une fraction de micron qui balaie une surface de 200 x 200 microns sur l'objet, en l'occurrence la plaque d'émulsion. Un tube photomultiplicateur (PM) recueille la lumière transmise et chaque fois que le spot rencontre un obstacle - un grain opaque d'argent développé - le courant du PM baisse. Ce courant (amplifié)

*Le grossissement dépend de plusieurs choses: amplitude (ajustable) du balayage sur le tube à rayons cathodiques, distance entre celui-ci et le miroir objectif utilisé. Les chiffres cités ici s'appliquent aux images vues sur les figures 4, 5 et 6.

module l'intensité du faisceau du moniteur qui balaie en synchronisme avec le premier. On voit ainsi sur l'écran une image agrandie d'environ 1,500 fois de l'objet.

La figure 4 montre un évènement dans l'émulsion nucléaire tel qu'il apparaît sur l'écran du moniteur. Pour diminuer le bruit de fond on inverse souvent le signal de façon à avoir un négatif sur l'écran (fig. 5).

La synchronisation du balayage est celle des téléviseurs ordinaires. Le taux de répétition des images est de 30 par seconde.

Chaque image compte 525 lignes de 63.5 microsecondes chacune: 52.5 μ sec. pour le balayage actif de gauche à droite et 11 μ sec. pour la retrace de droite à gauche pendant laquelle le faisceau est coupé.

L'image qui dure $1/30^e$ seconde comporte deux demi-cadres entrelacés: l'impair (lignes 1, 3, 5, ..., 525) et le pair (lignes 2, 4, 6, ..., 524). Ces deux demi-cadres comptent $262\frac{1}{2}$ lignes chacun dont $250\frac{1}{2}$ sont actives et 12 sont coupées pour permettre la retrace (de bas en haut).

Le générateur du synchronisation a été modifié afin de permettre de noircir une des 500 lignes de l'écran, et de noircir une position sur chacune des lignes. On peut voir sur l'écran deux traits noirs, horizontal et vertical, dont les coordonnées (visibles sur des voyants lumineux) permettent de mesurer des positions directement sur l'écran.

Une unité de l'écran digitisé correspond à environ 0.4 micron dans l'émulsion.

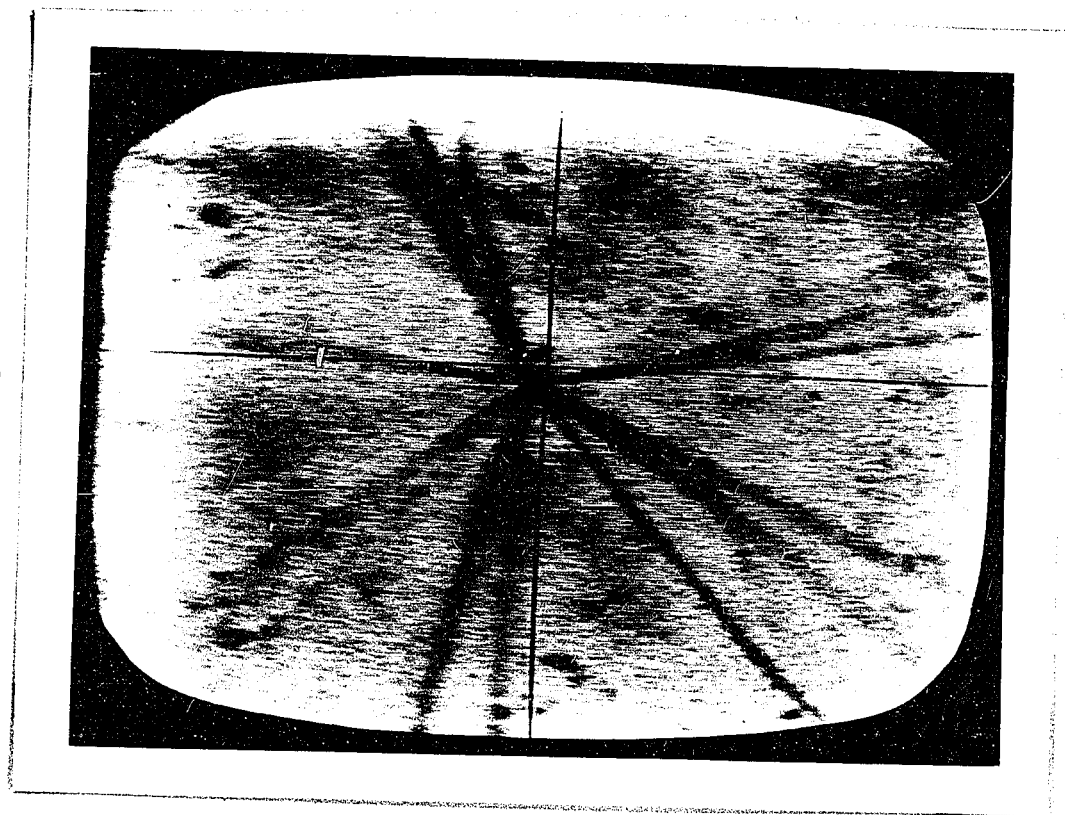


Figure 4-Evénement cosmique inscrit dans l'émulsion ionographique, vu sur l'écran moniteur



Figure 5-Le même événement, image renversée

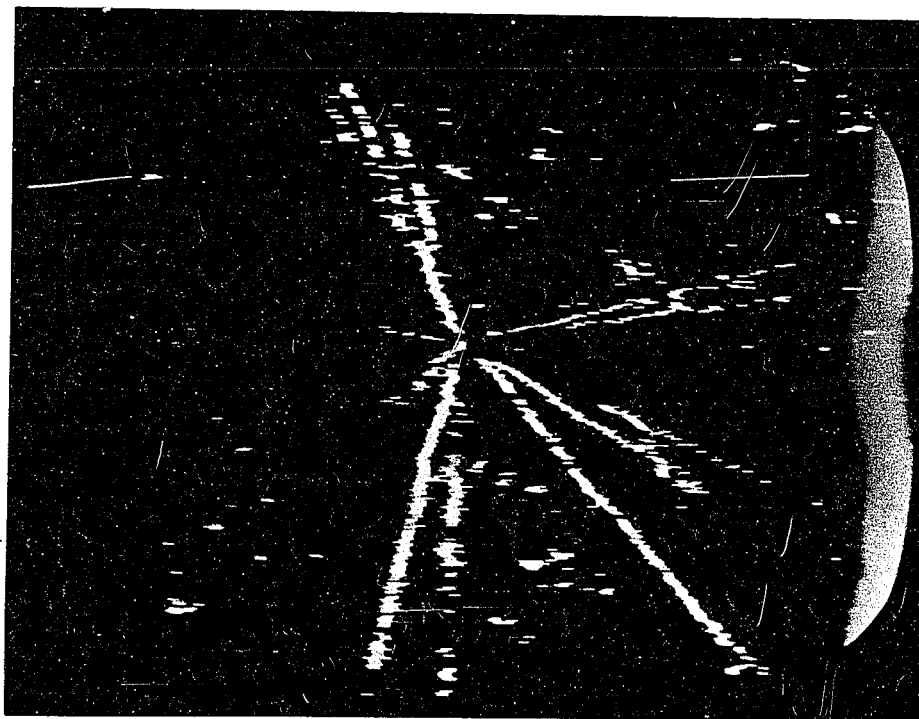


Figure 6-Le même événement,
image inversée et discriminée

B) Le discriminateur et l'interface.

Le microscope à faisceau balayeur transforme l'information optique contenue dans les grains d'argent de l'émulsion en un signal électrique (à la sortie de l'amplificateur qui suit le photomultiplicateur). Comme le mini-ordinateur n'accepte que les données binaires, il faut discriminer ce signal pour éliminer toute nuance de gris et ne laisser que blanc et noir.*

Pour ce faire, la sortie de l'amplificateur est branchée à l'entrée d'un discriminateur** qui donne une impulsion d'une durée de 500 nanosecondes*** chaque fois que le seuil est franchi. En utilisant la sortie du discriminateur comme entrée du téléviseur, on peut voir son effet sur l'image. (Comparer les figures 5 et 6).

L'information image est transmise pour la région comprise entre les lignes 122 et 422 et entre les positions horizontales 60 et 444 (ce qui correspond à 38 μ secondes de balayage sur les 52.5 que dure chaque ligne). Le périmètre de la surface balayée par le spot est évité parce que c'est là que les distortions sont le plus importantes.

* Certains appareils de reconnaissance de forme distinguent plusieurs nuances de gris, mais, en physique des hautes énergies, tous (HPD etc.) les appareils s'en tiennent au noir et blanc sans nuances. La nature même des images est telle qu'il est préférable de s'en tenir à deux tons.

** Ce fut d'abord un ORTEC 841 ("Integral Discriminator"). Par la suite, un discriminateur plus simple et plus conforme à nos besoins fut construit. Le seuil et la durée de l'impulsion de sortie sont ajustables.

*** Ceci correspond à une distance de deux microns dans l'émulsion.

Une horloge de 2.5 Mhz contrôle la transmission des données sur chaque ligne: à toutes les 0.4 microsecondes, un bit 1 ou 0 est envoyé en mémoire selon que la sortie du discriminateur est 1 ou 0. De cette façon, 96 bits (8 mots de 12 bits) sont transmis au PDP 8/E pour chaque ligne.

Une image n'est transmise qu'à la demande du mini-ordinateur. Si la demande a lieu au moment où un demi-cadre pair est prêt, les lignes 122, 124, 126, ..., 420, 422 sont transmises, mais les lignes 124, 128, 132, ..., 416, 420 sont supprimées pendant le transfert de sorte que, celui-ci terminé, les 76 lignes 122, 126, 130, ..., 422 sont en mémoire. De même, si la transmission a lieu pendant un demi-cadre impair, les lignes 123, 127, 131, ..., 423 sont en mémoire à la fin du transfert.

Dans les deux cas, l'ordinateur a une image de 76 rangées par 96 colonnes en mémoire.

Le transfert de l'image est expliqué de façon plus détaillée à l'appendice E .

La figure 7 montre l'effet de cette réduction de l'information pour trois cas.

- a) une trace moyennement inclinée: bien transmise car ceux des points qui en sont transmis permettront de la reconstituer. (à gauche).
- b) une tache: effet heureux puisqu'il n'en reste qu'une mince lisière. (au centre).
- c) une trace presque parallèle au sens du balayage: effet moins heureux puisque seuls des points éloignés les uns

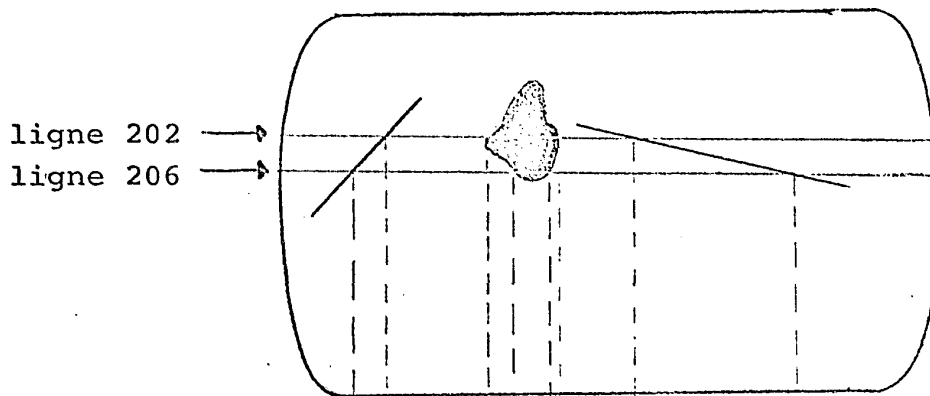
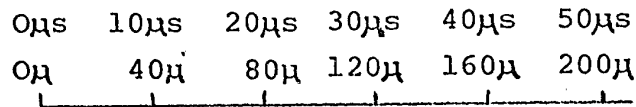


Image non discriminée mais inversée (traces et taches blanches sur fond noir).

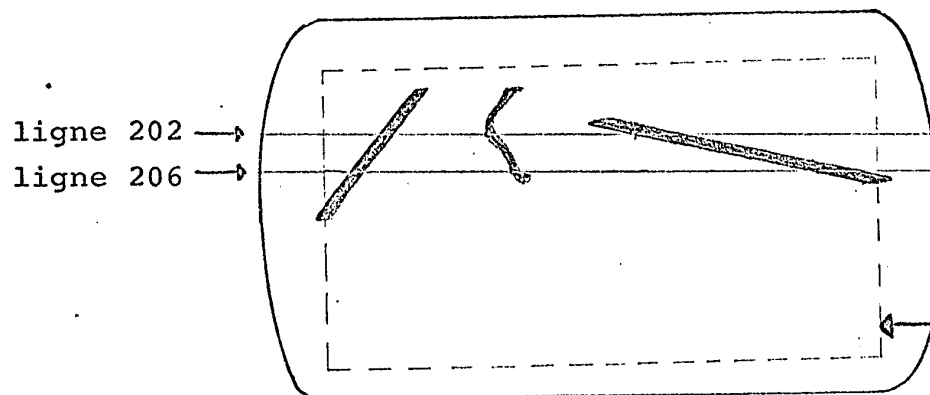
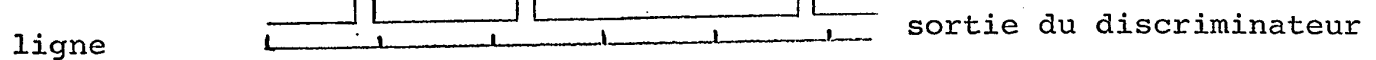
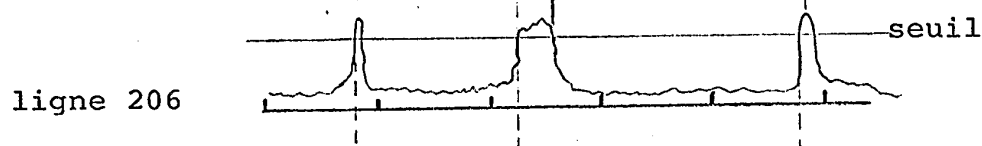
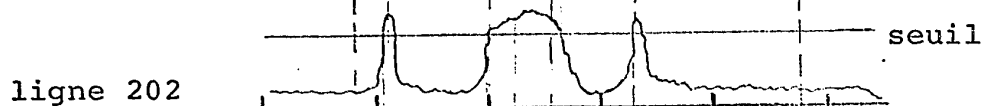


Image discriminée (lignes blanches sur fond noir).



Dans la mémoire du PDP 8/E.

Fig. 7 - Effet du discriminateur et de l'interface.

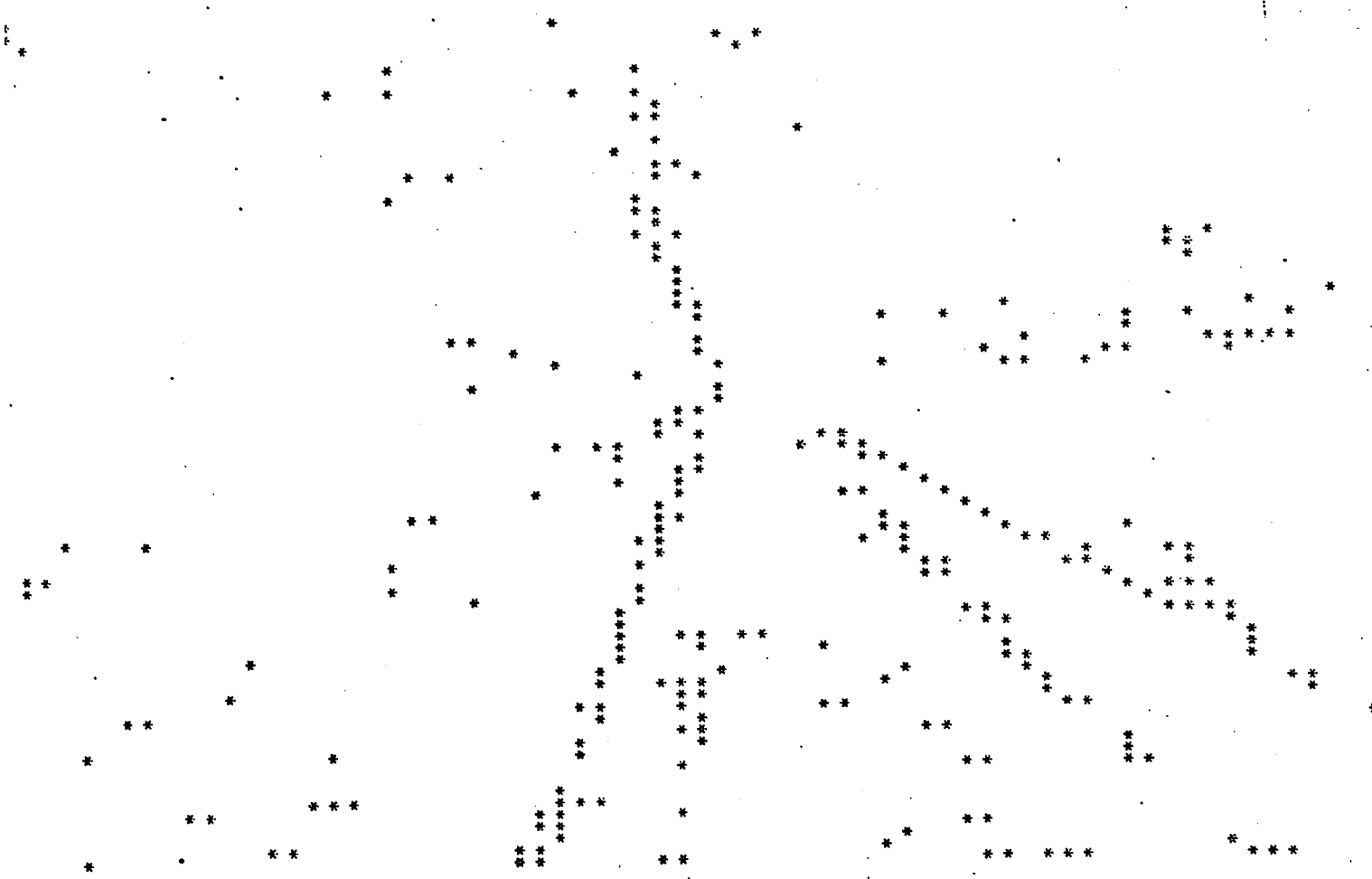


Figure 8 - L'événement cosmique (fig. 4, 5, 6)
tel que transmis au PDP 8/E.

des autres en sont transmis, de sorte que la trace ne pourra pas être reconstituée par le programme (à droite). On estime que 10% à 20% des traces sont perdues de cette façon.

Remarquons que la ligne 202, citée comme exemple dans la figure 7 est la 21^e des 76 lignes transmises et gardées par le PDP 8/E (V=21) tandis que la ligne 206 est la 22^e (V=22).*

*Pour un demi-cadre pair.

C) Le mini-ordinateur PDP 8/E

Le PDP 8/E est un mini-ordinateur à orientation générale avec une mémoire de 4,096* mots de 12 bits** et un cycle de base de 1.2 μ sec. L'addition s'effectue en 2.6 μ sec. et la soustraction en 3.8 μ sec. La multiplication et la division sont faites par des sous-routines qui prennent quelque 100 μ sec. (appendice C-2). Les calculs en virgule flottante ont été évités parce qu'ils exigent des sous-routines complexes.

L'unité centrale de traitement comporte deux registres principaux:

- a) l'accumulateur (AC) et son bit de surplus ("LINK" ou LK) où sont effectuées toutes les opérations arithmétiques et logiques etc.
- b) le multiplicateur-quotient (MQ) qui est utilisé avec AC dans les opérations de multiplication et division, et qui sert aussi de registre à accès très rapide.

Le déroulement du programme est normalement séquentiel, mais cet ordre peut être changé par des instructions de saut analogues au "IF" et au "GO TO" du FORTRAN.

Le programme peut aussi modifier ses propres instructions pendant l'exécution, ce qui permet beaucoup de flexibilité. (Il est impossible de faire cela en FORTRAN).

L'appendice C-1 donne davantage de précisions sur le jeu d'instructions du PDP 8/E.

* Avec possibilité d'expansion jusqu'à 32,768 mots.
 ** Numérotés, de gauche à droite, 0, 1, 2, ..., 10, 11.

L'entrée-sortie s'effectue normalement par le teletype qui comporte une lectrice et une perforatrice de ruban en plus du clavier et de l'imprimante. On peut aussi changer des mots en mémoire et contrôler l'exécution du programme avec la console du PDP 8/E.

III. Le programme - généralités.

L'appareil a pour but de détecter les événements dans l'émulsion. L'organigramme général (figure 7) résume comment cela se fait.

La plaque d'émulsion se déplace lentement. A toutes les deux secondes, environ, une image est transmise au mini-ordinateur (étape 1) sous forme d'une matrice logique (points noirs sur fond blanc). Le programme analyse cette image et la condense en une liste de traces qui contient les coordonnées du point initial et du point final de chacune. Dans cette analyse de l'image, c'est l'étape 3 qui est essentielle. Le filtrage la facilite tandis que les étapes 4 à 6 corrigent certains lacunes.

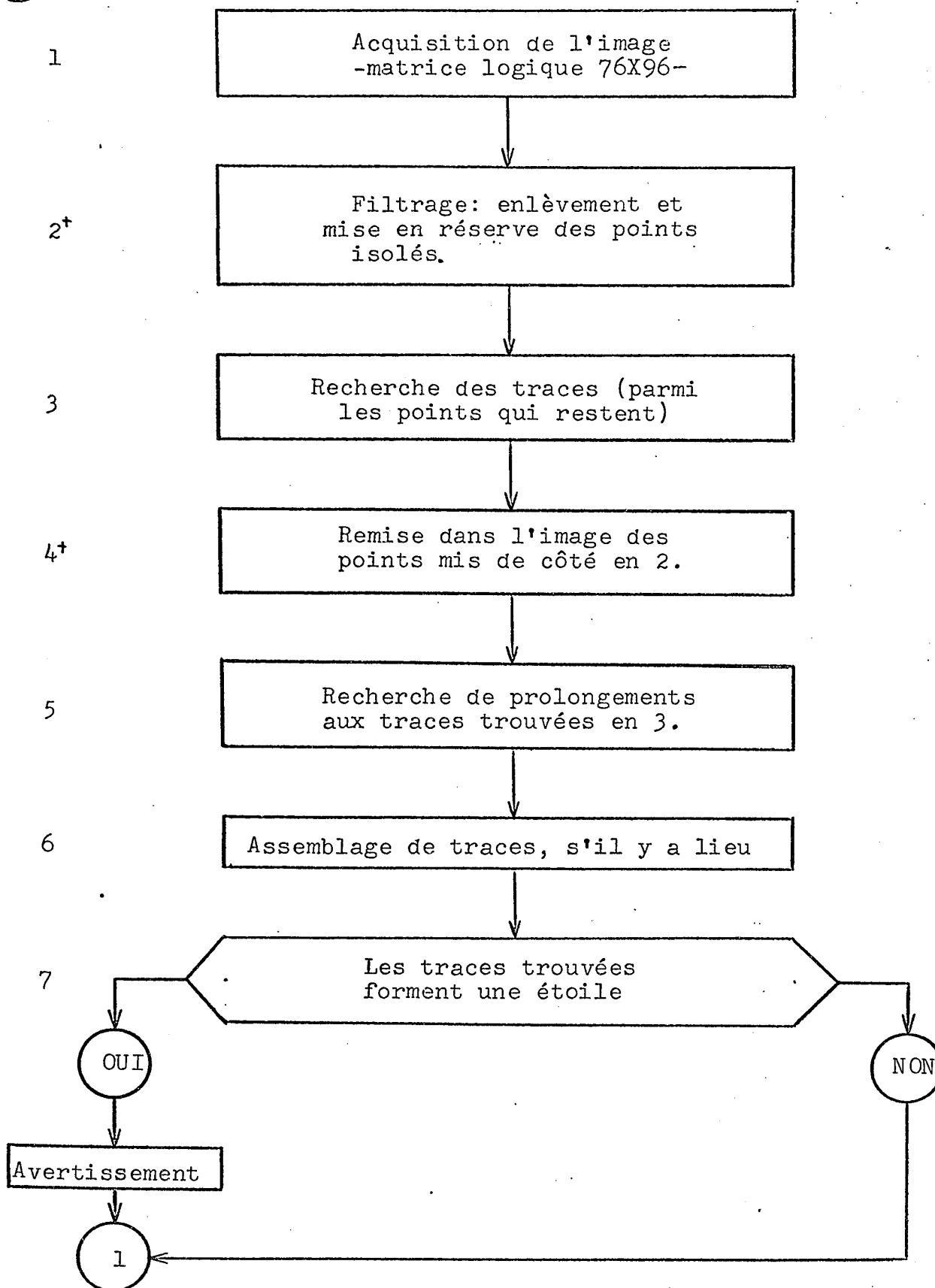
Enfin, une fois obtenue cette liste de traces, le programme décide s'il y a événement ou pas en cherchant si une paire d'entre elles forment soit un "V", soit un "T". (étape 7). Les critères de sélection sont ajustables. La cloche du télétype sonne quand un événement est trouvé.

Après cette analyse, une autre étoile est transmise au mini-ordinateur, et le cycle recommence.

Des programmes de dessin de l'image et des traces permettent de voir les résultats des diverses étapes.

ORGANIGRAMME GENERAL

1



⁺ Les parties 2 et 4 peuvent être omises.

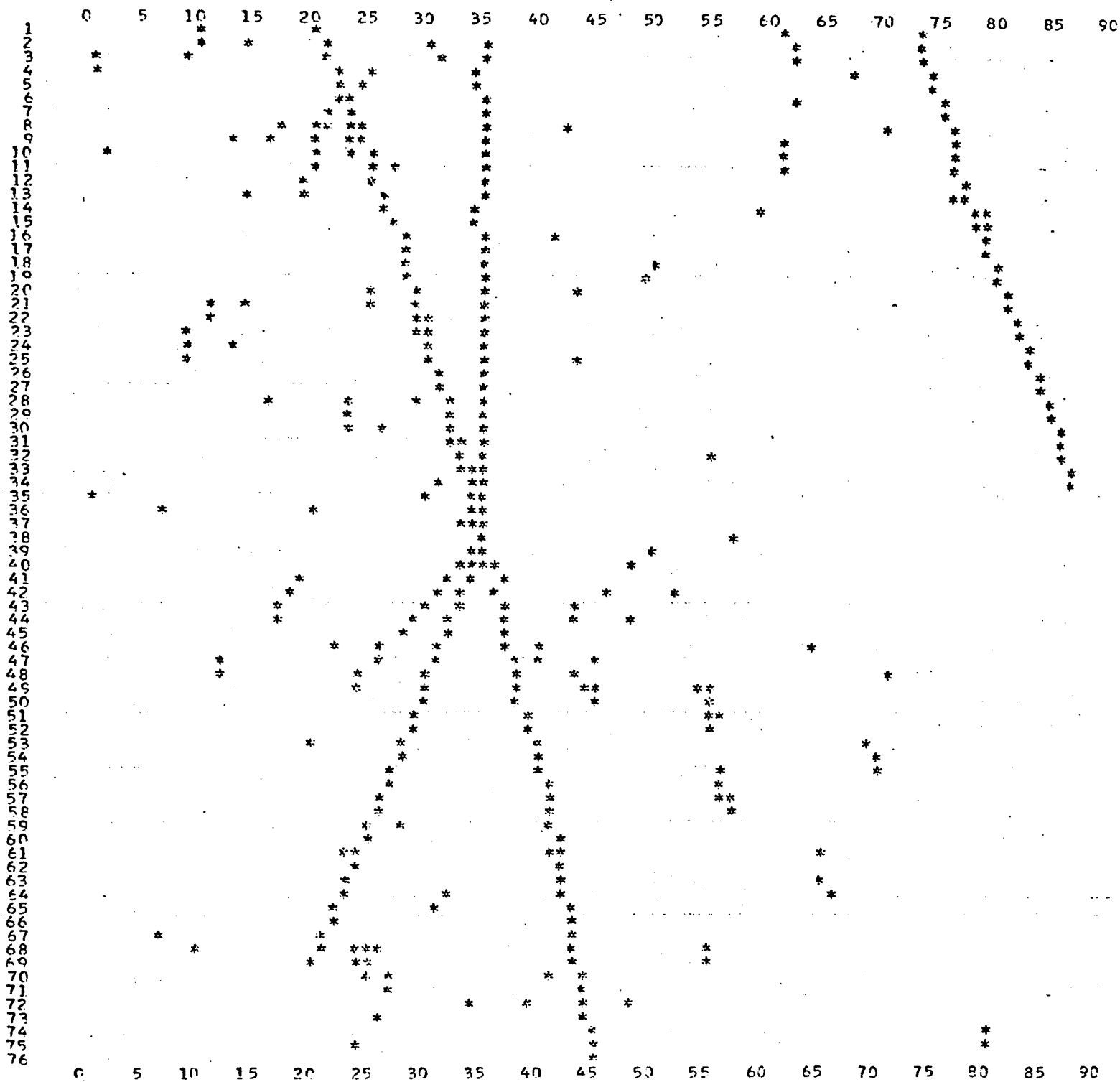


Figure 9 - 1^{ère} étoile digitisée manuellement.

ALL TIME FILTERAGE, NUMBER = 243

Figure 10 - 1^{ère} étoile dans la mémoire du PDP 8/E.

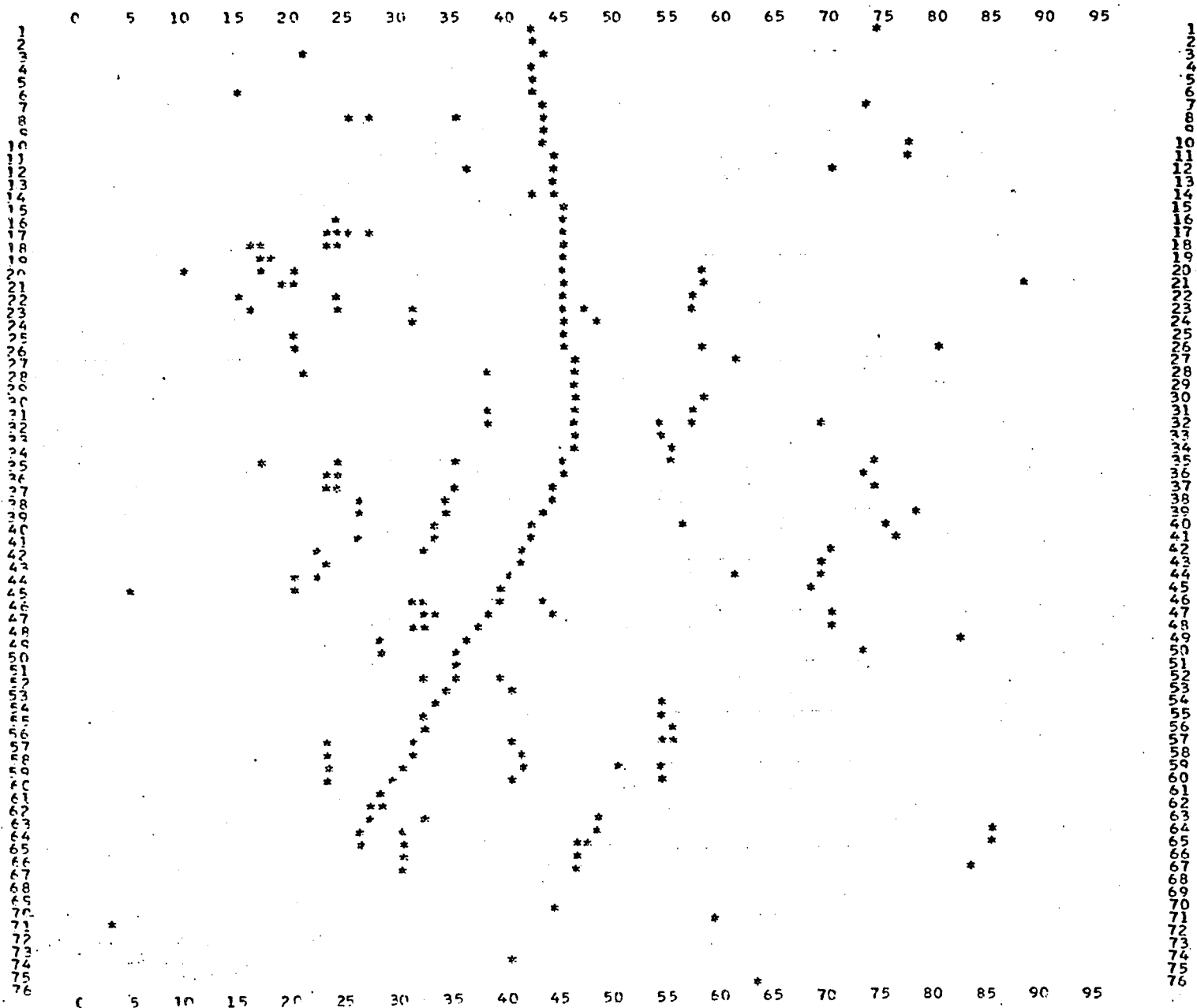


Figure 11 - 2^e étoile digitisée manuellement.

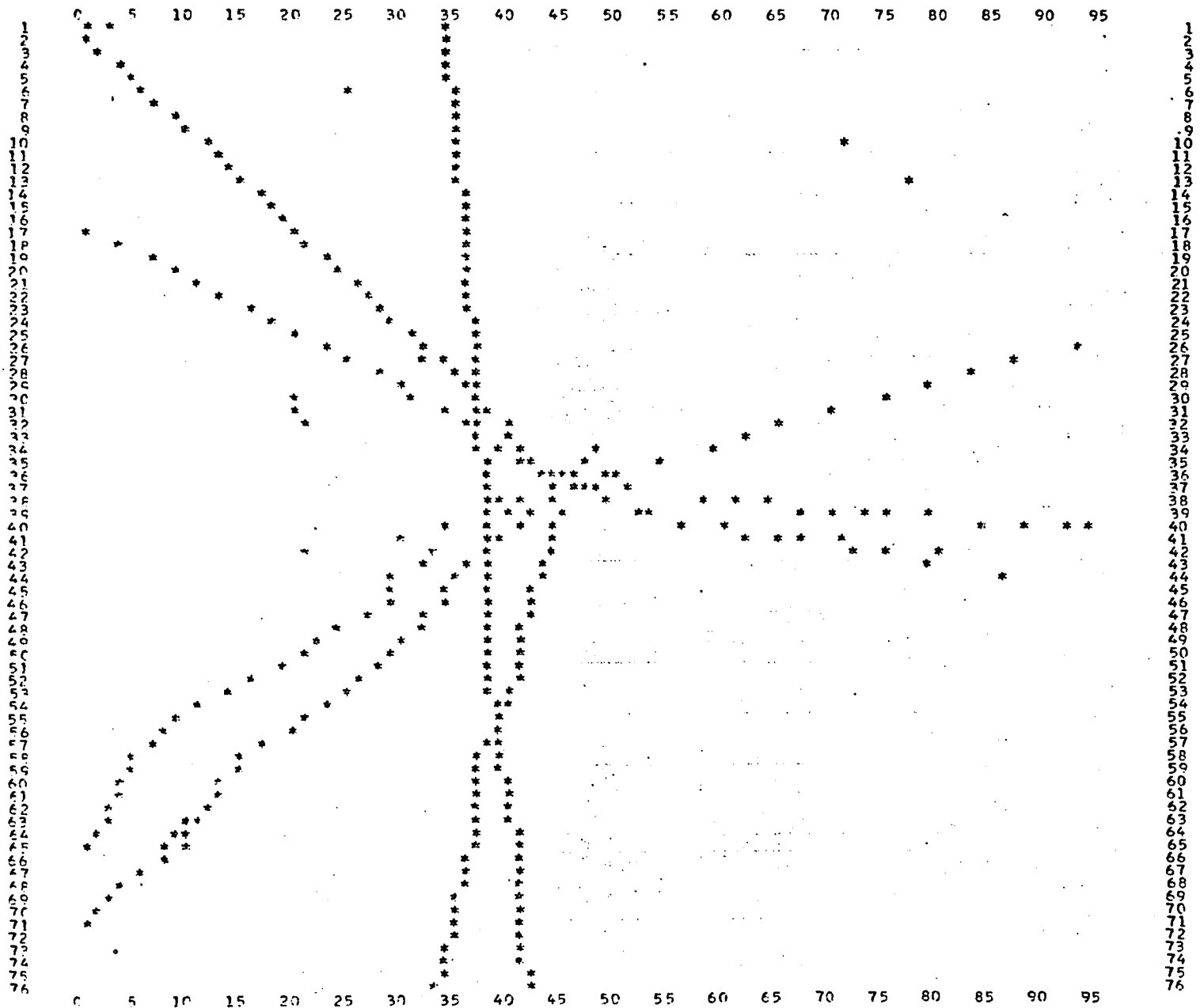


Figure 12 - 3^e étoile digitisée manuellement.

A) L'image initiale.

L'image est transmise à l'ordinateur sous forme d'une matrice logique "L" de 76 x 96. On utilise comme coordonnées les indices de la matrice "L".

L(V,H) est vrai si le point (V,H) est noir
(s'il y a un grain d'argent opaque)

L(V,H) est faux si le point (V,H) est blanc
(s'il n'y a pas de grain d'argent opaque).*

V, la coordonnée verticale, augmente de 1 en haut à 76 en bas; tandis que H, la coordonnée horizontale, augmente de 1, à gauche, à 96, à droite.

Pour mettre au point le programme FORTRAN, trois étoiles ont été codées en utilisant l'écran digitisé. La région de l'écran entre les lignes 150 et 450, entre les positions horizontales 100 et 483 a été retenue pour observation.**

Le codage s'effectue en examinant les 76 lignes 150, 154, 158, ..., 446, 450. Sur chacune, on note la coordonnée horizontale de chaque point noir*** entre 100 et 484. Une carte de données est tapée pour chaque ligne, avec le numéro de la ligne (U) et les coordonnées horizontales (W(J), J = 1,15)).

* L'image compte, en tout, $76 \times 96 = 7296$ points. Sur une image typique, environ 200 à 400 sont des points noirs. De ceux-ci, de 30% à 50% sont du bruit tandis que les autres 50% à 70% font partie des traces.

** Contre 122 à 422 et 60 à 444 respectivement pour l'image transmise au PDP 8/E.

*** L'image inversée a été utilisée. Donc, les grains opaques d'argent paraissaient blancs à l'écran et ce sont leurs positions qui ont été notées. Les deux premières étoiles ont été codées sans le discriminateur, tandis que l'image discriminée fut utilisée pour la troisième.

Cette liste de points noirs est transformée en matrice "L" par les énoncés ISN 0018 à 0035 dans les programmes de filtrage et de recherche des traces.

D'abord, la matrice L est vidée (ISN 0018-0020). Puis, les coordonnées sont lues et transformées en paires d'indices de rangée et de colonne (R,C) comme suit:

$$\begin{aligned} R &= (U-150)/4 + 1 \\ C &= (W(J)-100)/4 + 1 \end{aligned} \quad \begin{array}{l} \text{(Division entière,} \\ \text{reste tronqué).} \end{array}$$

de façon à avoir $1 \leq R \leq 76$ et $1 \leq C \leq 96$. Pour chaque paire d'indices ainsi obtenue, le programme met l'élément de matrice correspondant vrai ($L(R,C) = .TRUE.$), ce qui signifie que le point de coordonnées ($V = R, H = C$) est noir.

Une sous-routine, "DESL" (appendice B-1) a été écrite pour dessiner les images ainsi codées: un astérisque est imprimé pour chaque point noir, un blanc pour chaque point blanc. (fig. 9, 11, 12)

Dans le PDP 8/E, les 96 éléments de matrice de chaque ligne sont regroupés en 8 mots de 12 bits chacun. Par une variante du sous-programme de dessin, DESLP (appendice B-1), on peut obtenir une représentation de l'image telle que stockée dans la mémoire du mini-ordinateur. (fig. 10) Les bits 1 sont indiqués par des "1", les bits 0 par des ".". Pour faciliter la lecture, les mots et les groupes de 8 lignes (100_8 mots) sont séparés par des espaces blancs. Le cadrage indique la position de chacun de ces mots en mémoire.

Pour examiner un bit particulier, il faut utiliser la technique du masquage. Des masques, pour les bits 0 à 11 ont été mis dans les mots 1120 à 1133 et, dans l'ordre inverse, dans les mots 1135 à 1150. Chacune de ces deux suites de masques est suivie d'un mot vide (mots 1134 et 1151). (Cf. appendice A-5, sous-routine SUIVRE).

Le tableau 1 explique la correspondance entre position dans l'image et position dans la mémoire.

Position dans l'image		Position dans la mémoire			
V	H	MOT	BIT	MASQUES DU BIT (ADBIT)	
				1ère liste	2e liste
1	1	2000	0	1120	1150
1	2	2000	1	1121	1147
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
1	12	2000	11	1133	1135
1	13	2001	0	1120	1150
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
1	96	2007	11	1133	1135
2	1	2010	0	1120	1150
.	.	.	.	.	.
.	.	.	.	.	.
2	96	2017	11	1133	1135
.	.	.	.	.	.
.	.	.	.	.	.
76	1	3130	0	1120	1150
.	.	.	.	.	.
.	.	.	.	.	.
76	96	3137	11	1133	1135

Tableau 1 - Correspondance position dans l'image - position dans la mémoire.

Un programme de dessin de l'image a aussi été rédigé en langage d'assemblée pour dessiner les images dans la mémoire de la calculatrice sur le télétype. A cause du format du papier télétype, il faut tourner l'image de 90° en l'imprimant, ce qui complique un peu le programme. (appendice D-1). (Voir fig. 8).

Il faut souvent convertir d'un système de coordonnées à l'autre.

La sous-routine "CONV" convertit la position en mémoire, exprimée sous la forme d'une adresse (MOT) et de l'adresse du masque du bit (ADBIT, 1^{ère} liste: 1120 à 1133)⁺, en position dans l'image sous la forme (V,H).

L'algorithme utilisé est:

$$\begin{aligned} H &= (\text{MOT modulo } 8) + (\text{ADBIT} - 1120) + 1 \\ &= \text{MOT} \wedge 7 + \text{ADBIT} - 1117 \end{aligned}$$

$$V = 1 + (\text{MOT} - 2000_8) / 8 = (\text{MOT} - 1770_8) / 10_8$$

(quotient tronqué).

MOT modulo 8 s'obtient facilement en masquant les trois derniers bits de MOT qui contiennent la valeur désirée.

La division par 8 s'effectue facilement par trois décalages à droite. (en prenant soin de vider le bit de retenue à chaque fois!).

⁺A la sortie de certaines sous-routines, ADBIT peut être soit de la première ou de la deuxième liste. La sous-routine "CHSYAL" ("changer s'il y a lieu") change ADBIT en son équivalent de la première liste, par exemple pour le bit 11, 1135 est changé en 1133, mais 1133 est laissé tel quel.

La sous-routine "CONVINV" effectue la conversion inverse.

$$\text{MOT} = 2000_8 + (V-1)*8 + (H-1)/12_{10} \text{ (quotient tronqué)}$$

$$\text{ADBIT} = 1120_8 + (H-1) \text{ modulo } 12_{10}$$

$$\text{ou MOT} = 1770_8 + V*8 + (H-1)/14_8$$

$$\text{ADBIT} = 1134_8 + ((H-1) \text{ modulo } 14_8 - 14_8).$$

La multiplication par 8 est faite par un triple décalage à gauche.

La division par 12 (14_8) se fait par soustraction répétée jusqu'à ce qu'un nombre négatif soit obtenu. Ce nombre est justement $((H-1) \text{ modulo } 14_8 - 14_8)$.

Les listings de ces trois sous-programmes sont à l'appendice D-3.

B) Liste des traces

Le nombre de traces découvertes dans l'image est NTRF.

Les renseignements sur les traces sont gardées dans la matrice TRACEF qui constitue notre liste de traces. On garde 7 informations sur chaque trace dans le programme FORTRAN, contre 6 dans le programme du PDP 8/E. Le tableau 2 indique les éléments retenus pour la K^e trace ($1 \leq K \leq \text{NTRF}$).

Elément de matrice	Nom	Définitions et limites
TRACEF (K,1)	HI	Coordonnée horizontale du point initial $1 \leq HI \leq 96$
TRACEF (K,2)	VI	Coordonnée verticale du point initial $1 \leq VI \leq 76$
TRACEF (K,3)	HF	Coordonnée horizontale du point final $1 \leq HF \leq 96$
TRACEF (K,4)	VF	Coordonnée verticale du point final $VI \leq VF \leq 76^*$
TRACEF (K,5)	D	Orientation de la trace (selon le système "D" qui sera expliqué au chapitre V). Calculée directement à partir de HI, VI, HF et VF par le sous-programme "DMOYEN". $1 \leq D \leq 41$
TRACEF (K,6)	NS	Nombre de segments inclus dans la trace $1 \leq NS \leq 10$
TRACEF (K,7)	NP	Nombre de points noirs inclus dans la trace. $1 \leq NP \leq 96$

Tableau 2 - Information sur les traces dans le programme FORTRAN.

Les quatre premiers renseignements sont les seuls essentiels. L'orientation est gardée en mémoire parce qu'il serait peu commode de la recalculer à chaque fois qu'elle est requise. Quant aux deux derniers éléments d'information, ils ne servent guère qu'à

*Le point final d'une trace est toujours plus bas que son point initial.

faciliter la vérification du bon fonctionnement du programme.

Un sous programme ("DESTR", voir appendice B-1b) a été écrit pour dessiner les traces. Il y a un symbole assigné à chaque trace, selon l'ordre dans lequel le programme l'a trouvée: "1", "2", ..., "9", "0" pour les traces no. 1 à 10, "A", "B", ..., pour les suivantes. Pour chaque trace, ce symbole est tapé au point initial, au point final, et à chaque point entre les deux, tel que calculé par interpolation. (Voir figures 18 à 21)

Dans le PDP 8/E, les mêmes renseignements (sauf le dernier, NP) sont gardés sur les traces dans les mots de mémoire 3400 à 3575*. Le nombre de traces, NTRF, est dans le mot 0045.

Le tableau 3 montre le rangement de l'information sur les traces dans la mémoire.

Mot	Information
3400	HI de la 1 ^{ère} trace
3401	VI de la 1 ^{ère} trace
3402	HF de la 1 ^{ère} trace
3403	VF de la 1 ^{ère} trace
3404	D de la 1 ^{ère} trace ($4301_8 \leq D \leq 4351_8$)

*On garde 6 renseignements par trace et on leur réserve 126 (176₈) mots. Le nombre maximum de traces est donc 21 (25₈). Cela ne pose pas de difficulté car il est très rare qu'une image assez complexe pour compter plus de 21 traces se présente. Néanmoins, le programme compte les traces et cesse d'en accepter de nouvelles si 21 ont été trouvées, car en accepter plus détruirait les parties de programme dans les mots 3600 et suivants.

Mot	Information
3405	NS de la 1 ^{ère} trace
3406	HI de la 2 ^e trace
3407	VI de la 2 ^e trace
3410	HF de la 2 ^e trace
.	.
.	.
.	.
3575	NS de la 21 ^e trace (s'il y en a 21)

Tableau 3 - Information sur les traces dans la mémoire du PDP 8/E.

Un programme de dessin des traces a aussi été écrit pour le PDP 8/E et le télétype. Encore une fois, à cause du format, on doit tourner l'image de 90°. (Appendice D-1 b). On peut voir quelques dessins aux figures 26-30.

C) Quelques définitions.

- 1) Distance entre deux points $P_1(V_1, H_1)$ et $P_2(V_2, H_2)$:

$$\text{Maximum de } (|V_1 - V_2|, |H_1 - H_2|)$$

Soit, une trace de point initial (VI, HI) et de point final (VF, HF) .

- 2) Longueur de la trace:

$$\text{Maximum de } (VF - VI, |HF - HI|)$$

- 3) Distance entre la trace et un point $P(V, H)$:

- a) Si la trace est plutôt verticale (c.à.d. si

$$(VF - VI) \geq |HF - HI|$$

- i) pour que la distance soit définie, il faut que

$$VI \leq V \leq VF$$

- ii) Calcule HP , la coordonnée horizontale du point de la trace qui est à la hauteur de P , par interpolation:

$$HP = HI + \frac{(V - VI) * (HF - HI)}{(VF - VI)} \quad \left(\begin{array}{l} \text{quotient} \\ \text{arrondi}^+ \end{array} \right)$$

- iii) Si la condition i) est satisfaite, la distance est égale à

$$|HP - H|$$

- b) Si la trace est plutôt horizontale, (c.à.d. si

$$(V - VI) \leq |HF - HI|$$

⁺ La sous-routine DIVIS que nous avons faite donne un quotient arrondi plutôt que tronqué. Par contre, dans les programmes FORTRAN, il a été nécessaire d'ajouter au numérateur la moitié du dénominateur afin d'obtenir le quotient arrondi désiré. C'est ce qui explique l'apparence un peu étrange de plusieurs énoncés du programme FORTRAN. Dorénavant, dans le texte et dans les organigrammes, la division signifiera un quotient arrondi.

- i) Pour que la distance soit définie, il faut que

$$HI \leq H \leq HF \text{ ou } HF \leq H \leq HI$$

- ii) Calculer VP, la coordonnée verticale du point de la trace qui est vis à vis P, par interpolation:

$$VP = VI + \frac{(H-HI) * (VF-VI)}{(HF-HI)}$$

- iii) Si la condition i) est satisfaite, la distance est égale à

$$|VP-V|.$$

D) Mise au point du programme FORTRAN.

Quatre programmes FORTRAN sont présentés en appendice:

- 1) Programme de calcul des matrices "NEC", dont le but sera expliqué au chapitre V.
- 2) Programme de filtrage et de restauration des points enlevés par le filtre (étapes 1, 2 et 4).
- 3) Programme de recherche, prolongement et assemblage des traces (étapes 1, 3, 5 et 6).
- 4) Programme de recherche des étoiles (étape 7), utilisant comme données les traces trouvées par le programme précédent.

Nous expliquerons à l'appendice B comment réunir ces trois derniers programmes de façon à avoir, dans le programme FORTRAN, les 7 étapes énumérées à l'organigramme général.

Coder une image manuellement est très long, mais pour mettre au point les programmes 3) et 4), il fallait les éprouver sur des situations assez nombreuses et diverses. C'est pourquoi nous avons fait tourner l'image de 20° , 40° , ..., 180° pour voir, ensuite, ce que donnerait le programme de recherche des traces.* Ses résultats étaient perforés sur cartes et utilisés pour tester le programme de recherche des étoiles.

*Ceci a été fait avant que l'interface ne soit construite. Pour ne pas allonger inutilement les listes des programmes, les parties qui effectuent ces rotations ont été enlevées.

Le programme poinçonnait l'angle de rotation et le nombre de traces sur une première carte (TETADG et NTRF), puis une carte avec les sept renseignements pour chaque trace.

La rotation s'effectuait en faisant tourner l'image d'un angle θ (20° , 40° , ..., 180°) autour du point (38, 48) dans le sens anti-horaire.

Chaque point noir (V,H) était transformé en un nouveau point noir (V',H') selon la formule

$$(H'-48) = (H-48) \cos \theta - (38-V) \sin \theta$$

$$(38-V') = (H-48) \sin \theta + (38-V) \cos \theta$$

Grâce à cette méthode, nous avons pu apporter de nombreuses améliorations à l'algorithme de reconnaissance des traces et inclure les sections nécessaires au traitement de certains cas particuliers.

C'est en expérimentant avec ces trois images codées à la main et vues sous divers angles que nous avons constaté que la première version du programme nécessitait de sérieuses améliorations. (Section V-C).

E) Mise au point du programme du PDP 8/E.

Pour mettre au point le programme du PDP 8/E, nous avons utilisé les trois événements codés manuellement. Ils ont été imprimés selon le format de la figure 9 et mis dans la mémoire du PDP 8/E tels quels en utilisant la console. Puis, le contenu des mots de mémoire 2000 à 3137 fut perforé sur ruban* afin de les conserver. En utilisant ces images comme données, nous comparions les résultats du programme du mini-ordinateur à ceux du programme FORTRAN.**

La traduction ne fut pas "littérale", comme l'eut été celle d'un compilateur car, dans ce cas, le programme n'aurait pu être contenu dans les 4,096 mots de mémoire et le temps d'exécution aurait été prohibitif.

Des compteurs (COMPTE(1) à COMPTE(10)) ont été placés dans certaines parties du programme afin de voir combien de fois celles-ci étaient utilisées. C'est ce qui nous a montré la nécessité de travailler particulièrement certaines boucles. Par exemple, la boucle intérieure de la sous-routine "SUIVRE" (chapitre V) est utilisée plus de 15,000 fois pendant l'analyse d'une seule image. Par contre, d'autres parties du programme comportent des calculs exécutés assez peu souvent, et la tra-

* En utilisant le programme "ODT" qui fait partie du software de la machine.

** En vérifiant si le filtre enlevait les mêmes points dans les deux cas et si le programme de recherche des traces obtenait les mêmes résultats dans les mots 3400 et les suivants que le programme FORTRAN, dans la matrice "TRACEF".

duction, à partir du FORTRAN, est plus directe. C'est particulièrement le cas des sections "assemblage des traces" et "recherche des étoiles" (étapes 6 et 7).

Nous expliquons, ici, comment ont été traduites certaines opérations qui reviennent très souvent.

1) Examen des éléments de la matrice TRACEF.

On sait qu'un programme FORTRAN calcule l'adresse d'un élément de la matrice TRACEF(K,J) comme suit:

$$\begin{aligned} \text{Adresse de TRACEF(K,J)} &= \text{adresse de TRACEF(1,1)} + (K-1) \\ &\quad + (J-1) * NR \end{aligned}$$

où NR est le nombre de rangées dans la matrice. (qui serait 21 si cette technique était employée dans le programme du PDP 8/E).

Or, nous avons dit qu'il fallait éviter les multiplications, autant que possible.

Le fait que les éléments de la matrice TRACEF sont toujours examinés l'un à la suite de l'autre nous indique la solution: utiliser comme adresse un des registres à indexage. Ainsi, on prend, par exemple, AUT11. On le met d'abord à 3377 (3400-1) et on utilise les instructions

BCLE, (AUT11) → HI

(AUT11) → VI

.

.

.

etc.

BCLE, TAD 1 AUT11

DCA HI

TAD 1 AUT11

DCA . VI

.

.

etc.

On obtient ainsi tous les renseignements sur la première trace en utilisant 6 fois ce registre indexé. AUTILL a maintenant la valeur 3405 et, si on recommence (à BCLE), on obtient maintenant les six renseignements de la deuxième trace, et ainsi de suite.

2) Examen séquentiel de points dans l'image.

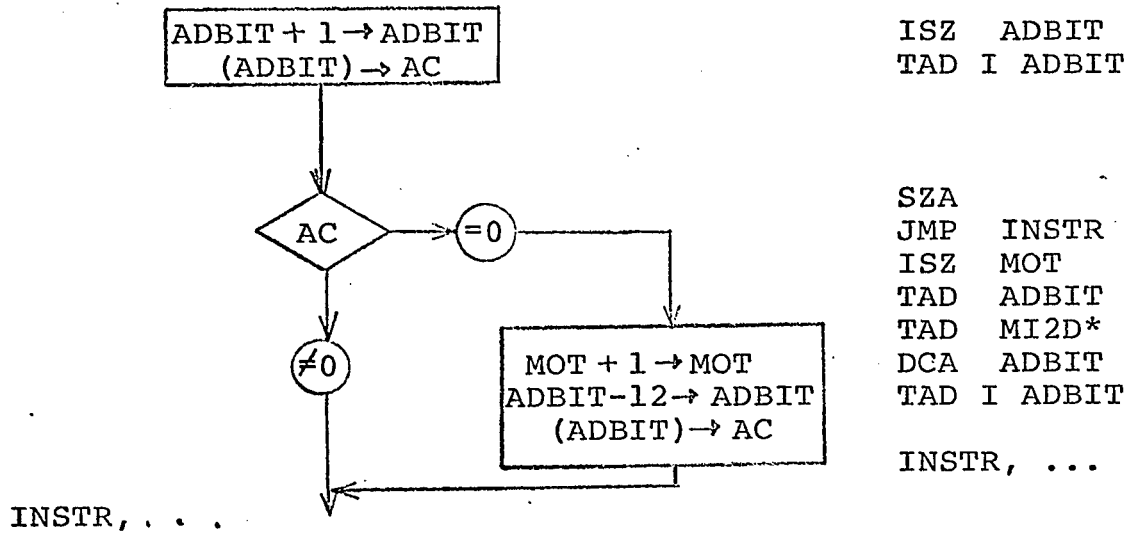
a) $V+1 \rightarrow V$ (Descendre une unité plus bas.)

On voit (figure 9) que le point en dessous de celui qui est désigné par (MOT, ADBIT) est (MOT+10₈, ADBIT). Ainsi (tableau 1), le point (V,H) = (1,1) est dans le mot 2000 et le bit 0 (ADBIT = 1120 ou 1150) tandis que (2,1) est au mot 2010, même bit.

b) $H+1 \rightarrow H$. (Aller une unité à droite.)

On utilise les masques de la première liste (1120 à 1133 pour les bits 0 à 11, respectivement). Pour incrémenter H, on incrémente ADBIT, l'adresse du masque. Puis, on teste ce masque pour voir s'il est vide. Si oui, c'est que H était le dernier bit du mot et H+1 est le premier bit du mot suivant. Par exemple, (1,11) correspond à MOT = 2000 et ADBIT = 1132 (bit 10). Le décalage à droite pour aller à (1,12) consiste à incrémenter ADBIT de 1132 à 1133 (bit 11). Par contre, passer de (1,12) à (1,13), c'est aller de MOT = 2000, ADBIT = 1133 (bit 11) à MOT = 2001, ADBIT = 1120 (bit 0). Quand ADBIT est incrémenté, le résultat est 1134, qui est l'adresse du mot vide. C'est ce vide qui avertit le programme qu'il faut

- i) incrémenter MOT ii) soustraire 12 (14_8) de ADBIT pour le ramener à 1120.



AC contient maintenant
le bit à examiner et "MOT",
l'adresse du mot à examiner.

Equivalent en langage d'assemblée du déplacement à droite $H+1 \rightarrow H$.

Dans les cas où on veut examiner les points une fois chacun, il est souvent préférable d'utiliser des registres d'indexation comme adresses du mot et du masque du bit à examiner, pour qu'ils soient incrémentés automatiquement sans qu'une instruction ISZ ne soit nécessaire (par exemple l'emploi de MOTI et ADBITI, registres "AUTO INDEX" 0016 et 0017, dans le programme de recherche des traces).

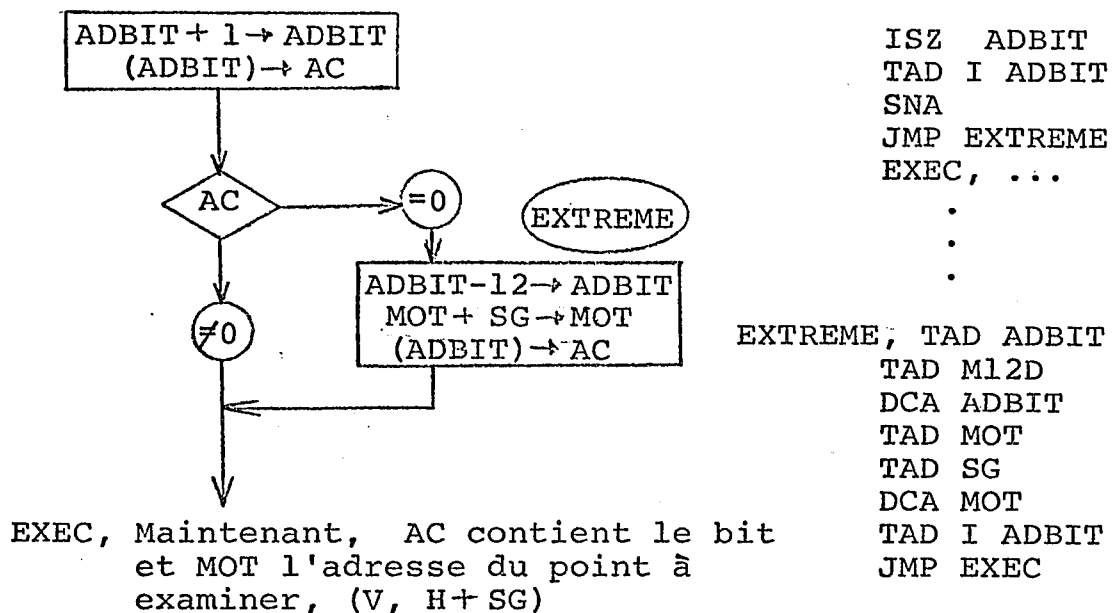
c) $H-1 \rightarrow H$ (aller une unité à gauche)

Pour cela, les masques de la 2^e liste (mots 1150, ..., 1135 pour les bits 0, ..., 11, respectivement) sont utilisés. De cette

* M12D = -12 = -14_8 ou 7764_8 en arithmétique de complément binaire.

façon, incrémenter l'adresse du bit produit maintenant un décalage à gauche. Quand le décalage à gauche fait passer au dernier bit du mot précédent, la même technique que ci-dessus est utilisée: le registre 1151 est vide et ce vide avertit le programme qu'il faut i) diminuer MOT ii) soustraire 148 de ADBIT pour le ramener à 1135 (bit 11). C'est dire que l'opération s'écrit exactement de la même façon pour $H-1 \rightarrow H$ que pour $H+1 \rightarrow H$ à ceci près qu'il faut $MOT-1 \rightarrow MOT$ plutôt que $MOT+1 \rightarrow MOT$.

La sous-routine SUIVRE, qui peut avoir à faire $H-1 \rightarrow H$ ou $H+1 \rightarrow H$ selon le cas utilise une variable $SG = \pm 1$. Ainsi, $H+SG \rightarrow H$ se traduit par:



Equivalent en langage d'assemblée du décalage horizontal

$H+SG \rightarrow H \quad (SG = \pm 1)$

Note: si $SG = +1$, (ADBIT) est un masque de la première liste (1120 à 1133); si $SG = -1$, (ADBIT) est de la seconde liste (1135 à 1150).

IV. Le filtrage

L'étape du filtrage a pour but de réduire le bruit de fond en enlevant les points parasites et, ainsi, faciliter le travail de recherche des traces.

Toutefois, on ne peut réduire le bruit de fond sans enlever en même temps quelques points valables qui font partie des traces. C'est pourquoi les points que le filtre perçoit comme étant du bruit ne sont pas enlevés définitivement, mais mis de côté pour être remis dans l'image après l'étape de recherche des traces, afin que l'étape du prolongement des traces puisse ajouter aux traces ceux d'entre eux qui avaient été enlevés à tort.

A) Elimination et mise en réserve des points isolés.

Ce filtre est conçu pour mettre de côté les points qui n'ont pas de proche voisin, les points isolés. Un point noir (VI, HI)* est considéré comme isolé si, parmi les 15 points du rectangle 3 x 5,

$$VI \leq V \leq VI + 2$$

$$HI - 2 \leq H \leq HI + 2$$

il n'y a pas au moins 3 points noirs (donc, au moins deux points noirs (V,H) voisins de (VI,HI), puisque ce dernier est dans le rectangle et est noir).

* Les points (VI,HI) des deux dernières lignes ($VI \geq 74$), des deux dernières colonnes ($HI > 94$), et des deux premières colonnes ($HI < 3$) sont exclus de ce test parce que le rectangle 3 x 5 contiendrait des points en dehors du cadre de l'image.

Ce critère est purement empirique: plusieurs autres ont été essayés, mais c'est celui-ci qui s'est avéré le plus efficace.

Ecrire cet algorithme en FORTRAN est assez simple et direct (organigramme 2; appendice B-2).

Par contre, la version du PDP 8/E est plus compliquée à cause de la nécessité de définir le rectangle 3 x 5 en termes de mots et bits dans la mémoire. La façon de le faire est la suivante: d'abord, la boucle double:

Pour VI = 1,74

Pour HI = 3,94

devient une boucle double: "BCLMOT" et "BCLBIT" (section A, B, C, D de l'organigramme 3). Pour examiner tous les points tour à tour, et déceler les points noirs parmi eux, la technique décrite au chapitre précédent est utilisée ("MOT" et "ADBIT" sont ici, les registres à indexage AUTI11 et AUTI12).

Quand un point noir est trouvé, la technique du masquage est utilisée pour prendre, dans le mot en question et les deux en dessous, les bits qui entrent dans le cadre 3 x 5.

Par exemple, si le bit 5 (masque 1125) du mot 2000 est à 1 (point (1,6) noir), le rectangle 3 x 5 consiste en les bits 3, 4, 5, 6 et 7 des mots 2000, 2010 et 2020. Pour obtenir l'adresse du masque, 4500 est ajouté à l'adresse du bit (section F) et, dans le mot 5625, se trouve le masque approprié: 000 111 110 000. La sous-routine COMPTE masque les trois mots et compte (CTR) le nombre de bits 1. Pour réduire le temps d'exécution, la sous-routine "COMPTE" cesse de compter dès que trois points noirs

ont été trouvés plutôt que de terminer le rectangle, puisqu'on sait dès lors que le point ne sera pas enlevé (Y). Le comptage s'effectue par des décalages à gauche et en testant le bit "LINK". Là aussi, la technique utilisée minimise le temps d'exécution (X).

Dans le cas où le point noir à examiner est dans le bit 0 ou 1, il faut examiner les derniers bits du mot précédent pour compléter le rectangle 3 x 5. (H) Si, de plus le mot est le premier de la ligne, ceci correspond à la situation $H \leq 2$ et le programme laisse ce point noir dans l'image et cesse le comptage (I). Des techniques analogues s'appliquent pour le cas où le point noir est dans le bit 10 ou 11 ((L) et (M)).

L'enlèvement du point noir, lorsqu'il y a lieu, se fait par un masque complément du bit, c'est à dire, s'il s'agit du bit 5 (000 001 000 000), un ET logique est effectué entre le mot et un masque égal à 111 110 111 111. Ainsi, le bit 5 est enlevé tandis que les autres bits restent intacts ((S), (T)).

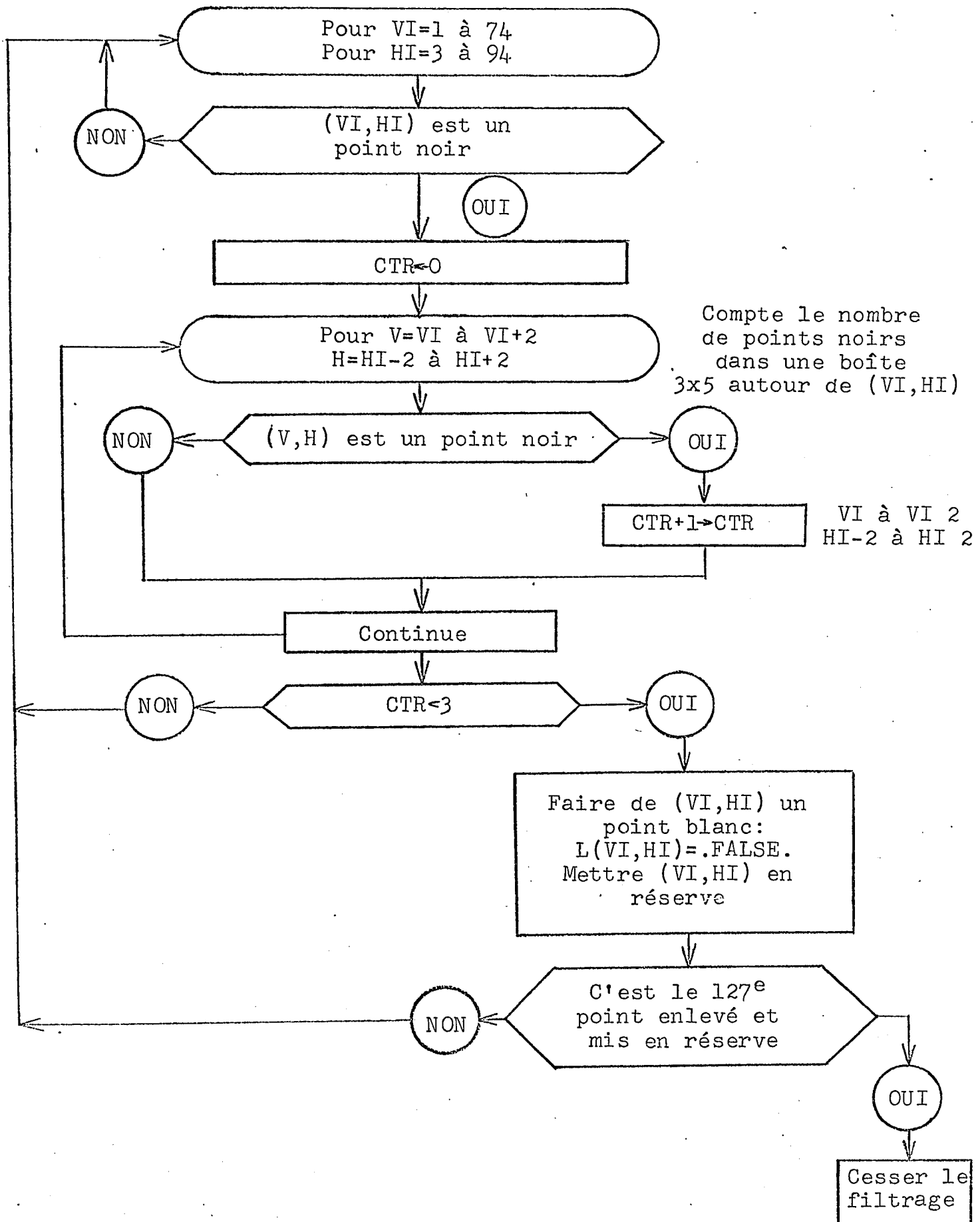
Les points enlevés sont mis en réserve dans une liste qui occupe les mots 200 à 575. L'adresse du mot et celle du bit sont conservées (S). Un mot vide termine cette liste (R).

Cette façon de procéder facilite l'étape de remise dans l'image des points enlevés. (Organigramme 4).

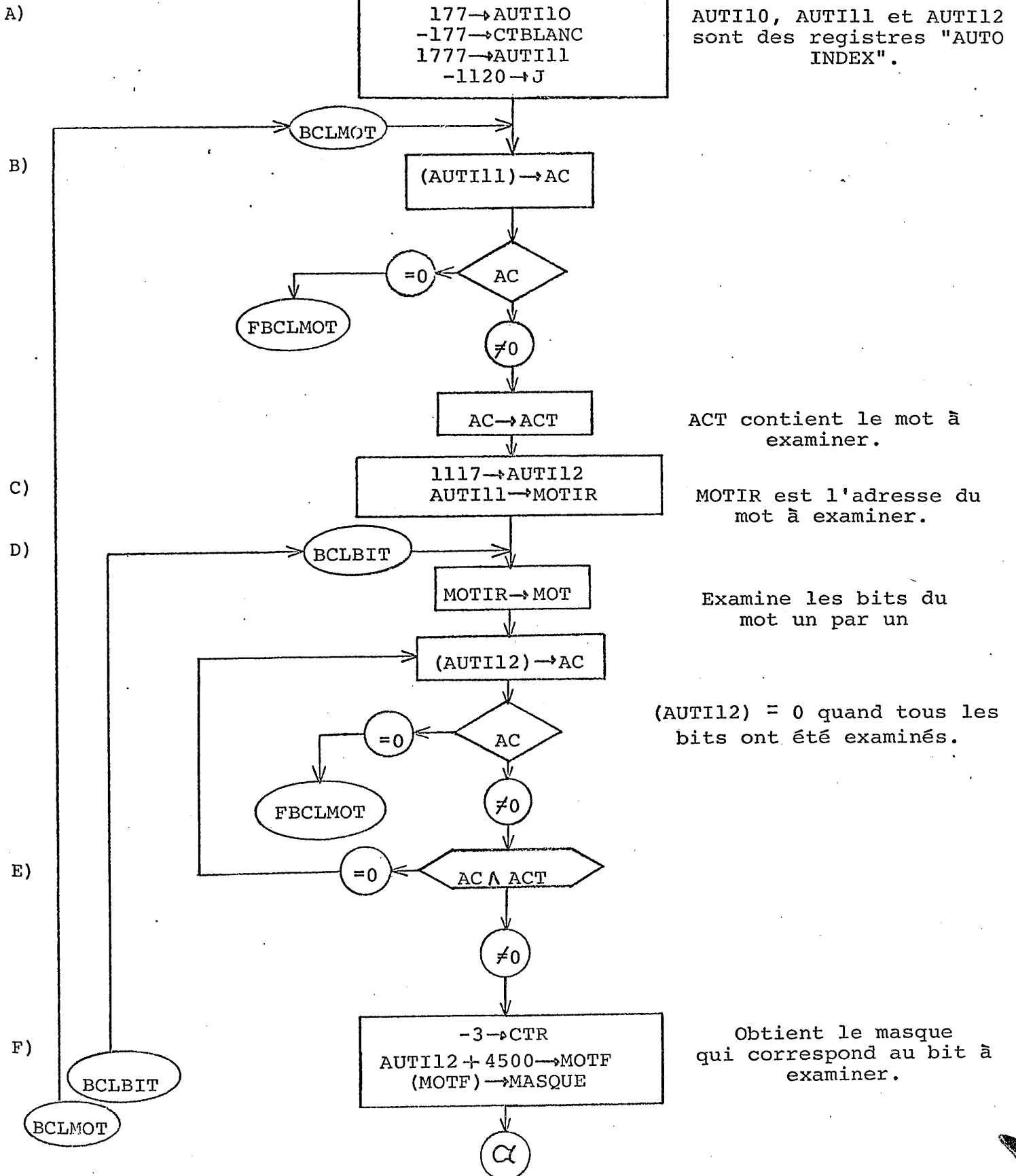
2-FILTRAGE

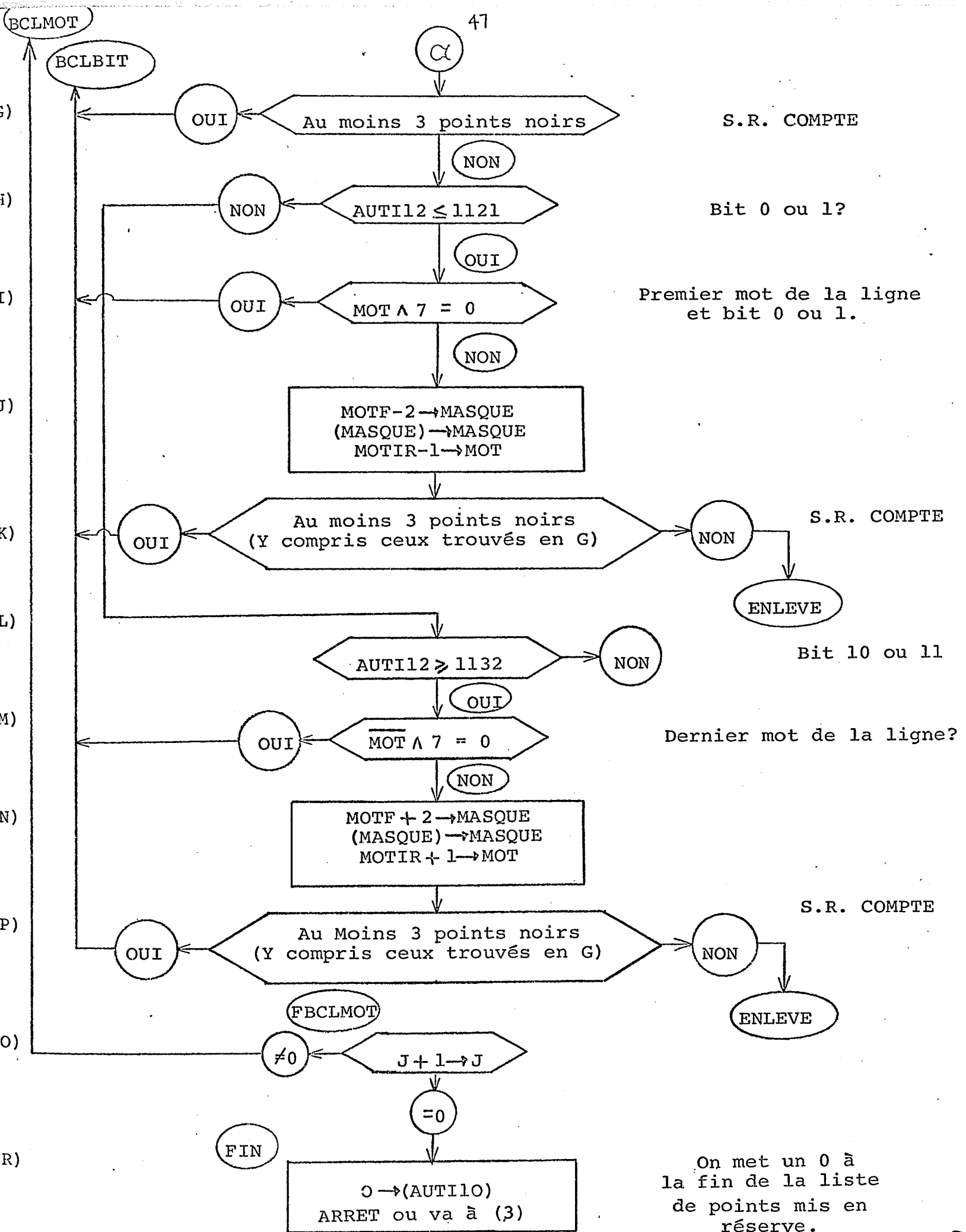
2

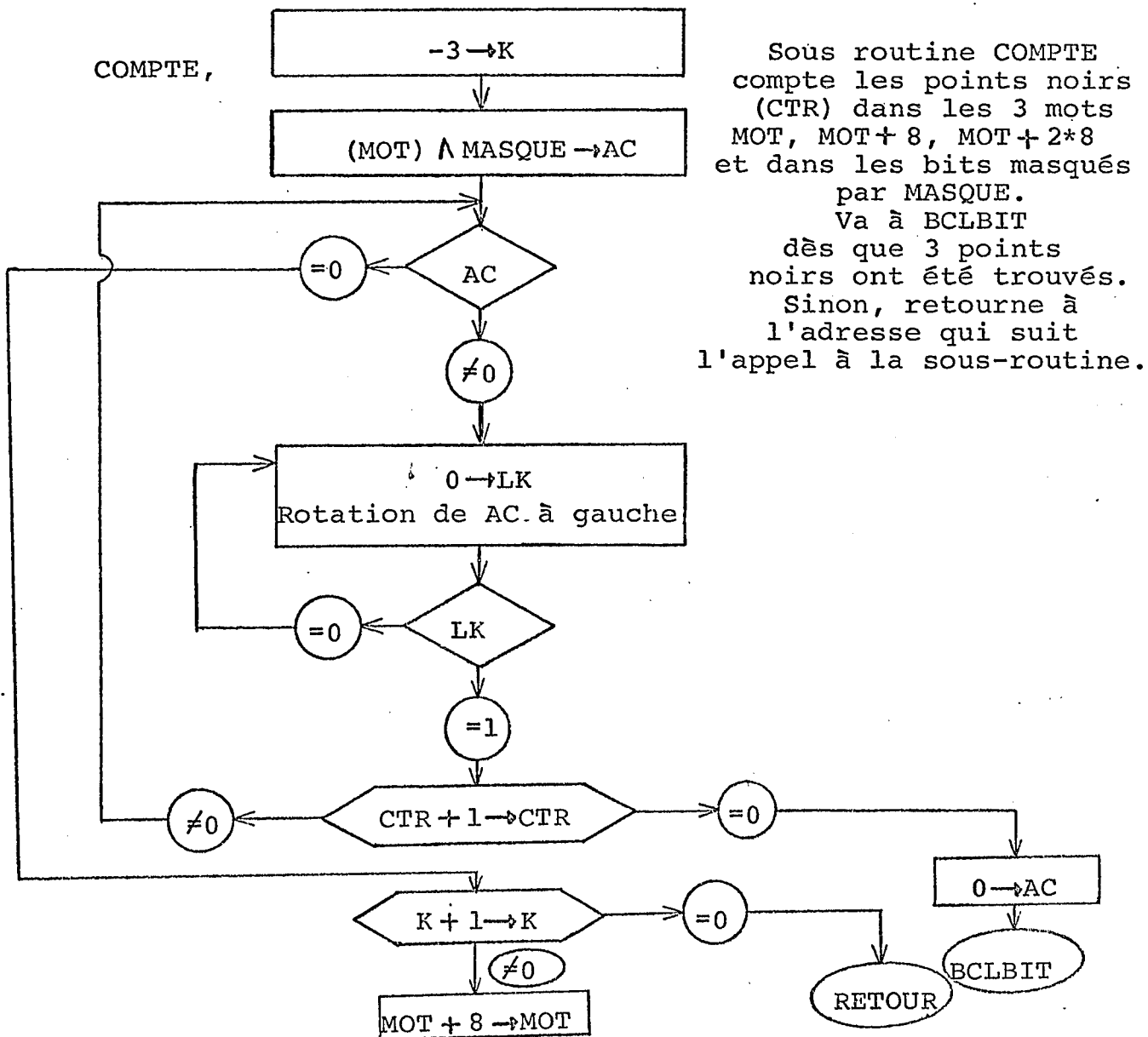
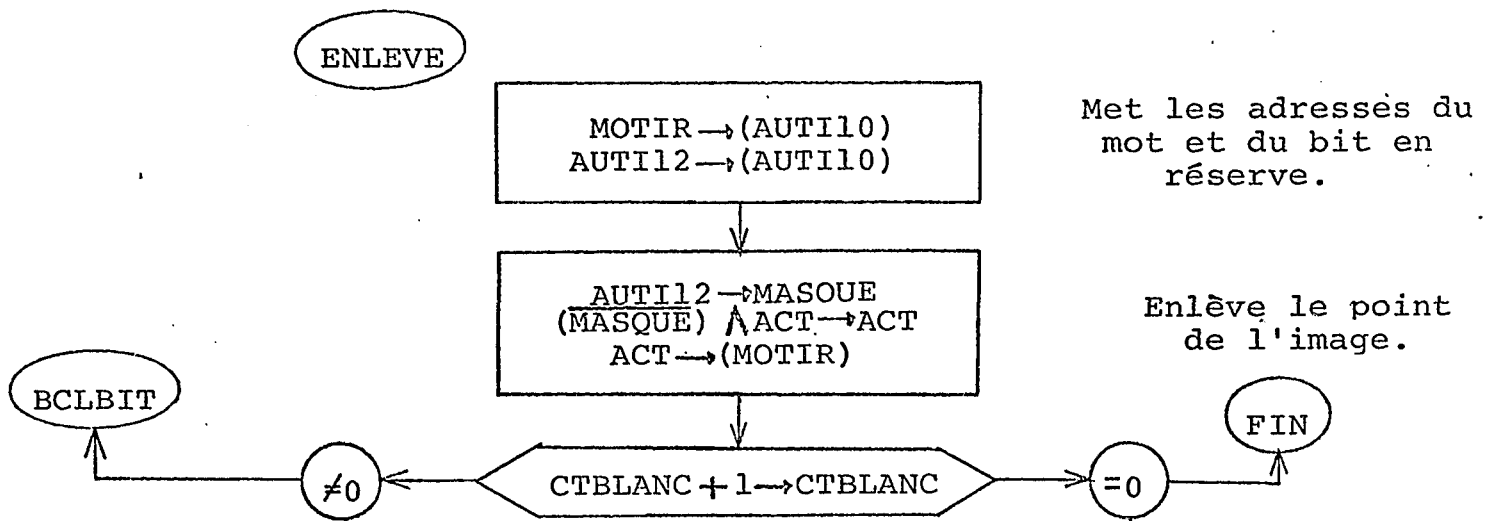
Elimination et mise en réserve des points isolés



2- Filtrage - Version ASSEMBLEUR







4

4- Restauration des points enlevés.

A)

177 → AUTI10

B)

(AUTI10) → AC

=0

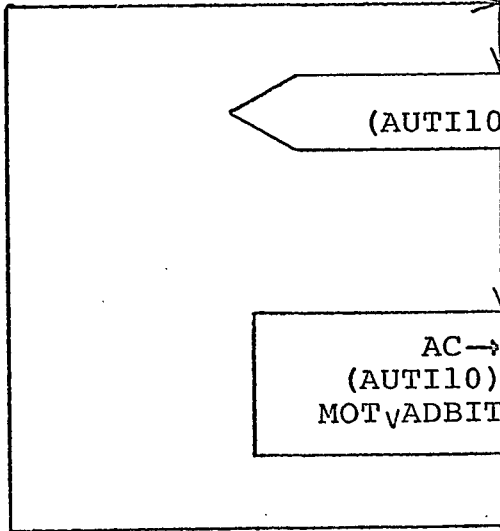
Un 0 indique
la fin de la
liste de
réserve.

arrêt
ou va à 5

C)

AC → MOT
(AUTI10) → ADBIT
MOT ∨ ADBIT → (MOT)

On remet le
bit dans le mot
par un OU logique
entre le masque du
bit et le mot.



B) Résultats du filtre.

Le filtrage réduit considérablement le temps nécessaire à la recherche des traces en enlevant les points isolés. Toutefois, il a tendance à enlever quelques points au bout des traces, et à agrandir les lacunes si les traces ne sont pas denses.

L'effet est bénéfique pour des images où les traces sont longues et denses, et où il y a beaucoup de bruit (figures 13, 14, 15) mais nocif quand les traces comportent déjà des lacunes (figure 16).

Le fait de remettre les points enlevés dans l'image avant l'étape de prolongement permet à cette dernière de compléter les traces avec les bouts que le filtre pourrait lui avoir enlevés. Toutefois, dans le cas de l'image 3 (fig. 16), on voit que certaines traces se font enlever tellement de points qu'elles sont tout à fait manquées à l'étape 3 (trace de 11 heures, par exemple).

C'est pourquoi, deux versions du programme ont été réalisées: l'une avec, et l'autre sans le filtre.

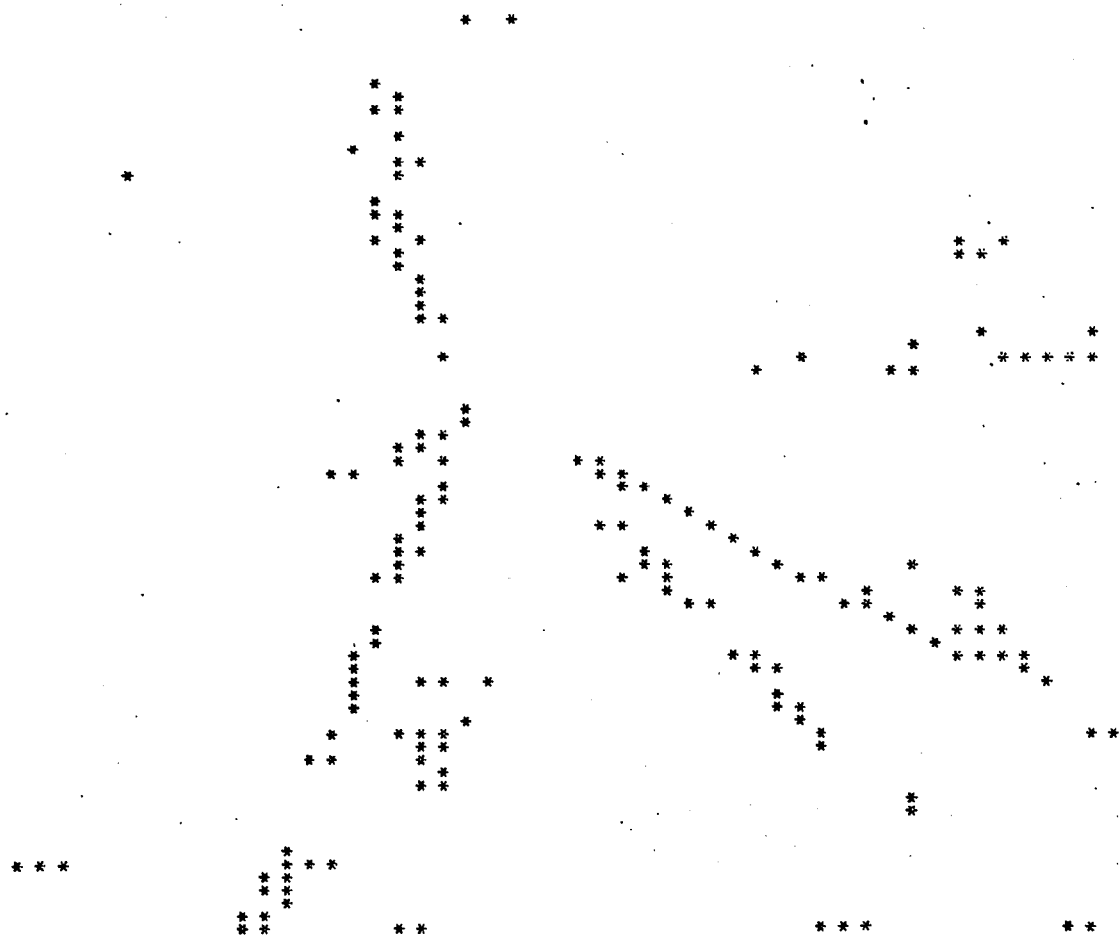


Figure 13 - Evénement cosmique, après filtrage.

(Comparer à la figure 8)

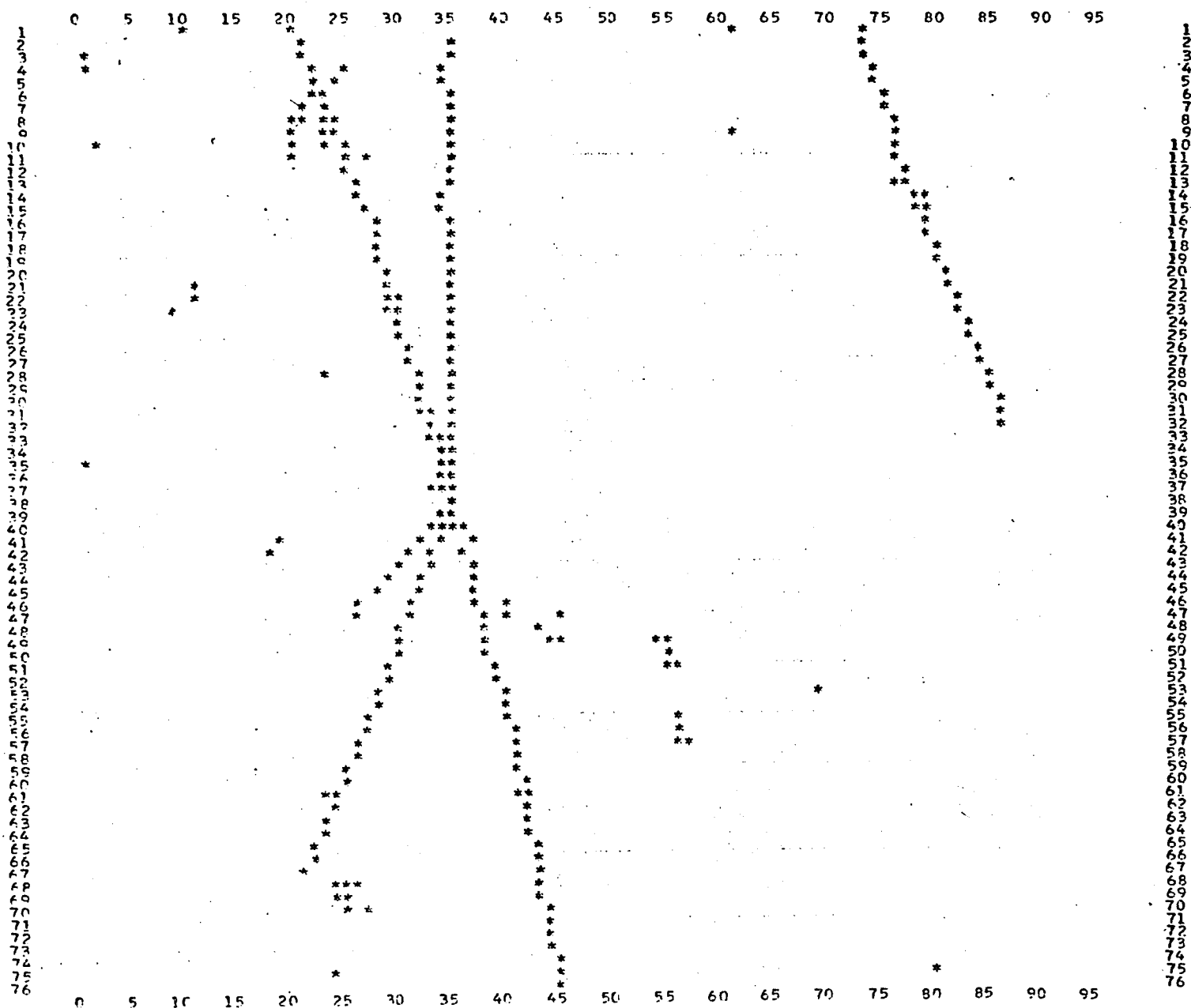


Figure 14—Etoile no.1 après filtrage

(Comparer à la figure 9)

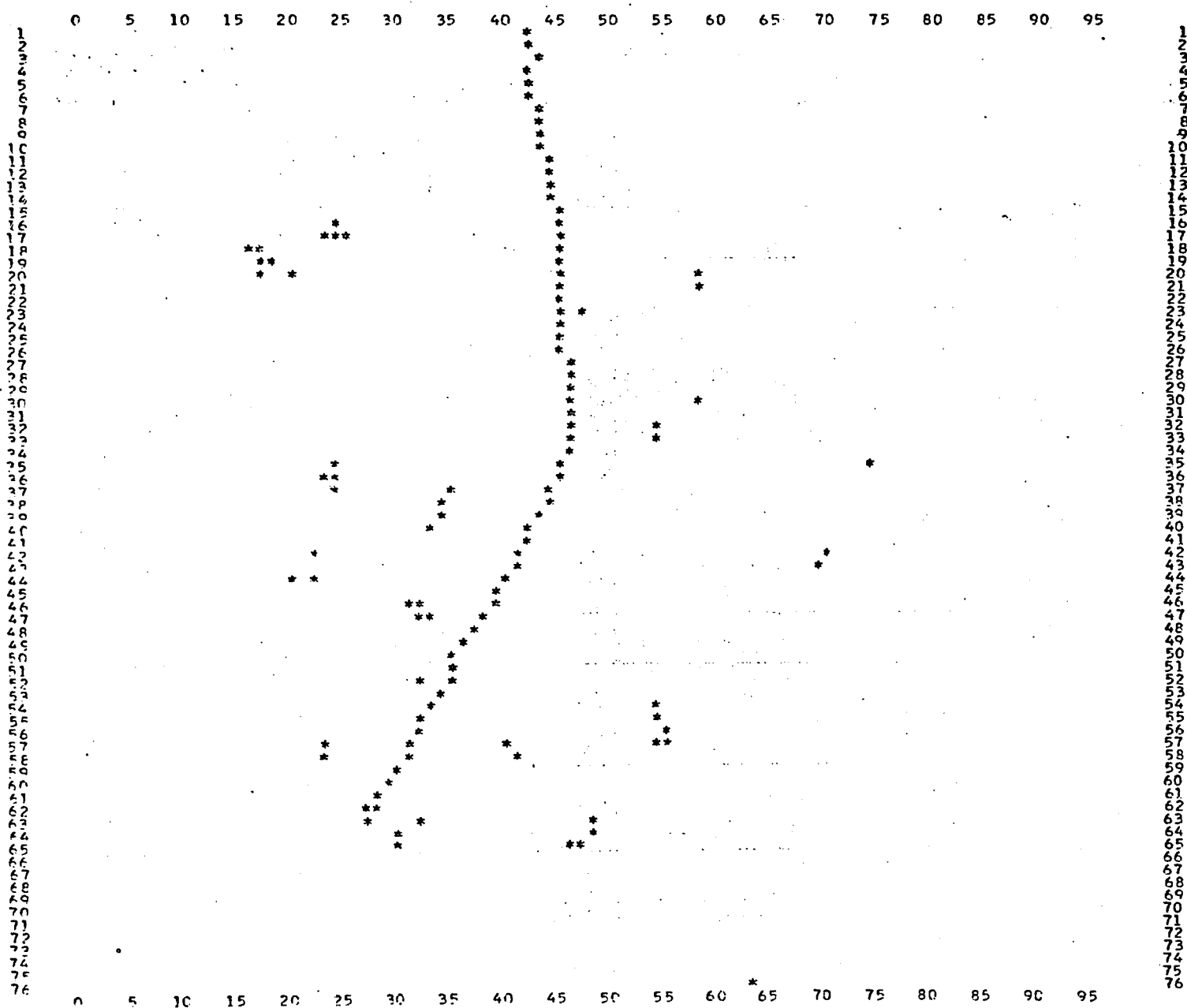


Figure 15 — Etoile no.2 après filtrage
(Comparer à la figure 11)

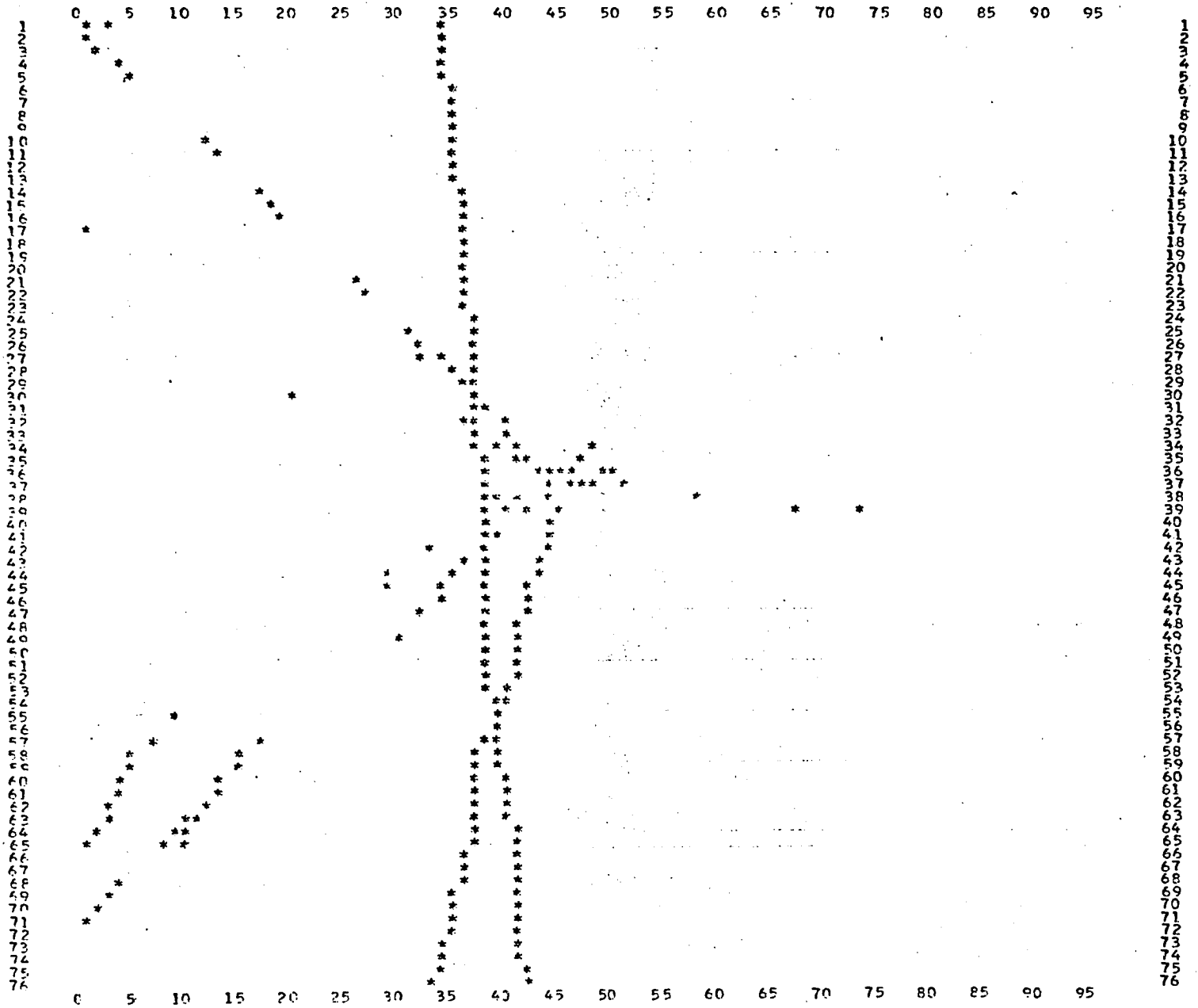


Figure 16—Etoile no.3 après filtrage
(Comparer à la figure 12)

C) Autres méthodes de filtrage.

Deux autres tentatives ont été faites pour réduire le bruit.

Afin d'empêcher certains amas de points noirs (que l'interface n'a pas suffisamment réduits) de nuire, un programme a été écrit pour les enlever et les mettre en réserve. Cette méthode est à l'opposé de la précédente: le nombre de points noirs est compté dans chaque bloc de 11 x 12 (11 mots de 12 bits - cette division étant la plus commode) et, s'il dépasse, disons, 30 (sur 132), le bloc est mis en réserve, puis vidé. Cette méthode a l'inconvénient d'enlever les centres des étoiles, puisqu'il y a plusieurs points noirs là où des traces se rencontrent.

Nous avons aussi tenté d'améliorer la qualité de l'image en superposant deux images transmises l'une après l'autre. La première est placée dans un bloc de mémoire, la seconde dans un autre, et un ET logique, bit par bit, est effectué entre les deux, de sorte que seuls les points qui sont noirs sur les deux images sont considérés noirs. Cette méthode ne donne pas, non plus, de bons résultats et a, par conséquent, été abandonnée.

V - Recherche des traces

La recherche des traces s'effectue en cherchant des batonnets (segments) de 10 points de longueur et en les unissant pour former des traces (organigramme 11).

Comme c'est cette étape qui prend le plus de temps, elle a été conçue avec soin et de façon à minimiser le temps requis. Le programme - surtout la version du PDP 8/E - est sous forme de sous-routines qui s'appellent l'une l'autre.

Le programme principal, en FORTRAN, fait appel à la sous-routine EVPROX pour éviter de chercher des traces là où il y en a déjà et à la sous-routine CHSEG pour chercher les segments. La structure du programme est résumée à l'organigramme 10 (A $\Rightarrow$ B signifie que le programme ou sous-programme A fait appel au sous-programme B). Pour ne pas alourdir ces organigrammes, les appels aux sous-routines de minimum et maximum, de multiplication, et de division ne sont pas mentionnés.

A) Orientations et interpolation

La figure 17 illustre les 41 orientations (D) utilisées dans ce programme. En définissant θ_D comme l'angle entre le demi-axe horizontal positif et la ligne d'orientation D, on distingue quatre groupes d'orientations:

- 1) $\tan \theta_1, \theta_2, \dots, \theta_{10}, = 0.0, 0.1, \dots, 0.9$ et les coordonnées du 10^e point (par rapport au point initial) sont (V_{10}, H_{10})
 $1, 2, \dots, 10 = (0, 10), (1, 10), \dots, (9, 10)$ respectivement.

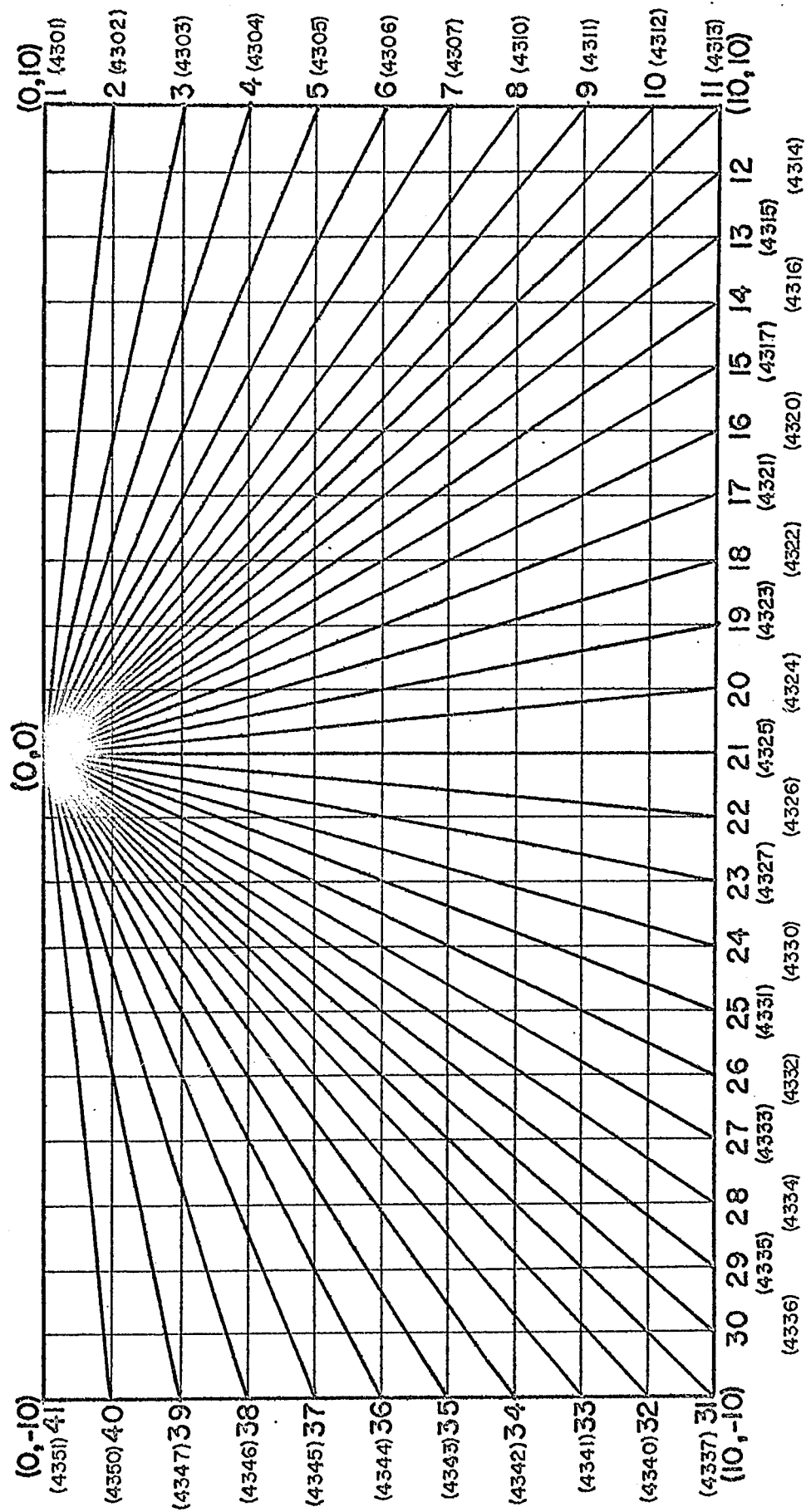


Figure 17 - Les 41 orientations "D".

2) $\cotan \theta_{11}, \theta_{12}, \dots, \theta_{21} = 1.0, 0.9, \dots, 0.0$ et le 10^e point est

$$(V_{10}, H_{10})_{11, 12, \dots, 21} = (10, 10), (10, 9), \dots, (10, 0)$$

3) $\cotan \theta_{22}, \theta_{23}, \dots, \theta_{31} = -0.1-0.2, \dots, -1.0$ et le 10^e point est

$$(V_{10}, H_{10})_{22, 23, \dots, 31} = (10, -1), (10, -2), \dots, (10, -10)$$

4) $\tan \theta_{32}, \theta_{33}, \dots, \theta_{41} = -0.9, -0.8, \dots, 0.0$ et le 10^e point est

$$(V_{10}, H_{10})_{32, 33, \dots, 41} = (9, -10), (8, -10), \dots, (0, -10).$$

A partir du point initial (0^e point) et du 10^e point, on peut calculer les coordonnées des autres points par interpolation. Par exemple, pour $D = 4$, (groupe 1) $(V_{10}, H_{10}) = (3, 10)$ et, par conséquent, les points sont à

$$(0, 0), (\frac{1*3}{10}, 1), (\frac{2*3}{10}, 2), \dots, (\frac{9*3}{10}, 9), (\frac{10*3}{10}, 10)$$

Donc, la coordonnée horizontale du J^e point est J , tandis que la verticale est calculée par interpolation. Il en va de même pour le groupe (4), sauf que la coordonnée horizontale est $-J$. Par contre, pour les groupes 2 et 3, la coordonnée verticale du J^e point est J tandis que la coordonnée horizontale est calculée par interpolation.

L'appendice A-1) montre le programme qui effectue ce calcul d'interpolation tandis que la table d'interpolation complète est en A-2).

Faire mémoriser cette table telle quelle à l'ordinateur eut été coûteux en termes de mémoire et peu utile. Pour nos besoins, il est préférable de définir la position du J^e point à partir de celle du $(J-1)^e$ puisque, partout dans le programme, les points sont examinés l'un à la suite de l'autre.

On définit, pour une orientation D donnée,

NECH(J,D): VRAI si $H_J = H_{J-1} + SG *$ (J = 1 à 10)
 FAUX si $H_J = H_{J-1}$

NECV(J,D): VRAI si $V_J = V_{J-1} + 1$
 FAUX si $V_J = V_{J-1}$

Par exemple, pour $D = 38$, les 0^e , 1^e et 2^e points sont $(0,0)$, $(0,-1)$, $(1,-2)$ et $SG = -1$.

Donc, NECV (1,38) = FAUX NECV (2,38) = VRAI
 NECH (1,38) = VRAI NECH (2,38) = VRAI

Dans le programme FORTRAN, les matrices logiques NECV et NECH sont calculées au début du programme de recherche des traces (ISN 0039-ISN 0064).

Grâce à ces matrices, le calcul des positions des 10 points qui suivent (VI,HI), selon l'orientation D s'effectue comme suit:

* SG = 1 si D = 1 à 21 (groupes 1 et 2)
 SG = -1 si D = 22 à 41 (groupes 3 et 4).

H = HI

V = VI

SG = 1

IF (D.GT.21) SG = -1

Do n J = 1,10

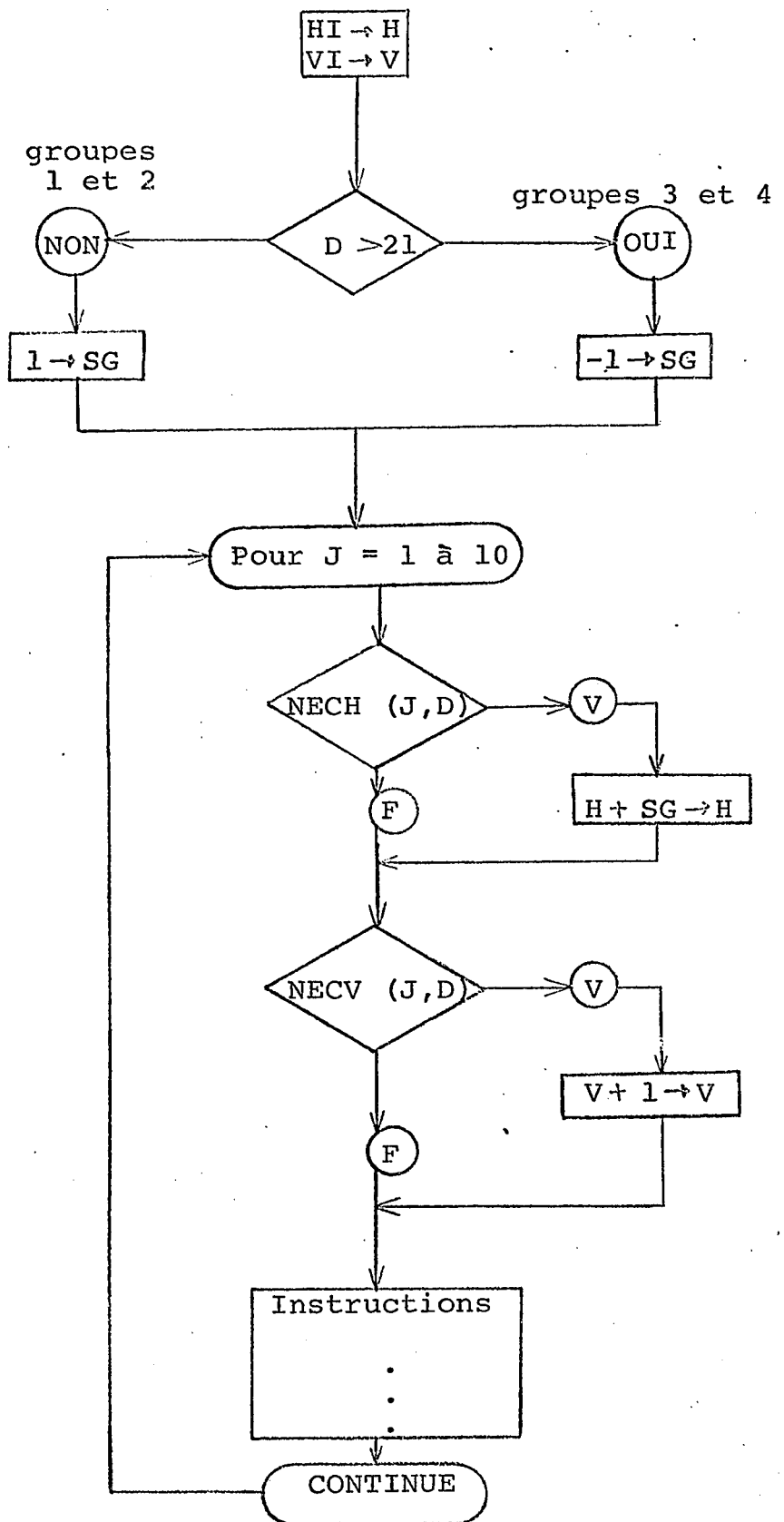
IF (NECH(J,D)) H = H + SG

IF (NECV(J,D)) V = V + 1

Instructions

·
·
·

n CONTINUE



Interpolation selon D dans le programme FORTRAN

Pour le programme en langage d'assemblée, cette même technique est appliquée, mais avec quelques modifications.

D'abord, en regardant les matrices NECV et NECH (appendice A-3), on constate que NECH est entièrement vraie pour les groupes 1 et 4 tandis que NECV l'est pour les groupes 2 et 3. Pour chaque orientation, il suffit donc de retenir NECV pour les groupes 1 et 4, NECH pour les groupes 2 et 3. Ces 10 éléments binaires (pour chaque orientation) sont retenus dans les bits 0 à 9 des mots 4301, 4302, ..., 4351 pour $D = 1, 2, \dots, 41^*$.

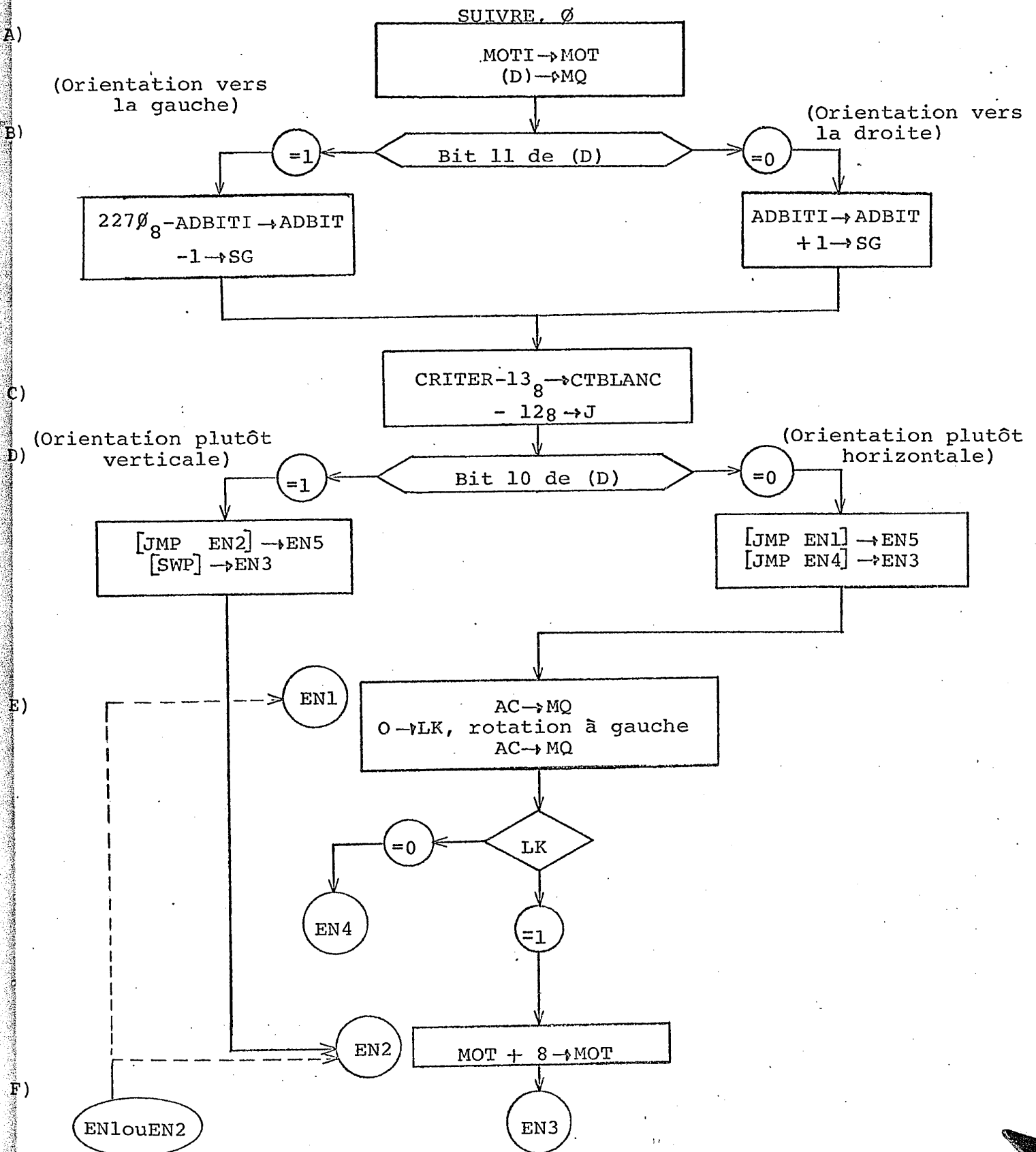
Le bit 10 du mot D distingue les groupes 1 et 4 (bit 10 = 0) des groupes 2 et 3 (bit 10 = 1). Le bit 11 sert de bit de signe: il est 0 pour les groupes 1 et 2, 1 pour les groupes 3 et 4.

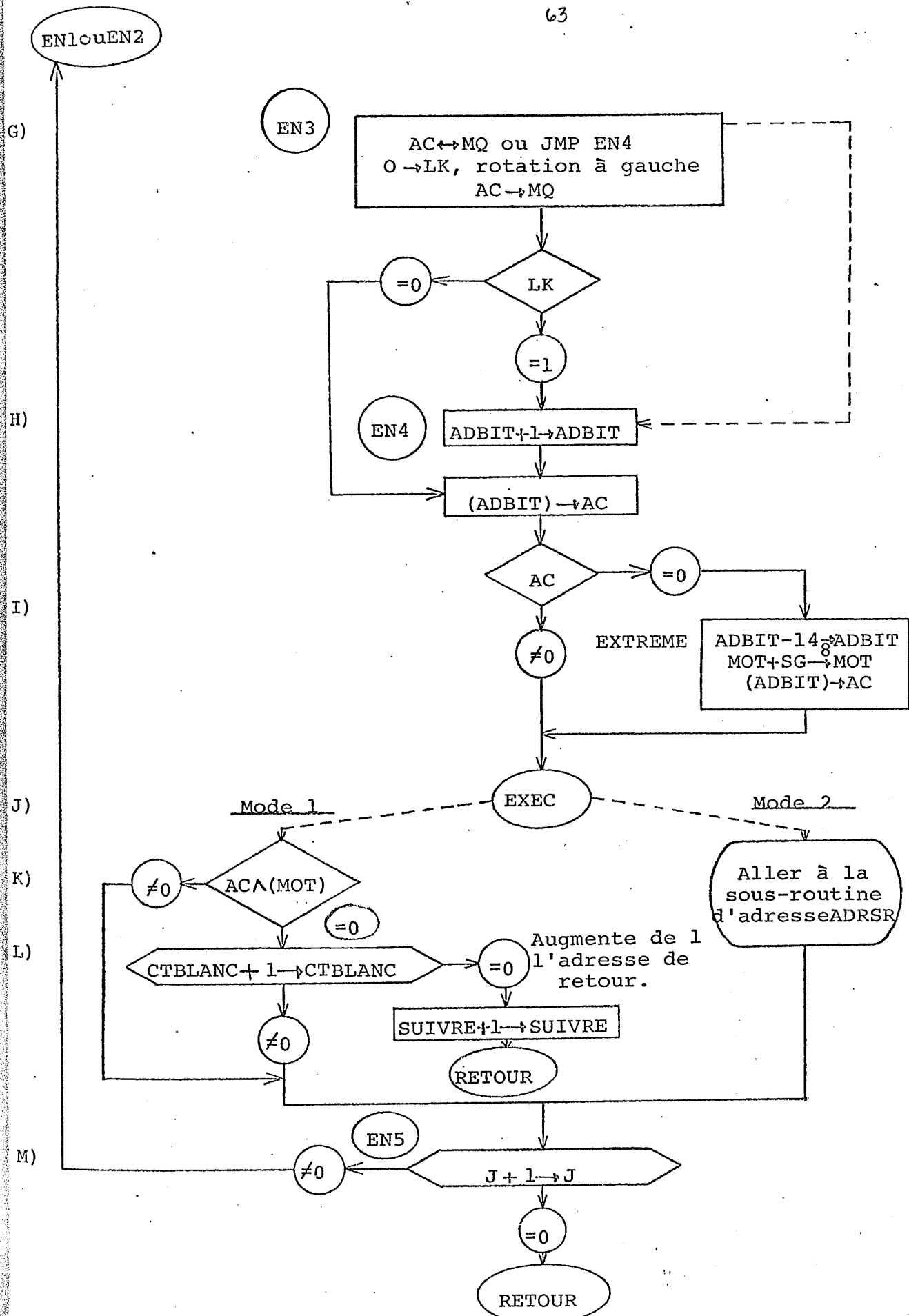
La sous-routine SUIVRE calcule les positions des 10 points qui suivent le point de bit de masque "ADBITI"** du mot "MOTI", selon l'orientation "D". Faire appel à cette sous-routine équivaut à la séquence FORTRAN écrite ci-dessus.

D'abord, le mot "D" est placé dans le MQ, où l'accès sera plus rapide (MQ → AC: 1.2 us. contre 2.6 usec. pour mémoire → AC). Le "MOT" est mis à "MOTI" et, selon le groupe de D, $SG = \pm 1$ et ADBIT est mis à ADBITI (1^{ère} liste de masques) ou $2270_g - ADBITI$

* Seules les valeurs 4301, ..., 4351 sont utilisées dans le programme en langage d'assemblée, parce que c'est plus commode. On transforme du "D" du FORTRAN à celui du PDP-8/E en traduisant le nombre en octal, puis, en ajoutant 4300_g. Par exemple: $D = 29 \rightarrow D = 35_g + 4300_g = 4335_g$ et NECH (J, 29) se trouve dans les bits 0, 1, ..., 9 du mot 4335 (pour $J = 1, 2, \dots, 10$).

** Il faut que $1120 \leq ADBITI \leq 1133$.

S.R. SUIVRE



(masque de la 2^e liste pour le même bit que ADBITI). (Organigramme 5 A, B).

Le décalage vertical s'effectue à EN2(F) tandis que le déplacement horizontal est à EN4 (H,I).

A EN1,(E) le bit approprié du MQ est examiné pour décider s'il y a lieu de faire un décalage vertical, pour les groupes 1 et 4, tandis que le bit du MQ est examiné à EN3 et y décide du décalage horizontal, pour les groupes 2 et 3. Dans chaque cas, les instructions de saut, placées à l'endroit approprié par la partie (D), déterminent celui des deux tests qui sera exécuté à l'intérieur de la boucle.

Section EXEC

Une fois calculée la position du point,* la section EXEC contient les instructions à exécuter. Dans la grande majorité des cas (90% environ) il s'agit de chercher une orientation comptant au moins "CRITER"** points noirs sur 10 points examinés. Une instruction "JMS" ferait alors perdre quelques cycles et, il est préférable de faire fonctionner la S.R. SUIVRE dans un mode distinct, le mode 1.

Il s'avère plus rapide de compter les points blancs. Si CRITER = 5 et que les 6 premiers points sont blancs, il est inutile d'examiner les 4 autres. CTBLANC est d'abord mis à CRITER - 11 (-13_8) (C) et est incrémentée à chaque point blanc. Dès que CTBLANC atteint 0, il ne peut y avoir CRITER points

* A ce point-ci, AC contient le bit et MOT, l'adresse du point à examiner.

** D'ordinaire, CRITER = 4, 5 ou 6.

noirs et l'adresse de retour est incrémentée afin que le programme qui appelle agisse en conséquence.

Dans le mode 2, la partie EXEC consiste simplement en un saut à la sous-routine d'adresse ADRSR.

La sous-routine SUIVRE est mise dans un mode ou dans l'autre en modifiant ses instructions (EXEC et la suivante: 1065 et 1066), ce que font les sous-routines "MODE1" et "MODE2".

La sous-routine SUIVRE peut fonctionner dans le sens inverse du sens usuel. Il suffit, pour cela, de modifier deux instructions pour que F) devienne MOT - 8 → MOT et que le test B) soit inversé. La sous-routine "ARRIERE" effectue ces changements, tandis que la sous-routine "AVANT" remet ces instructions dans leur état usuel (qui est celui imprimé sur le listing de la S.R. SUIVRE, appendice A-5).

Afin de s'assurer que la sous-routine SUIVRE fonctionnera dans le sens usuel, il est fait appel à "AVANT" dès le début du programme de recherche des traces.

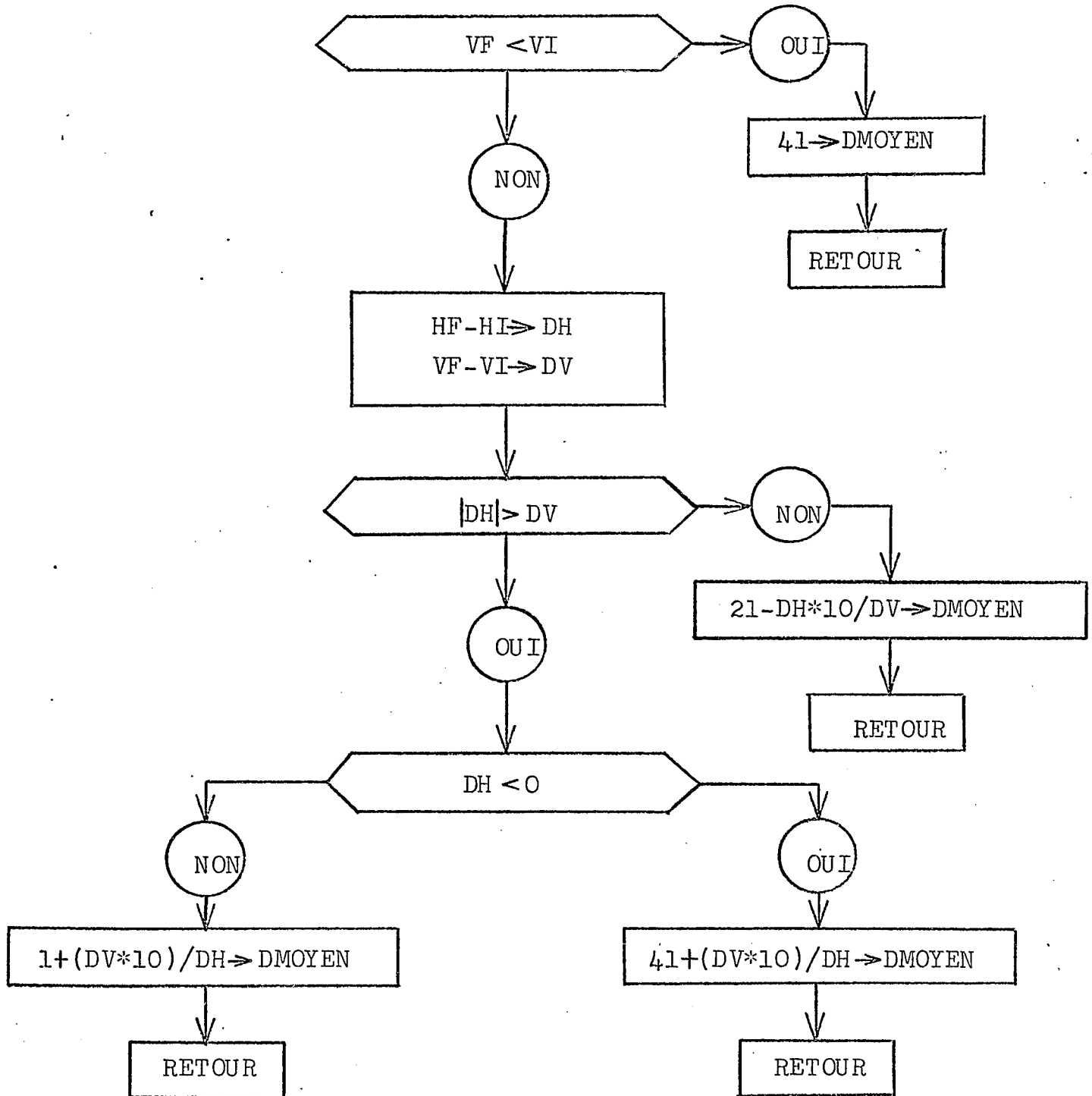
SOUS-PROGRAMME "DMOYEN"

Ce sous-programme calcule l'orientation d'une trace qui va de (VI, HI) à (VF, HF) en calculant la tangente ou la cotangente (selon celle de deux quantités qui est entre -1 et 1) et en attribuant D en conséquence.

D = 1 pour les tangentes entre 0 et 0.05

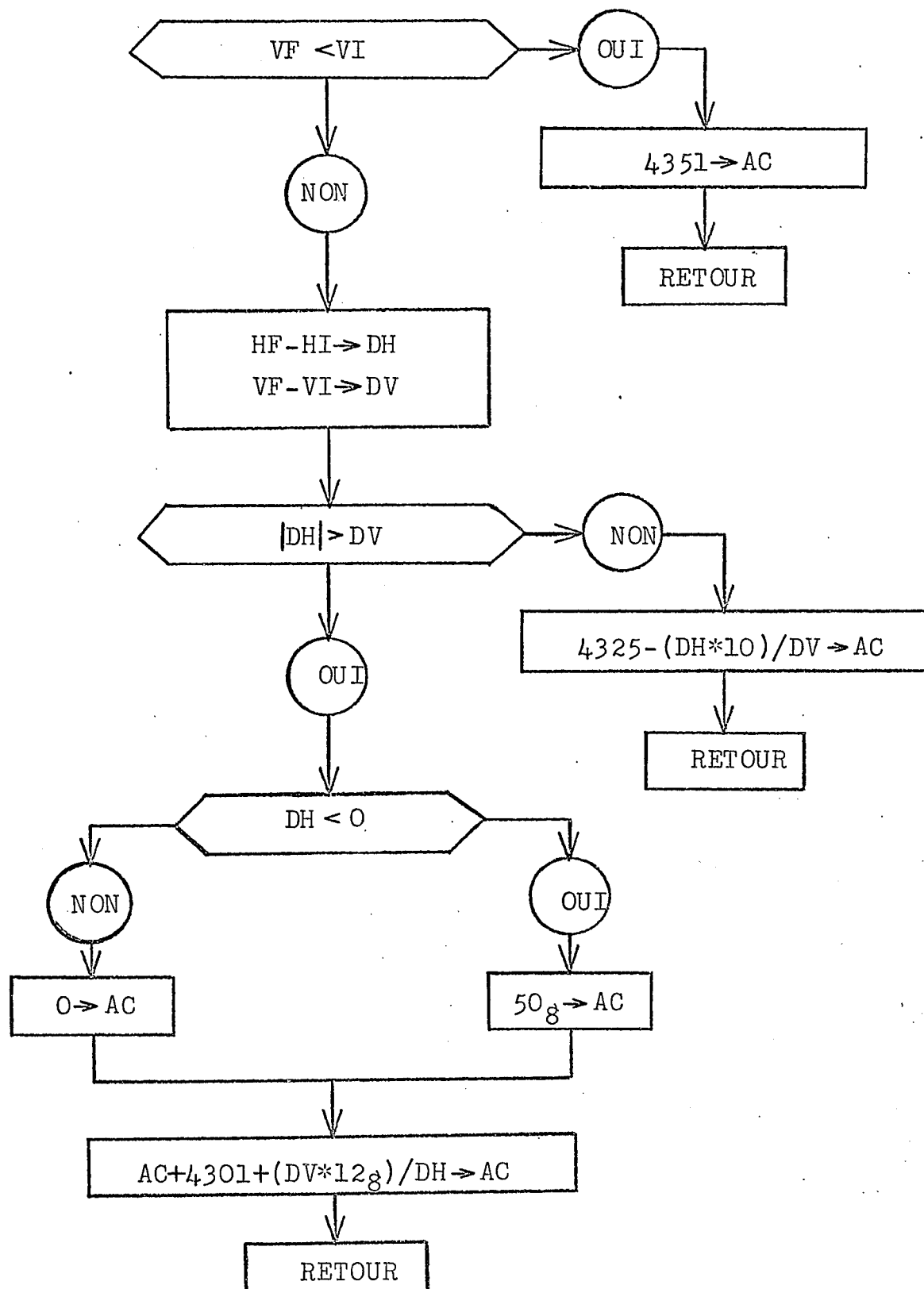
D = 2 pour les tangentes entre 0.05 et 0.15 etc.

D = 12 pour les cotangentes entre 0.85 et 0.95 etc.

SOUS-ROUTINE DMOYEN

SOUS-ROUTINE DMOYEN—VERSION "ASSEMBLER"

7



B) Recherche des segments - Sous-routine CHSEG.

La sous-routine CHSEG a pour but de chercher un bâtonnet qui parte du point initial donné (VI,HI) ou (MOTI, ADBITI), qui ait une orientation D entre DI et DF* et au moins CRITER points noirs sur 10. CRITER, DI, DF et le point initial sont des données qui viennent du programme principal. La sous-routine retourne au programme principal avec la variable logique CAVE = FAUX (FORTRAN) ou à l'adresse de retour normale (ASSEMBLEUR) si aucun segment n'est trouvé. Par contre, si un segment est trouvé, CAVE = VRAI (FORTRAN) ou l'adresse de retour est incrémentée (ASSEMBLEUR).

S'il y a plus d'une direction qui compte au moins CRITER points noirs, la meilleure est choisie, c'est-à-dire, celle le long de laquelle il y a le plus de points noirs ($D = \text{BEST}$). Si, par exemple, il y a 7, 6, 7 points noirs pour les directions 10, 11, 12, $\text{BEST} = 10$.

Une fois décidée l'orientation BEST, la sous-routine SUIVRE regarde les 10 points afin de trouver le dernier point noir. Si, dans la direction BEST, ce sont les 7 premiers points qui sont noirs tandis que les 3 derniers sont blancs, c'est le 7^e qui est retenu comme point final du segment. Ceci permet une reconstitution des traces nettement plus fidèle que l'ancienne version de la sous-routine qui retenait toujours le 10^e point comme étant le point final. Puis, les points de ce segment sont effacés, (sauf le dernier point noir) afin d'éviter que le programme "redécouvre"

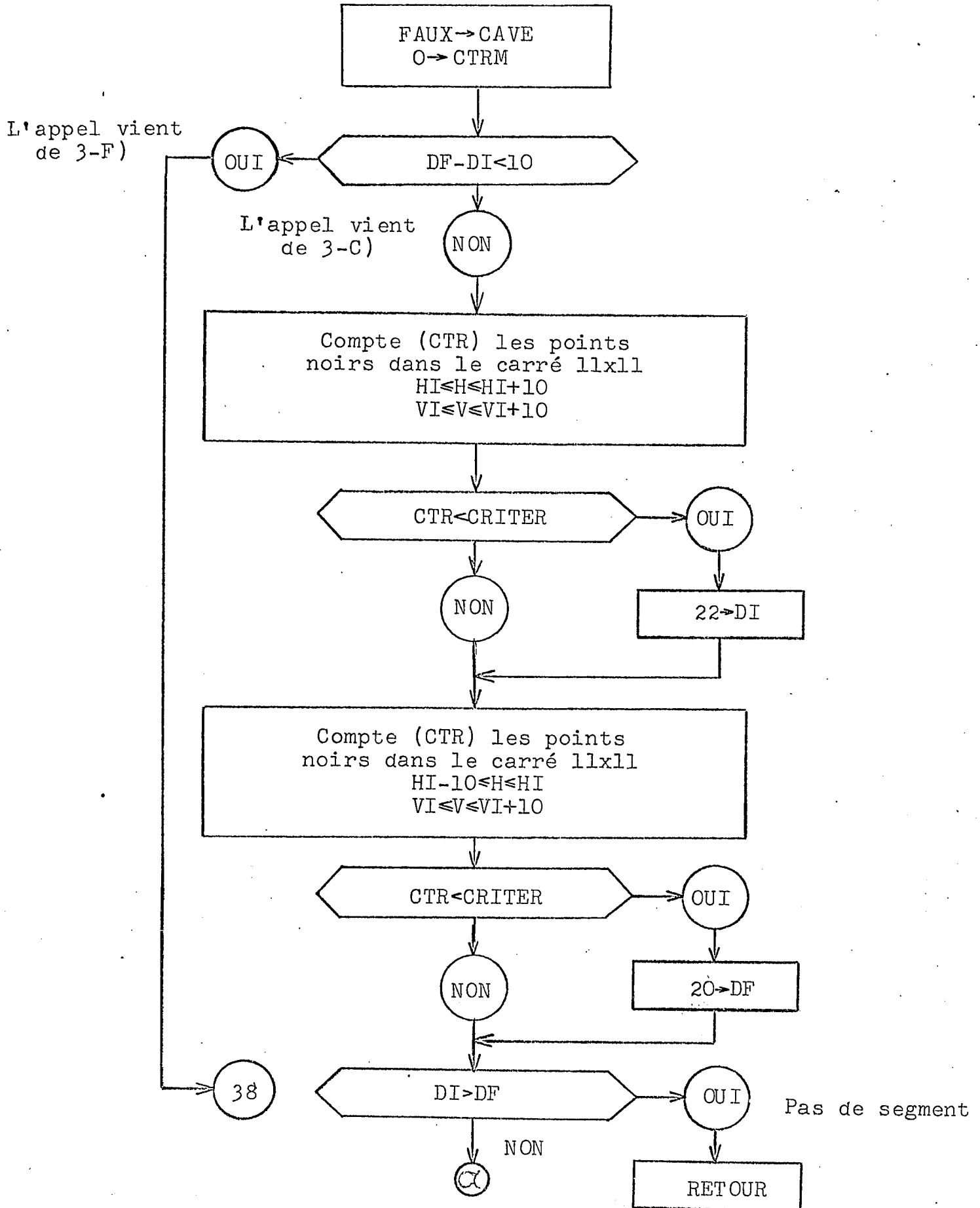
* En fait, comme nous le verrons à la section suivante, deux situations peuvent se présenter:

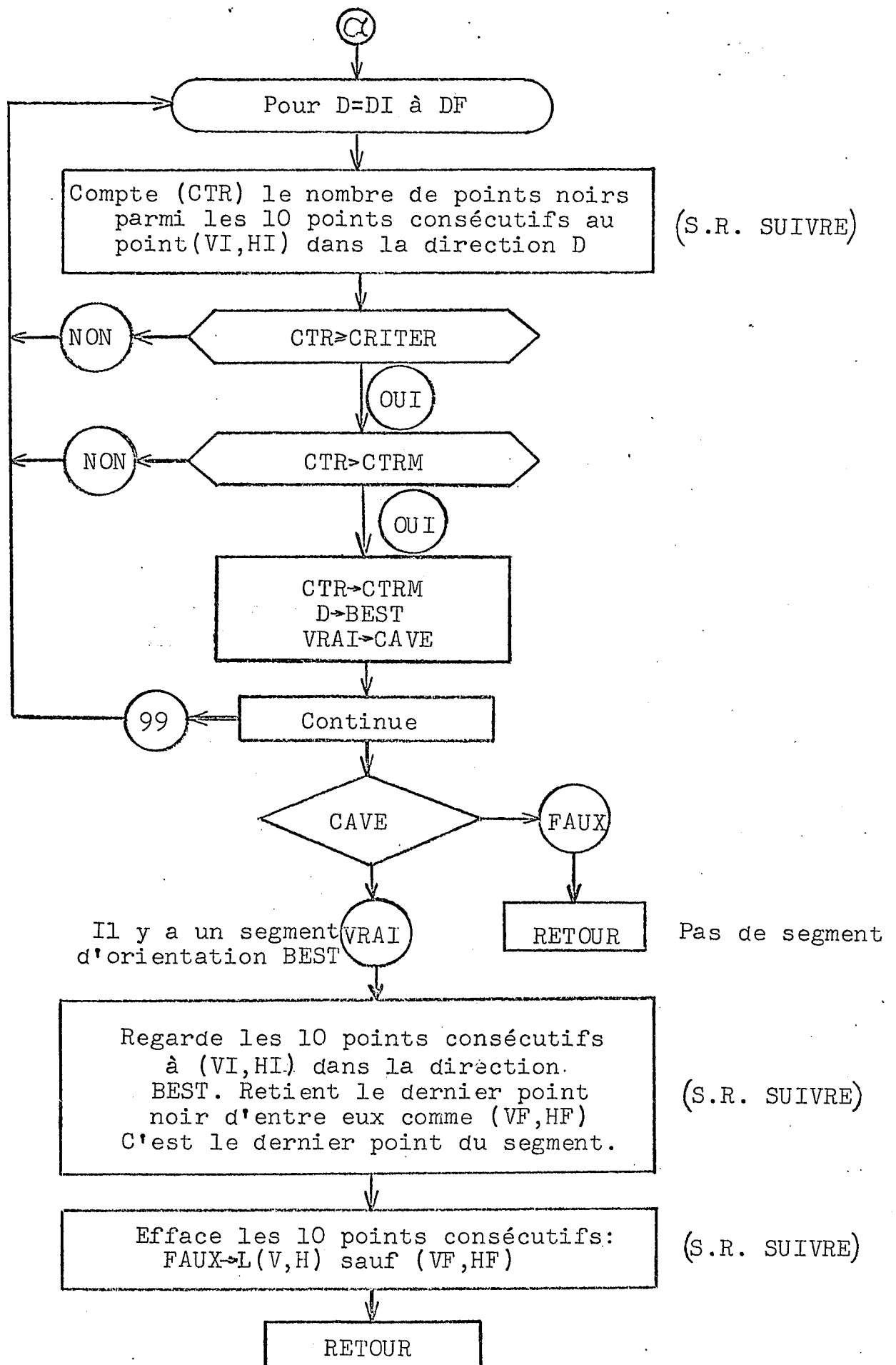
- 1) $DI=1$, $DF = 40$, c'est à dire, recherche dans toutes les directions
- 2) $DF-DI = 4,2$ ou 0 - recherche dans 5,3 ou seulement 1 direction.

69

plusieurs fois la même trace.*

* Nous avons aussi essayé d'effacer en plus les points immédiatement voisins de chacun de ces points, mais ceci brisait trop les traces qui se croisent.

SOUS-ROUTINE CHSEG



Sous-routine CHSEG—Version "ASSEMBLEUR"

Cherche un segment d'au moins CRITER points noirs (sur 10 à partir du point (MOTI,ABITI) ou (VI,HI) dont l'orientation D est entre DI et DF. S'il y en a plusieurs, prends celui qui a le plus de points noirs. (Orientation "BEST")

A)

1 → CTRM

B)

DI < 4301

OUI

4301 → DI

Pour s'assurer de chercher seulement dans les directions permises.

NON

DF > 4351

OUI

4351 → DF

NON

C)

MOTI A7=0
ou
MOTI A7=0

OUI

Retour

Ne cherche pas de segment à partir du premier ou du dernier mot (xxx0 ou xxx7) d'une ligne. (i.e. si HI < 13 ou HI > 84).

NON

D)

MOTI < 2000
ou
MOTI > 3017

OUI

Retour

Ne cherche pas de segment si VI 1 ou VI > 66

NON

E)

DF - DI ≤ 10

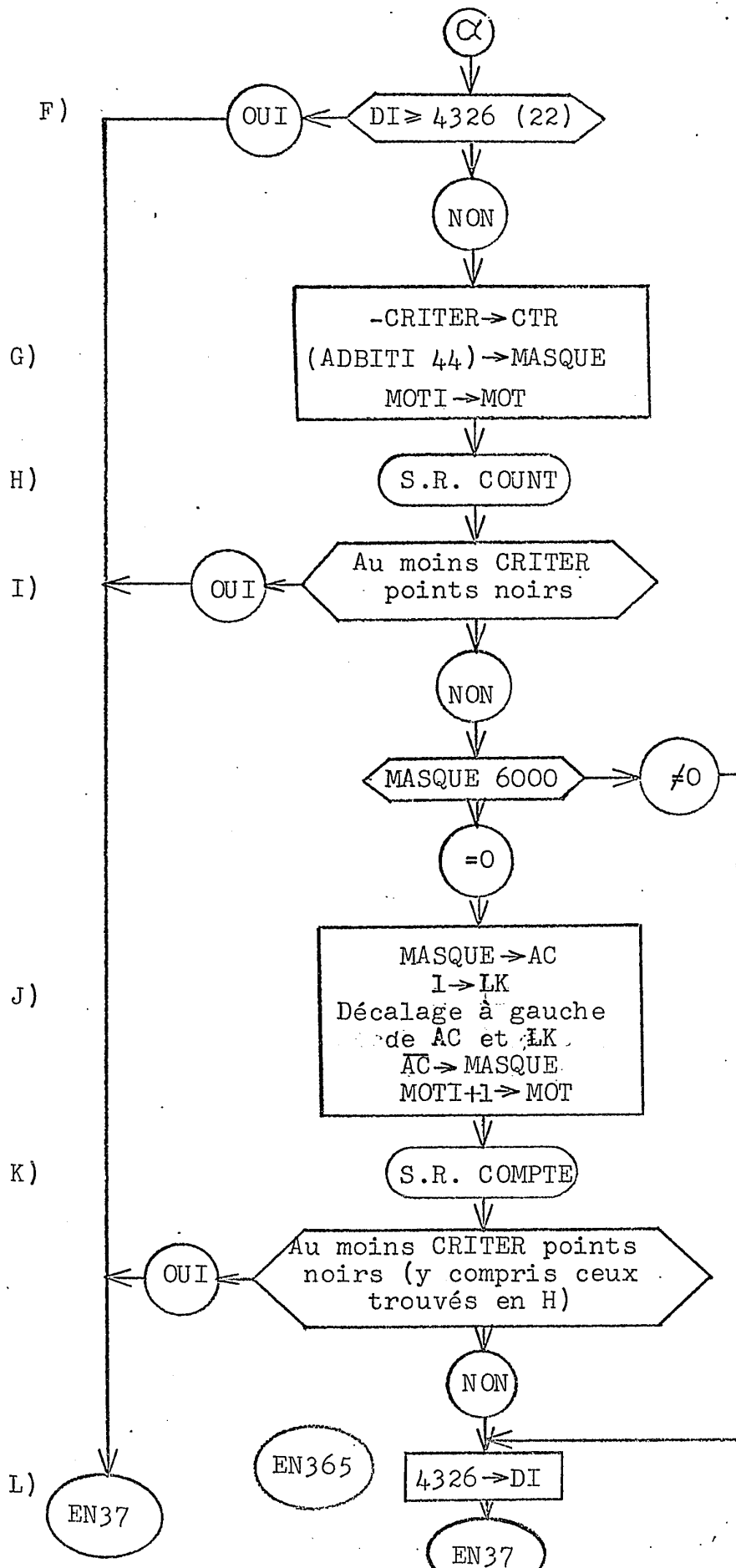
OUI

EN38

Ne compte pas tous les points des carreaux de gauche et de droite s'il n'y a que quelques orientations à examiner.

NON

α

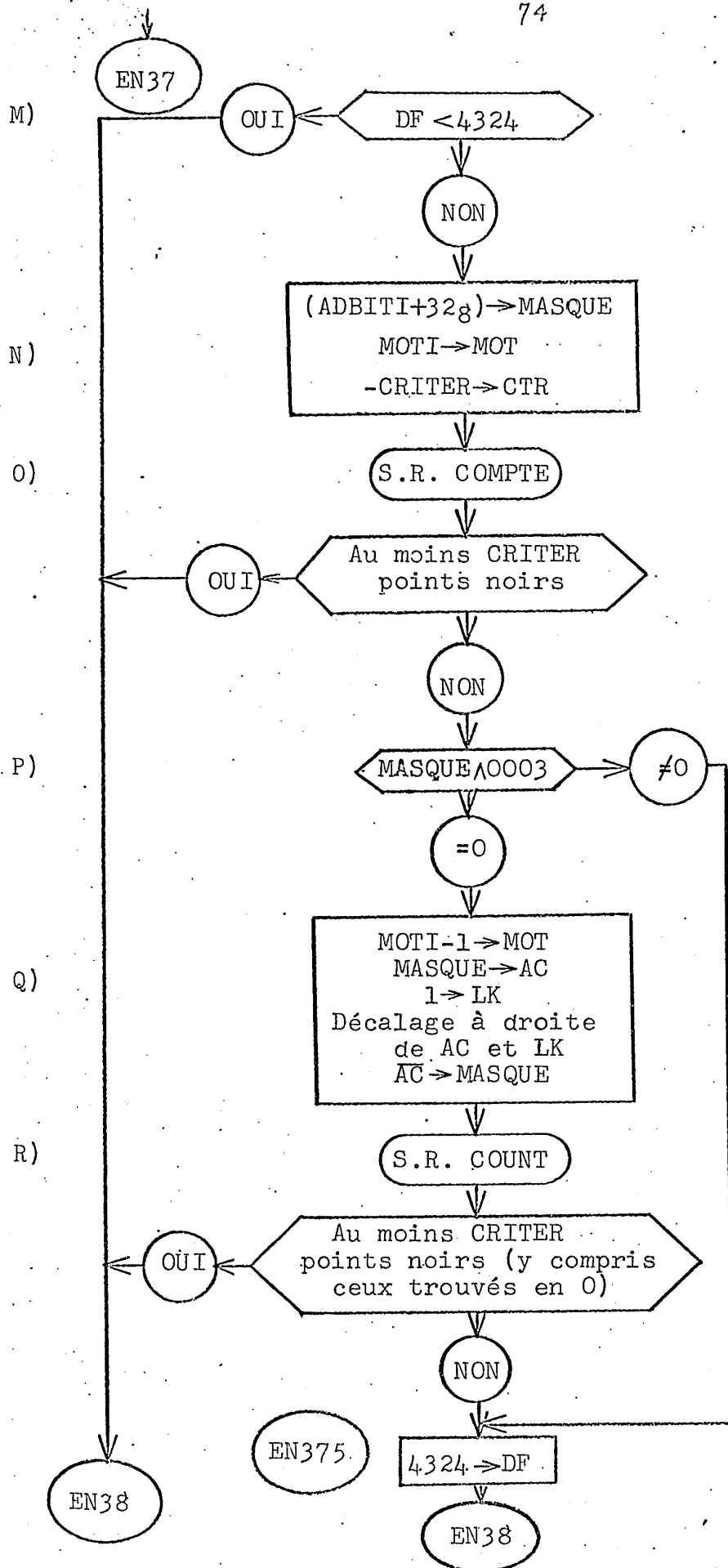


Compte les points noirs dans le carré 11x11 à droite de (VI,HI):
 $VI \leq V \leq VI+11$
 $HI \leq H \leq HI+11$
 Va à EN37 dès que CRITER points noirs ont été trouvés.

Point initial dans le bit 0 ou 1
 Le carré ne comprend aucun bit du mot à droite

Sinon calcule le masque des bits du mot suivant pour compléter le carreau et compte les points noirs dans la tranche de droite du carré.

S'il n'y a pas au moins CRITER points noirs dans le carré de droite, il ne peut y avoir de segment dans ce carreau, i.e. parmi les orientations 1 à 21



Compte les points noirs dans le carré 11x11 à gauche de (VI, HI):

$VI \leq V \leq VI + 11$

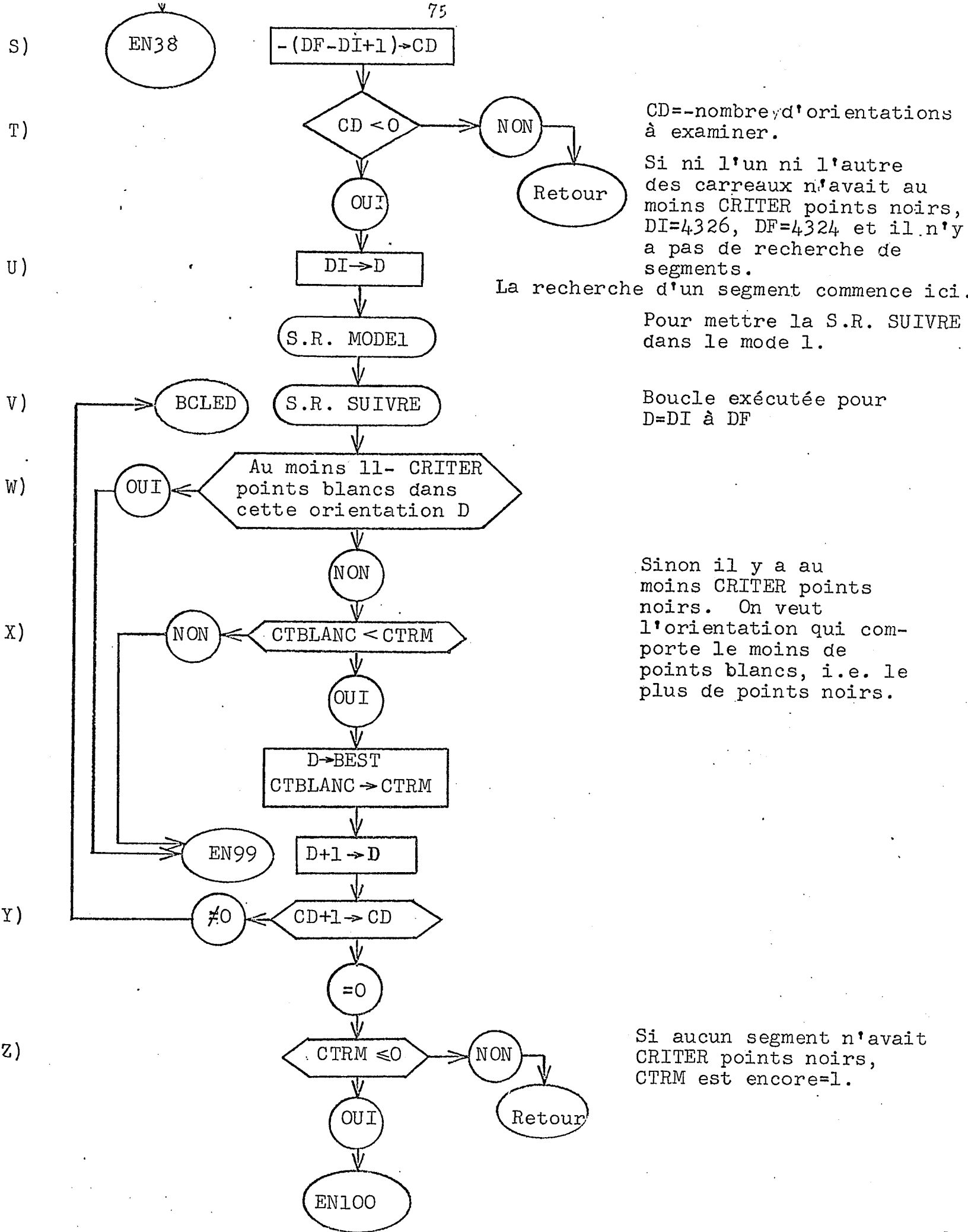
$HI - 11 \leq H \leq HI$

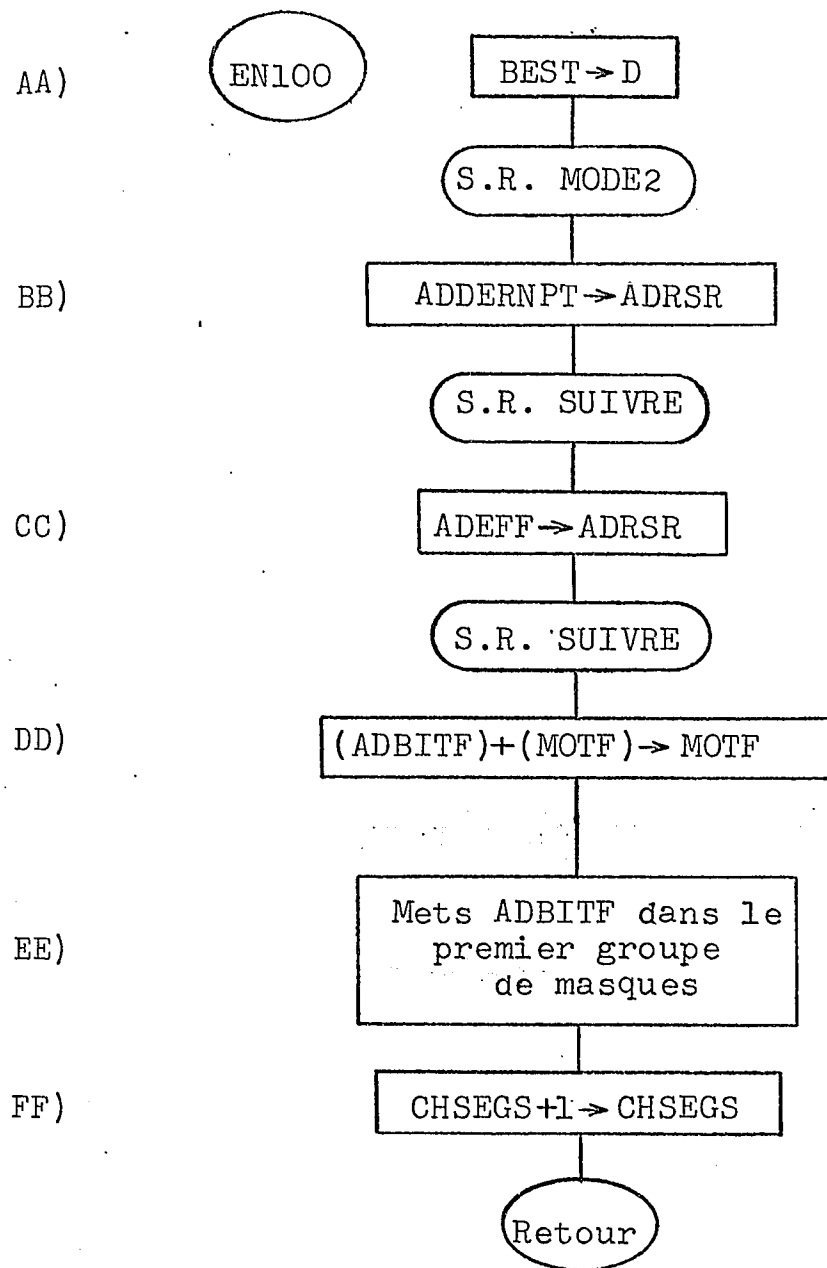
Va à EN38 dès que CRITER points noirs ont été trouvés.

Point initial dans le bit 10 ou 11. Le carré ne comprend aucun bit du mot de gauche.

Sinon calcule le masque des bits du mot précédent pour compléter le carreau et compte les points dans la tranche de gauche du carré.

S'il n'y a pas au moins CRITER points noirs dans le carré de droite, il ne peut y avoir de segment dans ce carreau, i.e. parmi les orientations 21 à 41





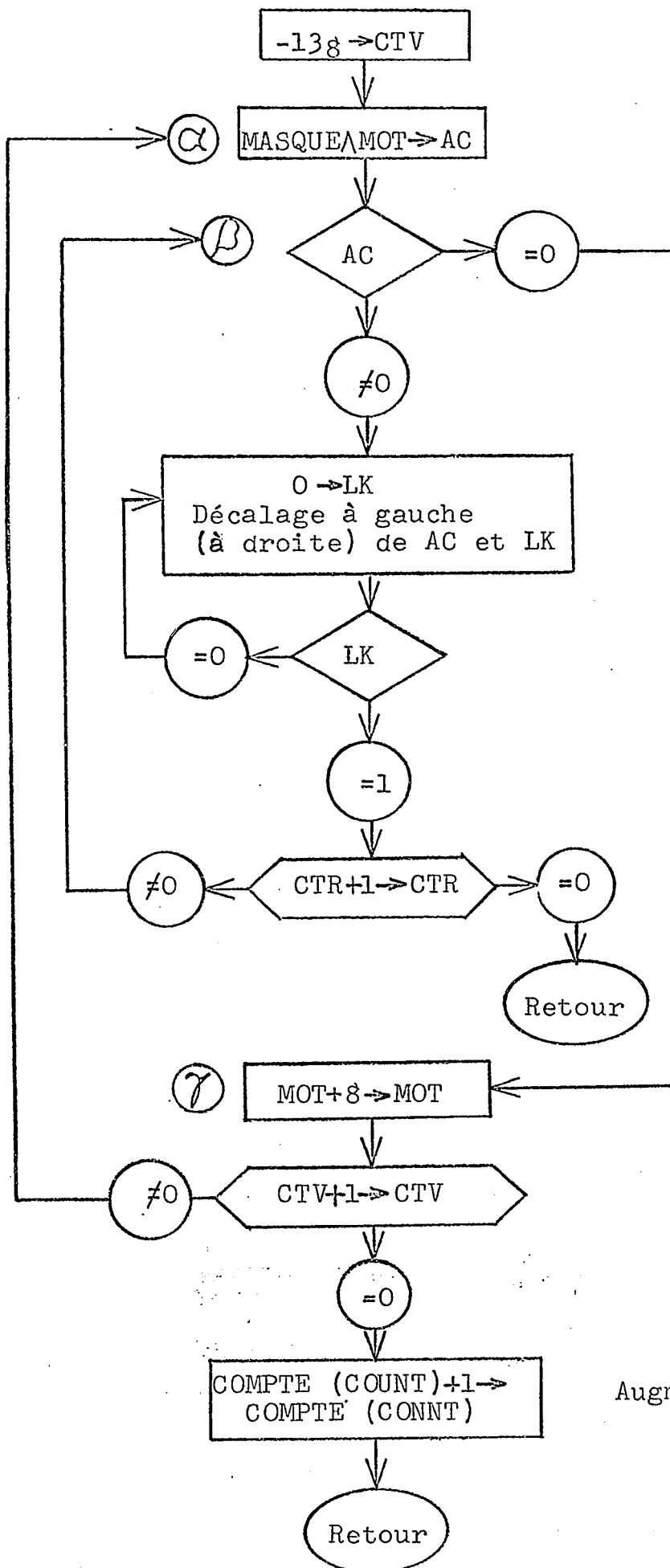
Pour mettre la S.R. SUIVRE dans le mode 2.

Cherche le dernier point noir de l'orientation BEST. Garde le dans MOTF, ADBITF. (La S.R. SUIVRE fait appel à la S.R. DERNPT après avoir calculé la position de chaque point).

Efface tous les points du segment d'orientation BEST. (La S.R. SUIVRE fait appel à la S.R. EFF après avoir calculé la position de chaque point) Mais on garde le dernier point noir

S.R. CHSYAL

Augmente l'adresse de retour de 1, ce qui indique qu'un segment a été trouvé.

S.R. COMPTE (et COUNT)

Pour sauver du temps,
utilise COMPTE ou COUNT
selon que les bits du
masque sont plutôt à
gauche ou plutôt à
droite.

Augmente l'adresse de retour

C) Recherche des traces

1) Ancienne version

Dans l'ancienne version du programme, on recherchait seulement des bâtonnets de 10 points de longueur.

Les points noirs étaient examinés tour à tour pour voir s'ils étaient à l'origine d'un bâtonnet. Si oui, il était mis en réserve. Ces bâtonnets étaient associés en traces par un programme d'assemblage des traces (semblable à l'étape 6 du programme actuel). Il n'y avait pas de "prolongement des traces" non plus. (Organigramme 11).

Ce programme prenait beaucoup de temps, et donnait de mauvais résultats dans les situations un peu complexe - par exemple le point de rencontre de quelques traces.

2) Nouvelle version

Devant l'insuccès relatif de la première méthode, il a fallu refaire le programme. La nouvelle version profite de l'information acquise dès qu'un segment est trouvé pour en chercher un autre au-delà (organigramme 12).

Le premier bâtonnet (segment) est trouvé en regardant dans toutes les directions à partir d'un point noir donné (C), mais les suivants sont trouvés en n'examinant que quelques directions (F).

Tous les points noirs (VI, HI) qui sont dans les limites:

$$1 \leq VI \leq 66 \quad ; \quad 13 \leq HI \leq 84$$

peuvent être le point initial d'une trace. Ces limites nous assurent qu'en cherchant un segment, le programme ne cherchera pas

en dehors des limites de l'image.*

Pour qu'un point noir puisse être à l'origine d'une autre trace, il doit être à une distance suffisante (plus que 2) de toute autre trace déjà trouvée. Pour cela, il est fait appel à la sous-routine (EVPROX" (éviter la proximité) avant que ne commence l'analyse.** S'il y a une trace trop près de (VI,HI), le programme passe au prochain point noir (retour à (B) dans l'organigramme 12).

Cette précaution s'est avérée utile, d'abord parce que certaines traces très denses et larges avaient tendance à donner une trace fantôme tout près, (malgré l'effaçage effectué par la sous-routine CHSEG quand un segment est trouvé), et de plus, nous avons constaté que ceci réduisait de beaucoup le temps consacré à la recherche des traces, en évitant d'en chercher d'autres à partir des quelques points que l'effaçage avait laissés noirs.

Si le point noir est acceptable, la sous-routine CHSEG cherche un segment partant de là, et ce dans toutes les directions (D = 1 à 40). S'il y a un segment, ses coordonnées (point initial, point final, orientation) sont celles de la nou-

* Dans le programme du PDP 8/E, ces points noirs sont recherchés par le programme principal, en utilisant la technique décrite en III -E2), tout en prenant soin de sauter le premier et le dernier mot de chaque ligne. Dès qu'un point noir est trouvé, il est fait appel à la sous-routine YAUNPT pour voir si ce point est à l'origine d'une trace.

** Voir l'organigramme 14. Dans le programme FORTRAN, c'est la variable logique "CAVE" qui indique si une trace proche a été trouvée ou non. Dans le programme en langage d'assemblée, l'adresse de retour est augmentée de 1 s'il n'y a pas de trace proche du point.

velle trace (organigramme 12, (E)).

A partir de ce point final, et de quelques autres points près de celui-ci, la sous-routine CHSEG cherche un autre segment. S'il y en a un, son point final devient le point final de la trace, l'orientation "BEST" est recalculée (sous-routine DMOYEN), et un troisième segment est cherché, et ainsi de suite, (F, G et H).

Précisons ce que nous entendons par "près du point final". Les positions (VII,HII) du point final, et des 5 positions suivantes dans la direction "D" de la trace (calculées par l'algorithme d'interpolation décrit à la section A de ce chapitre) sont utilisées. Si la trace est plutôt horizontale, les 5 points examinés comme origine possible d'un autre segment sont, dans l'ordre, pour chaque (VII,HII);

- i) (VII,HII)
- ii) (VII+1,HII)
- iii) (VII-1,HII)
- iv) (VII+2,HII)
- v) (VII-2,HII)

Si la trace est plutôt verticale, les cinq points correspondants sont;

- i) (VII,HII)
- ii) (VII,HII+1)
- iii) (VII,HII-1)
- iv) (VII,HII-2)
- v) (VII,HII+2)

Un segment est cherché à partir de ceux de ces 30 (6 x 5) points qui sont noirs seulement. Très souvent, un segment est trouvé à partir du premier de ceux-ci, c'est à dire le point final du segment précédent, et alors, il n'est pas nécessaire de chercher à partir des 29 autres. Par contre, si aucun segment ne part de ce dernier, le suivant est essayé, et ainsi de suite.

Dans le programme FORTRAN, les positions de ces 30 points sont calculées au fur et à mesure qu'elles sont requises, tandis que dans le programme en langage d'assemblage, ces positions sont toutes mises en mémoire au moyen des sous-routines SUIVRE et DEPH ou DEPV (selon que la trace est plutôt horizontale ou plutôt verticale - cf. organigrammes 15 et 16).

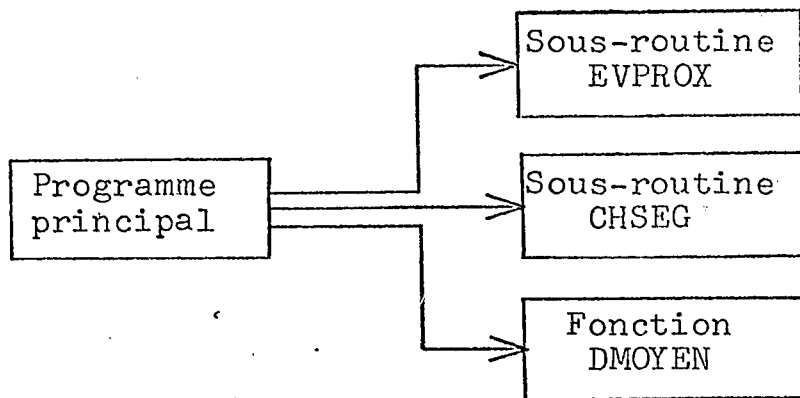
Enfin, "presque la même orientation" signifie une tolérance de ± 2 sur "D" pour un 2^e segment de trace, ± 1 pour un 3^e, 4^e ou 5^e etc. Cette tolérance est réduite de 1 lorsque le point de départ est éloigné de 2 unités de la trace extrapolée (points (iv) et (v) des deux listes ci-dessus).

Le nombre de points noirs (sur 10) exigés pour un premier segment est généralement plus élevé que pour les suivants. Ces critères (ISN 0086 et ISN 0141 dans le programme FORTRAN; NOICRITER et NO2CRITER, mots 0107 et 0110, pour le PDP 8/E) peuvent être changés par l'utilisateur selon les besoins. L'effet d'une variation de ces critères est très important (comparer, par exemple, les figures 20 et 21, CRITER = 6 et 5 signifie qu'on exige 6/10 pour le premier segment, et 5/10 pour les suivants*).

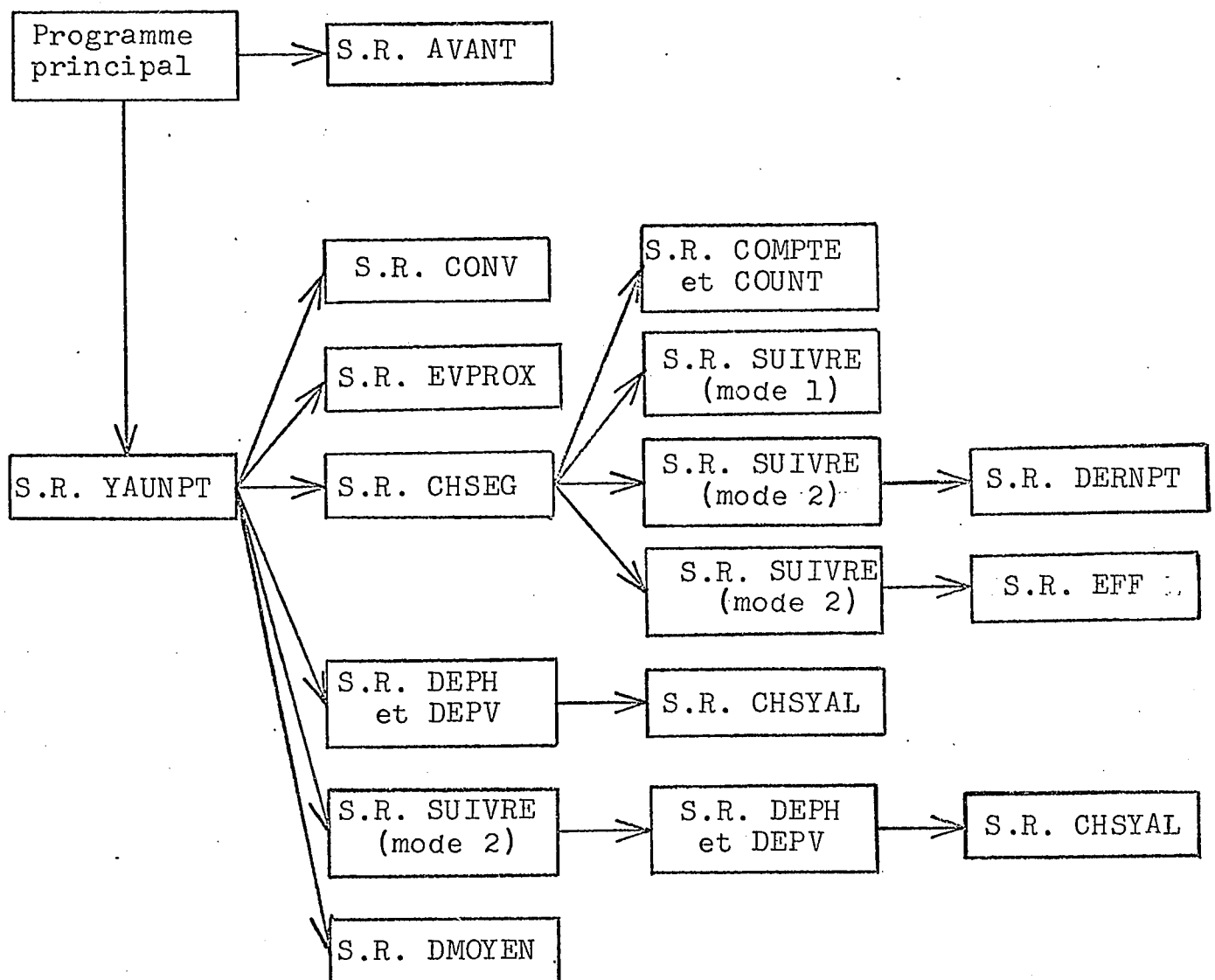
* En fait, ces critères sont plutôt 7/11 et 6/11 puisqu'on exige aussi que le point initial soit noir.

Bien que plus complexe, cette façon de procéder s'est avérée à la fois rapide et sûre. En profitant de l'information acquise sur un segment pour chercher le suivant, on réussit à la fois, à réduire le temps d'exécution (en ne cherchant que dans 5, 3 ou 1 directions) et à améliorer le résultat, puisque, dans les régions confuses, les longues traces sont trouvées et enlevées avant que ne soit entreprise la recherche des autres. (voir, par exemple, la trace 1 dans la figure 18).

Structure du programme de recherche des traces



Structure du programme FORTRAN



Structure d'appel du programme ASSEMBLEUR.

3-RECHERCHE DES TRACES (ANCIENNE VERSION)

A)

Faire pour tous les points noirs:
VIP=1 à 66, HIP=13 à 84

B)

Prochain
point noir

Soit le point noir A(VIP,HIP)

C)

Regarder dans toutes les
directions (D=1 à 40) les 10
points consécutifs à partir de A

Sous-routine
CHSEG

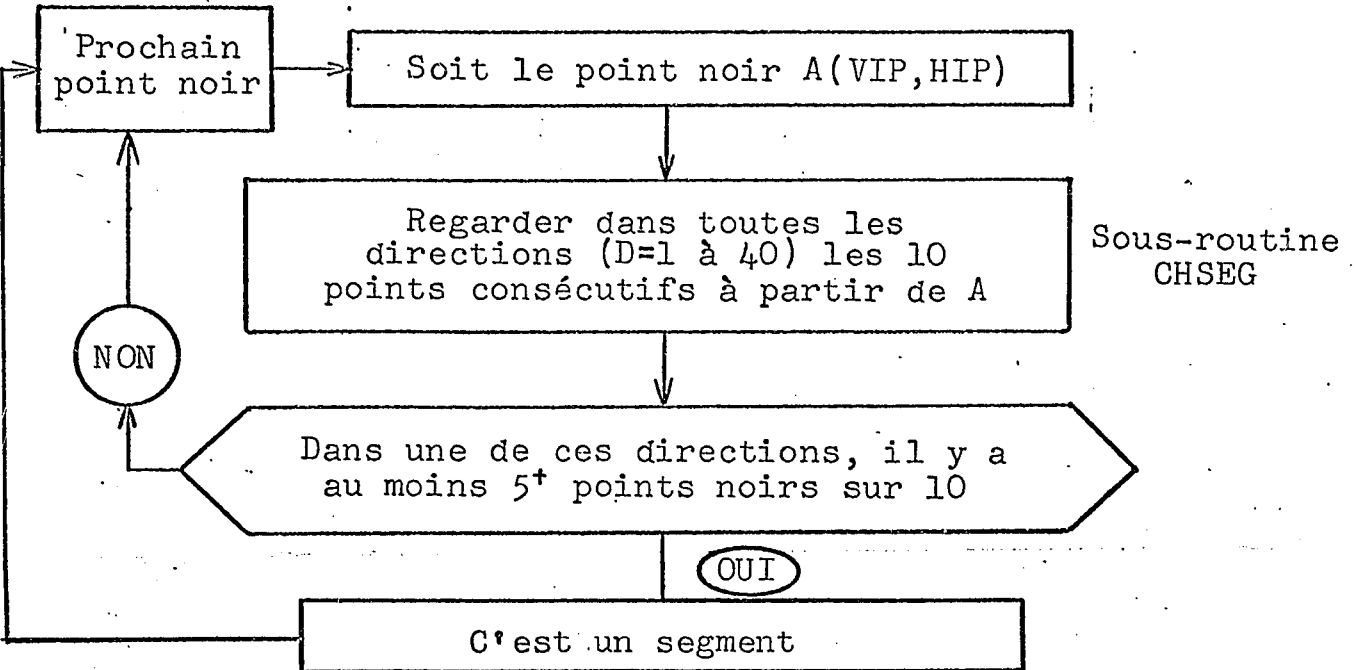
D)

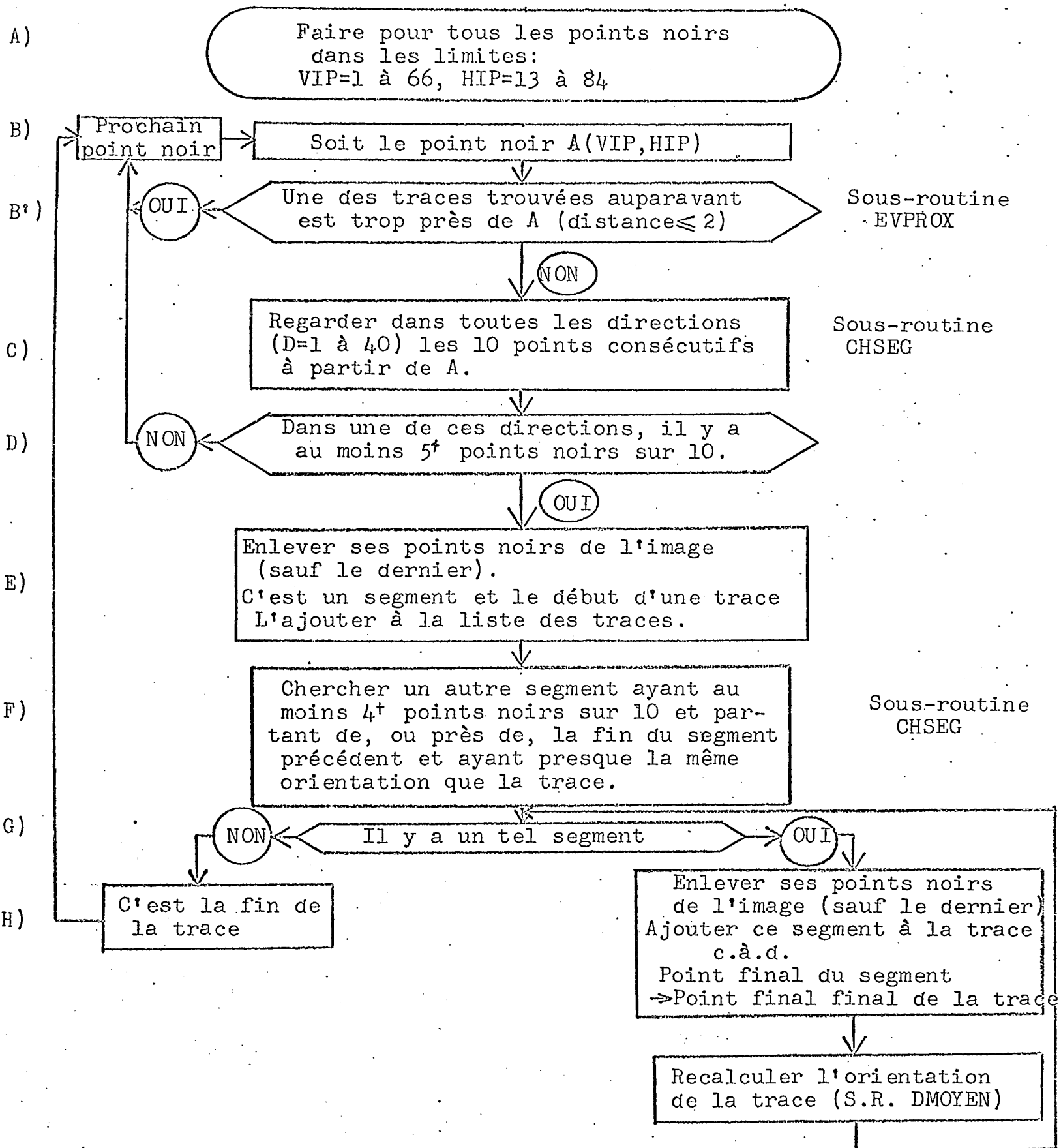
NON

Dans une de ces directions, il y a
au moins 5⁺ points noirs sur 10

OUI

C'est un segment

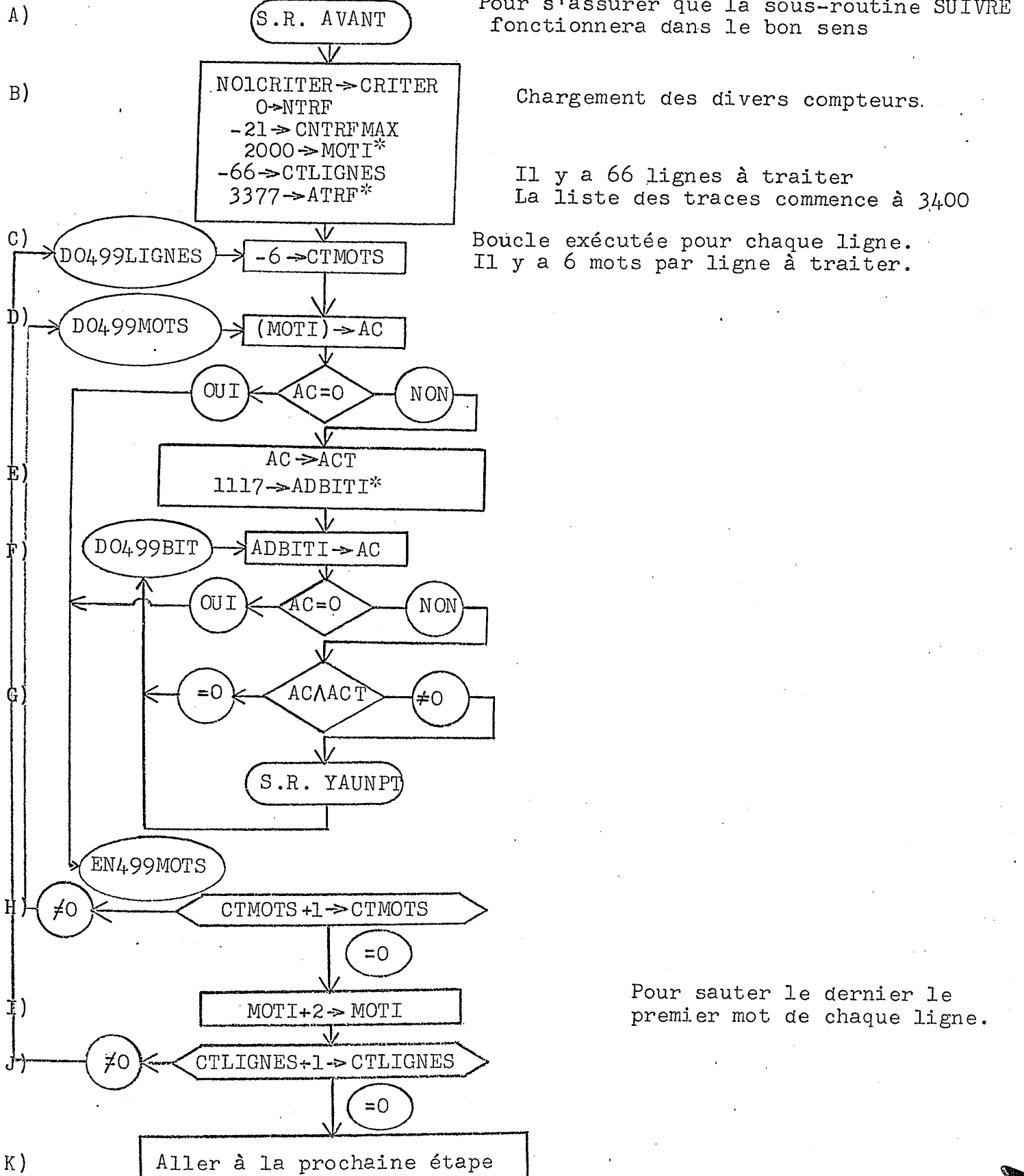


3-RECHERCHE DES TRACES

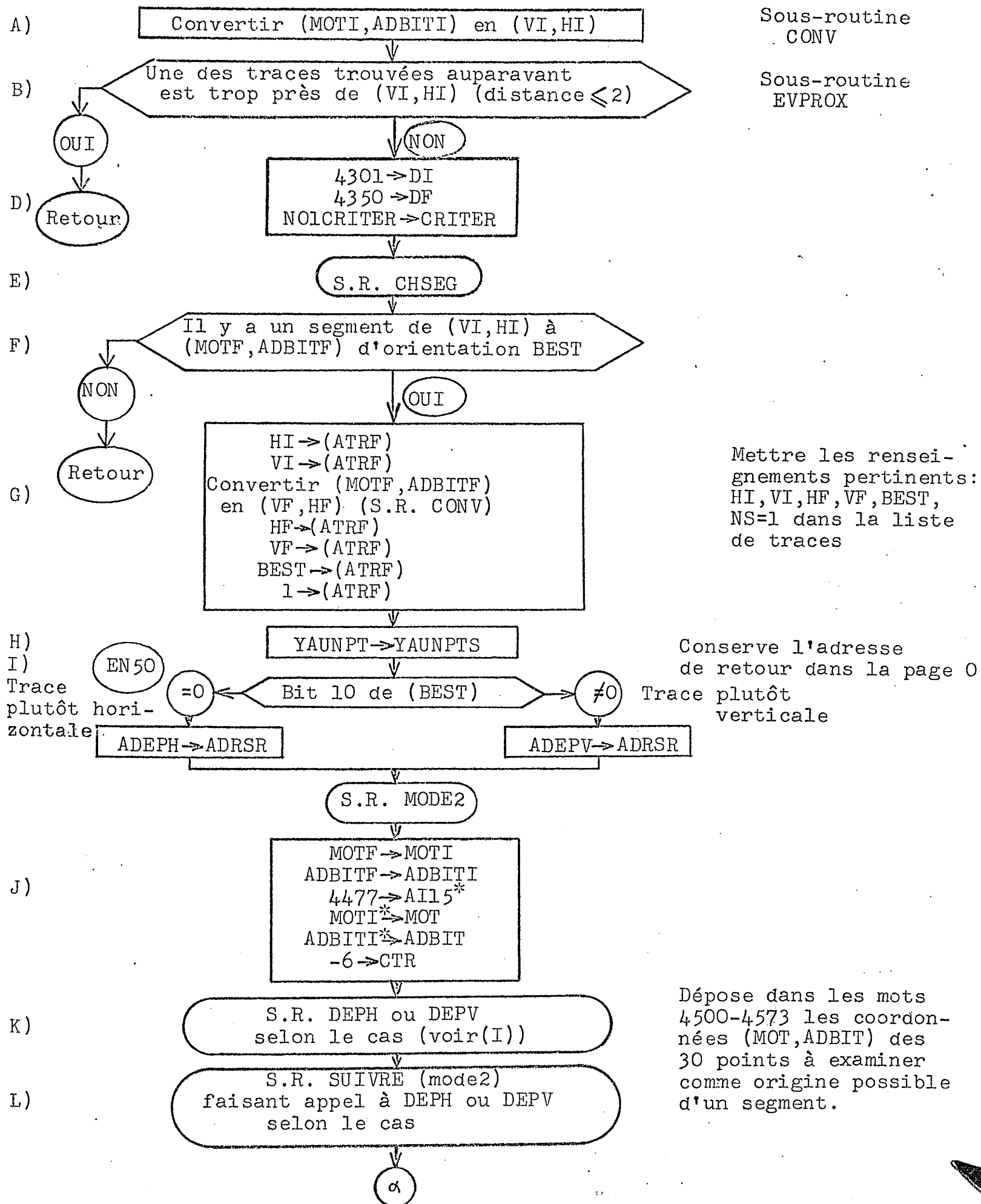
† Ces critères sont ajustables.

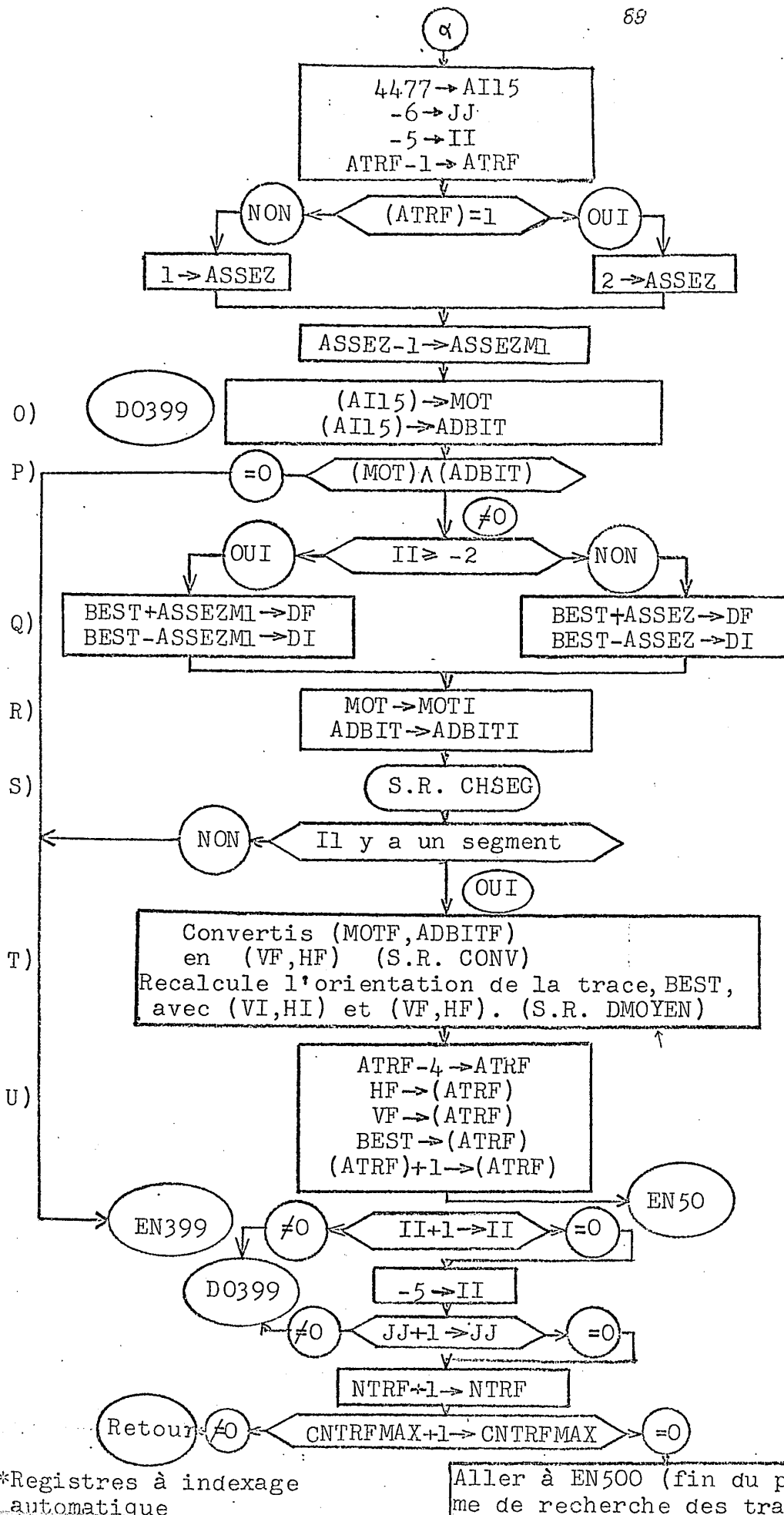
Programme principal (adresse de début:1710)

DEBUT (1710)



*Registres à indexage automatique.





NS=1?

Essaie les 30 points
comme origine possible
d'un nouveau segment.

Tolérance moins grande
sur la pente pour les
points initiaux plus
éloignés de la trace.

Dans la liste de traces,
le point final du segment
devient celui de la trace
L'orientation recalculée
remplace l'ancienne et
NS est augmenté de 1.

Si on arrive ici, c'est
qu'il n'y a plus de seg-
ment à ajouter à la trace
Si la trace est la 21^e
cesse d'en cher-
cher d'autres.

*Registres à indexage
automatique

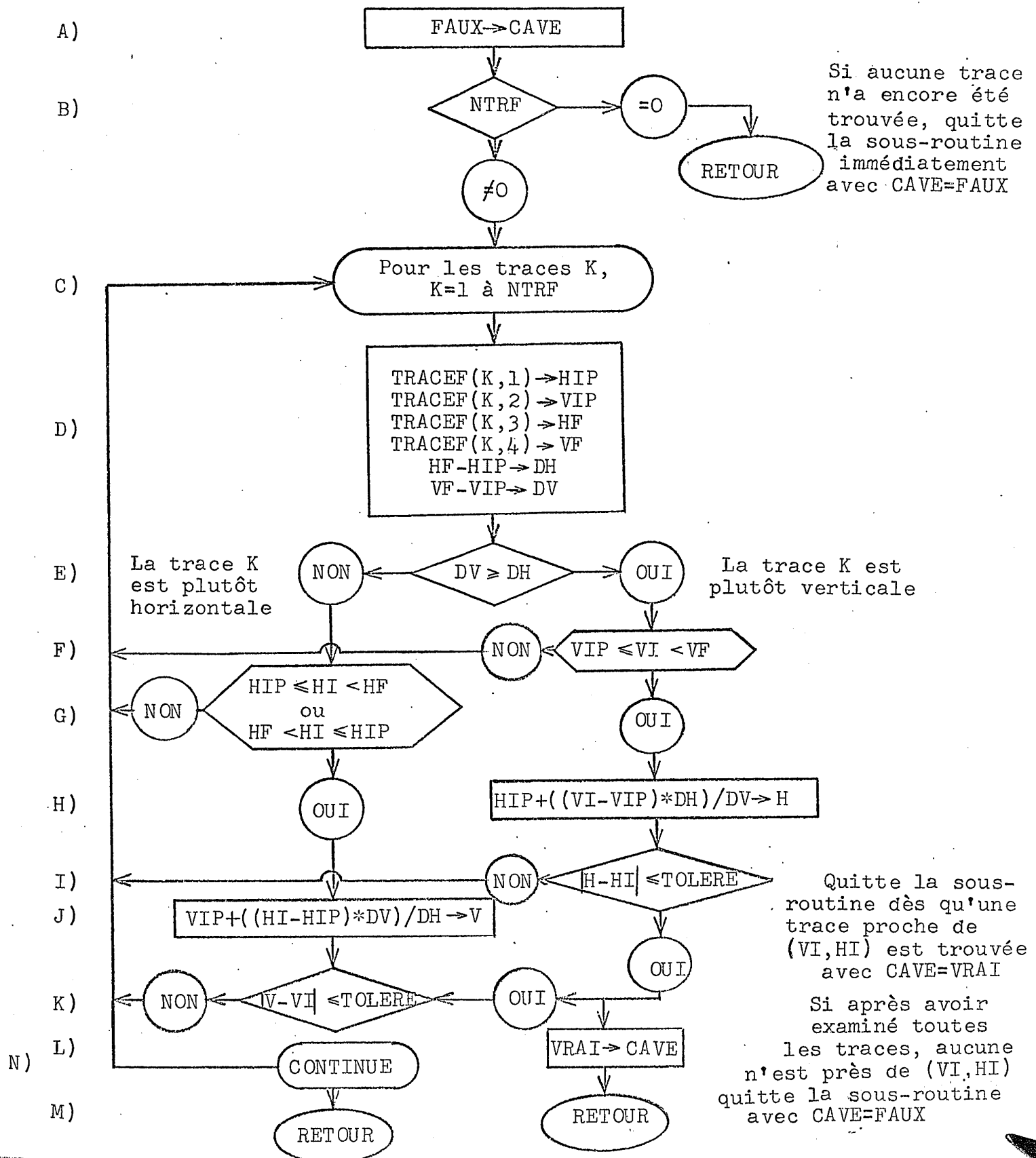
Aller à EN500 (fin du program-
me de recherche des traces)

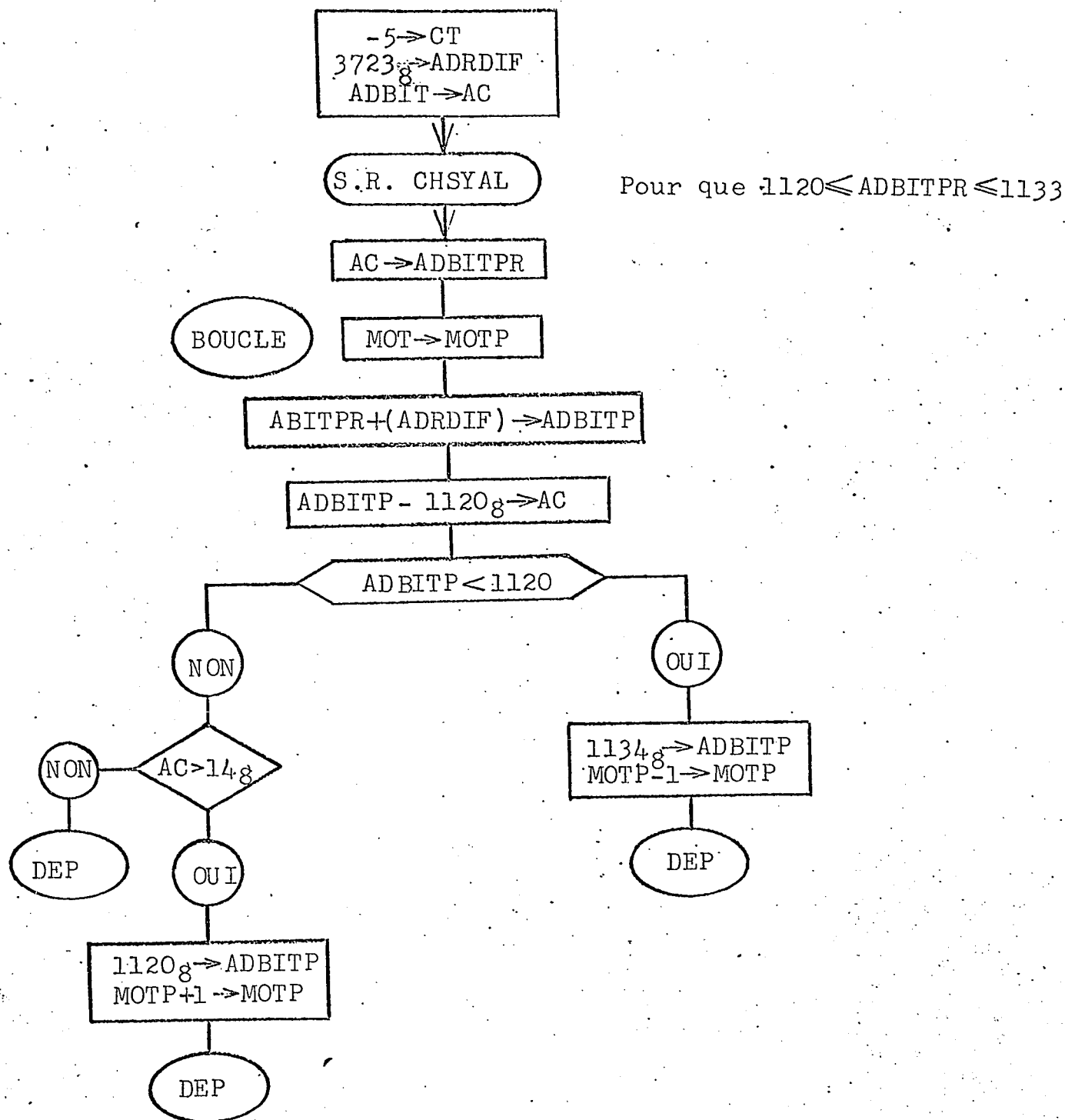
14

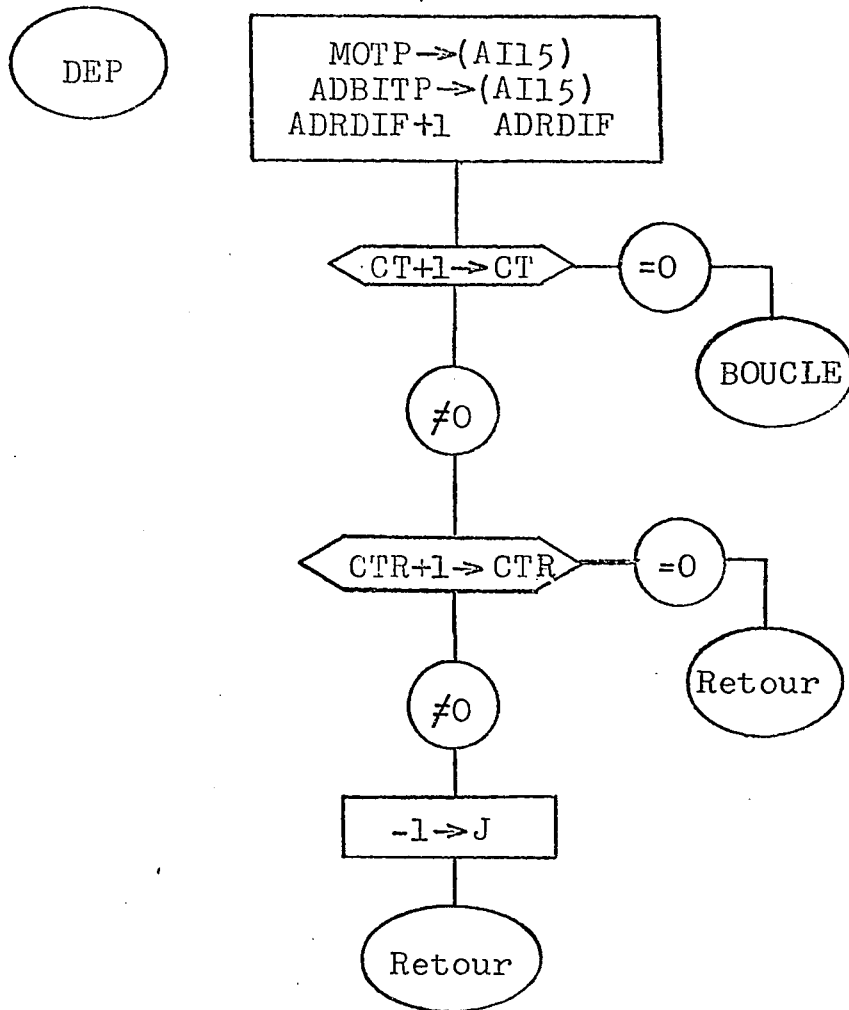
SOUS-ROUTINE EVPROX

Vérifie si le point (VI,HI) est près (distance $\leq$ TOLERE, où TOLERE=2) d'une trace déjà trouvée.

Si oui, la variable logique CAVE est VRAI au retour de la sous-routine.
Sinon, la variable logique CAVE est FAUX au retour de la sous-routine.

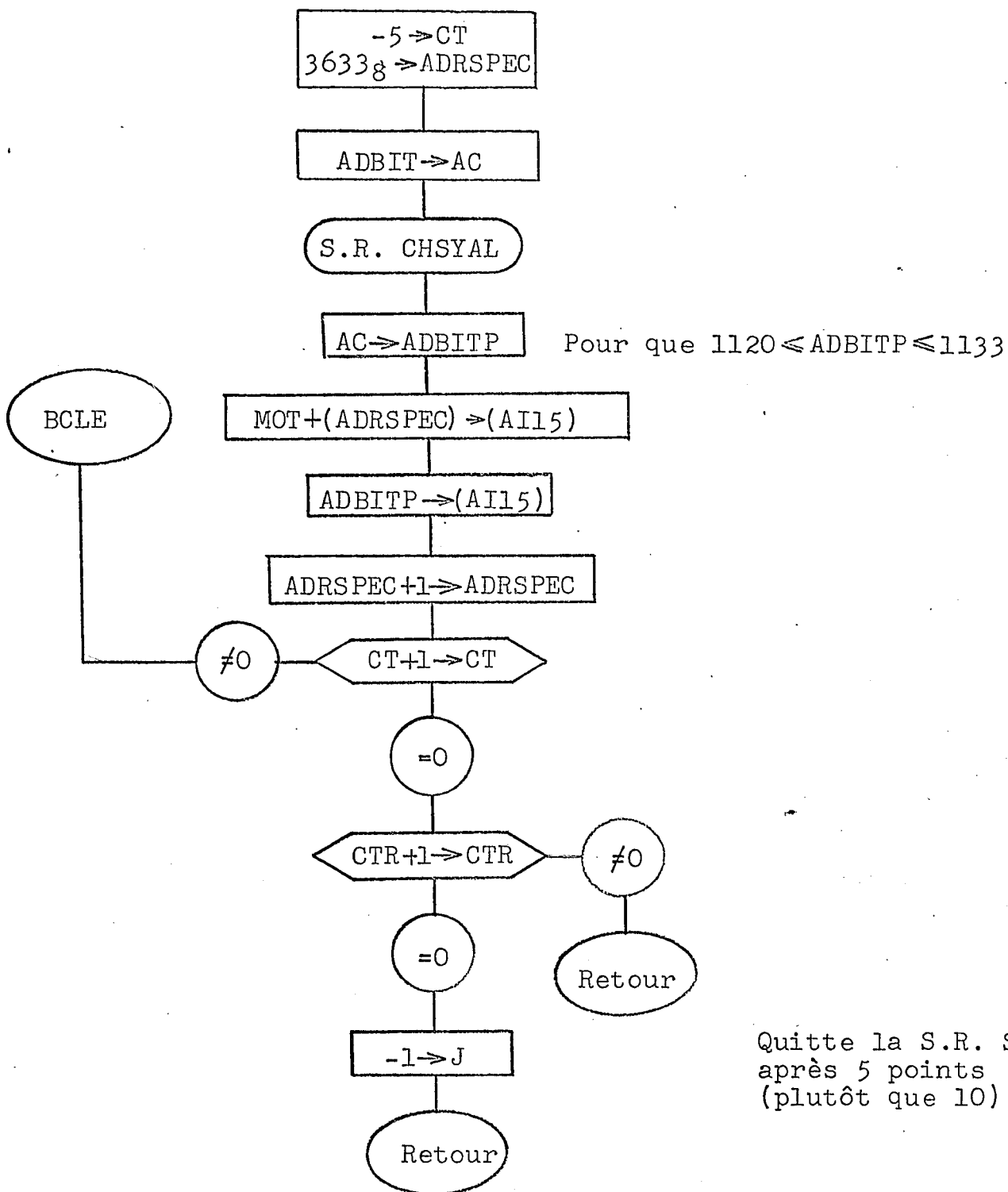


S.R. DEPV



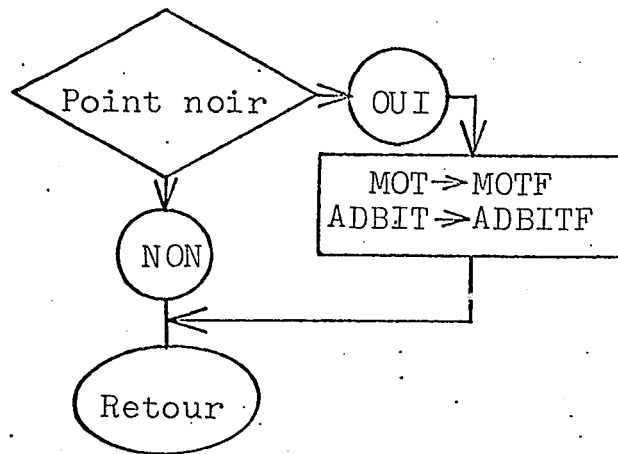
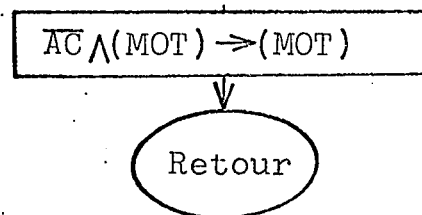
Quitte la S.R. SUIVRE
après 5 points
(plutôt que 10).

*3723
DIF1,0
DIF2,1
DIF3,7777
DIF4,2
DIF5,7776



Quitte la S.R. SUIVRE
après 5 points
(plutôt que 10)

*3633
SPEC1, 0
SPEC2, 10
SPEC3, 7770
SPEC4, 20
SPEC5, 7760

S.R. DERNPTS.R. EFF

# DE TPACE	HI	VI	HF	VF	D	# DE SEGMENTS	# DE PTS
1	20	1	43	66	17	7	55
2	73	1	86	31	17	3	27
3	35	2	34	41	21	4	33
4	32	44	20	69	26	3	25
5	54	49	57	58	18	1	6
6	43	66	45	76	19	1	9

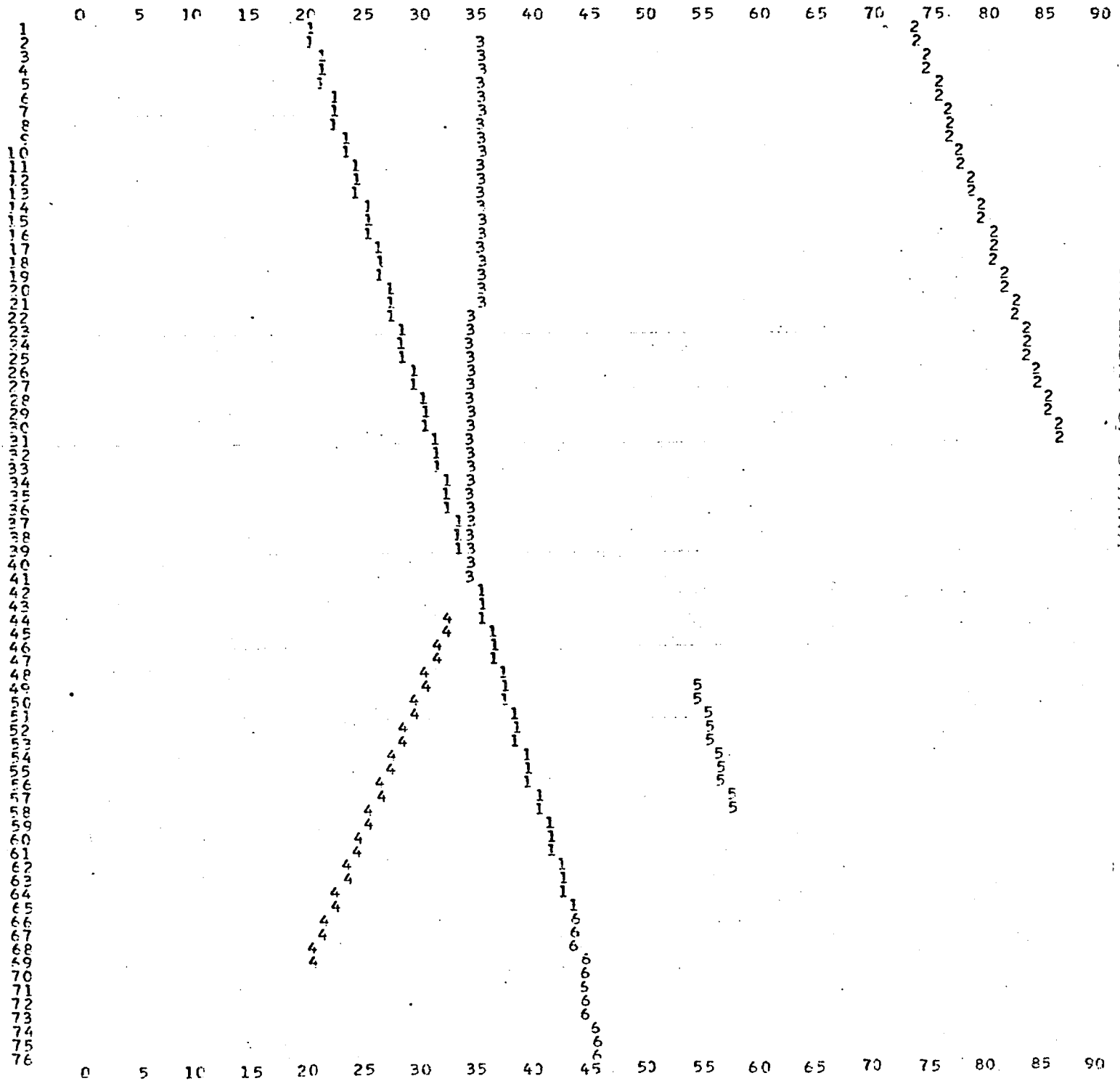


Figure 18 - Etoile no. 1 après l'étape de recherche des traces.

(CRITER: 6 et 5)

VI- Prolongement et assemblage des traces.

A) Prolongement des traces

Cette étape est un complément, une correction des précédentes. Les points qui appartiennent aux traces mais qui avaient été manqués à l'étape 3, sont ajoutés aux traces afin que celles-ci représentent plus exactement l'image véritable.

Pour ce faire, le programme recherche, à partir du point final et, près de celui-ci (organigramme no.18 , (C), (D)) dans la direction des traces et des directions adjacentes ((B),(E)), d'autres points noirs au delà du point final.* Cette recherche continue tant que trois points blancs consécutifs n'ont pas été trouvés (K). Les points noirs sont comptés (L) et le dernier point noir du meilleur prolongement (celui qui compte le plus de points noirs, parmi les 15 combinaisons différentes d'orientation et de point de départ) devient le nouveau point final de la trace ((N),-(R)). L'orientation de la trace est alors recalculée (R).

Il s'est avéré utile de donner un "boni" de 3 points noirs afin de favoriser le prolongement ayant la même orientation que la trace et partant du bout de celle-ci (M). Ceci favorise la recherche de prolongements qui sont bien dans le sens de la trace sans exclure totalement des points qui pourraient être un peu à

*Le calcul des positions des points dans la direction D et partant du point de départ donné s'effectue selon la méthode décrite au chapitre V, section A. Quand le programme arrive au 10e point, il utilise ce dernier comme nouveau point initial et peut calculer les positions des 10 points suivants, et ainsi de suite.

côté pour une raison ou une autre. Une recherche de prolongement en deça du point initial est aussi effectuée pour chaque trace ("prolongement à rebours").

Dans les deux sens, le programme procède presque de la même façon, mais il y a certaines différences.

- a) Ne cherche pas plus loin que 5 points du point initial.
- b) Dans le calcul des positions des points noirs, la sous-routine SUIVRE fonctionne à rebours (sens inverse).

Dans le programme FORTRAN, "H-SG→H" et V-1→V" remplacent "H+SG→H" et "V+1→V".

- c) Le point initial remplace le point final. Donc. TRACEF(K,1) et TRACEF(K,2) tiennent le rôle que tenaient TRACEF(K,3) et TRACEF(K,4) dans le prolongement au-delà du point final.

Dans le programme en langage d'assemblée, l'adressage à l'intérieur du tableau des traces doit tenir compte de cette différence.

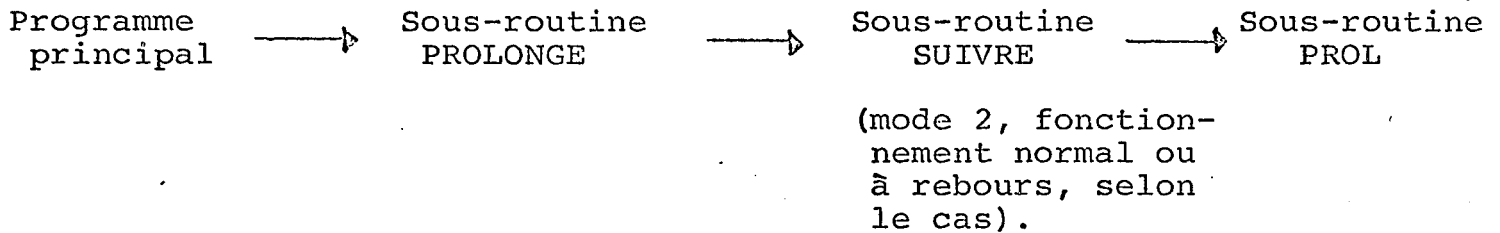
Dans le programme FORTRAN, les deux parties de prolongement sont séparées (ISN0170 - 0244 et ISN 0245-0317). Comme ces deux parties sont très semblables, aucun organigramme n'a été dessiné.

Dans le programme en langage d'assemblée, pour sauver de l'espace, c'est la sous-routine "PROLONGE" qui remplit les deux fonctions. Le minuscule programme principal* fait appel à cette sous-routine deux fois:

*Mots 5160-5175

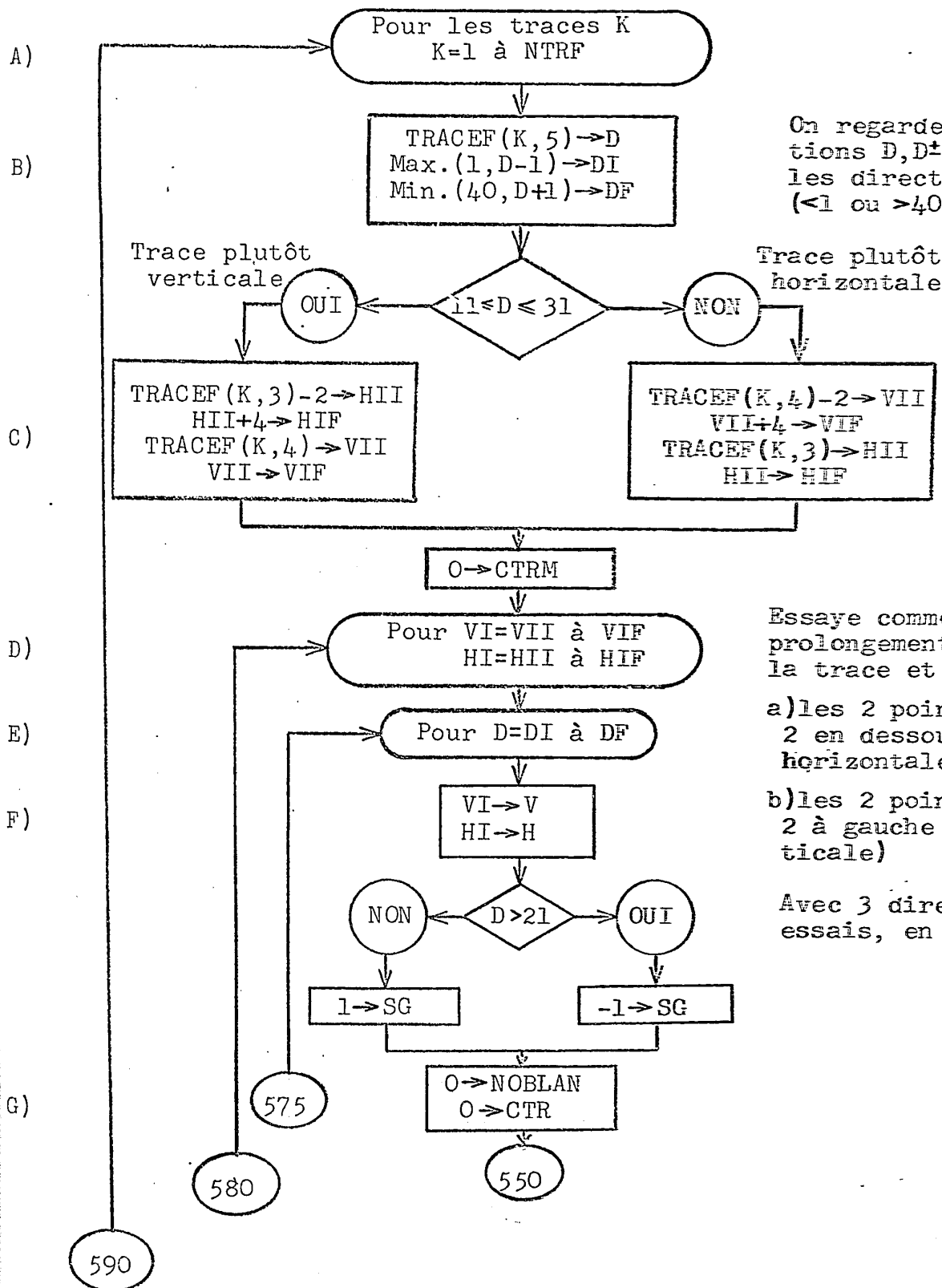
- 1) Pour le prolongement au delà du point final, avec l'indicateur "REBOURS" = 0 et la sous-routine "SUIVRE" dans son état normal
- 2) Pour le prolongement en deça du point initial avec "REBOURS" = 1 et la sous-routine "SUIVRE" fonctionnant à rebours.

La structure d'appel est donc



On peut voir l'effet du prolongement en comparant les figures 18 et 19.

5-PROLONGEMENT DES TRACES



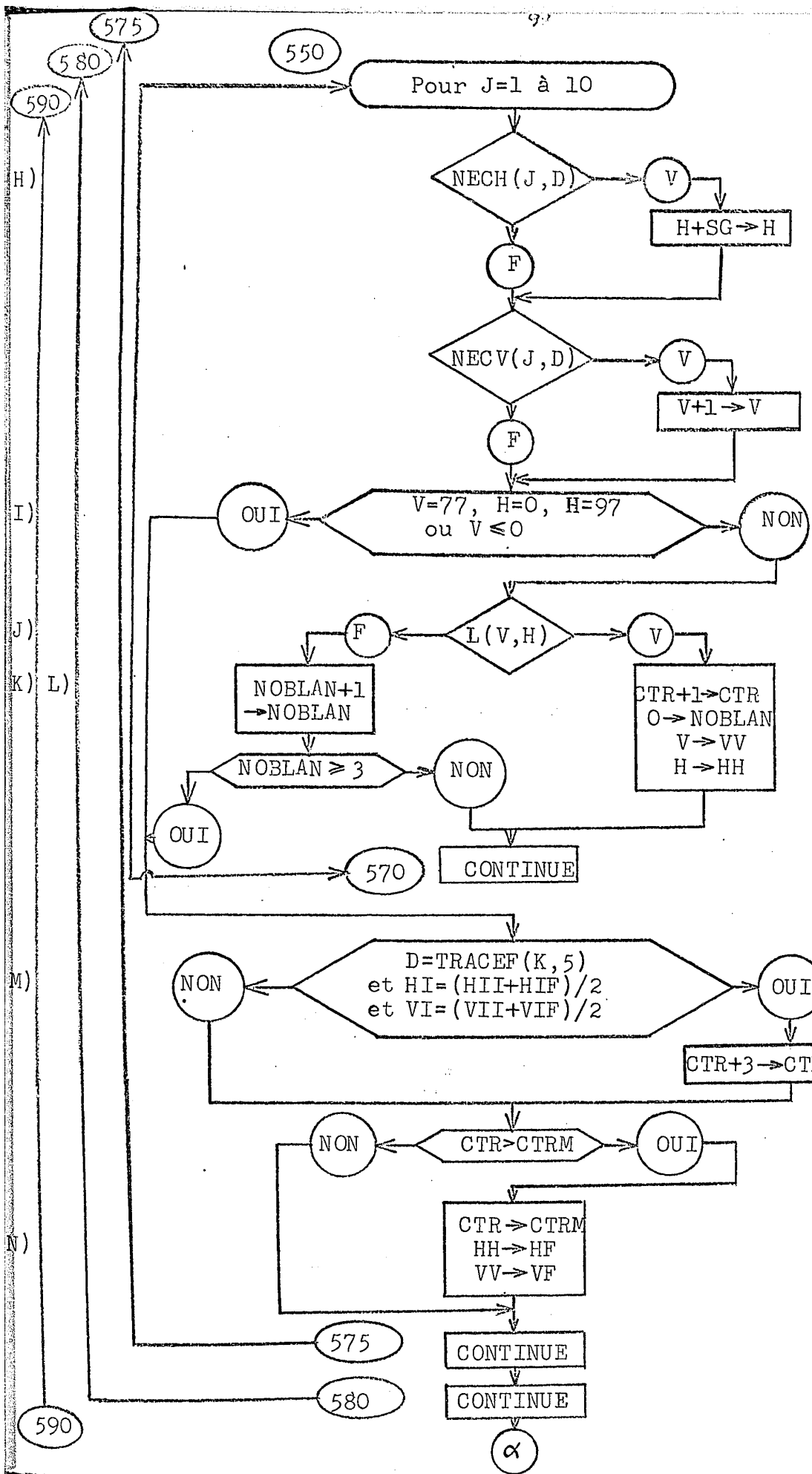
On regardera dans les directions $D, D \pm 1$, mais on évitera les directions impossibles (< 1 ou > 40).

Essaye comme point initial du prolongement le point final de la trace et

a) les 2 points au dessus et les 2 en dessous (trace plutôt horizontale)

b) les 2 points à droite et les 2 à gauche (trace plutôt verticale)

Avec 3 directions, ça fait 15 essais, en tout.



Calcule les positions (V,H) des points au-delà de la fin de la trace, à partir du point (VI,HI), dans la direction D.

Pour ne pas sortir du cadre de l'image.

(V,H) est-il un point noir?
CTR: nombre de points noirs
(VV,HH): coordonnées du dernier point noir.
NOBLAN: nombre de point blancs depuis le dernier point noir. Dès qu'il atteint 3, cesse de regarder dans cette direction.

Accorde un boni de 3 à CTR si l'orientation est celle de la trace et le point de départ est exactement le bout de la trace.

Choisis le prolongement ayant le plus grand nombre de points noirs (CTR maximum).

Conserve le dernier point noir de celui-ci, qui deviendra le nouveau point final de la trace.

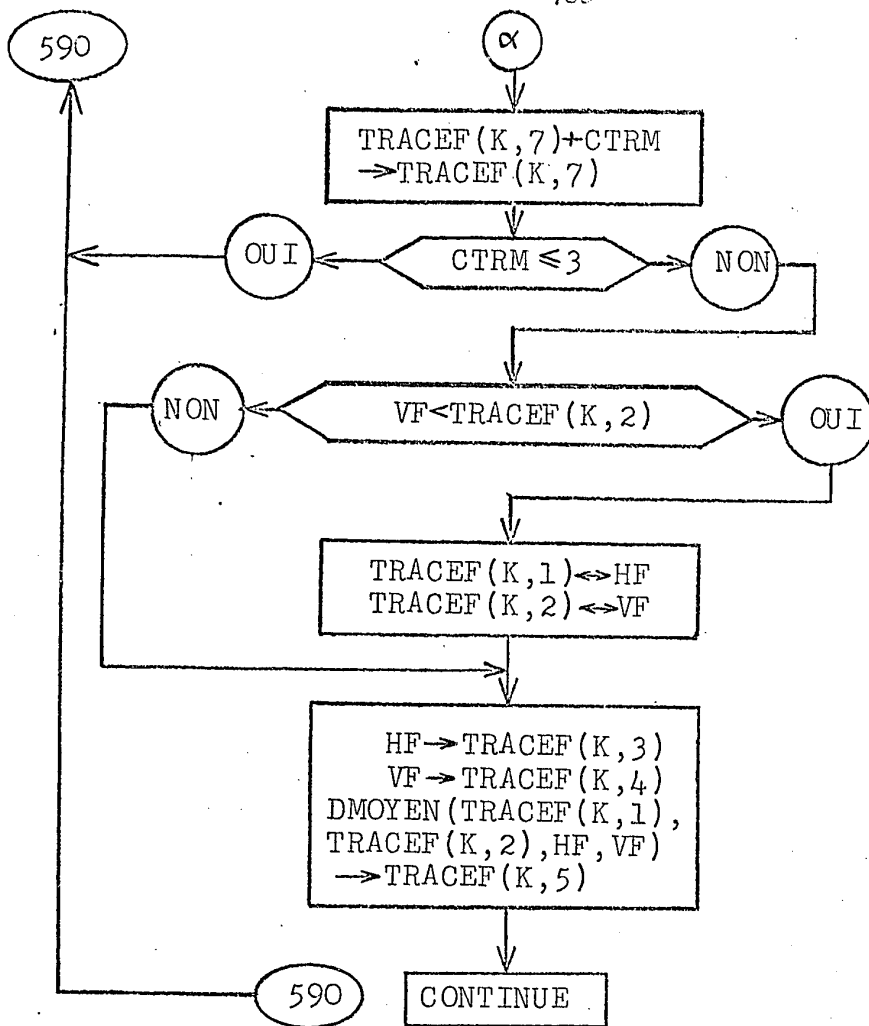
O)

P)

Q)

R)

S)



Ne change rien si aucun prolongement valable n'a été trouvé.

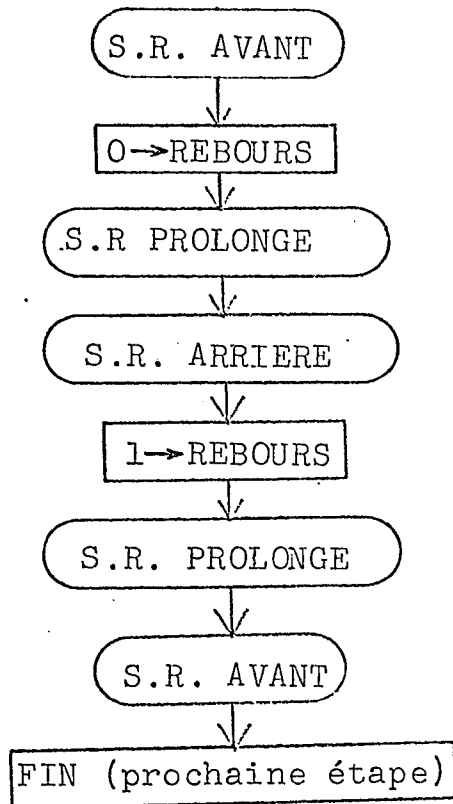
Si le nouveau point final est plus haut que le point initial, échange les.

Le dernier point noir du meilleur prolongement devient le point final de la trace.

Recalcule l'orientation de la trace.

5- Prolongement des traces-Version ASSEMBLEUR

19

Programme principalDébut

Prolonge au delà du point final.

Pour que la S.R. SUIVRE fonctionne dans le sens normal.

Prolonge en deça du point initial.

Pour que la S.R. SUIVRE fonctionne à rebours.

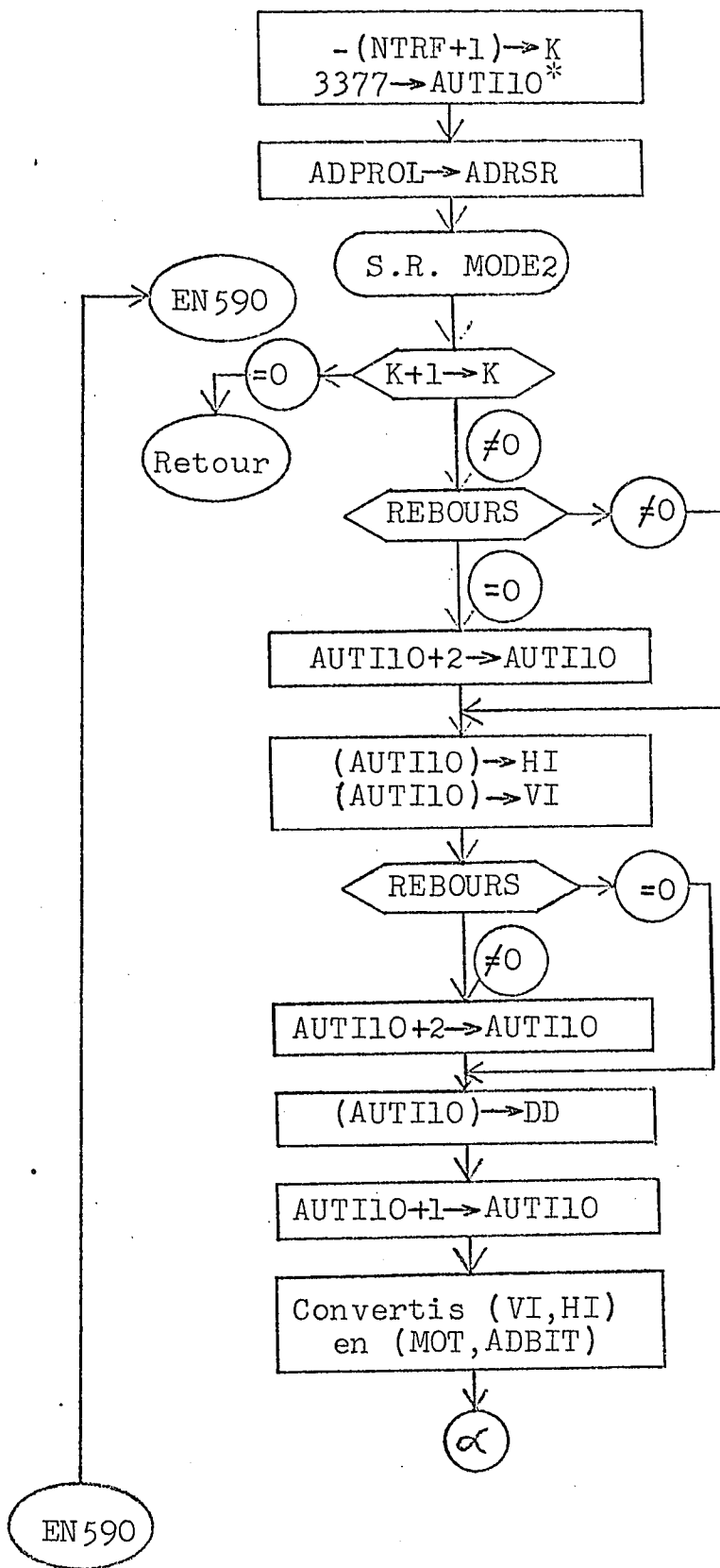
Pour remettre la S.R. SUIVRE dans son état normal.

Sous-routine PROLONGE

PROLONGE, 0

A)

B)



Prolonger toutes les traces.

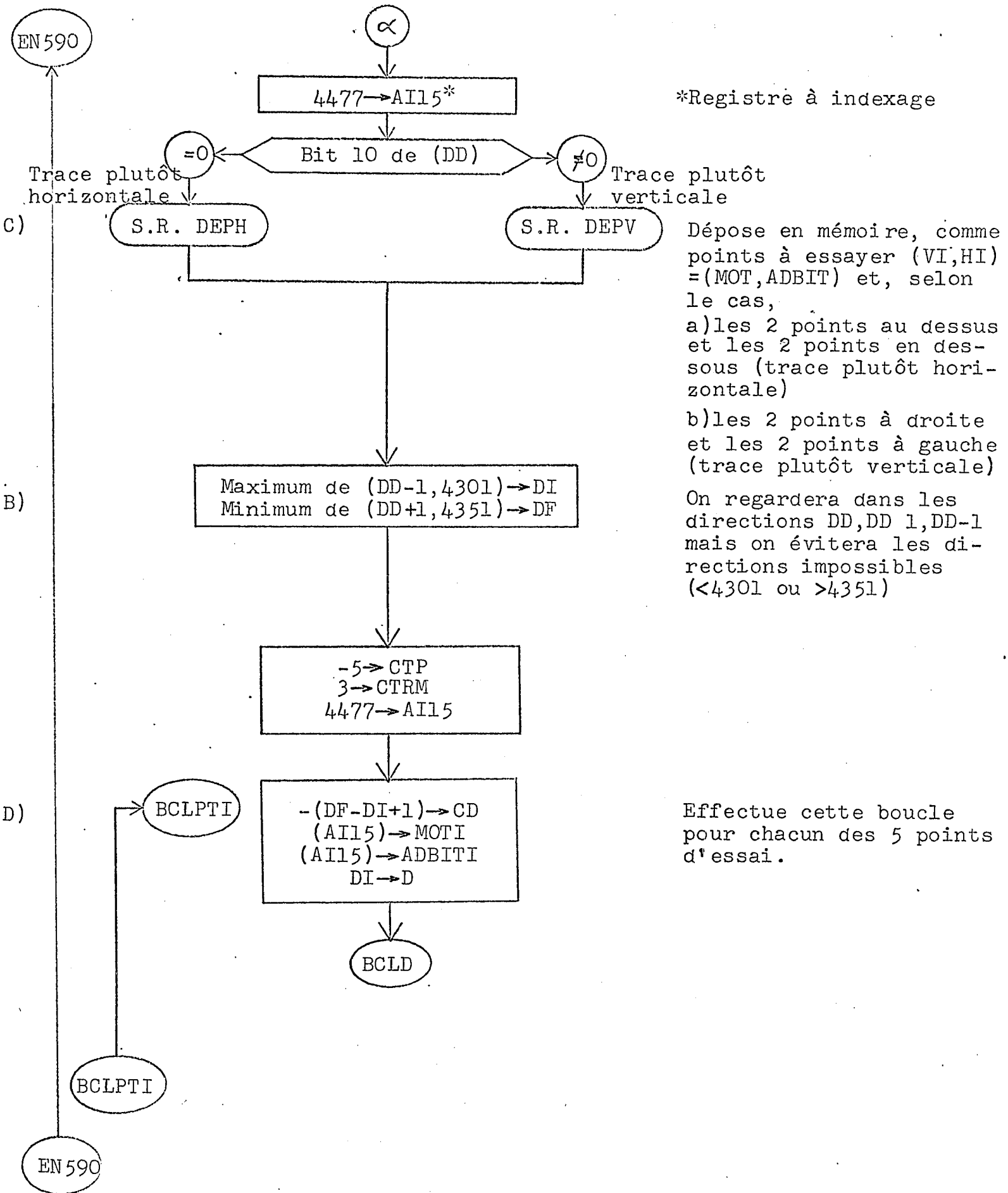
*Registre à indexage

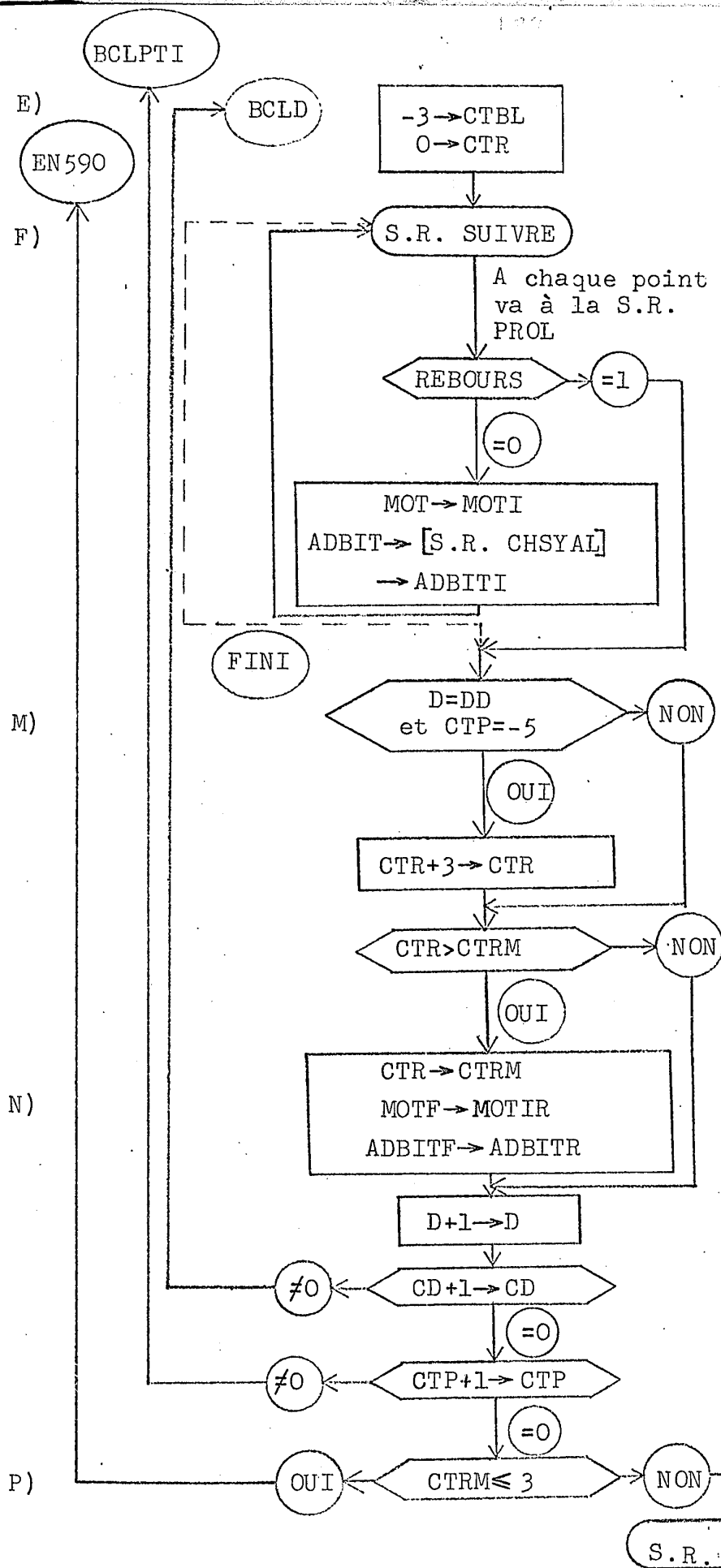
Pour que la sous-routine SUIVRE saute à la sous-routine PROL après avoir calculé la position de chaque point.

(VI, HI) est le point final de la trace pour le prolongement au-delà du point final; c'est le point initial pour le prolongement en deça du point initial.
(Cf. structure de la liste des traces, tableau 3)

DD: orientation de la trace

S.R. CONV





Effectue cette boucle pour chacune des 3 (en général) orientations et pour chacun des 5 points de départ, ce qui fait 15 essais en tout.

Pour le prolongement à rebours seulement 1 longueur (10 unités) est examinée.

Dans le prolongement vers l'avant, on regarde tant que les limites n'ont pas été atteintes et qu'il n'y a pas eu 3 points blancs(---)

Si cet essai est celui qui part exactement du bout de la trace et a la même orientation que celle-ci, donne un boni de 3 points noirs.

Choisis le prolongement ayant le plus grand nombre de points noirs (CTR maximum)

Conserve le dernier point noir de celui-ci, qui deviendra le nouveau point final (initial) de la trace.

Ne change rien si aucun prolongement valable n'a été trouvé.

AILLEURS, 0

Cette partie est dans une sous-routine pour des raisons de pagination.

C'est ici que le point final (initial) de la trace est changé.

AUT110 est mis à 1 de moins que l'adresse où se trouve HF (HI).

R)

S.R. CHSYAL et CONV

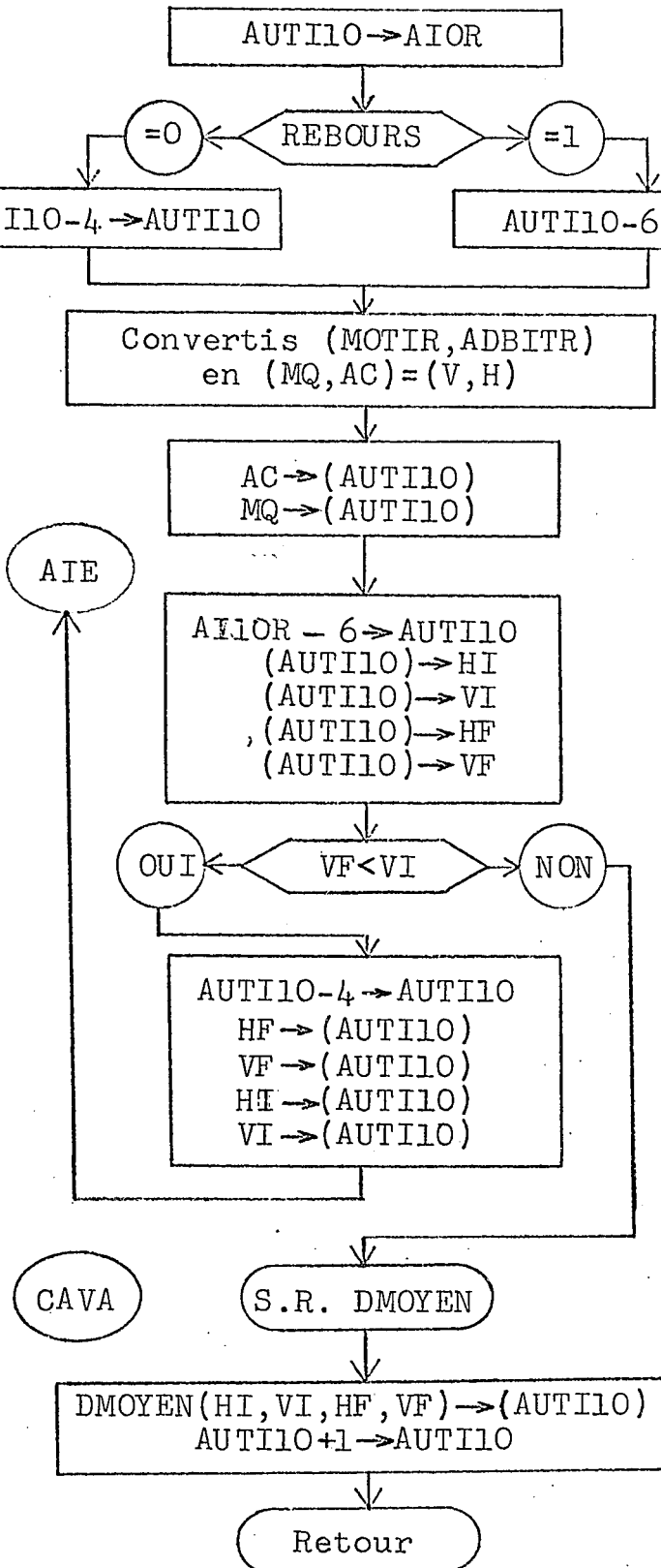
Le bout de la trace est changé ici.

Q)

Si le point final est plus haut que le point initial, échange les.

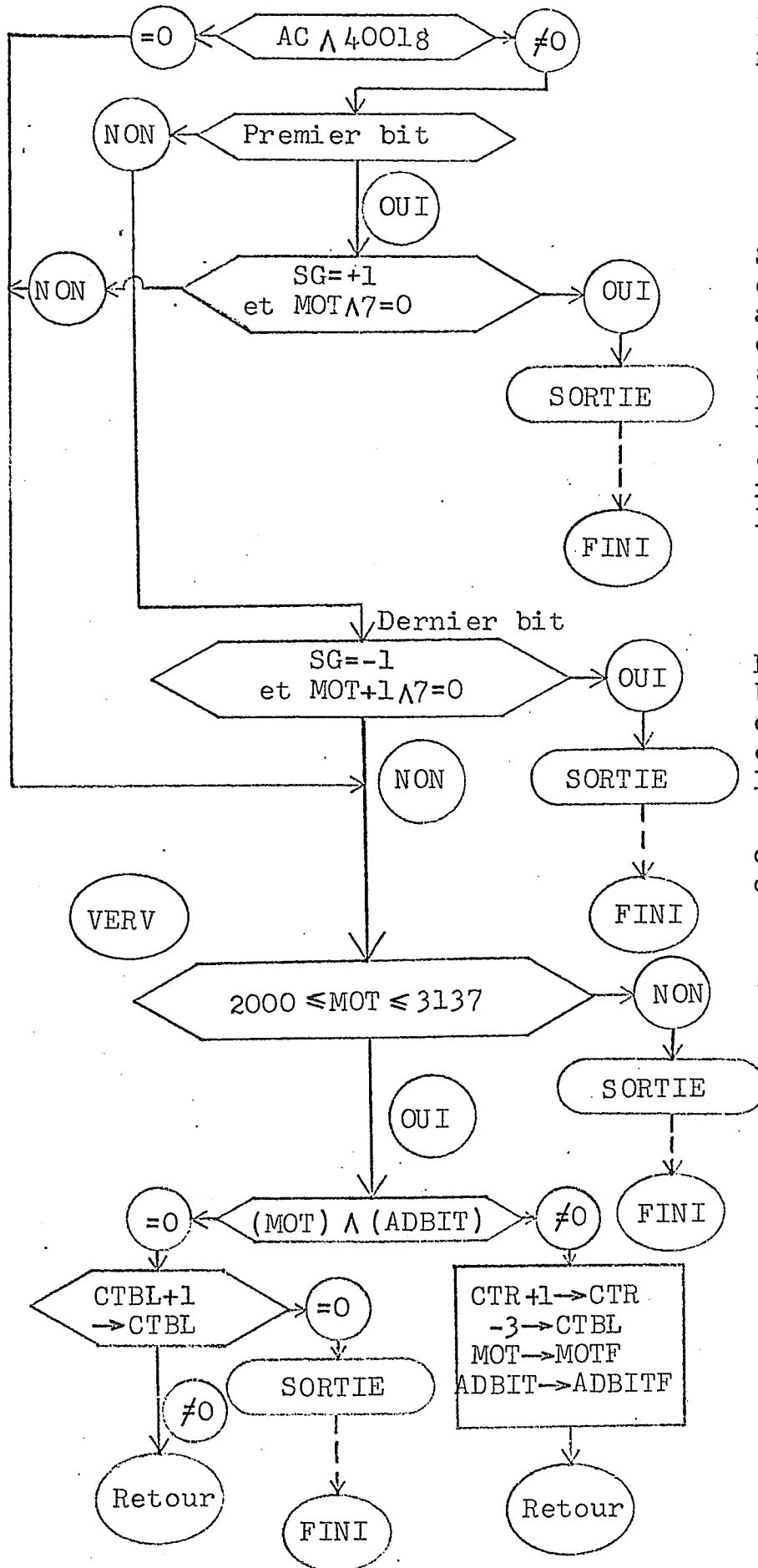
Retourne à AIE pour avoir les bonnes valeurs de HI, VI, HF, VF, pour recalculer l'orientation.

Recalcule l'orientation de la trace.



Sous-routine PROL
PROL, 0

I)



Premier ou dernier bit d'un mot (bit 0 ou bit 11)

Si c'est le premier bit d'une ligne (bit 0 mot xxx0 à la suite d'un décalage à droite, c'est à cause de la structure de la mémoire qui fait qu'un décalage à droite de (V,96) donne (V+1,1) et non pas (V,97) mais cela indique qu'on est sorti des limites de l'image, en fait

De même si c'est le dernier bit de la ligne à la suite d'un décalage à gauche, c'est qu'on est sorti des limites.

(V,1) → (V-1,96) après un décalage à gauche. Mais ça devrait être (V,0)!

Vérifie que $1 \leq V \leq 76$

Point blanc ou noir?

CTR: Nombre de points noirs
(MOTF, ADBITF): dernier point noir.

CTBL: Compteur de points blancs consécutifs. Cesse le prolongement dès que 3 points blancs consécutifs ont été perçus.

J)

K)L)

# DE TRACE	HI	VI	HF	VF	D	# DE SEGMENTS	# DE PTS
1	20	1	43	66	17	8	61
2	73	1	87	34	17	4	36
3	35	2	34	41	21	5	39
4	34	41	20	69	26	4	34
5	54	49	57	58	18	2	12
6	41	56	45	76	19	2	17

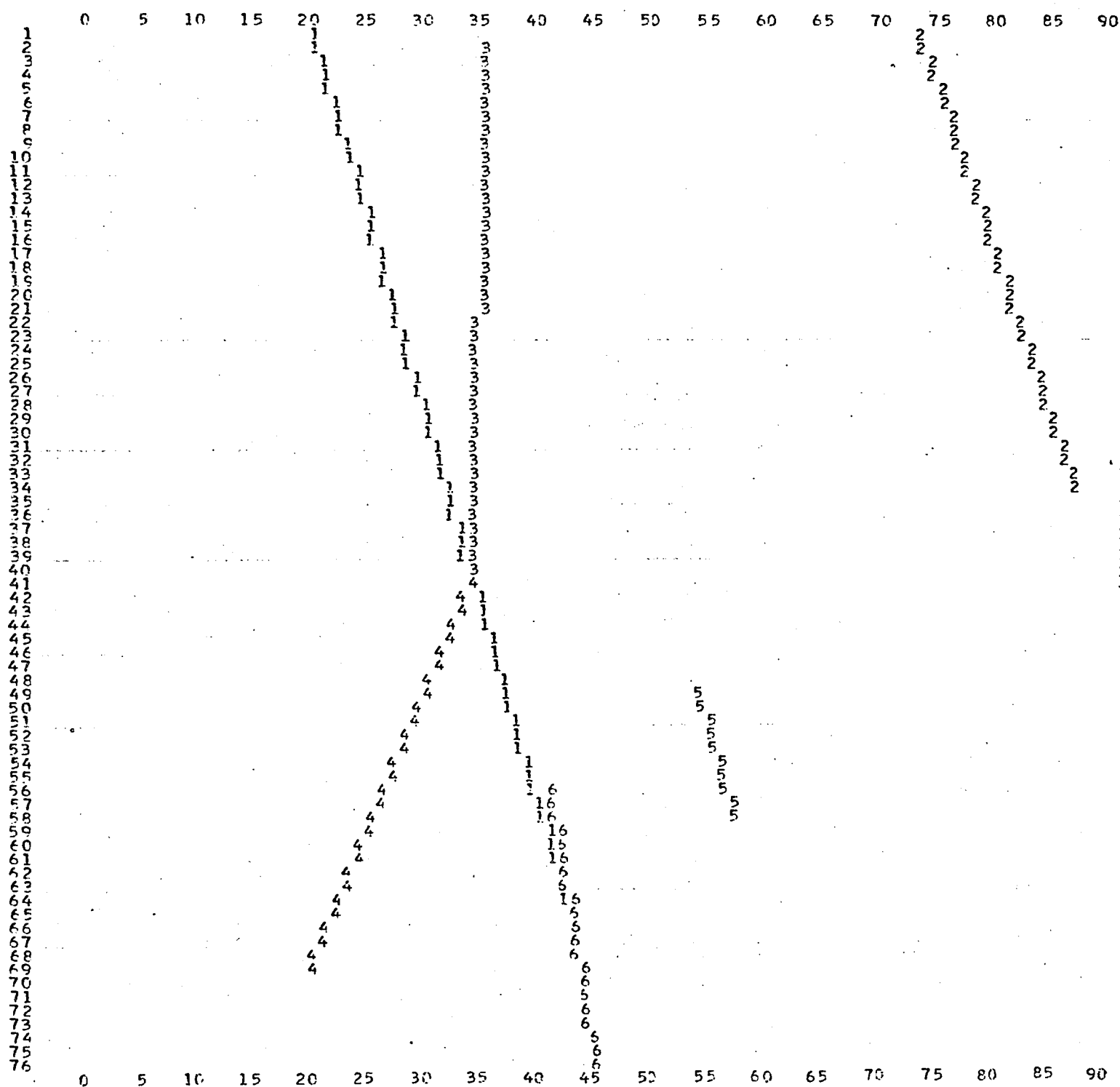


Figure 19 - Etoile no. 1 après l'étape de prolongement des traces.
(CRITER= 6 et 5 à l'étape 3)

B. Assemblage des traces

Cette partie du programme réunit en une seule les traces que la partie 3 a pu séparer à tort.

Pour que deux traces soient réunies, il faut que leurs orientations soient presque identiques (différence $\leq$ ASSEZ) (voir l'organigramme 20, (H)) et que le point initial de l'une soit tout près du point final de l'autre (distance ≤ 2 , (I) et (M)).

Le programme réunira aussi deux traces proches mais non bout à bout*. Il faut alors que le point initial de la seconde trace soit près du point final de la première, et que le point final de la seconde soit au-delà de celui de la première. ((I)-(N)).

Un cas plus particulier a aussi été prévu. Un coup d'oeil à la figure 17 montre que, par exemple, une trace d'orientation $D = 41$, et une autre d'orientation $D = 1$, dont les points initiaux coïncideraient devraient être fondues en une seule dont le point initial serait le point final de la seconde et le point final, le point final de la première.

C'est pour ce cas et d'autres, analogues, qu'ont été prévues les étapes (G) et (P) - (AA).

Une fois qu'a été décidée la réunion de deux traces, la nouvelle trace est créée en changeant son point final (initial)

*Distance ≤ 2 . Pour les fins de l'assemblage la définition de la distance (section III-C) du point initial de la 2e trace à la première trace est modifiée afin de permettre que le point soit jusqu'à 5 unités au-delà du point final. C'est grâce à ce prolongement tentatif de la trace que l'algorithme peut resouder les traces brisées.

en celui de l'autre ((O),(U),(AA)) et la seconde est éliminée en rétrogradant toutes les suivantes d'un rang ((EE),(FF)).

L'algorithme est exécuté quatre fois, avec ASSEZ = 0,1,2 (bis). (ASSEZ est la tolérance sur l'orientation). On assemble ainsi d'abord les traces dont les orientations sont le plus proches et, de plus, cette technique permet d'assembler en une seule plus de deux traces. Ces caractéristiques se sont avérées essentielles dans certains cas, et le temps supplémentaire requis est infime.

La section "assemblage des traces" permet donc de souder les traces brisées et les traces dédoublées, parce que trop épaisses, en certains endroits.

Cela peut s'avérer crucial lors de la décision de retenir ou pas une image comme événement. En effet, l'algorithme de "recherche des étoiles" exige deux branches ayant une longueur suffisante.

Pour l'assemblage des traces, la version en langage d'assemblée suit de très près la version FORTRAN. L'organigramme 21 donne les détails.

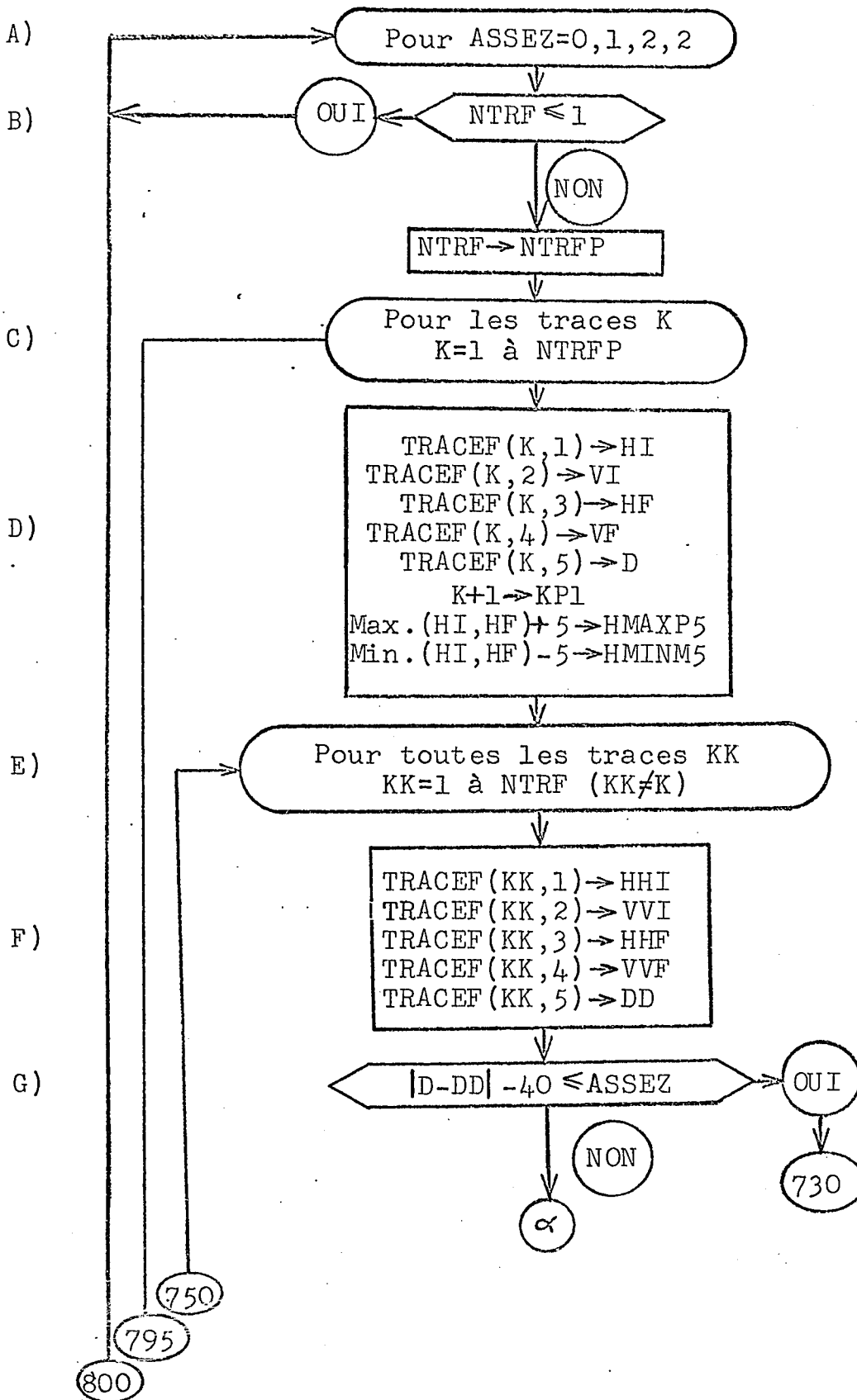
On peut voir un exemple d'assemblage en comparant les figures 19 et 20.

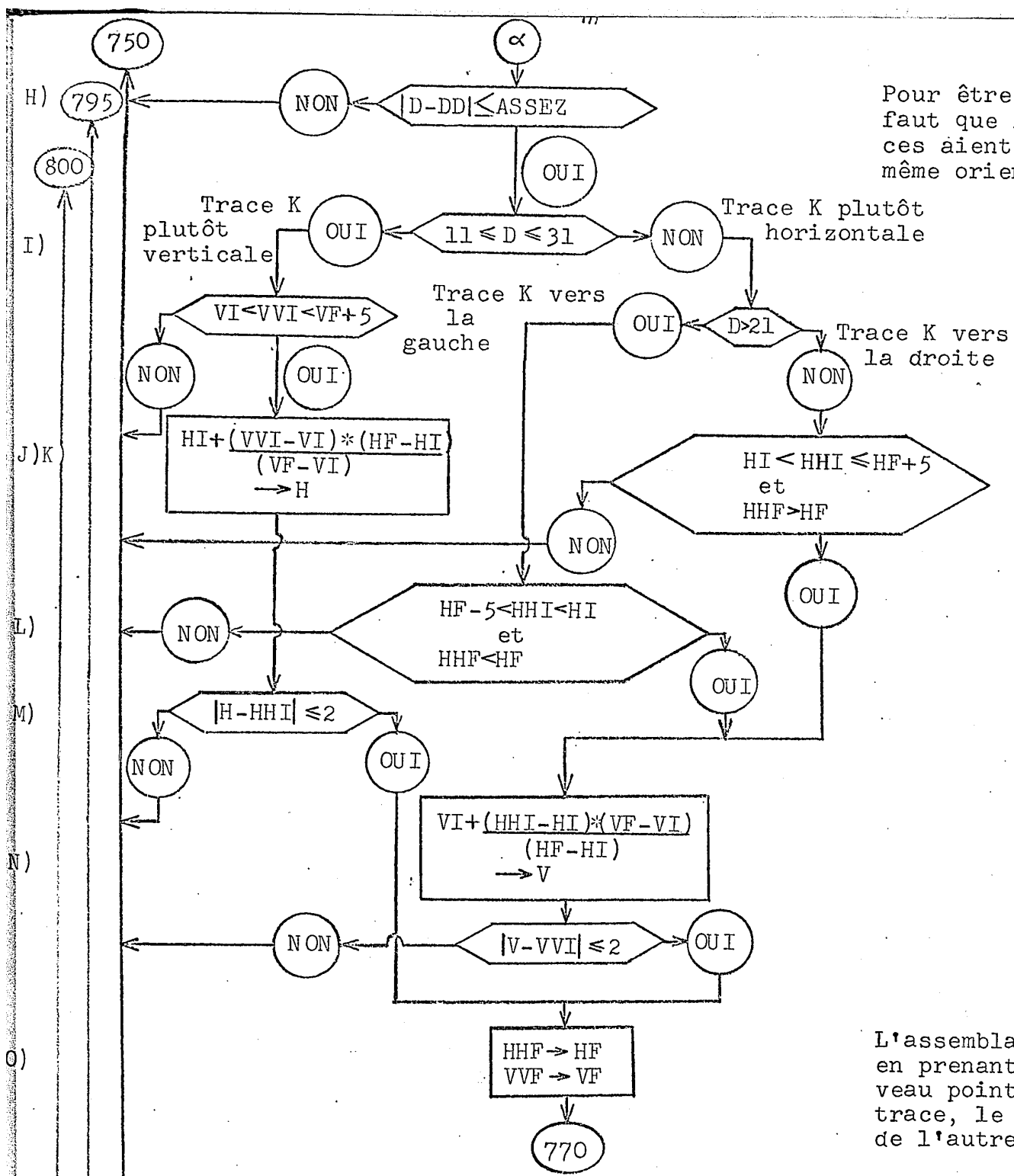
6-ASSEMBLAGE DES TRACES

Pour que l'algorithme soit exécuté avec les valeurs ASSEZ=0,1,2. (2 fois avec ASSEZ=2)

Examine toute les paires de traces.

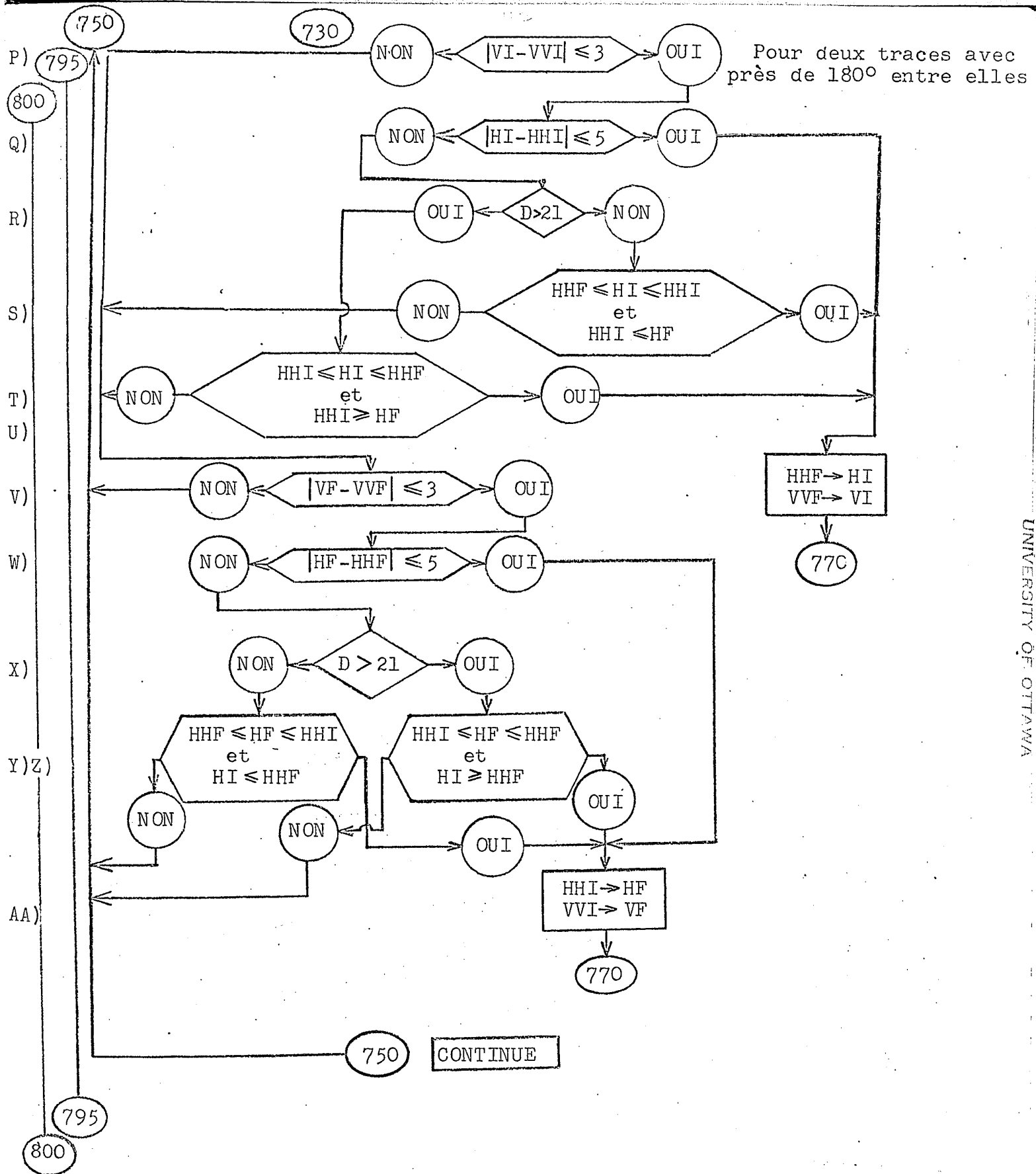
Pour le cas où on a deux traces quasi horizontales, mais de sens opposé.
($D \approx 1$ et $DD \approx 41$, ou $D \approx 41$ et $DD \approx 1$)

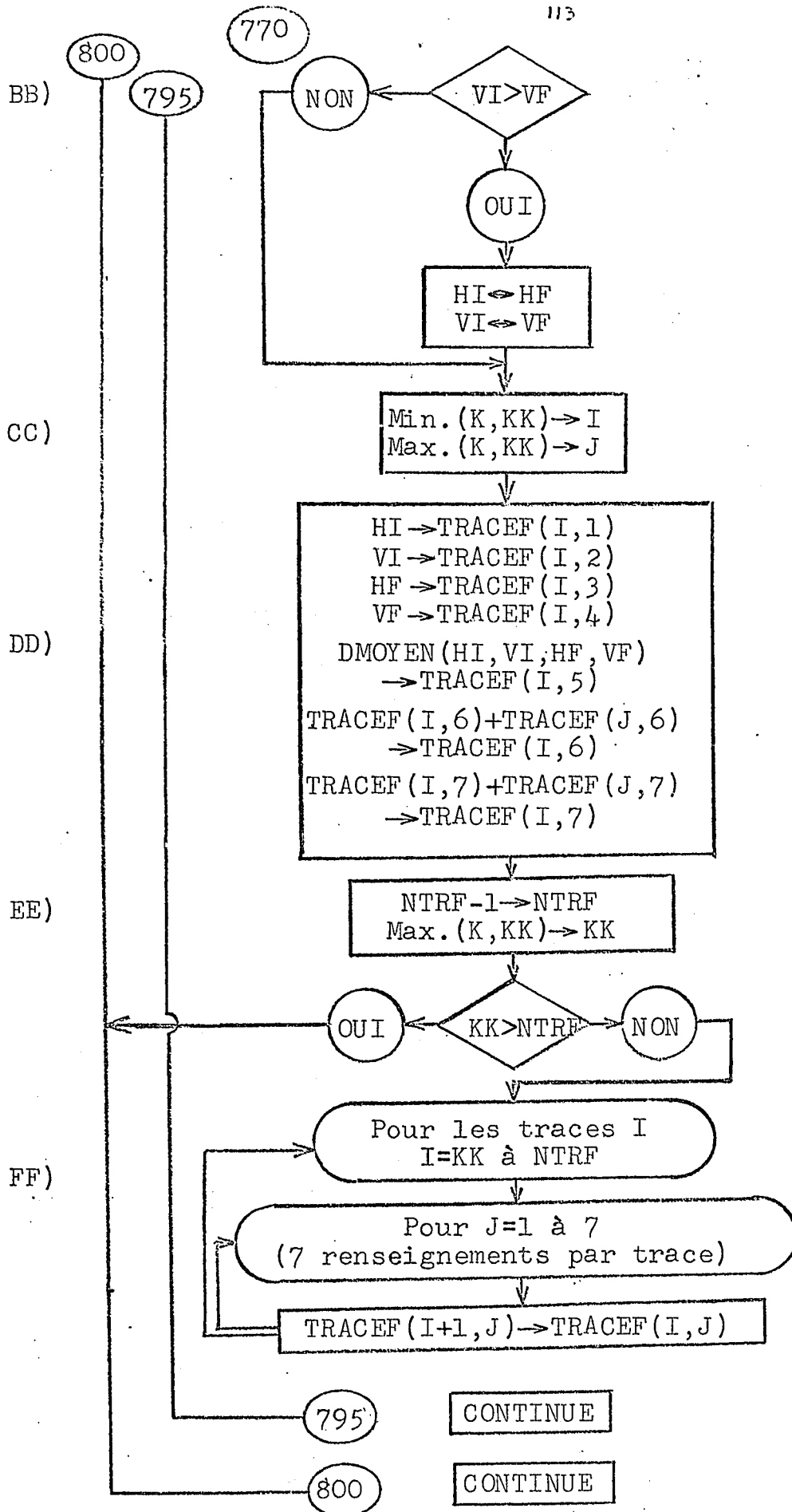




Pour être jointes, il faut que les deux traces aient presque la même orientation

L'assemblage se fait en prenant comme nouveau point final de la trace, le point final de l'autre trace.





Si le point initial est plus bas que le point final, il faut les échanger.

La trace nouvelle remplace l'ancienne trace I.

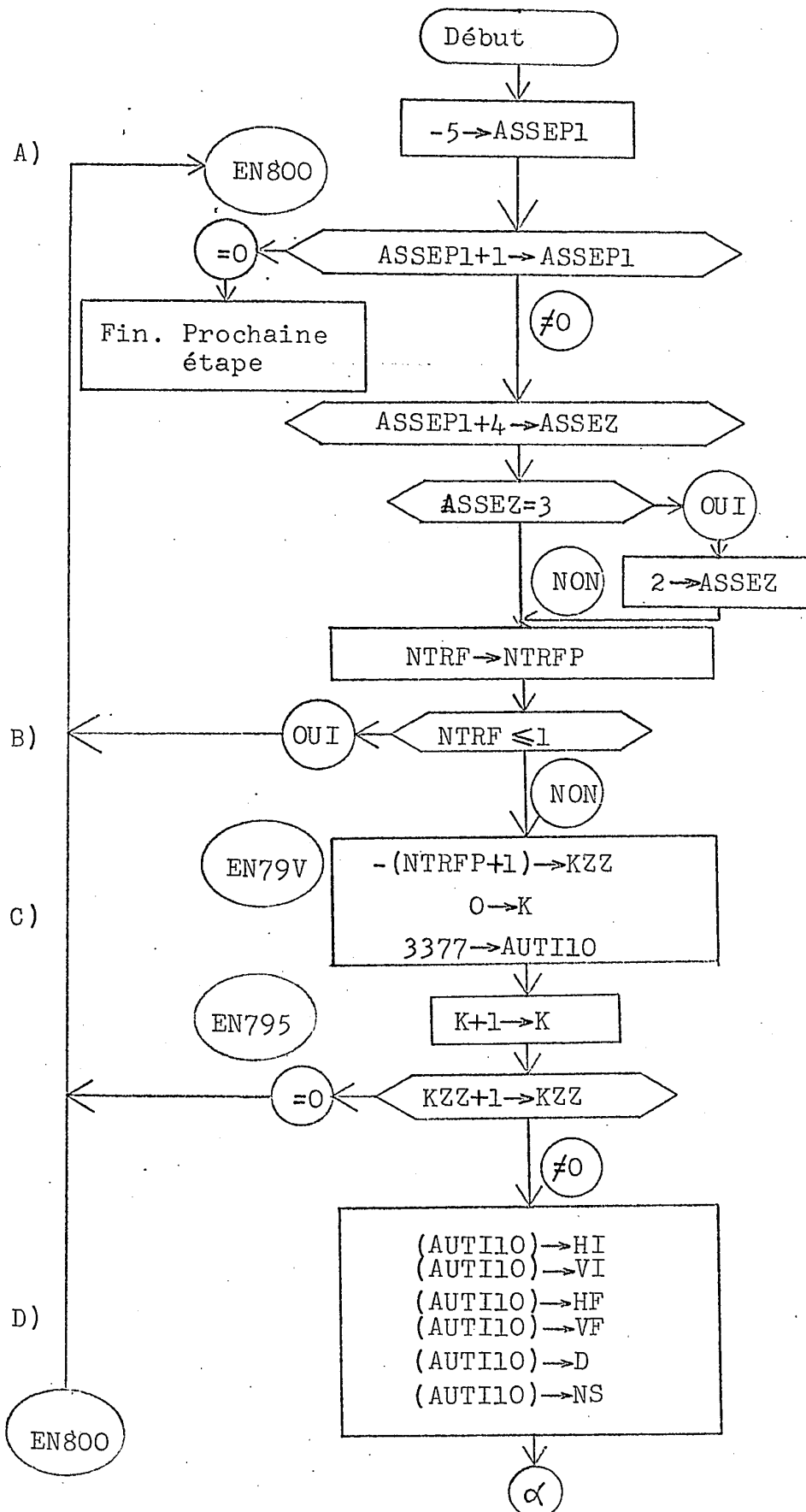
Recalcule l'orientation

La trace J est éliminée de la liste des traces.

Les traces suivantes sont rétrogradées d'un rang:

trace J+1 → trace J
trace J+2 → trace J+1
etc.

Assemblage des traces-Version "ASSEMBLEUR"

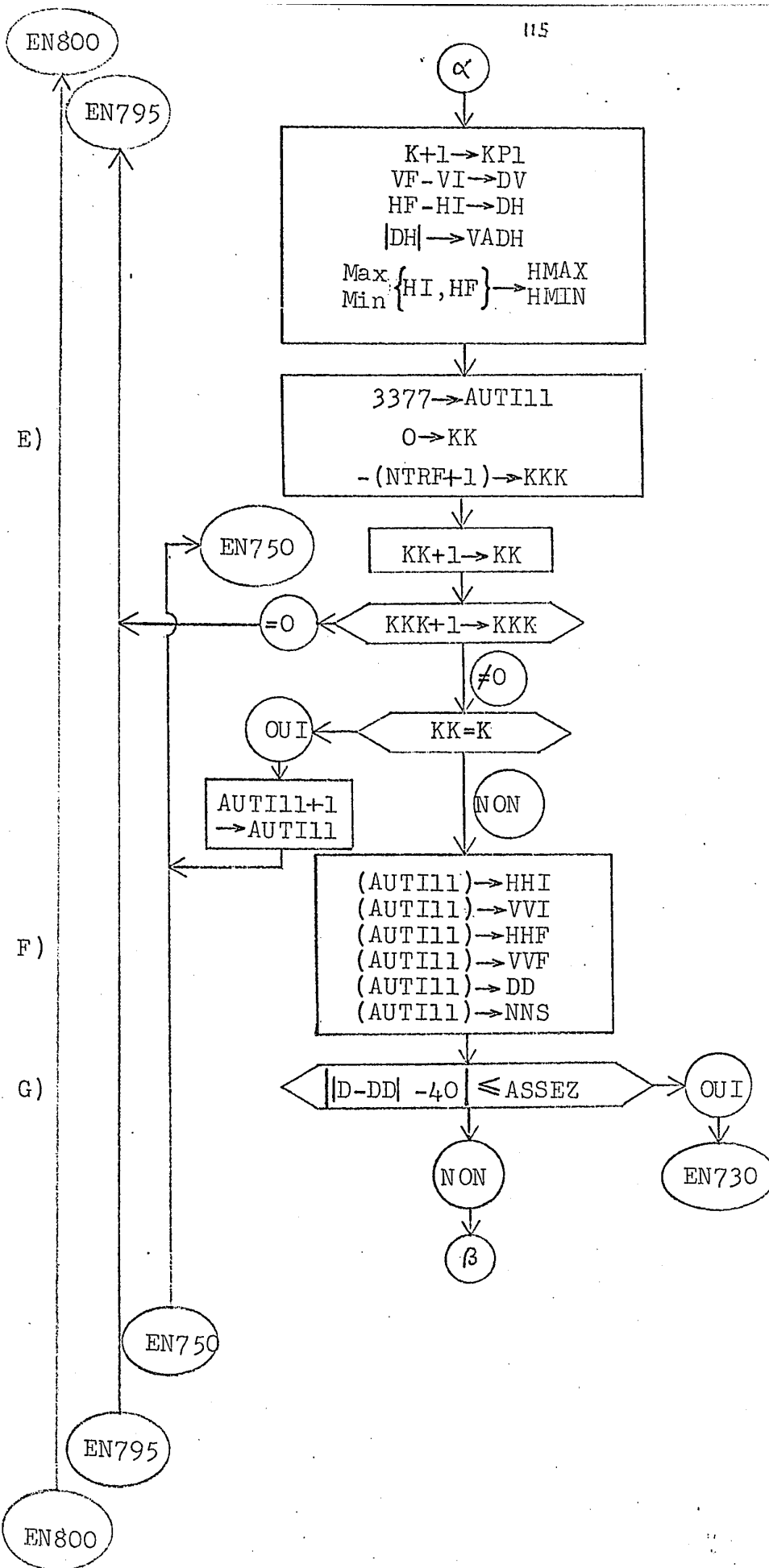


Exécute l'algorithme
avec ASSEZ=0,1,2,2
(2 fois avec ASSEZ=2)

KZZ: Compteur à
rebours

K: Compteur

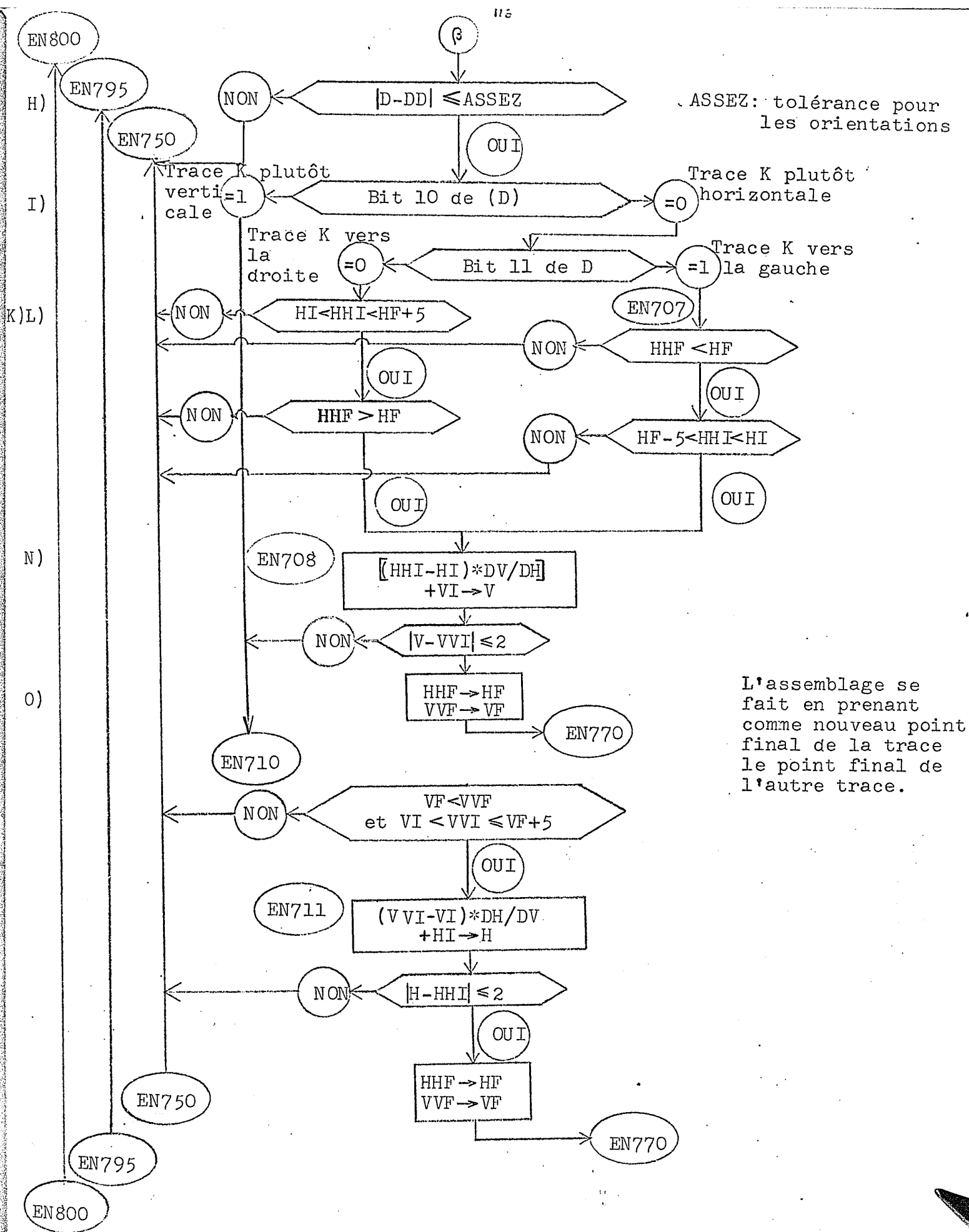
Faire pour les traces
K, K=1 à NTRFP



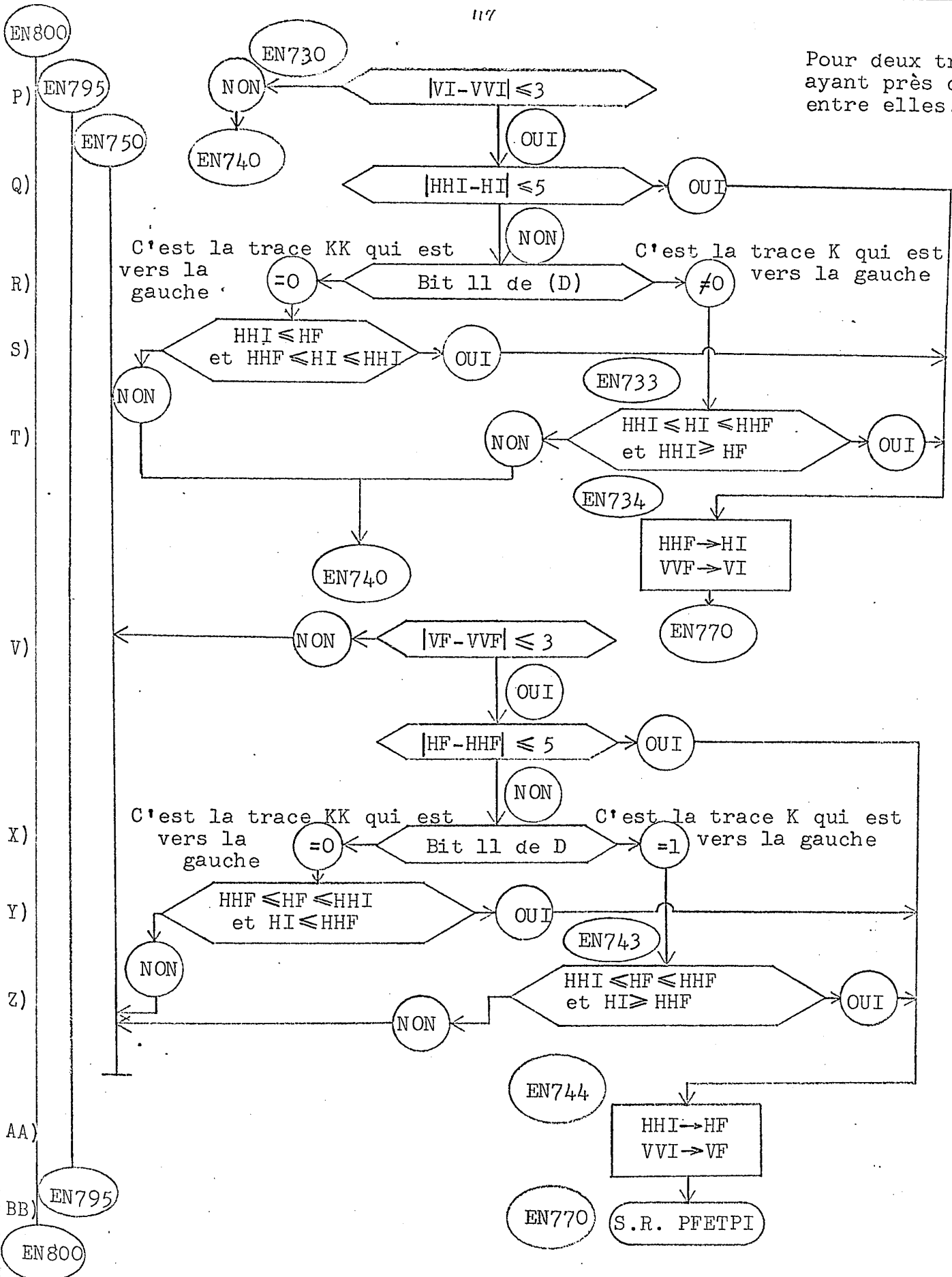
KKK: Compteur à rebours
KK: Compteur

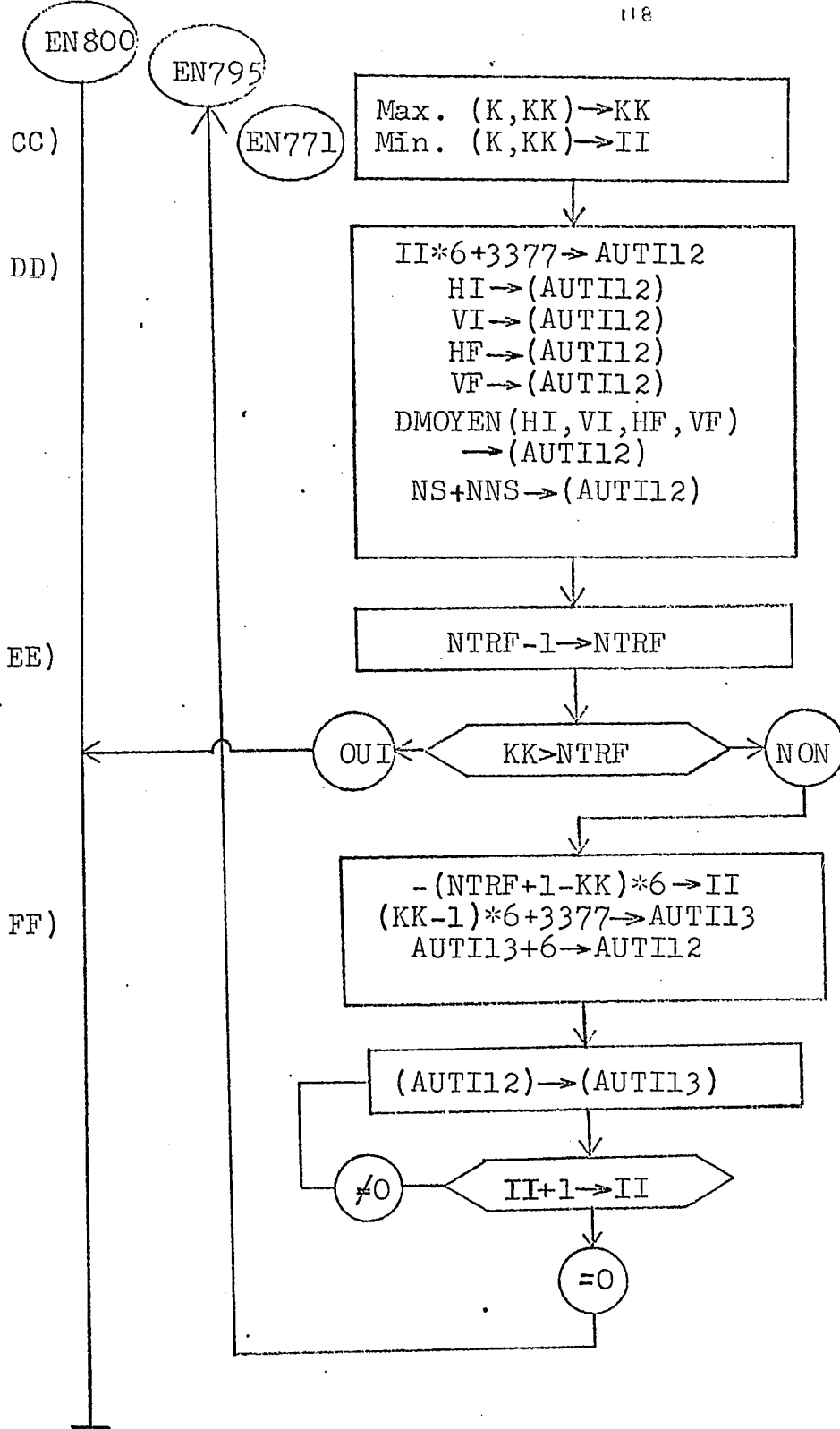
Examine toutes les paires de traces.

Pour le cas où on a deux traces quasi horizontales mais de sens opposé.
(D≈4301 et DD≈4351
ou D≈4351 et DD≈4301)



Pour deux traces
ayant près de 180°
entre elles.



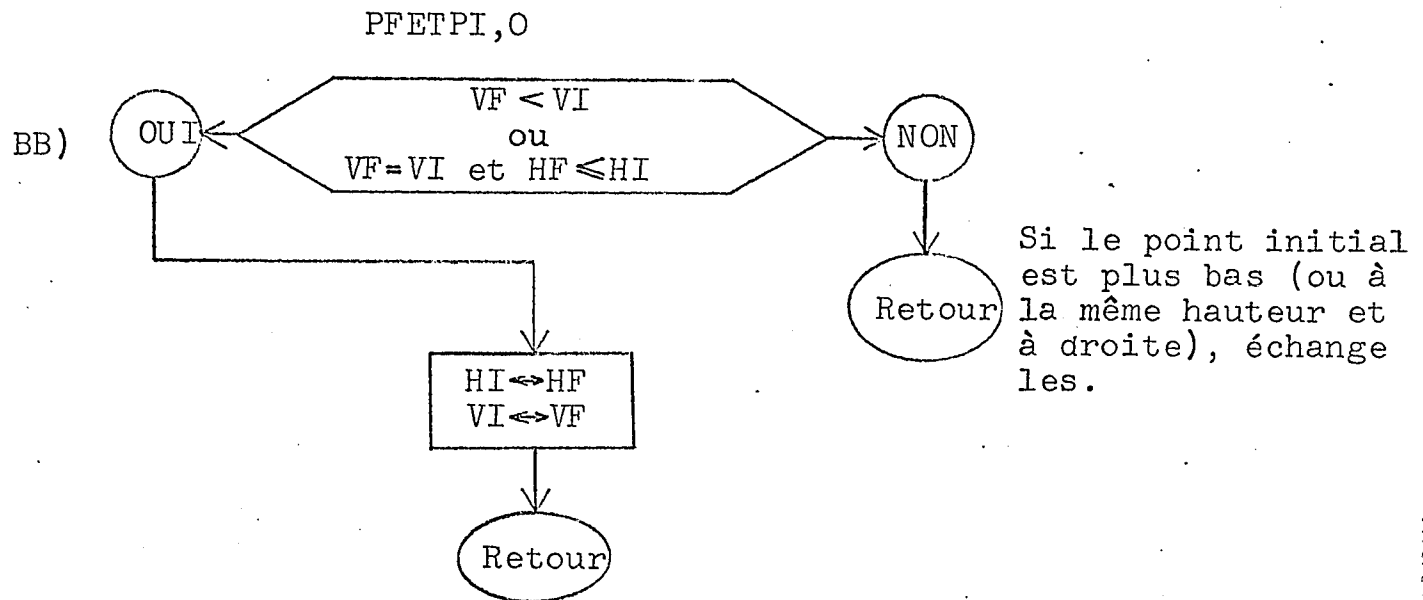


La nouvelle trace
remplace l'ancienne
trace II.

Recalcule l'orientation
de la trace. (S.R.
DMOYEN)

La trace KK est éli-
minée de la liste
des traces.

Les traces suivantes
sont rétrogradées d'un
rang.

Sous-routine PFETPI

# DE TRACE	HI	VI	HF	VF	D	# DE SEGMENTS	# DE PTS
1	20	1	45	76	18	10	78
2	73	1	87	34	17	4	36
3	35	2	34	41	21	5	39
4	34	41	20	69	26	4	34
5	54	49	57	58	18	2	12

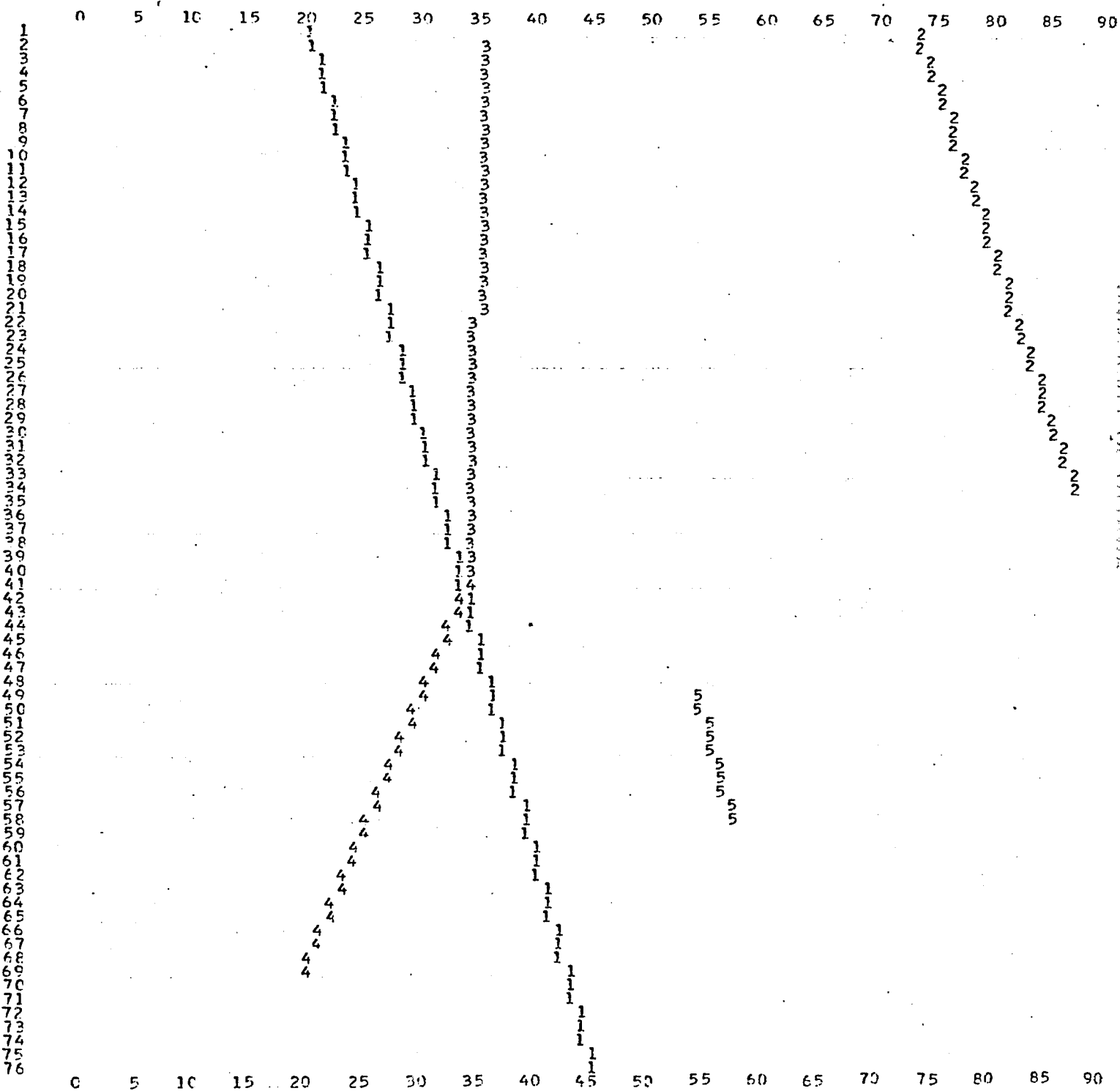


Figure 20 - Etoile no. 1 à la fin de l'analyse
(CRITER:: 6 et 5 à l'étape 3)

# DE TRACE	H1	VI	HF	VF	D	# DE SEGMENTS	# DE PTS
1	20	1	45	76	18	10	76
2	73	1	67	34	17	4	36
3	35	2	34	41	21	5	39
4	62	2	61	11	22	2	11
5	33	40	24	49	31	2	13
6	34	41	20	69	26	4	34
7	54	49	57	58	18	2	12

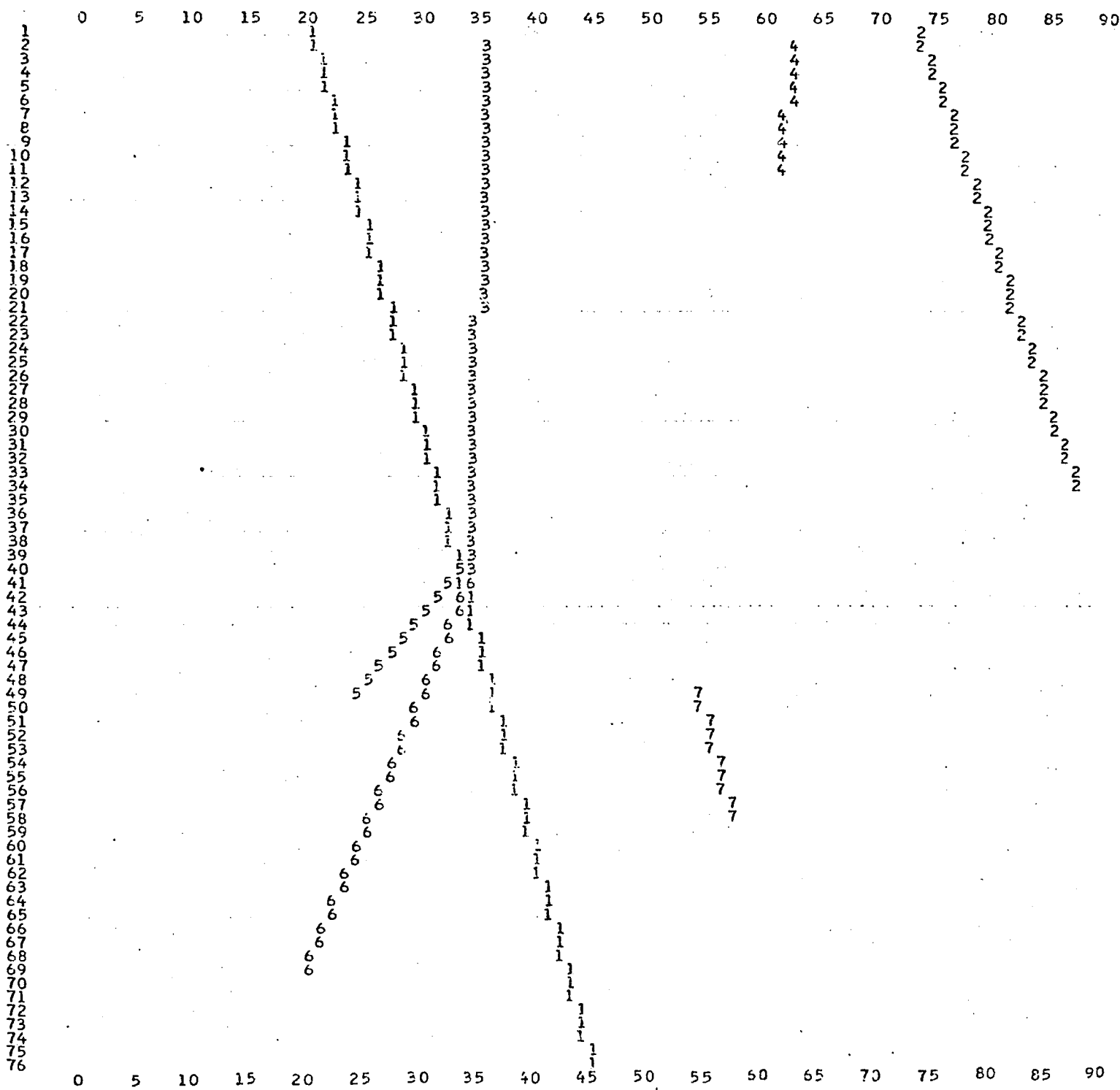


Figure 21 - Etoile no. 1 (CRITER = 5 et 4 à l'étape 3)

# DE TRACE	HI	VI	HF	VF	D	# DE SEGMENTS	# DE PTS
1	42	1	46	34	20	5	32
2	38	32	32	42	27	2	12
3	46	34	26	64	28	4	32

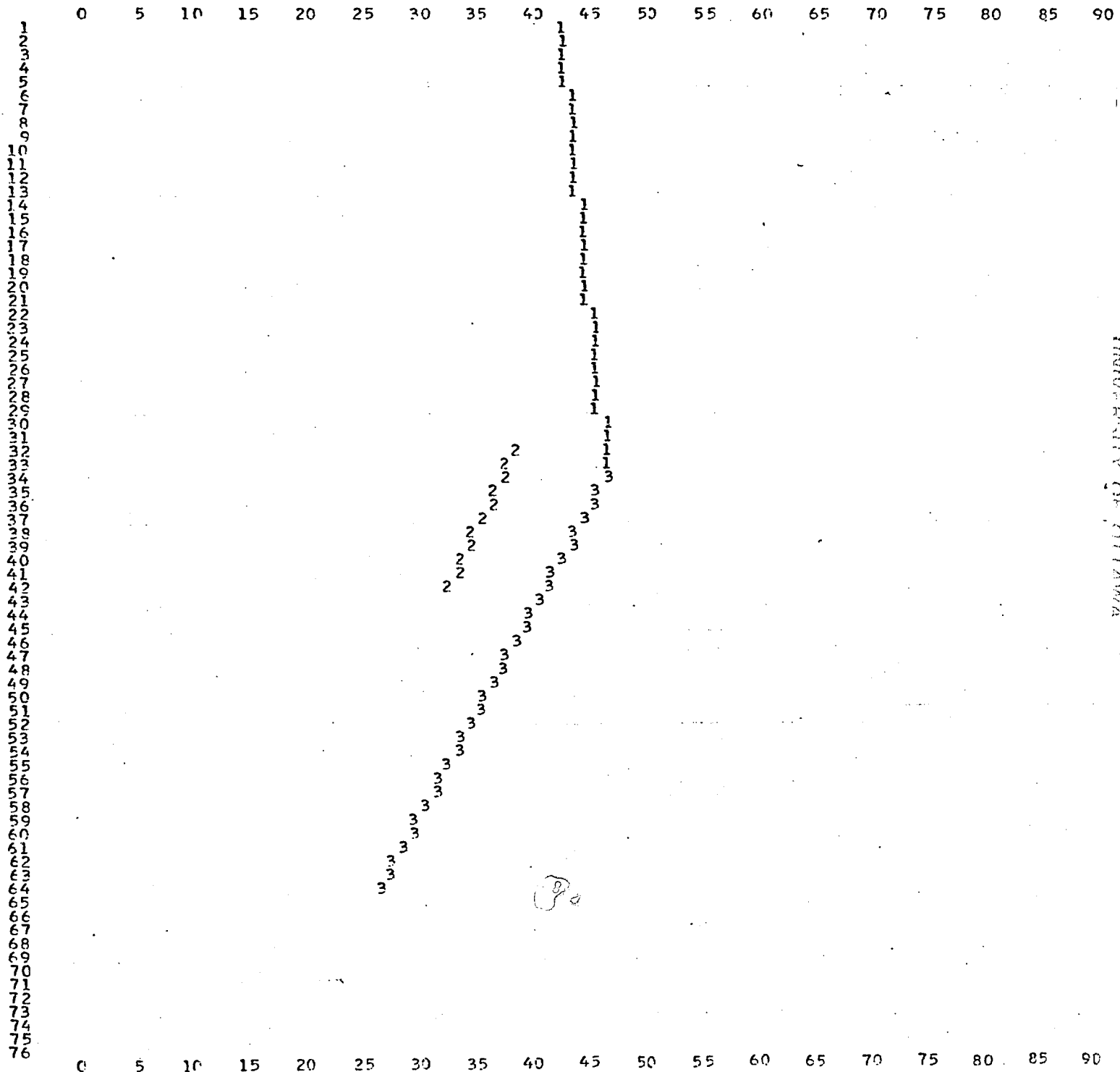


Figure 22 - Etoile no. 2 (CRITER = 6 et 5 à l'étape 3)

# DE TRACE	H1	V1	HF	VF	D	# DE SEGMENTS	# DE PTS
1	42	1	46	34	20	5	32
2	25	17	16	23	34	2	11
3	16	18	21	28	16	2	11
4	38	31	32	42	26	2	12
5	46	34	26	64	28	4	32
6	74	35	68	45	27	2	11

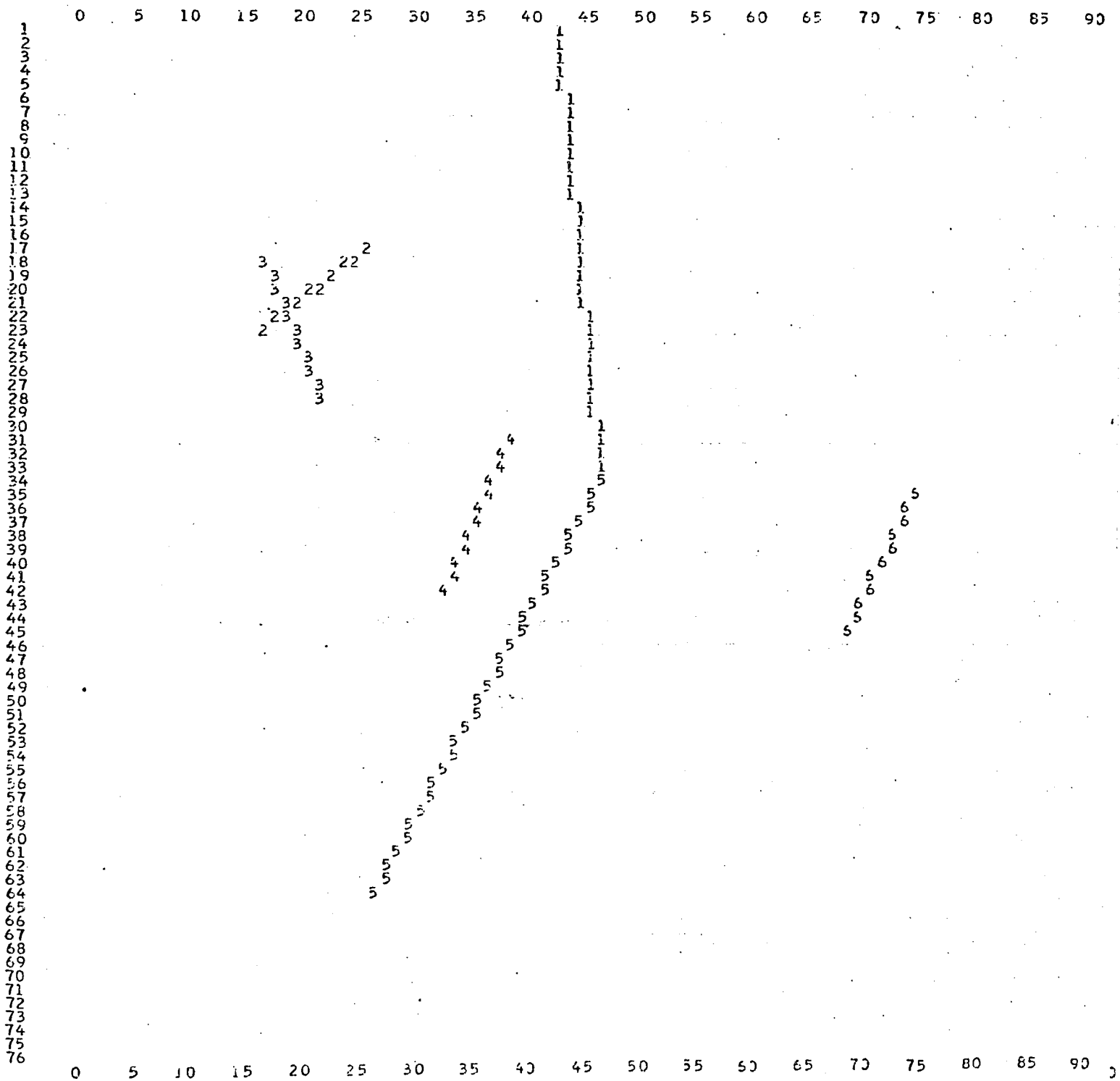


Figure 23 - Etoile no. 2 (CRITER = 5 et 4 à l'étape 3)

# DE TRACE	HI	VI	HF	VF	D	# DE SEGMENTS	# DE PTS
1	34	1	42	76	20	8	70
2	5	5	40	32	9	4	28
3	30	29	56	40	5	3	22
4	48	34	33	76	25	4	29
5	41	38	25	53	32	3	17
6	17	57	1	71	32	2	13

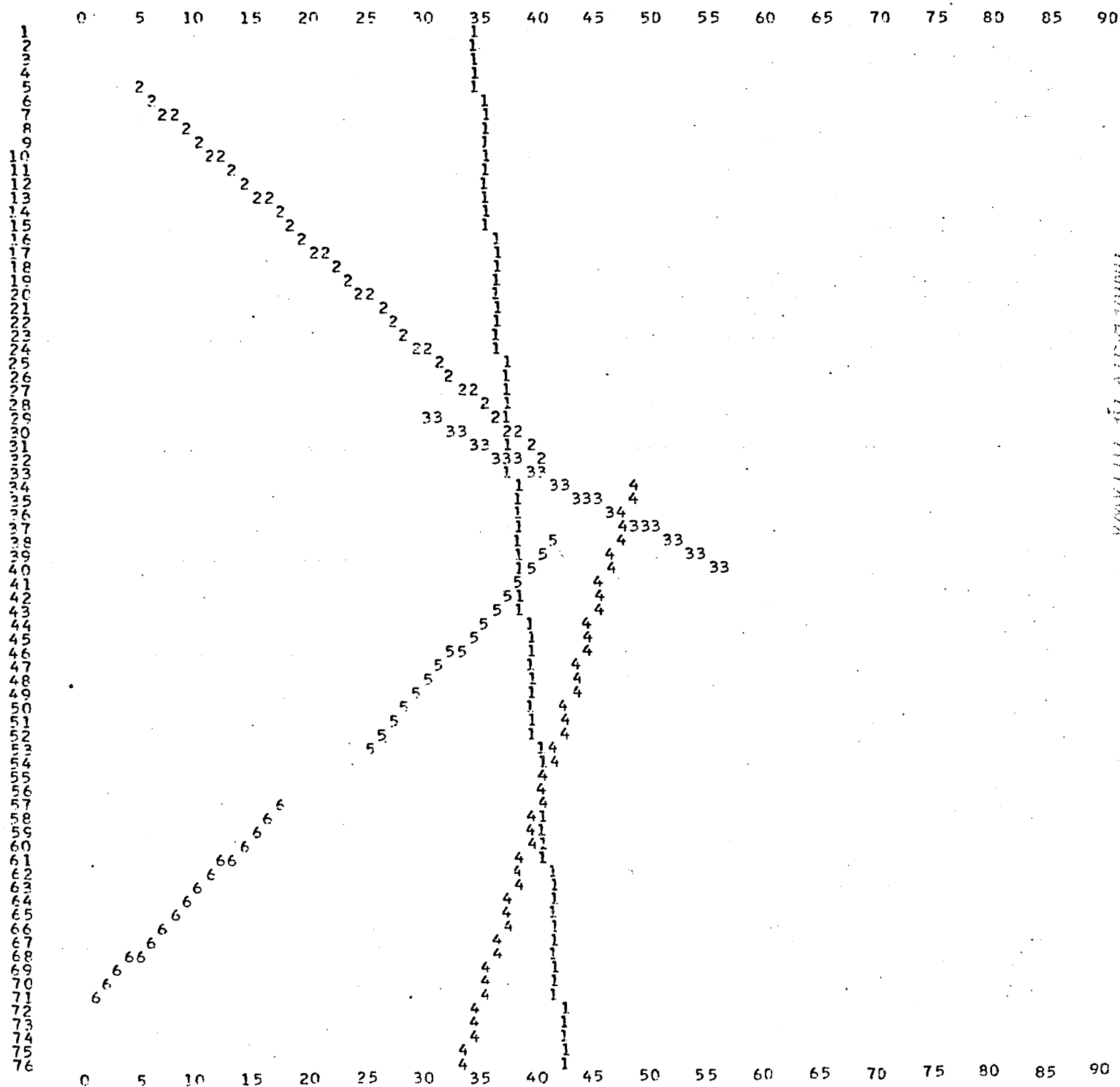


Figure 24 - Etoile no. 3 (CRITER = 6 et 5 à l'étape 3)

# DE TRACE	HI	VI	HF	VF	D	# DE SEGMENTS	# DE PTS
1	34	1	42	76	20	8	70
2	5	5	49	38	9	5	32
3	18	24	46	37	6	3	17
4	48	34	33	76	25	4	30
5	50	36	29	44	37	3	16
6	36	43	1	71	33	3	23
7	24	48	14	53	36	1	11

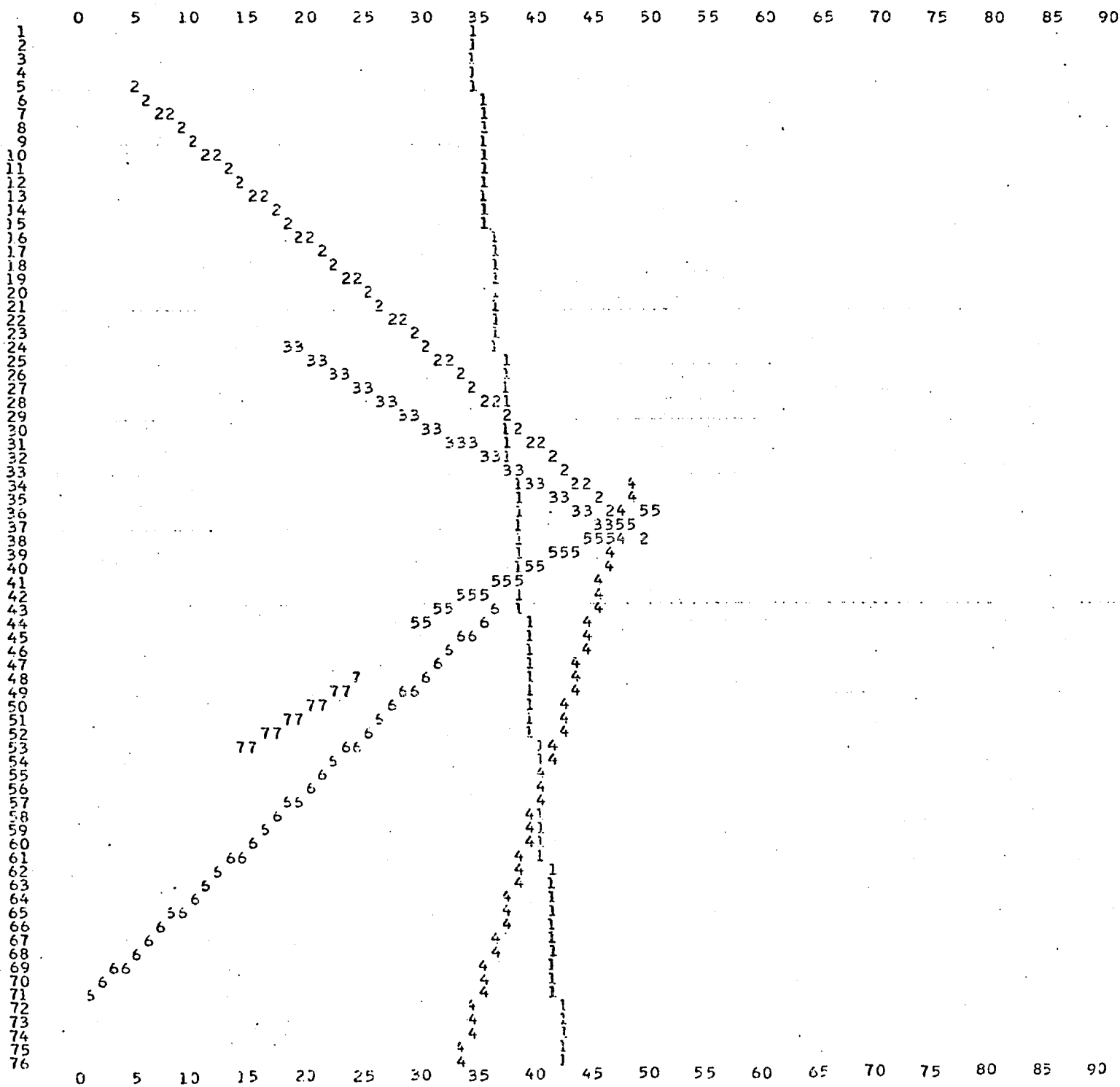


Figure 25 - Etoile no. 3 (CRITER = 5 et 4 à l'étape 3)

Figure 26 - Evénement cosmique (analysé sans
filtrage, avec CRITER = 7 et 6 à
l'étape 3).

Comparer aux figures 4, 5, 6 et 8.

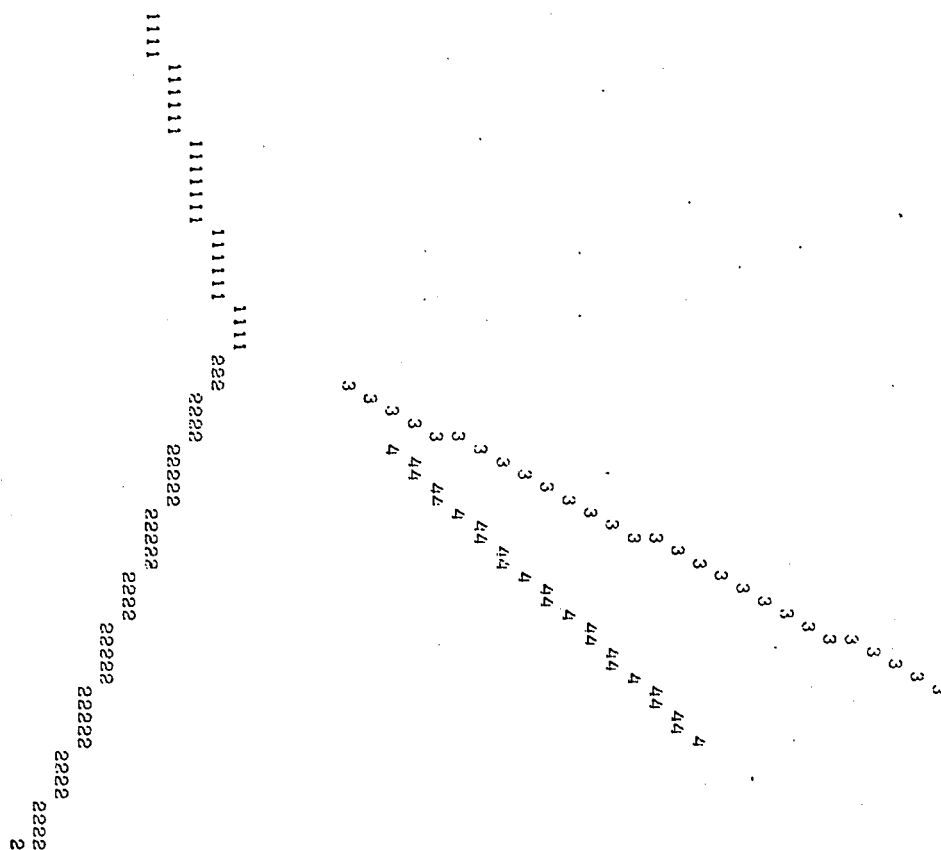


Figure 27 - Evénement cosmique (analysé sans
filtrage, avec CRITER = 5 et 4 à
l'étape 3)

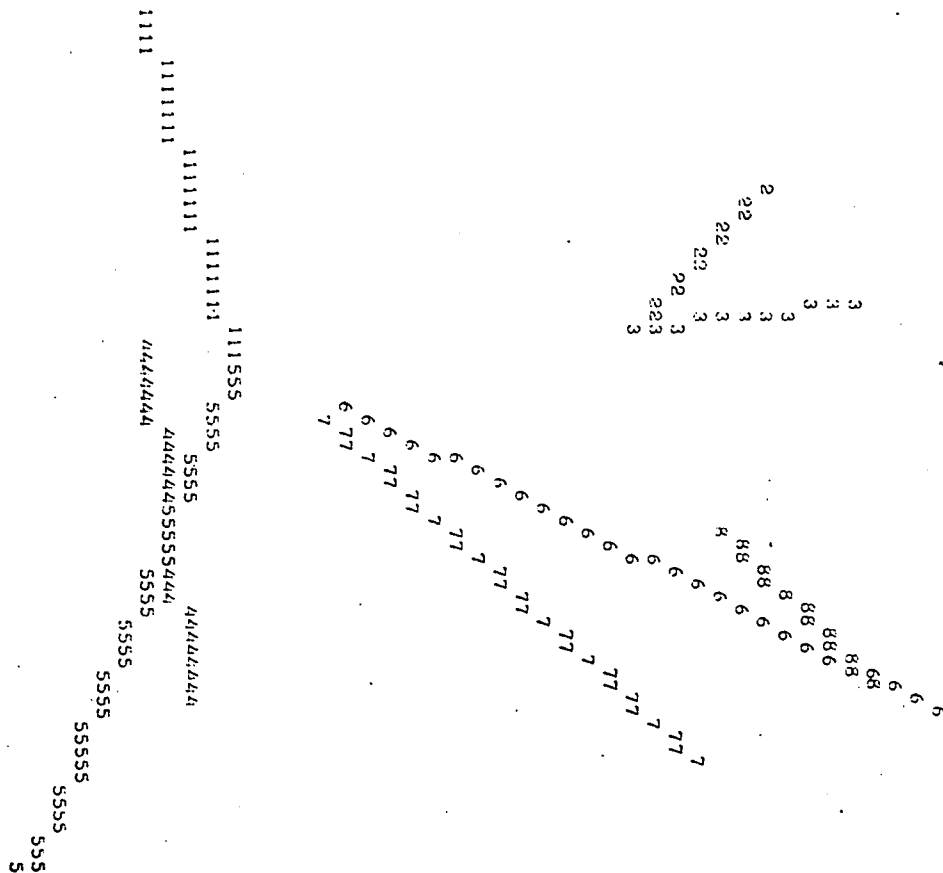


Figure 28 - Evénement cosmique (analysé sans
filtrage, avec CRITER = 6 et 5 à
l'étape 3)

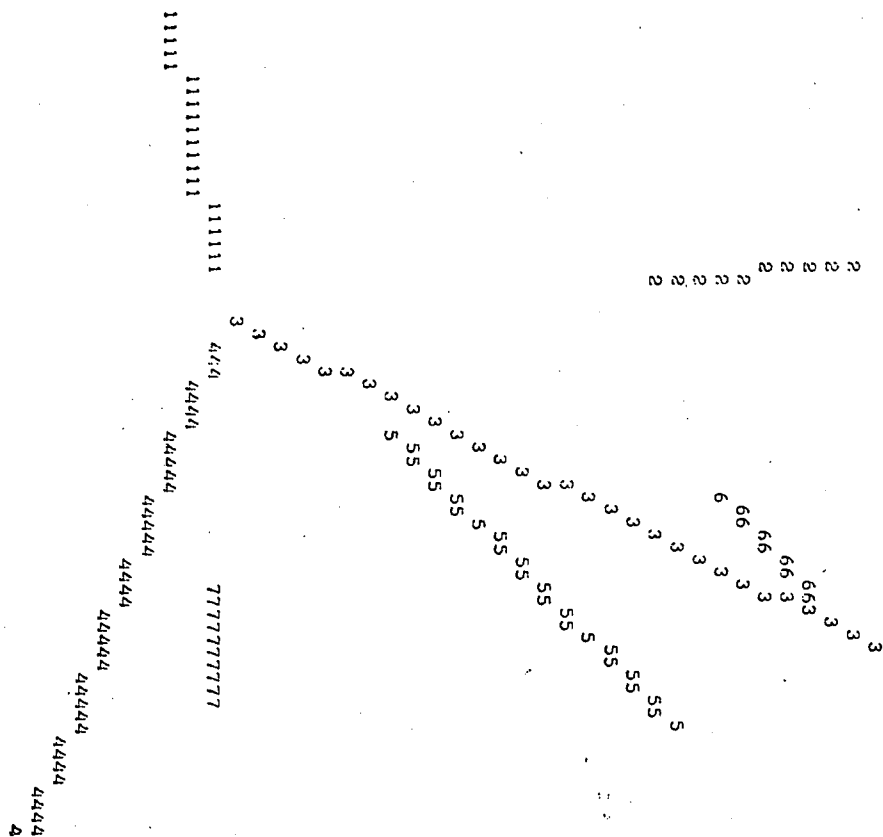


Figure 29 - Evénement cosmique (analysé avec filtrage, avec CRITER = 5 et 4 à l'étape 3)

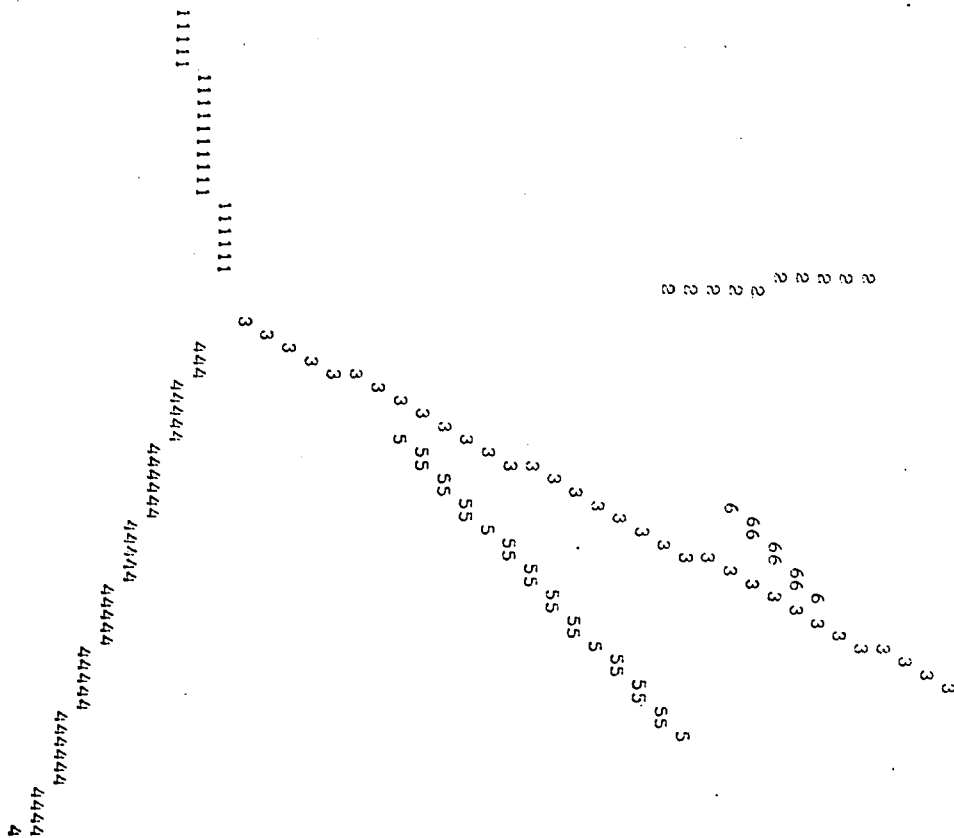
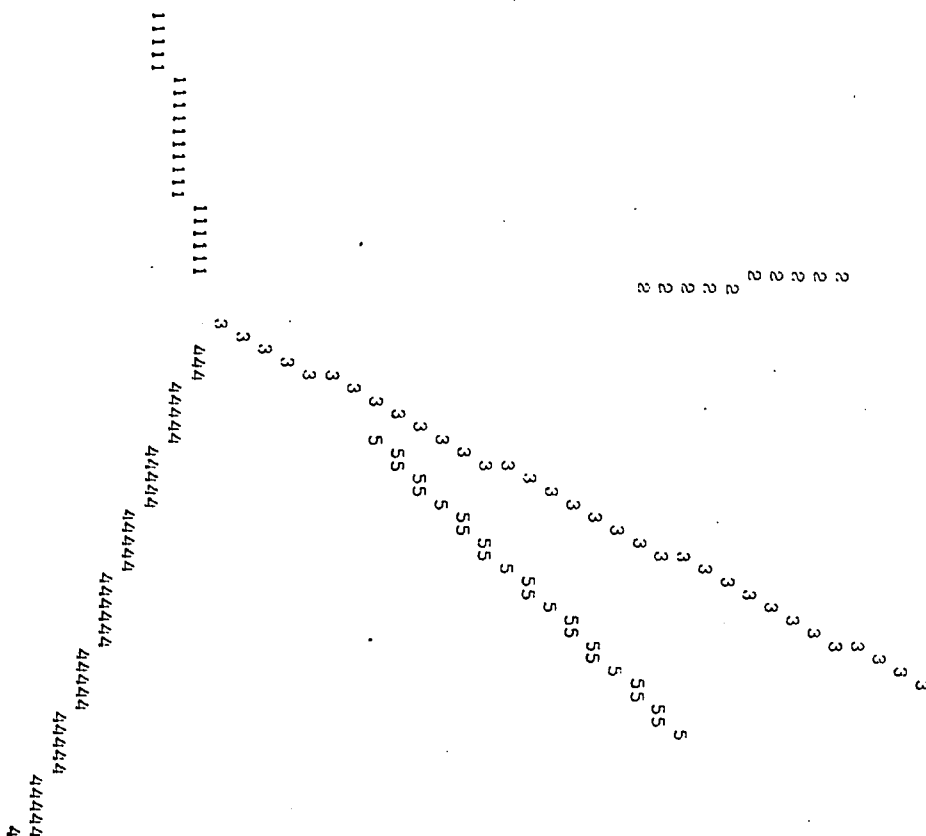


Figure 30 - Evénement cosmique (analysé avec filtrage, avec CRITER = 6 et 5 à l'étape 3)



VII - Recherche des étoiles.

C'est à cette dernière étape que le programme décide s'il y a événement (étoile) ou pas.

Cette section compte 6 critères faciles à modifier.* Les valeurs indiquées entre parenthèses sont celles qui ont été utilisées dans le programme, à la suite de l'expérience (limitée) acquise avec les événements codés manuellement.

Les critères choisis représentent un compromis. Des critères trop sévères feraient manquer plusieurs événements valables, tandis que des critères insuffisants donneraient plusieurs faux événements.

Lorsqu'il y a événement, le programme fait sonner la cloche du télétype et s'arrête. S'il n'y en a pas, il va au mot 0600, prêt à recevoir la prochaine image. Dans un système automatique, le tintement de la cloche ferait évidemment place à l'enregistrement dans une mémoire de masse de ses coordonnées.

Il y a automatiquement événement dès que le nombre de traces ("NTRF") dépasse "EXCES" (8). (Organigramme 22, (AA)).

S'il n'y a pas plus de "EXCES" traces, mais que deux traces forment un "V" ou un "T", il y a événement.

Pour cela, il faut qu'elles aient une longueur minimale

* Pour le PDP 8/E, ces critères peuvent être changés directement sur la console. "EXCES", "CRITRI", ..., "CRITR5" occupent respectivement les mots 0135, 0136, ..., 0142. Dans le programme FORTRAN, ils sont définis aux énoncés ISN0016, 0029, 0052, 0055, 0066, et 0070 respectivement

de "CRITR1" (16) et "CRITR2" (16) respectivement*. ((C) et (G)).

Leurs orientations ("D" et "DD") doivent différer d'au moins "CRITR3" (4) unités. ((H))

La distance entre un bout de la première et un bout de la seconde ne doit pas dépasser "CRITR4" (6) ((J)) (pour des "V") ou la distance** entre un bout de la première trace et la deuxième trace doit être égale ou inférieure à "CRITR5" (5). ((O)) (pour des "T").

Ces "T" sont souvent des événements où, après la collision, deux particules sont parties presque dos à dos. Dans ces cas, les deux traces semblent en être une seule pour le programme.

Le programme en langage d'assemblée est très semblable au programme "FORTRAN". L'organigramme 23 donne plus de détails.

* Ici, "CRITR1" et "CRITR2" sont égaux, mais "CRITR2" pourrait être réduit à 10, par exemple, afin d'accepter aussi les événements comportant une branche longue et une courte.

** Voir la définition de distance, entre un point et une trace, à la section III-C.

7-RECHERCHE DES ETOILES

AA)

NTRF > EXCES

OUI

EUREKA

Il y a une étoile.

NON

A)

Pour K=1 à NTRF

B)

```

TRACEF(K,1) → HI
TRACEF(K,2) → VI
TRACEF(K,3) → HF
TRACEF(K,4) → VF
TRACEF(K,5) → D
TRACEF(K,6) → NS
VF-VI → DV
HF-HI → DH
Max.( |DH| , DV) → DIFMAX

```

C)

NON

DIFMAX ≥ CRITR1 (=16)

OUI

D)

Pour MEC=1,2

=1

MEC

=2

```

HI → H
VI → V

```

```

HF → H
VF → V

```

E)

Pour KK=1 à NTRF (KK ≠ K)

F)

```

TRACEF(KK,1) → HHI
TRACEF(KK,2) → VVI
TRACEF(KK,3) → HHF
TRACEF(KK,4) → VVF
TRACEF(KK,5) → DD
TRACEF(KK,6) → NNS
VVF-VVI → DDV
HHF-HHI → DDH
Max.( |DDH| , DDV) → DDIFMA

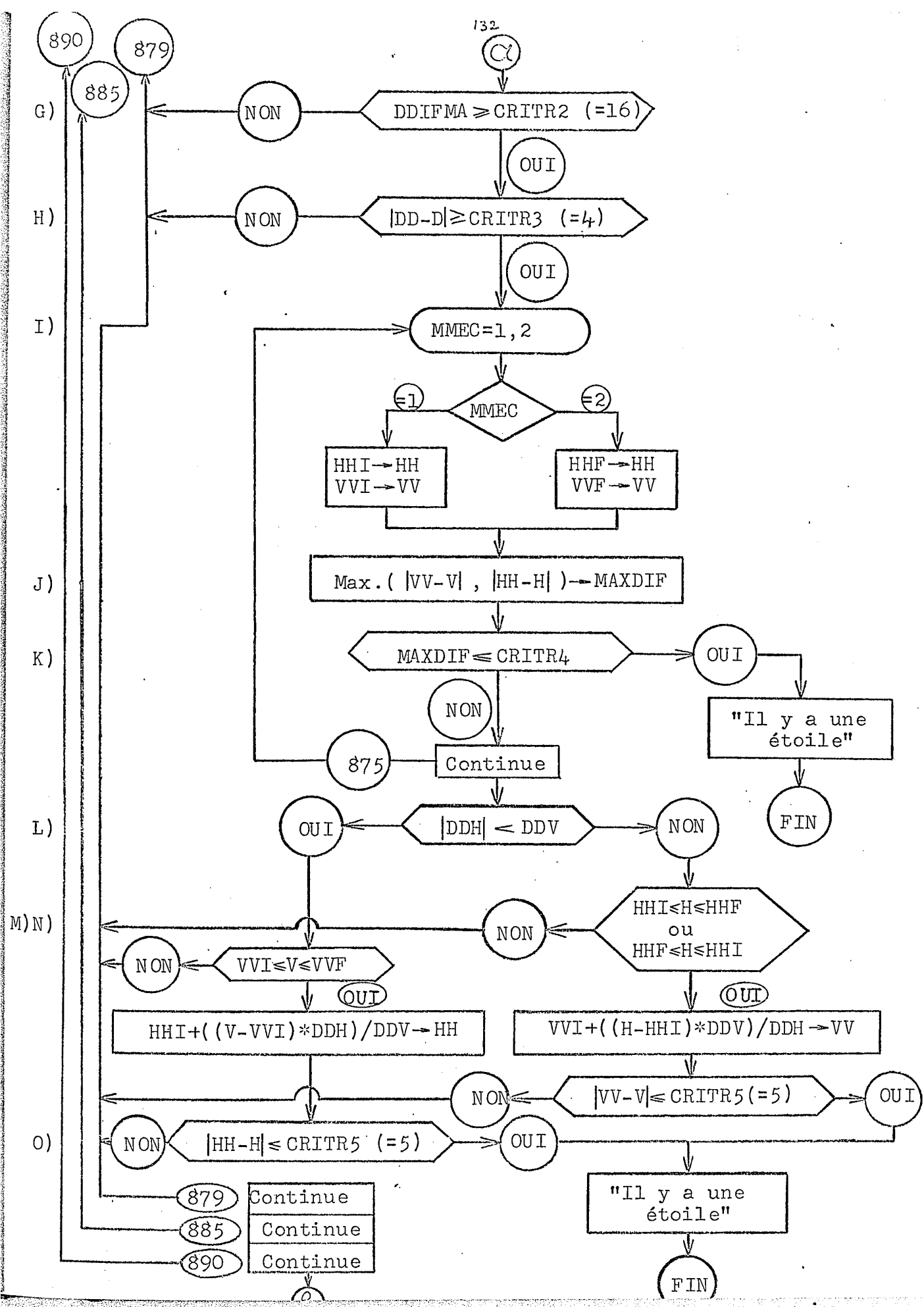
```

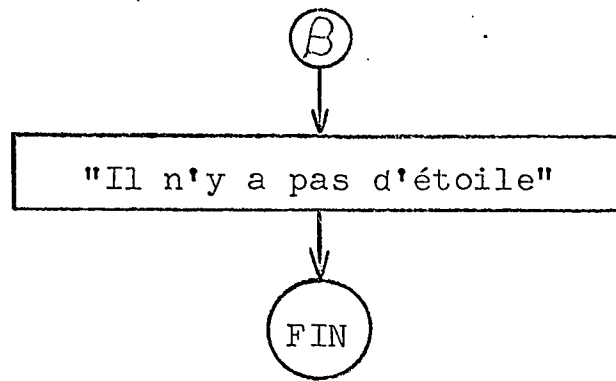
α

890

885

879





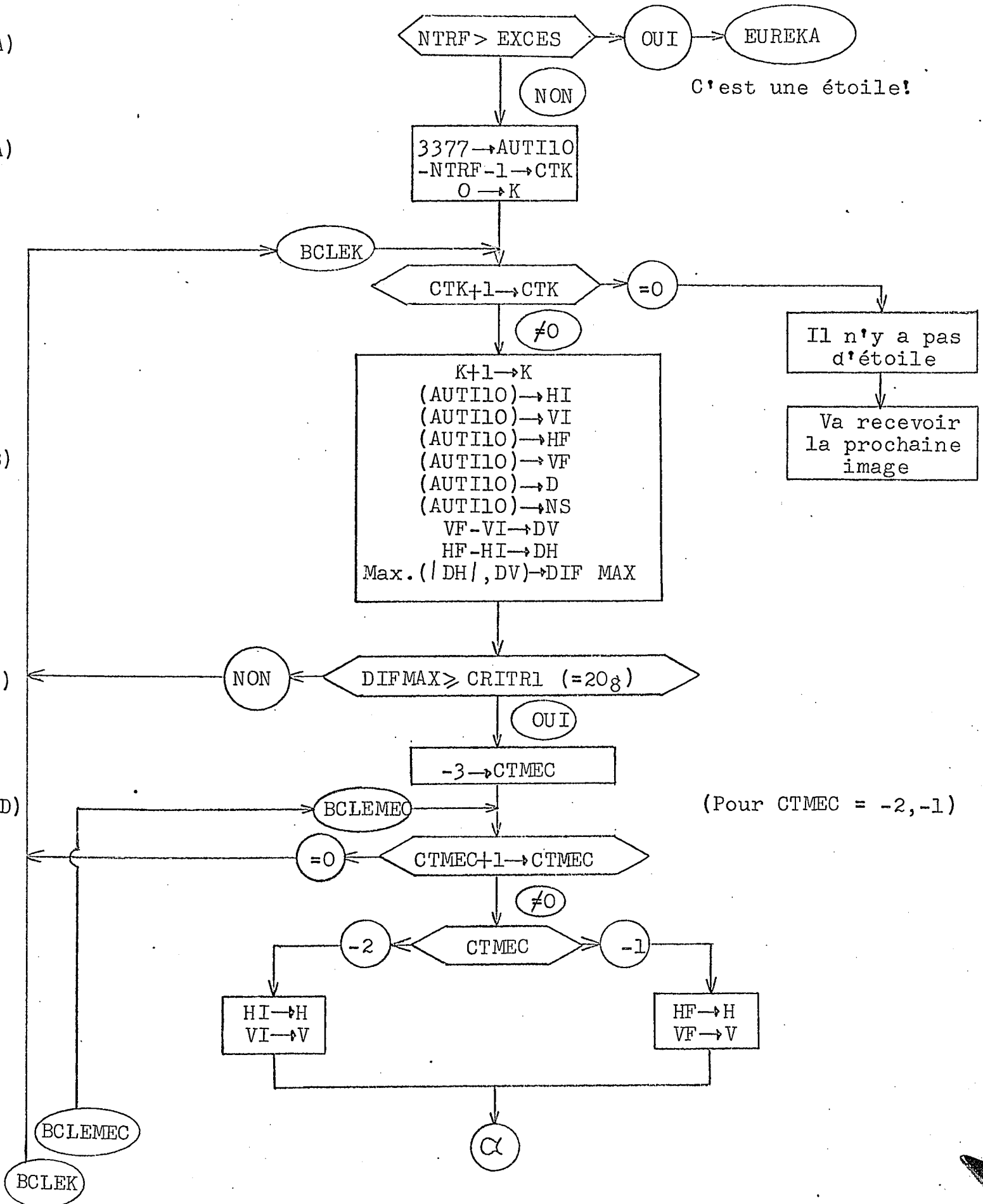
AA)

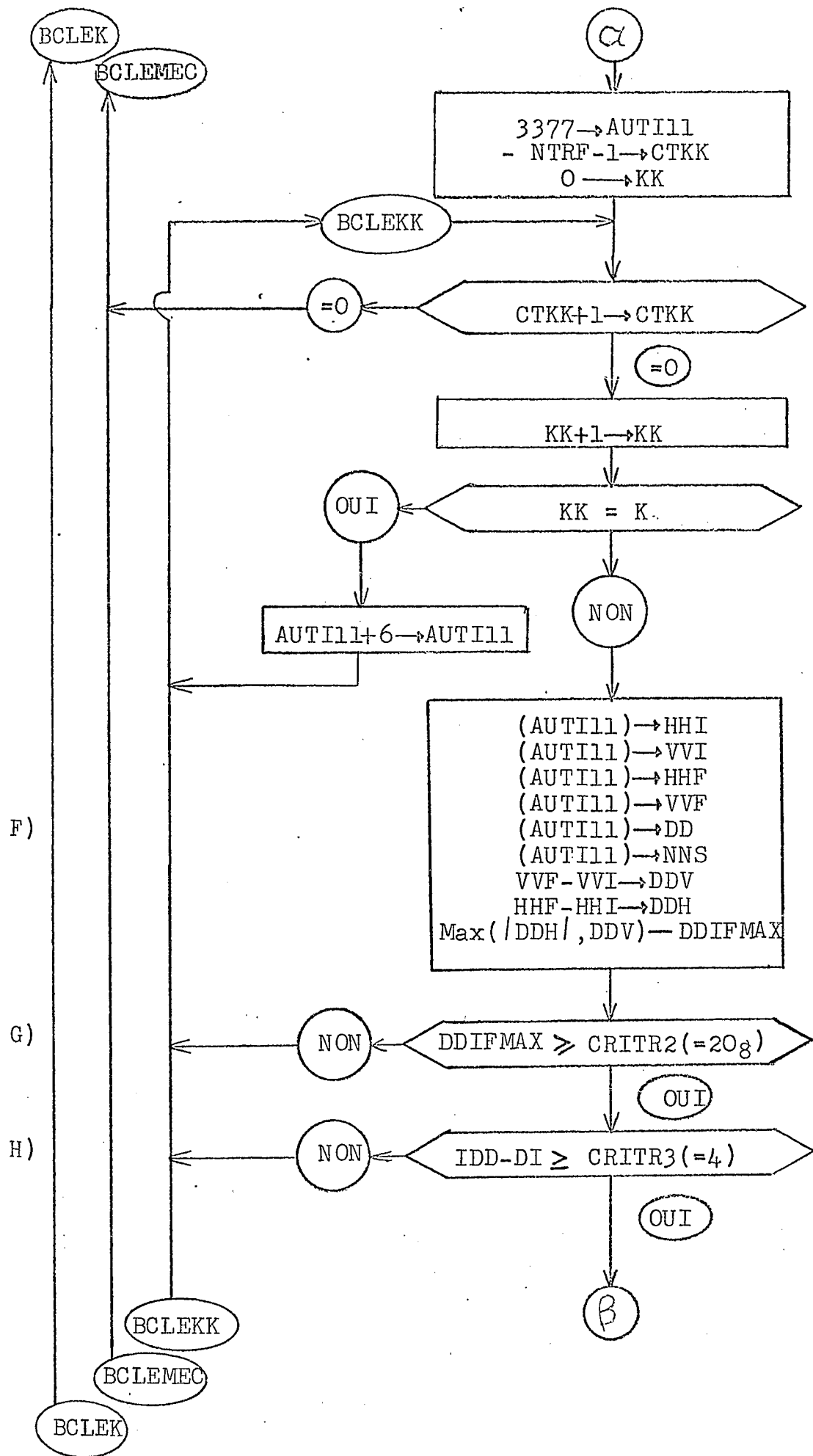
A)

B)

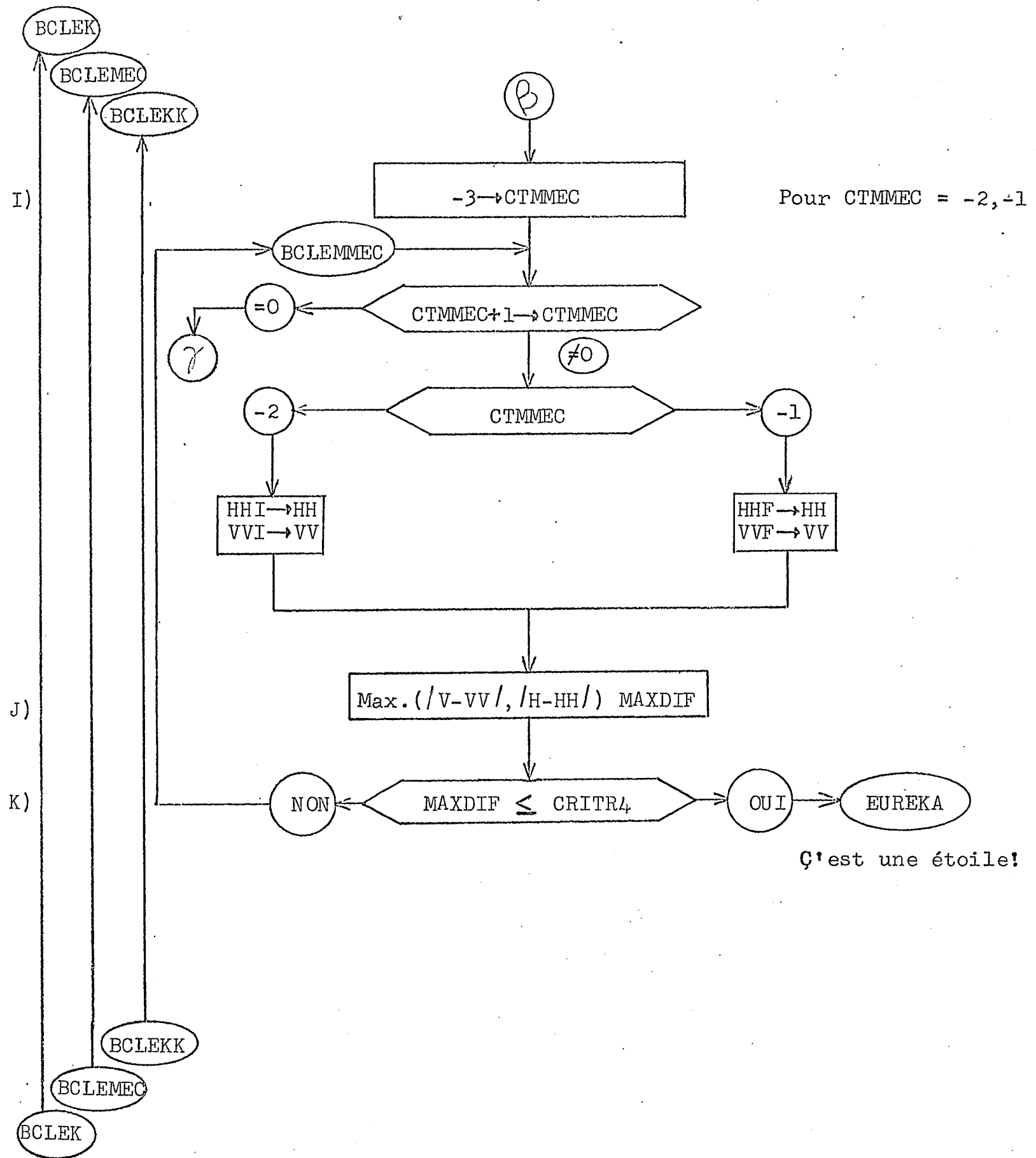
C)

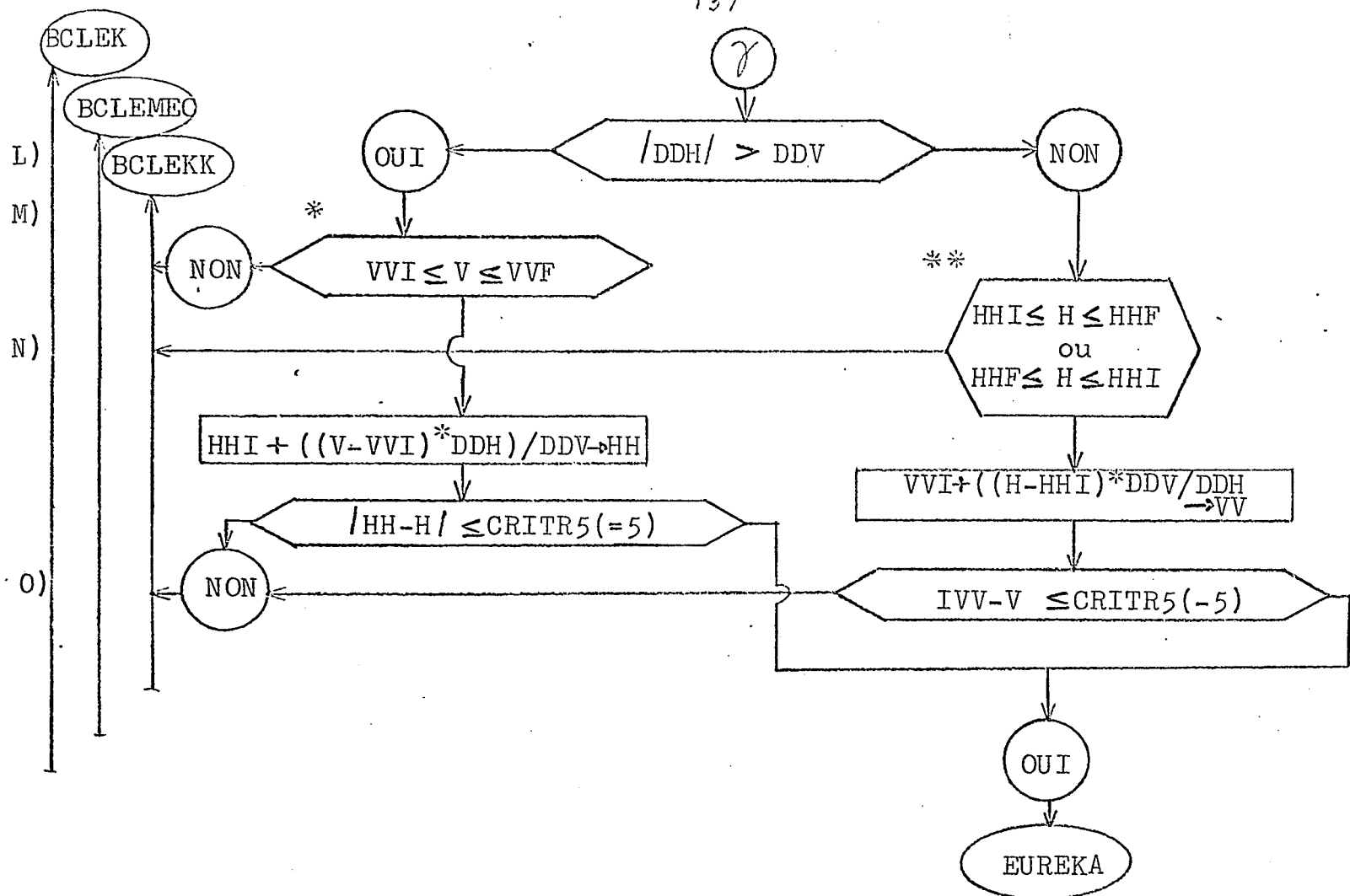
D)



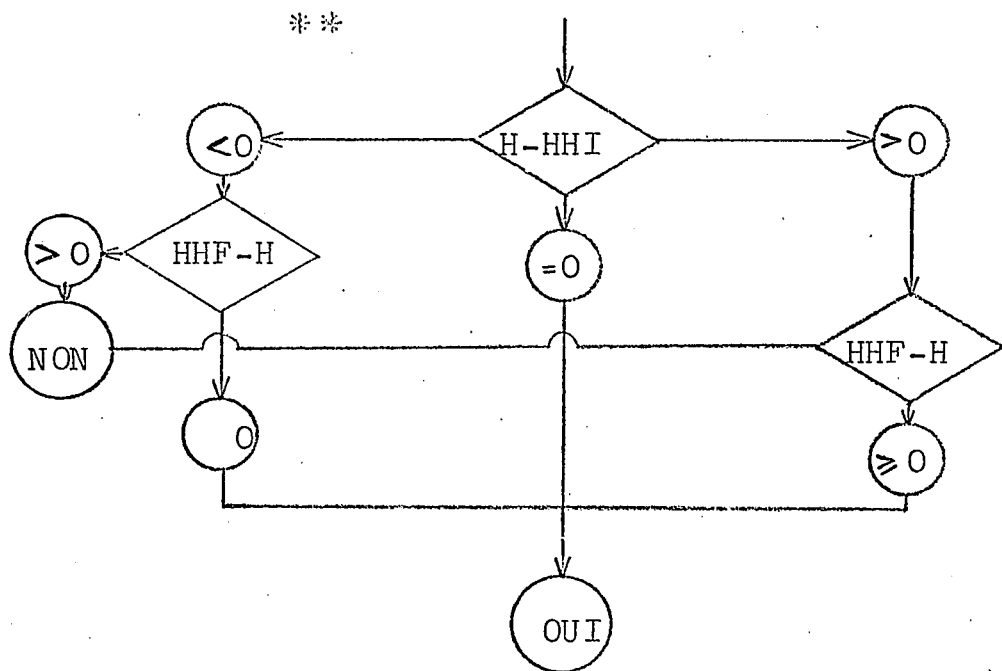
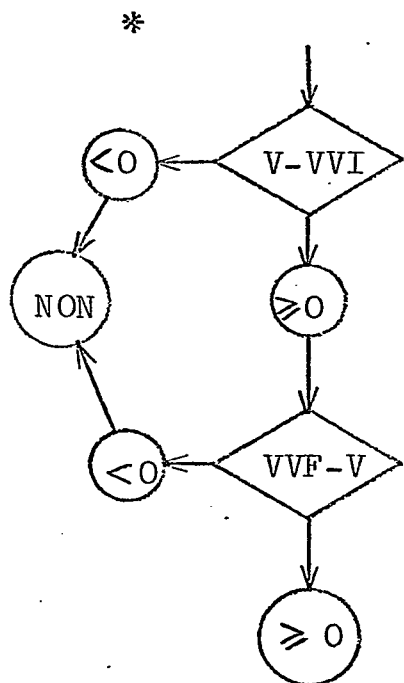


Pour $\text{KK} = 1$ à NTRF ($\text{KK} \neq \text{K}$)





C'est une étoile!



VIII-MESURES EXPERIMENTALES

Des essais ont été effectués afin de mettre le système à l'épreuve et de définir les conditions d'utilisation.

D'abord, dix étoiles, ayant trois branches longues et noires ou plus, ont été placées dans le champ de vision. Les conditions d'utilisation pour cet essai étaient les mêmes que celles qui ont donné l'image de la figure 4, ce qui donne une image transmise de $150 \times 120 \times 30\mu^3$. L'objectif se déplaçait lentement ($10\mu/\text{sec}$) verticalement et les images étaient continuellement transmises et analysées. La version du programme sans filtrage et avec les critères 6 et 5 à l'étape 3 (voir l'organigramme 12) a été utilisée.

Huit des dix étoiles ont été détectées dans ces conditions et les deux autres manquées, perdues dans le bruit dû aux taches ou à la grille.

Un second essai a consisté à explorer un centimètre carré dans une plaque cosmique avec le système et comparer les résultats avec ceux d'un observateur humain. L'objectif a été changé pour réduire le grossissement et transmettre au PDP 8/E une image de $200 \times 150 \times 70\mu^3$.

Pour explorer la plaque dans ces conditions, on la fait défiler à $100\mu/\text{sec}^*$ sous l'objectif. Un opérateur humain contrôle ce déplacement et l'arrête pour noter les événements détectés. Vu la grande profondeur de champ, il suffit de couvrir le centimètre carré deux fois: une fois pour le haut,

* Seule vitesse permise par les moteurs qui déplacent la platine.

une autre pour le bas de la plaque (qui a une épaisseur de 150 μ , après affaïssement).

La version du programme avec filtre et les critères 5 et 4 à l'étape 3 a été employée.

Tandis qu'au microscope optique, un observateur humain trouve 74 événements; le microscope à faisceau balayeur en détecte 44, soit une efficacité comparée de 60%. Environ cent faux événements sont toutefois signalés à cause de rayures, de taches ou de la grille.

La performance de l'appareil, comparé à celle de l'être humain est sensiblement celle à laquelle il fallait s'attendre, vu les caractéristiques de l'appareil et du programme décrites dans les chapitres précédents.

Les étoiles comptant au moins deux traces noires longues de 40 microns ou plus et situées dans le plan de la plaque sont détectées au moins aussi efficacement, sinon plus, que par un humain. Par contre les traces trop grises, trop courtes ou très plongeantes sont manquées.

On constate, en reportant sur un graphique les positions des événements trouvés par l'observateur humain et par l'appareil, que ce dernier a une efficacité nettement plus uniforme, plus régulière.

La profondeur de champ, beaucoup trop grande, explique que le bruit soit très considérable et qu'il y ait autant de faux événements détectés. Il faut, pour remédier à ce problème, des plaques très propres et sans grilles. Il peut aussi être utile, comme l'a montré l'essai mentionné plus haut, d'utiliser

un plus fort grossissement.

Le déplacement rapide de la platine ne laisse que deux secondes à l'ordinateur pour analyser l'image. ($200\mu \div 100\mu/\text{sec}$). Comme une image n'est transmise qu'après l'analyse de la précédente si le PDP 8/E met trop de temps à l'analyser, c'est une partie de la région adjacente qui sera manquée, n'étant jamais transmise ni analysée.

Le meilleur remède à cette difficulté nous semble être de déclencher une horloge au moment de la transmission de l'image et, si l'analyse prenait plus d'une seconde disons, que le moteur soit arrêté jusqu'à la fin de l'analyse.

IX- CONCLUSION

La sélection des événements est obtenue après quatre réductions de l'information contenue dans la plaque d'émulsion nucléaire.

D'abord, le microscope à faisceau balayeur présente, à l'écran, l'image d'une tranche d'un volume de, disons, $200 \times 200 \times 30 \mu^3$. La résolution est meilleure qu'un micron (figures 4 et 5).

Le discriminateur diminue la résolution dans le sens horizontal à plus d'un micron, sans changer celle dans le sens vertical (figure 6).

La transmission des données élimine la périphérie de cette image. La résolution horizontale reste la même, la verticale est diminuée d'un facteur 4 puisque seule une ligne sur quatre est transmise parmi les 300 de la région transmise de l'image. Ainsi, l'image est de 76×96 éléments noirs ou blancs soit 76 lignes de 96 bits (8 mots de 12 bits) ou 7296 bits (608 mots) au total (figure 8).

Enfin, cette image est réduite à une liste de traces (21, au plus) dont on ne conserve, essentiellement, que les points initial et final. (figures 26 à 30)

C'est à partir de cette liste de traces que le programme décide s'il y a événement.

Nous avons vu au chapitre précédent que l'appareil donnait une efficacité de 60%, comparé à un humain. L'uniformité du système s'est avérée supérieure à celle d'un humain. L'appareil trouve efficacement les événements comportant de grandes branches noires peu plongeantes, mais manque les petites étoiles. Il y a aussi la détection d'un certain nombre de faux événements, en

147

présence de la grille ou de taches, choses que le discriminateur et le filtre ne suffisent pas à enlever.

Diverses mesures seraient susceptibles d'améliorer les performances de l'appareil.

Moins d'événements seraient manqués si le mouvement de la platine était arrêté lorsque le programme met plus de temps que d'habitude à analyser une image. La décision d'utiliser le filtre ou pas, ainsi que la valeur des critères d'acceptation des segments, à l'étape 3, pourraient être fonction du nombre total de points noirs de l'image. Les points noirs sont d'ailleurs comptés immédiatement après la transmission des données. (voir l'appendice E).

Il faudrait utiliser des plaques propres et sans grille.*

Les méthodes de transmission des données (une ligne sur quatre de l'écran) et d'analyse font que les traces inclinées à moins de trente degrés par rapport au sens du balayage du spot sont manquées. Ceci indique qu'il serait certes utile d'effectuer deux explorations, en tournant la plaque de 90° pour la seconde.

Plusieurs autres méthodes pourraient être envisagées.

Il aurait été intéressant de voir si la recherche de bâtonnets (segments) plus longs que 10 unités aurait donné de meilleurs résultats. Avec une machine à 16 bits par mot (PDP 11 ou NOVA, par exemple), on pourrait chercher des bâtonnets de 14 unités selon 57 orientations différentes (plutôt que 10 et 41,

* Deux autres contraintes sont imposées par la nature du microscope à faisceau balayeur: les plaques jaunies ou trop épaisses (plus de 150 à 200 μ) sont inutilisables car elles absorbent trop la lumière bleue-violet du tube à rayons cathodiques.

respectivement) avec un sous-programme analogue à notre sous-routine SUIVRE.

Le fait de développer l'algorithme d'abord, en FORTRAN pour un ordinateur IBM 360/65 a permis de tester diverses méthodes avec un effort relativement modeste. Ceci a plus que compensé la nécessité de coder des images test à la main. En outre, la version FORTRAN est très utile comme documentation du programme. Ce n'est qu'une fois l'algorithme bien au point qu'il a été transposé et optimisé en langage d'assemblée pour le mini-ordinateur PDP 8/E.* Ceci est indispensable pour un mini-ordinateur. Le programme s'exécute en quelques secondes et n'occupe que 3K de mémoire, l'autre K est occupé par les données (l'image) et les résultats (les traces).

L'emploi d'un mini-ordinateur, programmé de façon à en tirer le maximum, a ainsi permis la réalisation d'un système peu coûteux capable d'effectuer la tâche fastidieuse du dépouillement des plaques d'émulsion nucléaire.

* Pour adapter l'algorithme à d'autres mini-ordinateurs, tels le PDP 11 ou le NOVA, il faudrait refaire complètement le programme. Pour un PDP 8, PDP 8/I ou PDP 8/L, il suffirait de quelques modifications mineures. Voir, à l'appendice F, "Mode d'emploi".

BIBLIOGRAPHIE

1. P.V.C. Hough et B.W. Powell, Nuovo Cim. 18, 1184 (1960)
2. P. Villemoes, Proc. of the 1970 CERN Computing and Data Processing School. 105. CERN, 71-6, (1971)
R.K. Böck et J.Z. al. CERN Courier 3, 70 (1972)
W. Blair et J.C. Gouache. CERN Courier 3, 72 (1972)
D. Lord CERN Courier 3, 85 (1972)
3. Methods in Computational Physics 5
Rédacteurs: B. Alder, S. Fernback, M. Rotenberg
Academic Press, New York (1966)
4. Programming for Flying Spot Devices
Conférence - Bologne, (1964)
Rédacteurs: W.G. Moorhead, B.W. Powell, CERN, 65-11 (1965)
5. Programming for Flying Spot Devices. Conf. (1965)
Rédacteur: D.H. Tycko, New York (1966)
6. Int. Conf. (1967) Programming for Flying Spot Devices
Rédacteurs: B.W. Powell, P. Seyboth, Munich (1967)
7. Proc. Int. Conf. (1968) Advanced Data Processing for Bubble and Spark Chamber
Rédacteur: R.J. Royston Argonne ANL 7515
8. Proc. Int. Conf. (1970) Data Handling Systems in High Energy Physics, Cambridge
Rédacteurs: D.H. Lord, B.W. Powell, CERN, 70-21 (1970)
9. R.B. Marr et G. Rabinowitz, Meth. Comp. Phys. 5, 213
Academic Press, New York (1966).
K. Abrams, J. Cockrill, P. Connally, D. Crennal, P. Hough, L. Potter, D. Stonehill, R. Strand et W. Thompson
Programming for Flying Spot Devices, Conf. Bologne (1964)
CERN, 65-11, p. 97 (1965).
N. Webre, D. Burd, L. Garder, D. Tycko, B. Zaslavsky;
Programming for Flying Spot Devices, Conf. (1964) Bologne.
CERN 65-11, p. 153 (1965).
G. Rabinowitz, D. Burd, R. Teitelbaum
Int. Conf. (Munich 1967) Programming for Flying Spot Devices.
Munich, 390 (1967).
B. Burd, L. Gardner, D. Tycko, N. Webre, B. Zaslavsky
Programming for Flying Spot Devices, Conf. (1965) Columbia.
New York 1.4 - 1.6 (1966)
B. Weneser, Programming for Flying Spot Devices, Conf. (1965)
Columbia. New York 5.1 (1966)
G. Rabinowitz, Programming for Flying Spot Devices, Conf. (1965)
Columbia. New York 5.2 (1966)

C. Baltay, M. Brod, D. Burd, W. Cooper, L. Garder, G. Rabinowitz. Proc. Int. Conf. (1968) Advanced Data Processing for Bubble and Spark Chambers. Argonne National Laboratory ANL 7515 p. 266 (1969).
C. Baltay, F. Alvarez, D. Burd, W.A. Cooper, J. Feinman, P. Franzini, L.C. Garder, M. Habili, R. Newman, E. Petilli, G. Rabinowitz, G. Weinstein. Int. Conf. on Data Handling Systems in High Energy Physics. Cambridge (1970). CERN, 70-21 p. 311 (1970).

10. D. Hodges, Int. Conf. (Munich 1967) Programming for Flying Spot Devices. Munich, 329 (1967)
R.K. Clark Ibid. page 336.
J.G. Loken et W.W.M. Allison Proc. Int. Conf. (1968) Advanced Data Processing for Bubble and Spark Chambers. Argonne National Laboratory ANL 7515; pp. 244; 390 (1969)
W.W.M. Allison, F. Beck, G. Charlton, D. Hodges, J.G. Loken, B. Musgrave, H.B. Phillips, R. Royston et R.A. Sunde, Int. Conf. (1970) Data Handling Systems in High-Energy Physics CERN, 70-21, p. 325 (1970).
11. C. Guignard et A. Volte Coll. int. sur la reconnaissance de forme. Grenoble 1968.
Rédacteur M. Henry, L.E.T.I. Grenoble 224 (1968).
B. Equer, C. Guignard, G. Reboul A. Volte Proc. Int. Conf. (1968) Advanced Data Processing for Bubble and Spark Chambers. Argonne National Laboratory ANL 7515, p. 6 (1969).
B. Equer, G. Fontaine, G. Reboul Int. Conf. (1970) on Data Handling Systems in High-Energy Physics. CERN, 70-21 371;383 (1970).
12. M. Goldwasser, J.C. Michau, J. Mullie, B. Pichard et G. Riols. BIST (CEA) 177, 47 (1967).
A. Dillet, M. Goldwasser, J.C. Michau, J. Mullie, B. Pichard, et G. Riols. Int. Conf. (1970) Data Handling Systems in High-Energy Physics. CERN 70-21, p. 859 (1970).
13. Photographie Corpusculaire II; Conf. int. Montréal (1958).
Rédacteur: P. Demers. Presses universitaires de Montréal, 5e partie, (1959).
Photographie Corpusculaire III; Conf. int. Moscou (1960).
Rédacteur: P. Demers. Presses universitaires de Montréal. 5e partie (1964).
Korpuskularphotographie IV; Conf. int. Munich (1962).
Rédacteurs: H. Frieser et G. Heimann, Munich, section V, (1963).
Vth Int. Conf. on Nuclear Photography, CERN (1964),
Rédacteur: E. Dahl-Jensen. CERN, 65-4, section VII (1965).
VIth Int. Conf. on Corpuscular Photography. Florence (1966).
Rédacteur: M. Della Corte. Editions C.E.P.I. Section "Automation" (1966).

14. J. Rüfenacht, R. Weill, M. Gailloud et J. Lagorse.
Phot. Corp. II, Montréal, 349 (1959).
C. Galtieri, A. Manfredini et V. Rossi.
Phot. Corp. III, Montréal, 378 (1964).
15. W. H. Barkas, H.H. Heckman, J.C. Hodges, Korpuskularphoto-
graphie IV, Munich (1962) Munich, 416 (1963)
H. Annoni, G. Baumann, H. Braun, E. Christophel, P. Volmer.
Vth Int. Conf. on Nucl. Phot. CERN, 65-4, VII-1 (1965).
16. A.E. Voronkov, I.O. Murin, L.V. Sukhov et I.V. Chchtranikh.
Phot. Corp. III, Moscou (1960), Montréal 315 (1964).
J.P. Barbier, A. Bernheim, J.C. Fayalle, M. Morand.
Nucl. Instr. and Meth. 49 38, (1967).
17. P.V.C. Hough Nucl. Instr. and Meth 6, 33 (1960)
J.R. Erskine et R.H. Vonderohe. Nucl. Instr. and Meth. 81,
221 (1970).
18. H. Brettel, C. Carathassis et L. Kollmeder. Int. Conf. (1970)
on Data Handling Systems in High-Energy Physics. CERN 70-21
851 (1970).
19. F. Roberts et J.Z. Young. Proc. Inst. Elect. Eng. 99 (3A):
747 (1952).
20. J.G. McEwen et J. Hébert. Phot. Corp. II, Montréal, PUM,
370 (1959).
21. B. Wiggins, Thèse, Université d'Ottawa (1966).
22. PDP8/E Small Computer Handbook Digital Equipment Corporation,
Maynard (1971).
23. Introduction to Programming. Digital Equipment Corporation,
Maynard (1970).

Appendice A.
Interpolation selon "D"

APPENDICES

Appendice A INTERPOLATION SELON "D"

1.	Programme FORTRAN.....	A-1
2.	Table d'interpolation.....	A-2
3.	Matrices "NECV" et "NECH".....	A-3
4.	Matrice "NEC".....	A-4
5.	Sous-routine "SUIVRE".....	A-5
6.	Sous-routine "MODE 1" et "MODE 2"....	A-6
7.	Sous-routine "AVANT" et "ARRIERE"....	A-7

COMPILER OPTIONS - NAME= MAIN,OPT=02,LINECNT=45,SIZE=3000K,
SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NGEDIT,NOID,NOXREF

C

C NOUVELLE MATRICE "NEC"

C

C IMPLICIT INTEGER (A-Z)

LOGICAL NEC(12,41),NECV(10,41),NECH(10,41)

DIMENSION OCTAL(4),HJ(10,41),VJ(10,41)

C

DO 1 D=1,10

NEC(12,D)=.FALSE.

NECV(11,D)=.FALSE.

DO 1 J=1,10

HJ(J,D)=J

VJ(J,D)=((D-1)*J+5)/10

1 CONTINUE

C

DO 2 D=11,31

NEC(12,D)=.FALSE.

IF (D.GT.21) NEC(12,D)=.TRUE.

NECV(11,D)=.TRUE.

SG=1

IF (D.GT.21) SG=-1

DO 2 J=1,10

VJ(J,D)=J

HJ(J,D)=((21-D)*J+5*SG)/10

2 CONTINUE

C

DO 3 D=32,41

NEC(12,D)=.TRUE.

NECV(11,D)=.FALSE.

DO 3 J=1,10

HJ(J,D)=-J

VJ(J,D)=((41-D)*J+5)/10

3 CONTINUE

C

DO 24 D=1,41

DO 24 J=1,10

VP=0

IF (J.NE.1) VP=VJ(J-1,D)

NECV(J,D)=.FALSE.

IF (VJ(J,D).NE.VP)NECV(J,D)=.TRUE.

HP=0

```

ISN 0040      IF (J,NE,1) HP=HJ(J-1,D)
ISN 0042      NECH(J,D)=.FALSE.
ISN 0043      IF (HJ(J,D).NE.HP) NECH(J,D)=.TRUE.
C
ISN 0045      NEC(J,D)=NECV(J,D)
ISN 0046      IF (11.LE.D.AND.D.LE.31) NEC(J,D)=NECH(J,D)
ISN 0048      24 CONTINUE
C
C      IMPRIME LES COORDONNEES DES 10 POINTS DU TRAIT D'ORIENTATION D PARTANT DE
C      (0,0).
C
ISN 0049      WRITE (3,185)
ISN 0050      WRITE (3,12) (J,J=1,10)
ISN 0051      12 FORMAT (//20X,'COORDONNEES DU J EME POINT SELON L ORIENTATION D'//
          1,9X,'D',4X,10(5X,'J:',12))
ISN 0052      DO 113 D=1,41
ISN 0053      113 WRITE (3,13) D, (VJ(J,D),HJ(J,D),J=1,10)
ISN 0054      13 FORMAT (//110,5X, 9(2X, '(',12,',',12,',') ),2X,'(',12,',',13,',')'
C
C      IMPRIMER LES MATRICES "NECV" ET "NECH" UTILISEES PAR LE PROGRAMME FORTRAN.
C
ISN 0055      WRITE (3,185)
ISN 0056      WRITE (3,10)
ISN 0057      10 FORMAT (//5X,'MATRICES',18X,'NECV',19X,'ET',18X,'NECH'//)
ISN 0058      WRITE (3,112) (J,J=1,10), (K,K=1,10)
ISN 0059      112 FORMAT ( /9X,'J',2(9X,10I3)//9X,'D'//)
ISN 0060      DO 111 D=1,41
ISN 0061      111 WRITE (3,11) D, (NECV(J,D),J=1,10),(NECH(J,D),J=1,10)
ISN 0062      11 FORMAT (110,2(9X,10I3)//)
C
C      IMPRIMER LA MATRICE "NEC" SOUS FORME LOGIQUE (BINAIRE) ET L'EQUIVALENT OCTAL
C      CETTE MATRICE "NEC" EST CELLE UTILISEE PAR LE PROGRAMME DE LA PDP8/E.
C
ISN 0063      WRITE (3,185)
ISN 0064      185 FORMAT ('1')
ISN 0065      WRITE (3,15)
ISN 0066      15 FORMAT (//15X,'MATRICE NEC'//)
ISN 0067      WRITE (3,14) (J,J=1,12)
ISN 0068      14 FORMAT (5X,'J=',5X,12I3 //)
ISN 0069      DO 6 D=1,41
ISN 0070      DO 5 Z=1,4
ISN 0071      OCTAL(Z)=0
ISN 0072      DO 4 J=1,3

```

ISN 0073 IF (NEC((Z-1)*3+J,D)) OCTAL(Z)=OCTAL(Z)+2**(3-J)
ISN 0075 4 CONTINUE
ISN 0076 5 CONTINUE
ISN 0077 6 WRITE (3,7) D,(NEC(J,D),J=1,12),(OCTAL(Z),Z=1,4)
ISN 0078 7 FORMAT (1X,/,2X,'D=',13,5X,12L3,5X,4I1)
ISN 0079 STOP
ISN 0080 END

COORDONNÉES DU J ÈME POINT SELON L ORIENTATION D.

D	J: 1	J: 2	J: 3	J: 4	J: 5	J: 6	J: 7	J: 8	J: 9	J: 10
1	(0, 1)	(0, 2)	(0, 3)	(0, 4)	(0, 5)	(0, 6)	(0, 7)	(0, 8)	(0, 9)	(0, 1
2	(0, 1)	(0, 2)	(0, 3)	(0, 4)	(1, 5)	(1, 6)	(1, 7)	(1, 8)	(1, 9)	(1, 1
3	(0, 1)	(0, 2)	(1, 3)	(1, 4)	(1, 5)	(1, 6)	(1, 7)	(2, 8)	(2, 9)	(2, 1
4	(0, 1)	(1, 2)	(1, 3)	(1, 4)	(2, 5)	(2, 6)	(2, 7)	(2, 8)	(3, 9)	(3, 1
5	(0, 1)	(1, 2)	(1, 3)	(2, 4)	(2, 5)	(2, 6)	(3, 7)	(3, 8)	(4, 9)	(4, 1
6	(1, 1)	(1, 2)	(2, 3)	(2, 4)	(3, 5)	(3, 6)	(4, 7)	(4, 8)	(5, 9)	(5, 1
7	(1, 1)	(1, 2)	(2, 3)	(2, 4)	(3, 5)	(4, 6)	(4, 7)	(5, 8)	(5, 9)	(6, 1
8	(1, 1)	(1, 2)	(2, 3)	(3, 4)	(4, 5)	(4, 6)	(5, 7)	(6, 8)	(6, 9)	(7, 1
9	(1, 1)	(2, 2)	(2, 3)	(3, 4)	(4, 5)	(5, 6)	(5, 7)	(6, 8)	(7, 9)	(8, 1
10	(1, 1)	(2, 2)	(3, 3)	(4, 4)	(5, 5)	(5, 6)	(6, 7)	(7, 8)	(8, 9)	(9, 1
11	(1, 1)	(2, 2)	(3, 3)	(4, 4)	(5, 5)	(6, 6)	(7, 7)	(8, 8)	(9, 9)	(10, 1
12	(1, 1)	(2, 2)	(3, 3)	(4, 4)	(5, 5)	(6, 5)	(7, 6)	(8, 7)	(9, 8)	(10,
13	(1, 1)	(2, 2)	(3, 2)	(4, 3)	(5, 4)	(6, 5)	(7, 6)	(8, 6)	(9, 7)	(10,
14	(1, 1)	(2, 1)	(3, 2)	(4, 3)	(5, 4)	(6, 4)	(7, 5)	(8, 6)	(9, 6)	(10,
15	(1, 1)	(2, 1)	(3, 2)	(4, 2)	(5, 3)	(6, 4)	(7, 4)	(8, 5)	(9, 5)	(10,
16	(1, 1)	(2, 1)	(3, 2)	(4, 2)	(5, 3)	(6, 3)	(7, 4)	(8, 4)	(9, 5)	(10,
17	(1, 0)	(2, 1)	(3, 1)	(4, 2)	(5, 2)	(6, 2)	(7, 3)	(8, 3)	(9, 4)	(10,
18	(1, 0)	(2, 1)	(3, 1)	(4, 1)	(5, 2)	(6, 2)	(7, 2)	(8, 2)	(9, 3)	(10,
19	(1, 0)	(2, 0)	(3, 1)	(4, 1)	(5, 1)	(6, 1)	(7, 1)	(8, 2)	(9, 2)	(10,
20	(1, 0)	(2, 0)	(3, 0)	(4, 0)	(5, 1)	(6, 1)	(7, 1)	(8, 1)	(9, 1)	(10,
21	(1, 0)	(2, 0)	(3, 0)	(4, 0)	(5, 0)	(6, 0)	(7, 0)	(8, 0)	(9, 0)	(10,
22	(1, 0)	(2, 0)	(3, 0)	(4, 0)	(5, -1)	(6, -1)	(7, -1)	(8, -1)	(9, -1)	(10, -
23	(1, 0)	(2, 0)	(3, -1)	(4, -1)	(5, -1)	(6, -1)	(7, -1)	(8, -2)	(9, -2)	(10, -
24	(1, 0)	(2, -1)	(3, -1)	(4, -1)	(5, -2)	(6, -2)	(7, -2)	(8, -2)	(9, -3)	(10, -
25	(1, 0)	(2, -1)	(3, -1)	(4, -2)	(5, -2)	(6, -2)	(7, -3)	(8, -3)	(9, -4)	(10, -
26	(1, -1)	(2, -1)	(3, -2)	(4, -2)	(5, -3)	(6, -3)	(7, -4)	(8, -4)	(9, -5)	(10, -
27	(1, -1)	(2, -1)	(3, -2)	(4, -2)	(5, -3)	(6, -4)	(7, -4)	(8, -5)	(9, -5)	(10, -
28	(1, -1)	(2, -1)	(3, -2)	(4, -3)	(5, -4)	(6, -4)	(7, -5)	(8, -6)	(9, -6)	(10, -
29	(1, -1)	(2, -2)	(3, -2)	(4, -3)	(5, -4)	(6, -5)	(7, -6)	(8, -6)	(9, -7)	(10, -
30	(1, -1)	(2, -2)	(3, -3)	(4, -4)	(5, -5)	(6, -5)	(7, -6)	(8, -7)	(9, -8)	(10, -
31	(1, -1)	(2, -2)	(3, -3)	(4, -4)	(5, -5)	(6, -6)	(7, -7)	(8, -8)	(9, -9)	(10, -1
32	(1, -1)	(2, -2)	(3, -3)	(4, -4)	(5, -5)	(5, -6)	(6, -7)	(7, -8)	(8, -9)	(9, -1
33	(1, -1)	(2, -2)	(3, -3)	(3, -4)	(4, -5)	(5, -6)	(6, -7)	(6, -8)	(7, -9)	(8, -1
34	(1, -1)	(1, -2)	(2, -3)	(3, -4)	(4, -5)	(4, -6)	(5, -7)	(6, -8)	(6, -9)	(7, -1
35	(1, -1)	(1, -2)	(2, -3)	(2, -4)	(3, -5)	(4, -6)	(4, -7)	(5, -8)	(5, -9)	(6, -1
36	(1, -1)	(1, -2)	(2, -3)	(2, -4)	(3, -5)	(3, -6)	(4, -7)	(4, -8)	(5, -9)	(5, -1
37	(0, -1)	(1, -2)	(1, -3)	(2, -4)	(2, -5)	(2, -6)	(3, -7)	(3, -8)	(4, -9)	(4, -1
38	(0, -1)	(1, -2)	(1, -3)	(1, -4)	(2, -5)	(2, -6)	(2, -7)	(2, -8)	(3, -9)	(3, -1
39	(0, -1)	(0, -2)	(1, -3)	(1, -4)	(1, -5)	(1, -6)	(1, -7)	(2, -8)	(2, -9)	(2, -1
40	(0, -1)	(0, -2)	(0, -3)	(0, -4)	(1, -5)	(1, -6)	(1, -7)	(1, -8)	(1, -9)	(1, -1
41	(0, -1)	(0, -2)	(0, -3)	(0, -4)	(0, -5)	(0, -6)	(0, -7)	(0, -8)	(0, -9)	(0, -1

J	1	2	3	4	5	6	7	8	9	10		1	2	3	4	5	6	7	8	9	10
D																					
1	F	F	F	F	F	F	F	F	F	F		T	T	T	T	T	T	T	T	T	T
2	F	F	F	F	T	F	F	F	F	F		T	T	T	T	T	T	T	T	T	T
3	F	F	T	F	F	F	F	T	F	F		T	T	T	T	T	T	T	T	T	T
4	F	T	F	F	T	F	F	F	T	F		T	T	T	T	T	T	T	T	T	T
5	F	T	F	T	F	F	T	F	T	F		T	T	T	T	T	T	T	T	T	T
6	T	F	T	F	T	F	T	F	T	F		T	T	T	T	T	T	T	T	T	T
7	T	F	T	F	T	T	F	T	F	T		T	T	T	T	T	T	T	T	T	T
8	T	F	T	T	T	F	T	T	F	T		T	T	T	T	T	T	T	T	T	T
9	T	T	F	T	T	T	T	F	T	T		T	T	T	T	T	T	T	T	T	T
10	T	T	T	T	T	F	T	T	T	T		T	T	T	T	T	T	T	T	T	T
11	T	T	T	T	T	T	T	T	T	T		T	T	T	T	T	T	T	T	T	T
12	T	T	T	T	T	T	T	T	T	T		T	T	T	T	F	T	T	T	T	T
13	T	T	T	T	T	T	T	T	T	T		T	T	F	T	T	T	F	T	T	T
14	T	T	T	T	T	T	T	T	T	T		T	F	T	T	F	T	T	F	T	T
15	T	T	T	T	T	T	T	T	T	T		T	F	T	F	T	F	T	F	T	T
16	T	T	T	T	T	T	T	T	T	T		T	F	T	F	T	F	T	F	T	F
17	T	T	T	T	T	T	T	T	T	T		F	T	F	T	F	F	T	F	T	F
18	T	T	T	T	T	T	T	T	T	T		F	T	F	F	T	F	F	F	T	F
19	T	T	T	T	T	T	T	T	T	T		F	F	T	F	F	F	F	T	F	F
20	T	T	T	T	T	T	T	T	T	T		F	F	F	F	T	F	F	F	F	F
21	T	T	T	T	T	T	T	T	T	T		F	F	F	F	F	F	F	F	F	F
22	T	T	T	T	T	T	T	T	T	T		F	F	F	F	T	F	F	F	F	F
23	T	T	T	T	T	T	T	T	T	T		F	F	T	F	F	F	F	T	F	F
24	T	T	T	T	T	T	T	T	T	T		F	T	F	F	T	F	F	F	T	F
25	T	T	T	T	T	T	T	T	T	T		F	T	F	T	F	F	T	F	T	F
26	T	T	T	T	T	T	T	T	T	T		T	F	T	F	T	F	T	F	T	F
27	T	T	T	T	T	T	T	T	T	T		T	F	T	F	T	T	F	T	F	T
28	T	T	T	T	T	T	T	T	T	T		T	F	T	T	T	F	T	T	F	T
29	T	T	T	T	T	T	T	T	T	T		T	T	F	T	T	T	F	T	T	T
30	T	T	T	T	T	T	T	T	T	T		T	T	T	T	T	F	T	T	T	T
31	T	T	T	T	T	T	T	T	T	T		T	T	T	T	T	T	T	T	T	T
32	T	T	T	T	T	F	T	T	T	T		T	T	T	T	T	T	T	T	T	T
33	T	T	F	T	T	T	T	F	T	T		T	T	T	T	T	T	T	T	T	T
34	T	F	T	T	T	F	T	T	F	T		T	T	T	T	T	T	T	T	T	T
35	T	F	T	F	T	T	F	T	F	T		T	T	T	T	T	T	T	T	T	T
36	T	F	T	F	T	F	T	F	T	F		T	T	T	T	T	T	T	T	T	T
37	F	T	F	T	F	F	T	F	T	F		T	T	T	T	T	T	T	T	T	T
38	F	T	F	F	T	F	F	F	T	F		T	T	T	T	T	T	T	T	T	T
39	F	F	T	F	F	F	F	T	F	F		T	T	T	T	T	T	T	T	T	T
40	F	F	F	F	T	F	F	F	F	F		T	T	T	T	T	T	T	T	T	T
41	F	F	F	F	F	F	F	F	F	F		T	T	T	T	T	T	T	T	T	T

MATRICE NEC

[illegible]

*4301

4301	0000	0000
4302	0200	0200
4303	1020	1020
4304	2210	2210
4305	2450	2450
4306	5250	5250
4307	5324	5324
4310	5664	5664
4311	6754	6754
4312	7674	7674
4313	7776	7776
4314	7676	7676
4315	6756	6756
4316	5666	5666
4317	5326	5326
4320	5252	5252
4321	2452	2452
4322	2212	2212
4323	1022	1022
4324	0202	0202
4325	0002	0002
4326	0203	0203
4327	1023	1023
4330	2213	2213
4331	2453	2453
4332	5253	5253
4333	5327	5327
4334	5667	5667
4335	6757	6757
4336	7677	7677
4337	7777	7777
4340	7675	7675
4341	6755	6755
4342	5665	5665
4343	5325	5325
4344	5251	5251
4345	2451	2451
4346	2211	2211
4347	1021	1021
4350	0201	0201
4351	0001	0001

/SOUS-ROUTINE SUIVRE

*1000

1000 0000 SUIVRE,0
 1001 1016 TAD MOTI
 1002 3025 DCA MOT
 1003 1424 TAD I D
 1004 7421 MQL
 1005 7701 ACL
 1006 7110 CLL RAR
 1007 7620 SNL CLA
 1010 5217 JMP.+7
 1011 1017 TAD ADBITI
 1012 7041 CMA IAC
 1013 1310 TAD K2270
 1014 3026 DCA ADBIT
 1015 7240 CLA CMA
 1016 5222 JMP.+4
 1017 1017 TAD ADBITI
 1020 3026 DCA ADBIT
 1021 7201 CLA IAC
 1022 3027 DCA SG
 1023 1030 TAD CRITER
 1024 1311 TAD M13
 1025 3032 DCA CTBLANC
 1026 1312 TAD M10D
 1027 3031 DCA J
 1030 7701 ACL
 1031 7112 CLL RTR
 1032 7620 SNL CLA
 1033 5241 JMP.+6
 1034 1313 TAD JMPEN2
 1035 3275 DCA EN5
 1036 1314 TAD KSWP
 1037 3255 DCA EN3
 1040 5252 JMP EN2
 1041 1315 TAD JMPEN1
 1042 3275 DCA EN5
 1043 1316 TAD JMPEN4
 1044 3255 DCA EN3

1045 7521 EN1,SWP
 1046 7104 CLL RAL
 1047 7421 MQL
 1050 7420 SNL
 1051 5261 JMP EN4

1052 1025 EN2,TAD MOT
 1053 1102 TAD K8
 1054 3025 DCA MOT

1055 0000 EN3,0 /SWP OU JMP EN4
 1056 7104 CLL RAL
 1057 7421 MQL
 1060 7430 SZL

1061 2026 EN4,ISZ ADBIT
 1062 1426 TAD I ADBIT
 1063 7450 SNA
 1064 5300 JMP EXTREME

1065	0000	EXEC,0	/AND 1 MOT	OU	JMS 1 ADRSR
1066	0000	0	/SZA CLA	OU	JMP.+6(5274)
			/MODE 1	OU	MODE 2

1067 5274 JMP.+5
 1070 2032 ISZ CTBLANC
 1071 5274 JMP.+3
 1072 2200 ISZ SUIVRE
 1073 5600 JMP I SUIVRE
 1074 2031 ISZ J

A-5/:

1075 0000 EN5,0 /JMP EN1 OU JMP EN2
 1076 5600 JMP I SUIVRE

*1100

1100 1026 EXTREME, TAD ADBIT
 1101 1317 TAD M14
 1102 3026 DCA ADBIT
 1103 1025 TAD MOT
 1104 1027 TAD SG
 1105 3025 DCA MOT
 1106 1426 TAD I ADBIT
 1107 5265 JMP EXEC

*1110

1110 2270 K2270, 2270
 1111 7765 M13, 7765
 1112 7766 M10D, 7766
 1113 5252 JMPEN2, JMP EN2
 1114 7521 KSWP, SWP
 1115 5245 JMPEN1, JMP EN1
 1116 5261 JMPEN4, JMP EN4
 1117 7764 M14, 7764

*1120

/BITS, ORDRE GAUCHE A DROITE.

1120	4000	4000	/	100	000	000	000
1121	2000	2000	/	010	000	000	000
1122	1000	1000	/	001	000	000	000
1123	0400	0400	/	000	100	000	000
1124	0200	0200	/	000	010	000	000
1125	0100	0100	/	000	001	000	000
1126	0040	0040	/	000	000	100	000
1127	0020	0020	/	000	000	010	000
1130	0010	0010	/	000	000	001	000
1131	0004	0004	/	000	000	000	100
1132	0002	0002	/	000	000	000	010
1133	0001	0001	/	000	000	000	001
1134	0000	0000	/	000	000	000	000

*1135

/BITS, ORDRE DROITE A GAUCHE

1135	0001	0001	/	000	000	000	001
1136	0002	0002	/	000	000	000	010
1137	0004	0004	/	000	000	000	100
1140	0010	0010	/	000	000	001	000
1141	0020	0020	/	000	000	010	000
1142	0040	0040	/	000	000	100	000
1143	0100	0100	/	000	001	000	000
1144	0200	0200	/	000	010	000	000
1145	0400	0400	/	000	100	000	000
1146	1000	1000	/	001	000	000	000
1147	2000	2000	/	010	000	000	000
1150	4000	4000	/	100	000	000	000
1151	0000	0000	/	000	000	000	000

*4354

4354 0000 MODE1,0
4355 1372 TAD ANDIMOT
4356 3773 DCA I ADEEXEC
4357 1375 TAD SZACLA
4360 3774 DCA I AEXECP1
4361 5754 JMP I MODE1

*4364

4364 0000 MODE2,0
4365 1376 TAD JMSIADRSR
4366 3773 DCA I ADEEXEC
4367 1377 TAD JMPP6
4370 3774 DCA I AEXECP1
4371 5764 JMP I MODE2

4372 0425 ANDIMOT,AND I MOT
4373 1065 ADEEXEC,1065
4374 1066 AEXECP1,1066
4375 7640 SZACLA,SZA CLA
4376 4444 JMSIADRSR,JMS I ADRSR
4377 5274 JMPP6,5274

0102 0010 *0102
K8,10

0145 7770 *0145
M8,7770

/S.R. AVANT ET ARRIERE

*4756
4756 0000 AVANT, 0
4757 1374 TAD KSNLCLA
4760 3772 DCA I K1007
4761 1375 TAD KTADK8
4762 3773 DCA I K1053
4763 5756 JMP I AVANT

*4764
4764 0000 ARRIERE,0
4765 1376 TAD KSZLCLA
4766 3772 DCA I K1007
4767 1377 TAD KTADM8
4770 3773 DCA I K1053
4771 5764 JMP I ARRIERE

4772 1007 K1007,1007
4773 1053 K1053,1053
4774 7620 KSNLCLA,SNL CLA
4775 1102 KTADK8,TAD K8
4776 7630 KSZLCLA,SZL CLA
4777 1145 KTADM8,TAD M8

ARRIER 4764
AVANT 4756
KSNLCL 4774
KSZLGL 4776
KTADK8 4775
KTADM8 4777
K1007 4772
K1053 4773
K8 0102
M8 0145

APPENDICE B

Programmes FORTRAN

Table des matières - (suite)

Appendice B - Programmes FORTRAN

1. Programmes de dessin.....	B-1
2. Filtre et remise dans l'image des points enlevés par le filtre.....	B-2
3. Recherche, prolongement et assemblage des traces..	B-3
4. Recherche des traces	B-4

```

COMPILER OPTIONS - NAME= MAIN,OPT=02,LINECNT=45,SIZE=0000K,
SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,NOID,NOXREF

ISN 0002      SUBROUTINE DESL
ISN 0003      IMPLICIT INTEGER (A-Z)
ISN 0004      LOGICAL*1 L,NECV,NECH,CAVE
ISN 0005      COMMON DESSIN(40),TRACEF(50,9),NTRF,BLANC,POINT,COMPTE(10),
1 L(76,96),NECV(10,41),NECH(10,41)
ISN 0006      DIMENSION A(96),CADRE(20)
ISN 0007      WRITE (3,185)
ISN 0008      185 FORMAT ('1:')
ISN 0009      DO 13 J=1,20
ISN 0010      13 CADRE(J)=(J-1)*5
ISN 0011      WRITE (3,12) (CADRE(J),J=1,20)
ISN 0012      12 FORMAT (8X,20I5)
ISN 0013      DO 1013 I=1,76
ISN 0014      DO 1012 J=1,96
ISN 0015      A(J)=BLANC
ISN 0016      IF (L(I,J)) A(J)=POINT
ISN 0018      1012 CONTINUE
ISN 0019      1013 WRITE (3,10) I,(A(J),J=1,96),I
ISN 0020      10 FORMAT (3X,I5,5X,96A1,I7)
ISN 0021      WRITE (3,12) (CADRE(J),J=1,20)
ISN 0022      WRITE (3,185)
ISN 0023      RETURN
ISN 0024      END

```

COMPILER OPTIONS - NAME= MAIN,OPT=02,LINECNT=45,SIZE=0000K,

ISN 0002 SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,NOID,NOXREF
ISN 0003 SUBROUTINE DESLP(UNP,POINTP)

ISN 0004 IMPLICIT INTEGER (A-Z)

ISN 0005 LOGICAL*1 L,NECV,NECH,CAVE

COMMON DESSIN(40),TRACEF(50,9),NTRF,BLANC,POINT,CMPTE(10),

1 L(76,96),NECV(10,41),NECH(10,41)

ISN 0006 DIMENSION A(96),CADRE(20)

ISN 0007 WRITE (3,86)

ISN 0008 86 FORMAT (13X,'0000',9X,'0001',9X,'0002',9X,'0003',9X,'0004',9X,

1,'0005',9X,'0006',9X,'0007')

ISN 0009 NO=1770

ISN 0010 DO 8 I=1,76

ISN 0011 IF ((I-1)/8*8,NE.(I-1)) GO TO 88

ISN 0013 IF ((I-1)/64*64,EQ.(I-1)) NO=NO+200

ISN 0015 WRITE (3,91)

ISN 0016 91 FORMAT (/)

ISN 0017 NO=NO+30

ISN 0018 GO TO 89

ISN 0019 88 NO=NO+10

ISN 0020 89 DO 7 J=1,96

ISN 0021 A(J)=POINTP

ISN 0022 IF (L(I,J)) A(J)=UNP

ISN 0024 7 CONTINUE

ISN 0025 8 WRITE (3,10) NO,(A(J),J=1,96)

ISN 0026 10 FORMAT (3X,15,1X,8(1 X,12A1))

C

ISN 0027 RETURN

ISN 0028 END

```

COMPILER OPTIONS - NAME= MAIN,OPT=02,LINECNT=45,SIZE=0000K,
SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,NOID,NOXREF

ISN 0002 SUBROUTINE DESTR
ISN 0003 IMPLICIT INTEGER (A-Z)
ISN 0004 LOGICAL*1 L,NECV,NECH,CAVE
ISN 0005 COMMON DESSIN(40),TRACEF(50,9),NTRF,BLANC,POINT,COMPT(10),
1 L(76,96),NECV(10,41),NECH(10,41)

ISN 0006 DIMENSION A(96),CADRE(20)
ISN 0007 WRITE (3,185)
ISN 0008 185 FORMAT ('11')

ISN 0009 WRITE (3,186)
ISN 0010 186 FORMAT (6X,'# DE TRACE',6X,'HI',10X,'VI',10X,'HF',10X,'VF',
1 11X,'D',5X,'# DE SEGMENTS',3X,'# DE PTS',/ )

ISN 0011 WRITE (3,187) (I,(TRACEF(I,J),J=1,7),I=1,NTRF)
ISN 0012 187 FORMAT(/8112)
ISN 0013 WRITE (3,188)
ISN 0014 188 FORMAT (////)
ISN 0015 DO 13 J=1,20
ISN 0016 13 CADRE(J)=(J-1)*5
ISN 0017 WRITE (3,12) (CADRE(J),J=1,20)
ISN 0018 12 FORMAT (8X,20I5)
ISN 0019 DO 50 I=1,76
ISN 0020 DO 3 J=1,96
ISN 0021 3 A(J)=BLANC
ISN 0022 IF (NTRF.EQ.0) GO TO 50
ISN 0024 DO 45 K=1,NTRF
ISN 0025 HI=TRACEF(K,1)
ISN 0026 VI=TRACEF(K,2)
ISN 0027 HF=TRACEF(K,3)
ISN 0028 VF=TRACEF(K,4)
ISN 0029 IF (VI.GT.1.OR.VF.LT.1) GO TO 45
ISN 0031 DH=HF-HI
ISN 0032 DV=VF-VI
ISN 0033 IF (DV.LT.IABS(DH)) GO TO 20
ISN 0035 V=I
ISN 0036 DELTAH=( (V-VI)*IABS(DH)+DV/2) /DV
ISN 0037 IF (DH.LT.0) DELTAH=-DELTAH
ISN 0039 H=HI+DELTAH
ISN 0040 A(H)=DESSIN(K)
ISN 0041 GO TO 45
ISN 0042 20 VADH=IABS(DH)
ISN 0043 IF (DH.LT.0) GO TO 22
ISN 0045 SG=1

```

ISN 0046	HMIN=HI
ISN 0047	HMAX=HF
ISN 0048	GO TO 23
ISN 0049	22 SG=-1
ISN 0050	HMIN=HF
ISN 0051	HMAX=HI
ISN 0052	23 DO 25 H=HMIN,HMAX
ISN 0053	V=VI+(IABS(H-HI)*OV+VADH/2)/VADH
ISN 0054	IF (V.EQ.1) A(H)=DESSIN(K)
ISN 0056	25 CONTINUE
ISN 0057	45 CONTINUE
ISN 0058	WRITE (3,10) I,(A(J),J=1,96),I
ISN 0059	10 FORMAT (3X,15,5X,96A1,17)
ISN 0060	50 CONTINUE
ISN 0061	WRITE (3,12) (CADRE(J),J=1,20)
ISN 0062	WRITE (3,185)
ISN 0063	RETURN
ISN 0064	END

COMPILER OPTIONS - NAME= MAIN,DPT=02,LINECNT=45,SIZE=0000K,
SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NREDIT,NOID,VOXREF

ISN 0002

IMPLICIT INTEGER (A-Z)

ISN 0003

LOGICAL*1 L,NECV,NECH,CAVE

ISN 0004

COMMON DESSIN(40),TRACEF(50,9),NTRF,BLANC,POINT,COMPTE(10),

1.L(76,96),NECV(10,41),NECH(10,41)

ISN 0005

DIMENSION W(15),DIF(7),RESERV(127,2)

C

C EXECUTER LE PROGRAMME POUR LES TROIS ETOILES.

C

ISN 0006 DO 7000 MACHIN=1,3

C

ISN 0007 DIF(1)=0

ISN 0008 DIF(2)=1

ISN 0009 DIF(3)=-1

ISN 0010 DIF(4)=2

ISN 0011 DIF(5)=-2

ISN 0012 DIF(6)=3

ISN 0013 DIF(7)=-3

C LECTURE ET DEFINITION DE LA MATRICE ' L ,

ISN 0014 READ(1,3) UNP,POINTP

ISN 0015 3 FORMAT(2A1)

ISN 0016 READ(1,2) BLANC,POINT

ISN 0017 2 FORMAT(2A1)

ISN 0018 DO 6 I=1,76

ISN 0019 DO 6 J=1,96

ISN 0020 6 L(I,J)=.FALSE.

ISN 0021 DO 16 I=1,76

ISN 0022 READ(1,9) U,(W(J),J=1,15)

ISN 0023 9 FORMAT(16I5)

C

ISN 0024 R=(U-146)/4

ISN 0025 DO 15 J=1,15

ISN 0026 IF (W(J).EQ.0) GO TO 16

ISN 0028 C=(W(J)-96)/4

ISN 0029 IF (C.LE.0 .OR. C.GT.96) GO TO 15

ISN 0031 14 L(R,C)=.TRUE.

ISN 0032 15 CONTINUE

ISN 0033 16 CONTINUE

C

ISN 0034 CALL DESL

ISN 0035 CALL DESLP(UNP,POINTP)

C

C FILTRES

```

ISN 0036      NOMBRE=0
ISN 0037      DO 32 V=1,76
ISN 0038      DO 32 H=1,94
ISN 0039      IF (L(V,H)) . NOMBRE=NOMBRE+1
ISN 0041      32 CONTINUE
ISN 0042      WRITE(3,34) NOMBRE
ISN 0043      34 FORMAT(/ 5X,'SANS FILTRE,NOMBRE= ',I5 )
C DEBUT DE LA PORTION DE DANIEL.
ISN 0044      NR=0
ISN 0045      NOMBRE=0
ISN 0046      DO 1009 VI=1,74
ISN 0047      DO 1009 HI=3,94
ISN 0048      IF (.NOT. L(VI,HI)) GO TO 1009
C
C COMPTÉ LE NOMBRE DE POINTS NOIRS DANS LE RECTANGLE 3X5
C
C VI<=V<=VI+2 ; HI-2<=HI+2 .
C
ISN 0050      CTR=0
ISN 0051      VI=VI
ISN 0052      V2=VI+2
ISN 0053      HI=HI-2
ISN 0054      H2=HI+2
ISN 0055      DO 23 V=V1,V2
ISN 0056      DO 23 H=H1,H2
ISN 0057      IF (L(V,H)) CTR=CTR+1
ISN 0059      23 CONTINUE
ISN 0060      IF (CTR.GE.3) GO TO 1009
ISN 0062      L(VI,HI)=.FALSE.
C
C NR:NOMBRE DE POINTS EN RESERVE
C
ISN 0063      IF (NR.GE.127) GO TO 7000
ISN 0065      NR=NR+1
ISN 0066      RESERV(NR,1)=HI
ISN 0067      RESERV(NR,2)=VI
ISN 0068      1009 CONTINUE
ISN 0069      DO 1010 V=1,76
ISN 0070      DO 1010 H=1,96
ISN 0071      IF (L(V,H)) NOMBRE=NOMBRE+1
ISN 0073      1010 CONTINUE
ISN 0074      PORTIO=1
ISN 0075      WRITE(3,1011) PORTIO,NOMBRE

```

ISN 0076 1011 FORMAT(/ 5X,'AU',I5,'IEME FILTRAGE,NOMBRE=',I5)

ISN 0077 1015 CONTINUE

ISN 0078 1016 CONTINUE

ISN 0079 CALL DESL

ISN 0080 CALL DESLP(UNP,POINTP)

C FIN DE LA PORTION DE DANIEL

C

C REMISE DANS L'IMAGE DES POINTS ENLEVES PAR LE FILTRE.

C

ISN 0081 IF (NR.EQ.0) STOP

ISN 0083 DO 2000 I=1,NR

ISN 0084 HI=RESERV(I,1)

ISN 0085 VI=RESERV(I,2)

ISN 0086 2000 L(VI,HI)=.TRUE.

C

C

ISN 0087 CALL DESL

ISN 0088 NOMBRE=0

ISN 0089 DO 2010 V=1,76

ISN 0090 DO 2010 H=1,96

ISN 0091 IF (L(V,H)) NOMBRE=NOMBRE+1

ISN 0093 2010 CONTINUE

ISN 0094 WRITE(3,34) NOMBRE

C

ISN 0095 7000 CONTINUE

ISN 0096 STOP

ISN 0097 END

COMPILER OPTIONS - NAME=		MAIN,OPT=02,LINECNT=45,SIZE=0000K,
SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,NOID,NOXREF		
ISN 0002	IMPLICIT INTEGER (A-Z)	
ISN 0003	LOGICAL*1 L,NECV,NECH,CAVE	
ISN 0004	COMMON DESSIN(40),TRACEF(50,9),NTRF,BLANC,POINT,COMPTE(10),	
	1-L(76,96),NECV(10,41),NECH(10,41)	
ISN 0005	DIMENSION W(15),DIF(7)	
C	POUR QUE LE PROGRAMME SOIT EXECUTE POUR LES TROIS ETOILES.	
ISN 0006	DO 7000 MACHIN=1,3	
C		
ISN 0007	DIF (1)=0	
ISN 0008	DIF (2)=1	
ISN 0009	DIF(3)=-1	
ISN 0010	DIF(4)=2	
ISN 0011	DIF(5)=-2	
ISN 0012	DIF(6)=3	
ISN 0013	DIF(7)=-3	
ISN 0014	READ (1,2) BLANC,POINT	
ISN 0015	2 FORMAT (2A1)	
ISN 0016	DO 3 I=1,10	
ISN 0017	3 COMPTE(I)=0	
C		
C	LECTURE DE L'IMAGE	
C		
ISN 0018	DO 6 I=1,76	
ISN 0019	DO 6 J=1,96	
ISN 0020	6 L(I,J)=.FALSE.	
ISN 0021	DO 16 I=1,76	
ISN 0022	READ(1,9) U,(W(J),J=1,15)	
ISN 0023	9 FORMAT (16I5)	
ISN 0024	R=(U-146)/4	
ISN 0025	DO 15 J=1,15	
ISN 0026	IF (W(J).EQ.0) GO TO 16	
ISN 0028	C=(W(J)-96)/4	
ISN 0029	IF (R.LE.0 .OR.R.GT.76) GO TO 15	
ISN 0031	IF (C.LE.0 .OR.C.GT.96) GO TO 15	
ISN 0033	14 I(R,C)=.TRUE.	
ISN 0034	15 CONTINUE	
ISN 0035	16 CONTINUE	
ISN 0036	CALL DESI	
C		
C	LECTURE DES SYMBOLES ALPHANUMERIQUES QUI SERVIRONT DANS LE DESSIN DES	
C	TRACES.	

```

C
ISN 0037      READ(1,701) (DESSIN(K),K=1,39)
ISN 0038      701  FORMAT(39A1)
C
ISN 0039      C  DEFINITION DES MATRICES,NECV ET NECH
ISN 0040      DO 22 J=1,10
ISN 0041      DO 19 D=1,10
ISN 0042      Q1=((D-1)*J +5)/10
ISN 0043      Q2=((D-1)*(J-1)+5)/10
ISN 0044      NECV(J,D)=.TRUE.
ISN 0045      IF (Q1-Q2.EQ.0)      NECV(J,D)=.FALSE.
ISN 0046      19  NECH(J,D)=.TRUE.
ISN 0047      DO 20 D=1,31
ISN 0048      M=1
ISN 0049      IF (D.GT. 21) M=-1
ISN 0050      Q1=((21-D)*J+5*M)/10
ISN 0051      Q2=((21-D)*(J-1)+5*M)/10
ISN 0052      NECH(J,D)=.TRUE.
ISN 0053      IF (Q1-Q2.EQ.0) NECH(J,D)=.FALSE.
ISN 0054      20  NECV(J,D)=.TRUE.
ISN 0055      DO 21 D=32,41
ISN 0056      Q1=((41-D)*J+5)/10
ISN 0057      Q2=((41-D)*(J-1)+5)/10
ISN 0058      NECV(J,D)=.TRUE.
ISN 0059      IF (Q1-Q2.EQ.0) NECV(J,D)=.FALSE.
ISN 0060      21  NECH(J,D)=.TRUE.
ISN 0061      22  CONTINUE
C
ISN 0062      DO 102 I=1,10
ISN 0063      102 COMPTE(I)=0
C
ISN 0064      C  COMPTE LE NOMBRE DE POINTS NOIRS DANS L'IMAGE.
C
ISN 0065      NOMBRE=0
ISN 0066      DO 32 V=1,76
ISN 0067      DO 32 H=1,96
ISN 0068      IF (L(V,H)) NOMBRE=NOMBRE+1
ISN 0069      32  CONTINUE
ISN 0070      WRITE(3,34) NOMBRE
ISN 0071      34  FORMAT(/ 5X,'SANS FILTRE,NOMBRE= ',I5 )
ISN 0072      C
ISN 0073      C  C'EST ICI QU'ON METTRAIT LE FILTRE SI ON LE VOULAIT.
ISN 0074      C

```

C RECHERCHE DES TRACES : SEGMENTS INDEPENDANTS DE 10 PTS, RESTRICTION
C LATÉRALE , CHOIX DE TOUT SEGMENT AYANT CTR SUFFISANT.

ISN 0075
ISN 0076
ISN 0077
ISN 0078
ISN 0080
ISN 0081
ISN 0082

NTRF=0
DO 499 VIP=1,66
DO 499 HIP=13,84
IF (,NOT, L(VIP,HIP)) GO TO 499
VI=VIP
HI=HIP
TOLERE=2

C

C POUR EVITER D'AVOIR DES TRACES TROP RAPPROCHEES.

C

ISN 0083
ISN 0084
ISN 0086
ISN 0087
ISN 0088
ISN 0089
ISN 0090

CALL FVPROX(HI,VI,TOLERE,CAVE)
IF (CAVE) GO TO 499
CRITER=5
DI=1
DF=40
CALL CHSEG(HI,VI,HF,VF,BEST,CTRM,DI,DF,CRITER,CAVE)
IF (,NOT, CAVE) GO TO 499

ISN 0092
ISN 0093
ISN 0095
ISN 0096
ISN 0097
ISN 0098
ISN 0099
ISN 0100
ISN 0101
ISN 0102
ISN 0103

NTRF=NTRF+1
IF (NTRF,GT,50) GO TO 500
TRACEF(NTRF,1)=HI
TRACEF(NTRF,2)=VI
TRACEF(NTRF,3)=HF
TRACEF(NTRF,4)=VF
TRACEF(NTRF,5)=BEST
TRACEF(NTRF,6)=1
TRACEF(NTRF,7)=CTRM
TRACEF(NTRF,8)=NTRF
TRACEF(NTRF,9)=0

C

ISN 0104
ISN 0106
ISN 0108
ISN 0109
ISN 0111
ISN 0112
ISN 0113
ISN 0114
ISN 0115
ISN 0117
ISN 0118
ISN 0119

50 IF (VF,GT,66) GO TO 499
IF (HF,GT,84,OR,HF,LT,13) GO TO 499
MH=0
IF (11,LE,BEST,AND,BEST,LE,31) MH=1
MV=1-MH
HI=HF
VII=VF
SG=1
IF (BEST,GT,21) SG=-1
DO 399 JPI=1,6
J=JPI-1
IF (J,EQ,0) GO TO 56

ISN 0121	IF (NECH(J,BEST)) HII=HII+SG
ISN 0123	IF (NECV(J,BEST)) VII=VII+1
ISN 0125	56 DO 399 I=1,5
ISN 0126	HI=HII+DIE(I)*MH
ISN 0127	VI=VII+DIF(I)*MV
ISN 0128	IF (VI,LE,0,OR,VI,GE,66) GO TO 399
ISN 0130	IF (HI,LF,12,OR,HI,GE,85) GO TO 399
ISN 0132	ASSEZ=2
ISN 0133	IF (TRACEF(NTRF,6),GE,2) ASSEZ=1
ISN 0135	IF (.NOT.L(VI,HI)) GO TO 399
ISN 0137	IF (1,GT,3) ASSEZ=ASSEZ-1
ISN 0139	DI=TRACEF(NTRF,5)-ASSEZ
ISN 0140	DF=TRACEF(NTRF,5)+ASSEZ
C	
C	
ISN 0141	CRITER=4
ISN 0142	CALL CHSEG(HI,VI,HF,VF,BEST,CTRM,DI,DF,CRITER,CAVE)
ISN 0143	IF (.NOT.CAVE) GO TO 399
ISN 0145	TRACEF(NTRF,3)=HF
ISN 0146	TRACEF(NTRF,4)=VF
ISN 0147	TRACEF(NTRF,5)=DMOYEN(TRACEF(NTRF,1),TRACEF(NTRF,2),HF,VF)
ISN 0148	TRACEF(NTRF,6)=TRACEF(NTRF,6) +1
ISN 0149	TRACEF(NTRF,7)=TRACEF(NTRF,7)+CTRM
C	
ISN 0150	GO TO 50
ISN 0151	399 CONTINUE
ISN 0152	499 CONTINUE
ISN 0153	500 CONTINUE
ISN 0154	IF (NTRF,LE,0) GO TO 6000
C	
C	AU CAS OU UNE DES TRACES AURAIT SON POINT FINAL PLUS HAUT QUE SON POINT
C	INITIAL.
C	
ISN 0156	DO 505 I=1,NTRF
ISN 0157	IF (TRACEF(I,4),GE,TRACEF(I,2)) GO TO 505
ISN 0159	HF=TRACEF(I,1)
ISN 0160	VF=TRACEF(I,2)
ISN 0161	HI=TRACEF(I,3)
ISN 0162	VI=TRACEF(I,4)
ISN 0163	TRACEF(I,1)=HI
ISN 0164	TRACEF(I,2)=VI
ISN 0165	TRACEF(I,3)=HF
ISN 0166	TRACEF(I,4)=VF

ISN 0167 TRACEF(I,5)=DMOYEN(HI,VI,HE,VE)

ISN 0168 505 CONTINUE

C

ISN 0169 CALL DESIR

C

C

C

C C'EST ICI QU'ON REMETTRAIT DANS L'IMAGE LES POINTS ENLEVES PAR LE FILTRE
C SI ON L'AVAIT UTILISE.

C

C

C

C

C

C PROLONGEMENT DES TRACES.

C

ISN 0170 DO 590 K=1,NTRF

ISN 0171 D=TRACEF(K,5)

ISN 0172 DI=MAXO(1,D-1)

ISN 0173 DF=MINO(40,D+1)

ISN 0174 IF (D.GE.11.AND.D.LE.31) GO TO 510

ISN 0176 VII=TRACEF(K,4)-2

ISN 0177 VIF=VII+4

ISN 0178 HII=TRACEF(K,3)

ISN 0179 HIF=HII

ISN 0180 GO TO 515

510 HII=TRACEF(K,3)-2

ISN 0182 HIF=HII+4

ISN 0183 VII=TRACEF(K,4)

ISN 0184 VIF=VII

ISN 0185 515 CIRM=0

ISN 0186 DO 580 VI=VII,VIF

ISN 0187 DO 580 HI=HII,HIF

ISN 0188 DO 575 D=DI,DF

ISN 0189 V=VI

ISN 0190 H=HI

ISN 0191 SG=1

ISN 0192 IF (D.GT.21) SG=-1

ISN 0194 NOBLAN=0

ISN 0195 CTR=0

ISN 0196 550 DO 570 J=1,10

ISN 0197 COMPTE(8)=COMPTE(8)+1

ISN 0198 IF (NECH(J,D)) H=H+SG

ISN 0200	IF (NECV(J,D)) V=V+1	
ISN 0202	IF (V.EQ.77.OR.H.EQ.0.OR.H.EQ.97) GO TO 571	
ISN 0204	IF (V.LE.0) GO TO 571	
ISN 0206	IF (L(V,H)) GO TO 555	
ISN 0208	GO TO 560	
ISN 0209	555 CTR=CTR+1	
ISN 0210	NOBLAN=0	
ISN 0211	VV=V	
ISN 0212	HH=H	
ISN 0213	GO TO 570	
ISN 0214	560 NOBLAN=NOBLAN+1	
ISN 0215	IF (NOBLAN.GE.3) GO TO 571	
ISN 0217	570 CONTINUE	
ISN 0218	GO TO 550	
ISN 0219	571 IF (D.EQ.TRACEF(K,5)	.AND.(HI.EQ.(HII+HIF)/2)
	1 .AND. (VI.EQ. (VII+VIF)/2))	CTR=CTR+3
ISN 0221	IF (CTR.LE.CTRM) GO TO 575	
ISN 0223	CTRM=CTR	
ISN 0224	HF=HH	
ISN 0225	VF=VV	
ISN 0226	575 CONTINUE	
ISN 0227	580 CONTINUE	
ISN 0228	TRACEF(K,7)=TRACEF(K,7)+CTRM	
ISN 0229	IF (CTRM.LE.3) GO TO 590	
ISN 0231	IF (TRACEF(K,2).LE.VF) GO TO 585	
ISN 0233	H=HF	
ISN 0234	HF=TRACEF(K,1)	
ISN 0235	TRACEF(K,1)=H	
ISN 0236	V=VF	
ISN 0237	VF=TRACEF(K,2)	
ISN 0238	TRACEF(K,2)=V	
ISN 0239	585 TRACEF(K,3)=HF	
ISN 0240	TRACEF(K,4)=VF	
ISN 0241	TRACEF(K,5)=DMOYEN(TRACEF(K,1),TRACEF(K,2),HF,VF)	
ISN 0242	590 CONTINUE	
ISN 0243	CALL DESTR	
ISN 0244	WRITE (3,6668) (I,COMPTE(I),I=1,10)	
	C	
	C	
	C	PROLONGEMENT A REBOURS DES TRACES. 'A REBOURS' SIGNIFIE PARTANT
	C	DU POINT INITIAL DE LA TRACE ET ALLANT DANS LANS LE SENS OPPOSE A CELUI
	C	D'UNE TRACE AYANT CETTE VALEUR DE D
	C	

ISN 0245	DO 490 K=1,NIRF
ISN 0246	D=TRACEF(K,5)
ISN 0247	DI=MAX0(D-1,1)
ISN 0248	DF=MIN0(D+1,40)
ISN 0249	IF (11.LE.D.AND.D.LE.31) GO TO 410
ISN 0251	HII=TRACEF(K,1)
ISN 0252	HIF=HII
ISN 0253	VII=TRACEF(K,2)-1
ISN 0254	VIF=VII+2
ISN 0255	GO TO 415
ISN 0256	410 HII=TRACEF(K,1)-1
ISN 0257	HIF=HII+2
ISN 0258	VII=TRACEF(K,2)
ISN 0259	VIF=VII
ISN 0260	415 CTRM=0
ISN 0261	DO 480 VI=VII,VIF
ISN 0262	DO 480 HI=HII,HIF
ISN 0263	DO 475 D=DI,DF
ISN 0264	V=VI
ISN 0265	H=HI
ISN 0266	SG=1
ISN 0267	IF (D.GT.21) SG=-1
ISN 0269	NOBLAN=0
ISN 0270	CTR=0
ISN 0271	DO 470 J=1,10
ISN 0272	COMTE(10)=COMTE(10)+1
ISN 0273	IF (NECH(J,D)) H=H-SG
ISN 0275	IF (NECV(J,D)) V=V-1
ISN 0277	IF (V.LE.0 .OR. V.GT.76 .OR. H.LE.0 .OR. H.GT.96) GO TO 471
ISN 0279	IF (L(V,H)) GO TO 460
ISN 0281	NOBLAN=NOBLAN+1
ISN 0282	IF (NOBLAN.GE.3) GO TO 471
ISN 0284	GO TO 470
ISN 0285	460 CTR=CTR+1
ISN 0286	NOBLAN=0
ISN 0287	VV=V
ISN 0288	HH=H
ISN 0289	470 CONTINUE
ISN 0290	471 IF (D.EQ. TRACEF(K,5) .AND. (HI.EQ. (HII+HIF)/2))
	1 .AND. (VI.EQ. (VII+VIF)/2)) CTR=CTR+3
ISN 0292	IF (CTRM.LE. CTRM) GO TO 475
ISN 0294	CTRM=CTR
ISN 0295	HIP=HH

ISN 0296	VIP=VV
ISN 0297	475 CONTINUE
ISN 0298	480 CONTINUE
ISN 0299	TRACEF(K,7)=TRACEF(K,7)+CTRM
ISN 0300	IF (CTRM.LE.3) GO TO 490
ISN 0302	IF (VIP.LE.TRACEF(K,4)) GO TO 485
ISN 0304	TRACEF(K,1)=TRACEF(K,3)
ISN 0305	TRACEF(K,3)=HIP
ISN 0306	TRACEF(K,2)=TRACEF(K,4)
ISN 0307	TRACEF(K,4)=VIP
ISN 0308	GO TO 486
ISN 0309	485 TRACEF(K,1)=HIP
ISN 0310	TRACEF(K,2)=VIP
ISN 0311	486 TRACEF(K,5)=DMOYEN(TRACEF(K,1),TRACEF(K,2),TRACEF(K,3),TRACEF(K,4) 1)
ISN 0312	490 CONTINUE
	C
ISN 0313	DO 600 K=1,NTRF
ISN 0314	IF (TRACEF(K,7)-TRACEF(K,8).GE.5) TRACEF(K,6)=TRACEF(K,6)+1
ISN 0316	600 CONTINUE
	C
ISN 0317	CALL DESTR
	C
	C
	C
	C
	NOUVEAU SYSTEME D'ASSEMBLAGE DES TRACES.
	C
ISN 0318	DO 800 ASSEP1=1,4
ISN 0319	ASSEP=ASSEP1-1
ISN 0320	IF (ASSEP.EQ.3) ASSEP=2
	C
	AINSI L'ALGORITHME SERA EXECUTE AVEC LES VALEURS ASSEP=0,1,2,2
	C
	(I.E. DEUX FOIS AVEC ASSEP=2).
ISN 0322	NTRF=NTRF
ISN 0323	IF (NTRF.LE.1) GO TO 800
ISN 0325	DO 795 K=1,NTRF
ISN 0326	HI=TRACEF(K,1)
ISN 0327	VI=TRACEF(K,2)
ISN 0328	HF=TRACEF(K,3)
ISN 0329	VF=TRACEF(K,4)
ISN 0330	VF=TRACEF(K,4)
ISN 0331	D=TRACEF(K,5)
ISN 0332	KP1=K+1
ISN 0333	DV=VF-VI
ISN 0334	DH=HF-HI

ISN 0335	VADH=IABS(DH)
ISN 0336	SG=1
ISN 0337	IF (D.GT.21) SG=-1
ISN 0339	VFP5=VF+5
ISN 0340	VFM5=VF-5
ISN 0341	HMIN=MINO(HI,HF)
ISN 0342	HMAX=HMIN+VADH
ISN 0343	HMINM5=HMIN-5
ISN 0344	HMAXP5=HMAX+5
ISN 0345	HFM5=HF-5
ISN 0346	HFP5=HF+5
ISN 0347	DO 750 KK=1,NTRF
ISN 0348	IF (KK.EQ.K) GO TO 750
ISN 0350	HHI=TRACEF(KK,1)
ISN 0351	VVI=TRACEF(KK,2)
ISN 0352	HHF=TRACEF(KK,3)
ISN 0353	VVF=TRACEF(KK,4)
ISN 0354	DD=TRACEF(KK,5)
ISN 0355	VADIFD=IABS(D-DD)
ISN 0356	IF (IABS(VADIFD-40).LE.ASSEZ) GO TO 730
ISN 0358	IF (VADIFD.GT.ASSEZ) GO TO 750
ISN 0360	IF (11.LE.D.AND.D.LE.31) GO TO 710
ISN 0362	IF (D.GT.21) GO TO 707
ISN 0364	IF (HI.LT.HHI.AND.HHI.LE.HFP5.AND.HHF.GT.HF) GO TO 708
ISN 0366	GO TO 750
ISN 0367	707 IF (HI.GT.HHI.AND.HHI.GT.HFM5.AND.HHF.LT.HF) GO TO 708
ISN 0369	GO TO 750
ISN 0370	708 V=VI+(IABS(HHI-HI)*DV+VADH/2)/VADH
ISN 0371	IF (IABS(V-VVI).GT.2) GO TO 750
ISN 0373	HF=HHF
ISN 0374	VF=VVF
ISN 0375	GO TO 770
ISN 0376	710 IF (VI.LT.VVI.AND.VVI.LE.VFP5.AND.VVF.GT.VF) GO TO 711
ISN 0378	GO TO 750
ISN 0379	711 H=HI+SG*((VVI-VI)*VADH+DV/2)/DV
ISN 0380	IF (IABS(H-HHI).GT.2) GO TO 750
ISN 0382	HF=HHF
ISN 0383	VF=VVF
ISN 0384	GO TO 770
ISN 0385	730 IF (IABS(VI-VVI).GT.3) GO TO 740
ISN 0387	IF (IABS(HI-HHI).LE.5) GO TO 734
ISN 0389	IF (D.GT.21) GO TO 733

ISN 0391	IF (HHE,LE,HI,AND,HI,LE,HHI,AND,HHI,LE,HF) GO TO 734
ISN 0393	GO TO 740
ISN 0394	733 IF (HHI,LE,HI,AND,HI,LE,HHF,AND,HHI,GE,HF) GO TO 734
ISN 0396	GO TO 740
ISN 0397	734 HI=HHF
ISN 0398	VI=VVF
ISN 0399	GO TO 770
ISN 0400	740 IF (IABS(VF-VVF),GT,3) GO TO 750
ISN 0402	IF (IABS(HF-HHF),LE,5) GO TO 744
ISN 0404	IF (D,GT,21) GO TO 743
ISN 0406	IF (HHE,LE,HF,AND,HF,LE,HHI,AND,HI,LE,HF) GO TO 744
ISN 0408	GO TO 750
ISN 0409	743 IF (HHI,LE,HF,AND,HF,LE,HHF,AND,HI,GE,HHF) GO TO 744
ISN 0411	GO TO 750
ISN 0412	744 HF=HHI
ISN 0413	VE=VVI
ISN 0414	GO TO 770
ISN 0415	750 CONTINUE
ISN 0416	GO TO 795
ISN 0417	770 IF (VI,LE,VF) GO TO 771
ISN 0419	H=HI
ISN 0420	V=VI
ISN 0421	HI=HF
ISN 0422	VI=VF
ISN 0423	HE=H
ISN 0424	VF=V
ISN 0425	771 I=MINO(K,KK)
ISN 0426	J=MAXO(K,KK)
ISN 0427	TRACEF(I,1)=HI
ISN 0428	TRACEF(I,2)=VI
ISN 0429	TRACEF(I,3)=HF
ISN 0430	TRACEF(I,4)=VF
ISN 0431	TRACEF(1,5)=DMOYEN(HI,VI,HF,VF)
ISN 0432	TRACEF(1,6)=TRACEF(1,6)+TRACEF(J,6)
ISN 0433	TRACEF(1,7)=TRACEF(1,7)+TRACEF(J,7)
ISN 0434	NTRF=NTRF-1
ISN 0435	IF (KK,LT,K) KK=K
ISN 0437	IF (KK,GT,NTRF) GO TO 800
ISN 0439	DO 775 I=KK,NTRF
ISN 0440	DO 775 J=1,7
ISN 0441	775 TRACEF(I,J)=TRACEF(I+1,J)
ISN 0442	795 CONTINUE
ISN 0443	800 CONTINUE

ISN 0444 CALL DESTR

C
C

ISN 0445 WRITE (3,6668) (I,COMPTE(I),I=1,10)
ISN 0446 WRITE(3,6666)
ISN 0447 6666 FORMAT(5X,'TRAVAIL EFFECTUE ')
ISN 0448 6668 FORMAT (10(' COMPTE(' ,I2,')=' ,I7 /))
ISN 0449 6000 CONTINUE
ISN 0450 7000 CONTINUE
ISN 0451 STOP
ISN 0452 END

```
COMPILER OPTIONS - NAME= MAIN,QPT=02,LINECNT=45,SIZE=0000K,
SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,NOID,NOXREF
ISN 0002      INTEGER FUNCTION DMOYEN(HI,VI,HF,VF)
ISN 0003      IMPLICIT INTEGER (A-Z)
ISN 0004      IF (VF.LT.VI) GO TO 3
ISN 0006      DH=HF-HI
ISN 0007      VADH=IABS(DH)
ISN 0008      DV=VF-VI
ISN 0009      SG=1
ISN 0010      IF (DH.LT.0)SG=-1
ISN 0012      IF (IABS(DH).GT.DV) GO TO 1
ISN 0014      DMOYEN=21-SG*((VADH*10+DV/2)/(DV)
ISN 0015      RETURN
ISN 0016      1 IF (DH.LT.0) GO TO 2
ISN 0018      DMOYEN=(DV*10+DH/2)/DH+1
ISN 0019      RETURN
ISN 0020      2 DMOYEN=41-(DV*10+VADH/2)/VADH
ISN 0021      RETURN
ISN 0022      3 DMOYEN=41
ISN 0023      RETURN
ISN 0024      END
```

COMPILER OPTIONS - NAME=

SOURCE, EBCDIC, NOLIST, NODECK, LOAD, MAP, NOEDIT, NOID, NOXREF
 SUBROUTINE EVPROX(HI, VI, TOLERE, CAVE)
 IMPLICIT INTEGER (A-Z)
 LOGICAL*1 L, NECV, NECH, CAVE
 COMMON DESSIN(40), TRACEF(50,9), NTRF, BLANC, POINT, CCMPT(10),
 1 L(76,96), NECV(10,41), NECH(10,41)

ISN 0002

ISN 0003

ISN 0004

ISN 0005

ISN 0006

ISN 0007

ISN 0008

ISN 0010

ISN 0011

ISN 0012

ISN 0013

ISN 0014

ISN 0015

ISN 0016

ISN 0017

ISN 0018

ISN 0019

ISN 0020

ISN 0022

ISN 0024

ISN 0026

ISN 0027

ISN 0029

ISN 0030

ISN 0032

ISN 0033

ISN 0035

ISN 0036

ISN 0037

ISN 0038

ISN 0039

ISN 0040

CCMPT(6)=CCMPT(6)+1

CAVE=.FALSE.

IF (NTRF.EQ.0) RETURN

DO 10 K=1, NTRF

CCMPT(7)=CCMPT(7)+1

HIP=TRACEF(K,1)

VIP=TRACEF(K,2)

HF=TRACEF(K,3)

VF=TRACEF(K,4)

DH=HF-HIP

VADH=IABS(DH)

DV=VF-VIP

SG=1

IF (DH.LT.0) SG=-1

IF (DV.GE.VADH) GO TO 5

IF ((HI.LT.HIP.AND.HI.LE.HF).OR.(HI.GT.HIP.AND.HI.GE.HF)) GO TO 10

V=VIP+(IABS(HI-HIP)*DV+VADH/2)/VADH

IF (IABS(V-VI).LE.TOLERE) GO TO 15

GO TO 10

5 IF (VI.LT.VIP.OR.VI.GE.VF) GO TO 10

H=HIP+SG*((VI-VIP)*VADH+DV/2)/DV

IF (IABS(H-HI).LE.TOLERE) GO TO 15

10 CONTINUE

RETURN

15 CAVE=.TRUE.

CCMPT(9)=CCMPT(9)+1

RETURN

END

COMPILER OPTIONS - NAME= MAIN,OPT=02,LINECNT=45,SIZE=0000K,

SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NODEDIT,NOID,NOXREF

ISN 0002

SUBROUTINE CHSEG(HI,VI,HF,VF,BEST,CTRM,DI,DF,CRITER,CAVE)

ISN 0003

IMPLICIT INTEGER (A-Z)

ISN 0004

LOGICAL*1 L,NECV,NECH,CAVE

ISN 0005

COMMON DESSIN(40),TRACEF(50,9),NTRF,BLANC,POINT,COMPTE(10),

1 L(76,96),NECV(10,41),NECH(10,41)

ISN 0006

COMPTE(2)=COMPTE(2)+1

C

ISN 0007

CTRM=0

ISN 0008

CAVE=.FALSE.

ISN 0009

IF (VI.LE.0 .OR. VI.GT.66 .OR. HI.LE.12 .OR. HI.GE.85) RETURN

ISN 0011

IF (DI.LT.1) DI=1

ISN 0013

IF (DF.GT.41) DF=41

C

ISN 0015

IF (DF-DI.LT.10) GO TO 38

C

C

C COMPTE LE NOMBRE DE POINTS NOIRS DANS LE CARRE A DROITE DE (VI,HI):

C

C VI<V<=VI+10 ; HI<H<=HI+10

C

C S'IL N'Y EN N'A PAS AU MOINS CRITER,IL EST IMPOSSIBLE QU'IL Y AIT UN

C

C SEGMENT AVEC D=1 A 21.

C

ISN 0017

CTR=0

ISN 0018

VIP10=VI+10

ISN 0019

HIP10=HI+10

ISN 0020

HIM10=HI-10

ISN 0021

IF (DI.GE.22) GO TO 37

ISN 0023

DO 35 V=VI,VIP10

ISN 0024

DO 35 H=HI,HIP10

ISN 0025

COMPTE(4)=COMPTE(4)+1

ISN 0026

IF (L(V,H)) CTR=CTR+1

ISN 0028

35 CONTINUE

ISN 0029

IF (CTR.LT.CRITER) DI=22

C

C

C MEME CHOSE POUR LE CARRE A GAUCHE:VI<V<=VI+10 ;HI-10<H<=HI.

C

C S'IL N'Y A PAS AU MOINS CRITER POINTS NOIRS,IL EST IMPOSSIBLE QU'IL Y AIT

C

C UN SEGMENT AVEC D DE 21 A 41.

C

ISN 0031

37 CTR=0

ISN 0032

IF (DF.LE.20) GO TO 38

ISN 0034

DO 36 V=VI,VIP10

ISN 0035	DO 36 H=HIM10,HI	
ISN 0036	COMPT(4)=COMPT(4)+1	
ISN 0037	IF (L(V,H)) CTR=CTR+1	
ISN 0039	36 CONTINUE	
ISN 0040	IF (CTR.LT.CRITER) DF=20	
ISN 0042	C 38 IF (DF.LT.DI) RETURN	
	C	
	C S'IL N'Y AVAIT PAS ASSEZ DE POINTS NOIRS DANS AUCUN DES DEUX CARRES,	
	C ON A,ICI,DI=22 ET DF=20.AUCUN SEGMENT NE POUVANT S'Y TROUVER,ON QUITTE LA	
	C SOUS-ROUTINE.	
	C	
ISN 0044	COMPT(3)=COMPT(3)+1	
ISN 0045	DO 99 D=D1,DF	
ISN 0046	20 CTR=0	
ISN 0047	H=HI	
ISN 0048	V=VI	
ISN 0049	SG=1	
ISN 0050	IF (D.GT.21) SG=-1	
ISN 0052	NOBLAN=0	
ISN 0053	DO 30 J=1,10	
ISN 0054	COMPT(5)=COMPT(5)+1	
ISN 0055	IF (NECH(J,D)) H=H+SG	
ISN 0057	IF (NECV(J,D)) V=V+1	
ISN 0059	IF (H.LF.0.OR.H.GT.96.OR.V.LE.0.OR.V.GT.76) GO TO 1100	
ISN 0061	IF (L(V,H)) CTR=CTR+1	
ISN 0063	IF (.NOT.L(V,H)) NOBLAN=NOBLAN+1	
ISN 0065	IF (NOBLAN.GT.10-CRITER) GO TO 31	
ISN 0067	30 CONTINUE	
ISN 0068	31 CONTINUE	
ISN 0069	C IF (CTR.LT.CRITER) GO TO 99	
ISN 0071	IF (CTR.LE.CIRM) GO TO 99	
ISN 0073	CTRM=CTR	
ISN 0074	BEST=D	
ISN 0075	CAVE=,TRUE.	

ISN 0076	99 CONTINUE
ISN 0077	IF (NOT.CAVE) GO TO 120
C	
C	RETROUVER LE DERNIER POINT NOIR DANS LE MEILLEUR SEGMENT.
C	
ISN 0079	56 SG=1
ISN 0080	IF (BEST.GI. 21) SG=-1
C	
ISN 0082	H=HI
ISN 0083	V=VI
ISN 0084	DO 57 J=1,10
ISN 0085	IF (NECH(J,BEST)) H=H+SG
ISN 0087	IF (NECV(J,BEST)) V=V+1
C	
ISN 0089	IF (NOT.L(V,H)) GO TO 57
ISN 0091	VF=V
ISN 0092	HF=H
ISN 0093	57 CONTINUE
C	
C	EFFACE LE SEGMENT,MAIS LAISSE LE DERNIER POINT NOIR.
C	
ISN 0094	V=VI
ISN 0095	H=HI
C	
ISN 0096	DO 58 J=1,10
ISN 0097	IF (NECV(J,BEST)) V=V+1
ISN 0099	IF (NECH(J,BEST)) H=H+SG
ISN 0101	58 L(V,H)=.FALSE.
ISN 0102	L(VF,HF)=.TRUE.
C	
ISN 0103	GO TO 120
C	
C	POUR AVERTISSEMENT SEULEMENT.
ISN 0104	1100 WRITE (3,1101) HI,VI,D,J,CRITER,CTR,NJAF,H,V
ISN 0105	1101 FORMAT (5X,'AHA',9I10 //)
ISN 0106	120 RETURN
ISN 0107	END

COMPILER OPTIONS - NAME= MAIN,OPT=02,LJNECNT=45,SIZE=3000K,
 SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,NOID,NOXREF
 C RECHERCHE DES ETOILES
 IMPLICIT INTEGER (A-Z)

LOGICAL CAVE

COMMON TRACEF(20,7),BLANC,DESSIN(40),NTRF

READ (1,810) BLANC,(DESSIN(K),K=1,40)

810 FORMAT (41A1)

800 READ (1,801,END=899) TETA0G,NTRF

801 FORMAT (2I5)

WRITE (3,185)

185 FORMAT ('1')

WRITE (3,802) TETA0G,NTRF

802 FORMAT (2I10//)

READ (1,804) ((TRACEF(I,J),J=1,7),I=1,NTRF)

804 FORMAT (7I5)

CALL DESTR

C

C S'IL Y A PLUS DE 'EXCES' TRACES,ON CONSIDERE QU'IL Y A UNE ETOILE.

C

EXCES=8

IF (NTRF.GT.EXCES) GO TO 896

C

C EXAMINER TOUTES LES PAIRES DE TRACES:K=1 A NTRF,KK=1 A NTRF(KK DIFFERENT DE

C K)

C

DO 890 K=1,NTRF

HI=TRACEF(K,1)

VI=TRACEF(K,2)

HF=TRACEF(K,3)

VF=TRACEF(K,4)

D=TRACEF(K,5)

NS=TRACEF(K,6)

DV=VF-VI

DH=HF-HI

DIFMAX=MAX0(IABS(DH),DV)

C

C IL FAUT DES TRACES SUFFISAMMENT LONGUES POUR FORMER UNE ETOILE.

C

CRITR1=16

IF (DIFMAX.LT.CRITR1) GO TO 890

C

C MEC=1:REGARDE LE POINT INITIAL DE LA TRACE K.

C MEC=2:REGARDE LE POINT FINAL DE LA TRACE K.

C

ISN 0032 DO 885 MEC=1,2

ISN 0033 V=VF

ISN 0034 H=HF

ISN 0035 IF (MEC.EQ.1) V=VI

ISN 0037 IF (MEC.EQ.1) H=HI

ISN 0039 DO 879 KK=1,NTRF

ISN 0040 IF (KK.EQ.K) GO TO 879

ISN 0042 HHI=TRACEF (KK,1)

ISN 0043 VVI=TRACEF (KK,2)

ISN 0044 HHF=TRACEF (KK,3)

ISN 0045 VVF=TRACEF (KK,4)

ISN 0046 DD=TRACEF (KK,5)

ISN 0047 NNS=TRACEF (KK,6)

ISN 0048 DDV=VVF-VVI

ISN 0049 DDH=HHF-HHI

ISN 0050 VADDH=IABS(DDH)

ISN 0051 DDIFMA=MAX0(VADDH,DDV)

C DDIFMA=DDIFMAX (ON QMET LA DERNIERE LETTRE A CAUSE DE LA LIMITE DE SIX

C CARACTERES)

C

C IL FAUT DES TRACES SUFFISAMMENT LONGUES POUR FORMER UNE ETOILE.

C

ISN 0052 CRITR2=16

ISN 0053 IF (DDIFMA.LT.CRITR2) GO TO 879

C

C IL FAUT QUE LES DEUX TRACES AIENT DES ORIENTATIONS SUFFISAMMENT DIFFERENTES.

C

ISN 0055 CRITR3=4

ISN 0056 IF (IABS(D-DD).LT.CRITR3) GO TO 879

C

C MMEC=1:REGARDE LE POINT INITIAL DE LA TRACE KK.

C MMEC=2:REGARDE LE POINT FINAL DE LA TRACE KK.

C

ISN 0058 DO 875 MMEC=1,2

ISN 0059 HH=HHF

ISN 0060 VV=VVF

ISN 0061 IF (MMEC.EQ.1) HH=HHI

ISN 0063 IF (MMEC.EQ.1) VV=VVI

C

C

C LES DEUX BOUTS DE TRACE DOIVENT ETRE SUFFISAMMENT RAPPROCHES.

ISN 0065	C	MAXDIF=MAX0(IABS(VV-V), IABS(HH-H))
ISN 0066	C	CRITR4=6
ISN 0067		IF (MAXDIF, LE, CRITR4) GO TO 894
ISN 0069		875 CONTINUE
ISN 0070	C	
ISN 0071	C	CRITR5=5
ISN 0072	C	SSG=1
ISN 0074		IF (DDH, LT, 0) SSG=-1
ISN 0076		IF (VADDH, LT, DDV) GO TO 5
ISN 0078		IF ((H, LT, HH), AND, H, LT, HHF), OR, (H, GT, HH), AND, H, GT, HHF)) GJ TO 879
ISN 0079		VV=VVI+(IABS(H-HH))*DDV+VADDH/2)/VADDH
ISN 0081		IF (IABS(VV-V), LE, CRITR5) GO TO 897
ISN 0082	C	GO TO 879
ISN 0084		5 IF (V, LT, VVI), OR, V, GT, VVF) GO TO 879
ISN 0085		HH=HHI+SSG*((V-VVI)*VADDH+DDV/2)/DDV
ISN 0087		IF (IABS(H-HH), LE, CRITR5) GO TO 897
ISN 0088		879 CONTINUE
ISN 0089		885 CONTINUE
ISN 0090		890 CONTINUE
ISN 0091		WRITE (3, 808)
ISN 0092		808 FORMAT (4X, 'IL N Y A PAS D ETOILE')
ISN 0093		GO TO 800
ISN 0094		894 WRITE (3, 805) K, KK, MAXDIF
ISN 0095		805 FORMAT (4X, 'IL Y A UNE ETOILE', /3I10)
ISN 0096	C	GO TO 800
ISN 0097	C	EUREKA : IL Y A UNE ETOILE
ISN 0098		895 WRITE (3, 811) NTRF
ISN 0099		811 FORMAT (4X, 'IL Y A UNE ETOILE', /, I10)
ISN 0100		GO TO 800
ISN 0101		897 WRITE (3, 809) K, KK
ISN 0102		809 FORMAT (4X, 'IL Y A UNE ETOILE', 2I10)
ISN 0103		GO TO 800
ISN 0104		896 WRITE (3, 812) NTRF
ISN 0105		812 FORMAT (4X, 'IL Y A UNE ETOILE CAR IL Y A ', I5, ' TRACES')
ISN 0106		GO TO 800
ISN 0107		899 STOP
ISN 0108		END

Mode d'emploiDonnées

Dessin: Pour le sous-programme de dessin de l'image "DESL", il faut une carte de données avec un blanc et un astérisque (*); dans les colonnes 1 et 2.

Pour le sous-programme de dessin DESLP les caractères un et point (1 et .) doivent être tapés dans les colonnes 1 et 2 de la carte de données.

Pour le sous-programme de dessin des traces, DESTR, il faut une carte de données avec les caractères
1234567890ABCDEFGHIJKLMNPOQRSTUVWXYZ
dans les colonnes 1 à 36, respectivement.

Ces cartes de données sont lues par le programme principal. Les deux premières viennent avant les données sur l'image, la troisième, après.

Image: Il doit y avoir 76 cartes de données sur l'image, soit une par ligne.

Sur chacune, on place d'abord la coordonnée verticale de cette ligne, puis les coordonnées horizontales de chaque point noir sur cette ligne (on peut avoir de 0 à 14 de ces coordonnées). Le format de lecture est 16I5. La fin de la liste, sur chaque carte, est perçue au moyen d'un test pour 0, car les blancs sur une carte de données sont lus comme des 0.

Dans notre cas, les coordonnées verticales et horizontales allaient de 150 à 450 et de 100 à 479 respectivement, pour donner, après transformation (ISN 0024 et ISN 0028), des indices de rangée et de colonne variant entre 1 et 76, 1 et 96 respectivement. On peut facilement changer de cadre pour l'image, à condition de changer en conséquence les énoncés de transformation (ISN 0024 et ISN 0028).

Si 0 doit être parmi les valeurs possibles de la coordonnée horizontale, on ne peut plus utiliser les blancs pour indiquer la fin de la liste des coordonnées horizontales sur les cartes de données. On utilisera plutôt un nombre spécial, en dehors du domaine des valeurs possibles, et qu'on tapera sur chaque carte, après le dernier nombre. On modifiera l'énoncé de test (ISN 0026) en conséquence.

Par exemple, si la grille a, comme coordonnées verticales V: -76 à +75 (de bas en haut), et horizontales H: -96 à +95 (de gauche à droite). Prenons 217 comme valeur spéciale.

S'il y a des points noirs à ($V = -7$, $H = -20$) et ($V = -7$, $H = 35$), on tape sur la carte de données:

colonne	4,5,	8,9,10,	13,14,15	18,19,20
	- 7	- 2 0	3 5	2 1 7

Les énoncés ISN 0024, 0026 et 0026, sont changés comme suit

ISN 0024 $R = 1 + (75 - V)/2$

ISN 0026 IF (W(J).EQ.217) GO TO 16

ISN 0028 $C = 1 + (H + 96)/2$

On vérifiera qu'ainsi, R ira de 1 à 76 (de haut en bas) et C de 1 à 96 de gauche à droite.

ISN 0006 DO 7000 MACHIN = 1, n

permet de répéter n fois le programme, pour n images différentes. Pour chacune, il faut les 78 cartes de données.

Réunification: Pour regrouper en un seul programme A) le filtre, B) la recherche, le prolongement et l'assemblage des traces, et C) la recherche des étoiles, il faut réunir, dans l'ordre, (sans DESLP) les cartes

1) ISN 0002 à 0074 de B)

2) ISN 0043 à 0079 de A), en changeant

ISN 0063 IF (NR. GE. 127) GO TO 7000

à

IF (NR. GE. 127) GO TO 7005

et en ajoutant

7005 CONTINUE

après ISN 0079

3) ISN 0075 à 0169 de B)

4) ISN 0081 à 0094 de A)

5) ISN 0170 à 0444 de B)

6) ISN 0016 à 0100 de C)

7) ISN 0445 à 0452 de B)

8) Les sous-routines DESL et DEST

9) Toutes les sous-routines de B)

Ainsi, on aura un programme FORTRAN qui effectuera les sept étapes de l'organigramme 1.

Appendice C

Programmation du mini-ordinateur PDP 8/E

Table des matières - (suite)

Appendice C - Programmation du mini-ordinateur PDP 8/E

Généralités.....	C-1
1. Instructions de référence à la mémoire.....	C-2
2. Instructions d'entrée - sortie	C-5
3. Instructions d'opération.....	C-6
4. Quelques exemples de programmation.....	C-9

Programmation du mini-ordinateur PDP 8/E

Nous donnons dans cet appendice un résumé du jeu d'instructions du mini-ordinateur et expliquons quelques techniques de programmation. On trouvera tous les détails dans les manuels de la société DEC. (22,23)

Le PDP 8/E effectue ses calculs en arithmétique de complément binaire. Les instructions et les nombres (constantes et variables) résident tous en mémoire sous forme de mots de 12 bits. Les nombres doivent être à l'intérieur des limites $0 \leq N \leq 4,095 (2^{12}-1)$ ou $2048 \leq N \leq 2047 (2^{11}-1)$.

Les programmes ont été écrits en langage d'assemblée "PAL III". Dans ce langage, le programmeur donne à chaque instruction son code mnémonique. Il doit, de plus, assigner lui-même la valeur et la place en mémoire aux constantes et aux variables.* Chaque ligne de texte symbolique produit** un mot en langage machine: c'est pourquoi on dit que le langage d'assemblée est orienté vers la machine. Le code octal est employé comme abréviation du binaire, quatre chiffres exprimant 12 bits (ex. 7041 pour 111 000 100 001).

Dans les imprimés de programmes des appendices suivants on trouve le programme en langage machine, résumé en octal: la colonne de gauche est l'adresse du mot; celle du centre, son

* On assigne d'ordinaire une valeur initiale de 0 aux variables.

** Le programme qui traduit les symboles du langage d'assemblée en code machine est appelé un "assembleur".

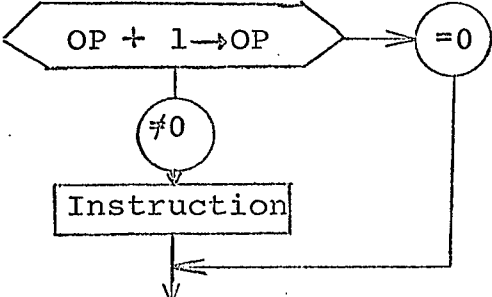
contenu; et celle de droite le texte du programme en langage d'assemblée.

Il y a trois types principaux d'instruction, et les trois premiers bits du mot permettent de les distinguer.

- 1) Référence à la mémoire: codes 0 à 5
- 2) Entrée - sortie: code 6
- 3) Instructions d'opération: code 7

1) Instructions de référence à la mémoire.

Les 3 premiers bits (premier chiffre octal) désignent l'instruction, les 9 derniers (3 derniers chiffres en octal), l'adresse de l'opérande (OP). Les temps d'exécution indiqués sont pour a) adressage direct b) adressage indirect c) adressage indexé. L'adressage est expliqué après ce tableau.

Mnémonique	Code	Opération	Temps (μsec.)
AND	0000	ET booléen bit par bit entre l'accumulateur et l'opérande. Résultat dans l'accumulateur. $OP \wedge AC \rightarrow AC$	2.6, 3.8, 4.0
TAD	1000	Additionne l'opérande à l'accumulateur. Résultat dans l'accumulateur. $OP + AC \rightarrow AC$	2.6, 3.8, 4.0
ISZ	2000	Incrémente l'opérande. Saute la prochaine instruction si l'opérande est 0. 	2.6, 3.8, 4.0

Mnémonique	Code	Opération	Temps (usec.)
DCA	3000	Dépose le contenu de l'accumulateur dans l'opérande, puis vide l'accumulateur AC → OP O → AC	2.6, 3.8, 4.0
JMS	4000	Saute à la sous-routine. Dépose l'adresse qui suit l'instruction JMS (adresse de retour) dans l'opérande, puis saute à l'instruction qui suit l'opérande. Le retour au programme principal s'effectue par un saut indirect (JMP I OP) via l'adresse de retour.	2.6, 3.8, 4.0
JMP	5000	Saute à l'opérande. Semblable au "GO TO" du FORTRAN.	1.2, 2.4, 2.6

Adressage: Pour fins d'adressage, la mémoire est divisée en 32 (40₈) pages de 128 (200₈ ou 2⁷) mots. Il y a trois modes d'adressage: direct, indirect et indexé. Pour le premier, il y a un 0 dans le bit 3 de l'instruction, pour les deux autres, un 1.

- a) Adressage direct: Le bit 4 indique si (=0) l'opérande est dans la page 0 (mots 0 à 177) ou si (=1) elle est dans la page actuelle, c.-à-d. dans la page où se trouve l'instruction. Les bits 5 à 11, additionnés à l'adresse du début de la page donnent l'adresse de l'opérande. Donc, on a accès directement aux 128 mots de la page 0 et aux 128 de la page actuelle.

- b) Adressage indirect: L'adresse que donnent les bits 4 à 11 (de la même façon que pour l'adressage direct) est celle du mot qui contient, non plus l'opérande elle même, mais plutôt l'adresse de l'opérande. Comme l'adresse est un mot de 12 bits, les 4,096 (2^{12}) mots de la mémoire sont tous accessibles de cette façon.
- c) Adressage indexé: ("Auto indexing"): Si on utilise un des huit registres "AUTO INDEX" (10 à 17) comme adresse indirecte, il est incrémenté automatiquement avant de servir d'adresse indirecte.

Exemples: Soient des instructions de saut (JMP) dans la page 7 (mots 1600 à 1777). Pour donner un exemplar concret: supposons que les autres mots contiennent des instructions d'arrêt (HLT), sauf ceux qui servent d'adresse. (De cette façon, le programme arrêtera à l'adresse indiquée après le saut).

Adresse	Code octal	Texte en langage d'assemblée	Commentaires
0007	0400	MOT7,400	
0010	0377	AUT10,377	registre "AUTO INDEX"
0160	7402	MOT160,HLT	
0400	7402	HLT	
0550	7402	MOT550,HLT	

Adresse	Code octal	Texte en langage d'assemblée	Commentaires
1000	7402	HLT	
1640	1000	K1000,1000	
1600	7402	M01660,HLT	
1700	5160	JMP MOT160	/Saut direct à la page /0. Va au mot 160.
1710	5260	JMP M01660	/Saut direct à la page /actuelle. Va au mot /1660.
1720	-	JMP MOT550	/Pas valide: on ne peut /sauter directement à une /autre page que la page 0 /ou la page actuelle.
1730	5407	JMP I MOT7	/Adressage indirect via /un mot de la page 0. /Va au mot 400.
1740	5640	JMP I K1000	/Adressage indirect via /un mot de la page actuel- /le. Va au mot 1000.
1750	5410	JMP I AUT110	/Adressage indexé. Le /mot 10 est incrémenté de /377 à 400, puis le saut /a lieu, via 10, à 400.

2) Instructions d'entrée-sortie: (Code d'instruction: 6000)

Ce sont ces instructions qui permettent au PDP 8/E de recevoir et transmettre des données à des organes périphériques. Les bits 0 à 2 doivent contenir 6, tandis que les bits 3 à 8 contiennent le code qui désigne l'unité périphérique (03 pour le clavier et la lectrice de ruban du télétype, 04 pour l'imprimante et la perforatrice du ruban) et les trois derniers bits spécifient l'opération à effectuer.

Comme le télétype est beaucoup plus lent que l'ordinateur, celui-ci est programmé pour l'attendre avant de recevoir ou transmettre chaque caractère.

Pour plus de détails, on consultera le chapitre 5 de "Introduction to Programming". (23)

L'interface est activée par l'instruction 6142, (14 est le numéro assigné à cette interface).

3) Instructions d'opération: (Code d'instruction: 7000; temps d'exécution: 1.2 μ sec.).

Groupe I: Opérations sur l'accumulateur et le bit de retenue.

Il y a 11 instructions principales dans ce groupe.

Chacune est exécutée à un des quatre temps du cycle.

Il est donc possible de les combiner pour obtenir des instructions composées qui durent quand même 1.2 usec.

Code	Mnémonique	Opération	Ordre
7000	NOP	Pas d'opération	-
7200	CLA	Vide l'accumulateur: $0 \rightarrow AC$	1
7100	CLL	Met à 0 le bit de retenue: $0 \rightarrow LK$	1
7040	CMA	Complémente l'accumulateur (complément booléen, bit par bit) $\overline{AC} \rightarrow AC$	2
7020	CML	Complémente le bit de retenue $\overline{LK} \rightarrow LK$	2
7010	RAR	Décalage (rotation) à droite de l'accumulateur et du bit de retenue. $LK \rightarrow \text{bit } 0 \text{ de } AC, \text{ bit } 0 \rightarrow \text{bit } 1, \dots, \text{ bit } 10 \rightarrow \text{bit } 11, \text{ bit } 11 \rightarrow LK$	4
7004	RAL	Décalage (rotation) à gauche de l'accumulateur et du bit de retenue. $LK \rightarrow \text{bit } 0 \text{ de } AC, \text{ bit } 0 \rightarrow \text{bit } 1, \dots, \text{ bit } 10 \rightarrow \text{bit } 11, \text{ bit } 11 \rightarrow LK$	4

Code	Mnémonique	Opération	Ordre
7012	RTR	Double décalage à droite de l'accumulateur et du bit de retenue.	4
7006	RTL	Double décalage à gauche de l'accumulateur et du bit de retenue.	4
7001	IAC	Incrémente l'accumulateur $AC + 1 \rightarrow AC$	3
7002	BSW	Echange la première et la deuxième moitiés de l'accumulateur	4

Exemples d'instructions composées

7240	CLA CMA	$0 \rightarrow AC, \overline{AC} \rightarrow AC$ i.e. $7777 \rightarrow AC$ ou $-1 \rightarrow AC$	1,2
7041	CMA IAC (ou CIA)	$\overline{AC} \rightarrow AC, AC + 1 \rightarrow AC$ i.e. $-AC \rightarrow AC$ (complément binaire)	1,3

Groupe II: Outre les instructions d'arrêt du programme (HLT, 7402) et de lecture dans l'accumulateur du registre de la console (LAS, 7604) ce groupe comprend les instructions de saut. La prochaine instruction est sautée si certaine(s) condition(s) sont satisfaites par l'accumulateur et/ou le bit de retenue.

Code	Mnémonique	Condition pour un saut
7500	SMA	$AC < 0$
7440	SZA	$AC = 0$
7510	SPA	$AC \geq 0$
7450	SNA	$AC \neq 0$
7420	SNL	$LK = 1$
7430	SZL	$LK = 0$
7410	SKP	saut inconditionnel

Code	Mnémonique	Condition pour un saut
7540	SMA SZA	$AC < 0$
7550	SPA SNA	$AC > 0$
7460	SZA SNL	$AC=0$ ou $LK=1$ <u>ou</u> les deux
7530	SPA SZL	$AC > 0$ <u>et</u> $LK = 0$
7470	SNA SZL	$AC \neq 0$ <u>et</u> $LK = 0$

On peut aussi ajouter un CLA (et additionner 0200 au code d'opération) pour vider l'accumulateur après le test. Le temps d'exécution reste 1.2 usec.

Exemple:

7740 SMA SZA CLA Saute si AC 0. Vide l'accumulateur après le test, qu'il y ait saut ou pas.

Groupe III: Opérations sur AC et MQ

Code	Mnémonique	Opération
7421	SQL	Met l'accumulateur dans le multiplicateur - quotient puis vide l'accumulateur $AC \rightarrow MQ, 0 \rightarrow AC$
7501	MQA	OU booléen entre le multiplicateur quotient et l'accumulateur. Résultat dans l'accumulateur. $AC \vee MQ \rightarrow AC$
7621	CAM	Vide l'accumulateur et le multiplicateur - quotient. $0 \rightarrow AC, 0 \rightarrow MQ$
7521	SWP	Echange l'accumulateur et le multiplicateur - quotient. $AC \leftrightarrow MQ$

Code	Mnémonique	Opération
7701	ACL	Met le multiplicateur-quotient dans l'accumulateur.

MQ $\rightarrow$ AC

7721	CLA SWP	Met le multiplicateur quotient dans l'accumulateur, puis vide le multiplicateur quotient.
------	---------	---

MQ $\rightarrow$ AC, 0 $\rightarrow$ MQ

Quelques exemples de programmation:

L'adressage indirect est indiqué par des parenthèses dans les organigrammes.

Ainsi, TAD I MEC est $AC + (MEC) \rightarrow AC$.

L'adresse de l'opérande, MEC, est entre parenthèses pour distinguer de TAD MEC: $AC + MEC \rightarrow AC$ où MEC est l'opérande elle même.

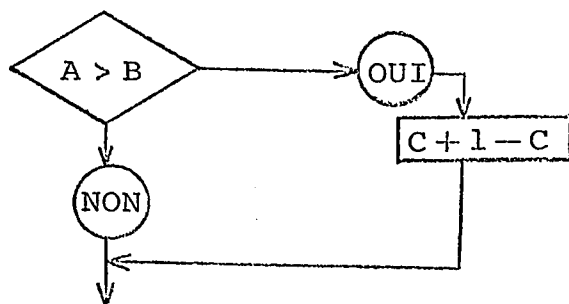
Le changement d'opération s'effectue au moyen d'une constante égale au code d'opération qu'on veut mettre. Ceci est indiqué, dans les organigrammes, par des parenthèses carrées. Par exemple: supposons que KJMPE3, E1, E2 et E3 sont tous dans la même page on veut faire

(E1) [JMP E3] $\rightarrow$ E2 pour que le mot dési-

gné par E2 contienne une instruction de saut à E3.

KJMPE3, JMP E3	/ Constante. Code d'opération de JMP E3
E1, CLA	/ Met le code d'opération pour saut à
TAD KJMPE3	/ E3, (JMP E3) dans E2.
DCA E2	
E2, 0	/ Instruction variable
E3, ...	/ Instruction à laquelle le programme
	/ transférera à partir de E2.

Enfin, donnons un exemple de test:



en FORTRAN:

IF (A.GT.B) C = C + 1

En langage d'assemblée, ce simple énoncé FORTRAN nécessite 6 instructions. (A, B, C tous dans la page actuelle ou dans la page 0).

CLA / Vide AC. 0→AC

TAD A / Additionne A à AC: AC+A→AC ; AC = A

CIA / Complément binaire de AC: -AC→AC ; AC = -A

TAD B / Additionne B à AC: AC+B→AC ; AC = B-A

SPA CLA / Saute la prochaine instruction si AC ≥ 0
 / i.e. B-A ≥ 0 ou B ≥ A.

ISZ C / Incrémente C: C+1→C.
 / Cette instruction est exécutée s'il n'y a pas eu
 / de saut. i.e. si A > B.

Appendice D

Sous-routines utilitaires

- 1) MINIMAX
- 2) MULTS
- 3) DIVIS

Table des matières - (suite)

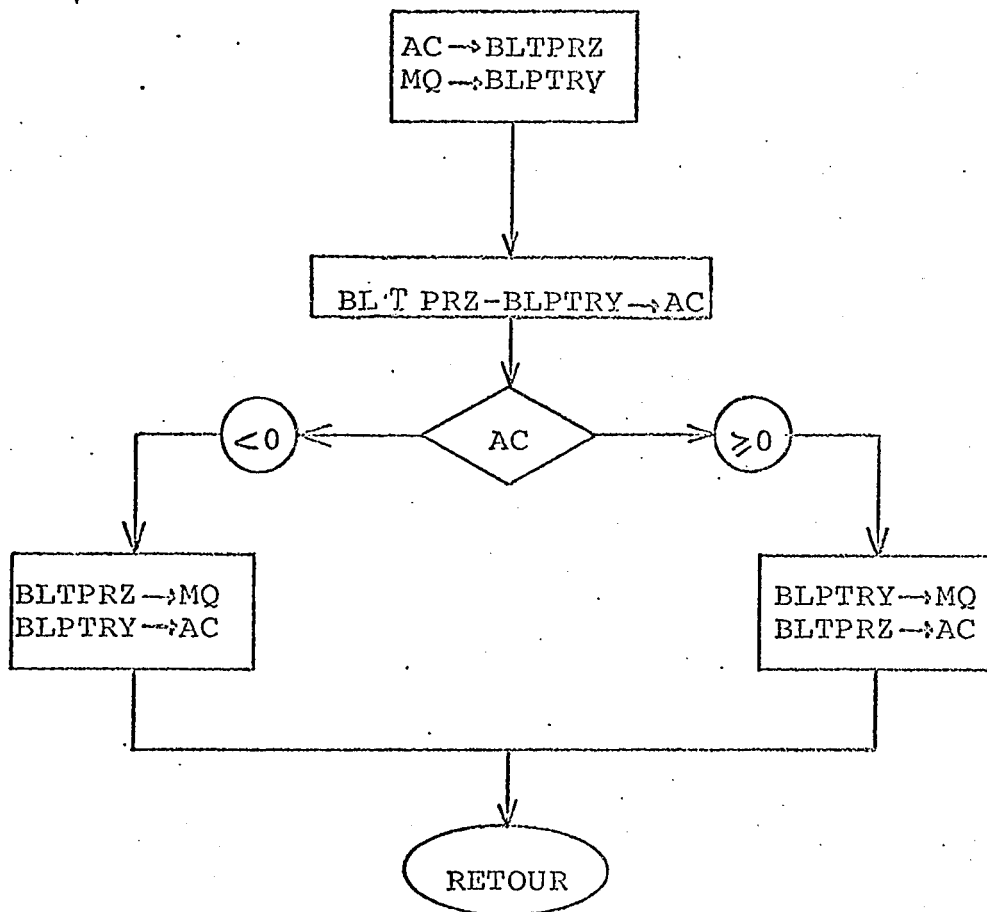
Appendice D - Sous-programmes utilitaires

1. Sous-routine MINIMAX (minimum et maximum) D-1
2. Sous-routine MULTS (multiplication) D-2
3. Sous-routine DIVIS (division) D-3

Sous-routine MINIMAX (minimum et maximum)

Avant: Deux nombre, un dans AC, l'autre dans MQ.

Après: Le plus grand des deux dans AC, le plus petit dans MQ.



/S.R. MINIMAX

/DONNEES:2 NOMBRES: UN DANS AC,L'AUTRE DANS MQ
/RESULTATS:LE PLUS GRAND DES DEUX DANS AC,
/ L'AUTRE DANS MQ

*6200

6200	0000	MINIMAX,0
6201	3224	DCA BLTPRZ
6202	7701	ACL
6203	3223	DCA BLPTRY
6204	1223	TAD BLPTRY
6205	7041	CIA
6206	1224	TAD BLTPRZ
6207	7510	SPA
6210	5216	JMP.+6
6211	7200	CLA
6212	1223	TAD BLPTRY
6213	7421	MQL
6214	1224	TAD BLTPRZ
6215	5600	JMP I MINIMAX
6216	7200	CLA
6217	1224	TAD BLTPRZ
6220	7421	MQL
6221	1223	TAD BLPTRY
6222	5600	JMP I MINIMAX

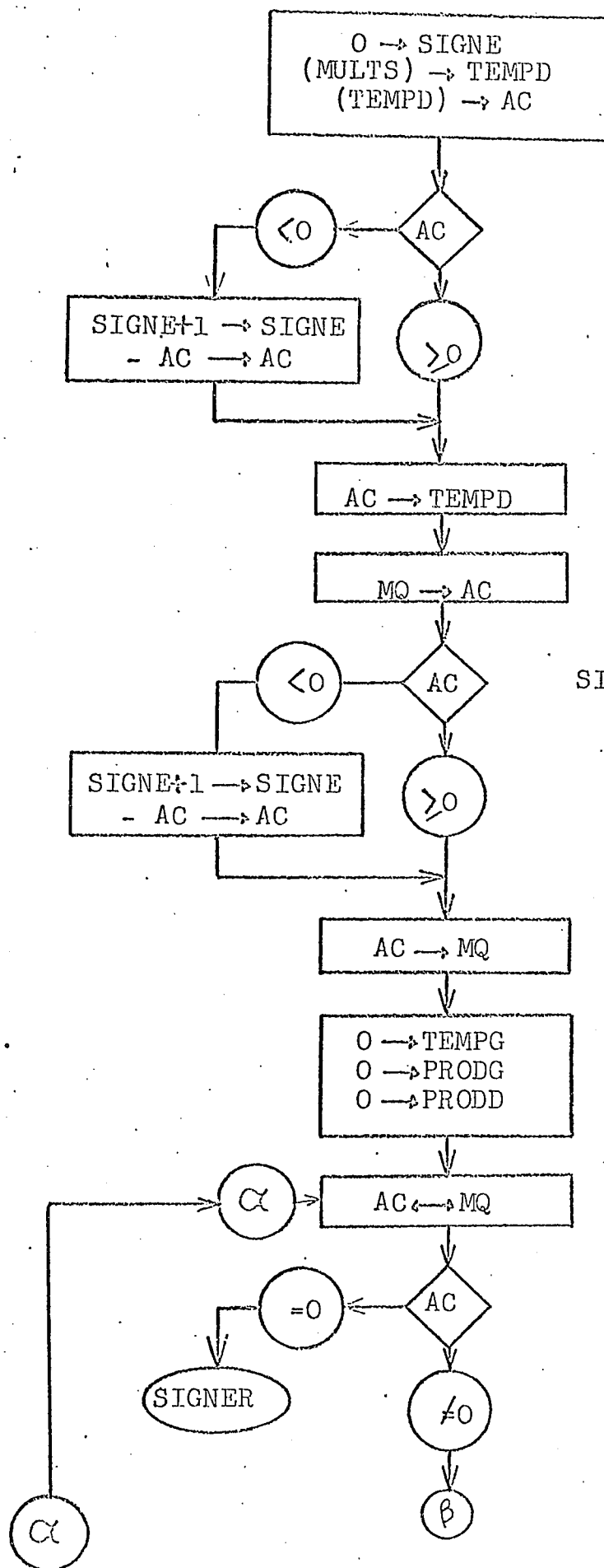
ACL=7701

MQL=7421

6223	0000	BLPTRY, 0
6224	0000	BLTPRZ, 0

ACL	7701
BLPTRY	6223
BLTPRZ	6224
MINIMA	6200
MQL	7421

Sous-routine MULTS (Multiplication de A par B)



A dans MQ. Adresse de B dans (MULTS) i.e. dans le mot qui suit l'appel à la S.R. MULTS.

|B| dans TEMPD.

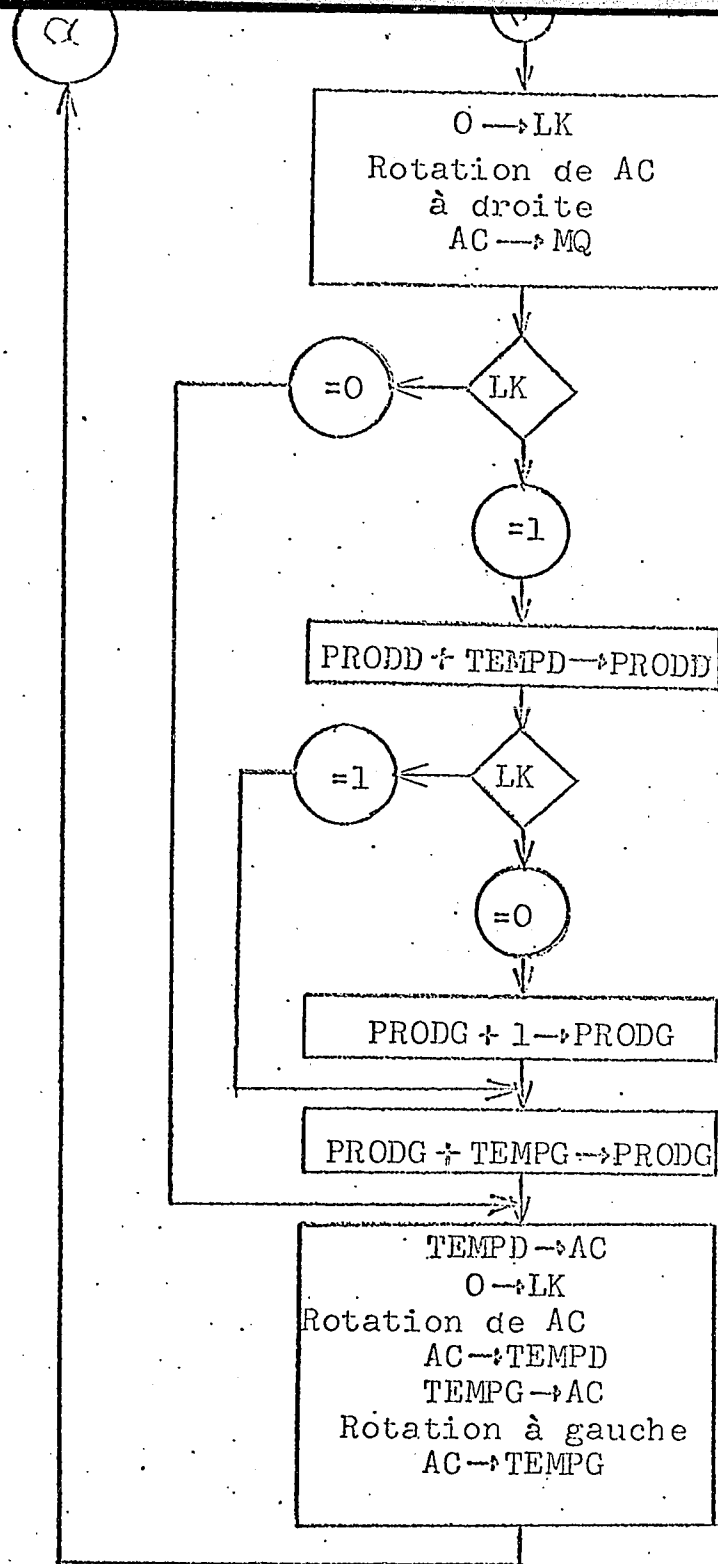
SIGNE = 0 si A et B sont positifs
 = 1 si l'un est négatif, l'autre négatif
 = 2 s'il sont tous deux négatifs

|A| dans MQ

PROD: Produit |A| * |B|
 TEMP: $B * 2^n$ $n = 0, 1, \dots, n_{max}$.
 max: ordre de B (ne peut dépasser 10 en aucun cas)

$MQ = A / 2^n$

On a fini quand les bits $2^n, 2^{n+1}, 2^{n+2} \dots$ sont tous 0.



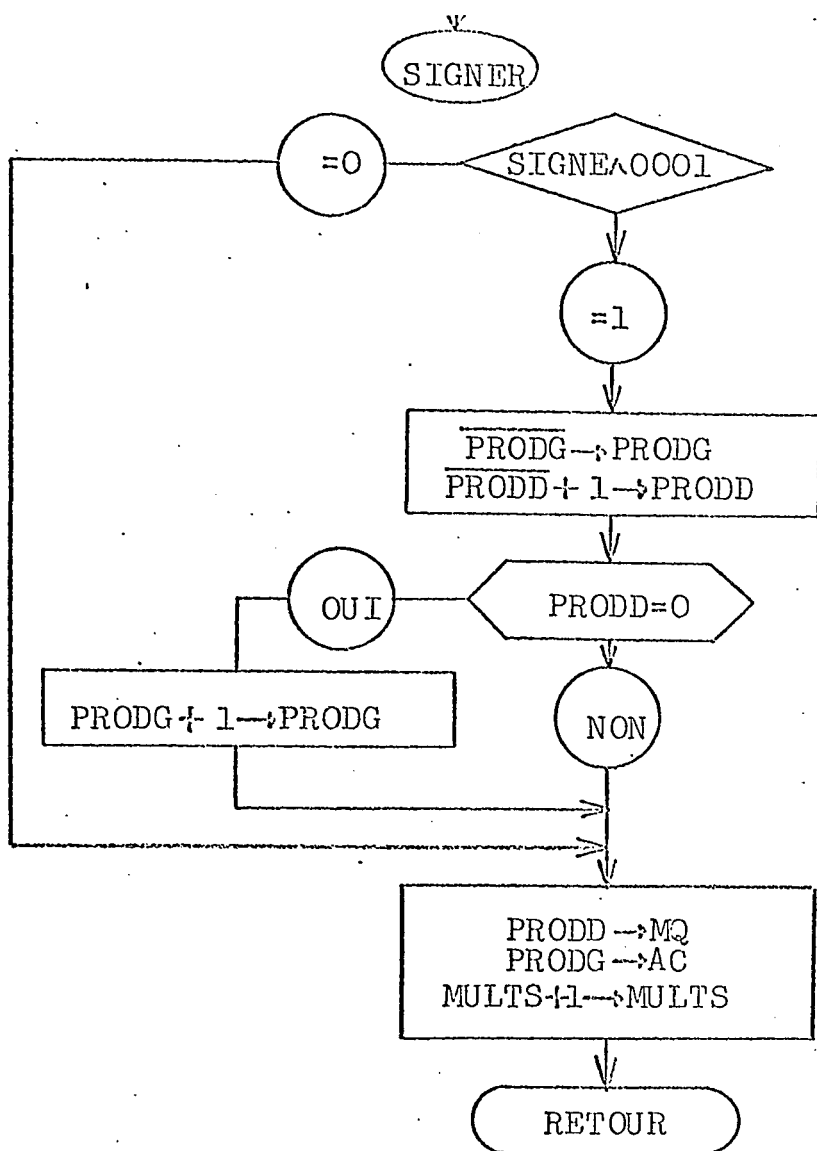
Bit 2^n de $|A|$ dans LK

$MQ/2$ MQ

Si bit 2^n de $|A| = 1$,
on additionne $B \cdot 2^n$ (TEMP)
au produit (PROD)

Le LINK est complémenté
quand il y a retenue.
On augmente PRODG de
si l'addition de droite
avait une retenue.

$TEMP \cdot 2 \rightarrow TEMP$
($n+1 \rightarrow n$)



SIGNE pair ou impair

$-\text{PROD} \rightarrow \text{PROD}$

Maintenant $\text{PROD} = \text{A} * \text{B}$

Pour retourner à la deuxième
adresse qui suit l'appel à
la S.R. MULTS.

		*4400
4400	0000	MULTS, 0
4401	3273	DCA SIGNE
4402	1600	TAD I MULTS
4403	3274	DCA TEMPD
4404	1674	TAD I TEMPD
4405	7500	SMA
4406	5211	JMP.+3
4407	2273	ISZ SIGNE
4410	7041	CMA IAC
4411	3274	DCA TEMPD
4412	7521	SWP
4413	7500	SMA
4414	5217	JMP.+3
4415	2273	ISZ SIGNE
4416	7041	CMA IAC
4417	7421	MQL
4420	3275	DCA TEMPG
4421	3277	DCA PRODG
4422	3276	DCA PRODD
4423	7521	ALPHA, SWP
4424	7450	SNA
4425	5251	JMP SIGNER
4426	7110	CLL RAR
4427	7421	MQL
4430	7420	SNL
4431	5242	JMP.+11
4432	1276	TAD PRODD
4433	1274	TAD TEMPD
4434	3276	DCA PRODD
4435	7420	SNL
4436	2277	ISZ PRODG
4437	1277	TAD PRODG
4440	1275	TAD TEMPG
4441	3277	DCA PRODG
4442	1274	TAD TEMPD
4443	7104	CLL RAL
4444	3274	DCA TEMPD
4445	1275	TAD TEMPG
4446	7004	RAL
4447	3275	DCA TEMPG
4450	5223	JMP ALPHA

4451 7201 SIGNER, CLA IAC
4452 0273 AND SIGNE
4453 7650 SNA CLA
4454 5265 JMP. +11
4455 1277 TAD PRODG
4456 7040 CMA
4457 3277 DCA PRODG
4460 1276 TAD PRODD
4461 7041 CMA IAC
4462 7450 SNA
4463 2277 ISZ PRODG
4464 3276 DCA PRODD

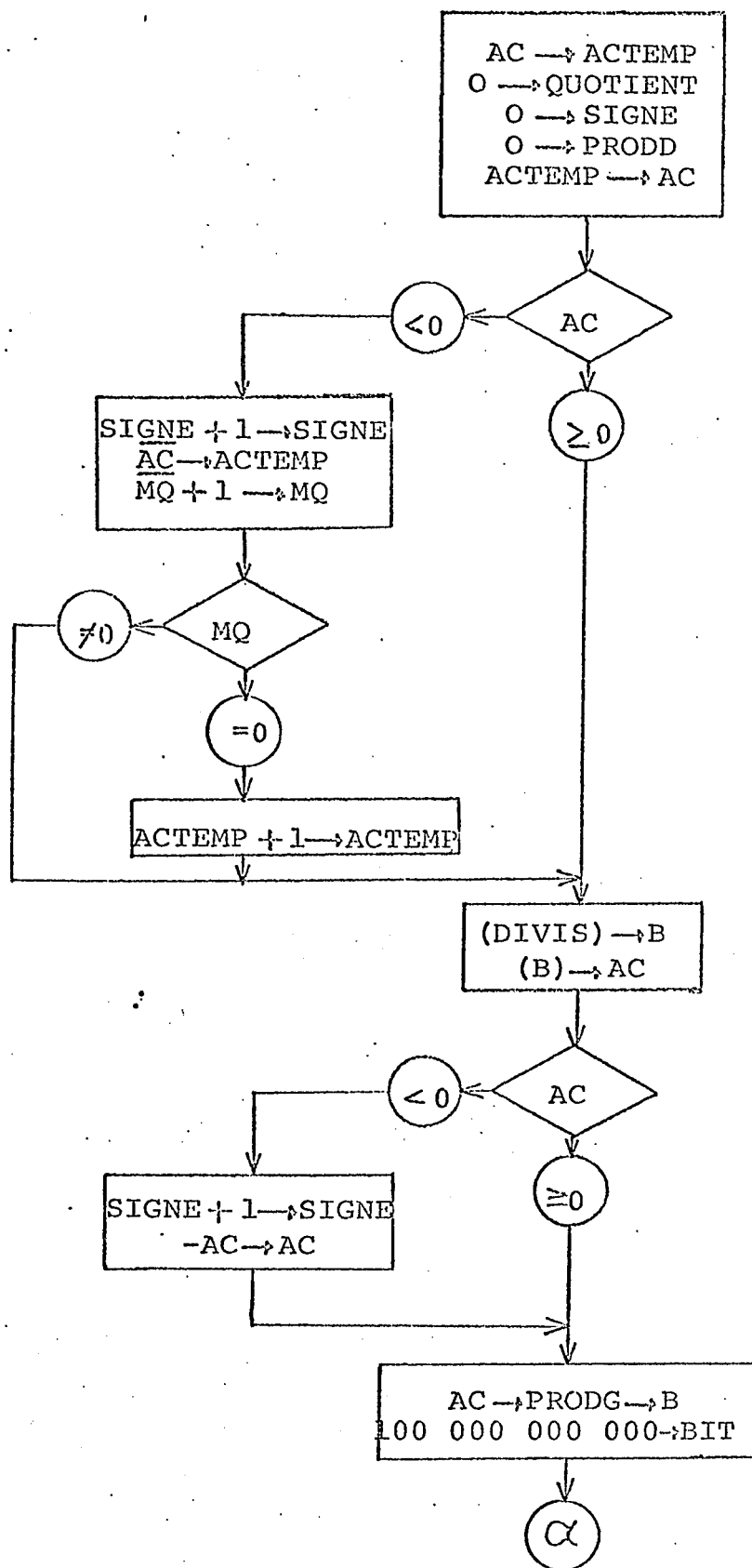
4465 1276 TAD PRODD
4466 7421 MQL
4467 1277 TAD PRODG
4470 2200 ISZ MULTS
4471 5600 JMP I MULTS

*4473

4473 0000 SIGNE, 0
4474 0000 TEMPD, 0
4475 0000 TEMPG, 0
4476 0000 PRODD, 0
4477 0000 PRODG, 0

Avant: La partie la plus significative de A est dans AC, la moins significative dans MQ. L'adresse de B suit l'appel à la sous-routine DIVIS.

Résultat: Le quotient (arrondi, pas tronqué) est dans MQ. Le reste est dans AC. Le quotient ne doit pas dépasser 2047 (3777_8) en valeur absolue.



SIGNE - Même technique
que dans la sous-routine.
MULTS

Partie la plus significative
de $|A|$ est dans ACTEMP,
la moins significative dans
MQ.

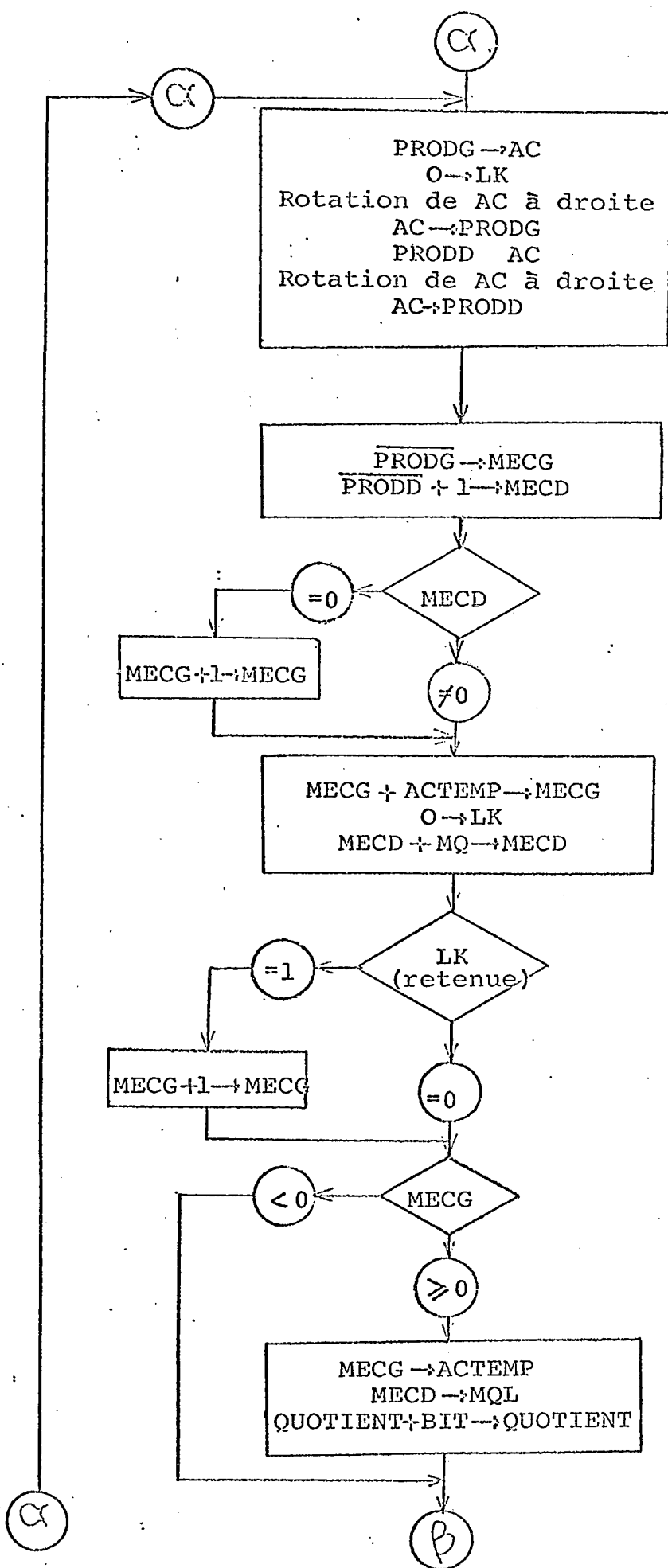
"B" = PRODD = $|B|$

BIT = 2^n
n prend les valeurs
11, 10, 9, ..., 0 dans la boucle.

PROD: $|B| * 2^n$

PRODD: Partie la plus signifi-
cative de PROD.

PRODD: Partie la moins signifi-
cative de PROD.



$$\text{PROD}/2 \rightarrow \text{PROD}$$

$$\text{MEC} = - \text{PROD}$$

MECG: Partie la plus
significative de MEC

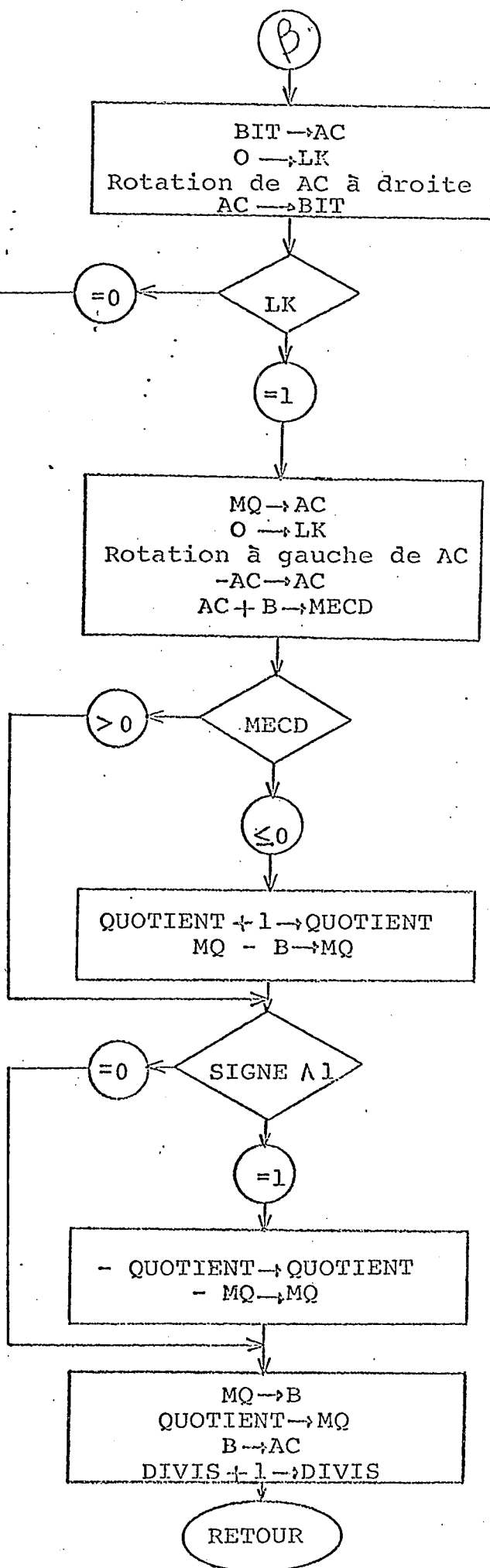
MECD: Partie la moins
significative de MEC

$$\text{MEC} = \text{RESTE} - \text{PROD}$$

MQ: Partie la plus significative
du RESTE

ACTEMP: Partie la moins
significative du
RESTE

Si $\text{RESTE} \geq \text{PROD}$
on soustrait PROD du RESTE
et ajoute BIT ($=2^n$) au
QUOTIENT.



$BIT/2 \rightarrow BIT$
 Si $LK = 1$ c'est que $n = -1$,
 i.e. que la rotation et sous
 traction répétée son
 terminées.

A ce point-ci RESTE ne peut
 être plus grand que 2047 (3777₈)

Si le RESTE $\geq \frac{1}{2}B$
 (B est ici $|B|$, la valeur
 absolue du diviseur), on
 augmente QUOTIENT
 de 1 et on soustrait B
 du RESTE

$$|MECD| = |B| - 2 * RESTE$$

SIGNE pair ou impair?

Si on a un signe -
 il faut complémentier
 le QUOTIENT et le
 RESTE

/SOUS-ROUTINE DIVIS. A/B

D-3/4

/PARTIE LA PLUS SIGNIFICATIVE DE A EST DANS AC MOINS
/SIGN DANS MQ.

/ADRESSE DE B SUIT L'APPEL A LA S.R. DIVIS.

/QUOTIENT (ARRONDI) DANS MQ, RESTE DANS AC

MQL=7421

SWP=7521

ACL=7701

*4600

4600	0000	DIVIS,0
4601	3345	DCA ACTEMP
4602	3346	DCA QUOTIENT
4603	3347	DCA SIGNE
4604	3350	DCA PRODD
4605	1345	TAD ACTEMP
4606	7500	SMA
4607	5220	JMP.+11
4610	2347	ISZ SIGNE
4611	7040	CMA
4612	3345	DCA ACTEMP
4613	7521	SWP
4614	7041	CMA IAC
4615	7450	SNA
4616	2345	ISZ ACTEMP
4617	7421	MQL
4620	7200	CLA
4621	1600	TAD I DIVIS
4622	3355	DCA B
4623	1755	TAD I B
4624	7500	SMA
4625	5230	JMP.+3
4626	7041	CMA IAC
4627	2347	ISZ SIGNE
4630	3351	DCA PRODG
4631	1351	TAD PRODG
4632	3355	DCA B
4633	7120	CLL CML
4634	7010	RAR
4635	3352	DCA BIT
4636	1351	ALPHA, TAD PRODG
4637	7110	CLL RAR
4640	3351	DCA PRODG
4641	1350	TAD PRODD
4642	7010	RAR
4643	3350	DCA PRODD
4644	1351	TAD PRODG
4645	7040	CMA
4646	3353	DCA MECG
4647	1350	TAD PRODD
4650	7041	CMA IAC
4651	7450	SNA
4652	2353	ISZ MECG
4653	3354	DCA MECD

4654	1353	TAD MECC
4655	1345	TAD ACTEMP
4656	3353	DCA MECC
4657	7100	CLL
4660	7701	ACL
4661	1354	TAD MECD
4662	3354	DCA MECD
4663	7430	SZL
4664	2353	ISZ MECC
4665	7000	NOP
4666	1353	TAD MECC
4667	7510	SPA
4670	5277	JMP.+7
4671	3345	DCA ACTEMP
4672	1354	TAD MECD
4673	7421	MQL
4674	1346	TAD QUOTIENT
4675	1352	TAD BIT
4676	3346	DCA QUOTIENT
4677	7200	CLA
4700	1352	TAD BIT
4701	7110	CLL RAR
4702	3352	DCA BIT
4703	7420	SNL
4704	5236	JMP ALPHA
4705	7701	ACL
4706	7104	CLL RAL
4707	7041	CMA IAC
4710	1355	TAD B
4711	3354	DCA MECD
4712	1354	TAD MECD
4713	7740	SMA SZA CLA
4714	5324	JMP.+10
4715	2346	ISZ QUOTIENT
4716	1355	TAD B
4717	7041	CMA IAC
4720	3354	DCA MECD
4721	7521	SWP
4722	1354	TAD MECD
4723	7421	MQL
4724	7201	CLA IAC
4725	0347	AND SIGNE
4726	7650	SNA CLA
4727	5336	JMP.+7
4730	1346	TAD QUOTIENT
4731	7041	CMA IAC
4732	3346	DCA QUOTIENT
4733	7521	SWP
4734	7041	CMA IAC
4735	7421	MQL
4736	7521	SWP
4737	3355	DCA B
4740	1346	TAD QUOTIENT
4741	7421	MQL
4742	1355	TAD B
4743	2200	ISZ DIVIS
4744	5600	JMP I DIVIS

4745	0000	ACTEMP,0
4746	0000	QUOTIENT,0
4747	0000	SIGNE,0
4750	0000	PRODD,0
4751	0000	PRODG,0
4752	0000	BIT,0
4753	0000	MECG,0
4754	0000	MECD,0
4755	0000	B,0

/S.R. DMOYEN, RESULTAT DANS AC

		*20
0020	0000	HI,0
0021	0000	VI,0
0022	0000	HF,0
0023	0000	VF,0

		*0100
0100	4400	AMULT,4400
0101	4600	ADIVIS,4600

		*700
0700	0000	DMOYEN,0
0701	7200	CLA
0702	1021	TAD VI
0703	7041	CMA IAC
0704	1023	TAD VF
0705	7500	SMA
0706	5312	JMP.+4
0707	7200	CLA
0710	1360	TAD K4351
0711	5700	JMP I DMOYEN
0712	3356	DCA DV
0713	1020	TAD HI
0714	7041	CMA IAC
0715	1022	TAD HF
0716	3357	DCA DH

0717 1357 TAD DH
0720 7500 SMA
0721 7041 CMA IAC
0722 1356 TAD DV

/AC=DV-!DH!

0723 7710 SPA CLA
0724 5337 JMP.+13
0725 1357 TAD DH
0726 7421 MQL
0727 4500 JMS I AMULT
0730 0755 K10D
0731 4501 JMS I ADIVIS
0732 0756 DV
0733 7521 SWP
0734 7041 CMA IAC
0735 1361 TAD K4325
0736 5700 JMP I DMOYEN
0737 1356 TAD DV
0740 7421 MQL
0741 4500 JMS I AMULT
0742 0755 K10D
0743 4501 JMS I ADIVIS
0744 0757 DH
0745 7521 SWP
0746 3356 DCA DV
0747 1357 TAD DH
0750 7710 SPA CLA
0751 1362 TAD K40D
0752 1363 TAD K4301
0753 1356 TAD DV
0754 5700 JMP I DMOYEN

0755 0012 K10D,12
0756 0000 DV,0
0757 0000 DH,0
0760 4351 K4351,4351
0761 4325 K4325,4325
0762 0050 K40D,50
0763 4301 K4301,4301

Appendice E

Transmission de l'image

Table des matières - (suite)

Appendice E - Transmission de l'image

1. Accès direct à la mémoire ("Data Break")	E-1
2. Compteur de ligne	E-1
3. Lignes transmises	E-2
4. Action de l'interface	E-3
5. Programme pour le "Data Break" (organigramme)	E-4
6. Etapes de transfert des données	E-5
7. Programme pour le "Data Break" (texte)	E-9
8. Note - Comment éviter de sauter une ligne sur deux...	E-11

Transmission de l'image.

Accès direct à la mémoire ("Data Break")

Le "Data Break" trois cycles* permet de transférer directement des données en mémoire. Il est activé par une instruction d'entrée-sortie, 6142,

Deux mots de la mémoire, WC ("Word Count") et CA ("Counter Address")** servent le premier à compter le nombre de mots transférés et l'autre d'adresse où doit être transféré le mot.

Dès que l'unité périphérique a un mot à envoyer à l'ordinateur, le "Data Break" interrompt l'unité centrale de traitement (et donc le programme en cours d'exécution) pendant 3 cycles.

1^{er} cycle: Incrémente WC. Si WC = 0, termine la transmission des données (l'indicateur WC est alors activée) et le programme reprend. si WC $\neq$ 0, va au 2^e cycle.

2^e cycle: Incrémente CA, l'adresse.

3^e cycle: Le registre tampon de l'unité périphérique est transféré à l'adresse indiquée par le registre CA.

Compteur de lignes: L'oscillateur principal du générateur de synchronisation*** a une impulsion à toutes les demi-lignes. Le compteur est affirmé lorsque 122 de ces impulsions ont été reçues. Si le demi-cadre pair est considéré, la première ligne qui vient après l'affirmation du

* Référence 22, chapitre 6.

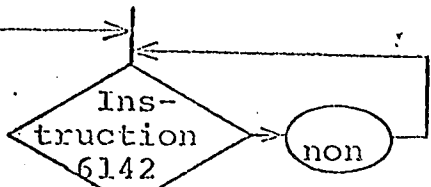
** Registres 0002 et 0003 pour ce cas-ci.

*** Référence 21, p. 11.

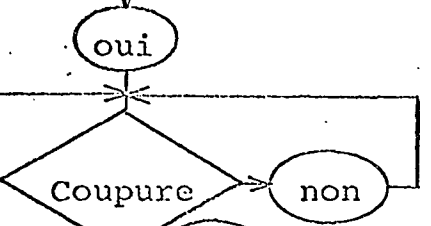
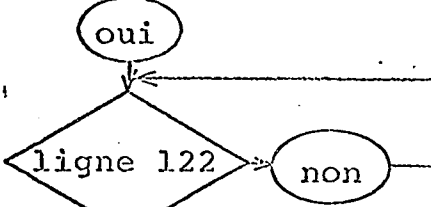
compteur est la ligne 122. Si c'est un demi-cadre impair, la ligne 123 est la première après les 122 coups de l'horloge.

Lignes transmises: Toutes les lignes du demi-cadre pair sont transmises (122, 124, 126 ..., 418, 420, 422, 424) mais, après les lignes 124, 128, ..., 416, 420, CA est diminué de 8 de sorte que la ligne suivante (126, 130, ..., 418, 422 respectivement) est mise à sa place. Ainsi, à la fin, il ne reste en mémoire que les 76 lignes 122, 126, 130, ..., 418, 422*.

*La ligne 424 est aussi transmise mais, comme c'est la 77^e ligne en mémoire, elle n'est pas utilisée par le programme d'analyse des traces. Elle ne nuit donc pas. Mais, il ne faut pas mettre de nombres ni d'instructions dans les mots 3140 à 3147 qu'elle occupe.



Le programme de l'ordinateur se déroule sans interruption.

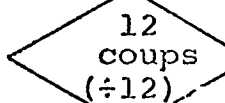


attend le début de la ligne

délai (6µs)

active l'horloge

horloge



transfert d'un mot

Registre à glissement

i)

transfert en parallèle du registre à glissement au registre tampon

Registre tampon

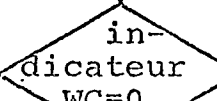
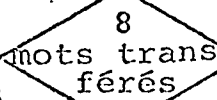
j)

Amplis de sortie

k)

vers la mémoire

3^e cycle du "Data Break" signal "B BREAK"



vide tout

Interface retourne en a)

Interface conçue et construite par Paul St-Arnaud[†] et Bill Davidson.

Programme pour le "Data Break".

PR.

a)

-2300 → WC
1777 → CA
17₈ (=000 000 001 111) → MQ

b)

6142

Active l'interface.

c)

CA=1777

oui

Le transfert n'a pas encore débuté.

d)

non

MQ ∧ WC

≠ 0

Le nombre de mots déjà transférés est un multiple de 16 (20₈) (si les 4 derniers bits de WC sont 0).

e)

BOUCLE

CA - 8 → CA

Diminue l'adresse de 8 pour que la dernière ligne transmise soit remplacée par la prochaine.

f)

-WC → COMP

g)

WC = 0

OUI

La transmission des données est terminée.

Programme

NON

h)

WC ÷ COMP

= 0

Retourne à BOUCLE si et seulement si WC a changé depuis (f).

Etapes du transfert des données.

1) Avant le transfert

Le programme commence par mettre ses compteurs aux valeurs appropriées. $5500_8 = -2300_8$ (ou -152×8) pour WC puisque 152 lignes sont transmises et 1777 pour CA puisque les données résident dans les mots 2000 et suivants. Enfin, la constante 17_8 est placée dans le registre MQ. (Elle sert en(d)). (PR.(a)).

L'instruction 6142 active l'interface (PR.(b), INT.(a)). Ensuite, il faut attendre que la ligne 122* arrive (INT.(b)). PR.(c)). Le programme attend en comparant CA à sa valeur initiale 1777: tant que CA n'a pas changé, le programme ne quitte pas cette boucle. S'il n'y avait pas la boucle (c), le CA serait diminué de 8 avant même que le transfert ne commence.

2) Le transfert des données

Quand le compteur indique 122, l'interface attend l'impulsion de coupure du générateur de synchronisation. Un monostable déclenche l'horloge 6 μ sec. après la fin de l'impulsion de coupure* (INT.(c), (d)).

* Pour fixer les idées, nous ne parlerons que du cas du demi-cadre pair. La description que nous faisons s'applique au demi-cadre impair à condition d'ajouter 1 à chaque numéro de ligne cité.

** Comme la montée de l'impulsion de coupure est plus carrée que la descente, le monostable produit un délai de 17 μ sec. après le début de l'impulsion de coupure plutôt que 6 μ sec. après la fin. (L'impulsion de coupure dure 11 μ sec.)

A chaque coup horloge, la sortie du discriminateur est transmise au registre à glissement (INT. (e), (i)). Au 12^e coup,* le contenu du registre à glissement est transféré, en parallèle, au registre tampon. Dès cet instant, l'interface réclame 3 cycles pour un transfert, et ils lui sont accordés dès la fin du cycle en cours (donc, le temps d'attente ne peut dépasser 1.4 μ sec). Puis, dans un premier cycle, WC est incrémenté, puis CA, enfin, les amplis de sortie effectuent le transfert du registre tampon à la mémoire (INT. (k)). Ces 3 cycles prennent, en tout, 4 μ sec. Comme il vient un mot (12 bits, 12 coups horloge) à toutes les 4.8 μ sec. environ, on voit qu'il y place pour aucun ou (parfois) un cycle du programme entre deux séquences de transfert. Le registre tampon garde l'information tant que le mot suivant ne lui est pas envoyé du registre à glissement, soit 4.8 μ sec.

Un deuxième compteur (compteur B) arrête l'horloge après que 8 mots ont été transférés. (INT. (g)).

Une ligne dure 63.5 μ sec., mais seulement 38 μ sec. de balayage actif sont utilisés. C'est dire qu'entre le transfert du dernier mot d'une ligne et celui du premier de la ligne suivante, il s'écoule quelques 25 μ sec. pendant lesquelles le programme de l'ordinateur n'est pas interrompu. C'est à ce moment que le programme teste si WC est un multiple de 16 (PR. (d)). Ce test

* Aucun bit n'est perdu car le transfert discriminateur - registre à glissement a lieu à la montée de chaque impulsion horloge, tandis que le transfert parallèle a lieu à la chute du 12^e coup. Un compteur $\div$ 12 donne l'impulsion au moment approprié.

prend 5 μ sec. S'il est affirmatif, CA est diminué de 8, ce qui prend 7.8 μ sec. (PR. (f)). Après quoi, le programme teste si $WC = 0$ et, sinon, va à la boucle (h) tant que WC ne change pas. La boucle (h) évite que CA ne soit diminué de 8 une seconde fois, ce qui fausserait les choses. Le minutage des instructions a été conçu pour que CA soit diminué de 8 une et une seule fois à toutes les deux lignes.

Ainsi, après le transfert de 8 mots de la première ligne, (ligne 122), $WC = 5510$ (pas divisible par 20₈) et l'information image de cette ligne réside dans les mots 2000 à 2007. Ensuite, la ligne 124 est transférée dans les mots 2010 à 2017 et, à la fin, $CA = 2017$ et $WC = 5520$, qui est un multiple de 16. Donc, CA est ramené à 2007. Puis, la ligne 126 est transférée dans les mots 2010 à 2017, effaçant ainsi la ligne 124. Comme $WC = 5530$, CA n'est pas diminué et la ligne 128 sera transmise aux mots 2020 à 2027 ... à la fin de la ligne 420, $WC = 7760$, $CA = 3137$. Comme WC est un multiple de 16, CA est diminuée de 8, à 3127. La ligne 422 est transférée dans les mots 3130 à 3137 et, après, $WC = 7770$, $CA = 3137$. Puis, la ligne 424 va dans les mots 3140 à 3147 et, enfin, $WC = 0$.

Donc, à la fin, les 76 lignes 122, 126, 130, ... 418, 422 sont dans les mots 2000 à 3137 et la ligne 424 est dans les mots 3140-3147.

3) Fin du transfert

Dès que $WC = 0$,* un indicateur est affirmé et l'interface

* Il faut que le nombre de mots transférés soit un multiple de 8 (nombre entier de lignes) car cet indicateur est testé à tous les 8 mots seulement.

est coupée, de sorte que le programme n'est plus interrompu après. (INT. (h), (l)), et ne le sera pas tant qu'une autre instruction 6142 n'aura pas été exécutée.

Quant à l'ordinateur, il va exécuter la partie suivante du programme, qui inverse l'ordre des bits dans les mots 2000 à 3137 et compte le nombre de points noirs. S'il y a plus de $528 \cdot (1000_8)$ points noirs, l'image est refusée et la prochaine image disponible est transférée. (PR. (g) et instructions 0644-0675; 0766-0773).

/PROGRAMME D'INVERSION ET DE COMPTAGE DES BITS.
 /POUR LES MOTS 2000 A 3137, INVERSE L'ODRE DES
 /BITS: ECHANGE LES BITS 0 ET 11, 1 ET 10, 2 ET 9,
 /..., 5 ET 6. EN MEME TEMPS, COMPTE LE NOM-
 /BRE DE BITS 1 (NO. DE POINTS NOIRS). SI CE
 /NOMBRE (NODEPTS) DEPASSE QUOTA, ON
 /VA A "TROP"(600), SINON, A "CAVA"(1710).
 /

MQL=7421

ACL=7701

SWP=7521

*2

0002 0000 WC,0

0003 0000 CA,0

*600

0600 7200 CLA

0601 1236 TAD M2300

0602 3002 DCA WC

0603 1237 TAD K1777

0604 3003 DCA CA

0605 1240 TAD K17

0606 7421 MQL

0607 6142 6142

/MAINTENANT LA MACHINE EST PRETE
 /A RECEVOIR L'IMAGE

0610 1241 TAD M1777

0611 1003 TAD CA

0612 7650 SNA CLA

0613 5210 JMP.-3

0614 7701 BOUCLE, ACL

0615 0002 AND WC

0616 7640 SZA CLA

0617 5214 JMP BOUCLE

0620 1242 TAD M8

0621 1003 TAD CA

0622 3003 DCA CA

0623 1002 TAD WC

0624 7041 CIA

0625 3243 DCA COMP

0626 1002 TAD WC

0627 7450 SNA

0630 5244 JMP INVCBITS

/VA INVERSER ET COMPTER LES BITS.

0631 1243 TAD COMP

0632 7640 SZA CLA

0633 5214 JMP BOUCLE

0634 1002 TAD WC

0635 5231 JMP.-4

0636 5500 M2300, 5500

0637 1777 K1777, 1777

0640 0017 K17, 17

0641 6001 M1777, 6001

0642 7770 M8, 7770

0643 0000 COMP,0

*644

0644 7300 INVCBITS,CLA CLL /PROGRAMME QUI INVERSE ET COMPTE LES
 /BITS.L'IMAGE NE DOIT EN AUCUN CAS
 /AVOIR PLUS DE 4096 POINTS NOIRS.

0645 3377 DCA NODEPTS
 0646 1237 TAD K1777
 0647 3003 DCA CA
 0650 1276 TAD M1140
 0651 3002 DCA WC
 0652 2003 ALFA, ISZ CA
 0653 1403 TAD I CA
 0654 7450 SNA
 0655 5273 JMP FALFA
 0656 7421 MQL
 0657 1277 TAD M12D
 0660 3243 DCA COMP
 0661 7521 BETA, SWP
 0662 7010 RAR
 0663 7430 SZL
 0664 2377 ISZ NODEPTS
 0665 7000 NOP
 0666 7521 SWP
 0667 7004 RAL
 0670 2243 ISZ COMP
 0671 5261 JMP BETA
 0672 3403 DCA I CA
 0673 2002 FALFA, ISZ WC
 0674 5252 JMP ALFA
 0675 5366 JMP AUDELA

0676 6640 M1140, 6640
 0677 7764 M12D, 7764

*766

0766 1377 AUDELA, TAD NODEPTS
 0767 7041 CIA
 0770 1376 TAD QUOTA
 0771 7710 SPA CLA /AC=QUOTA-NODEPTS.
 0772 5774 JMP I TROP
 0773 5775 JMP I CAVA

0774 0600 TROP, 600
 0775 1710 CAVA, 1710
 0776 1000 QUOTA, 1000
 0777 0000 NODEPTS, 0

Appendice F

Programmes en langage d'assemblage

Table des matières - (suite)

Appendice F - Programmes en langage d'assemblage

1.	Page 0.....	F-1
2.	Dessin de l'image.....	F-2
3.	Dessin des traces.....	F-3
4.	Sous-routines de conversion.....	F-4
5.	Filtrage.....	F-5
6.	Sous-routine CHSEG.....	F-6
7.	Recherche des traces.....	F-7
8.	Sous-routines DEPH et DEPV.....	F-8
9.	Sous-routine EVPROX.....	F-9
10.	Prolongement des traces.....	F-10
11.	Assemblage des traces.....	F-11
12.	Recherche des étoiles.....	F-12
13.	Mode d'emploi.....	F-13

MQL=7421

SWP=7521

ACL=7701

*2

0002	0000	WC,0
0003	0000	CA,0

*10

0010	0000	AUT110,0
0011	0000	AUT111,0
0012	0000	AUT112,0
0013	0000	AUT113,0
0014	0000	ATRF,0
0015	0000	AI15,0
0016	0000	MOT1,0
0017	0000	ADBIT1,0
0020	0000	HI,0
0021	0000	VI,0
0022	0000	HF,0
0023	0000	VF,0
0024	0000	D,0
0025	0000	MOT,0
0026	0000	ADBIT,0
0027	0000	SG,0
0030	0000	CRITER,0
0031	0000	J,0
0032	0000	CTBLANC,0
0033	0000	DI,0
0034	0000	DF,0
0035	0000	BEST,0
0036	0000	CTRM,0
0037	0000	CHSEGS,0
0040	0000	CTR,0
0041	0000	MASQUE,0
0042	0000	ADBITF,0
0043	0000	MOTF,0
0044	0000	ADRSR,0
0045	0000	NTRF,0
0046	0000	K,0
0047	0000	CNTRFMAX,0
0050	0000	ACT,0
0051	0000	YA1PTS,0
0052	0000	MOTIR,0
0053	0000	ADBITR,0
0054	0000	CTBL,0
0055	0000	ADFINI,0
0056	0000	NTRFP,0
0057	0000	KK,0
0060	0000	NS,0
0061	0000	NNS,0
0062	0000	HHI,0
0063	0000	VVI,0
0064	0000	HHF,0
0065	0000	VVF,0
0066	0000	DD,0
0067	0000	REBOURS,0

0070	0000	H,0
0071	0000	V,0
0072	0000	DV,0
0073	0000	DH,0
0074	0000	HH,0
0075	0000	VV,0
0076	0000	DDV,0
0077	0000	DDH,0

*100

0100	4400	AMULT,4400
0101	4600	ADIVIS,4600
0102	0010	K8,10
0103	0012	K10D,12
0104	0007	K7,7
0105	1000	ADSUIVRE,1000
0106	1330	ADCHSEG,1330
0107	0005	N01CRITER,5
0110	0004	N02CRITER,4
0111	1260	ADEFFH,1260
0112	1230	ADEFFV,1230
0113	2000	K2000,2000
0114	4354	AMODE1,4354
0115	4364	AMODE2,4364
0116	3730	ACHSYAL,3730
0117	3200	ADEVPROX,3200
0120	1117	K1117,1117
0121	0700	ADMOYEN,700
0122	3600	ADEPH,3600
0123	3640	ADEPV,3640
0124	1200	ADCONV,1200
0125	6200	ADMINIMAX,6200
0126	7773	M5,7773
0127	0005	K5,5
0130	3377	K3377,3377
0131	6003	AEN800,6003
0132	6133	AEN750,6133
0133	0006	K6,6
0134	7372	AEUR,7372
0135	0010	EXCES,10
0136	0020	CRITR1,20
0137	0020	CRITR2,20
0140	0004	CRITR3,4
0141	0006	CRITR4,6
0142	0005	CRITR5,5
0143	7775	M3,7775
0144	0600	ADTBK,600
0145	7770	M8,7770
0146	7000	ADCHET,7000
0147	0466	ADTAPE,466

/PEUVENT ETRE CHANGES.

/ADRESSE DE EUREKA

/CES CRITRS PEUVENT ETRE CHANGES.

5644	7200	CLA
5645	6046	TLS
5646	1354	TAD K2007
5647	3355	DCA MOTDL
5650	1145	TAD M8
5651	3357	DCA CTGRL
5652	1360	ALPHA, TAD M12D
5653	3361	DCA CTL
5654	3041	DCA MASQUE
5655	7120	CLL CML
5656	6046	BETA, TLS
5657	1041	TAD MASQUE
5660	7004	RAL
5661	3041	DCA MASQUE
5662	1363	TAD M76D
5663	3364	DCA CTCOL
5664	1355	TAD MOTDL
5665	3025	DCA MOT
5666	1366	TAD RETURN
5667	4547	JMS I ADTAPE
5670	1367	TAD LINEFEED
5671	4547	JMS I ADTAPE
5672	1373	TAD K5523
5673	3010	DCA AUTI10
5674	7200	GAMMA, CLA
5675	1425	TAD I MOT
5676	0041	AND MASQUE
5677	7650	SNA CLA
5700	5303	JMP. +3
5701	1370	TAD POINT
5702	7410	SKP
5703	1371	TAD BLANC
5704	3410	DCA I AUTI10
5705	1025	TAD MOT
5706	1372	TAD C8
5707	3025	DCA MOT
5710	2364	ISZ CTCOL
5711	5274	JMP GAMMA
5712	1363	TAD M76D
5713	3364	DCA CTCOL
5714	1373	TAD K5523
5715	3010	DCA AUTI10
5716	3375	DCA CT
5717	1410	DELTA, TAD I AUTI10
5720	1376	TAD MBLANC
5721	7650	SNA CLA
5722	5326	JMP. +4
5723	1010	TAD AUTI10
5724	1374	TAD M5523
5725	3375	DCA CT
5726	2364	ISZ CTCOL
5727	5317	JMP DELTA

5730 1375 TAD CT
5731 7040 CMA
5732 3364 DCA CTCOL
5733 1373 TAD K5523
5734 3010 DCA AUTI10

5735 2364 EPSILON, ISZ CTCOL
5736 7410 SKP
5737 5343 JMP. +4
5740 1410 TAD I AUTI10
5741 4547 JMS I ADTAPE
5742 5335 JMP EPSILON

5743 7100 CLL
5744 2361 ISZ CTL
5745 5256 JMP BETA
5746 7240 CLA CMA
5747 1355 TAD MOTDL
5750 3355 DCA MOTDL
5751 2357 ISZ CTGRL
5752 5252 JMP ALPHA
5753 7402 HLT

5754 2007 K2007, 2007
5755 0000 MOTDL, 0
5756 7770 M8, 7770
5757 0000 CTGRL, 0
5760 7764 M12D, 7764
5761 0000 CTL, 0
5762 0000 MASQUE, 0
5763 7664 M76D, 7664
5764 0000 CTCOL, 0
5765 0000 MOT, 0
5766 0215 RETURN, 215
5767 0212 LINEFEED, 212
5770 0252 POINT, 252
5771 0240 BLANC, 240
5772 0010 C8, 10
5773 5523 K5523, 5523
5774 2255 M5523, 2255
5775 0000 CT, 0
5776 7540 MBLANC, 7540

```

*200
0200 7300 CLA CLL
0201 1357 TAD K141
0202 3362 DCA H
0203 6046 TLS
0204 7240 BCLEH, CLA CMA
0205 1362 TAD H
0206 7450 SNA
0207 7402 HLT
0210 3362 DCA H
0211 1360 TAD K1777
0212 3010 DCA AUTI10
0213 1361 TAD M114
0214 3367 DCA CTV
0215 1363 TAD K240
0216 3410 DCA I AUTI10
0217 2367 ISZ CTV
0220 5215 JMP.-3
0221 7201 CLA IAC
0222 1045 TAD NTRF
0223 7041 CIA
0224 3046 DCA K
0225 1130 TAD K3377
0226 3011 DCA AUTI11
0227 2046 EN40, ISZ K
0230 5244 JMP.+14
0231 4756 JMS I ACR
0232 4775 JMS I AREFER
0233 4776 JMS I ACALCT
0234 1360 TAD K1777
0235 3010 DCA AUTI10

0236 2377 BCLEV, ISZ CT
0237 7410 SKP
0240 5204 JMP BCLEH
0241 1410 TAD I AUTI10
0242 4755 JMS I ATAPE
0243 5236 JMP BCLEV

0244 1411 TAD I AUTI11
0245 3020 DCA HI
0246 1411 TAD I AUTI11
0247 3021 DCA VI
0250 1411 TAD I AUTI11
0251 3022 DCA HF
0252 1411 TAD I AUTI11
0253 3023 DCA VF
0254 2011 ISZ AUTI11
0255 2011 ISZ AUTI11
0256 1020 TAD HI
0257 7041 CIA
0260 1022 TAD HF
0261 3370 DCA DH
0262 1370 TAD DH
0263 7510 SPA
0264 7041 CIA
0265 3374 DCA VADH
0266 1021 TAD VI
0267 7041 CIA
0270 1023 TAD VF
0271 3371 DCA DV

```

0272	1022	TAD HF
0273	7421	MQL
0274	1020	TAD HI
0275	4525	JMS I ADMINIMAX
0276	3372	DCA HMAX
0277	7701	ACL
0300	3373	DCA HMIN
0301	1362	TAD H
0302	7041	CIA
0303	1372	TAD HMAX
0304	7710	SPA CLA
0305	5227	JMP EN40
0306	1373	TAD HMIN
0307	7041	CIA
0310	1362	TAD H
0311	7710	SPA CLA
0312	5227	JMP EN40
0313	1371	TAD DV
0314	7041	CIA
0315	1374	TAD VADH
0316	7710	SPA CLA
Q317	5341	JMP FAUX
0320	1020	TAD HI
0321	7041	CIA
0322	1362	TAD H
0323	7421	MQL
0324	4500	JMS I AMULT
0325	0371	DV
0326	4501	JMS I ADIVIS
0327	0370	DH
0330	7701	ACL
0331	1021	TAD VI
0332	3364	DCA V
0333	1360	TAD K1777
0334	1364	TAD V
0335	3366	DCA ADR
0336	4765	JMS I ADRSK
0337	3766	DCA I ADR
0340	5227	JMP EN40

0341	7201	FAUX, CLA IAC
0342	1371	TAD DV
0343	7041	CIA
0344	3367	DCA CTV
0345	1021	TAD VI
0346	3364	DCA V

0347	1021	RETICI, TAD VI
0350	7041	CIA
0351	1364	TAD V
0352	7421	MQL
0353	5754	JMP I AUTPAZ

0354	0400	AUTPAZ, 400
0355	0466	ATAPE, TAPE
0356	0474	ACR, CR
0357	0141	K141, 141
0360	1777	K1777, 1777
0361	7664	M114, 7664
0362	0000	H, 0
0363	0240	K240, 240
0364	0000	V, 0
0365	0440	ADRSK, SYMBK
0366	0000	ADR, 0
0367	0000	CTV, 0
0370	0000	DH, 0
0371	0000	DV, 0
0372	0000	HMAX, 0
0373	0000	HMIN, 0
0374	0000	VADH, 0
0375	0512	AREFER, REFER
0376	0531	ACALCT, CALCT
0377	0000	CT, 0

*400

0400	4500	JMS I AMULT
0401	0370	DH
0402	4501	JMS I ADIVIS
0403	0371	DV
0404	7701	ACL
0405	1020	TAD HI
0406	3235	DCA HP
0407	1235	TAD HP
0410	7041	CIA
0411	1630	TAD I ADH
0412	7640	SZA CLA
0413	5221	JMP. +6
0414	1631	TAD I AV
0415	1232	TAD J1777
0416	3233	DCA ADRS
0417	4240	JMS SYMBK
0420	3633	DCA I ADRS
0421	2631	ISZ I AV
0422	7000	NOP
0423	2634	ISZ I ACTV
0424	5626	JMP I ADRETICI
0425	5627	JMP I ADEN40
0426	0347	ADRETICI, RETICI
0427	0227	ADEN40, EN40
0430	0362	ADH, H
0431	0364	AV, V
0432	1777	J1777, 1777
0433	0000	ADRS, 0
0434	0367	ACTV, CTV
0435	0000	HP, 0

*440

0440	0000	SYMBK,0
0441	4261	JMS PLACER
0442	1306	TAD M12
0443	7640	SZA CLA
0444	5247	JMP.+3
0445	1304	TAD K260
0446	5640	JMP I SYMBK
0447	4261	JMS PLACER
0450	1306	TAD M12
0451	7700	SMA CLA
0452	5256	JMP.+4
0453	4261	JMS PLACER
0454	1304	TAD K260
0455	5640	JMP I SYMBK
0456	4261	JMS PLACER
0457	1305	TAD K267
0460	5640	JMP I SYMBK

0461	0000	PLACER,0
0462	7201	CLA IAC
0463	1046	TAD K
0464	1045	TAD NTRF
0465	5661	JMP I PLACER

0466	0000	TAPE,0
0467	6041	TSF
0470	5267	JMP.-1
0471	6046	TLS
0472	7200	CLA
0473	5666	JMP I TAPE

0474	0000	CR,0
0475	1303	TAD K215
0476	4266	JMS TAPE
0477	1302	TAD K212
0500	4266	JMS TAPE
0501	5674	JMP I CR

0502	0212	K212,212
0503	0215	K215,215
0504	0260	K260,260
0505	0266	K267,266
0506	7766	M12,7766
0507	7640	M140,7640
0510	0340	A340,340
0511	0252	K252,252

0512	0000	REFER,0
0513	1630	TAD I ADH
0514	1307	TAD M140
0515	7640	SZA CLA
0516	5322	JMP.+4
0517	1311	TAD K252
0520	3710	DCA I A340
0521	5330	JMP.+7
0522	7240	CLA CMA
0523	1630	TAD I ADH
0524	7640	SZA CLA
0525	5330	JMP.+3
0526	1311	TAD K252
0527	3710	DCA I A340
0530	5712	JMP I REFER

/S.R.CALCT

0531	0000	CALCT,0
0532	1353	TAD M76D
0533	3354	DCA CTM
0534	1355	TAD C1777
0535	3010	DCA AUTI10
0536	7240	CLA CMA
0537	3757	DCA I ADCT

0540	1410	BCLE,TAD I AUTI10
0541	1360	TAD MBLANC
0542	7650	SNA CLA
0543	5350	JMP FBCLE
0544	1010	TAD AUTI10
0545	1356	TAD M1777
0546	7040	CMA
0547	3757	DCA I ADCT

0550	2354	FBCLE,ISZ CTM
0551	5340	JMP BCLE
0552	5731	JMP I CALCT

0553	7664	M76D,7664
0554	0000	CTM,0
0555	1777	C1777,1777
0556	6001	M1777,6001
0557	0377	ADCT,CT
0560	7540	MBLANC,7540

*3730
3730 0000 CHSYAL,0

/"CHANGE S'IL Y A LIEU"
/CHANGE ADRESSE DU BIT SI CELLE-CI EST DE 1135
/A 1150, DROITE A GAUCHE, POUR L'ADRESSE DU MEME
/BIT, MAIS DE 1120 A 1133, GAUCHE A DROITE.
/CECI EST NECESSAIRE POUR UTILISER L'ADRESSE DU
/BIT DANS LA SR CONV OU POUR LE POINT INITIAL DANS
/SUIVRE.

3731 3344 DCA ACTM

3732 1344 TAD ACTM /ADRESSE DE 1120 A 1151 DANS AC, RESULTAT 1120 A
/1133 DANS AC

3733 1226 TAD M1134

3734 7700 SMA CLA

3735 5340 JMP.+3

3736 1344 TAD ACTM

3737 5730 JMP I CHSYAL

3740 1344 TAD ACTM

3741 7041 CIA

3742 1345 TAD K2270

3743 5730 JMP I CHSYAL

3744 0000 ACTM,0

3745 2270 K2270,2270

3746 7773 M5,7773

/"S.R. CONV
/MOT DANS MQ.
/ADRESSE DU BIT DANS AC (1120 A 1133 INCL.
/ INCLUSIVEMENT
/V DANS MQ(DE 1 A 76 INCL)
/H DANS AC (DE 1 A 96 INCL)

*1200

1200 0000 CONV,0

1201 1224 TAD M1117

1202 3226 DCA ACT

1203 7701 ACL

1204 0104 AND K7

1205 7106 CLL RTL

1206 3227 DCA ACTP

1207 1227 TAD ACTP

1210 7104 CLL RAL

1211 1227 TAD ACTP

1212 1226 TAD ACT

1213 3226 DCA ACT

1214 7701 ACL

1215 1225 TAD M1770

1216 7110 CLL RAR

1217 7110 CLL-RAR

1220 7110 CLL RAR

1221 7421 MQL

1222 1226 TAD ACT

1223 5600 JMP I CONV

1224 6661 M1117,6661

1225 6010 M1770,6010

1226 0000 ACT,0

1227 0000 ACTP,0

/S.R. CONVINV

MQL=7421
 ACL=7701

/DONNEES: "H" DANS AC, "V" DANS
 /MQ
 /RESULTAT: MOT DANS AC, ADRESSE DU
 /BIT DANS MQ (DE 1120 A 1133)

*3750
 CONVINV, 0

3750	0000		
3751	3372	DCA	ACTM
3752	7701	ACL	
3753	7104	CLL	RAL
3754	7106	CLL	RTL
3755	1376	TAD	K1770
3756	3373	DCA	MOTPR
3757	7240	CLA	CMA
3760	1372	TAD	ACTM
3761	1374	TAD	M12D
3762	7510	SPA	
3763	5366	JMP.	+3
3764	2373	ISZ	MOTPR
3765	5361	JMP.	-4
3766	1375	TAD	K1134
3767	7421	MQL	
3770	1373	TAD	MOTPR
3771	5750	JMP	I CONVINV

3772	0000	ACTM,	0
3773	0000	MOTPR,	0
3774	7764	M12D,	7764
3775	1134	K1134,	1134
3776	1770	K1770,	1770

ACL	7701
ACTM	3772
CONVIN	3750
K1134	3775
K1770	3776
MOTPR	3773
MQL	7421
M12D	3774

/FILTRAGE

*5400

5400	7200	CLA	
5401	1371	TAD K177	
5402	3010	DCA AUTI10	
5403	1361	TAD M177	
5404	3032	DCA CTBLANC	/ON NE PEUT ENLEVER ET METTRE DE /COTE PLUS DE 127 (177 OCTAL) /POINTS ISOLES.
5405	1362	TAD K1777	
5406	3011	DCA AUTI11	
5407	1363	TAD M1120	
5410	3031	DCA J	
5411	1411	BCLMOT, TAD I AUTI11	
5412	7450	SNA	
5413	5306	JMP FBCLMOT	
5414	3050	DCA ACT	
5415	1120	TAD K1117	
5416	3012	DCA AUTI12	
5417	1011	TAD AUTI11	
5420	3052	DCA MOTIR	
5421	1052	BCLBIT, TAD MOTIR	
5422	3025	DCA MOT	
5423	1412	TAD I AUTI12	
5424	7450	SNA	
5425	5306	JMP FBCLMOT	
5426	0050	AND ACT	
5427	7650	SNA CLA	
5430	5223	JMP.-5	
5431	1143	TAD M3	
5432	3040	DCA CTR	
5433	1012	TAD AUTI12	
5434	1366	TAD K4500	
5435	3043	DCA MOTF	
5436	1443	TAD I MOTF	
5437	3041	DCA MASQUE	
5440	4332	JMS COMPTE	
5441	1012	TAD AUTI12	
5442	1367	TAD M1121	
5443	7740	SMA SZA CLA	
5444	5263	JMP.+17	
5445	1025	TAD MOT	
5446	0104	AND K7	
5447	7650	SNA CLA	
5450	5221	JMP BCLBIT	
5451	1364	TAD M2	
5452	1043	TAD MOTF	
5453	3041	DCA MASQUE	

5454	1441	TAD I MASQUE
5455	3041	DCA MASQUE
5456	7240	CLA CMA
5457	1052	TAD MOTIR
5460	3025	DCA MOT
5461	4332	JMS COMPTE
5462	5313	JMP ENLEVE

5463	1012	TAD AUTI12
5464	1370	TAD M1132
5465	7710	SPA CLA
5466	5313	JMP ENLEVE
5467	1025	TAD MOT
5470	7040	CMA
5471	0104	AND K7
5472	7650	SNA CLA
5473	5221	JMP BCLBIT
5474	1365	TAD K2
5475	1043	TAD MOTF
5476	3041	DCA MASQUE
5477	1441	TAD I MASQUE
5500	3041	DCA MASQUE
5501	7201	CLA IAC
5502	1052	TAD MOTIR
5503	3025	DCA MOT
5504	4332	JMS COMPTE
5505	5313	JMP ENLEVE

5506	2031	FBCLMOT, ISZ J
5507	5211	JMP BCLMOT

5510	7200	FIN, CLA
5511	3410	DCA I AUTI10
5512	7402	HLT

5513	1052	ENLEVE, TAD MOTIR
5514	3410	DCA I AUTI10
5515	1012	TAD AUTI12
5516	3410	DCA I AUTI10
5517	1012	TAD AUTI12
5520	3041	DCA MASQUE
5521	1441	TAD I MASQUE
5522	7040	CMA
5523	0050	AND ACT
5524	3050	DCA ACT
5525	1050	TAD ACT
5526	3452	DCA I MOTIR
5527	2032	ISZ CTBLANC
5530	5221	JMP BCLBIT
5531	5310	JMP FIN

/S.R. COMPTE

5532	0000	COMPTE, 0
5533	1143	TAD M3
5534	3046	DCA K

5535	1425	BCLEK, TAD I MOT
5536	0041	AND MASQUE
5537	7450	SNA
5540	5352	JMP FBCLEK

5541	7104	BCLEROT, CLL RAL
5542	7420	SNL
5543	5341	JMP. -2
5544	2040	ISZ CTR
5545	5350	JMP. +3
5546	7200	CLA
5547	5221	JMP BCLBIT
5550	7440	SZA
5551	5341	JMP BCLEROT

5552	2046	FBCLEK, ISZ K
5553	7610	SKP CLA
5554	5732	JMP I COMPTE
5555	1025	TAD MOT
5556	1102	TAD K8
5557	3025	DCA MOT
5560	5335	JMP BCLEK

5561	7601	M177, 7601	
5562	1777	K1777, 1777	
5563	6660	M1120, 6660	
5564	7776	M2, 7776	
5565	0002	K2, 2	
5566	4500	K4500, 4500	/=5620-1120
5567	6657	M1121, 6657	
5570	6646	M1132, 6646	
5571	0177	K177, 177	/ MASQUES POUR LE FILTRE

*5616

5616	0003	0003	/ 000 000 000 011
5617	0001	0001	/ 000 000 000 001

*5620

5620	7000	7000	/ 111 000 000 000
5621	7400	7400	/ 111 100 000 000
5622	7600	7600	/ 111 110 000 000
5623	3700	3700	/ 011 111 000 000
5624	1740	1740	/ 001 111 100 000
5625	0760	0760	/ 000 111 110 000
5626	0370	0370	/ 000 011 111 000
5627	0174	0174	/ 000 001 111 100
5630	0076	0076	/ 000 000 111 110
5631	0037	0037	/ 000 000 011 111
5632	0017	0017	/ 000 000 001 111
5633	0007	0007	/ 000 000 000 111

*5634

5634	4000	4000	/ 100 000 000 000
5635	6000	6000	/ 110 000 000 000

*1120

1120	4000	4000
1121	2000	2000
1122	1000	1000
1123	0400	0400
1124	0200	0200
1125	0100	0100
1126	0040	0040
1127	0020	0020
1130	0010	0010
1131	0004	0004
1132	0002	0002
1133	0001	0001
1134	0000	0000

MQA=7501

/REMISE DANS L'IMAGE DES POINTS QUE LE FILTRE
/AVAIT ENLEVES

*5600

5600	1236	TAD C177
5601	3010	DCA AUTI10
5602	1410	BCLE, TAD I AUTI10
5603	7450	SNA
5604	7402	HLT
5605	3025	DCA MOT
5606	1410	TAD I AUTI10
5607	3026	DCA ADBIT
5610	1426	TAD I ADBIT
5611	7421	MQL
5612	1425	TAD I MOT
5613	7501	MQA
5614	3425	DCA I MOT
5615	5202	JMP BCLE

*5636

5636 0177 C177,177

*1152

1152	4000	4000	/	100	000	000	000
1153	6000	6000	/	110	000	000	000
1154	7000	7000	/	111	000	000	000
1155	7400	7400	/	111	100	000	000
1156	7600	7600	/	111	110	000	000
1157	7700	7700	/	111	111	000	000
1160	7740	7740	/	111	111	100	000
1161	7760	7760	/	111	111	110	000
1162	7770	7770	/	111	111	111	000
1163	7774	7774	/	111	111	111	100
1164	7776	7776	/	111	111	111	110
1165	3777	3777	/	011	111	111	111
1166	1777	1777	/	001	111	111	111
1167	0777	0777	/	000	111	111	111
1170	0377	0377	/	000	011	111	111
1171	0177	0177	/	000	001	111	111
1172	0077	0077	/	000	000	111	111
1173	0037	0037	/	000	000	011	111
1174	0017	0017	/	000	000	001	111
1175	0007	0007	/	000	000	000	111
1176	0003	0003	/	000	000	000	011
1177	0001	0001	/	000	000	000	001

*1330

1330	0000	CHSEG,0
1331	7201	CLA IAC
1332	3036	DCA CTRM
1333	1033	TAD DI
1334	7041	CIA
1335	1374	TAD K4301
1336	7750	SPA SNA CLA
1337	5342	JMP.+3
1340	1374	TAD K4301
1341	3033	DCA DI
1342	1034	TAD DF
1343	7041	CIA
1344	1375	TAD K4351
1345	7700	SMA CLA
1346	5351	JMP.+3
1347	1375	TAD K4351
1350	3034	DCA DF

1351 1016 TAD MOTI
 1352 0104 AND K7
 1353 7450 SNA
 1354 5730 JMP I CHSEG
 1355 7040 CMA
 1356 0104 AND K7
 1357 7650 SNA CLA
 1360 5730 JMP I CHSEG

1361 1016 TAD MOTI
 1362 7041 CIA
 1363 1113 TAD K2000
 1364 7700 SMA CLA
 1365 5730 JMP I CHSEG
 1366 1016 TAD MOTI
 1367 7041 CIA
 1370 1376 TAD K3017
 1371 7710 SPA CLA
 1372 5730 JMP I CHSEG
 1373 5377 JMP FPAGE

1374 4301 K4301, 4301
 1375 4351 K4351, 4351
 1376 3017 K3017, 3017

*1377

1377 1330 FPAGE, TAD CHSEG
 1400 3037 DCA CHSEGS /CONSERVE L'ADRESSE DE RETOUR DANS
 1401 1034 TAD DF /LA PAGE 0.
 1402 7041 CIA
 1403 1033 TAD DI
 1404 1103 TAD K10D /AC=10-(DF-DI)
 1405 7700 SMA CLA
 1406 5377 JMP EN38

1407 1033 TAD DI
 1410 7041 CIA
 1411 1335 TAD K4326 /AC=4326-DI
 1412 7750 SPA SNA CLA
 1413 5247 JMP EN37 /SI DI EST PLUS GRAND =4326

1414 1030 TAD CRITER
 1415 7041 CIA
 1416 3040 DCA CTR
 1417 1017 TAD ADBITI
 1420 1336 TAD K44
 1421 3041 DCA MASQUE
 1422 1441 TAD I MASQUE
 1423 3041 DCA MASQUE
 1424 1016 TAD MOTI
 1425 3025 DCA MOT
 1426 4350 JMS COUNT
 1427 5247 JMP EN37

1430 1041 TAD MASQUE
1431 0337 AND K6000
1432 7640 SZA CLA
1433 5245 JMP EN365
1434 1041 TAD MASQUE
1435 7124 CLL CML RAL
1436 7040 CMA
1437 3041 DCA MASQUE
1440 1016 TAD MOTI
1441 7001 IAC
1442 3025 DCA MOT
1443 4310 JMS COMPTE
1444 5247 JMP EN37

1445 1335 EN365, TAD K4326
1446 3033 DCA DI

1447 1034 EN37, TAD DF
1450 7041 CIA
1451 1340 TAD K4324
1452 7700 SMA CLA
1453 5377 JMP EN38

1454 1017 TAD ADBITI
1455 1341 TAD K32
1456 3041 DCA MASQUE
1457 1441 TAD I MASQUE
1460 3041 DCA MASQUE
1461 1016 TAD MOTI
1462 3025 DCA MOT
1463 1030 TAD CRITER
1464 7041 CIA
1465 3040 DCA CTR
1466 4310 JMS COMPTE
1467 5377 JMP EN38

1470 1041 TAD MASQUE
1471 0342 AND K3
1472 7640 SZA CLA
1473 5305 JMP EN375
1474 7240 CLA CMA
1475 1016 TAD MOTI
1476 3025 DCA MOT
1477 1041 TAD MASQUE
1500 7130 CLL CML RAR
1501 7040 CMA
1502 3041 DCA MASQUE
1503 4350 JMS COUNT
1504 5377 JMP EN38

1505 1340 EN375, TAD K4324
1506 3034 DCA DF
1507 5377 JMP EN38

1510	0000	COMPTE,0
1511	1343	TAD M13
1512	3344	DCA CTV
1513	1041	ALPHA,TAD MASQUE
1514	0425	AND I MOT
1515	7450	BETA,SNA
1516	5326	JMP GAMMA
1517	7104	CLL RAL
1520	7420	SNL
1521	5317	JMP.-2
1522	2040	ISZ CTR
1523	5315	JMP BETA
1524	7200	CLA
1525	5710	JMP I COMPTE
1526	1025	GAMMA,TAD MOT
1527	1102	TAD K8
1530	3025	DCA MOT
1531	2344	ISZ CTV
1532	5313	JMP ALPHA
1533	2310	ISZ COMPTE
1534	5710	JMP I COMPTE
1535	4326	K4326,4326
1536	0044	K44,44
1537	6000	K6000,6000
1540	4324	K4324,4324
1541	0032	K32,32
1542	0003	K3,3
1543	7765	M13,7765
1544	0000	CTV,0

*1550

1550	0000	COUNT,0
1551	1343	TAD M13
1552	3344	DCA: CTV
1553	1041	ALFA, TAD MASQUE
1554	0425	AND I MOT
1555	7450	BETTA, SNA
1556	5366	JMP GAMA
1557	7110	CLL RAR
1560	7420	SNL
1561	5357	JMP. -2
1562	2040	ISZ CTR
1563	5355	JMP BETTA
1564	7200	CLA
1565	5750	JMP I COUNT
1566	1025	GAMA, TAD MOT
1567	1102	TAD K8
1570	3025	DCA MOT
1571	2344	ISZ CTV
1572	5353	JMP ALFA
1573	2350	ISZ COUNT
1574	5750	JMP I COUNT

*1577

1577	7200	EN38, CLA
------	------	-----------

*1600
 1600 1034 TAD DF
 1601 7001 IAC
 1602 7041 CIA
 1603 1033 TAD DI
 1604 3263 DCA CD
 1605 1263 TAD CD
 1606 7700 SMA CLA
 1607 5437 JMP I CHSEGS
 1610 1033 TAD DI
 1611 3024 DCA D
 1612 4514 JMS I AMODE1

1613 4505 BCLED, JMS I ADSUIVRE
 1614 7410 SKP
 1615 5227 JMP EN99
 1616 1032 TAD CTBLANC
 1617 7041 CIA
 1620 1036 TAD CTRM
 1621 7750 SPA SNA CLA
 1622 5227 JMP EN99

1623 1024 TAD D
 1624 3035 DCA BEST
 1625 1032 TAD CTBLANC
 1626 3036 DCA CTRM

1627 2024 EN99, ISZ D
 1630 2263 ISZ CD
 1631 5213 JMP BCLED
 1632 1036 TAD CTRM
 1633 7740 SMA SZA CLA
 1634 5437 JMP I CHSEGS

1635 1035 EN100, TAD BEST
 1636 3024 DCA D
 1637 4515 JMS I AMODE2
 1640 1264 TAD ADDERNPT
 1641 3044 DCA ADRSR
 1642 4505 JMS I ADSUIVRE

1643 1265 TAD ADEFF
 1644 3044 DCA ADRSR
 1645 4505 JMS I ADSUIVRE
 1646 1443 TAD I MOTF
 1647 1442 TAD I ADBITF
 1650 3443 DCA I MOTF

/POUR QUE LE DERNIER POINT DU SEGMENT
 /RESTE NOIR.

1651 1042 TAD ADBITF
 1652 4516 JMS I ACHSYAL
 1653 3042 DCA ADBITF

/POUR QUE ADBITF SOIT UN MASQUE DE LA
 /1 ERE LISTE: 1120≤ADBITF≤1133

1654 1036 TAD CTRM
 1655 1030 TAD CRITER
 1656 7041 CIA
 1657 1103 TAD K10D
 1660 3036 DCA CTRM
 1661 2037 ISZ CHSEGS
 1662 5437 JMP I CHSEGS

1663 0000 CD, 0
 1664 1236 ADDERNPT, DERNPT
 1665 1230 ADEFF, EFF

/S.R. EFF

*1230

1230	0000	EFF,0
1231	7040	CMA
1232	0425	AND I MOT
1233	3425	DCA I MOT
1234	5630	JMP I EFF

*1236

1236	0000	DERNPT,0	/POUR TROUVER LE DERNIER POINT NOIR D'UN
1237	0425	AND I MOT	/SEGMENT
1240	7650	SNA CLA	
1241	5636	JMP I DERNPT	
1242	1025	TAD MOT	
1243	3043	DCA MOTF	
1244	1026	TAD ADBIT	
1245	3042	DCA ADBITF	
1246	5636	JMP I DERNPT	

		*1710
1710	7200	CLA
1711	4763	JMS I ADAVANT
1712	1107	TAD NO1CRITER
1713	3030	DCA CRITER
1714	3045	DCA NTRF
1715	1354	TAD M21D
1716	3047	DCA CNTRFMAX
1717	1113	TAD K2000
1720	3016	DCA MOTI
1721	1362	TAD M66D
1722	3356	DCA CTLIGNES
1723	1355	TAD K3377
1724	3014	DCA ATRF
1725	1357	DO499LIGNES, TAD MOINS6
1726	3360	DCA CTMOTS
1727	1416	DO499MOTS, TAD I MOTI
1730	7450	SNA
1731	5345	JMP EN499MOTS
1732	3050	DCA ACT
1733	1120	TAD K1117
1734	3017	DCA ADBITI
1735	1417	DO499BIT, TAD I ADBITI
1736	7450	SNA
1737	5345	JMP EN499MOTS
1740	0050	AND ACT
1741	7650	SNA CLA
1742	7410	SKP
1743	4761	JMS I ADYAUNPT
1744	5335	JMP DO499BIT
1745	2360	EN499MOTS, ISZ CTMOTS
1746	5327	JMP DO499MOTS
1747	2016	ISZ MOTI
1750	2016	ISZ MOTI
1751	2356	ISZ CTLIGNES
1752	5325	JMP DO499LIGNES
1753	5764	EN500, JMP I ADPROLONGE
1754	7753	M21D, 7753
1755	3377	K3377, 3377
1756	0000	CTLIGNES, 0
1757	7772	MOINS6, 7772
1760	0000	CTMOTS, 0
1761	4200	ADYAUNPT, 4200
1762	7676	M66D, 7676
1763	4756	ADAVANT, 4756
1764	5160	ADPROLONGE, 5160

*4200
4200 0000 YAUNPT,0
4201 1016 TAD MOTI
4202 7421 MQL
4203 1017 TAD ADBITI
4204 4524 JMS I ADCONV
4205 3020 DCA HI
4206 7701 ACL
4207 3021 DCA VI
4210 4517 JMS I ADEVPROX
4211 5600 JMP I YAUNPT

4212 1252 TAD K4301
4213 3033 DCA DI
4214 1253 TAD K4350
4215 3034 DCA DF
4216 1107 TAD NO1CRITER
4217 3030 DCA CRITER
4220 4506 JMS I ADCHSEG
4221 5600 JMP I YAUNPT

4222 1110 TAD NO2CRITER
4223 3030 DCA CRITER
4224 1016 TAD MOTI
4225 3052 DCA MOTIR
4226 1017 TAD ADBITI
4227 3053 DCA ADBITR

4230 1020 TAD HI
4231 3414 DCA I ATRF
4232 1021 TAD VI
4233 3414 DCA I ATRF
4234 1043 TAD MOTF
4235 7421 MQL
4236 1042 TAD ADBITF
4237 4524 JMS I ADCONV
4240 3414 DCA I ATRF
4241 7701 ACL
4242 3414 DCA I ATRF
4243 1035 TAD BEST
4244 3414 DCA I ATRF
4245 7201 CLA IAC
4246 3414 DCA I ATRF
4247 1200 TAD YAUNPT
4250 3051 DCA YA1PTS
4251 5654 JMP I ADEN50

4252 4301 K4301,4301
4253 4350 K4350,4350
4254 4000 ADEN50,4000

*4000

F-7/3

4000 1435 EN50, TAD I BEST
4001 7112 CLL RTR
4002 7620 SNL CLA
4003 5206 JMP.+3
4004 1123 TAD ADEPV
4005 7410 SKP
4006 1122 TAD ADEPH
4007 3044 DCA ADRSR
4010 4515 JMS I AMODE2

4011 1043 TAD MOTF
4012 3016 DCA MOTI
4013 1042 TAD ADBITF
4014 3017 DCA ADBITI
4015 1360 TAD K4477
4016 3015 DCA AI15

4017 1016 TAD MOTI
4020 3025 DCA MOT
4021 1017 TAD ADBITI
4022 3026 DCA ADBIT
4023 1361 TAD M6
4024 3040 DCA CTR
4025 4444 JMS I ADRSR

4026 4505 JMS I ADSUIVRE

/LES 30 POINTS SONT EN MEMOIRE DE
/4500 A 4574

4027 1360 TAD K4477
4030 3015 DCA AI15
4031 1361 TAD M6
4032 3372 DCA JJ
4033 1362 TAD M5
4034 3371 DCA II
4035 7240 CLA CMA
4036 1014 TAD ATRF
4037 3014 DCA ATRF
4040 7240 CLA CMA
4041 1414 TAD I ATRF
4042 7750 SPA SNA CLA
4043 7001 IAC
4044 7001 IAC
4045 3364 DCA ASSEZ
4046 7240 CLA CMA
4047 1364 TAD ASSEZ
4050 3365 DCA ASSEZM1

4051 1415 D0399, TAD I AI15
4052 3025 DCA MOT
4053 1415 TAD I AI15
4054 3026 DCA ADBIT
4055 1425 TAD I MOT
4056 0426 AND I ADBIT
4057 7650 SNA CLA
4060 5332 JMP EN399
4061 1371 TAD II
4062 1363 TAD K2
4063 7700 SMA CLA
4064 5267 JMP.+3
4065 1364 TAD ASSEZ
4066 7410 SKP
4067 1365 TAD ASSEZM1
4070 3366 DCA ACTM

4071	1366	TAD ACTM
4072	7041	CIA
4073	1035	TAD BEST
4074	3033	DCA DI
4075	1366	TAD ACTM
4076	1035	TAD BEST
4077	3034	DCA DF
4100	1025	TAD MOT
4101	3016	DCA MOTI
4102	1026	TAD ADBIT
4103	3017	DCA ADBITI
4104	4506	JMS I ADCHSEG
4105	5332	JMP EN399
4106	1043	TAD MOTF
4107	7421	MQL
4110	1042	TAD ADBITF
4111	4524	JMS I ADCONV
4112	3022	DCA HF
4113	7701	ACL
4114	3023	DCA VF
4115	4521	JMS I ADMOYEN
4116	3035	DCA BEST
4117	1367	TAD M4
4120	1014	TAD ATRF
4121	3014	DCA ATRF
4122	1022	TAD HF
4123	3414	DCA I ATRF
4124	1023	TAD VF
4125	3414	DCA I ATRF
4126	1035	TAD BEST
4127	3414	DCA I ATRF
4130	2414	ISZ I ATRF
4131	5200	JMP EN50
4132	2371	EN399, ISZ II
4133	5251	JMP D0399
4134	1362	TAD M5
4135	3371	DCA II
4136	2372	ISZ JJ
4137	5251	JMP D0399
4140	1052	TAD MOTIR
4141	3016	DCA MOTI
4142	1053	TAD ADBITR
4143	3017	DCA ADBITI
4144	1452	TAD I MOTIR
4145	3050	DCA ACT
4146	2045	ISZ NTRF
4147	2047	ISZ CNTRFMAX
4150	5451	JMP I YA1PTS
4151	5770	JMP I AEN500

*41 60
41 60. 4477 K4477, 4477
41 61 7772 M6, 7772
41 62 7773 M5, 7773
41 63 0002 K2, 2
41 64 0000 ASSEZ, 0
41 65 0000 ASSEZM1, 0
41 66 0000 ACTM, 0
41 67 7774 M4, 7774
41 70 1752 AEN500, 1752
41 71 0000 II, 0
41 72 0000 JJ, 0

/DOIT ETRE 1753. CORRIGE SUR LES GRANDS
/RUBANS.

*3600

3600	0000	DEPH,0
3601	7200	CLA
3602	1346	TAD M5
3603	3321	DCA CT
3604	1230	TAD K3633
3605	3227	DCA ADRSPEC
3606	1026	TAD ADBIT
3607	4330	JMS CHSYAL
3610	3231	DCA ADBITP
3611	1025	BCLE, TAD MOT
3612	1627	TAD I ADRSPEC
3613	3415	DCA I AI15
3614	1231	TAD ADBITP
3615	3415	DCA I AI15
3616	2227	ISZ ADRSPEC
3617	2321	ISZ CT
3620	5211	JMP BCLE
3621	2040	ISZ CTR
3622	5600	JMP I DEPH
3623	7240	CLA CMA
3624	3031	DCA J
3625	5600	JMP I DEPH

3626	6644	M1134, 6644
3627	0000	ADRSPEC,0
3630	3633	K3633, 3633
3631	0000	ADBITP,0
3632	0000	MOTP,0

*3633

3633	0000	SPEC1,0
3634	0010	SPEC2,10
3635	7770	SPEC3,7770
3636	0020	SPEC4,20
3637	7760	SPEC5,7760

		*3640	
3640	0000	DEPV,0	
3641	7200	CLA	
3642	1346	TAD M5	
3643	3321	DCA CT	
3644	1313	TAD K3723	
3645	3314	DCA ADDRIF	
3646	1026	TAD ADBIT	
3647	4330	JMS CHSYAL	
3650	3322	DCA ABITPR	
3651	1025	BOUCLE, TAD MOT	
3652	3232	DCA MOTP	
3653	1322	TAD ABITPR	
3654	1714	TAD I ADDRIF	
3655	3231	DCA ADBITP	
3656	1231	TAD ADBITP	
3657	1315	TAD M1120	
3660	7500	SMA	
3661	5270	JMP.+7	
3662	1316	TAD K1134	/SI ADBITP<1120,ON FAIT ADBITP+14
3663	3231	DCA ADBITP	/ADBITP MOTP-1->MOTP
3664	7240	CLA CMA	
3665	1232	TAD MOTP	
3666	3232	DCA MOTP	
3667	5276	JMP DEP	
3670	1317	TAD M14	
3671	7510	SPA	
3672	5276	JMP DEP	/SI ADBITP>=1134,ON FAIT ADBITP-14
			/ADBITP MOTP+1->MOTP
3673	1320	TAD K1120	
3674	3231	DCA ADBITP	
3675	2232	ISZ MOTP	

3676 7200 DEP,CLA
3677 1232 TAD MOTP
3700 3415 DCA I AI15
3701 1231 TAD ADBITP
3702 3415 DCA I AI15
3703 2314 ISZ ADRDIF
3704 2321 ISZ CT
3705 5251 JMP BOUCLE

3706 2040 ISZ CTR
3707 5640 JMP I DEPV
3710 7240 CLA CMA
3711 3031 DCA J
3712 5640 JMP I DEPV

3713 3723 K3723,3723
3714 0000 ADRDIF,0
3715 6660 M1120,6660
3716 1134 K1134,1134
3717 7764 M14,7764
3720 1120 K1120,1120
3721 0000 CT,0
3722 0000 ABITPR,0

*3723

3723 0000 DIF1,0
3724 0001 DIF2,1
3725 7777 DIF3,7777
3726 0002 DIF4,2
3727 7776 DIF5,7776

*3200

3200	0000	EUPROX,0
3201	1045	TAD NTRF
3202	7450	SNA
3203	5347	JMP CAVA
3204	7041	CIA
3205	3360	DCA CNTRF
3206	1361	TAD K3377
3207	3015	DCA AI15
3210	1415	DO10K, TAD I AI15
3211	3362	DCA HIP
3212	1415	TAD I AI15
3213	3363	DCA VIP
3214	1415	TAD I AI15
3215	3022	DCA HF
3216	1415	TAD I AI15
3217	3023	DCA VF
3220	2015	ISZ AI15
3221	2015	ISZ AI15

3222	1363	TAD VIP
3223	7041	CIA
3224	1023	TAD VF
3225	3364	DCA DV
3226	1362	TAD HIP
3227	7041	CIA
3230	1022	TAD HF
3231	3365	DCA DH

3232	1365	TAD DH
3233	7500	SMA
3234	7041	CIA
3235	1364	TAD DV
3236	7700	SMA CLA
3237	5310	JMP EN5

/CECOMM., EST POUR LA LIGNE PREC., AC=
/ DV-ABS., DH

3240	1362	TAD HIP
3241	7041	CIA
3242	1020	TAD HI
3243	7700	SMA CLA
3244	5252	JMP.+6
3245	1020	TAD HI
3246	7041	CIA
3247	1022	TAD HF
3250	7700	SMA CLA
3251	5345	JMP EN10

/AC=HF-HI

3252	1362	TAD HIP
3253	7041	CIA
3254	1020	TAD HI
3255	7750	SPA SNA CLA
3256	5264	JMP.+6
3257	1020	TAD HI
3260	7041	CIA
3261	1022	TAD HF
3262	7750	SPA SNA CLA
3263	5345	JMP EN10

3264	1362	TAD HIP	
3265	7041	CIA	
3266	1020	TAD HI	
3267	7421	MQL	
3270	4500	JMS I AMULT	
3271	3364	DCA DV	/ADRESSE DE DV
3272	4501	JMS I ADIVIS	
3273	3365	DCA DH	/ADRESSE DE DH
3274	7701	ACL	
3275	1363	TAD VIP	
3276	3366	DCA V	
3277	1021	TAD VI	
3300	7041	CIA	
3301	1366	TAD V	
3302	7500	SMA	
3303	7041	CIA	
3304	1370	TAD TOLERE	
3305	7700	SMA CLA	
3306	5600	JMP I EVPROX	
3307	5345	JMP EN10	
3310	1363	EN5, TAD VIP	
3311	7041	CIA	
3312	1021	TAD VI	
3313	7710	SPA CLA	
3314	5345	JMP EN10	
3315	1023	TAD VF	
3316	7041	CIA	
3317	1021	TAD VI	
3320	7700	SMA CLA	
3321	5345	JMP EN10	
3322	1363	TAD VIP	
3323	7041	CIA	
3324	1021	TAD VI	
3325	7421	MQL	
3326	4500	JMS I AMULT	
3327	3365	DCA DH	
3330	4501	JMS I ADIVIS	
3331	3364	DCA DV	
3332	7701	ACL	
3333	1362	TAD HIP	
3334	3367	DCA H	
3335	1020	TAD HI	
3336	7041	CIA	
3337	1367	TAD H	
3340	7500	SMA	
3341	7041	CIA	
3342	1370	TAD TOLERE	
3343	7700	SMA CLA	
3344	5600	JMP I EVPROX	

3345 2360 EN10,ISZ CNTRF
3346 5210 JMP D010K

3347 2200 CAVA,ISZ EVPROX

/SI CA VA,ON REVIENT A LA 2 IEME INS
/TRUCTION QUI SUIT L'APPEL A LA S.R.

3350 5600 JMP I EVPROX

*3360

3360 0000 CNTRF,0

3361 3377 K3377,3377

3362 0000 HIP,0

3363 0000 VIP,0

3364 0000 DV,0

3365 0000 DH,0

3366 0000 V,0

3367 0000 H,0

3370 0002 TOLERE,2

*5160/PROLONGEMENT DES TRACES,
 5160 7200 CLA/PROGRAMME PRINCIPAL.
 5161 4773 JMS I ADAVANT
 5162 3067 DCA REBOURS
 5163 4775 JMS I APROLONGE
 5164 7201 CLA IAC
 5165 3067 DCA REBOURS
 5166 4774 JMS I ADARRIERE
 5167 4775 JMS I APROLONGE
 5170 7200 CLA
 5171 4773 JMS I ADAVANT
 5172 7402 HLT

5173 4756 ADAVANT, 4756
 5174 4764 ADARRIERE, 4764
 5175 5200 APROLONGE, 5200

/PROLONGEMENT DES TRACES-"PROGRAMME PRINCIPAL"

*5200
 5200 0000 PROLONGE, 0
 5201 7201 CLA IAC
 5202 1045 TAD NTRF
 5203 7041 CIA
 5204 3046 DCA K
 5205 1130 TAD K3377
 5206 3010 DCA AUTI10
 5207 1371 TAD ADPROL
 5210 3044 DCA ADRSR
 5211 4515 JMS I AMODE2

 5212 2046 EN590, ISZ K
 5213 5215 JMP BCL590
 5214 5600 JMP I PROLONGE

 5215 1067 BCL590, TAD REBOURS
 5216 7640 SZA CLA
 5217 5222 JMP.+3
 5220 2010 ISZ AUTI10
 5221 2010 ISZ AUTI10
 5222 1410 TAD I AUTI10
 5223 3020 DCA HI
 5224 1410 TAD I AUTI10
 5225 3021 DCA VI
 5226 1067 TAD REBOURS
 5227 7650 SNA CLA
 5230 5233 JMP.+3
 5231 2010 ISZ AUTI10
 5232 2010 ISZ AUTI10
 5233 1410 TAD I AUTI10
 5234 3066 DCA DD
 5235 2010 ISZ AUTI10
 5236 1021 TAD VI
 5237 7421 MQL
 5240 1020 TAD HI
 5241 4772 JMS I ACONVINV

5242	3025	DCA MOT
5243	7701	ACL
5244	3026	DCA ADBIT
5245	1370	TAD K4477
5246	3015	DCA AI15
5247	1466	TAD I DD
5250	7112	CLL RTR
5251	7620	SNL CLA
5252	5255	JMP.+3
5253	4523	JMS I ADEPV
5254	7410	SKP
5255	4522	JMS I ADEPH
5256	7240	CLA CMA
5257	1066	TAD DD
5260	7421	MQL
5261	1373	TAD K4301
5262	4525	JMS I ADMINIMAX
5263	3033	DCA DI
5264	7201	CLA IAC
5265	1066	TAD DD
5266	7421	MQL
5267	1374	TAD K4351
5270	4525	JMS I ADMINIMAX
5271	7701	ACL
5272	3034	DCA DF
5273	1126	TAD M5
5274	3376	DCA CTP
5275	1377	TAD K3
5276	3036	DCA CTRM
5277	1370	TAD K4477
5300	3015	DCA AI15
5301	1034	BCLPTI, TAD DF
5302	7040	CMA
5303	1033	TAD DI
5304	3367	DCA CD
5305	1415	TAD I AI15
5306	3016	DCA MOTI
5307	1415	TAD I AI15
5310	3017	DCA ADBITI
5311	1033	TAD DI
5312	3024	DCA D
5313	1143	BCLD, TAD M3
5314	3054	DCA CTBL
5315	3040	DCA CTR
5316	4505	JMS I ADSUIVRE
5317	1067	TAD REBOURS
5320	7640	SZA CLA
5321	5330	JMP FINI
5322	1025	TAD MOT
5323	3016	DCA MOTI
5324	1026	TAD ADBIT
5325	4516	JMS I ACHSYAL
5326	3017	DCA ADBITI
5327	5316	JMP.-11

5330	1066	FINI, TAD DD
5331	7041	CIA
5332	1024	TAD D
5333	7650	SNA CLA
5334	1376	TAD CTP
5335	1127	TAD K5
5336	7650	SNA CLA
5337	1377	TAD K3
5340	1040	TAD CTR
5341	3040	DCA CTR
5342	1040	TAD CTR
5343	7041	CIA
5344	1036	TAD CTRM
5345	7700	SMA CLA
5346	5355	JMP. +7
5347	1040	TAD CTR
5350	3036	DCA CTRM
5351	1043	TAD MOTF
5352	3052	DCA MOTIR
5353	1042	TAD ADBITF
5354	3053	DCA ADBITR
5355	2024	ISZ D
5356	2367	ISZ CD
5357	5313	JMP BCLD
5360	2376	ISZ CTP
5361	5301	JMP BCLPTI
5362	1036	TAD CTRM
5363	1143	TAD M3
5364	7640	SZA CLA
5365	4775	JMS I ADAILLEURS
5366	5212	JMP EN590
5367	0000	CD, 0
5370	4477	K4477, 4477
5371	5000	ADPROL, 5000
5372	3750	ACONVINV, 3750
5373	4301	K4301, 4301
5374	4351	K4351, 4351
5375	5066	ADAILLEURS, AILLEURS
5376	0000	CTP, 0
5377	0003	K3, 3

5066	0000	AILLEURS,0
5067	1010	TAD AUTI10
5070	3352	DCA AI10R
5071	1067	TAD REBOURS
5072	7640	SZA CLA
5073	1347	TAD M2
5074	1350	TAD M4
5075	1010	TAD AUTI10
5076	3010	DCA AUTI10
5077	1052	TAD MOTIR
5100	7421	MQL
5101	1053	TAD ADBITR
5102	4516	JMS I ACHSYAL
5103	4524	JMS I ADCONV
5104	3410	DCA I AUTI10
5105	7701	ACL
5106	3410	DCA I AUTI10
5107	1352	AIE, TAD AI10R
5110	1351	TAD M6
5111	3010	DCA AUTI10
5112	1410	TAD I AUTI10
5113	3020	DCA HI
5114	1410	TAD I AUTI10
5115	3021	DCA VI
5116	1410	TAD I AUTI10
5117	3022	DCA HF
5120	1410	TAD I AUTI10
5121	3023	DCA VF
5122	1021	TAD VI/SI VF<VI(I.E.SI LE POINT INITIAL
5123	7041	CIA/EST PLUS BAS QUE LE POINT FINAL)
5124	1023	TAD VF/IL FAUT ECHANGER LES POINTS INITIAL
5125	7700	SMA CLA/ET FINAL.
5126	5343	JMP CAVA
5127	1010	TAD AUTI10
5130	1350	TAD M4
5131	3010	DCA AUTI10
5132	1022	TAD HF
5133	3410	DCA I AUTI10
5134	1023	TAD VF
5135	3410	DCA I AUTI10
5136	1020	TAD HI
5137	3410	DCA I AUTI10
5140	1021	TAD VI
5141	3410	DCA I AUTI10
5142	5307	JMP AIE
5143	4521	CAVA, JMS I ADMOYEN
5144	3410	DCA I AUTI10
5145	2010	ISZ AUTI10
5146	5666	JMP I AILLEURS
5147	7776	M2, 7776
5150	7774	M4, 7774
5151	7772	M6, 7772
5152	0000	AI10R,0

*5000

5000	0000	PROL,0
5001	0260	AND K4001
5002	7450	SNA
5003	5227	JMP VERV
5004	7104	CLL RAL
5005	7620	SNL CLA
5006	5217	JMP.+11
5007	1027	TAD SG
5010	7710	SPA CLA
5011	5227	JMP VERV
5012	1025	TAD MOT
5013	0104	AND K7
5014	7640	SZA CLA
5015	5227	JMP VERV
5016	5256	JMP SORTIE
5017	1027	TAD SG
5020	7700	SMA CLA
5021	5227	JMP VERV
5022	1025	TAD MOT
5023	7001	IAC
5024	0104	AND K7
5025	7650	SNA CLA
5026	5256	JMP SORTIE
5027	1025	VERV, TAD MOT
5030	1261	TAD M2000
5031	7710	SPA CLA
5032	5256	JMP SORTIE
5033	1025	TAD MOT
5034	1262	TAD M3140
5035	7700	SMA CLA
5036	5256	JMP SORTIE
5037	1425	TAD I MOT
5040	0426	AND I ADBIT
5041	7640	SZA CLA
5042	5246	JMP.+4
5043	2054	ISZ CTBL
5044	5600	JMP I PROL
5045	5256	JMP SORTIE
5046	2040	ISZ CTR
5047	1143	TAD M3
5050	3054	DCA CTBL
5051	1025	TAD MOT
5052	3043	DCA MOTF
5053	1026	TAD ADBIT
5054	3042	DCA ADBITF
5055	5600	JMP I PROL
5056	7300	SORTIE, CLA CLL
5057	5663	JMP I AFINI
5060	4001	K4001,4001
5061	6000	M2000,6000
5062	4640	M3140,4640
5063	5330	AFINI,FINI

*6000

6000	7300	CLA CLL
6001	1126	TAD JESS
6002	3356	DCA ASSEP1
6003	2356	EN800, ISZ ASSEP1
6004	5206	JMP.+2
6005	7402	HLT
6006	1356	TAD ASSEP1
6007	1360	TAD JAD4
6010	3357	DCA ASSEZ
6011	1361	TAD JES3
6012	1357	TAD ASSEZ
6013	7640	SZA CLA
6014	5217	JMP.+3
6015	1362	TAD JAD2
6016	3357	DCA ASSEZ
6017	1045	TAD NTRF
6020	3056	DCA NTRFP
6021	7240	CLA CMA
6022	1045	TAD NTRF
6023	7750	SPA SNA CLA
6024	5203	JMP EN800
6025	7200	EN79V, CLA
6026	1056	TAD NTRFP
6027	7001	IAC
6030	7041	CIA
6031	3364	DCA KZZ
6032	3046	DCA K
6033	1130	TAD K3377
6034	3010	DCA AUTI10
6035	2046	EN795, ISZ K
6036	2364	ISZ KZZ
6037	5241	JMP.+2
6040	5531	JMP I AEN800
6041	1410	TAD I AUTI10
6042	3020	DCA HI
6043	1410	TAD I AUTI10
6044	3021	DCA VI
6045	1410	TAD I AUTI10
6046	3022	DCA HF
6047	1410	TAD I AUTI10
6050	3023	DCA VF
6051	1410	TAD I AUTI10
6052	3024	DCA D
6053	1410	TAD I AUTI10
6054	3060	DCA NS
6055	1046	TAD K
6056	7001	IAC
6057	3365	DCA KP1
6060	1021	TAD VI
6061	7041	CIA
6062	1023	TAD VF
6063	3354	DCA DV
6064	1020	TAD HI
6065	7041	CIA
6066	1022	TAD HF
6067	3353	DCA DH
6070	1353	TAD DH
6071	7510	SPA
6072	7041	CIA
6073	3355	DCA VADH

6074	1023	TAD VF
6075	1127	TAD JAD5
6076	3376	DCA VFP5
6077	1023	TAD VF
6100	1126	TAD JES5
6101	3375	DCA VFM5
6102	1020	TAD HI
6103	7421	MQL
6104	1022	TAD HF
6105	4525	JMS I ADMINIMAX
6106	3370	DCA HMAX
6107	7701	ACL
6110	3367	DCA HMIN
6111	1367	TAD HMIN
6112	1126	TAD JES5
6113	3371	DCA HMINM5
6114	1370	TAD HMAX
6115	1127	TAD JAD5
6116	3372	DCA HMAXP5
6117	1022	TAD HF
6120	1126	TAD JES5
6121	3373	DCA HFMS
6122	1022	TAD HF
6123	1127	TAD JAD5
6124	3374	DCA HFP5

6125	1130	TAD K3377
6126	3011	DCA AUTI11
6127	3057	DCA KK
6130	1045	TAD NTRF
6131	7040	CMA
6132	3352	DCA KKK /KKK=-(NTRF+1)
6133	7300	EN750, CLA CLL
6134	2057	ISZ KK
6135	2352	ISZ KKK
6136	5340	JMP.+2
6137	5235	JMP EN795
6140	1057	TAD KK
6141	7041	CIA
6142	1046	TAD K
6143	7640	SZA CLA
6144	5351	JMP.+5
6145	1011	TAD AUTI11
6146	1133	TAD K6
6147	3011	DCA AUTI11
6150	5333	JMP EN750
6151	5777	JMP I APRSRMM

6152	0000	KKK, 0
6153	0000	DH, 0
6154	0000	DV, 0
6155	0000	VADH, 0
6156	0000	ASSEPI, 0
6157	0000	ASSEZ, 0
6160	0004	JAD4, 4
6161	7775	JES3, 7775
6162	0002	JAD2, 2
6163	7776	JES2, 7776
6164	0000	KZZ, 0
6165	0000	KP1, 0
6166	7752	JES22, 7752
6167	0000	HMIN, 0
6170	0000	HMAX, 0
6171	0000	HMINM5, 0
6172	0000	HMAXP5, 0
6173	0000	HFM5, 0
6174	0000	HFP5, 0
6175	0000	VFM5, 0
6176	0000	VFP5, 0
6177	6225	.APRSRMM, 6225

*6225

6225	1411	TAD I AUTI11
6226	3062	DCA HHI
6227	1411	TAD I AUTI11
6230	3063	DCA VVI
6231	1411	TAD I AUTI11
6232	3064	DCA HHF
6233	1411	TAD I AUTI11
6234	3065	DCA VVF
6235	1411	TAD I AUTI11
6236	3066	DCA DD
6237	1411	TAD I AUTI11
6240	3061	DCA NNS
6241	1066	TAD DD
6242	7041	CIA
6243	1024	TAD D
6244	7510	SPA
6245	7041	CIA
6246	3315	DCA VADIFD
6247	1315	TAD VADIFD
6250	1316	TAD JES40
6251	7500	SMA
6252	7041	CIA
6253	1771	TAD I ADASSEZ
6254	7700	SMA CLA
6255	5722	JMP I AEN730
6256	1315	TAD VADIFD
6257	7041	CIA
6260	1771	TAD I ADASSEZ
6261	7710	SPA CLA
6262	5532	JMP I AEN750

6263	1424	TAD I D
6264	7112	CLL RTR
6265	7630	SZL CLA
6266	5723	JMP I AEN710
6267	1424	TAD I D
6270	7110	CLL RAR
6271	7630	SZL CLA
6272	5324	JMP EN707
6273	1062	TAD HHI
6274	7041	CIA
6275	1770	TAD I ADHFP5
6276	7740	SMA SZA CLA
6277	5301	JMP.+2
6300	5532	JMP I AEN750
6301	1020	TAD HI
6302	7041	CIA
6303	1062	TAD HHI
6304	7740	SMA SZA CLA
6305	7410	SKP
6306	5532	JMP I AEN750
6307	1064	TAD HHF
6310	7041	CIA
6311	1022	TAD HF
6312	7700	SMA CLA
6313	5532	JMP I AEN750
6314	5343	JMP EN708

6315	0000	VADIFD,0
6316	7730	JES40,7730
6317	7741	JES31,7741
6320	0013	JAD11,13
6321	0025	JAD21,25
6322	6450	AEN730,EN730
6323	6401	AEN710,EN710
/LES ENONCES:		

6324	1064	EN707, TAD HHF
6325	7041	CIA
6326	1022	TAD HF
6327	7750	SPA SNA CLA
6330	5532	JMP I AEN750
6331	1767	TAD I ADHFM5
6332	7041	CIA
6333	1062	TAD HHI
6334	7750	SPA SNA CLA
6335	5532	JMP I AEN750
6336	1062	TAD HHI
6337	7041	CIA
6340	1020	TAD HI
6341	7750	SPA SNA CLA
6342	5532	JMP I AEN750

6343	1020	EN708,	TAD HI
6344	7041	CIA	
6345	1062	TAD HHI	
6346	7421	MQL	
6347	4500	JMS I	AMULT
6350	6154	DV	
6351	4501	JMS I	ADIVIS
6352	6153	DH	
6353	7701	ACL	
6354	1021	TAD VI	
6355	3365	DCA V	
6356	1063	TAD VVI	
6357	7041	CIA	
6360	1365	TAD V	
6361	7510	SPA	
6362	7041	CIA	
6363	1766	TAD I	ADJES2
6364	5372	JMP.	+6
6365	0000	V,0	
6366	6163	ADJES2,	JES2
6367	6173	ADHFM5,	HFM5
6370	6174	ADHFP5,	HFP5
6371	6157	ADASSEZ,	ASSEZ
6372	7740	SMA SZA	CLA
6373	5532	JMP I	AEN750
6374	1064	TAD HHF	
6375	3022	DCA HF	
6376	1065	TAD VVF	
6377	3023	DCA VF	
6400	5766	JMP I	AEN770
6401	1023	EN710,	TAD VF
6402	7041	CIA	
6403	1065	TAD VVF	
6404	7750	SPA SNA	CLA
6405	5532	JMP I	AEN750
6406	1063	TAD VVI	
6407	7041	CIA	
6410	1767	TAD I	ADVFP5
6411	7710	SPA	CLA
6412	5532	JMP I	AEN750
6413	1021	TAD VI	
6414	7041	CIA	
6415	1063	TAD VVI	
6416	7750	SPA SNA	CLA
6417	5532	JMP I	AEN750

6420	1021	EN711, TAD VI
6421	7041	CIA
6422	1063	TAD VVI
6423	7421	MQL
6424	4500	JMS I AMULT
6425	6153	DH
6426	4501	JMS I ADIVIS
6427	6154	DV
6430	7701	ACL
6431	1020	TAD HI
6432	3362	DCA H
6433	1062	TAD HHI
6434	7041	CIA
6435	1362	TAD H
6436	7500	SMA
6437	7041	CIA
6440	1763	TAD I ADJAD2
6441	7710	SPA CLA
6442	5532	JMP I AEN750
6443	1064	TAD HHF
6444	3022	DCA HF
6445	1065	TAD VVF
6446	3023	DCA VF
6447	5766	JMP I AEN770

6450	1063	EN730, TAD VVI
6451	7041	CIA
6452	1021	TAD VI
6453	7510	SPA
6454	7041	CIA
6455	1764	TAD I ADJES3
6456	7740	SMA SZA CLA
6457	5340	JMP EN740
6460	1062	TAD HHI
6461	7041	CIA
6462	1020	TAD HI
6463	7510	SPA
6464	7041	CIA
6465	1126	TAD JESS
6466	7750	SPA SNA CLA
6467	5333	JMP EN734
6470	1424	TAD I D
6471	7110	CLL RAR
6472	7630	SZL CLA
6473	5314	JMP EN733
6474	1022	TAD HF
6475	7041	CIA
6476	1062	TAD HHI
6477	7740	SMA SZA CLA
6500	5340	JMP EN740
6501	1020	TAD HI
6502	7041	CIA
6503	1062	TAD HHI
6504	7710	SPA CLA
6505	5340	JMP EN740
6506	1064	TAD HHF
6507	7041	CIA
6510	1020	TAD HI
6511	7710	SPA CLA
6512	5340	JMP EN740
6513	5333	JMP EN734

6514	1062	EN733, TAD HHI
6515	7041	CIA
6516	1020	TAD HI
6517	7710	SPA CLA
6520	5340	JMP EN740
6521	1020	TAD HI
6522	7041	CIA
6523	1064	TAD HHF
6524	7710	SPA CLA
6525	5340	JMP EN740
6526	1022	TAD HF
6527	7041	CIA
6530	1062	TAD HHI
6531	7710	SPA CLA
6532	5340	JMP EN740

6533	1064	EN734, TAD HHF
6534	3020	DCA HI
6535	1065	TAD VVF
6536	3021	DCA VI
6537	5766	JMP I AEN770

6540	1065	EN740, TAD VVF
6541	7041	CIA
6542	1023	TAD VF
6543	7510	SPA
6544	7041	CIA
6545	1764	TAD I ADJES3
6546	7740	SMA SZA CLA
6547	5532	JMP I AEN750
6550	1064	TAD HHF
6551	7041	CIA
6552	1022	TAD HF
6553	7510	SPA
6554	7041	CIA
6555	1126	TAD JESS
6556	7750	SPA SNA CLA
6557	5765	JMP I AEN744
6560	5370	JMP. +10

6561	6614	AEN743, EN743
6562	0000	H, 0
6563	6162	ADJAD2, JAD2
6564	6161	ADJES3, JES3
6565	6633	AEN744, EN744
6566	6637	AEN770, EN770
6567	6176	ADVFP5, VFP5

6570	1424	TAD I D
6571	7110	CLL RAR
6572	7630	SZL CLA
6573	5761	JMP I AEN743
6574	1064	TAD HHF
6575	7041	CIA
6576	1022	TAD HF
6577	7710	SPA CLA
6600	5532	JMP I AEN750
6601	1022	TAD HF
6602	7041	CIA
6603	1062	TAD HHI
6604	7710	SPA CLA
6605	5532	JMP I AEN750

6606 1020 TAD HI
6607 7041 CIA
6610 1064 TAD HHF
6611 7710 SPA CLA
6612 5532 JMP I AEN750
6613 5233 JMP EN744

6614 1062 EN743, TAD HHI
6615 7041 CIA
6616 1022 TAD HF
6617 7710 SPA CLA
6620 5532 JMP I AEN750
6621 1022 TAD HF
6622 7041 CIA
6623 1064 TAD HHF
6624 7710 SPA CLA
6625 5532 JMP I AEN750
6626 1064 TAD HHF
6627 7041 CIA
6630 1020 TAD HI
6631 7710 SPA CLA
6632 5532 JMP I AEN750

6633 1062 EN744, TAD HHI
6634 3022 DCA HF
6635 1063 TAD VVI
6636 3023 DCA VF

6637 4344 EN770, JMS PFETPI

6640 1046 EN771, TAD K
6641 7421 MQL
6642 1057 TAD KK
6643 4525 JMS I ADMINIMAX
6644 3057 DCA KK
6645 7701 ACL
6646 3336 DCA II
6647 7240 CLA CMA
6650 1336 TAD II
6651 7421 MQL
6652 4500 JMS I AMULT
6653 0133 K6

6654 7701 ACL
6655 1130 TAD K3377
6656 3012 DCA AUTI12
6657 1020 TAD HI
6660 3412 DCA I AUTI12
6661 1021 TAD VI
6662 3412 DCA I AUTI12
6663 1022 TAD HF
6664 3412 DCA I AUTI12
6665 1023 TAD VF
6666 3412 DCA I AUTI12
6667 4521 JMS I ADMOYEN
6670 3412 DCA I AUTI12
6671 1060 TAD NS
6672 1061 TAD NNS
6673 3412 DCA I AUTI12
6674 7240 CLA CMA

6675	1045	TAD NTRF	
6676	3045	DCA NTRF	
6677	1045	TAD NTRF	
6700	7041	CIA	
6701	1057	TAD KK	
6702	7740	SMA SZA CLA	
6703	5531	JMP I AEN800	
6704	1045	TAD NTRF	
6705	7040	CMA	
6706	1057	TAD KK	
6707	7421	MQL	
6710	4500	JMS I AMULT	
6711	0133	K6	
6712	7701	ACL	
6713	3336	DCA II	/II = -(NTRF+1-KK)*6
714	7240	CLA CMA	
6715	1057	TAD KK	
6716	7421	MQL	
6717	4500	JMS I AMULT	
6720	0133	K6	
6721	7701	ACL	
6722	1130	TAD K3377	
6723	3013	DCA AUTI13	
6724	1013	TAD AUTI13	
6725	1133	TAD K6	
6726	3012	DCA AUTI12	
6727	1412	TAD I AUTI12	
6730	3413	DCA I AUTI13	
6731	2336	ISZ II	
6732	5327	JMP.-3	
6733	5737	JMP I AEN795	
6734	6365	ADV, V	
6735	6562	ADH, H	
6736	0000	II, 0	
6737	6035	AEN795, EN795	

*6744

6744	0000	PFETPI, 0
6745	7200	CLA
6746	1021	TAD VI
6747	7041	CIA
6750	1023	TAD VF
6751	7510	SPA
6752	5362	JMP CHANGE
6753	7640	SZA CLA
6754	5744	JMP I PFETPI
6755	1022	TAD HF
6756	7041	CIA
6757	1020	TAD HI
6760	7710	SPA CLA
6761	5744	JMP I PFETPI

6762	7200	CHANGE, CLA
6763	1020	TAD HI
6764	3050	DCA ACT
6765	1022	TAD HF
6766	3020	DCA HI
6767	1050	TAD ACT
6770	3022	DCA HF

6771	1021	TAD VI
6772	3050	DCA ACT
6773	1023	TAD VF
6774	3021	DCA VI
6775	1050	TAD ACT
6776	3023	DCA VF
6777	5744	JMP I PFETPI

*6200

```

6200 0000 MINIMAX, 0
6201 3224 DCA BLTPRZ
6202 7701 ACL
6203 3223 DCA BLPTRY
6204 1223 TAD BLPTRY
6205 7041 CIA
6206 1224 TAD BLTPRZ
6207 7510 SPA /AC=BLTPRZ-BLTPRY
6210 5216 JMP.+6
6211 7200 CLA
6212 1223 TAD BLPTRY
6213 7421 MQL
6214 1224 TAD BLTPRZ
6215 5600 JMP I MINIMAX
6216 7200 CLA
6217 1224 TAD BLTPRZ
6220 7421 MQL
6221 1223 TAD BLPTRY /MAXIMUM DANS AC
6222 5600 JMP I MINIMAX /MINIMUM DANS MQ

6223 0000 BLPTRY, 0
6224 0000 BLTPRZ, 0

```

*7000

```

7000 7200 CLA
7001 1045 TAD NTRF
7002 7041 CIA
7003 1135 TAD EXCES
7004 7710 SPA CLA /AC=EXCES-NTRF
7005 5534 JMP I AEUR
7006 1130 TAD K3377
7007 3010 DCA AUTI10
7010 1045 TAD NTRF
7011 7040 CMA
7012 3366 DCA CTK
7013 3046 DCA K

7014 2366 BCLEK, ISZ CTK /POUR K DE 1 A NTRF.
7015 7410 SKP
7016 5544 JMP I ADTBK /VA RECEVOIR LA PROCHAINE IMAGE.

7017 2046 ISZ K
7020 1410 TAD I AUTI10
7021 3020 DCA HI
7022 1410 TAD I AUTI10
7023 3021 DCA VI
7024 1410 TAD I AUTI10
7025 3022 DCA HF
7026 1410 TAD I AUTI10
7027 3023 DCA VF
7030 1410 TAD I AUTI10
7031 3024 DCA D
7032 1410 TAD I AUTI10
7033 3060 DCA NS

```

7034	1021	TAD VI	
7035	7041	CIA	
7036	1023	TAD VF	
7037	3072	DCA DV	
7040	1020	TAD HI	
7041	7041	CIA	
7042	1022	TAD HF	
7043	3073	DCA DH	
7044	1073	TAD DH	
7045	7510	SPA	
7046	7041	CIA	
7047	7421	MQL	
7050	1072	TAD DV	
7051	4525	JMS I ADMINIMAX	
7052	3367	DCA DIFMAX	
7053	1367	TAD DIFMAX	
7054	7041	CIA	
7055	1136	TAD CRITR1	
7056	7740	SMA SZA CLA	/AC=CRITR1-DIFMAX
7057	5214	JMP BCLEK	
7060	1143	TAD M3	
7061	3370	DCA CTMEC	
7062	2370	BCLEMEC, ISZ CTMEC	/POUR CTMEC=-2,-1
7063	7410	SKP	
7064	5214	JMP BCLEK	
7065	7201	CLA IAC	
7066	1370	TAD CTMEC	
7067	7650	SNA CLA	
7070	5276	JMP.+6	
7071	1020	TAD HI	
7072	3070	DCA H	
7073	1021	TAD VI	
7074	3071	DCA V	
7075	5302	JMP.+5	
7076	1022	TAD HF	
7077	3070	DCA H	
7100	1023	TAD VF	
7101	3071	DCA V	
7102	1130	TAD K3377	
7103	3011	DCA AUTI11	
7104	1045	TAD NTRF	
7105	7040	CMA	
7106	3371	DCA CTKK	
7107	3057	DCA KK	
7110	2371	BCLEKK, ISZ CTKK	/POUR KK DE 1 A NTRF (SAUF KK=K)
7111	7410	SKP	
7112	5262	JMP BCLEMEC	
7113	2057	ISZ KK	
7114	1057	TAD KK	
7115	7041	CIA	
7116	1046	TAD K	
7117	7640	SZA CLA	
7120	5325	JMP.+5	
7121	1011	TAD AUTI11	
7122	1133	TAD K6	
7123	3011	DCA AUTI11	
7124	5310	JMP BCLEKK	

7125	1411	TAD I AUTI11	
7126	3062	DCA HHI	
7127	1411	TAD I AUTI11	
7130	3063	DCA VVI	
7131	1411	TAD I AUTI11	
7132	3064	DCA HHF	
7133	1411	TAD I AUTI11	
7134	3065	DCA VVF	
7135	1411	TAD I AUTI11	
7136	3066	DCA DD	
7137	1411	TAD I AUTI11	
7140	3061	DCA NNS	
7141	1063	TAD VVI	
7142	7041	CIA	
7143	1065	TAD VVF	
7144	3076	DCA DDV	
7145	1062	TAD HHI	
7146	7041	CIA	
7147	1064	TAD HHF	
7150	3077	DCA DDH	
7151	1077	TAD DDH	
7152	7510	SPA	
7153	7041	CIA	
7154	7421	MQL	
7155	1076	TAD DDV	
7156	4525	JMS I ADMINIMAX	
7157	3372	DCA DDIFMAX	
7160	1372	TAD DDIFMAX	
7161	7041	CIA	
7162	1137	TAD CRITR2	
7163	7740	SMA SZA CLA	/AC=CRITR2-DDIFMAX
7164	5310	JMP BCLEKK	
7165	5377	JMP FPAGE	

7166	0000	CTK,0	
7167	0000	DIFMAX,0	
7170	0000	CTMEC,0	
7171	0000	CTKK,0	
7172	0000	DDIFMAX,0	

*7177

7177	1066	FPAGE, TAD DD	
7200	7041	CIA	
7201	1024	TAD D	
7202	7500	SMA	
7203	7041	CIA	
7204	1140	TAD CRITR3	/AC=CRITR3-!D-DD!
7205	7740	SMA SZA CLA	
7206	5766	JMP I ABKK	

7207	1143	TAD M3
7210	3367	DCA CTMMEC
7211	2367	BCLEMMEC, ISZ CTMMEC /POUR CTMMEC=-2,-1.
7212	7410	SKP
7213	5254	JMP BETA
7214	7201	CLA IAC
7215	1367	TAD CTMMEC
7216	7650	SNA CLA
7217	5225	JMP.+6
7220	1062	TAD HHI
7221	3074	DCA HH
7222	1063	TAD VVI
7223	3075	DCA VV
7224	5231	JMP.+5
7225	1064	TAD HHF
7226	3074	DCA HH
7227	1065	TAD VVF
7230	3075	DCA VV
7231	1075	TAD VV
7232	7041	CIA
7233	1071	TAD V
7234	7510	SPA
7235	7041	CIA
7236	7421	MLL
7237	1074	TAD HH
7240	7041	CIA
7241	1070	TAD H
7242	7510	SPA
7243	7041	CIA
7244	4525	JMS I ADMINIMAX
7245	3370	DCA MAXDIF
7246	1370	TAD MAXDIF
7247	7041	CIA
7250	1141	TAD CRITR4
7251	7700	SMA CLA /AC=CRITR4-MAXDIF.
7252	5534	JMP I AEUR /IL Y A UNE ETOILE!
7253	5211	JMP BCLEMMEC

/CHERCHE SI LES TRACES K ET KK
 /FORMENT UN "T".
 /SI OUI, C'EST UNE ETOILE.

7254	1077	BETA, TAD DDH
7255	7500	SMA
7256	7041	CIA
7257	1076	TAD DDV
7260	7740	SMA SZA CLA /AC=DDV-1 DDH!
7261	5330	JMP TRVERT

7262	1062	TRHORIZ, TAD HHI	
7263	7041	CIA	
7264	1070	TAD H	
7265	7450	SNA	/ AC=H-HHI
7266	5304	JMP OUIHOR	
7267	7710	SPA CLA	
7270	5277	JMP.+7	
7271	1070	TAD H	
7272	7041	CIA	
7273	1064	TAD HRF	
7274	7710	SPA CLA	
7275	5766	JMP I ABKK	
7276	5304	JMP OUIHOR	
7277	1070	TAD H	
7300	7041	CIA	
7301	1064	TAD HRF	
7302	7740	SMA SZA CLA	
7303	5766	JMP I ABKK	
7304	1062	OUIHOR, TAD HHI	
7305	7041	CIA	
7306	1070	TAD H	
7307	7421	MQL	
7310	4500	JMS I AMULT	
7311	0076	DDV	
7312	4501	JMS I ADIVIS	
7313	0077	DDH	
7314	7701	ACL	
7315	1063	TAD VVI	
7316	3075	DCA VV	
7317	1071	TAD V	
7320	7041	CIA	
7321	1075	TAD VV	
7322	7500	SMA	
7323	7041	CIA	
7324	1142	TAD CRITR5	/AC=CRITR5-! VV-V!
7325	7700	SMA CLA	
7326	5534	JMP I AEUR	
7327	5766	JMP I ABKK	
7330	1063	TRVERT, TAD VVI	
7331	7041	CIA	
7332	1071	TAD V	
7333	7710	SPA CLA	/AC=V-VVI
7334	5766	JMP I ABKK	
7335	1071	TAD V	
7336	7041	CIA	
7337	1065	TAD VVF	
7340	7710	SPA CLA	/AC=VVF-V
7341	5766	JMP I ABKK	

7342	1063	OUIVERT, TAD VVI	
7343	7041	CIA	
7344	1071	TAD V	
7345	7421	MQL	
7346	4500	JMS I AMULT	
7347	0077	DDH	
7350	4501	JMS I ADIVIS	
7351	0076	DDV	
7352	7701	ACL	
7353	1062	TAD HHI	
7354	3074	DCA HH	
7355	1070	TAD H	
7356	7041	CIA	
7357	1074	TAD HH	
7360	7500	SMA	
7361	7041	CIA	
7362	1142	TAD CRITR5	/AC=CRITR5-!HH-H!
7363	7700	SMA CLA	
7364	5534	JMP I AEUR	/IL Y A UNE ETOILE!
7365	5766	JMP I ABKK	

7366	7110	ABKK, BCLEKK
7367	0000	CTMMEC, 0
7370	0000	MAXDIF, 0

*7372

/VIENT A EUREKA SI UNE ETOILE
/A ETE TROUVEE.

7372	7200	EUREKA, CLA
7373	6046	TLS
7374	1377	TAD CLOCHE
7375	4547	JMS I ADTAPE
7376	7402	HLT
7377	0207	CLOCHE, 207

Mode d'emploi

Voici d'abord les adresses de départ des sept étapes:

Etape	Adresse de départ
1. Transmission de l'image	0600
2.* Filtrage	5400
3. Recherche des traces	1710
4.* Remise dans l'image des points enlevés par le filtre	5600
5. Prolongement des traces	5160
6. Assemblage des traces	6000
7. Recherche des étoiles	7000
* Facultatifs	

Normalement, on fait commencer le programme à 0600. Les sept étapes seront alors effectuées. S'il y a événement, la cloche du Télétype sonne et l'ordinateur arrête. Sinon, il va à 0600 recevoir une autre image.

Pour tester avec des événements codés sur ruban perforé*, on fait lire le ruban BIN selon la procédure normale (avec le "BIN LOADER", adresse 7777). On met une halte (7402) dans 0600 pour que le programme s'y arrête s'il n'y a pas d'événement. Puis on démarre à 5400 ou 1710, selon qu'on veut du filtrage ou pas.

Pour dessiner l'image ou les traces, après une étape, mettre une instruction d'arrêt (7402) au début de l'étape suivante. Démarrer

* Codés manuellement et transformés en code binaire (BIN) où envoyés sur ruban après transmission, au moyen du programme "ODT". Le ruban contient, en code BIN, les contenus des mots 2000 à 3137.

le programme à l'adresse normale (0600, 5400 ou 1710, selon le cas).
A l'arrêt, faire exécuter le programme de dessin désiré.

En général, après le dessin des traces, on ne peut faire continuer le programme à partir du point où il a été interrompu pour dessiner. Le programme de dessin des traces utilise les mots 2000 à 2140 comme mémoire tampon.

Si on utilise le filtre, la lecture d'un programme de dessin détruit la liste des points mis de côté par le filtre (voir ci-dessous).

Les critères de l'étape 3, NO CRITER et NO 2 CRITER sont dans les mots 0107 et 0110 respectivement.

Grand ruban A (sans filtrage)

Ce programme exécute la version sans filtrage de l'algorithme. Les étapes 2 et 4 sont absentes. Les liaisons entre les cinq étapes sont incluses. Les programmes de dessin sont inclus dans le ruban.

L'adresse de départ du programme de dessin de l'image est 5644, celle du programme de dessin des traces: 0200

Grand ruban B (avec filtrage)

Ce programme exécute la version avec filtre de l'algorithme. Les étapes 2 et 4 sont présentes. Les liaisons entre les sept étapes sont incluses. Les programmes de dessin ne sont pas dans le programme et doivent être lus pour être utilisés. L'un et l'autre commencent à 0200 et occupent la place réservée à la liste de points mis de côté par le filtre. Un ruban spécial, du

programme de dessin des traces sans page 0 a été perforé (en utilisant "ODT") en code binaire afin d'éviter de détruire le nombre de traces trouvées, NTRF (mot 0045) en lisant le programme de dessin.

Ordinateurs PDP 8, PDP 8/I, PDP 8/L

Ces mini-ordinateurs sont très semblables au PDP 8/E. Toutefois, ils ne possèdent normalement pas de multiplicateur quotient (MQ). Les instructions qui y font référence doivent donc être supprimées.*

On peut toutefois obtenir un programme équivalent à celui décrit dans ce travail, (quoique le temps d'exécution sera plus long) en définissant trois mots de mémoire, un "MQ" et deux autres "TRAVAIL" et "MACHIN". On leur attribue trois registres inutilisés de la page 0, les mots 0005, 0006, 0007.

```
* 0005
  MQ,0
  TRAVAIL,0
  MACHIN,0
```

Alors, les instructions qui utilisent le multiplicateur-quotient peuvent utiliser le mot MQ.

Les commandements à l'assembleur qui définissent ces opérations doivent être changés.

* Si l'ordinateur est muni du module "Extended Arithmetic Element", ce problème ne se pose pas.

Les définitions actuelles:

MQL = 7421 /AC $\rightarrow$ MQ, 0 $\rightarrow$ AC

MQA = 7501 /OU logique entre AC et
/MQ; résultat dans MQ

CAM = 7621 /Vide AC et MQ

SWP = 7521 /AC $\rightarrow$ MQ: échange

ACL = 7701 /MQ $\rightarrow$ AC

doivent être changées en

MQL = 3006 /DCA MQ: AC $\rightarrow$ MQ, 0 $\rightarrow$ AC

MQA = 4154 /JMS OULOGIQUE: va à
/une sous-routine qui effectue le OU logique

CAM = 4164 /JMS VIDE: va à une sous-routine qui vide
/AC et MQ

SWP = 4170 /JMS ECHANGE: Va à une sous-routine qui
/échange AC et MQ.

ACL = 1006 /TAD MQ: AC - MQ $\rightarrow$ AC
/Cette traduction n'est valide que si AC a été
/vidé avant, ce qui est habituellement le cas.
/Sinon, insérer une CLA avant ACL dans le
/programme.

* 0154 /Les mots 0154-0177 ne sont pas utilisés par le
/programme et sont donc disponibles pour ces
/sous-routines

OULOGIQUE,0 /Sous-routine pour faire un OU logique entre AC
/et MQ. On utilise la règle de DE Morgan.

$$/AC \vee MQ = \overline{\overline{AC} \wedge \overline{MQ}}$$

CMA / $\overline{AC} \rightarrow$ AC

DCA TRAVAIL /TRAVAIL = $\overline{AC}$

TAD MQ

CMA /Complémente MQ et mets le dans AC

AND TRAVAIL /ET logique avec TRAVAIL ($\overline{AC}$)

CMA /Complémentaire pour obtenir le résultat final

JMP I OULOGIQUE /Retour

* 0164

VIDE,0 /Sous-routine qui vide AC et MQ

CLA /0 → AC

DCA MQ /AC (=0) → MQ

JMP I VIDE /Retour

* 0170

ECHANGE,0 /(Sous-routine qui échange AC et MQ.

DCA TRAVAIL /AC → TRAVAIL

TAD MQ

DCA MACHIN /MQ → MACHIN

TAD TRAVAIL

DCA MQ /TRAVAIL → MQ

TAD MACHIN /MACHIN → AC

JMP I ECHANGE /Retour

Le cycle des PDP 8, PDP 8/I et PDP 8/L étant plus lent que celui du PDP 8/E, il faudra être très prudent en adoptant notre méthode de transmission des données et de suppression d'une ligne sur deux.

Le reste du programme, moyennant les modifications mentionnées ci-dessus, fonctionnera sans aucune difficulté, quoique plus lentement.