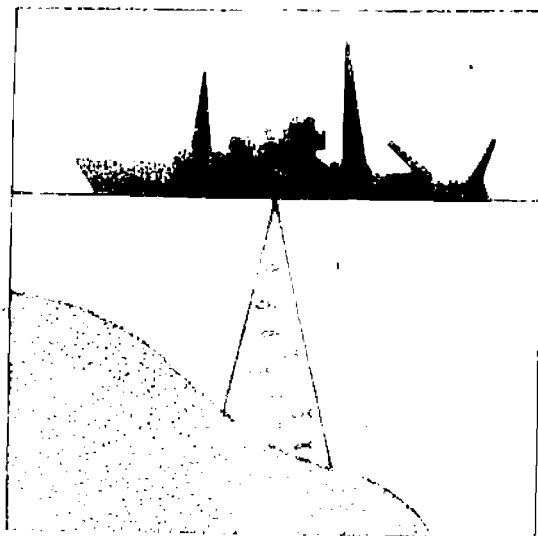


REPUBLIQUE DE COTE D'IVOIRE

CENTRE DE RECHERCHES
OCEANOGRAPHIQUES



RECUEIL DE NOTES TECHNIQUES
ET DE PROGRAMMES DE ROUTINE,
ELABORES SUR UNE CHAINE
D'ACQUISITION DE DONNEES

par

MORLIERE A., CITEAU J., NOEL J.,
CHUCHLA R. et THOMAS Ph.

MINISTERE DE LA
RECHERCHE SCIENTIFIQUE

ARCHIVES SCIENTIFIQUES

VOL. 2, N° 3, Août 1977



ABIDJAN BP V18

RECUEIL DE NOTES TECHNIQUES ET DE PROGRAMMES
DE ROUTINE ELABORES SUR UNE CHAINE D'ACQUISITION
DE DONNEES

par

MORLIERE A., CITEAU J., NOEL J.,
CHUCHLA R. et THOMAS Ph. *

Ce recueil rassemble les différentes notes techniques publiées depuis Mars 1971, relatives à la liaison Sonde - Ordinateur, au fonctionnement général du système informatique mis en place à cette époque à bord du CAPRICORNE, et remis à jour en fonction des améliorations apportées depuis.

Le sommaire se présente de la manière suivante :

Liaison Sonde-Ordinateur (Mars 1971)	pages A1 à A 9
Liaison Sonde-Ordinateur (Mai 1977)	pages B1 à B 4
Comment lever une station à l'ordinateur.....	pages C1 à C 4
Introduction aux opérations élémentaires.....	pages D1 à D15
Initiation au calculateur installé sur le CAPRICORNE	pages E1 à E20
Dépouillement automatique des mesures faites à l'aide de courantomètres AANDERAA	pages F1 à F14
Listings de programmes	pages G1 à G196

* Océanographes de l'ORSTOM au CRO, B. P. V 18 ABIDJAN,
(Côte d'Ivoire)

REPUBLIQUE DE COTE D'IVOIRE
MINISTERE DE LA PRODUCTION ANIMALE

OFFICE DE LA RECHERCHE
SCIENTIFIQUE & TECHNIQUE OBTRE-MER

CENTRE DE RECHERCHES OCEANOGRAPHIQUES
B.P. V 18 - ABIDJAN

NOTE TECHNIQUE
LIAISON SONDE - ORDINATEUR

par

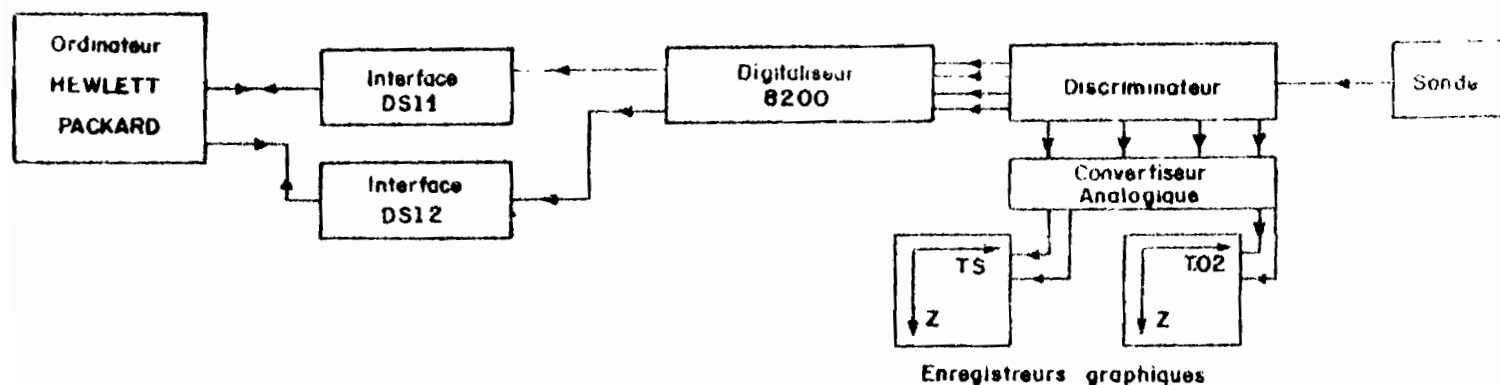
Alain MORLIERE
Océanographe physicien

La Liaison entre une sonde STD Bisset Berman et un ordinateur 2115 HEWLETT PACKARD a été réalisée sur le navire Océanographique Capricorne. Elle permet l'acquisition et le contrôle par ordinateur des données fournies par la sonde.

La sonde Bisset Berman est équipée de quatre capteurs (Température, Salinité, Oxygène, Profondeur), les grandeurs mesurées par ces capteurs sont transformées en fréquence par des oscillateurs, ces fréquences sont mélangées et envoyées vers la surface par un câble conducteur.

En surface les fréquences sont filtrées, séparées; envoyées sur un convertisseur analogique, elles fournissent un enregistrement graphique analogique continu; envoyées sur des fréquencemètres digitaliseurs elles permettent d'obtenir des valeurs digitales discrètes. Les quatre fréquencemètres sont regroupés dans le digitaliseur "8200" Bisset Berman.

Le digitaliseur "8200" est relié à l'ordinateur par l'intermédiaire de deux cartes interfaces (DSI) qui réalisent ainsi la liaison.



SCHEMA DU SYSTEME

I. - DIGITALISEUR "8200"a) - Principe de fonctionnement

Le "8200" contient quatre compteurs de fréquences chacun étant affecté à l'une des grandeurs mesurées par la sonde. Chaque compteur fournit une lecture décimale directe par affichage sur tube Nixi.

Il existe également une sortie en décimal codé binaire, elle est utilisée pour la liaison avec l'ordinateur.

Les fréquences fournies par la sonde sont des fréquences basses, elles doivent être multipliées pour obtenir des temps de comptage faible. Après multiplication, les fréquences sont transformées en impulsions qui sont comptées. Les quatre comptages commencent en même temps mais ils sont de durées différentes. Lorsque le plus long est terminé les quatre résultats sont affichés. Simultanément les résultats sont codés (décimal codé binaire) et envoyés sur des fiches de sortie où est également disponible un certain nombre de signaux de commande.

Chaque digitaliseur est équipé d'un dispositif permettant de multiplier la fréquence mesurée par une constante (a) et de lui ajouter une constante (b). La relation fréquence-grandeur mesurée pour les quatre paramètres est de la forme:

$$G = a f + b$$

Si bien que par action sur a (Gate Time) et sur b (Zéro Offset) on peut afficher directement la valeur de la grandeur mesurée.

b) - Signaux de sortie

Les grandeurs mesurées sont disponibles sur quatre fiches (J6 à J9 sous forme décimale codée binaire (BCD).

Les quatre séries de signaux ne sont pas disponibles simultanément mais arrivent par couple, l'arrivée de chaque couple étant précédée d'une impulsion de commande (PRINT COMMAND).

Un premier PRINT COMMAND précède l'arrivée de S et T sur J6 et J7 (également sur J8 et J9: J8 est monté en parallèle sur J6 de même que J9 sur J7); 50ms plus tard un deuxième PRINT COMMAND précède l'arrivée de O2 et P sur J6 et J7.

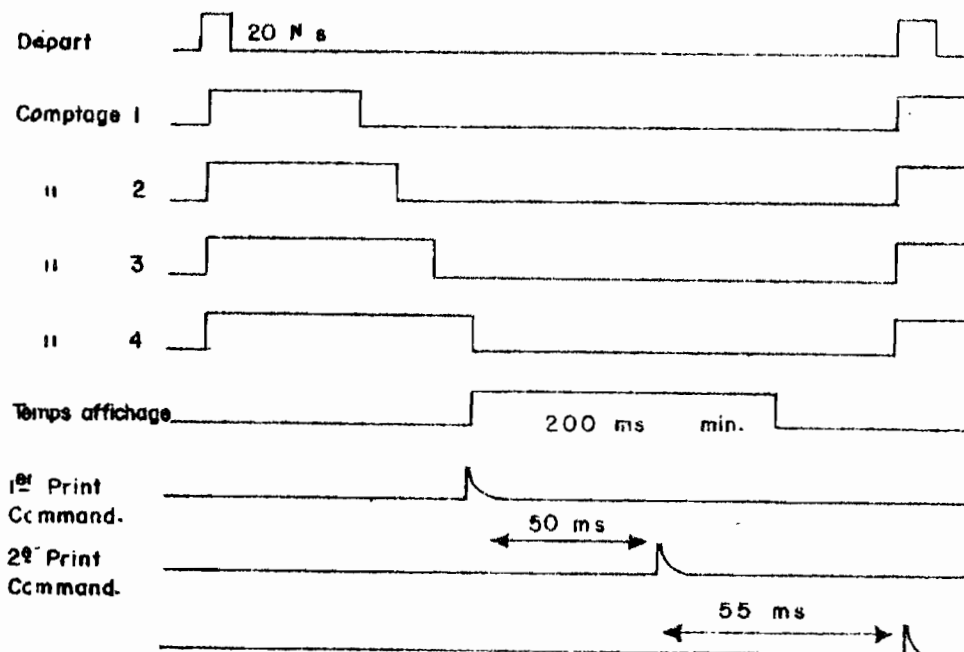


DIAGRAMME CHRONOLOGIQUE

II. - CARTE INTERFACE DSI

La carte DSI (Data Source Interface) permet le transfert de 32 bits vers un ordinateur HEWLETT PACKARD à partir d'une source de données digitalisées capable de fournir certains signaux de commande.

Elle n'a pas de possibilité de stockage, la source doit donc garder les données disponibles en attendant que la carte soit prête et ait fini le transfert.

Quand l'ordinateur a besoin des données de la source, il passe le contrôle à la carte DSI. La carte envoie vers la source un signal (HOLD) lui demandant de commencer la mesure.

Lorsque la mesure est terminée la source envoie vers la carte un signal (RECORD COMMAND) annonçant la disponibilité des données sous forme digitale. La carte interroge l'ordinateur pour savoir s'il est prêt à recevoir les données. Lorsqu'il est prêt, les circuits de transfert de l'interface sont débloqués. L'ordinateur reprend le contrôle et absorbe les données.

Les 32 bits sont dirigés vers les registres A et B de l'ordinateur à raison de 16 bits par accumulateur.

III. - FONCTIONNEMENT DE LA LIAISON SONDE ORDINATEUR REALISEE

a) - Généralités

La carte DSI commande le déclenchement de la mesure, elle pilote donc l'appareil de mesure. Or le digitaliseur "8200" a un fonctionnement autonome c'est à dire qu'il effectue ses mesures à intervalles réguliers sans qu'on ait besoin de lui en donner l'ordre.

La possibilité de pilotage du digitaliseur par la carte, donc par l'ordinateur est perdue.

Dans notre utilisation le "8200" va envoyer les résultats sur la carte à intervalles réguliers et l'ordinateur viendra les prendre quand il en aura besoin: ou plus exactement, étant donné sa plus grande rapidité, il attendra que les résultats soient disponibles.

Dans un premier temps la température et la salinité sont disponibles sur J6 et J7; ensuite l'oxygène et la pression. Ce sont des nombres de quatre chiffres décimaux qui nécessitent donc quatre bits de codage BCD soit 16 bits par donnée, sauf pour l'oxygène qui n'a que trois chiffres donc 12 bits.

En plus des digits, sont disponibles: le point décimal, le signe de la température et une lettre représentant le résultat affiché, soit 28 bits par donnée.

.../...

CODAGE DES SIGNAUX A LA SORTIE DU "8200"

	1	2	4	8
0	0	0	0	0
1	1	0	0	0
2	0	1	0	0
3	1	1	0	0
4	0	0	1	0
5	1	0	1	0
6	0	1	1	0
7	1	1	1	0
8	0	0	0	1
9	1	0	0	1
.	1	1	1	1
+	0	1	0	1
-	1	1	0	1

Etat "1" = terre

Etat "0" = +5.IV

Il est à noter une inversion de logique entre le "8200" et l'ordinateur en effet; pour l'ordinateur:

Etat "0" = 0V

Etat "1" = +5V

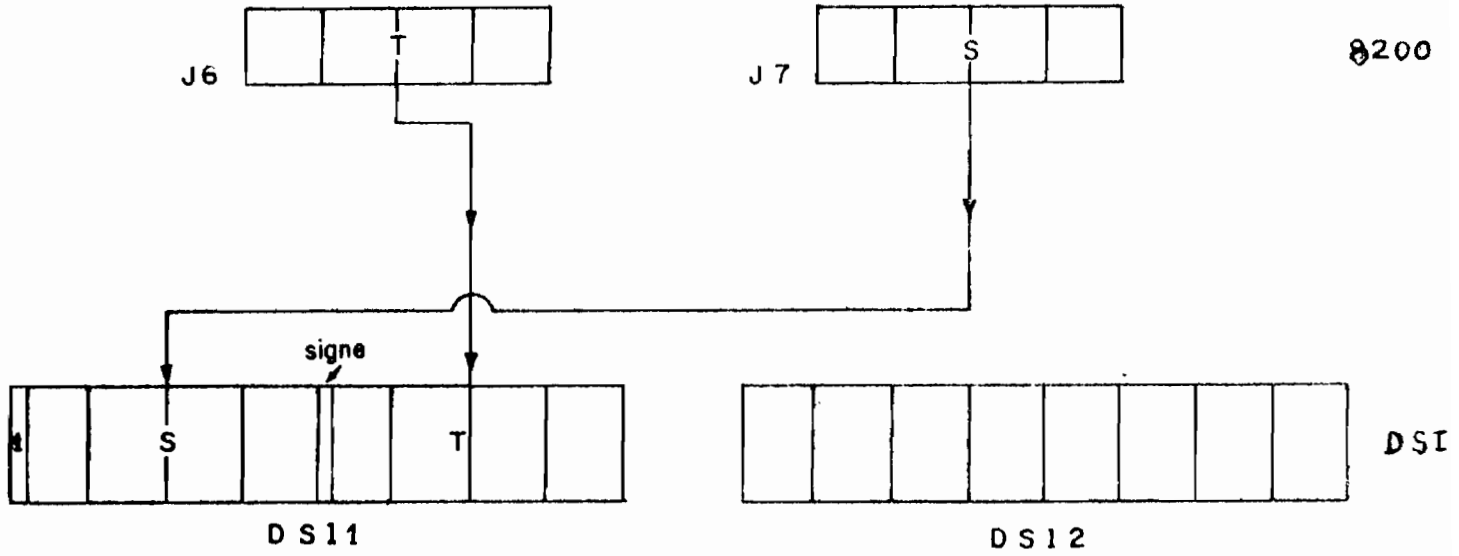
donc les valeurs des bits lus par l'ordinateur seront inverses de celles envoyées par le "8200" d'où la nécessité d'une conversion et des problèmes au niveau des raisonnements.

b) - Signaux utilisés

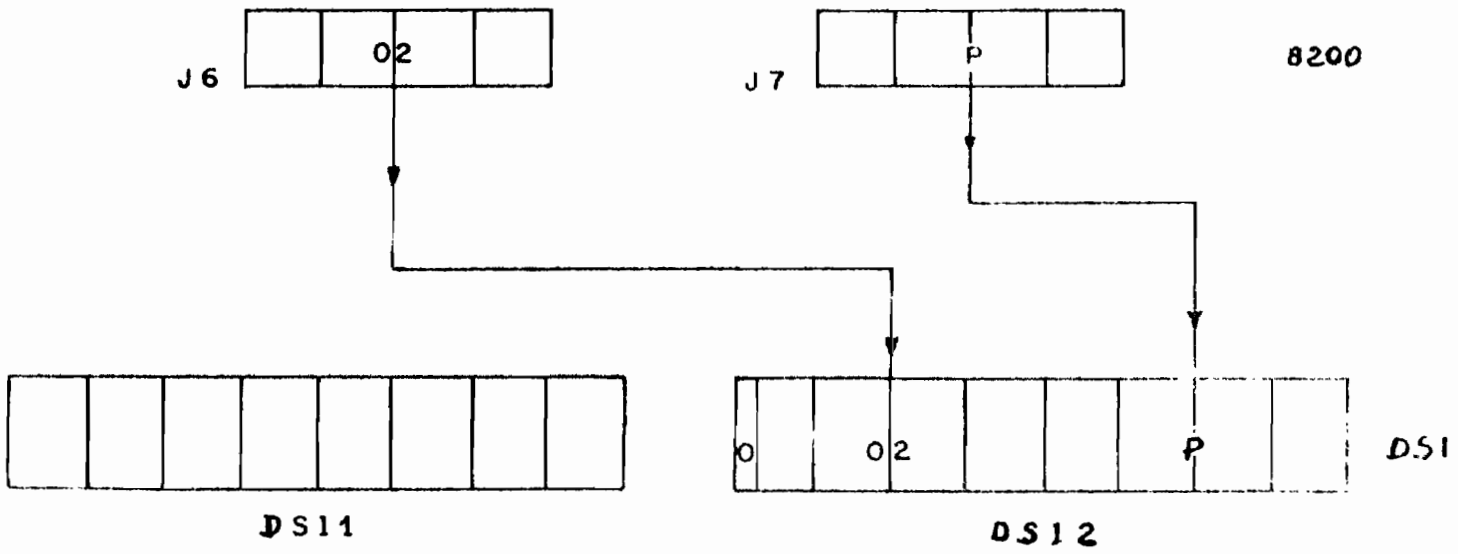
Nous ne transmettons que les bits correspondant à des chiffres soit 16 bits par donnée.

Le bit le plus élevé de la température qui dans la gamme des mesures de la sonde n'est jamais utilisé est remplacé par un bit de signe (borne 11 sur J6), la température pouvant être négative.

1^{er} Print command.



2^e Print command.



SCHEMA SORTIE

SIGNAUX UTILISES A LA SORTIE DU "8200"

Fiche J6	Poids	1 ^o Print Command	2 ^o Print Command
1	1		
2	2	Dernier Digit T	Dernier Digit 02
26	4		
27	8		
3	1		
4	2	2 ^o Digit T	Pas utilisé
28	4		
29	8		
5	1		
6	2	Pas utilisé	2 ^o Digit 02
30	4		
31	8		
7	1		
8	2	3 ^o Digit T	3 ^o Digit 02
32	4		
33	8		
9	1		
10	2	4 ^o Digit T	Pas utilisé
34	4		
11	0 ou 1	Signe de T 0= T POS. 1= T NEG.	Pas utilisé
23		Print Command	Print Command
50		Terre	Terre

Échelle J7	Poids	1 ^{er} Print Command	2 ^e Print Command
1	1		
2	2	Dernier Digit S	Dernier Digit P
26	4		
27	8		
3	1		
4	2	2 ^e Digit S	2 ^e Digit P
28	4		
29	8		
5	1		
6	2	Pas utilisé	3 ^e Digit P
30	4		
31	8		
7	1		
8	2	3 ^e Digit S	4 ^e Digit P
32	4		
33	8		
9	1		
10	2	4 ^e Digit S	Pas utilisé
34	4		
11		0 Bit de "lettre" c'est à dire 5.IV sur "8200"	1 Bit de "lettre"

Le bit le plus élevé de la salinité ainsi que celui de l'oxygène qui ne sont jamais utilisés sont remplacés par un bit de "lettre" (borne II sur J7) qui permet de reconnaître le couple de données actuellement transmis; si ce bit est "1" sur J6 nous sommes en présence du couple TS sinon c'est le couple O2-P.

o) - Cablage de la liaison

8200 J6 Prise 1	DSI 1 T	DSI 2 O2	8200 J7 Prise 2	DSI 1 S	DSI 2 P
1	4	5	1	5	4
2	B	C	2	C	B
26	J	K	26	K	J
27	L	M	27	M	L
3	T		3	U	T
4	V		4	W	V
28	6		28	7	6
29	8		29	9	8
5		V	5		2
6		W	6		D
30		7	30		F
31		9	31		N
7	2	3	7	3	R
8	D	E	8	E	X
32	F	H	32	H	Z
33	N	P	33	P	10
9	R		9	S	
10	X		10	Y	
34	Z		34	AA	
11	10		11	11	11
23	11	16			
50	24, BB	24, BB			

IV. - SOFTWARE DE CONTROLE DE LA LIAISON

La partie software est représentée par un sous programme en langage ASSEMBLER qui peut être appelé dans un programme FORTRAN.

APPEL FORTRAN: CALL STDC (JP, JT, JS, JO, JCLE).

Cet appel déclenche la lecture des résultats disponibles sur le "8200"; les valeurs sont affectées aux variables:

JP pour la profondeur
 JT pour la température
 JS pour la salinité
 JO pour la teneur en oxygène

JCLE est un argument qui permet de contrôler manuellement l'acquisition des données.

Si JCLE = 0 (toutes les clefs sont baissées), la liaison est prête mais la transmission ne se fait pas: le système est en attente; pour JCLE = 1 (clef "0" levée) la liaison est assurée et les résultats transférés à l'ordinateur.

On voit donc qu'en levant et abaissant la clef "0" (toutes les autres restant baissées), on peut acquérir "manuellement" les données.

D'autre part la transmission par JCLE du contenu du registre de clef vers le programme principal offre la possibilité d'autres tests de contrôle au choix de l'utilisateur.

Le programme STDC effectue la lecture des signaux fournis par le "8200", puis l'inversion des bits pour palier à l'opposition des logiques entre "8200" et ordinateur.

Il reconnaît, à partir du bit de "lettre", le couple de données envoyées (TS ou O2-P); si le premier couple est O2-P (2^e PRINT COMMAND), il l'abandonne et retourne en lecture pour lire dans l'ordre TS puis O2-P (c'est à dire 1^e PRINT COMMAND puis 2^e PRINT) ceci pour que les valeurs lues ensemble soient bien celles mesurées en même temps.

STDC effectue ensuite la conversion BCD - binaire.

Remarques: - STDC élimine toutes les valeurs correspondant à des immersions supérieures à 1500m (limite de la sonde), ceci dans le but d'éliminer les valeurs parasites qui apparaissent au moment de la mise à l'eau de la sonde.

Le software de la liaison comprend également un programme TESTD de test du bon fonctionnement de la liaison.

CENTRE DE RECHERCHES OCEANOGRAPHIQUES
B.P. V 18 - ABIDJAN - (C.I.)

Note Technique

NOUVELLE LIAISON SONDE - ORDINATEUR

par

A. MORLIERE et Ph. THOMAS

Mai 1977

INTRODUCTION

La liaison entre une sonde Bisset Berman et un ordinateur HEWLETT-PACKARD, précédemment réalisée sur le "Capricorne" a été modifiée dans le but d'une simplification des éléments physiques de liaison.

Pour cela une modification à l'intérieur du 8200 a été faite. Tous les signaux inutiles pour notre utilisation ont été éliminés, si bien que le multiplexage n'utilise que deux cartes au lieu de quatre et que la sortie des signaux se fait sur un connecteur au lieu de deux. Cela a permis de n'utiliser qu'une seule carte interface DSI dans l'ordinateur. La software d'acquisition a été modifiée en conséquence.

MODIFICATIONS DU 8200

I.- BCD SWITCH ASSEMBLY

Connexion de la sortie codage avec le multiplaxage

Connecteur JIIA-JIID	BIT	S	T	P	O ₂
A14 A 9 A11 A15	1 2 4 8	A12 (B) (C) (D) (E)	A10 (B) (C) (D) (E)	A12 (2) (3) (4) (5)	A10 (2) (3) (4) (5)
B 4 B 1 B 2 B 5	1 2 4 8	A12 (F) (H) (J) (K)	A10 (F) (H) (J) (K)	A12 (6) (7) (8) (9)	A10 (6) (7) (8) (9)
B11 B 6 B 8 B12	1 2 4 8	A12 (L) (M) (N) (P)	A10 (L) (M) (N) (P)	A12 (10) (11) (12) (13)	A10 (10) (11) (12) (13)
C 1 B13 B15 C 2	1 2 4 8	A12 (R) (J) (T) (U)	A10 (R) (S) (T) (U)	A12 (14) (15) (16) (17)	A10 (14) (15) (16) (17)

II.- LIAISON MULTIPLEXAGE AVEC CONNECTEUR DE SORTIE

Connecteur J1	Connecteur M.	Bit Poids	1 ^{er} Print Commande	2 ^e Print Commande
16 15 14 13	A10 - 1 2 A B	1 2 4 8 10^0	1 ^o digit T	1 ^o digit O ₂
12 11 10 9	A10 - 3 4 C D	1 2 4 8 10^1	2 ^o digit T	2 ^o digit O ₂
8 7 6 5	A10 - 5 6 E F	1 2 4 8 10^2	3 ^o digit T	3 ^o digit O ₂
4 3 2	A10 - 7 8 H	1 2 4 10^3	4 ^o digit T	4 ^o digit

Connecteur J1	Connecteur M.	Bit Poids	1 ^{er} Print Commande	2 ^e Print Commande
41 40 39 38	A12 - 1 2 A B	1 2 4 8 10^0	1 ^o digit S	1 ^o digit P
37 36 35 34	A12 - 3 4 C D	1 2 4 8 10^1	2 ^o digit S	2 ^o digit P
33 32 31 30	A12 - 5 6 E F	1 2 4 8 10^2	3 ^o digit S	3 ^o digit P
29 28 27	A12 - 7 8 H	1 2 4 10^3	4 ^o digit S	4 ^o digit P

Le signal Gx représente le premier Transfert de données, il sert de bit de reconnaissance pour le premier Print Commande.

CABLE DE LIAISON

Connecteur J1	Connecteur Ord.
16 _____	4
15 _____	B
14 _____	J
13 _____	L
12 _____	T
11 _____	V
10 _____	6
9 _____	8
8 _____	2
7 _____	D
6 _____	F
5 _____	W
4 _____	R
3 _____	X
2 _____	2
41 _____	5
40 _____	C
39 _____	K
38 _____	M
37 _____	V
36 _____	W
35 _____	7
34 _____	9
33 _____	3
32 _____	E
31 _____	H
30 _____	P
29 _____	S
28 _____	Y
27 _____	AA
20 Print Command	16
21 Masse	24, BB
22 Gx	11, 10

MODIFICATION DU SOFTWARE DE CONTROLE DE LA LIAISON

La liaison fonctionant sur une seule carte DSI au lieu de deux précédemment, le programme STDC a dû être modifié. Cependant l'utilisation du sous programme assembleur STDC reste la même.

La carte DSI utilisée doit être placée dans le canal 13.

NOTICE D'UTILISATION DE L'ORDINATEUR DU LA PRICORNE
POUR LEVER UNE STATION A L'ENCLUSTRE

Mise en route

- ordinateur sur 'on'
- placer la bande magnetique de la campagne sur son
lecteur :
 - START
 - LOAD POINT
 - (AUTO)- ON LINE sur 7970 b
- télétype sur 'on line'
- perforateur rapide TOUER ON - DC ON
(verifier que la reserve de papier est suffisante)

Initialisation

- lever les clefs 12 11 10 (16000g)
LOAD ADDRESS
- baisser les clefs 12 11 10
- PRESET
- RUN

L'ordinateur doit alors repondre par la teletype : NEXT ?

Voir le déroulement de la procedure.

Note : A compter de Mai 1977, le seul programme d'acquisition est SONDC adapté
à la nouvelle liaison "8200 - Ordinateur"

DEROULEMENT NORMAL de la PROCEDURE

*NEXT?	1	
:PRDG,SONDE	2	Ecrit par l'operateur sur la teletype
** SONDE **	3	
CONTROLE LIAISON		Lever la clef 0
679 1543 3473 65	4	A vérifier à l'affichage du 8200
DATE,HEURE,CRO,STA?	5	Baisser la clef 0
16,1,73,1130,7303,1	6	Ecrit par l'operateur sur la teletype
PAUSE	7	Verifier que toutes les clefs sont en bas. Faire RUN. Quand la sonde est prête et que l'immersion 0m s'affiche au display du 8200 la levee peut demarrer en levant la clef 0.
- FIN DE STATION -		
		Pour arrêter la levee de la station il faut :
		-Pour inscrire sur la Bande magnetique (Sinon voir *) Lever la clef 7.
COPIE	8	
PAUSE	9	Baisser la clef 7. Lever la clef 1. Faire RUN. Pendant la perforation controler qu'il n'y a pas de bourrage du perforateur.
COPIE	10	
PAUSE	11	Acquisition terminee. Couper le ruban perfore de la station, inscrire le numero de campagne et celui de la station sur le ruban obtenu.

ARRET DE L'ORDINATEUR

- mettre BM sur LOCAL, (puis BRAKE sur 2020 B)
- arrêter la télécrite et le lecteur optique
- arrêter le perforateur (POWER OFF)
- ordinateur sur OFF
- couper le disjoncteur

* Si l'inscription sur bande magnetique n'est pas desiree, aller en 9.

Problemes divers durant l'acquisition d'une station.

Phase	Problèmes	Remedes
1	Pas de NEXT?	Verifier que l'unité magnétique est sur auto.(ON LINE sur 7970b)
	TP ERR	Faire plusieurs essais. S'il y a recidive, le système est detruit par suite d'une fausse manoeuvre. Charger le programme SONDE à l'aide du BBL (procedure B).
	CS ERR NEXT?	Retaper correctement :PROG,SONDE Si vous êtes vraiment sûr d'avoir tape correctement c'est que le BOOTSTRAP est detruit il faut le charger à nouveau à l'aide du BBL (procedureA). Ou vous n'avez pas pris la bonne bande magnétique.
4	A l'issue du controle liaison rien n'est envoyé par l'ordinateur.	Verifier que le P200 est bien branche. S'il l'est et que les valeurs du display n'arrivent pas jusqu'à l'ordinateur, c'est que la liaison est defectueuse il faut la réparer. Verifier que le P200 n'est pas "bloqué", pour debloquer faire RESET.
	A l'issue du controle liaison les valeurs reçues sont fausses.	La liaison sonde-ordinateur est defectueuse.
7	Ennuis divers pendant l'acquisition.	Ne pas hesiter à recommencer la station.

PROCEDURE A (Remise en place du BOOTSTRAP)

Placer le ruban perforé marqué : BOOTSTRAP dans le lecteur optique

Lever les clefs 12 11 10 9 8 7 6 (17700g)

Load address

Baisser les clefs.

PRESET

LOADLR ENABLED

RUN

Le programme se charge.

LOADER DISABLED

Vérifier que la bande magnétique est prête (**START, LOAD PRIMI, AUTO**)

Lever la clef 6 (100g)

LOAD ADDRESS

Baisser la clef 6

PRESET

RUN

La télétype doit écrire **NEXT?**

PROCEDURE B (Chargement d'un programme absolu)

Placer le ruban perforé marqué **SONDE** dans le lecteur optique.

READ

Lever les clefs 12 11 10 9 8 7 6 (17700g)

LOAD ADDRESS

Baisser les clefs

LOADER ENABLED

PRESET

RUN

Le programme se charge

LOADER DISABLED

Lever la clef 1 (4g)

Load address

Baisser la clef 1 Vérifier que la bande magnétique est prête.

PRESET

RUN

Le programme **SONDE** est lancé.

INTRODUCTION AUX OPERATIONS ELEMENTAIRES

Description et mode d'emploi de l'ordinateur Hewlett-Packard

2115 A

1 - INTRODUCTION

Le but de ce fascicule est de familiariser l'utilisateur avec les commandes du calculateur 2115 A. En particulier, on trouvera ici la description des différentes instructions, la manière de les composer en binaire, de les ranger en mémoire à l'aide des clés du pupitre de commande et de les faire exécuter.

1.1 - Caractéristiques de la machine

La machine travaille sur des mots de 16 bits ; le nombre de registres disponibles est de 8 (y compris les deux accumulateurs) et son arithmétique est celle du complément à 2. Sa mémoire centrale est à tores magnétiques et peut contenir 8192 mots de 16 bits répartis en 8 pages de 1024 mots chacune. Les instructions ont une longueur d'un mot et elles sont à une seule adresse. Leur durée d'exécution est de un ou deux cycles mémoire (1 cycle = 2.0 microseconde) plus un cycle pour chaque temps d'adressage indirect.

On dispose de 14 instructions faisant intervenir la mémoire, de 43 instructions concernant un registre seul et de 13 instructions d'entrée/sortie, ce qui donne un total de 70 instructions. Pour les entrées/sorties, on dispose de 8 niveaux de priorité.

1.2 - Description des registres

A Registre A (premier accumulateur)

C'est un registre à 16 bits utilisé pour contenir des opérandes et les résultats d'opérations arithmétiques, logiques ou d'entrée-sortie. Les instructions se référant à la mémoire 0 en page 0 adressent explicitement le registre A. Toutes les instructions peuvent faire implicitement appel à A.

B Registre B (deuxième accumulateur)

Mêmes caractéristiques que A mais son adresse est 1 en page zéro et les instructions AND, IOR et XOR ne peuvent pas y faire appel.

E Extension et retenue

Ce registre à 1 bit sert à contenir une retenue arithmétique provenant de A ou de B. Il sert également de lien entre A et B grâce aux instructions de décalage circulaire et peut servir de clé dans un programme car il est accessible au programmeur.

V=OVERFLOW Bit de débordement

Ce registre à 1 bit sert à recueillir un débordement arithmétique éventuel provenant de A ou de B ; il peut servir également de clé dans un programme car il est accessible au programmeur.

M Registre d'adresse mémoire

Ce registre à 16 bits contient l'adresse de l'opérande ainsi que le bit d'adressage indirect éventuel.

T Registre d'entrée-sortie mémoire

Ce registre à 16 bits sert de tampon pour l'information lue ou rangée en mémoire. Ce registre n'est pas accessible au programmeur.

S Clés du pupitre

Ce registre comprend 16 clés servant à introduire manuellement l'information par la console ou par programme.

P Compteur ordinal

Ce registre à 15 bits contient l'adresse de l'instruction en cours d'exécution.

1.3 - Description de la mémoire

La mémoire centrale comprend 8192 mots de 16 bits décomposés en pages de 1024 mots. Son temps de cycle est de 2 microsecondes.

Une adresse a la structure suivante :

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
adresse de la page						adresse de mot dans la page									

La partie page spécifiée dans le registre P est appelée page courante. Toutes les instructions, quelle que soit leur place en mémoire, ne peuvent adresser directement que la page courante et la page zéro. Elles peuvent adresser toutes les pages de façon indirecte (ceci à plusieurs niveaux si on le désire).

1.4 - Formats des mots

Dans ce qui suit, nous adopterons les notations suivantes :

Z représente la page zéro
 C représente la page courante
 D représente un adressage direct
 I représente un adressage indirect

Par exemple : Z/C signifie que si le bit correspondant est à 0, on adresse la page zéro et que s'il est à 1, on adresse la page courante.

Même chose pour D/I.

Les instructions sont divisées en plusieurs groupes, c'est, en général, la partie code opération qui sert à repérer le groupe et la partie micro opération ou code secondaire qui définit l'opération elle-même à l'intérieur du groupe ; on verra plus loin comment composer de telles instructions.

- Instructions faisant appel à la mémoire

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
D/I	code op.				Z/C	adresse opérande									

- Instructions concernant un registre

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
code opération						micro opération									

- Instructions d'entrée/sortie

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
code opération						code secondaire				sélection					

- Adresse

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
D/I	adresse de page					adresse de mot									

- Opérande

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
signe	nombre							entier							

2 - DESCRIPTION DES INSTRUCTIONS

Les instructions se composent d'un seul mot de 16 bits et demandent un ou deux cycles mémoire à l'exécution sauf ISZ qui dure 3,6 microsecondes.

Les instructions concernant les registres peuvent être combinées selon les micro opérations, en plus de 1000 instructions différentes (voir exemples plus loin).

2.1 - Instructions faisant appel à la mémoire

Ces instructions peuvent employer l'adressage direct ou indirect et indiquer une adresse de la page courante ou de la page zéro. Dans ce qui suit, Y est l'adresse de l'opérande. Nous noterons (Y) le contenu du mot d'adresse Y.

Code Mnémonique

	AND	Y	ET logique de (A) avec (Y)	$(A) \leftarrow (A) \wedge (Y)$
Exclusive Or	XOR	Y	OU exclusif de (A) avec (Y)	$(A) \leftarrow (A) \oplus (Y)$
Inclusive Or	IOR	Y	OU inclusif de (A) avec (Y)	$(A) \leftarrow (A) \vee (Y)$
Jump to subroutine	JSB	Y	Branchement vers sous-programme débutant en Y	$\left\{ \begin{array}{l} (Y) \leftarrow (P) + 1 \\ (P) \leftarrow Y + 1 \end{array} \right.$
Jump	JMP	Y	Saut en Y	$(P) \leftarrow Y$
Increment and skip if zero	ISZ	Y	Incrémente (Y) et saute si 0	$(Y) \leftarrow (Y) + 1$ Si (Y) \neq 0 alors $(P) \leftarrow (P) + 1$ Si (Y) = 0 alors $(P) \leftarrow (P) + 2$
Add to A	ADA	Y	Additionner à A le (Y)	$(A) \leftarrow (A) + (Y)$ Si débordement $(V) \leftarrow 1$ Si retenue $(E) \leftarrow 1$
Add to B	ADB	Y	Additionner à B le (Y)	Même chose pour B
Compare to A	CPA	Y	Comparer (Y) à A	Si (A) = (Y) alors $(P) \leftarrow (P) + 1$ Si (A) \neq (Y) alors $(P) \leftarrow (P) + 2$
Compare to B	CPB	Y	Comparer (Y) à B	Même chose pour B
Load A	LDA	Y	Charger A avec (Y)	$(A) \leftarrow (Y)$
Load B	LDB	Y	Charger B avec (Y)	$(B) \leftarrow (Y)$
Store A	STA	Y	Ranger (A) en Y	$(Y) \leftarrow (A)$
Store B	STB	Y	Ranger (B) en Y	$(Y) \leftarrow (B)$

Par contre, on peut écrire :

ERA, CLE, SLA

Dans le cas d'un tel groupement, les instructions seront exécutées dans l'ordre de gauche à droite.

On trouvera plus loin un exemple de constitution en binaire d'une telle instruction.

ALS	A left shift	décalage arithmétique de (A) de 1 bit à gauche $A(0) \leftarrow \emptyset$ $A(15)$ inchangé
ARS	A right shift	décalage arithmétique de (A) de 1 bit à droite $A(14) \leftarrow A(15)$
RAL	Rotate A left	Rotation de (A) de 1 bit à gauche
RAR	Rotate A right	Rotation de (A) de 1 bit à droite
ALR	A left shift, Clear sign	Décalage arithmétique de (A) à gauche d'un bit $A(0) \leftarrow 0$ $A(15) = 0$
ERA	Rotate E and A right	Décalage circulaire de (E) et (A) à droite d'1 bit
ELA	Rotate E and A left	Décalage circulaire de (E) et (A) à gauche d'1 bit
ALF	A left four	Décalage circulaire de (A) de 4 bits à gauche
NOP	No operation	Instruction ineffective ($P \leftarrow (P) + 1$)
CLA	Clear A	$(A) \leftarrow 0$
CMA	Complement A	$(A) \leftarrow (\overline{A})$
CCA	Clear and complement A	$(A) \leftarrow -1$
CLE	Clear Extend	$(E) \leftarrow 0$
CME	Complement Extend	$(E) \leftarrow (\overline{E})$
SEZ	Skip if extend zero	Si $(E) = 0$ alors $(P) \leftarrow (P) + 2$
SSA	Skip if sign of A zero	Si $A(15) = 0$ alors $(P) \leftarrow (P) + 2$
SLA	Skip if lower bit of A zero	Si $A(0) = 0$ alors $(P) \leftarrow (P) + 2$
INA	Increment A	$(A) \leftarrow (A) + 1$ Si débordement $(V) \leftarrow 1$ Si retenue $(E) \leftarrow 1$
SZA	Skip if zero A	Si $(A) = 0$ alors $P \leftarrow P+2$
RSS	Reverse Sense of Skip	Inverse la condition de saut
STO	Set overflow	$(V) \leftarrow 1$
CLO	Clear overflow	$(V) \leftarrow 0$

SOC	H/C Skip if overflow Clear	Si (V) = 0 alors (P) ← (P) + 2 H ⇒ (V) ← (V) ; C ⇒ (V) ← ∅
SOS	H/C Skip if overflow Set	Si (V) = 1 alors (P) ← (P) + 2 H ⇒ (V) ← (V) ; C ⇒ (V) ← ∅

Dans ce groupe d'instructions, on obtient l'instruction équivalente pour le registre B en remplaçant la lettre A par B dans les codes opération.

2.3 - Instructions d'entrée/sortie

La machine dispose de nombreux périphériques ; il faut donc pouvoir préciser à quel périphérique s'adresse l'instruction d'entrée-sortie que l'on utilise. On dispose pour cela du code de sélection du périphérique (SC) appelé encore adresse du périphérique. Ce code de sélection permet d'avoir accès, par des instructions appropriées, à plusieurs éléments nécessaires aux opérations d'entrée-sortie :

- Le bit de contrôle, dont la valeur peut être fixée par programme, sert à initialiser l'opération d'entrée/sortie sur le périphérique quand il est mis sur 1. Dans ce cas, il autorise l'arrivée des interruptions en provenance du périphérique et généralement il permet à l'interface de produire une impulsion qui déclenche le fonctionnement du périphérique.

- Le bit drapeau dont la valeur peut être fixée par programme, sert à indiquer la fin d'une opération d'entrée/sortie ; généralement, on met ce bit à ∅ et c'est la fin d'opération signalée par le périphérique qui le fait passer sur 1.

- Le registre tampon contenu dans l'interface sert à accueillir l'information en provenance du périphérique ou allant vers le périphérique.

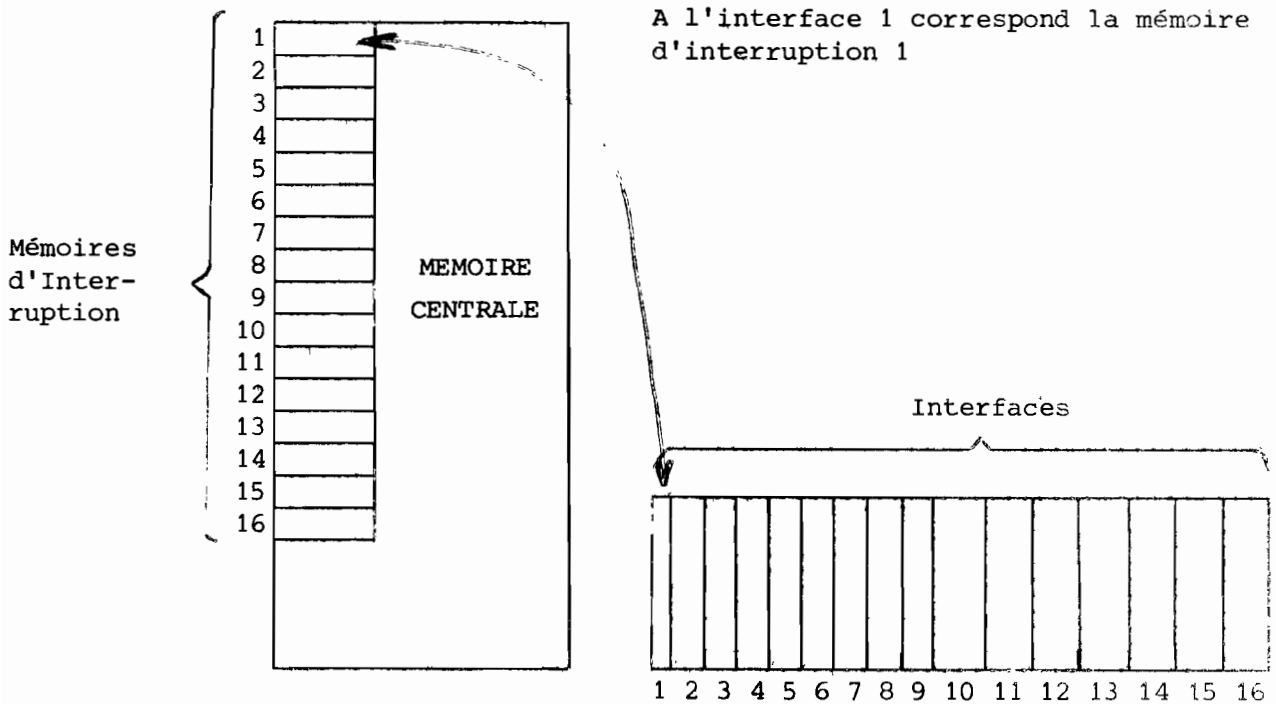
Dans ce qui suit, les lettres H et C dans la zone opérande précisent si après l'exécution de l'instruction correspondante, on veut que le bit drapeau garde la valeur qu'il avait (H = Hold), ou soit à zéro (C = Clear).

	Code Mnémorique		Fonction
HLT	H/C	SC	Arrêt du programme
STF		SC	(drapeau) ← 1
CLF		SC	(drapeau) ← 0
SFC		SC	Si (drapeau) = 0 alors (P) ← (P) + 2
SFS		SC	Si (drapeau) = 1 alors (P) ← (P) + 2
MIA	H/C	SC	(A) ← (A) ∨ (tampon)
LIA	H/C	SC	(A) ← (tampon)
OTA	H/C	SC	(Tampon) ← (A)
STC	H/C	SC	(bit de contrôle) ← 1 active le télétype
CLC	H/C	SC	(bit de contrôle) ← 0

Les mêmes instructions sont également possibles avec le registre B.

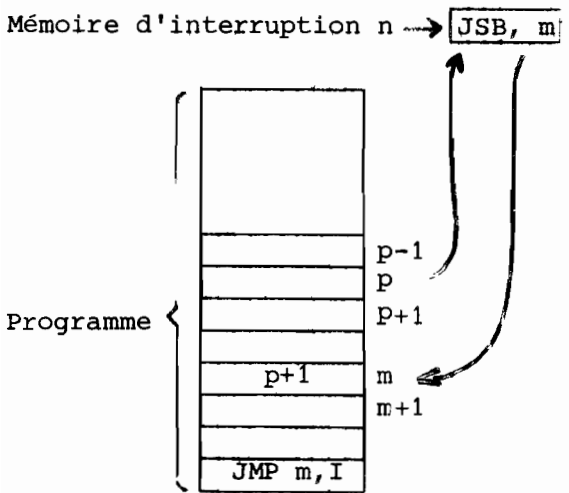
2.4 - Généralités sur les entrées/sorties

Les entrées/sorties se font par l'intermédiaire des canaux de base A/B, permettant soit de transférer un mot de 16 bits du registre A ou B vers un périphérique, soit l'inverse. Ces canaux permettent à l'information d'être échangée au rythme de 70000 mots par seconde entre les registres tampon contenus dans les interfaces et les registres A ou B. Le système de base comporte 8 de ces canaux hiérarchisés selon 8 niveaux de priorité correspondant chacun à une des huit mémoires d'interruption.



A l'interface 1 correspond la mémoire d'interruption 1

Apparition d'une demande d'interruption sur l'interface n



Si la demande apparaît au cours de l'exécution de l'instruction p, à la fin de l'exécution de cette instruction, la machine exécute l'instruction contenue dans la mémoire d'interruption n.

Le compteur ordinal ne change pas de valeur et reste à p+1. Si l'instruction (n) est un JSB vers un sous-programme rangé en m, la valeur du compteur ordinal sera automatiquement rangée en m et le sous-programme sera exécuté à partir de l'adresse m+1 jusqu'à l'instruction JMP m, I qui a pour effet de transférer le contenu de m dans le compteur ordinal et permet de reprendre l'exécution du programme principal en p+1.

Les éléments de l'interface sont :

- Le Registre Tampon. Ce registre sert à contenir momentanément les données circulant entre le périphérique et un des registres A/B/T.

- Le bit de contrôle. Selon qu'il vaut 1 ou 0, il autorise ou n'autorise pas les interruptions sur le niveau de l'interface où il se trouve. De plus, sa mise à 1 génère en général une impulsion qui excite le périphérique et initialise l'opération d'entrée/sortie.

- Le bit drapeau. Ce bit peut être mis à 1 ou 0 par programme (instruction STF ou CLF), soit mis à 1 par le périphérique qui indique ainsi qu'il a terminé une opération. Si le système d'interruption est "en" (grâce à une instruction STF \emptyset) la mise à 1 d'un bit drapeau provoque une demande d'interruption. Cette demande d'interruption ne provoque une interruption que si les bits drapeaux correspondant à des niveaux de priorité supérieurs ont été remis à zéro. Sinon, cette demande est mise en réserve (par le fait que le bit drapeau reste sur 1) et ne provoquera d'interruption que lorsque toutes les demandes d'interruption de priorité supérieure auront été traitées (bits drapeaux remis à zéro).

- Mémoire d'interruption. A chaque canal A/B est affecté par construction un mot de mémoire appelé mémoire d'interruption dont l'adresse est identique au code de sélection du canal. Quand une demande d'interruption est prise en compte, la machine exécute l'instruction contenue dans la mémoire d'interruption correspondante qui doit, par conséquent, contenir une instruction JSB vers un sous-programme traitant l'interruption.

Les mémoires 5 à 77_g (5 étant la priorité la plus élevée) servent de mémoire d'interruption. Les codes de sélection 0 à 4 étant réservés aux applications détaillées ci-dessous :

Code de sélection en OCTAL	Instructions possibles
\emptyset	$\left\{ \begin{array}{l} \text{STF } \emptyset \\ \text{CLF } \emptyset \end{array} \right.$ système d'interruption "en" système d'interruption "hors"
1	$\left\{ \begin{array}{l} \text{LIA } 1 \quad (A) \leftarrow (S) \\ \text{LIB } 1 \quad (B) \leftarrow (S) \\ \text{MIA } 1 \quad (A) \leftarrow (A) \vee (S) \\ \text{MIB } 1 \quad (B) \leftarrow (B) \vee (S) \end{array} \right.$
2,3,4,5,6,7	Réservées au canal d'accès direct en mémoire
10 à 77	Mémoires d'interruption proprement dites

Dans ce qui suit, nous appellerons MI une mémoire d'interruption

si (MI) = JMP Y (P) \leftarrow Y et AR (adresse de retour est détruite)

si (MI) = JSB Y (Y) \leftarrow AR et (P) \leftarrow Y+1

si (Mi) = instructions quelconques, exécution de cette instruction et reprise à (P) = AR

Les interruptions sont généralement prises en compte à la fin du cycle mémoire en cours. Cependant, le système d'interruption est déconnecté au cours de l'exécution de certaines instructions :

1) - Système "hors" jusqu'à la fin de l'exécution plus 1 cycle :

STF, CLF, STC, CLC, JMP et JSB indirects ou non

2) - Instructions exécutées après interruptions mettant le système "hors" jusqu'à la fin d'exécution plus 1 cycle :

JSB et JMP direct ou indirect et toutes les instructions durant 1 cycle ou 2 cycles

3) - Les instructions durant 3 cycles ne mettent le système "hors" que jusqu'à la fin de leur exécution.

3 - MODE D'EMPLOI DE LA MACHINE - PROGRAMMATION

3.1 - Rangement d'un programme en mémoire et exécution

A - Rangement d'un programme :

- 1 . Afficher l'adresse de rangement
- 2 . Presser LOAD ADDRESS
- 3 . Afficher l'instruction
- 4 . Presser LOAD MEMORY
L'incrementation à l'adresse suivante se fait automatiquement
- 5 . Aller en 3

B - Lecture d'un programme :

- 1 . Afficher l'adresse - 2 . Presser LOAD ADDRESS
- 3 . Presser DISPLAY MEMORY
L'adresse s'incrémente automatiquement
- 4 . Aller en 3

C - Exécution d'un programme en mode automatique :

- 1 . Afficher l'adresse d'exécution
- 2 . Presser LOAD ADDRESS
- 3 . Presser PRESET
- 4 . Presser RUN

D - Exécution instruction par instruction

L'ordinateur étant en mode HALT, on exécute une à une les instructions en appuyant sur SINGLE CYCLE.

3.2 - Programmation : Constitution d'une instruction

Pour constituer en binaire une instruction correspondant à un code symbolique donné, on se reportera à la feuille de codage fournie en annexe.

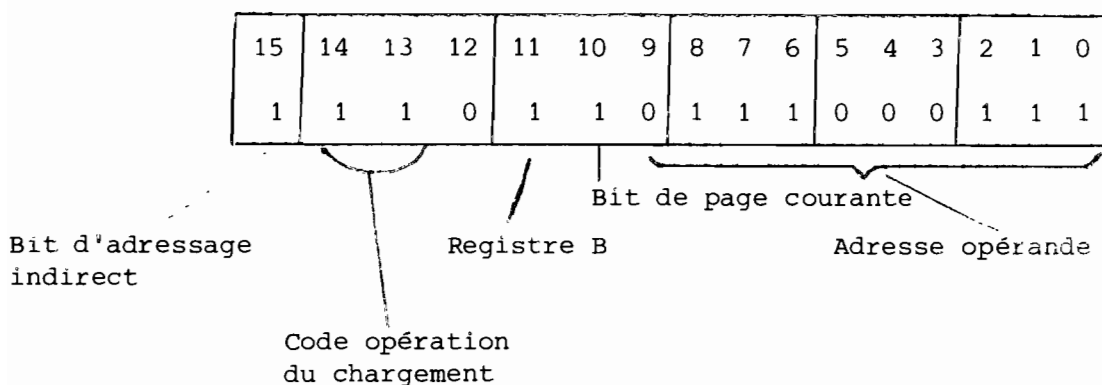
On trouvera ci-dessous des exemples de programme ainsi que des exemples d'instructions.

Exemple 1 Instruction du type "Registre à Mémoire"

Soit A un mot de mémoire, (A) = adresse de l'opérande, A est dans la même page que l'instruction, et B le registre à charger, soit symboliquement :

L D B A,I (I signifie adressage indirect)

Soit en octal 707 l'adresse de A, nous aurons :



soit

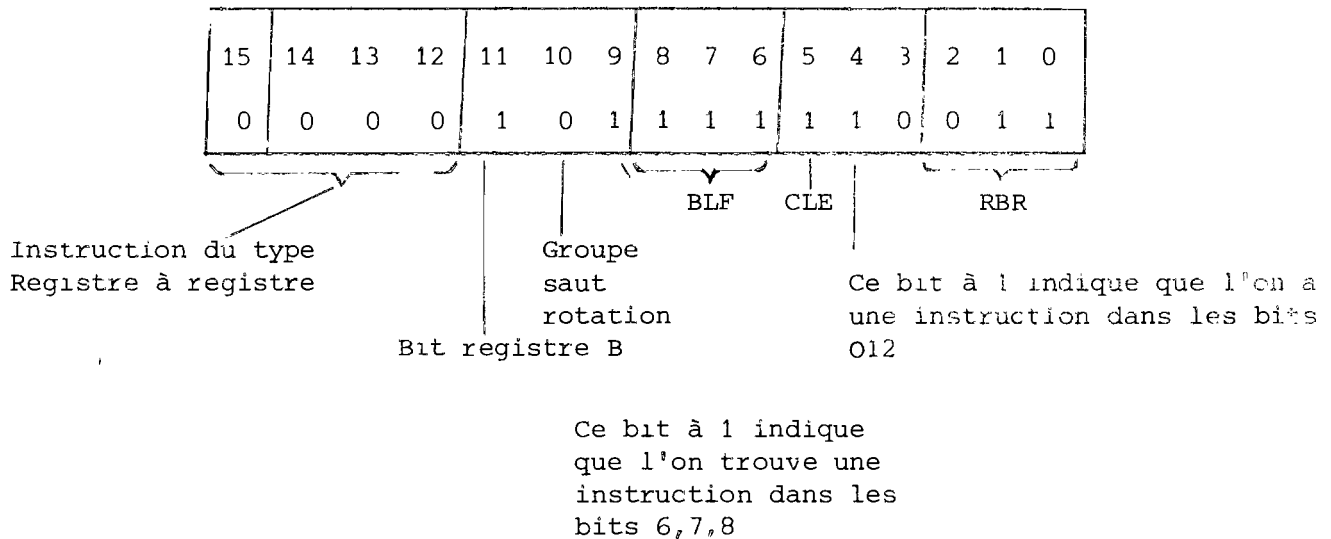
166707₈

l'indice 8 signifiant un nombre écrit en numérotation octale.

Exemple 2

Instruction registre à registre

Soit à décaler le registre B de trois positions à gauche et remettre le registre E à zéro. On aura le code suivant qui correspond au code symbolique BLF, CLE, RBR



soit

005763_gExemple 3

Programme d'entrée/sortie par télétype

Comme cela a été montré plus haut, le télétype est relié à la machine par une carte interface qui permet de commander les fonctions entrée ou sortie du télétype et également de leur envoyer de l'information (ou de la recueillir).

Soit n le numéro du canal A/B sur lequel est branché le télétype.

L'instruction OTA n permet de commander le mode de fonctionnement du télétype selon le contenu de A

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
A	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
A	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0

télétype en mode
"sortie" : écriture

télétype en mode
"entrée" : lecture

Le bit 15 positionné à 1 signale à l'interface qu'il s'agit d'un mot de commande et non pas d'un caractère à transmettre.

Exemple : Programme permettant d'imprimer le caractère A sur le télécype d'adresse 17 (Programme rangé à l'adresse 2000_g).

A l'adresse MOT, nous avons le mot de commande, soit (120000)_g

A l'adresse CAR, nous avons le caractère A , soit (100101)_g

2000	LDA	MOT	A est chargé par le mot de commande,	062010
2001	OTA	15	Le mot est transféré à l'interface	102617
2002	LDA	CAR	Le caractère A est dans l'accumulateur	062011
2003	OTA	15	Le caractère A est dans le tampon	102617
2004	STC	15,C	Commande d'activation du télécype	103717
2005	SFS	15	Saut si le bit drapeau est à 1	102317
2006	JMP	* -1	Retour ou test si le bit drapeau est à 0	026005
2007	HLT		Arrêt si le bit drapeau est à 1	102000
2010	MOT	OCT	120000	120000
2011	CAR	OCT	101	000101

Exemple 4 Programme permettant de calculer la somme de 10 entiers rangés en mémoire à partir de 120_g (Programme rangé en 100_g).

Adresse	Programme symbolique	Commentaire	Programme octal
100	LDB ADRES	(B)=adresse du 1 ^{er} nombre	064110
101	CLA	(A)=0	002400
102	BOUCL ADA 1,I	(A)=(A) + nombre dont l'adresse est en B	140001
103	IMB	(B)=(B)+1	006004
104	ISZ I	(I)=(I)+1 avec saut si I = 0	034111
105	JMP BOUCL	Branchement si non terminé	024104
106	HLT	Arrêt si terminé	102000
110	ADRES	DEF NOMB	Adresse du 1 ^{er} nombre
111	I	DEC -10	Compteur
120	NOMB	DEC 1	000001
121		DEC 2	000002
122		DEC 3	000003
123		DEC 4	000004
124		DEC 5	000005
125		DEC 6	000006
126		DEC 7	000010
127		DEC 8	000011
130		DEC 9	000012
131		DEC 10	000013

} Nombres à additionner

Table A-4. Consolidated Coding Table

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
D/I	AND	001	0	Z/C	← Memory Address →											
D/I	XOR	010	0	Z/C												
D/I	IOR	011	0	Z/C												
D/I	JSB	001	1	Z/C												
D/I	JMP	010	1	Z/C												
D/I	ISZ	011	1	Z/C												
D/I	AD*	100	A/B	Z/C												
D/I	CP*	101	A/B	Z/C												
D/I	LD*	110	A/B	Z/C												
D/I	ST*	111	A/B	Z/C												
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
0	SRG	000	A/B	0	D/E	*LS	000	CLE	D/E	SL*	*LS	000				
						*RS	001				*RS	001				
						R*L	010				R*L	010				
						R*R	011				R*R	011				
						*LR	100				*LR	100				
						ER*	101				ER*	101				
						EL*	110				EL*	110				
						*LF	111				*LF	111				
						NOP	000			000						
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
0	ASG	000	A/B	1	CL*	01	CLE	01	SEZ	SS*	SL*	IN*	SZ*	RSS		
					CM*	10	CME	10								
					CC*	11	CCE	11								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
1	MAC	000	A/B	0	← Select Code →											
1	IOG	000	A/B	1	H/C	HLT	000									
						STF	001									
						CLF	001									
						SFC	010									
						SFS	011									
						MI*	100									
						LI*	101									
						OT*	110									
						STC	111									
				0	1	H/C	STC	111								
				1	1	H/C	CLC	111								
							STO	001	000						001	
							CLO	001	000						001	
							SOC	010	000						001	
							SOS	011	000						001	

Notes: * = A or B.
D/I, A/B, Z/C, D/E, H/C coded: 0/1.

INITIATION AU CALCULATEUR
INSTALLÉ SUR LE "CAPRICORNE"



par

A. MORLIERE

I.- Généralités

A) - Présentation du matériel

Le système installé à bord du "Capricorne" comprend:

- une unité centrale HP2115A
- un bloc d'alimentation de l'unité centrale
- un télétype ASR-33
- une unité à bande magnétique
- un lecteur optique de ruban perforé

1° - Unité centrale

Elle comprend:

- la mémoire centrale
- des éléments de calcul et de contrôle
- un jeu de cartes interface

La mémoire centrale: est une mémoire à tores de ferrite, elle comprend 8192 mots de 16 bits. Elle peut être étendue par l'addition de module d'expansion.

Le module de mémoire est divisé en "page" de 1024 mots, une "page" étant définie comme étant le bloc de mémoire le plus grand qui peut être "adressé" par les 10 bits des instruction Memory reference.

Le cycle de base est de 2 microsecondes.

Une partie de la mémoire (64 positions) est réservée pour le Basic Binary Loader qui constitue le programme de base pour charger des informations binaires. Cette partie de la mémoire est protégée.

! Les registres de travail

Registre - T (Memory Data): tous les éléments qui entrent ou sortent de la mémoire transitent par T

Registre - P (Program Counter): il contient toujours l'adresse de la prochaine instruction à exécuter!

Registre - M (Memory Address): il contient l'adresse de la mémoire en train d'être lue ou écrite.

Accumulateurs - A et B: Les accumulateurs contiennent les résultats des opérations arithmétiques ou logiques

adresse de A = 0

adresse de B = 1

Registre Extend

Registre Overflow

! Les cartes interface

Les cartes interface sont des éléments de liaisons entre l'unité centrale et des dispositifs extérieurs. Le 2115A accepte 8 cartes interface, c'est à dire qu'il peut être relié à 8 périphériques au maximum.

Dans notre système l'unité à bande magnétique utilise deux interfaces, la TTY une, le lecteur optique une, reste quatre cartes:

1 horloge

2 cartes d'acquisition (DSI)

1 carte analogiques-digital

Les cartes sont rangées dans un certain ordre correspondant à un système de priorité, la liaison ne pouvant être assurée qu'avec un seul périphérique!

2° - Télétype (ASR-33) HP2752A

C'est une machine à écrire munie d'un perforateur et d'un lecteur de ruban!

La cadence de travail est de 10 caractères par seconde. Elle peut travailler en ligne avec l'ordinateur ou en "local" c'est à dire seule.

A noter que le zéro s'écrit Ø

TTY est relié à l'ordinateur par l'intermédiaire d'une interface.

3^e - Lecteur optique de ruban perforé (HP2737A)

Lit le ruban perforé à la vitesse de 300 caractères/seconde.

4^e - Unité à bande magnétique (HP2020B)

Effectue lecture et écriture sur des rubans magnétiques à 7 canaux NRZI (compatible IBM); deux densités possibles: 200 bpi et 556 bpi; vitesse 30 i/seconde.

5^e - Sonde Bisset-Berman

La sonde Bisset-Berman mesurant S, T, P et O₂ peut être considérée comme un périphérique de l'ordinateur puisque l'acquisition des données de la sonde est contrôlée par l'ordinateur.

La liaison sonde-ordinateur est assurée par deux cartes interface.

6^e - Traceur de courbe

Le traceur de courbe est branché en dérivation sur le télétype, il peut être commandé par l'ordinateur (TTY sur LINE) ou par le télétype (LOCAL).

B) - Systèmes de Nombres

1^e - Les systèmes numériques

Tout nombre entier peut se mettre sous la forme polynomiale:

$$N = a_n b^n + a_{n-1} b^{n-1} + \dots + a_2 b^2 + a_1 b^1 + a_0 = \sum_{i=1}^n a_i b^i$$

b = base a_i = chiffre entier.

Système décimal: (base 10)

$$\begin{aligned} 109 &= 1 \text{ centaine} + 0 \text{ dizaine} + 9 \text{ unités} \\ &= 1 \times 100 + 0 \times 10 + 9 \times 1 \\ &= 1 \times 10^2 + 0 \times 10^1 + 9 \times 1 \end{aligned}$$

Conversion décimal - octal:

On fait des divisions successives par 8 en conservant le reste.

Exemple:
$$\begin{array}{r} 109 \mid 8 \\ 5 \mid 13 \mid 8 \\ \quad 5 \mid 1 \mid 8 \\ \quad \quad 1 \mid 0 \end{array}$$

 reste 5
 reste 5
 reste 1
 résultat: 155_8

Le dernier reste donne le caractère le plus significatif.

II.- La représentation des nombres:

Les calculateurs opèrent sur des nombres binaires pour lesquels il n'y a que deux états à distinguer: l'état "0" et l'état "1".

Un nombre est représenté par l'état d'un mot machine, le constructeur fixe la taille du mot machine c'est à dire le nombre de bits à l'intérieure de chaque mot (pour HP $n = 16$).

Nombres négatifs:

Ils sont représentés par leur complément à deux, le bit le plus élevé étant utilisé comme bit de signe.

signe	valeur	
0	11011	nombre positif
1	00100	son complément à un
0	00001	on ajoute 1
1	00101	son complément à deux (négatif)

La représentation par le complément à deux permet d'éviter la soustraction; elle se ramène à l'addition du complément à deux.

1 010	000	000	000	000	(-60000)
0 000	111	000	000	000	(+07000)
1 010	111	000	000	000	-5 1000

Nombre fractionnaires:

virgule fixe: $N = a_n D^n + a_{n-1} D^{n-1} + \dots + a_1 D + a + a D^{-1} + \dots + a_m D^{-m}$
est la représentation polynomiale d'un nombre fractionnaire.

Il faut toujours préciser la position de la virgule, en fait le nombre de chiffre après la virgule.

1968,625
11110110000,101000

virgule flottante (ou semi logarithmique)

1968,625 = $10^3 \times 1,968625$ en décimal

En binaire un nombre pourra toujours s'écrire: $x = a \times 2^b$

a = mantisse

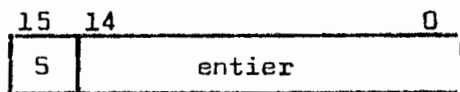
b = exposant

Pour représenter un nombre en virgule flottante dans un registre: les k premières cases codent b (nombre relatif), les (n - k) suivant codent la mantisse a.

L'intérêt de cette méthode est de traiter des informations ayant un ordre de grandeur différent en conservant une précision analogue. Les opérateurs d'addition et de multiplication deviennent cependant plus délicates.

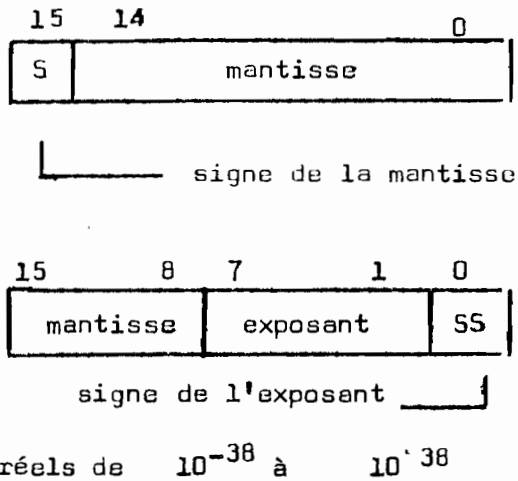
Cas du 2115A

Les entiers sont représentés par un mot de 16 bits le bit le plus significatif étant le bit de signe



entiers de: $- 2^{15}$ à $2^{15} - 1$ (32767)

Les réels sont représentés par 2 mots de 16 bits: 8 bits pour l'exposant et son signe, 24 bits pour la mantisse et son signe



La précision est de 7 chiffres décimaux.

III.- Procédure de programmation

A)-Principe

Un programme est un ensemble d'ordres donnés à la machine dans le but de lui faire exécuter un travail déterminé.

Il est rédigé dans un langage conventionnel que la machine est capable de comprendre.

En général les programmes sont rédigés dans un langage dit évolué c'est à dire proche du langage mathématique, c'est le cas du FORTRAN, de l'ALGOL du BASIC.

Ce langage doit être traduit en un langage plus simple pour la machine, cette traduction est faite par un "compilateur" (programme de traduction du langage machine).

Les programmes doivent être "chargés" dans la machine pour être exécutés.

Chaque instruction élémentaire du programme est installée dans une mémoire, cette mémoire est repérée par son adresse.

B) - Différentes formes d'un programme

1° - Programme source:

C'est le programme tel qu'il est écrit par l'utilisateur, pour

pouvoir être transmis dans la machine. Ce programme doit être perforé sur un ruban.

2° - Programme objet

C'est le résultat de la traduction par la machine du programme source en langage machine. Le programme objet, s'il n'est pas trop gros, peut être chargé dans la machine par l'intermédiaire d'un autre (BCS) pour être exécuté. Il ne faut pas qu'il soit trop gros car le BCS chargé avant lui occupe déjà une partie de la mémoire.

3° - Programme absolu

C'est la forme la plus élaborée du programme car sous cette forme le programme se suffit à lui-même pour fonctionner.

En effet un programme pour fonctionner a besoin de beaucoup d'éléments et en particulier d'un certains nombres de "routines" qu'il faut placer aux différents points où cela est nécessaire. Tout ceci est normalement effectué par le BCS mais si le programme est trop gros, les routines ne peuvent pas toutes entrer en mémoire centrale où il y a déjà le BCS et le programme. Pour remédier à cela on met le programme en absolu ce qui évitera de le charger par l'intermédiaire du BCS et libérera de la place en mémoire.

Pour obtenir le programme absolu, on fait un faux chargement par le BCS dont le résultat est une bande perforée comprenant le programme et l'ensemble des éléments qui lui sont nécessaires.

Les programmes absolus se suffisent à eux-mêmes ils nécessitent pour être chargés en mémoire centrale un programme très petit le BBL.

4° - Remarque

En fait la distinction entre objet et absolu est autre; un programme objet est un programme adressable c'est à dire qu'il peut être chargé à n'importe quel endroit de la mémoire. Les différentes instructions sont repérées

par leur adresse relative à la première instruction, si bien qu'il peut être placé n'importe où en mémoire par le BCS; il suffit de fixer l'adresse de la première instruction ce qui est le rôle du BCS.

Un programme absolu est par contre toujours chargé au même emplacement en mémoire, l'adresse de chaque instruction étant fixée au moment de la construction de l'absolu.

5° - Routines

On appelle routines un ensemble de petits programmes utilitaires adressables qui constituent une librairie de fonctions possibles. Il comprend des routines mathématiques (SIN, COS, ...) des routines de service etc...

Le BCS chargé de rassembler ces routines doit être chargé le premier, puis le programme et enfin la librairie.

C) - Mise en oeuvre d'un programme sur HP2115A

1° - FORTRAN - ALGOL - ASSEMBLER

Ces langages doivent être traduits par leurs compilateurs; il faut donc charger ceux-ci en mémoire avant de charger les programmes écrits dans ces langages (fig)

Lorsque le programme objet est obtenu sans erreur il doit être chargé avec les routines dont il a besoin.

2° - Basic

Pour le Basic les opérations sont beaucoup plus simples, le Basic étant un langage de conversation entre l'utilisateur et l'ordinateur.

On peut entrer le programme directement par le clavier les instructions sont corrigées, traduites au fur et à mesure et l'exécution peut commencer dès la fin de l'entrée du programme source sans qu'il y ait besoin de passer par une formule intermédiaire. En Basic il n'existe que des programmes sources.

III.- Software

"Software" est un terme général donné à tous les programmes et routines qui étendent les possibilités du calculateur.

A) - Basic binary Loader (BBL)

1^o - Rôle

Le BBL est un programme qui charge en mémoire des programmes absolus en utilisant certains périphériques d'entrée déterminés.

En fait c'est le chargeur de base de tout système, il est le programme minimum permettant d'en charger d'autres.

Il réside en permanence dans l'ordinateur dans une zone protégée (bouton LOADER ENABLED, PROTECTED); quand il est détruit il doit être rechargé à l'aide des clefs.

2^o - Utilisation

Pour pouvoir s'en servir il faut avoir accès à la zone de mémoire qui lui est réservée donc supprimer la protection.

```
UN PROTECTED (
17700          ) (adresse de départ du B.B.L)
LOAD ADDRESS  (
PRESET RUN    (
```

(chargement d'un programme absolu). En fin de chargement arrêt avec 102977 dans registre T

B) - Magnétique Tape System (M.T.S.)

1^o - Rôle

Le MTS est un ensemble de programme inscrit sur bande magnétique; il comprend des programmes de contrôle et d'organisation du fonctionnement de l'ordinateur et des programmes d'utilisation choisis par l'utilisateur.

Le rôle principal du MTS est de simplifier l'ensemble des opérations de chargement, ces opérations se résumant en un appel simple au clavier du

télétype des programmes d'utilisations désirés.

En fait en faisant un MTS on met en réserve sur une bande magnétique l'ensemble des programmes dont on peut avoir besoin.

2^e - Utilisation

Il faut mettre en place la bobine magnétique du MTS (en LOAD POINT).

16000 (adresse de départ du MTS)

LOAD ADDRESS

PRESET RUN

reponse: * NEXT ?

On appelle alors le programme désiré en respectant l'écriture:

: PROG, NDM

Le programme est alors recherché sur la bande et chargé en mémoire centrale.

3^e - Construction

Cette construction est régie par un programme absolu (PTS) dont le rôle est de demander les différents paramètres nécessaires pour le fonctionnement du MTS; il contrôle le chargement sur bande magnétique des différents programmes qui vont entrer dans la composition du MTS.

Le PTS fabrique également trois files sur la bande magnétique:

file 1: programmes absolus

file 2: programmes adressables (librairie)

file 3: laissé libre pour l'utilisateur.

Voir annexe A pour la procédure de construction du MTS.

C) - Basic Control System (BCS)

1^e - Rôle

C'est un programme absolu utilisé pour le traitement des programmes "objets" produit par les compilateurs.

Le BCS est un ensemble modulaire de plusieurs programmes qui dirige le chargement, combine et contrôle les routines nécessaires au fonctionnement du programme chargé, contrôle les procédures d'entrée-sortie.

Il comprend:

I/O Control programme generant les liaisons entre les demandes I/O

de l'utilisateur et les drivers appropriés.

I/O Drivers: programme qui a supervise la transmission des données dans un dispositif I/O donné.

Relocating Loader: programme permettant le chargement en mémoire des programmes objets de l'utilisateur.

2° - Construction

La fabrication d'un BCS configuré est obtenu à l'aide d'un programme PCS qui régit l'ordonnancement des éléments du BCS. Le PCS établit également "l'équipement table" qui permet de relier le hardware périphérique aux software d'entrée-sortie.

Voir annexe B pour la procédure de construction du BCS.

D) - Compilateurs

1° - Rôle

Ce sont des programmes qui traduisent les langages évolués utilisés par l'utilisateur en langage de la machine.

2° - Utilisation

Les compilateurs sont généralement intégrés dans le MTS et sont chargés en mémoire par l'intermédiaire de celui-ci.

E) - Editeur

1° - Rôle

C'est un programme de correction des bandes perforées.

2° - Utilisation

Intégrés au MTS la procédure d'appel est:

* NEXT ?

: PROG, EDIT

On place la bande à corriger dans le lecteur, on tape les corrections suivant les fonctions:

/I,r insertion d'une ligne ou plus après la ligne r
/D,r (r2) effacement de l'enregistrement r1
/R,r1 (r2) remplacement de l'enregistrement r,
/CI, r, C insertion d'un caractère à la ligne r, après le caractère
n° C
/CD, r, G (,C2)
/CR, r, C1 (,C2)
/L production d'un listing
/E fin

Voir annexe C pour la procédure complète.

IV.- Mise en place et utilisation d'un programme

Trois possibilités suivant la forme physique extérieure de ce programme.

A) - A partir de la bande perforée absolue:

Il faut charger le programme à l'aide du B.B.L
mise en place de la bande dans le lecteur.

17700

LOAD ADDRESS

LOADER ENABLED

PRESET RUN

lecture du ruban

PROTECTED

2 (ou SA du programme)

LOAD ADDRESS

RUN

B) - A partir de la bande magnétique

Le programme est contenu sur une bande magnétique dans un MTS.

mise en place de la bande magnétique

16000

LOAD ADDRESS

PRESET RUN

Reponse: * NEXT?

Tapier : : PROG, (NUM du programme)

C)- A partir du ruban perforé objet

Le programme doit être chargé par le BCS qui est contenu sur une bande système dans le MTS. Il faut donc appeler le BCS par la procédure précédente.

; PROG, BCS

mise en place de la bande objet dans lecteur optique

READ

quand la TTY écrit:

* LOAD

il faut charger la bibliothèque:

Lever clés n° 2 et 15

PRESET RUN

Réponse : LST

faire RUN

Réponse : * RUN

faire : RUN

Annexe A - PROCEDURE DE CONSTRUCTION D'UN SYSTEME MAGNETIQUE (MTS)

Charger le PTS à l'aide du BBL.

Afficher I00 aux clés (adresse de départ)

LOAD ADDRESS

Afficher I7 aux clés (pour configurer le PTS au départ du programme et lui indiquer que le canal de la Teletype est I7)

PRESET

RUN

PREPARE TAPE SYSTEM

PROGRAM INPUT DEVICE S.C.= ?

I5 (canal du lecteur optique)

ABSOLUTE PROGRAMS, FILE#1.

LOAD THESE TWO(2) MODULES FIRST:

.IPL.
S.SIO

I.D. NAME:
.IPL.
S.A.
100
* LOAD

BOOTSTRAP chargé par BBL
I00,LOAD ADDRESS ...

HP MAGNETIC TAPE SYSTEM

*BATCH OPTION DISABLED.

I.D. NAME:
S.SIO
S.A.
100
* LOAD

*NEXT?
: PFOG,BCS

I.D. NAME:
BCS
S.A.
2
* LOAD

De même pour tout autre programme à charger.

I.D. NAME:
/E
*EOF

RELOCATABLE LIBRARY, FILE#2.

* LOAD
* LOAD
* LOAD
* LOAD
* LOAD
* EOF

LIBRAIRIE
STDC
STD
SIGAL
cli 15 levé.

*END

Annexe B - PROCEDURE DE CONSTRUCTION D'UN BCS

Charger le PCS à l'aide du BBL.

2000

LOAD ADDRESS

I7

PRESET , RUN

HS 17P7

15

HS PUW7

16

FVA ME47

110

LVA ME47

15777

* LOAD ← BCS DRIVER D.23 -Bande Magnétique

D.23

14422 15777

* LOAD ← BCS DRIVER D.01 Lecteur Optique

D.01

14063 14421

* LOAD ← BCS DRIVER D.02 Perforateur rapide

D.02

13552 14062

* LOAD ← BCS DRIVER D.00 Teletype

D.00

13016 13551

* LOAD ← I/O CONTROL

ICC

12577 13015

* TABLE ENTRY

EQT?

10, D.23, U4 ←

15, D.01

16, D.02

17, D.00

18

Travail sur l'unité physique 0 en mode protégé

SQT?
 -XYED?
 12
 -TTY?
 12
 -LIB?
 7
 -PUNCH?
 11
 -INPUT?
 10
 -LIST?
 12

DMA?
 0

* LOADBCS RELOCATING LOADER

LOADR
 10153 12547

INTERRUPT LINKAGE ?

10, 20, i.23
 11, 21, C.23
 12, 0
 13, 0
 14, 0
 15, 22, i.01
 16, 23, i.02
 17, 24, i.00
 /E

.SQT. 12550
 .FQT. 12556
 D.23 14422
 i.23 15541
 C.23 15562
 DMA1 13014
 DMA2 13015
 .BUFR 12745
 iCERR 12773
 D.01 14063
 i.01 14200
 D.02 13552
 i.02 13667
 D.00 13016
 i.00 13172
 .iCC. 12577
 XSOT 13012
 XEOT 13013
 .LDR. 11651
 HALT 12536
 .4E4. 12543
 LST 10204

*SYSTEM LINK
 00110 00314

(pour faire un double,
 lever I5,RUN)

*ECS ABSOLUTE OUTPUT
 *END

Annexe C . EDITEUR

Charger EDIT soit avec le BBL (S.A.=100),
 soit à l'aide du système :

16000

LOAD ADDRESS

PRESET

RUN

La Teletype répond * NEXT ?

Taper : PROG,EDIT

HP SYMECLIC BATCH

EDIT FILE DEVICE?

/T

→ dirige sur TTY l'entrée des connexions

*

1P → " " L.OPT la lecture " "

/R, 16

IF(IR(2,J-1))2,3,2

/F

SYMECLIC FILE SOURCE DEVICE?

/P

SYMECLIC FILE DESTINATION DEVICE?

/P

**END OF TAPE

*

/F

COMPILATION ET SORTIE EN ABSOLU D'UN PROGRAMME FORTRAN

Sans Bande Magnétique .

Charger la PASS I du FORTRAN (par le BBL)

S.A.=50 ,LOAD ADDRESS,PRESET,RUN

Placer le ruban source dans le lecteur,

Taper FTN,B (pour sortie binaire),L (pour listing)

Un ruban RPI est perforé .Le mettre en place dans le lecteur optique

Charger le FORTRAN PASS 2 .Même procédure que plus haut

Starting Address =100

Un ruban objet est perforé . Etiquetter ,la Compilation est achevée

Sortie en Absolu

Charger le BCS par le BBL . S.A.=2

LOAD ADDRESS ,mettre le ruban objet dans le lecteur

Lever la clé I5 pour les messages de réservation mémoire

Lever la clé I4 pour obtenir la sortie au perforateur

PRESET,RUN

Après chaque message * LOAD ,mettre en place si nécessaire les sous-programmes dans le lecteur et faire RUN

Charger enfin la LIBRAIRIE de la même façon, en levant auparavant la clé 2 de manière à sélectionner les seules routines indispensables.

* LST

faire RUN ,RUN

En Assembleur

Pour Compiler un programme ,charger ASMB par le BBL

Placer la source dans le lecteur,S.A.=120 ,LOAD ADDRESS,PRESET,RUN,Taper sur :

la RTY :

ASMB,B,R (pour relocatable),L (pour listing)

Placer à nouveau le ruban source dans le lecteur

RUN

On obtient une sortie objet.

Sortie en Absolu : Repeter la procedure détaillée plus haut.

Avec un système magnétique, un certain nombre d'opérations sont automatiques, telles écriture de la PASS I sur la bande magnétique , puis appel de la PASS 2 du FORTRAN.

Si le BOOTSTRAP n'est pas détruit ,les procédures d'appel du FORTRAN

(:PROG,FOR),de l'assembleur (:PROG,ASMB),ou du BCS se font en initialisant la bande magnétique par I6000 ,LOAD ADDRESS ...

DEPOUILLEMENT AUTOMATIQUE DES MESURES
FAITES A L'AIDE DES COURANTOMETRES AANDERAA



par

A. MORLIERE et J. CITEAU

NOTE TECHNIQUE

MARS 1976

INTRODUCTION:

Le but de cet ensemble de programmes est de décoder les données issues des courantomètres Aanderaa de les traduire en vraie grandeur en effectuant les corrections qui sont nécessaires. Ces programmes ont été mis au point pour la chaîne d'acquisition de données du Centre ORSTOM d'Abidjan donc pour ordinateur Hewlett Packard.

Les résultats sont fournis sous deux formes: digitale et analogique; sous forme digitale les résultats sont sur ruban perforés (8 canaux ASCII + Binaire) ou sur bande magnétique (556 BPI - 7 canaux BCD + binaire); sous forme analogique les résultats sont sous forme de graphiques représentant les paramètres scalaires, les projections de vecteurs courant sur n'importe quel axe, les vecteurs eux-mêmes en fonction du temps ou de la profondeur (voir figures).

RESULTATS SOUS FORME DIGITALISEE

Les "stations" courantométrie sont individualisées par un entête comprenant les éléments de caractérisation de la station (date, heure, position, caractéristiques de l'appareil). L'entête est en code ASCII sur ruban perforé, en code BCD sur bande magnétique.

Les données proprement dites sont elles en binaire quelque soit le support utilisé, elle comprennent les 5 paramètres suivants: profondeur, température, salinité, composante ouest-est, composante sud-nord. La fin de station est matérialisée par un enregistrement de paramètres nuls.

Pour les stations provenant de courantomètres montés en "profilier" les composantes peuvent être corrigées d'une dérive.

Suivant les programmes utilisés les données peuvent être lissées par un filtre "antiparasite" ou par moyennes mobiles sur n valeurs. Ceci pour les mesures au mouillage, pour les mesures en "profilier" les données sont conservées brutes.

RESULTATS SOUS FORME ANALOGIQUE

Les résultats sont présentés sous forme de graphe de dimension 38 x 25cm (taille du traceur HP). Les données brutes ou des moyennes sur un intervalle de temps quelconque peuvent être tracées.

SOUS PROGRAMME ANDER.But

Ce sous programme écrit en ASSEMBLER permet de lire et de traduire les rubans perforés issus du décodeur AANDERAA.

Utilisation:

CALL ANDER (I1, IE1, I2, IE2, I3, IE3, I4, IE4, I5, IE5, I6, IE7)

I1 = Référence	IE1 = 0 OK
I2 = Température	i = 1,6 1 erreur sur paramètre I i
I3 = Conductivité	IE7 = 0 OK
I4 = Pression	1 erreur sur nombre de caractères
I5 = Direction	
I6 = Vitesse	

PROGRAMME DCODABut

Ce programme ainsi que ceux de la même série (DCODB, DECODP) permet d'obtenir sur support "informatique" les paramètres mesurés par un courantomètre Aanderaa, en vraie grandeur. Il utilise pour cela le sous programme ANDER ainsi que les coefficients d'étalonnage propres à chaque capteur.

Il utilise les rubans perforés par le décodeur Aanderaa et fournit soit un listing, soit un ruban perforé (code ASCII et binaire HP) soit un ruban magnétique (code BCD + binaire HP) sur lequel figure les éléments caractéristiques du lieu, de la date du début de la mesure, du courantomètre et l'ensemble des grandeurs mesurées (P,T,S,U,V) sur lesquelles on peut éventuellement effectuer un lissage par moyenne mobile.

Utilisation:

- adresse de départ: 2655
- choix du périphérique de sortie, déclinaison magnétique

question : OUT, DECL

réponse : 7, - 10

Ceci entraîne une sortie sur bande magnétique avec correction du Cap mesuré de la déclinaison magnétique de -10° .

OUT	sortie
2	listing
5	ruban perforé
7	bande magnétique

- moyenne mobile

question MOY SUR. CYCLES

réponse 10

Ceci entraîne un lissage par moyenne mobile sur dix cycles pour l'ensemble des paramètres mesurés. Dans ce cas la remise à l'heure et à la date du premier enregistrement est programmée car la moyenne mobile tronque un certain nombre d'information au début et à la fin de l'enregistrement. Dans certains cas le message MOIS DE JOURS demandera le nombre de jours du mois en cours.

Caractéristiques de la station

question: CR .. STA ** MO ** HEUR COUR

réponse 7606 002 260 276 L530

Ceci permet de fournir à l'ordinateur les éléments caractéristique de la station: n^o, date, position, appareil utilisé, temps d'échantillonnage.

Il est très important de respecter le format d'entrée de ces données en se basant sur le message écrit par la télétype. Faire très attention en tapant le temps d'échantillonnage, le chiffre doit être cadré à droite d'ou pour un temps de 10 minutes soit 600 secondes:

P E N D	et non	P E N D
6 0 0		6 0 0

En cas d'erreur on pourra refaire le message en répondant NON à la question CORRECT? qui va suivre.

- Caractéristiques des courantomètres

question	LIBRAIRIE - PAUSE
réponse	Mise de la librairie des caractéristiques dans le lecteur optique. Faire RUN PAUSE Mettre alors la station perforée dans le lecteur optique.

- Eléments de travail

question	CYCLES: DEBUT, FIN 10,163
----------	------------------------------

Ceci permettra d'éliminer systématiquement les 9 premiers cycles de mesures ainsi que tous les cycles au delà de 163.

- Fin de travail sur une station

Suivant l'option de sortie choisie, on est renvoyé sur une PAUSE (ruban perforé, bande magnétique) ou sur les caractéristiques de la station suivantes (télétype).

- Traitement de station par groupe

Pour lier entre elles sur le même support un groupe de station, après la PAUSE de fin de station; il faut lever une clé autre que (15) et faire RUN.

- Fin de travail sur un groupe de station

Pour terminer un ruban perforé ou une bande magnétique, après le PAUSE de fin de station, lever la clé (15) et faire RUN. Une terminaison est alors inscrite sur le support utilisé:

Programme DCODB

But

Mêmes fonctions essentielles que DCODA, mais au lieu d'effectuer une moyenne mobile sur les données, on effectue un lissage "antiparasites".

On définit un "parasite" par un changement isolé de concavité dans la succession des données.

Utilisation

- adresse de départ 2753

- Même chose que pour DCODA.

Programme DCODP

But

Il s'agit d'effectuer le même genre de travail qu'avec les programmes DCODA et DCODB mais à partir de données enregistrées par un courantomètre monté en profiler, et de ne conserver que les données correspondant à la descente du profiler; les données correspondant à la mise à l'eau et à remontée de l'appareil sont systématiquement éliminées.

Le temps d'échantillonnage est obligatoirement fixé à 30 s. Les données de courantométrie peuvent être au choix de l'utilisateur, corrigé d'une certaine dérive liée aux conditions d'exécution du profil de courant.

Pour certaines stations longues, il peut être nécessaire d'effectuer un deuxième passage dans le lecteur optique de la bande de données.

Utilisation:

La philosophie et les messages sont sensiblement les mêmes que pour l'ensemble des programmes de type DCOP, il y a cependant certaines différences liées à la méthode de travail.

- adresse de départ 2422

- choix de la profondeur de référence et de la dérive

question: LIST (1), DE Z1, A Z2

DERIVE (2), DIR VIT

réponse : Deux réponses possibles

Réponse 1: 1, 490, 500

Dans ce cas, on désire connaître les valeurs enregistrées entre 490 et 500 mètres afin de déterminer la valeur de la dérive que l'on va adopter pour corriger l'ensemble des données. Si les bornes fixées sont hors des limites des valeurs enregistrées on obtiendra un listing des 10 derniers mètres enregistrés.

Réponse 2: 2, 280, 12

Dans ce cas, on veut corriger l'ensemble des données d'une dérive apparente de 12 cm/s dans le 280 qui est le plus souvent la valeur du courant mesuré au point le plus profond. Elle correspond à une dérive réelle du bateau de 12 cm/s dans le 100. Pour un bateau au mouillage la réponse est 2,0,0.

- Station trop longue

Pour une station profonde, il peut y avoir impossibilité de stocker toutes les informations en mémoire, un deuxième passage de la bande perforée est alors demandée par le message: PASS 2

PAUSE

- Stations effectuées à suivre

Dans certains cas les stations "profilier" ont pu être exécutées à suivre, sans interruption des enregistrements. Dans ce cas, après le PAUSE de fin de station, lever les (1) (2) (3) et faire RUN, le ruban perforé se positionnera au début de la station suivante.

- Fin d'un groupe de station

Comme pour DCODA et DCODB lever la clé (15)

Programme FILER

But

Le programme a pour but de visualiser les mesures faite à l'aide d'un Aanderaa monté en "profilier". Il utilise les données issues du programme DCODP. Il fournit des graphiques représentant la température, la salinité, la composante du vecteur courant sur un axe quelconque, de vecteur courant aux différents niveaux de mesure; il permet aussi de tracer l'ellipse de courant de la station. L'axe Sud-Nord peut être choisi faisant un angle quelconque avec l'axe des y (petit axe).

Utilisation

- adresse de départ 2
- choix du périphérique d'entrée et de la station:

question	:	IN, STATION?
réponse	:	7 (BM), n 5 (RP)

- choix des échelles et de l'origine des axes

Le choix des échelles et de l'origine se fait en déterminant les valeurs de XMIN, WMAX et YMIN, YMAX.

X MIN = valeur minimale possible sur l'axe des x

X MAX = " maximale " "

L'axe des x représente toujours la profondeur, la longueur utile sur le graphe est de 380mm.

Question X MIN, MAX

Réponse 0,760

{ échelle telle que 1 mm = 2m
{ origine au point le plus à gauche de
{ l'axe des x

Pour l'axe des y la longueur utile est de 250mm.

- Projection des vecteur vitesse

question: T = 2 S= 3 PROJ. AXE = 4 VECT = 5 ELLIPSE = 6

Réponse : 4

Question: YMIN MAX

Réponse: - 100, 150

Question: DIR AXE/N

Réponse : 90

Cette procédure va permettre de tracer, en fonction de la profondeur, la projection du vecteur sur un axe situé au 90 (composante E-W), l'origine sera sur l'axe des y à 100m du coin inférieur gauche, l'échelle sera de 1cm pour 10 cm/s.

- Tracé du vecteur courant

Question: T = 2 S= 3 PROJ. AXE = 4 VECT = 5 ELLIPSE = 6

Réponse : 5

Question: YMIN, MAX

Réponse : 100, 150

Question: DIR AXE/N

Réponse: 45

Cette procédure va permettre le tracé des vecteurs vitesse aux différentes profondeurs d'observation, le long d'un axe parallèle à l'axe des x et placé à 150mm du coin inférieur gauche. La direction N-S se trouve à 45° de l'axe des Y (le programme trace cette direction dans le quart supérieur droit de la feuille). L'échelle est de: 1 cm = 10 cm/s.

- Tracé de l'ellipse de courant

Question: T = 2 S= 3 PROJ. AXE = 4 VECT = 5 ELLIPSE = 6

Réponse : 6

Question : YMIN, MAX

Réponse:- 200, 50

Question: DIR AXE/N

Réponse: 45

Cette procédure va tracer les vecteurs vitesse avec une origine commune placée au milieu de l'axe des x et à 200mm du bord inférieur du papier. Même échelle et même direction de l'axe N-S que dans la procédure précédente.

- Tracé de la température et la salinité

Il suffit de répondre par 2 ou 3 à la question principale et de choisir correctement les bornes de l'axe des y.

- Interruption en cours de tracé

Pour l'interruption le tracé il suffit de lever une clé quelconque.

Programme MOTRA

But

Ce programme permet de visualiser les mesures faites par un courantmètre AANDERAA au mouillage. Il utilise les données issues de DCODB ou DCODA. Il permet de moyennner les données (moyenne simple sur n valeurs), puis de les tracer en fonction du temps (l'unité de temps = 1 cycle est le

temps d'échantillonnage des courantomètres x n). Il peut tracer en fonction du temps: la pression, la température, la salinité, le vecteur courant, la projection du vecteur courant sur un axe quelconque, l'ellipse de courant et l'hodographe intégré (PROG = 7).

Utilisation:

Les méthodes utilisées sont les mêmes que pour le programme FILER la seule différence venant que l'on trace en fonction du nombre de cycles (axe des x). Le nombre de cycle ~~minimum~~ est 1, le maximum est à fixer par l'utilisateur en fonction des échelles de temps désirée et de la longueur de l'enregistrement.

Par exemple: question: NB CYCLES MAX

réponse: 190

Ceci entraîne que 380mm correspondent à 190 cycles, si le cycle de mesure est de 20 minutes on aura comme échelle 1mm = 10 minutes.

Autre différence entre FILER et MOTRA, le calcul de la moyenne simple de n mesure (NB VAL A MOY). Il est à noter que la durée du cycle est de: n x temps échantillonnage).

Exemple NB VAL MOY

3

T BASE = 30

P = 1 T = 2 S = 5 PROJ. AXE = 5 VECT = 6 PROG = 7 ELLIPSE = 8

6

Y MIN, MAX

= 180 , 70

NB CYCLES MAX

63

DIR AXE/N

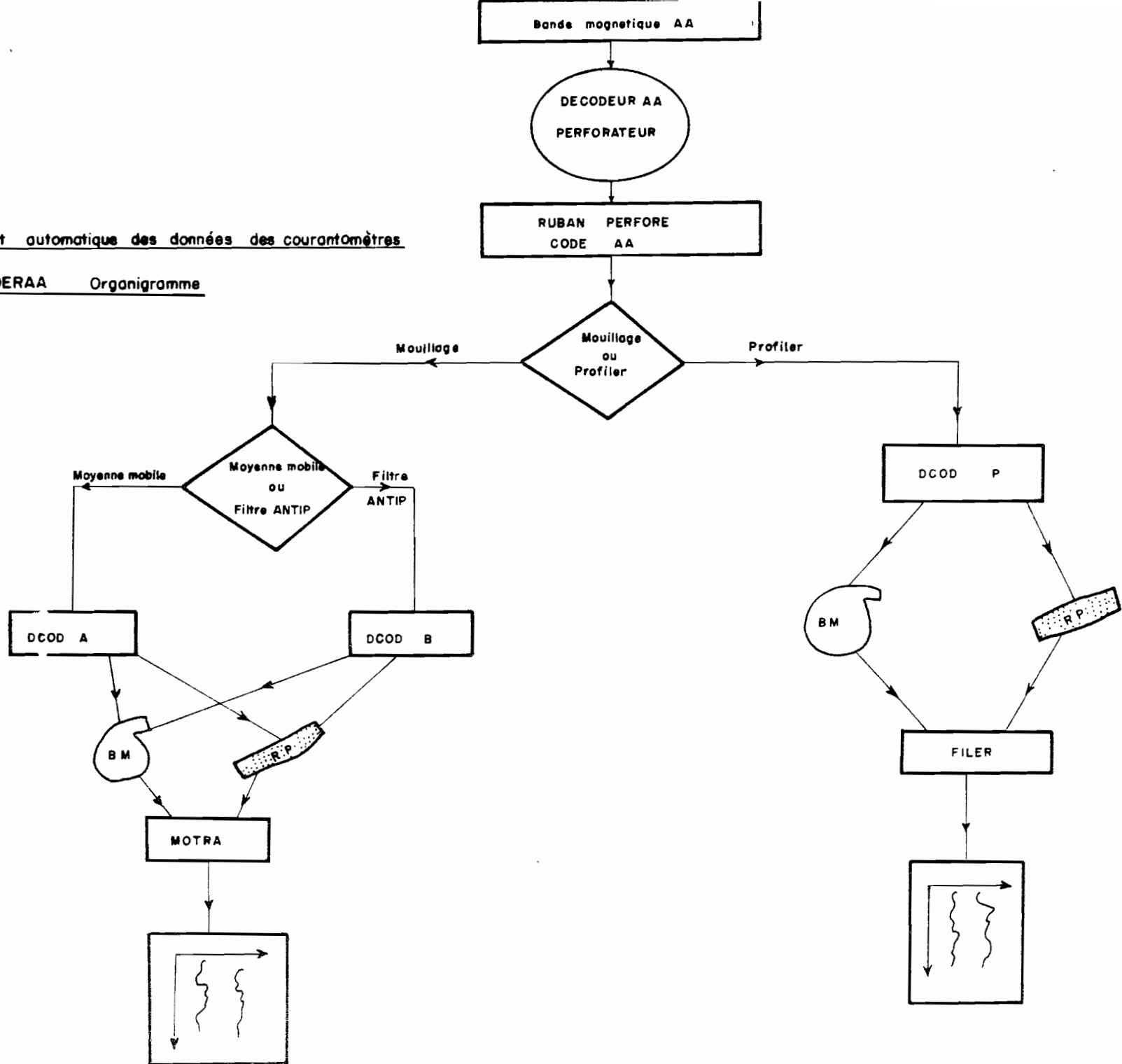
45

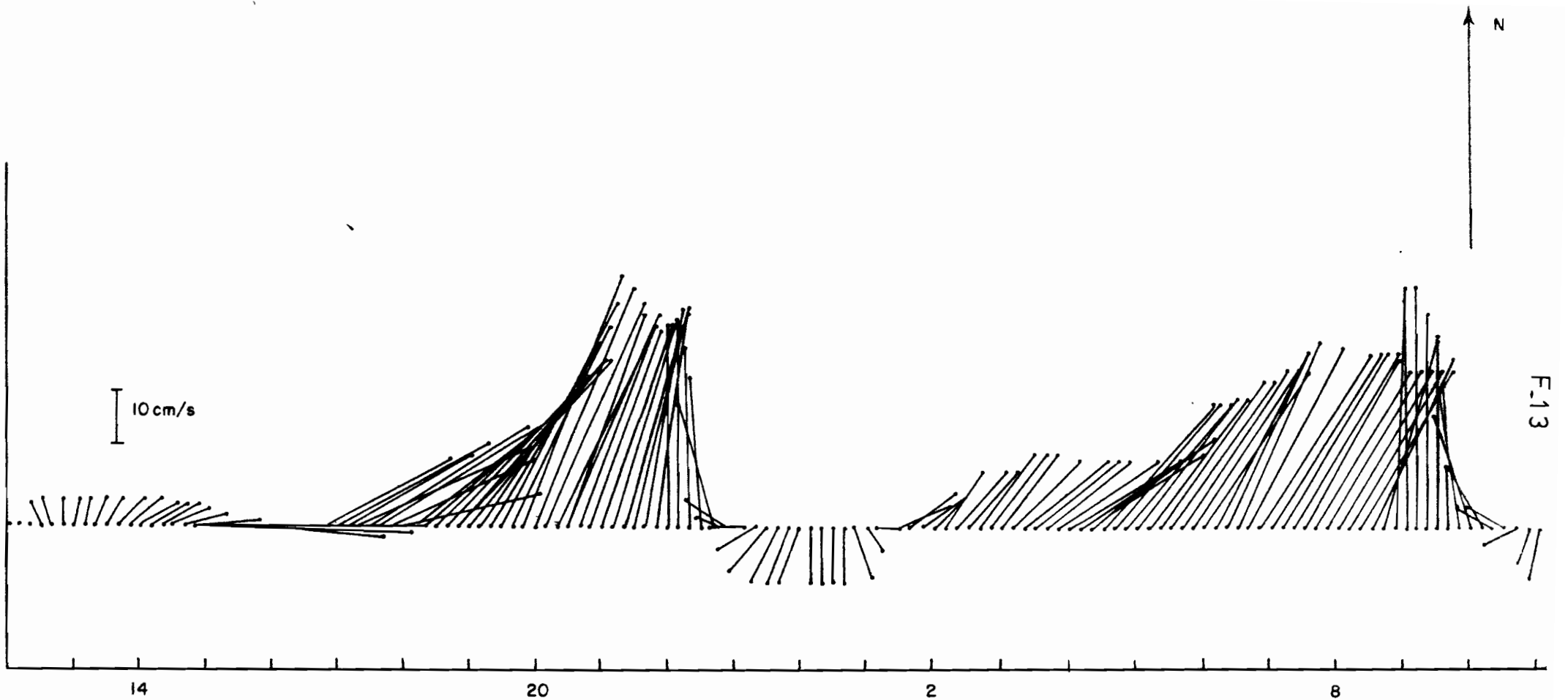
Cette procédure permet de tracer le vecteur courant moyenné sur 3 mesures en fonction du temps. Les échelles sont de: 1 mm = 5 minutes, 1 cm = 10 cm/s.

L'origine est à 180mm du coin inférieur gauche. L'axe S-N à 45° de l'axe des Y.

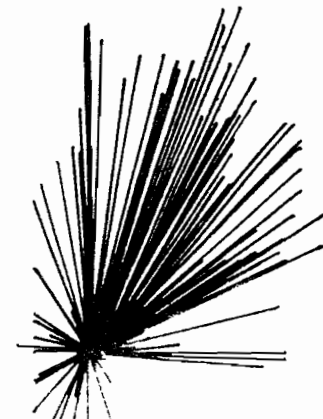
Depouillement automatique des données des courantomètres

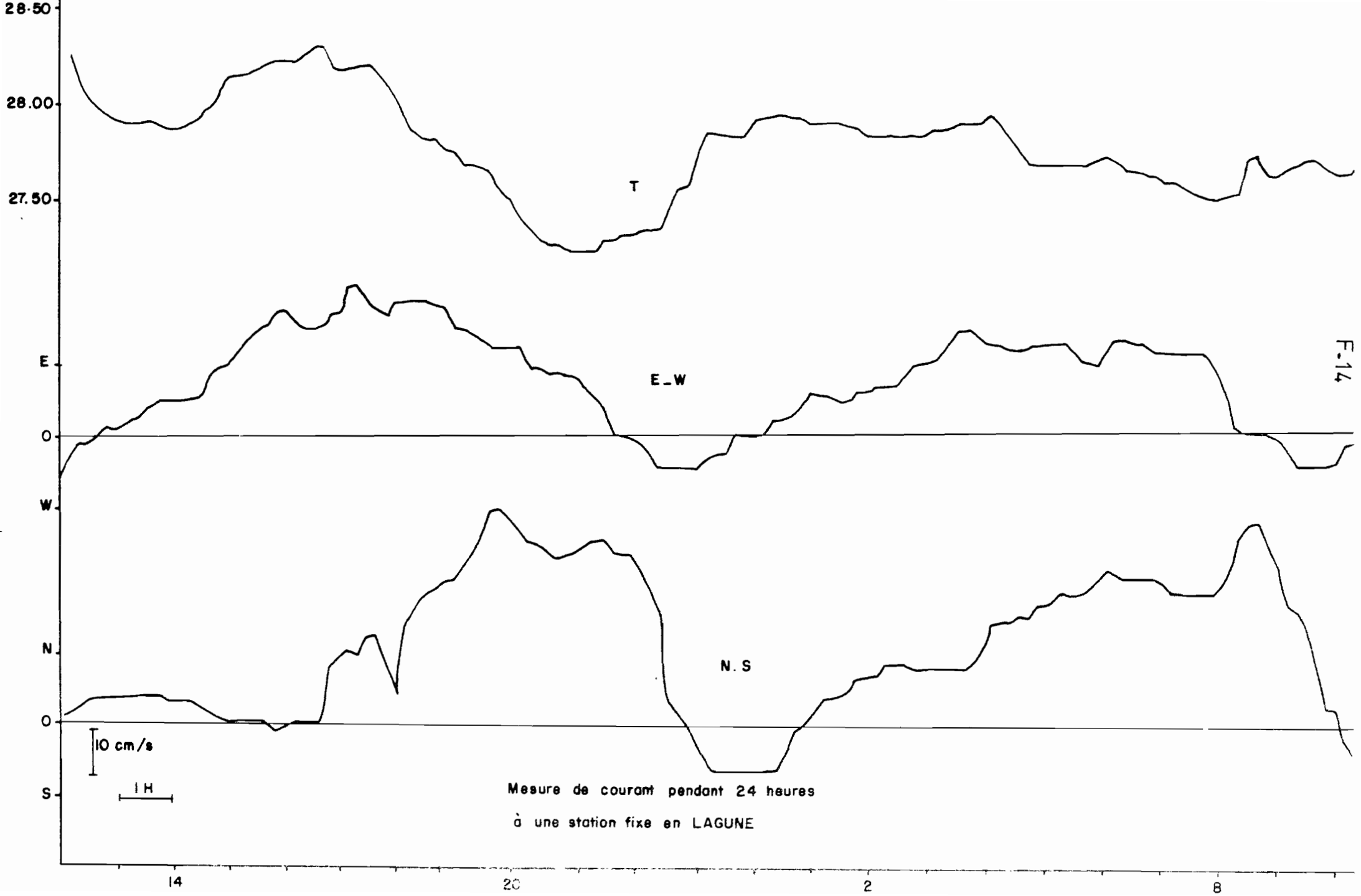
AANDERAA Organigramme





Mesure de courant pendant 24 heures
à une station fixe en LAGUNE





Mesure de courant pendant 24 heures
à une station fixe en LAGUNE

PLAN

A . CATALOGUE DES PROGRAMMES

B . ORGANIGRAMMES DES CHAINES DE TRAITEMENT

C . PROGRAMMES

Acquisition

Reduction

Controle - Correction

Tracés

Assemblage

Calcul Sortie

Sous programmes

Divers

ABREVIATIONS

	A	ASCII	FORMAT(5(3I4,I3))
	A*	ASCII	LIBRE (*)
	ASMB	ASSEMBLEUR	
	FØR	FØRTRAN	
	L	LISTING	
ETIQUETTES:	M	METEO etiq.sortie	{ J,M,A,H,C,ST,LAT, ^N S,LON, ^E W,T,T',TS,D,V,Z,MP FORMAT(3I3,I5,I6,I4,I6,A2,I7,A2,3I4,2I3,I5, I16)
	M'	METEO ruban entrée	{ C,ST,j,M,A,H,LAT, ^N S,lON, ^E W,T,T',TS,D,V,Z FORMAT(I4,I3,3I2,I4,I5,A1,I6,A1,3I3,2I2,I4)
	S	SIMPLE etiq.	{ J,M,A,H,C,ST FORMAT(3I3,I5,I6,I4)
	M _{AA}	"METEO" Format AANDERAA :	Crois,Sta,J,M,An,Hr,Lat, ¹ / ₁₀ N,Long, ¹ / ₁₀ E,No Cour.,Pendule (I4,I3,3I2,2I4,I4,A1,I5,I4,A1,2(1X14))
	P	PLOTTER	
	I	TTY (in)	
	2	TTY (out)	
	4	TAPE PUNCH	
	5	TAPE READER	
	7	MAGNETIQUE TAPE	
	B	BINAIRE	
	RP	RUBAN PERFORE	
	BM	BANDE MAGNETIQUE	

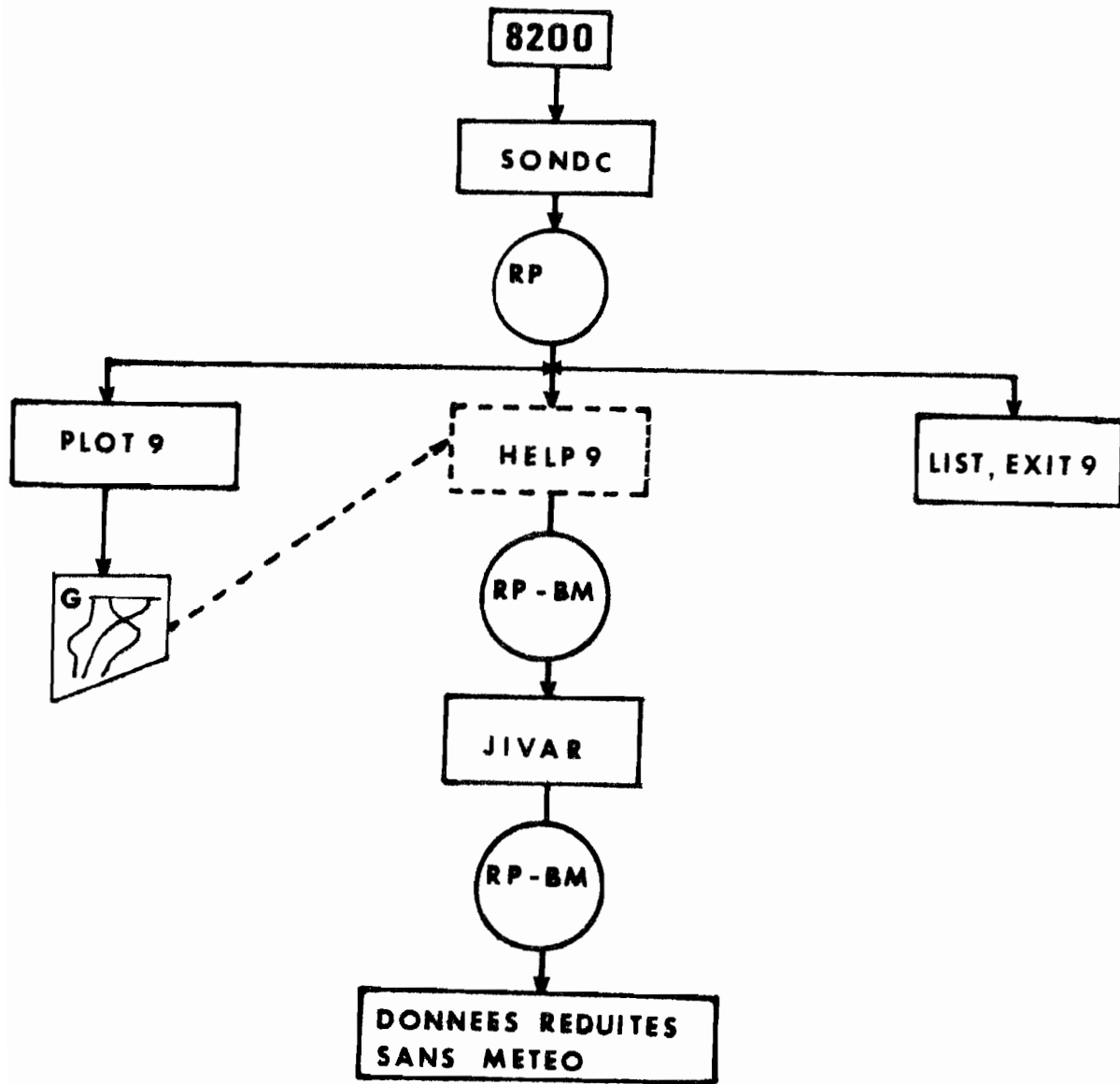
Note : La fin de station ,aussi bien en hydrologie qu'en courantométrie , est signalée par une valeur nulle de la température. De plus sur bande magnétique,un END OF FILE est inscrit à la fin de chaque station.Sur tous supports,une série de 9 indique la fin du ruban de données.

CATALOGUE des PROGRAMMES

TYPE	NOM	BUT	PERIPH.		FORMAT ETIQ.					
			IN	OUT	IN	OUT	IN	OUT	IN	OUT
ACQUISITION	SOND9 SONDC	Acquisition des données Sonde du 8200	8200	4,7	B	B	S	S	S	S
	SOND8	" + filtre et choix du coef de correction C(ΔT)	8200	4,7	B	B	S	S	S	S
	TOTAL	Enregistre sur B.M. ttes les valeurs du 8200 (brut)	8200	7	B	B	S	S	S	S
	HYDRØ (TANIA)	Perforation des données digitalisées aux graphiques	1,1 (5)	4,7	A	A	M	M	M	M
	DICØR	" + corrections divers (P, R, I, L, S%, OZ, D)	5	4	A	A	M	M	M	M
REDUCTION	JIVART	Réduction des données Sonde à moins de 100 h/s + filtre	7,5	2,4	B	A	S	S	S	S
	JIVARM	" + addition météo, sans B.M.	5	2,4	B	A	S	M	M	M
CORRECTION	HELP9	Correction des données Sonde binaires	7,5	4,7	B	B	S	S	S	S
	HELPR	Correction des données réduites binaire	7,5	4,7	A	A	M	M	M	M
	GØGØ	plot en continue 30 stations pour une param. T(P), S(P), OZ(P)	7,5	2	A	P	M	S	S	S
	DICØR	Corrections d'une station réduites ou digitalisées	5,1	4	A	A	M	M	M	M
	TESTD	Vérification de la liaison 8200-Ordinateur	8200	2	B	L	x	x	x	x
TRACÉS	PLØT9	Plot les profils T, S, OZ, TS, TOZ pour une station Bin.	7,5	2	B	P	S	L	L	L
	PLØTR	" " réduites	7,5	2	A	P	M	L	L	L
	DELTA	Plot les profils VT, ΔST	7,5	2	B	P	S	L	L	L
	PØINT	Plot les positions des stations avec le météo	5	2	A	P	M	L	L	L
	LØØK	Plot par point les isolignes d'une radiale	7	2	B	P	M	L	L	L
	LØØK2	Dans le plan des isanostères, il plot les isolignes T, S, OZ, Z	7	2	B	P	M	L	L	L
	LØØK3	Donne la cote dynamique des isobares	7	2	B	P	M	L	L	L
	TSOZ	pointes sur une m. feuille les diagramme TS, TOZ, SOZ d'un ensemble de stations	7	2	B	P	M	L	L	L
	GRADT	Visualisation des gradients thermiques	7	2	B	L	M	L	L	L

CATALOGUE (suite)

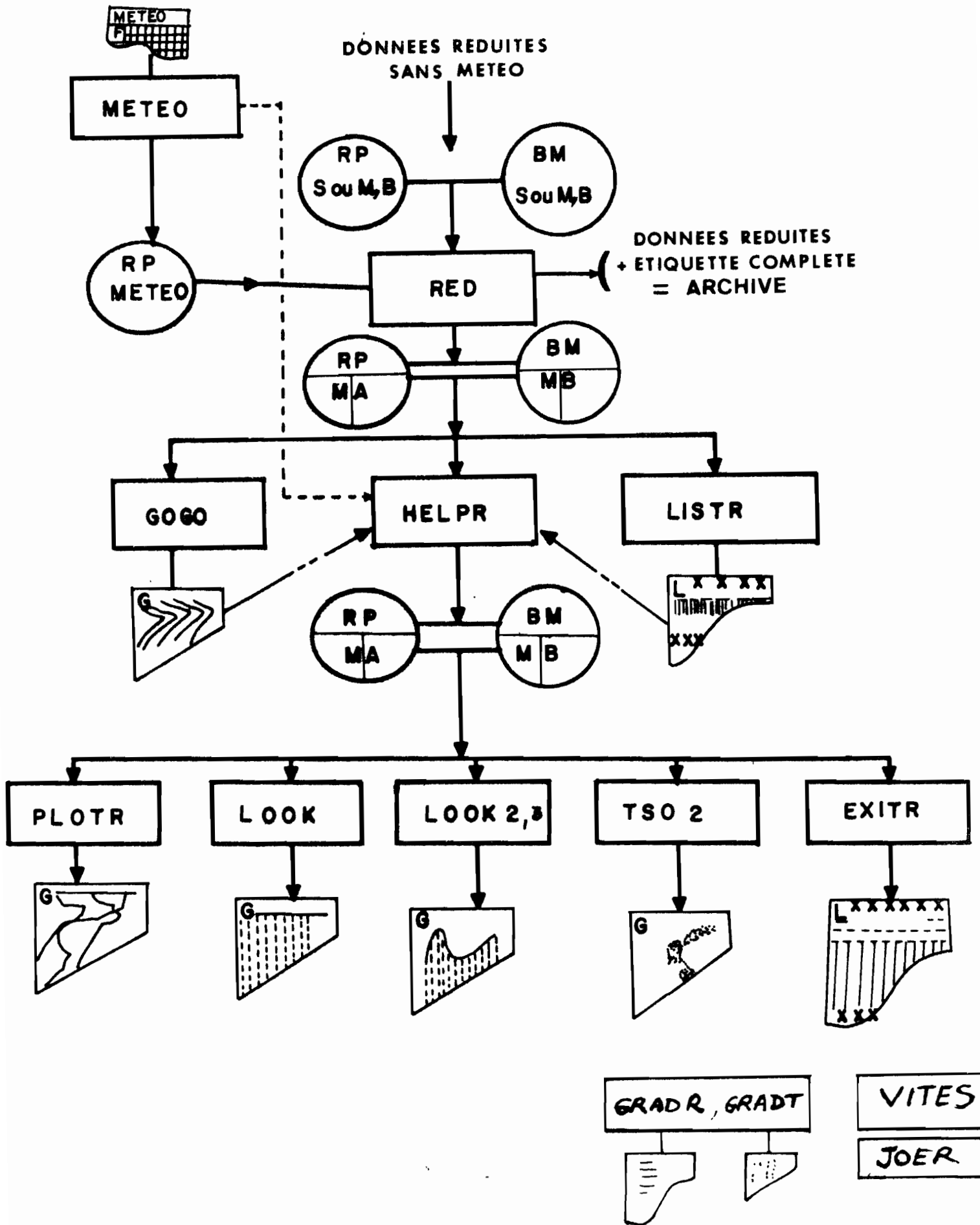
TYPE	NOM	BUT	peri.		format		etiq	
			IN	OUT	IN	OUT	IN	OUT
	TRAD	traduction Binaire - Ascii, Assemblage et transfert binaire	5,7	4,7	B/A	B	S	S
TRANSFERT	RED	Assemblage et transfert réduit + addition météo	5,7	2,4,7	A	A	M	M
	BANAL	Catalogue des stations sur bande Magnétique	7	2	B	L	S	S
CALCUL SORTIE	EXITR	Calcul dynamique d'une station réduite	5,7	2	A	L	M	L
	LISTR	Listings des profondeurs avec T, S, O ₂ réduites	5,7	2	A	L	M	L
	PYCNE	donne P, T, S, O ₂ sur les isomastères et les isopycnes	7	2	B	L	M	L
	VITES	Calcul des vitesses géostrophiques	5,7	2	B	L	M	L
	JOE-R	Calcul des hauteurs dynamiques et des faucheurs de hauteur	1,5,7	2	B	A	M	M
	GRADR	Calcul des gradient (T, S, O ₂ 1 ^{er} 2 ^{em} réduites) pour chaque profondeur	7	2	B	L	M	M
	GRADS	Calcul des gradients de T°, S‰, O ₂ , σ _t	5	2	A	L		
SOUS PROGRAMMES	CLE	Charge dans l'argument le contenu du registre clé						
	OMEGA	horloge, l'argument égal 1/10 seconde						
	STDC	Contrôle du hardware d'acquisition des données decodage, tri et mixe en forme des signaux bruts du 8200						
	SIGAL	Calcul σ_T , $\Delta\sigma$, arguments (P, T, S, σ_T , $\Delta\sigma$) (subroutine)						
	SIGMZ	Calcul σ_0 (fonction)						
	SIGMA	est appelé par SIGMZ arg(T, S) (fonction)						
	SUITE	Appel IPL (XCCR1, XCCR2, XCCR3) (choix C)						
	OXY	calcul le pourcentage d'oxygène dissous. (fonction)						
	OSCAR	Contrôle des entrées sortie des SOND8						
	COR	Entrée du coef. de correction des SOND8						
	SOS	Correction des tubaux mal perforés par MCTTY						
	ANDER	Convertit des nombres codés en FFI-PS en nombres de 0 à 1024	5					

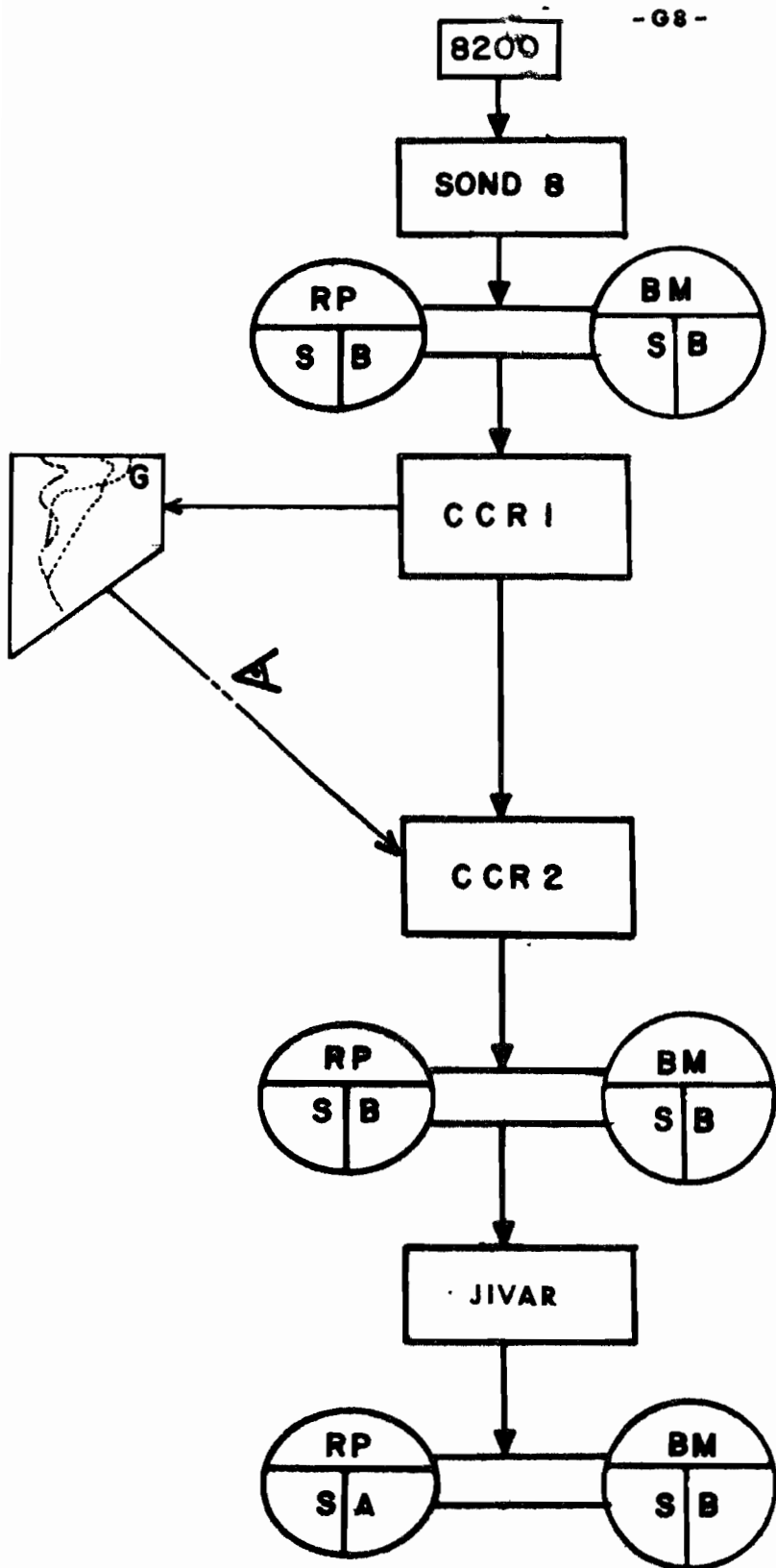


Acquisition en mer

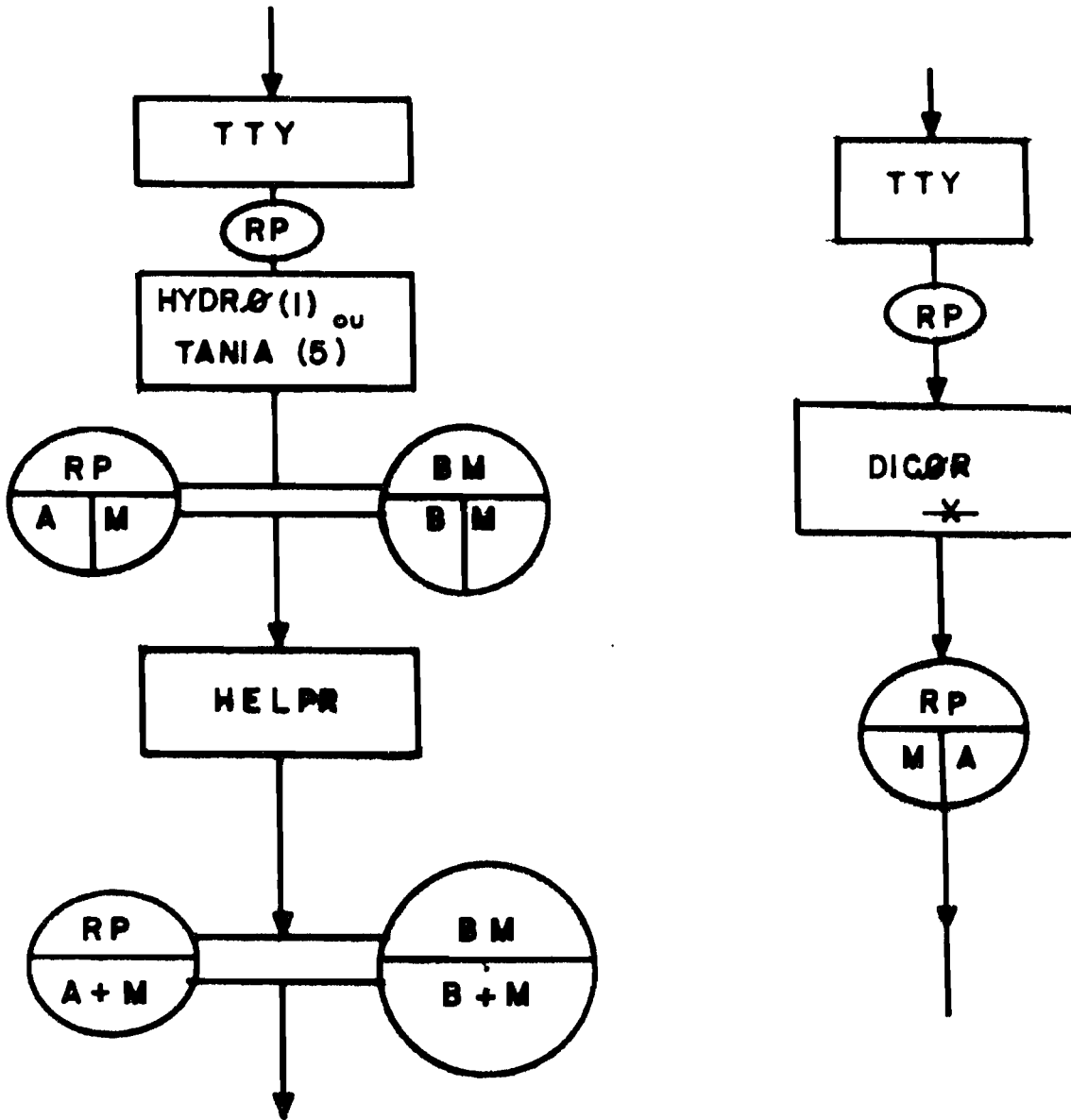
1) ORGANIGRAMME DES CHAINES DE TRAITEMENT
DES DONNEES

SONDE





ACQUISITION EN MER avec lissage et procédure d'appel automatique de tracés et de correction.



Acquisition des Données digitalisées avec les graphiques.

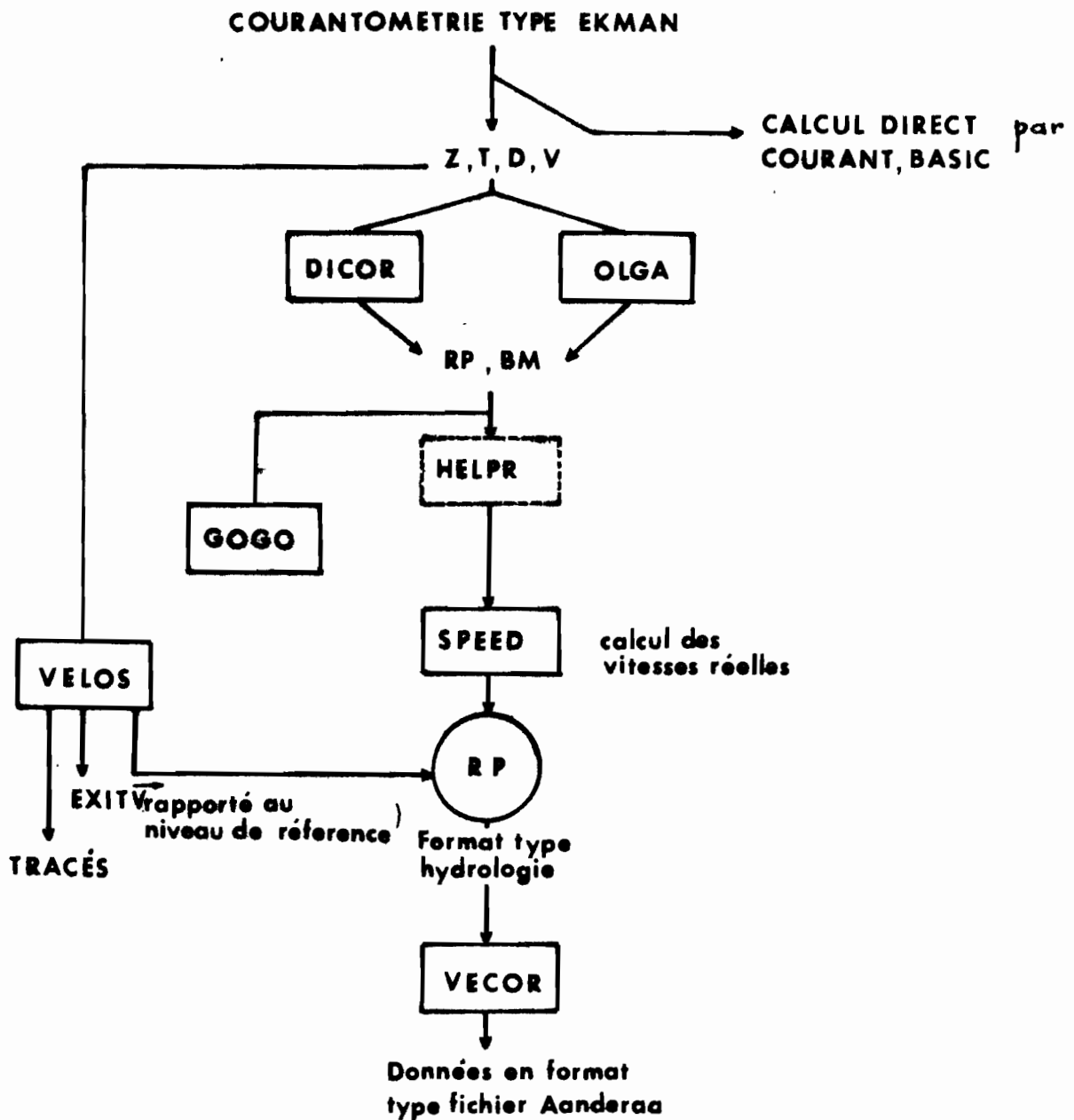
COURANTOMETRIE AANDERAA

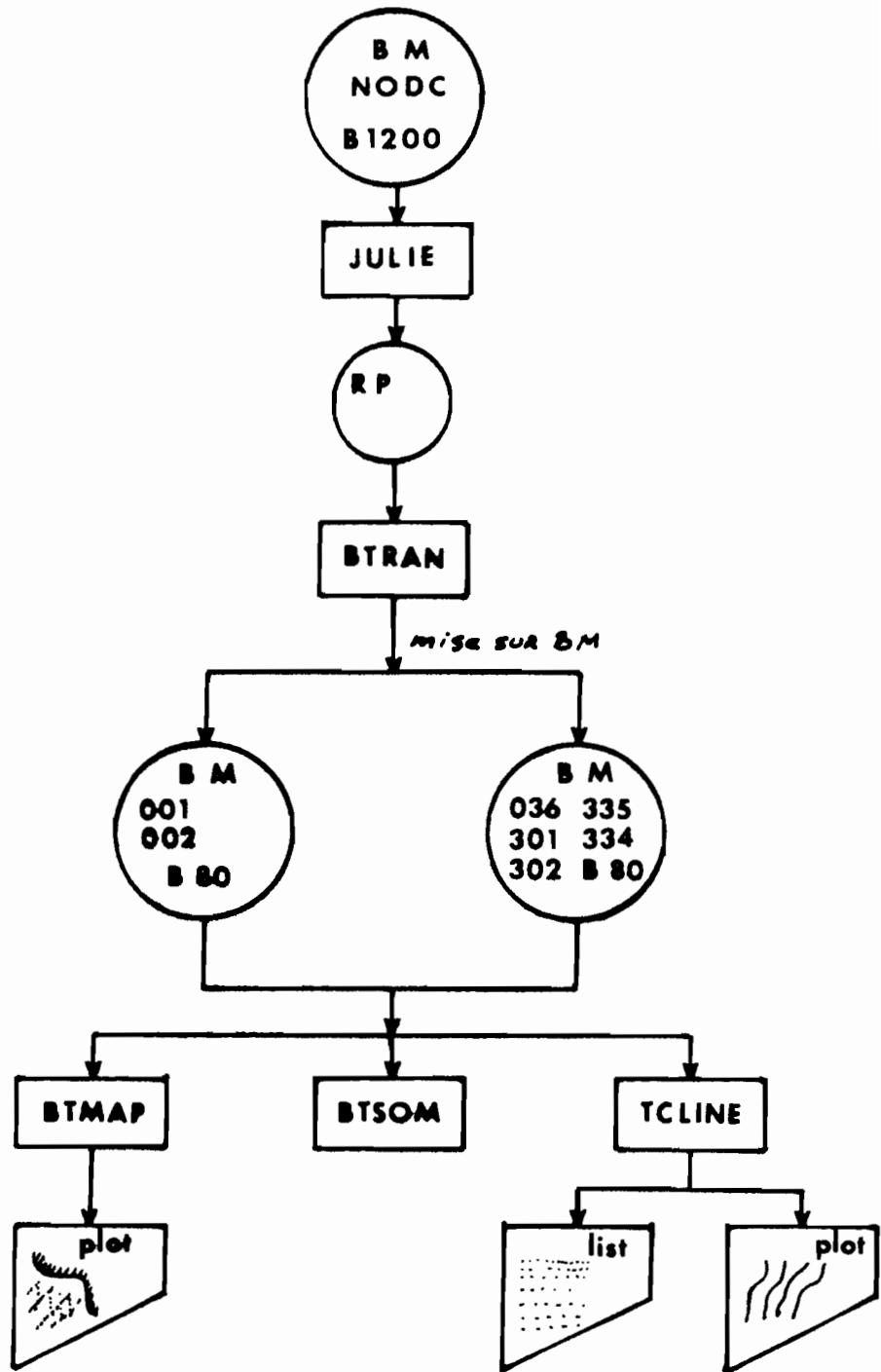
Profiler ou mouillage (voir note F: "Dépouillement automatique des mesures faites à l'aide des courantomètres Aanderaa.")



CONSTITUTION D'UN FICHIER

COURANTOMETRIE TYPE AANDERAA





BT.NODC

FICHE TECHNIQUE DU PROGRAMME

NOM: SONDB

TYPE	CHAINE			PERIPH.	IN	8200
BUT	Acquintance de données avec filtrage et correction au fur et à mesure.				OUT	74
				FORMAT	IN	B
			OUT		B	
Observations	voir fiche Soude 9			ETIQ.	IN	S1
Opérations	Acquisition, perforation				OUT	S
DATE DE MISE AU POINT				PROGRAMMEUR		
LANGAGE UTILISE	FORTRAN + S.P ASSEMBLER			MORLIERE		
SSP EMPLOYES	OSCAR, COA, OMEGA, CLE, STDC*(ancien)					
Starting ADDRESS	2					
Nb Source	Nb Objet		Nb Absolu			
Divers	*A partir de 77 nouvelle liaison 8200 - ordinateur -					
Clés employées	Voir SONDB					

FICHE TECHNIQUE DU PROGRAMME

NOM: SONDC

TYPE	CHAINE			PERIPH.	IN	8200
BUT	Acquisition de données.				OUT	84
				FORMAT	IN	B
			OUT		B	
Observations				ETIQ.	IN	S
Opérations	Acquisition, Perforation				OUT	S
DATE DE MISE AU POINT	1976			PROGRAMMEUR		
LANGAGE UTILISE	FORTRAN + SP. ASSEMBLER			Morliere		
SSP EMPLOYES	CLE, OMEGA, STDC*					
Starting ADDRESS	2					
Nb Source	Nb Objet		Nb Absolu			
Divers	*Nouvelle liaison 8200 - ordinateur (voir note technique)					
Clés employées						

CONTRÔLE LIAISON

3 2264 3081 32

DATE, HEURE, CRG, STA ?
21.06.77, 1339, 7700, 007

PAUSE
CCPIE
PAUSE
CCPIE
PAUSE

```
PROGRAM SOND8
DIMENSION ID(4,481),IM(4),IFILT(6,4)
C
C   COMMUN AVEC OSCAR POUR OUT40
COMMON ID40(41)
IMAX=0
N=6
IQ=3
CALL COR(C)
50 DO 1 I=1,4
DO 1 J=1,481
1 ID(I,J)=0
C
C   CONTROLE LIAISON, ENTETE
C
CALL OSCAR
CALL STDC(JP1,JT1,JS1,JO1,JCLE)
2 DO 178 IN=1,6
CALL STDC(JP,JT,JS,JO,JCLE)
IF(JCLE-1)4,4,6
C
C   CORRECTION DE GRADIENT THERMIQUE
C
4 CR=FLOAT(JT-JT1)
KN=JT
JT=JT1
JS=IFIX((1.-C*CR)*FLOAT(JS))
JT1=KN
C
C   FILTRAGE
C
IFILT(IN,1)=JP
IFILT(IN,2)=JT
IFILT(IN,3)=JS
178 IFILT(IN,4)=JO
DO 171 L=1,4
DO 177 K=1,IQ
IM(L)=0
M=N-K+1
C
C   MOYENNE VALEURS RESTANTES APRES CHAQUE ELIMINATION
C
DO 172 I=1,M
172 IM(L)=IM(L)+IFILT(I,L)
IM(L)=IFIX(FLOAT(IM(L))/FLOAT(M)+.5)
MAX=0
C
C   RECHERCHE VALEUR PLUS ELOIGNEE DE MOYENNE
C
DO 173 J=1,M
IA=IABS(IM(L)-IFILT(J,L))
IF(MAX-IA)179,179,173
179 MAX=IA
I=J
173 CONTINUE
M=N-K
C
C   ELIMINATION DE CETTE VALEUR
C
DO 177 J=I,M
177 IFILT(J,L)=IFILT(J+1,L)
M=N-IQ
```

```
      M=N-IQ
      IM(L)=0
C
C      CALCUL MOYENNE DES VALEURS RESTANTES
C
      DO 176 J=1,M
176  IM(L)=IM(L)+IFILT(J,L)
      IM(L)=IFIX(FLOAT(IM(L))/FLOAT(M)+.5)
171  CONTINUE
C
C      TRI - DE 300. + DE 300 M
C
      IF(IM(1)-302)20,20,21
20  IND=IM(1)+1
      GO TO 500
21  IND=(IM(1)-303)/5+304
      CALL OMEGA(10)
500  IF(ID(1,IND)-8)7,2,2
C
C      MISE EN MATRICE
C
      IO=IFIX(FLOAT(IM(4))/.14286+.5)
      ID(1,IND)=ID(1,IND)+1
      ID(2,IND)=ID(2,IND)+IM(2)
      ID(3,IND)=ID(3,IND)+IM(3)
      ID(4,IND)=ID(4,IND)+IO
C
C      PROF MAX ATTEINTE
C
      IF(IMAX-IND)175,175,2
175  IMAX=IND
      GO TO 2
C
C      VALEUR MOY /M
C
      6 DO 10 K=1,IMAX
        IF(ID(1,K))10,10,12
12  DO 13 L=2,4
13  ID(L,K)=IFIX(FLOAT(ID(L,K))/FLOAT(ID(1,K))+.5)
      ID(1,K)=K-1
10  CONTINUE
      IF(IMAX-302)302,302,71
C
C      PROF DE 5 EN 5 AU DELA DE 300 M
C
      71 M=300
        DO 60 K=303,IMAX
          M=M+5
60  ID(1,K)=M
302  JP=IMAX
C
C      BLOC NUL FIN DE STATION
C
      DO 200 I=1,10
        JP=JP+1
        DO 200 IW=1,4
200  ID(IW,JP)=0
```

C ELIMINATION DES TROUS
C

```
J1=0
40 J1=J1+1
   IF(ID(3,J1))40,40,41
41 DO 203 K=1,J1
   ID(1,K)=K-1
   203 =2,4
203 (,K)=ID(I,J1)
   K=IMAX-1
   DO 201 I=J1,K
   IF(ID(3,I))202,202,201
202 DO 205 J=2,4
205 ID(J,I)=(ID(J,I-1)+ID(J,I+1))/2
   ID(1,I)=I-1
201 CONTINUE
```

C
C CHOIX SORTIE
C

```
CALL CLE(IW)
IF(IW-7)600,700,700
```

C
C POSITIONNEMENT SUR BM
C

```
700 CALL FINBM(IC0)
   N=7
   IF(IC0-9999)300,301,301
300 CALL PTAPE(7,1,0)
   GO TO 700
301 CALL PTAPE(7,-1,-1)
   END FILE 7
```

C
C ENTETES
C

```
601 CALL LABEL(N)
   J1=1
   J10=10
310 M=2
```

C
C DONNEES (BLOC DE 40)
C

```
DO 325 J=J1,J10
DO 325 I=1,4
ID40(M)=ID(I,J)
325 M=M+1
   CALL OUT40(N)
   IF(ID(2,J10-1))320,320,326
326 J1=J10+1
   J10=J1+9
   GO TO 310
600 CALL LEADR(4,10)
   N=4
   GOTO 601
320 IF(N-7)605,603,603
```

C
C FIN BM
C

```
603 ENDFILE 7
   CALL NEUF(N)
   REWIND 7
   N=4
   GO TO 603
   FIN PP
605 CALLNEUF(N)
   CALL LEADR(4,10)
   CALL COPIE
   PAUSE
   GO TO 601
   END
   ENDS
```

```
0001      PROGRAM SQJDC
0002      DIMENSION ID(4,461)
0003      50 DO 1 I=1,4
0004          DO 1 J=1,461
0005              1 ID(I,J)=0
0006              WRITE(2,100)
0007      100 FORMAT(//"CONTROLE LIAISON")
0008      CALL STDC(JP1,JT1,JS1,JO1,JCLE)
0009      WRITE(2,130)JP1,JT1,JS1,JO1
0010      130 FORMAT(3I5,I4,/"DATE,HEURE,CRO,STA ?")
0011      READ(1,*)JR,MO,IA,IH,IC,IST
0012      PAUSE
0013      CALL STDC(JP1,JT1,JS1,JO1,JCLE)
0014      2 CALL STDC(JP,JT,JS,JO,JCLE)
0015      IF(JCLE-1)4,4,6
0016      4 IF(JP-302)20,20,21
0017      20 IND=JP+1
0018      GO TO 500
0019      21 IND=(JP-303)/5+304
0020      CALL OMEGA(10)
0021      500 IF(ID(1,IND)-8)7,23,23
0022      23 JT1=JT
0023      GO TO 2
0024      7 COR=FLOAT(JT-JT1)
0025      CORS=COR*1.6
0026      IT=JT1
0027      IS=JS-IFIX(CORS+.5)
0028      IO=IFIX(FLOAT(JO)/.14286+.5)
0029      ID(1,IND)=ID(1,IND)+1
0030      ID(2,IND)=ID(2,IND)+IT
0031      ID(3,IND)=ID(3,IND)+IS
0032      ID(4,IND)=ID(4,IND)+IO
0033      JT1=JT
0034      GO TO 2
0035      6 DO 10 K=1,IND
0036          IF(ID(1,K))10,10,12
0037      12 DO 13 L=2,4
0038      13 ID(L,K)=IFIX(FLOAT(ID(L,K))/FLOAT(ID(1,K))+.5)
0039      ID(1,K)=K-1
0040      10 CONTINUE
0041      IF(IND-302)302,302,71
0042      71 M=300
0043      DO 60 K=303,IND
0044          M=M+5
0045      60 ID(1,K)=M
0046      101 FORMAT(20X I4)
0047      302 JP=IND
0048          DO 200 I=1,10
0049              JP=JP+1
0050          DO 200 IW=1,4
0051      200 ID(IW,JP)=0
0052              J1=0
0053              40 J1=J1+1
0054                  IF(ID(3,J1))40,40,41
0055              41 K=IND-1
0056                  DO 201 I=J1,K
0057                      IF(ID(3,I))202,202,201
0058              202 DO 205 J=2,4
0059              205 ID(J,I)=(ID(J,I-1)+ID(J,I+1))/2
```

```
0060      ID(1,i)=i-1
0061 201 CONTINUE
0062      GO TO 600
0063 601 WRITE(N,602) JR,MO,IA,II,IC,IST
0064      J1=1
0065      J10=10
0066 310 WRITE(N)((ID(i,J),i=1,4),j=J1,J10)
0067      IF(ID(2,J10))605,605,325
0068 325 J1=J10+1
0069      J10=J1+9
0070      GO TO 310
0071 600 CALL LEADR(4,10)
0072      N=4
0073      GO TO 601
0074 605 WRITE(N,615)
0075 602 FORMAT(3i3,i5,i6,i4)
0076 615 FORMAT(24(1H9))
0077      CALL LEADR(4,10)
0078 300 WRITE(2,380)
0079 380 FORMAT("COPIE")
0080      PAUSE
0081      GO TO 201
0082      END
0083      ENDS
```


FICHE TECHNIQUE DU PROGRAMME

NOM: DICOR

TYPE	CORRECTION, ACQUISITION	CHAINE	D	PERIPH.	IN	5		
BUT	Correction et acquisition des donnees digitalisees ou reduites			OUT	4			
				FORMAT	IN	A, A*		
				OUT	A			
Observations				ETIQ.	IN	M		
Opérations	Lecture, /R /I /D /S% /O2 /L ,Perforation			OUT	H			
DATE DE MISE AU POINT	10 OCTOBRE 1974			PROGRAMMEUR				
LANGAGE UTILISE	FORTRAN			DARBOIS				
SSP EMPLOYES	CLE LEADR							
Starting ADDRESS	2							
Nb Source	I	Nb Objet	I				Nb Absolu	I
Divers								
Clés employées	Cle(n) en haut arret 9999							

FICHE EXPLOITATION

```

DICOR: ENTREE ?
IN: ASCII=1, **2
2
760600115057600300V00100F1231231231023456
 15 5 76 30 7606 1 10 02 312 310 234 56 0 0
CR=1, ETIQ=2
1
0, 1345, 3567, 345
2, 1344, 3567, 567
3, 1245, 3467, 676
3666
9999
/P=1 /R=2 /I=3 /S%=4 /O2=5, /L=6, /D=7
6
N1, N2?
0, 3
 0 1 2 3
4032 0 2 3
. 3960 1345 1344 1245
2913 3567 3567 3467
4346 345 567 676
/P=1 /R=2 /I=3 /S%=4 /O2=5, /L=6, /D=7
7
N0?
1
1
9999
/P=1 /R=2 /I=3 /S%=4 /O2=5, /L=6, /D=7
1
DICOR: ENTREE ?
    
```

```

0001      PROGRAM DICOR
0002      C
0003      C                               DIGITAL DATA CORRECTION
0004      C
0005      DIMENSION IR(4,100),MET(15),IDC(4)
0006      1  WRITE(2,200)
0007      CALL LEADR(4,15)
0008      WRITE(2,212)
0009      READ(1,*)KKK
0010      3  CONTINUE
0011      CALL CLE(IP1)
0012      IF(IP1)74,5,74
0013      5  CONTINUE
0014      C
0015      C                               LECTURE METEO + STATION
0016      C
0017      GOTO 94
0018      8  WRITE(2,605)JR,MO,IA,IH,IC,IST,(MET(I),I=7,15)
0019      WRITE(2,201)
0020      READ(1,*)LC
0021      GO TO(95,7),LC
0022      7  WRITE(2,211)
0023      READ(1,102)IC,IST,JR,MO,IA,IH,(MET(I),I=7,15)
0024      GO TO 8
0025      C
0026      C
0027      102 FORMAT(14,I3,3I2,14,I5,A1,I6,A1,3I3,2I2,14)
0028      625 FORMAT(3I3,15,I6,14,I6,A2,17,A2,3I4,2I3,15,I6)
0029      412 FORMAT(5(3I4,I3))
0030      99  FORMAT(24(1H9))
0031      200 FORMAT("DICOR")
0032      201 FORMAT("OK=1,ETIQ=2")
0033      203 FORMAT("/P=1 /R=2 /I=3 /S%=4 /O2=5, /L=6, /D=7")
0034      204 FORMAT("NO,2,T,S,0;99")
0035      206 FORMAT("S=S*(1-CDT) ,C?")
0036      207 FORMAT(1X,14I5)
0037      208 FORMAT("O2=A*SONDE+B ,A,B?")
0038      209 FORMAT("N1,N2?")
0039      110 FORMAT(3I3,15,I6,14)
0040      210 FORMAT("N0?")
0041      211 FORMAT("METEO")
0042      212 FORMAT("IN:ASCII=1,*=2")
0043      94  DO 11 J=1,100
0044          DO 11 I=1,4
0045              11 IR(I,J)=0
0046              NP=0
0047      C
0048          GOTO(1000,101),KKK
0049      95  GOTO(53,100),KKK
0050      1000 J1=1
0051          READ(5,625)JR,MO,IA,IH,IC,IST,(MET(I),I=7,15)
0052          GOTO 8
0053      53  J10=J1+9
0054          READ(5,412)((IR(I,J),I=1,4),J=J1,J10)
0055          J1=J10+1
0056          IF(IR(2,J10))52,52,53
0057      52  DO 10 J=1,100
0058          IF(IR(2,J))9,9,10
0059      10  NP=NP+1

```

```
0060      9 GO TO 249
0061 101 READ(5,102)IC,IST,JR,NO,IR,IR,(NET(I),I=7,16)
0062      GOTO 8
0063 100 NP=NP+1
0064      READ(5,*)(ID(I),I=1,4)
0065      DO 82 I=1,4
0066 82  IR(I,NP)=ID(I)
0067      IF(IR(I,NP)-9999)130,83,83
0068 80  DO 81 I=1,4
0069 81  IR(I,NP)=0
0070      NP=NP-1
0071 249 WRITE(2,203)
0072      READ(1,*)NC
0073      GOTO(251,253,253,502,603,703,803),NC
0074 C
0075 C          CORRECTION ET INSERTION NIVEAU
0076 C
0077 250 WRITE(2,204)
0078 252 READ(1,*)IL,I2,IT,IS,I0
0079      IF(IL-99)254,249,249
0080 254  IF(NC-2)251,253,255
0081 255  J1=IL+1
0082      IL=IL
0083      NP=NP+1
0084      J=NP
0085 13  DO 12 I=1,4
0086 12  IR(I,J)=IR(I,J-1)
0087      J=J-1
0088      IF(J-J1)253,13,13
0089 253  IR(1,IL)=I2
0090      IR(2,IL)=IT
0091      IR(3,IL)=IS
0092      IR(4,IL)=I0
0093      GO TO 252
0094 C
0095 C          SALINITE S=S*(1-CDT)
0096 C
0097 500 WRITE(2,206)
0098      READ(1,*)KC
0099      WRITE(2,207)(IR(1,J),J=3,16)
0100      WRITE(2,207)(IR(3,J),J=3,16)
0101      J1=NP-1
0102      DO 501 J=2,J1
0103      A1=FLOAT(IR(2,J)-IR(2,J-1))
0104      A2=FLOAT(IR(1,J)-IR(1,J-1))
0105      B1=FLOAT(IR(1,J+1)-IR(1,J))
0106      B2=FLOAT(IR(2,J+1)-IR(2,J))
0107      DT1=A1/A2
0108      DT2=B2/B1
0109      DT=(DT1+DT2)/2.
0110      C=FLOAT(KC)/1000000.
0111      IF(J-2)502,502,503
0112 502  S1=FLOAT(IR(3,1))
0113      S1=S1*(1.-C*DT1)
0114      IR(3,1)=IFIX(S1)
0115 503  IF(J-J1)505,504,504
0116 504  S2=FLOAT(IR(3,NP))
0117      S2=S2*(1.-C*DT2)
0118      IR(3,NP)=IFIX(S2)
0119 505  S1=FLOAT(IR(3,J))
```

EXECUTION DICOR

RP IN 5

7406055140974120005560N014370W27123727016344000

0 2702 3514 610

4 2700 3513 610

7 ...

...

9999 0 0 0

DICOR

RIIN

PAUSE

IN:ASCII=1,*=2

2

14 9 74 1200 7406 55 5560N 14370W 271 237 270 16 4 4000

OK=1,ETIQ=2

2

METEO

7406055140974120005560N014370W27123727016054000

14 9 74 1200 7406 55 5560N 14370W 271 237 270 16 5 4000

OK=1,ETIQ=2

1

/P=1 /R=2 /I=3 /S%=4 /O2=5, /L=6, /D=7

2

NO,Z,T,S,0;99

4,2650

4 12 2672 3513 610

99

/P=1 /R=2 /I=3 /S%=4 /O2=5, /L=6, /D=7

3

NO,Z,T,S,0;99

26

2 6 2700 3513 610

99

/P=1 /R=2 /I=3 /S%=4 /O2=5, /L=6, /D=7

4

S=S*(1-CDT) ,C?

125

6 7 12 34 40 44 45 50 60 72 76 85 91

3513 3513 3513 3514 3517 3523 3525 3564 3565 3560 3558 355632707

3518 3518 3513 3514 3518 3534 3550 3587 3577 3566 3552 355732707

/P=1 /R=2 /I=3 /S%=4 /O2=5, /L=6, /D=7

5

O2=A*SONDE+B ,A,B?

.7868,-15.2

/P=1 /R=2 /I=3 /S%=4 /O2=5, /L=6, /D=7

6

N1,N2?

1,3

1 2 3

0 4 6

2702 2700 2700

3514 3513 3518

464 464 464

/P=1 /R=2 /I=3 /S%=4 /O2=5, /L=6, /D=7

7

N3?

3

99

/P=1 /R=2 /I=3 /S%=4 /O2=5, /L=6, /D=7

1

PAUSE

CLE 0 t

DICOR

PAUSE

```
0120      SI=SI*(1.-C*DT)
0121      IR(3,J)=IFIX(SI)
0122      501 CONTINUE
0123      WRITE(2,207)(IR(3,J),J=3,15)
0124      GOTO 249
0125      C
0126      C          OXYGENE UR=A*5UNDE+5
0127      C
0128      600 WRITE(2,208)
0129      READ(1,*)A,B
0130      DO 601 J=1,NP
0131      601 IR(4,J)=IFIX(FLOAT(IR(4,J))*A+B)
0132      GOTO 249
0133      C
0134      C          CONTROLE
0135      C
0136      700 WRITE(2,209)
0137      READ(1,*)N1,N2
0138      J1=N1
0139      703 J14=J1+13
0140      IF(N2-J14)701,701,702
0141      701 J14=N2
0142      702 WRITE(2,207)(J,J=J1,J14)
0143      DO 705 I=1,4
0144      705 WRITE(2,207)(IR(I,J),J=J1,J14)
0145      J1=J14+1
0146      IF(N2-J1)704,703,703
0147      704 GOTO 249
0148      C
0149      C          DELETE LEVEL
0150      C
0151      800 WRITE(2,210)
0152      802 READ(1,*)N0
0153      IF(N0-99)801,249,249
0154      801 DO 803 J=N0,NP
0155      DO 803 I=1,4
0156      803 IR(I,J)=IR(I,J+1)
0157      NP=NP-1
0158      GOTO 802
0159      C
0160      C          PERFORATION STATION
0161      C
0162      251 WRITE(4,625)JR,MO,IA,IH,IC,IST,(MET(I),I=7,15),NP
0163      J1=1
0164      63 J10=J1+9
0165      WRITE(4,412)((IR(I,J),I=1,4),J=J1,J10)
0166      J1=J10+1
0167      IF(IR(2,J10))62,62,63
0168      62 CALL CLE(IP1)
0169      IF(IP1)74,6,74
0170      74 WRITE(4,99)
0171      CALL LEADR(4,15)
0172      GOTO 1
0173      END
0174      ENDS
```

**END-OF-TAPE

*

FICHE TECHNIQUE DU PROGRAMME

NOM: TOTAL

TYPE	ACQUISITION	CHAINE	PERIPH.	IN	820
BUT	Enregistre sur BM tts les valeurs P T S O qui arrivent du 8200		PERIPH.	OUT	7
Observations			FORMAT	IN	E
Opérations				OUT	E
			ETIQ.	IN	S
				OUT	S
DATE DE MISE AU POINT	AVRIL 1974		PROGRAMMEUR		
LANGAGE UTILISE	FORTRAN		DARBOIS		
SSP EMPLOYES	STDC PTAPE				
Starting ADDRESS	2				
Nb Source	Nb Objet	Nb Absolu			
Divers					
Clés employées					

FICHE EXPLOITATION

```
0001      PROGRAM TOTAL
0002      DIMENSION IA(4,30)
0003      99      WRITE(2,100)
0004      100     FORMAT("TOTAL"/"CONTROLE LIAISON")
0005      CALL STDC(IP,IT,IS,IO,IC)
0006      WRITE(2,200) IP,IT,IS,IO
0007      200     FORMAT(4I6)
0008      CALL PTAPE(7,1,0)
0009      PAUSE
0010      KFIN=1
0011      4      DO 1 I=1,30
0012             CALL STDC(I1,J,K,L,M)
0013             IF(1-1)2,2,3
0014      2      IA(1,I)=I1
0015             IA(2,I)=J
0016             IA(3,I)=K
0017             IA(4,I)=L
0018      1      CONTINUE
0019             KFIN=KFIN+1
0020             WRITE(7)((IA(K,J),K=1,4),J=1,30)
0021             GO TO 4
0022      3      DO 10 K=1,30
0023             DO 10 L=1,4
0024      10     IA(L,K)=0
0025             WRITE(7)((IA(K,J),K=1,4),J=1,30)
0026             CALL PTAPE(7,-1,0)
0027             DO 20 L=1,KFIN
0028             READ(7)((IA(K,J),K=1,4),J=1,30)
0029             WRITE(4)((IA(K,J),K=1,4),J=1,30)
0030      20     CONTINUE
0031             CALL LEADR(4,12)
0032             STOP
0033             END
0034             FND$
```

**END-OF-TAPE

*

FICHE TECHNIQUE DU PROGRAMME

NOM: **TOTO**

TYPE		CHAINE		PERIPH.	IN		
BUT	<i>Listing de toutes les données d'un ruban seule binaire.</i>			OUT			
				FORMAT	IN		
Observations				OUT			
Opérations				ETIQ.	IN		
				OUT			
DATE DE MISE AU POINT				PROGRAMMEUR			
LANGAGE UTILISE							
SSP EMPLOYES							
Starting ADDRESS	2						
Nb Source	Nb Objet		Nb Absolu				
Divers							
Clés employées							

FICHE EXPLOITATION

/E

SYMBOLIC FILE SOURCE DEVICE?

/P

```
0001      PROGRAM TOTO
0002      DIMENSION IA(4,3)
0003      1  WRITE(2,103)
0004      103 FORMAT("ENTREE : ?")
0005      READ(1,*)IN
0006      2  READ(IN)((IA(K,J),K=1,4),J=1,3)
0007      WRITE(2,700)((IA(K,J),K=1,4),J=1,3)
0008      700 FORMAT(4(3I6))
0009      CALL CLE (KC)
0010      IF(KC)3,2,2
0011      3  PAUSE
0012      GOTO 1
0013      END
0014      ENDS
```

**END-OF-TAPE

*

FICHE TECHNIQUE DU PROGRAMME

NOM: TANIA
HYDRO OLGA

TYPE	ACQUISITION	CHAINE	D	PERIPH.	IN	1,5
BUT	Acquisition des donnees P T S O digitalisees et perforees en Format libre			FORMAT	OUT	4,7
Observations				ETIQ.	IN	M
Opérations	Lecture, Perforation ou Ecriture sur BM			ETIQ.	OUT	M
DATE DE MISE AU POINT	AOUT 1974			PROGRAMMEUR		
LANGAGE UTILISE	FORTRAN			MORLIERE		
SSP EMPLOYES	CLE? LEADR,PTAPE					
Starting ADDRESS	22					
Nb Source	I	Nb Objet	I			
Divers						
Clés employées	Cle(x) enhaut arret 999,					

FICHE EXPLOITATION

ETIQUETTES METEO
7701002210677123004000W004000W27226028018074200
P, T, S, C, ... FIN: 9999, 0, 0, 0
0, 2340, 3560, 456
5, 2500, 4600, 456
10, 2300, 3600, 300
9999, 0, 0, 0
SCRTI E EM=1, RP=2
2
ETIQUETTES METEO

```

PROGRAM TANIA
DIMENSION IR(4,6 ),MET(17)
6 WRITE(2,200)
NST=0
WRITE(2,201)
201 FORMAT("IN")
READ(1,*)IN
DO 714 I=1,4
DO 714 J=1,61
714 IR(I,J)=0
200 FORMAT("ETIQUETTES METEO")
READ(IN,102)MET(5),MET(6), (MET(N),N=1,4),AT,IA1,IA2,ON,I01,I02
1, (MET(N),N=7,12)
102 FORMAT(I4,I3,3I2,I4,F4.2,I1,A1,F5.2,I1,A1,3I3,2I2,I4)
103 FORMAT(3I3,I5,I6,I4,I5,I1,A2,I6,I1,A2,3I4,2I3,I5,I6)
94 WRITE(2,95)
IAT=IFIX(AT*100.)
ION=IFIX(ON*100.)
NP=0
J=0
95 FORMAT("P,T,S,0,...FIN:9999,0,0,0")
100 READ(IN,*)IP1,IP2,IX,LC
96 J=J+1
NP=NP+1
IR(1,J)=IP1
98 IR(2,J)=IP2
IR(3,J)=IX
IR(4,J)=LC
IF(IP1-9999)100,80,80
80 WRITE(2,108)
108 FORMAT("SORTIE BM=1,RP=2")
READ(1,*)LC
GOTO(50,70),LC
50 WRITE(2,109)
109 FORMAT("FILE?")
READ(1,*)NF
REWIND 7
NS=7
CALL PTAPE(7,NF,-1)
ENDFILE 7
72 WRITE(NS,103) (MET(N),N=1,6), IAT, IA1, IA2, ION, I01, I02,
1 (MET(N),N=7,12),NP
J1=1
53 J10=J1+9
IF(NS-7)54,702,702
702 WRITE(NS)((IR(I,J),I=1,4),J=J1,J10)
GOTO 703
54 WRITE(NS,412)((IR(I,J),I=1,4),J=J1,J10)
703 J1=J10+1
412 FORMAT(5(3I4,I3))
IF(IR(2,J10))52,52,53
52 IF(NS-4)71,71,105
105 ENDFILE 7
999 WRITE(NS,99)
99 FORMAT(24(1H9))
GOTO 80
71 CALL CLE(IP1)
IF(IP1)6,74,74
74 WRITE(4,99)
CALL LEADR(4,15)
GOTO 6
70 NS=4
NST=NST+1
IF(NST-1)73,73,72
73 CALL LEADR(4,10)
GOTO 72
END

```



```
0001      PROGRAM METEO
0002      5 WRITE(2,105)
0003 105 FORMAT("METEO"/"ENTREE:TTY=1,L.OPT.=5,FIN=3")
0004      READ(1,*)NP
0005      IF(NP-3)5,3,5
0006      5 CONTINUE
0007      WRITE(2,106)
0008 106 FORMAT("EDITION=1,CORR.+ADD.+PERI.O.=2")
0009      READ(1,*)M
0010      GOTO(1,2),M
0011      1 WRITE(2,100)
0012 100 FORMAT(5/)
0013      WRITE(2,103)
0014      4 READ(NP,200)IC,IS,IJ,IM,IA,IH,ILA,LA,ILO,LO,JS,JH,JTS,IS,IS,
0015      IF(IS-999)45,50,50
0016      45 A=FLOAT(ILA)/1000.
0017      O=FLOAT(ILO)/1000.
0018 200 FORMAT(14,13,310,14,15,11,13,11,310,210,14)
0019      WRITE(2,101)IC,IS,IJ,IM,IA,IH,LA,O,LO,JS,JH,JTS,IS,IS,
0020 101 FORMAT(1X"*CRUI.STA. J/ H/AO H..LATITUDE*LONG...*TEMPERATURE
0021      IENT (M/S).FOND"/"
0022      IM.SURF. DIR.VIT.")
0023 101 FORMAT(16,14,13,2(1H/10),15,F7.3,A1,F8.3,A1,14,14,15,4M13,11)
0024      GOTO 4
0025      50 WRITE(2,100)
0026      GOTO 5
0027      2 CALL LEADR(4,15)
0028      22 WRITE(2,107)
0029 107 FORMAT("ST A CORRIGER?,PAS DE CORRECTION=999")
0030      READ(1,*)IS1
0031      7 READ(NP,200)IC,IS,IJ,IM,IA,IH,ILA,LA,ILO,LO,JS,JH,JTS,IS,IS,
0032      M=1
0033      IF(IS-IS1)10,8,10
0034      8 IF(IS1-999)11,15,15
0035      11 WRITE(2,108)
0036      WRITE(2,200)IC,IS,IJ,IM,IA,IH,ILA,LA,ILO,LO,JS,JH,JTS,IS,IS,
0037      READ(1,200)IC,IS,IJ,IM,IA,IH,ILA,LA,ILO,LO,JS,JH,JTS,IS,IS,
0038      M=2
0039      10 WRITE(4,200)IC,IS,IJ,IM,IA,IH,ILA,LA,ILO,LO,JS,JH,JTS,IS,IS,
0040      GOTO(7,22),M
0041 108 FORMAT("CORRIGER LA LIGNE,SUIVANT LE FORMAT:")
0042      15 WRITE(2,109)
0043 109 FORMAT("ADDITION=2 FIN=3")
0044      READ(1,*)IX
0045      IF(IX-2)23,20,999
0046      23 WRITE(2,115)
0047 115 FORMAT("ENTREE TTY=1 LECT.OPT.=5")
0048      READ(1,*)NP
0049      GOTO 22
0050      999 WRITE(4,501)
0051      501 FORMAT(20(1H9))
0052      CALL LEADR(4,15)
0053      3 WRITE(2,110)
0054 110 FORMAT("BONJOUR CHEZ VOUS")
0055      STOP
0056      END
0057      ENDS
```

NOM : METEQ

Programmeur: DARBOIS

But: Confection d'un ruban METEQ à partir
d'un ruban complet (R+M)

```
PROGRAM METEQ
DIMENSION IR(4,10),MET(10)
4 WRITE(2,200)
CALL LEADR(4,10)
PAUSE
2 READ(5,201)JR,MO,IA,IH,IC,IST,(MET(I),I=1,10)
IF(IST-999)3,4,4
3 WRITE(4,202)IC,IST,JR,MO,IA,IH,(MET(I),I=1,10)
1 READ(5,203)((IR(I,J),I=1,4),J=1,10)
IF(IR(2,10))2,2,1
200 FORMAT("IN=5: RP A+M; OUT=4 RP METEQ")
201 FORMAT(3I3,15,16,14,16,A2,17,A2,3I4,2I3,15,16)
202 FORMAT(14,13,3I2,14,15,A1,16,A1,3I3,2I2,14)
203 FORMAT(5(3I4,13))
END
END*
```

Interet . utilisation du programme POINT qui
fonctionne les stations d'une campagne

FICHE TECHNIQUE DU PROGRAMME

NOM: JIVART

TYPE	REDUCTION	CHAINE	A . C	PERIPH.	IN	I,5,
BUT	Reduction des donnees SONDE a moins de 100 pts				OUT	7,2,4
	Calcul des points caracteristiques			FORMAT	IN	.B
Precision : 0.02 c, 0.002, 0.07 ml/l					OUT	A
Observations	Travail en continue les cles en bas			ETIQ.	IN	B
Opérations	Lecture, Filtrage(2), Reduction, Perforation					OUT
DATE DE MISE AU POINT	Avril 1974			PROGRAMMEUR		
LANGAGE UTILISE	FORTRAN			NOEL		
SSP EMPLOYES	CLE ? LEABB? PTAPE					
Starting ADDRESS	2					
Nb Source	I	Nb t	I			
Divers						
Clés employées	toutes cles en haut : arret 99999					

FICHE EXPLICATION

EXPLOITATION DE JIVART

IN, OUT

5, 4

IN, OUT

5, 4

14 1 13 1000 15 6 VP= 74

STOP

IN, OUT

```
PROGRAM JIVAR
DIMENSION IR(4,451),ID(4,10),IV(4)
WRITE(2,2)
2 FORMAT("IN,OUT")
READ(1,*)NU,IOUT
70 IV(2)=2
   IV(3)=2
   IV(4)=7
   IF(IOUT-7)73,75,75
75 REWIND 7
   GO TO 76
73 CALL LEADR(IOUT,12)
```

C
C
C

LECTURE STATION

```
76 IF((NU-7)*(IOUT-7))25,78,25
78 WRITE(2,79)
79 FORMAT("FILE?")
   READ(1,*)IFIL
   CALL PTAPE(7,IFIL,0)
25 READ(NU,100)JR,MO,IA,IH,IC,IST
   IF(IST-9999)101,330,330
100 FORMAT(3I3,I5,I6,I4)
101 N=1
   5 READ(NU)((ID(I,J),I=1,4),J=1,10)
   DO 3 K=1,10
   DO 4 I=1,4
   4 IR(I,N)=ID(I,K)
   3 N=N+1
10 IF(ID(2,10))5,11,5
```

C
C
C

FILTRE

```
11 IF(NU-7)9,8,8
8 CALL PTAPE(7,1,0)
9 DO 50 NF=1,2
   DO 51 J=1,451
   DO 51 I=1,4
   IF(IR(2,J+3))52,50,52
52 IG1=IR(I,J+1)-IR(I,J)
   IG2=IR(I,J+2)-IR(I,J+1)
   IF(IG1*IG2)53,51,51
53 IR(I,J+1)=(IR(I,J)+IR(I,J+2))/2
51 CONTINUE
50 CONTINUE
```

C
C
C

REDUCTION

```
12 DO 700 NK=2,4
   L=1
400 IF(IR(2,L+1))700,700,998
998 LP=L+2
   DO 500 K=LP,451
   IF(IR(2,K))900,900,501
501 L1=L+1
   LD=K-1
```



```

DO 500 M=L1,LD
IF(IR(1,LD)) 503, 200, 200
200 IEM=IABS(IR(NK,M)-IR(NK,M-1))
IF(IEM-IV(NK)) 510, 511, 511
510 IEM=IV(NK)
511 IVCM=IR(NK,L)+(IR(NK,K)-IR(NK,L))*(IABS(IR(1,M))-
1IABS(IR(1,L)))/(IABS(IR(1,K))-IABS(IR(1,L)))
IEC=IABS(IR(NK,M)-IVCM)
IF(IEC-IEM) 500, 500, 503
500 CONTINUE
503 IR(1,LD)=-IABS(IR(1,LD))
L=K-1
GO TO 400
900 IR(1,K-1)=-IABS(IR(1,K-1))
700 CONTINUE

```

C
C NB DE POINTS
C

```

NP=0
DO 250 J=1,451
IF(IR(2,J)) 275, 276, 275
275 IF(IR(1,J)) 251, 251, 250
251 NP=NP+1
250 CONTINUE

```

C
C CHOIX
C
276 WRITE(2,976)JR,MO,IA,IH,IC,IST,NP
976 FORMAT(3I3,I5,I6,I4," NP=",I4)
IF(NP-100)991,991,990
990 IV(2)=IV(2)+1
IV(3)=IV(3)+1
DO 992 J=1,451
992 IR(1,J)=IABS(IR(1,J))
GO TO 12

C
C SERRAGE - METEO
C

```

991 WRITE(IOUT,100)JR,MO,IA,IH,IC,IST
830 NP=1
320 DO 300 I=1,4
DO 300 J=1,10
300 ID(I,J)=0
DO 301 J=1,10
IF(IR(2,NP)) 306, 310, 306
306 IF(IR(1,NP)) 303, 303, 307
307 NP=NP+1
GO TO 306
303 DO 305 I=1,4
305 ID(I,J)=IR(I,NP)
ID(1,J)=IABS(ID(1,J))
301 NP=NP+1

```

C
C

PERFORATI ON

```
310 IF(I OUT-7)800,888,888
888 WRITE(7)((ID(I,J),I=1,4),J=1,10)
GO TO 802
800 WRITE(I OUT,777)((ID(I,J),I=1,4),J=1,10)
777 FORMAT(5(3I4,I3))
802 IF(ID(2,10))320,321,320
```

C
C
C

RETOUR

```
321 IF(I OUT-7)333,334,334
334 END FILE 7
GO TO 333
330 PAUSE
333 CALL CLE(KLE)
IF(KLE)30,25,30
30 WRITE(I OUT,31)
31 FORMAT(70(1H9))
IF(I OUT-7)340,341,341
340 CALL LEADR(I OUT,15)
STOP
341 REWIND7
END
END$
```

FICHE TECHNIQUE DU PROGRAMME

NOM: HELPR

TYPE	CORRECTION	CHAINE	B	PERIPH.	IN	5,7
BUT	Correction des donnees SONDE Reduites				OUT	4,7
				FORMAT	IN	A
					OUT	X
Observations				ETIQ.	IN	M
Opérations	GLOBAL ETIQ CONTROL OUT NIVEAU				OUT	M
DATE DE MISE AU POINT	SEPTEMBRE 1974			PROGRAMMEUR		
LANGAGE UTILISE	FORTRAN			MORLIERE		
SSP EMPLOYES	CLE LEADR PTAPE					
Starting ADDRESS	2					
Nb Source	Nb Objet		Nb Absolu			
Divers						
Clés employées	cle(n) en haut arret 9999					

FICHE EXPLOITATION

INPUT EM=1 RP=2

2

31 5 76 1830 7606 24 1000 N 5000 E 249 229 271 16 9 3750 70

GLCEAL=1 ETIQ=2 CONTROL=3 OUT=4 /NIVEAU=5

1

CCR. X=A*T+E*X+C*P+D ENTRE P1, P2

P1, P2, X, A, E, C, D (X=1, 2, 3, 4 POUR PTSO)

0, 500, 4, 0, 1, 0, 0

CCR. X=A*T+E*X+C*P+D ENTRE P1, P2

P1, P2, X, A, E, C, D (X=1, 2, 3, 4 POUR PTSO)

0, 0, 0, 0

IMMERSION?

9999

EM=1 RP=2 COR=3 CONT=4 /NIV=5

5

NIVEAU A CORRIGER

0

P, T, S, C

0, 2756, 3567, 456

NIVEAU A CORRIGER

9999

P, T, S, C

0, 0, 0, 0, 0

EM=1 RP=2 COR=3 CONT=4 /NIV=5

2

INPUT EM=1 RP=2

Note: la Serie **EXIT9**, **HELPR9**
PLOT9 travaille sur des données
(RP binaires) non réduites.
la Serie **EXITR**, **HELPR**
PLOTR travaille sur des données
réduites (RP ASCII)

```
0001      PROGRAM HELPR
0002      DIMENSION IR(4,110),MET(19)
0003      NST=0
0004      600 J1=1
0005      WRITE(2,200)
0006      200 FORMAT(/"INPUT BM=1 RP=2"/)
0007      READ(1,*)LC
0008      GOTO(201,202),LC
0009      202 NE=5
0010      GOTO 204
0011      201 WRITE(2,100)
0012      NE=7
0013      100 FORMAT(/"STATION?"/)
0014      READ(1,*)IS0
0015      1 CALL PTAPE(7,1,0)
0016      READ(7,110)(MET(N),N=1,19)
0017      110 FORMAT(3I3,15,I6,14,15,11,1XA1,I6,11,1XA1,3I4,2I3,15,I6)
0018      IF(IS0-MET(6))1,2,1
0019      2 WRITE(2,700)
0020      700 FORMAT(/"ATTENTION"/"ANNULATION STATION=546 SINON=2"/)
0021      READ(1,*)I
0022      IF(I-546)502,501,502
0023      501 CALL PTAPE(7,0,-1)
0024      WRITE(7,503)
0025      503 FORMAT(120(1H0))
0026      GOTO 600
0027      502 CALL PTAPE(7,0,-1)
0028      204 READ(NE,110)(MET(N),N=1,19)
0029      402 J10=J1+9
0030      IF(NE-7)800,801,801
0031      801 READ(NE)((IR(I,J),I=1,4),J=J1,J10)
0032      GOTO 802
0033      800 READ(NE,999)((IR(I,J),I=1,4),J=J1,J10)
0034      999 FORMAT(5(3I4,I3))
0035      802 J1=J10+1
0036      IF(IR(2,J10))402,34,402
0037      34 WRITE(2,110)(MET(N),N=1,19)
0038      WRITE(2,102)
0039      102 FORMAT(/"GLOBAL=1 ETIQ=2 CONTROL=3 OUT=4 /NIVEAU=5"/)
0040      READ(1,*)LC
0041      GOTO(13,6,20,80,94),LC
0042      6 WRITE(2,103)
0043      103 FORMAT("METEO")
0044      READ(1,810)MET(5),MET(6),(MET(I),I=1,4),(MET(I),I=7,18)
0045      810 FORMAT(I4,I3,3I2,2I4,11,A1,I5,11,A1,3I3,2I2,I4)
0046      GOTO 34
0047      13 WRITE(2,104)
0048      104 FORMAT("COR. X=A*T+B*X+C*P+D ENTRE P1,P2"/"P1,P2,X,A,B,C,D (
0049      13,4 POUR PTS0)")
0050      READ(1,*)IP1,IP2,IX,A,B,C,D
0051      IF(IX)20,20,22
0052      22 J=0
0053      11 J=J+1
0054      IF(J-J10)14,14,13
0055      14 IF(IP1-IR(1,J))11,12,11
0056      12 IR(IX,J)=IFIX(A*FLOAT(IR(2,J))+B*FLOAT(IR(IX,J))+C*FLOAT(IR(
0057      1D)
0058      J=J+1
0059      IF(J-J10)16,16,13
```

```
0060      16 IF(IP2-IR(1,J))13,12,12
0061      20 WRITE(2,106)
0062     106 FORMAT(/"IMMERSION?")
0063      READ(1,*)IM
0064      DO 30 J=1,J10
0065      IF(IR(1,J)-IM)30,40,30
0066      30 CONTINUE
0067      GOTO 80
0068      94 WRITE(2,612)
0069     612 FORMAT("NIVEAU A CORRIGER")
0070      READ(1,*)IP1
0071      WRITE(2,95)
0072      95 FORMAT("P,T,S,0")
0073      READ(1,*)IP0,IP2,IX,LC
0074      IF(IP1-9999)97,80,80
0075      97 J=0
0076      96 J=J+1
0077      IF(IP1-IR(1,J))80,98,96
0078      98 IR(2,J)=IP2
0079      IR(1,J)=IP0
0080      IR(3,J)=IX
0081      IR(4,J)=LC
0082      GOTO 94
0083      80 WRITE(2,108)
0084     108 FORMAT(/"BM=1 RP=2 COR=3 CONT=4 /NIV=5")
0085      READ(1,*)LC
0086      GOTO(50,70,13,20,94),LC
0087      40 KF=J+9
0088      WRITE(2,107)((IR(I,K),K=J,KF),I=1,4)
0089     107 FORMAT(10I5)
0090      GOTO 20
0091      50 WRITE(2,109)
0092     109 FORMAT("FILE?")
0093      READ(1,*)NF
0094      REWIND 7
0095      NS=7
0096      CALL PTAPE(7,NF,-1)
0097      ENDFILE 7
0098      72 WRITE(NS,110)(MET(N),N=1,19)
0099      J1=1
0100      53 J10=J1+9
0101      IF(NS-7)54,803,803
0102     803 WRITE(NS)((IR(I,J),I=1,4),J=J1,J10)
0103      GOTO 804
0104      54 WRITE(NS,999)((IR(I,J),I=1,4),J=J1,J10)
0105     804 J1=J10+1
0106      IF(IR(2,J10))52,52,53
0107      52 IF(NS-4)71,71,105
0108     105 ENDFILE 7
0109      WRITE(NS,99)
0110      99 FORMAT(72(1H9))
0111      GOTO 80
0112      71 CALL CLE(IP1)
0113      IF(IP1)600,74,74
0114      74 WRITE(4,99)
0115      CALL LEADR(4,15)
0116      GOTO 600
0117      70 NS=4
0118      NST=NST+1
0119      IF(NST-1)73,73,72
```

FICHE TECHNIQUE DU PROGRAMME

NOM: GGGG

TYPE	CONTROLE	CHAINE	B	PERIPH.	IN	5,7		
BUT	Trace 30 profils T, S ou O2 sur une même feuille			OUT	P			
				FORMAT	IN			
Observations				OUT	P	L		
Opérations				ETIQ.	IN	S/M		
				OUT		L		
DATE DE MISE AU POINT	AVRIL 1974			PROGRAMMEUR				
LANGAGE UTILISE	fortran			NOEL				
SSP EMPLOYES	PTAPE							
Starting ADDRESS	2							
Nb Source	I	Nb Objet	I				Nb Absolu	I
Divers								
Clés employées	NON							

FICHE EXPLOITATION

GGC

IN ?

5

T=2, S=3, C2=4

2

P1000=1, P500=2

2

31 5 76 1830 7606 24

FL TL

2826 9999

2826 9739

282

```
PROGRAM GOGO
DIMENSION ID(4,110),MIN(4),MAX(4)
50 WRITE(2,1)
  1 FORMAT(/"GOGO"//"IN ?")
  READ(1,*)IN
  IF(IN-7)100,101,101
101 WRITE(2,102)
102 FORMAT("FILE")
  READ(1,*)IFIL
  CALL PTAPE(7,IFIL,0)
100 WRITE(2,10)
  10 FORMAT("T=2,S=3,02=4")
  READ(1,*)NC
  N=1
  WRITE(2,11)
  11 FORMAT("P1000=1,P500=2")
  READ(1,*)LP
  40 READ(IN,2)JR,MO,IA,IH,IC,IST
  2 FORMAT(3I3,I5,I6,I4)
  WRITE(2,2)JR,MO,IA,IH,IC,IST
  IF(IST-9999)7,41,41
  7 J1=1
  5 J10=J1+9
  IF(IN-7)110,111,111
110 READ(IN,112)((ID(I,J),I=1,4),J=J1,J10)
112 FORMAT(5(3I4,I3))
  GO TO 113
111 READ(7)((ID(I,J),I=1,4),J=J1,J10)
113 IF(ID(2,J10))3,4,3
  3 J1=J10+1
  GO TO 5
  4 MIN(2)=600-N*200
  MAX(2)=8800-N*200
  MIN(3)=3470-N*20
  MAX(3)=4230-N*20
  MIN(4)=150-N*50
  MAX(4)=2300-N*50
  WRITE(2,20)
  20 FORMAT("PLTL")
  DO 30 J=1,110
  IF(ID(2,J))21,22,21
  21 IX=FIX(9999./FLOAT(MAX(NC)-MIN(NC))*FLOAT(ID(NC,J)-
  1MIN(NC)))
  IY=9999-ID(1,J)*10*LP
  IF(IY-10000)120,22,22
120 WRITE(2,30)IX,IY
  30 FORMAT(2I5)
  22 WRITE(2,23)
  23 FORMAT("PLTT")
  N=N+1
  IF(IN-7)150,151,151
151 CALL PTAPE(7,1,0)
150 IF(N-31)40,41,41
  41 PAUSE
  N=1
  GO TO 40
  END
  ENDS
```

FICHE TECHNIQUE DU PROGRAMME

NOM: TRAD

TYPE	TRANSFERT	CHAINE	A	C	PERIPH.	IN	5,7
BUT	Assemblage et Transfert des stations binaires + Traduction ASCII-BINAIRE				OUT		4,7
Observations					FORMAT	IN	A,B
Opérations					OUT		B
					ETIQ.	IN	S
					OUT		S
DATE DE MISE AU POINT	AVRIL 1973				PROGRAMMEUR		
LANGAGE UTILISE	FORTRAN				NOEL MORLIERE		
SSP EMPLOYES	CLE	PTAPE	LEADR				
Starting ADDRESS	2						
Nb Source	I	Nb Objet	I	Nb Absolu			
Divers							
Clés employées	<i>travail en continu lever la clé 15</i>						

FICHE EXPLOITATION

TRAD

```
PROGRAM TRAD
DIMENSION ID(4,481)
CALL LEADR(4,15)
WRITE(2,100)
100 FORMAT(//"TRADUCTION=1,ASSEMBLAGE=2")
READ(1,*)NW
WRITE(2,110)
110 FORMAT("IN,OUT")
READ(1,*)NE,NS
IF(NE-7)26,400,400
400 WRITE(2,401)
401 FORMAT("FILE")
READ(1,*)IF
REWIND 7
CALL PTAPE(7,IF,0)
GOTO 25
26 IF(NS-7)25,400,400
25 READ(NE,102)JR,MO,IA,IH,IC,IS
WRITE(2,102)JR,MO,IA,IH,IC,IS
102 FORMAT(3I3,15,16,14)
IF(IS-999)150,160,160
150 IF(NW-1)200,200,300
160 PAUSE
IF(NE-7)700,600,600
700 CALL CLE(IS)
GOTO 500
200 J1=1
2 J10=J1+9
READ(5,105)((ID(I,J),I=1,4),J=J1,J10)
105 FORMAT(5(3I4,13))
J1=J10+1
IF(ID(2,J10))10,10,2
300 J1=1
301 J10=J1+9
READ(NE)((ID(I,J),I=1,4),J=J1,J10)
J1=J10+1
IF(ID(2,J10))10,10,301
10 IF(NE-7)800,801,801
801 CALL PTAPE(7,1,0)
800 WRITE(NS,102)JR,MO,IA,IH,IC,IS
J1=1
7 J10=J1+9
WRITE(NS)((ID(I,J),I=1,4),J=J1,J10)
J1=J10+1
IF(ID(2,J10))20,20,7
20 CALL CLE(IS)
IF(NS-7)500,501,501
531 ENDFILE 7
500 IF(IS)25,30,30
30 IF(NS-7)600,601,601
600 WRITE(4,155)
155 FORMAT(24(IH9))
CALL LEADR(4,15)
STOP
601 WRITE(7,155)
END
ENDS
```

FICHE TECHNIQUE DU PROGRAMME

NOM: AMORS

TYPE	CHAINE			PERIPH.	IN	X
BUT	<i>Initialisation des bandes magnetiques pour l'acquisition des données sondes</i>			OUT		7
Observations				FORMAT	IN	
Opérations				OUT		
DATE DE MISE AU POINT				ETIQ.	IN	
LANGAGE UTILISE	FORTRAN			OUT		
SSP EMPLOYES				PROGRAMMEUR		
Starting ADDRESS				<i>Morlière</i>		
Nb Source	Nb Objet		Nb Absolu			
Divers						
Clés employées						

FICHE EXPLOITATION

FICHE TECHNIQUE DU PROGRAMME

NOM: BANAL

TYPE	ASSEMBLAGE	CHAINE		A,B,C	PERIPH.	IN	7
BUT	<i>donne le catalogue des stations enregistrées sur BM</i>			OUT		2	
Observations				FORMAT	IN	B	
Opérations				OUT		L	
DATE DE MISE AU POINT				ETIQ.	IN	S	
LANGAGE UTILISE	FORTRAN			OUT		S	
SSP EMPLOYES				PROGRAMMEUR			
Starting ADDRESS	2			<i>MORLIÈRE</i>			
Nb Source	Nb Objet		Nb Absolu				
Divers							
Clés employées							

```

0001      PROGRAM AMORS
0002      1 WRITE(2,200)
0003      WRITE(7,100)
0004      END FILE 7
0005      WRITE(7,101)
0006      REVIND 7
0007      PAUSE
0008      GO TO 1
0009      200 FORMAT(/"INITIALISATION D.M.")
0010      100 FORMAT(123(1H*))
0011      101 FORMAT(123(1H9))
0012      END
0013      END$
0014      **END-OF-TAPE
0015

```

BANAL

```

*
0016      PROGRAM BANAL
0017      REVIND 7
0018      WRITE (2,100)
0019      100 FORMAT (/ "PROGRAM BANAL" / "FILE DE DEBUT ?")
0020      READ (1,*)NF
0021      CALL PTAPE(7,NF,0)
0022      WRITE (2,102)
0023      102 FORMAT (5/4HFILE,3X3HCRO,3X,2HST,5X,4HDATE,4X,5HHEURE/)
0024      10 READ (7,101)JR,MO,IA,IH,IC,IS
0025      101 FORMAT(3I3,15,16,14)
0026      IF(JR-99)30,20,20
0027      30 WRITE (2,103)NF,IC,IS,JR,MO,IA,IH
0028      103 FORMAT (14,2X,2I5,3X2(12,1H/),12,3X14)
0029      CALL PTAPE (7,1,0)
0030      NF=NF+1
0031      GO TO 10
0032      20 REVIND 7
0033      WRITE (2,105)
0034      105 FORMAT (10/)
0035      END
0036      END$

```

FICHE TECHNIQUE DU PROGRAMME

NOM: RED

TYPE	ASSEMBLAGE	CHAINE	B	PERIPH.	IN	5,7
BUT	Assemblage et Transfert des stations reduites avec ou sans meteo + Addition METEO			PERIPH.	OUT	2,4,7
				FORMAT	IN	A
Observations					FORMAT	OUT
				Opérations		ETIQ.
			OUT		S ou M	
DATE DE MISE AU POINT	AVRIL 1974			PROGRAMMEUR		
LANGAGE UTILISE	FORTRAN			NOEL		
SSP EMPLOYES	CLE LEADR PTAPE					
Starting ADDRESS	2					
Nb Source	I	Nb Objet	I			
Divers						
Clés employées	clef(n) en haut arret 9999					

FICHE EXPLOITATION

TRANSFERT= 0, AD. METEO= 1
0

IN, OUT?
5,4
5,7
METEO 0 OU 1
1
FILE?
1

```
PROGRAM RED
DIMENSION IR(4,110),MAT(17)
25 WRITE(2,20)
20 FORMAT(//"TRANSFERT= 0,AD. METEO=1")
  READ(1,*)IPRO
  IF(IPRO)21,21,22
22 WRITE(2,600)
600 FORMAT("AD. MET"//"DEBUT=1  SUITE=2  FIN =3")
  READ(1,*)ICON
  IF(ICON-3)460,461,461
460 WRITE(2,400)
400 FORMAT("IN, OUT?")
  READ(1,*)IN,IOUT
  IF((IN-7)*(IOUT-7))401,402,401
402 REWIND 7
  WRITE(2,403)
403 FORMAT("FILE /")
  READ(1,*)IFIL
  CALL PTAPE(7,IFIL,0)
401 IF(ICON-2)450,631,461
450 IF(IOUT-7)451,631,631
451 CALL LEADR(IOUT,15)
631 READ(IN,650)JR,MO,IA,IH,IC,IST
650 FORMAT(3I3,I5,I6,I4)
  IF(IST-9999)601,630,630
601 READ(1,603)(MAT(I),I=1,16)
603 FORMAT(I4,I3,3I2,I4,I5,A1,I6,A1,3I3,2I2,I4)
  IF(MAT(2)-IST)601,605,601
605 J1=1
606 J10=J1+9
  IF(IN-7)404,405,405
404 READ(IN,130)((IR(I,J),I=1,4),J=J1,J10)
  GO TO 406
405 READ(7)((IR(I,J),I=1,4),J=J1,J10)
406 IF(IR(2,J10))607,610,607
607 J1=J10+1
  GO TO 606
610 NP=0
  DO 611 J=1,110
    IF(IR(2,J))611,620,611
611 NP=NP+1
620 WRITE(IOUT,625)JR,MO,IA,IH,IC,IST,(MAT(I),I=7,16),NP
625 FORMAT(3I3,I5,I6,I4,I6,A2,I7,A2,3I4,2I3,I5,I6)
  J1=1
640 J10=J1+9
  IF(IOUT-7)410,411,411
411 WRITE(7)((IR(I,J),I=1,4),J=J1,J10)
  GO TO 412
410 WRITE(IOUT,130)((IR(I,J),I=1,4),J=J1,J10)
412 IF(IR(2,J10))641,642,641
641 J1=J10+1
  GO TO 640
```

```
        642 IF(IN-7)420,421,421
421 CALL PTAPE(7,1,0)
      GO TO 630
420 IF(IOUT-7)630,423,423
423 FND FILE 7
630 CALL CLF(KLE)
      IF(KLE)22,631,22
461 WRITE(IOUT,632)
      IF(IOUT-7)462,463,463
462 CALL LEADR(IOUT,15)
      GO TO 25
463 REWIND 7
      GO TO 25
 21 WRITE(2,1)
    1 FORMAT(//"IN, OUT?")
      READ(1,*)IN,IOUT
      WRITE(2,2)
    2 FORMAT("METEO 0 OU 1")
      READ(1,*)MET
      IF((IN-7)*(IOUT-7))10,11,10
 11 WRITE(2,12)
 12 FORMAT("FILE?")
      READ(1,*)IFIL
      CALL PTAPE(7,IFIL,0)
      IF(IOUT-7)10,100,100
 10 CALL LEADR(IOUT,15)
100 CALL CLF(KLE)
      IF(KLE)900,101,900
101 IF(MET)103,120,103
103 READ(IN,625)(MAT(I),I=1,17)
      GO TO 111
120 READ(IN,650)(MAT(I),I=1,6)
111 IF(MAT(6)-9999)112,113,112
113 PAUSE
      GO TO 100
112 J1=1
132 J10=J1+9
      IF(IN-6)133,133,134
133 READ(IN,130)((IR(I,J),I=1,4),J=J1,J10)
130 FORMAT(5(3I4,I3))
      GO TO 140
134 READ(7)((IR(I,J),I=1,4),J=J1,J10)
140 IF(IR(2,J10))131,500,131
131 J1=J10+1
      GO TO 132
500 IF(MET)501,502,501
501 WRITE(IOUT,625)(MAT(I),I=1,17)
      IF(IOUT-4)505,30,30
 30 WRITE(2,625)(MAT(I),I=1,17)
      GO TO 505
502 WRITE(IOUT,650)(MAT(I),I=1,6)
      IF(IOUT-4)505,31,31
 31 WRITE(2,650)(MAT(I),I=1,6)
505 J1=1
542 J10=J1+9
      IF(IOUT-5)510,510,511
510 WRITE(IOUT,130)((IR(I,J),I=1,4),J=J1,J10)
      GO TO 540
```

```
511 WRITE(7)((IR(I,J),I=1,4),J=J1,J10)
540 IF(IR(2,J10))541,550,541
541 J1=J10+1
    GO TO 542
550 IF(IOUT-6)440,440,551
440 IF(IN-7)100,441,441
441 CALL PTAPE(7,1,0)
    GO TO 100
551 END FILE 7
    GO TO 100
900 IF(MET)901,902,901
901 WRITE(IOUT,632)
632 FORMAT(70(1H9))
    GO TO 920
902 WRITE(IOUT,911)
911 FORMAT(24(1H9))
920 IF(IOUT-6)930,930,931
930 CALL LEADR(IOUT,15)
    GO TO 25
931 REWIND 7
    GO TO 25
    END
    ENDS
```

FICHE TECHNIQUE DU PROGRAMME

NOM: PLOTR

TYPE	TRACES	CHAINE	B	PERIPH.	IN	5,7		
BUT	Trace sur le plotter les profils P T S O , TS, T02, σ_c , $\Delta_s r$			OUT	P			
				FORMAT	IN	A		
Observations				OUT	P			
Opérations				ETIQ.	IN	M/S		
				OUT	L			
DATE DE MISE AU POINT	OCTOBRE 1974/mod. Aout 75			PROGRAMMEUR				
LANGAGE UTILISE	FORTRAN			MORLIERE				
SSP EMPLOYES	CLE LEADR ØTAPE							
Starting ADDRESS	2							
Nb Source	I	Nb Objet	I				Nb Absolu	I
Divers	clef(n) en haut arret							
Clés employées	clef(15) en haut PLTP (par points) en bas PLTL (par lignes)							

FICHE EXPLOITATION

PLCTR
 E=1 RP=2
 2
 SCRTIE PLCTR=2 RP=4
 2
 0-380=1 0-760=2 0-1520=3 CHCIX=4
 4
 PRCF, MIN, MAX
 0, 333
 31 5 76 1830 7606 24
 TS=1 T=2 S=3 O=4 TC=5 SIG=6 DEL=7 MCL=8 FIN=9
 3
 PLTL


```
PROGRAM PLOTR
  DIMENSION ID(4,110),MIN(4),MAX(4)
915 WRITE(2,100)
100 FORMAT("PLOTR"/"BM=1  RP=2")
  READ(1,*)NE
  WRITE(2,458)
458 FORMAT("SORTIE PLOTR=2  RP=4")
  READ(1,*)N12
  GOTO(917,36),NE
36 NE=5
  READ(5,601)JR,MO,IA,IH,IC,IS
  GOTO 911
917 WRITE(2,37)
  37 FORMAT("STATION?")
  READ(1,*)ISO
904 CALL PTAPE(7,1,3)
  READ(7,601)JR,MO,IA,IH,IC,IS
  NE=7
  IF(ISO-IS)904,911,904
911 J1=1
912 J10=J1+7
  IF(NE-7)110,111,111
110 READ(NE,112)((ID(I,J),I=1,4),J=J1,J10)
112 FORMAT(5(314,13))
  GOTO 113
111 READ(NE)((ID(I,J),I=1,4),J=J1,J10)
113 J1=J10+1
  IF(ID(2,J10))912,60,912
  60 WRITE(2,200)
200 FORMAT("0-380=1 0-760=2 0-1520=3  CHUIX=4")
  ZM=0.
  READ(1,*)IZ
  GOTO(201,202,303,304),IZ
303 Z=1520.
  GOTO 203
201 Z=380.
  GOTO 203
304 WRITE(2,800)
300 FORMAT("PROF, MIN,MAX")
  READ(1,*)ZM,Z
  GOTO 203
202 Z=760.
203 MIN(2)=400
  MAX(2)=2900
  MIN(3)=3200
  MAX(3)=3700
  MIN(4)=0
  MAX(4)=1250
  31 WRITE(2,102)JR,MO,IA,IH,IC,IS
102 FORMAT(313,15,216/"TS=1 T=2 S=3 O=4 TO=5 SIG=6 DEL=7 MOD=8
  IFIN=9")
  READ(1,*)IG
  IF(IG-8)900,901,915
901 WRITE(2,902)
902 FORMAT("PARA,MIN,MAX")
  READ(1,*)IG,MIN(IG),MAX(IG)
900 CALL CLE(18)
  IF(I8)401,400,401
401 WRITE(N12,402)
```

```
402-FORMAT("PLTP")
      GOTO 403
400-WRITE(N12,105)
105-FORMAT("PLTL")
601-FORMAT(3I3,15,16,14)
403-DO 10 K=1,J10
      CALL CLE(I8)
      IF(I8)50,30,30
30 IF(ID(2,K))50,50,11
11 GOTO(21,12,12,12,25,610,610,915),IG
12 IY=IFIX(9999.*FLOAT(ID(IG,K)-MIN(IG))/FLOAT(MAX(IG)-MIN(IG)))
      GOTO 16
610 T=FLOAT(ID(2,K))/100.
      S=FLOAT(ID(3,K))/100.
      IF(IG-6)915,611,612
611 IY=IFIX(9999.*(SIGMA(T,S)-21.75)/6.25)
E-0010: 0611 +0000
      GOTO 16
612 A=SIGMA(T,S)*1.E-3
      A=(.02736-A/(1.+A))*1.E+5
      IY=IFIX(9999.*A/500.)
16 IX=IFIX(9999.*(FLOAT(ID(1,K))-ZM)/(Z-ZM))
      IF(ID(1,K)-IFIX(Z))15,15,50
21 IY=IFIX(9999.*FLOAT(ID(2,K)-400)/2500.)
      IX=IFIX(9999.*FLOAT(ID(3,K)-3300)/380.)
      GOTO 15
25 IY=IFIX(9999.*FLOAT(ID(2,K)-400)/2500.)
      IX=IFIX(9999.*FLOAT(ID(4,K))/950.)
15 IF(K-1)700,700,701
701 WRITE(N12,104)IX,IY
      GOTO 10
700 WRITE(N12,107)IX,IY
107 FORMAT(14,15,1H)
104 FORMAT(14,15)
10 CONTINUE
50 WRITE(N12,106)
106 FORMAT("PLTT")
      IF(N12-4)203,457,203
457 CALL LEADR(4,15)
      GOTO 203
      END
      ENDS
```

FICHE TECHNIQUE DU PROGRAMME

NOM: DELTA

TYPE		CHAINE		PERIPH.	IN	5,7
BUT	<i>Trace de Δst et de βt en fonction de la profondeur - Mêmes caractéristiques que PLOT</i>			OUT	P	
				FORMAT	IN	B
				OUT	P	
Observations				ETIQ.	IN	S
Opérations					OUT	S
DATE DE MISE AU POINT				PROGRAMMEUR		
LANGAGE UTILISE	<i>FORTRAN</i>			<i>Moliere</i>		
SSP EMPLOYES	<i>sigma - cléf</i>					
Starting ADDRESS	<i>2</i>					
Nb Source	Nb Objet		Nb Absolu			
Divers						
Clés employées	<i>15 (levée = Trace par point) sinon ligne</i>					

FICHE EXPLOITATION

DELTA

```
PROGRAM DELTA
DIMENSION ID(4,481),MIN(4),MAX(4)
915 WRITE(2,100)
100 FORMAT("DELTA"//"BM=1  RP=2")
READ(1,*)NE
GOTO(917,36),NE
36 NE=5
READ(5,601)JR,MO,IA,IH,IC,IS
GOTO 911
917 WRITE(2,37)
37 FORMAT("STATION?")
READ(1,*)ISO
904 CALL PTAPE(7,1,0)
READ(7,601)JR,MO,IA,IH,IC,IS
NE=7
IF(ISO-IS)904,911,904
911 J1=1
912 J10=J1+9
READ(NE)((ID(I,J)),I=1,4),J=J1,J10)
J1=J10+1
IF(ID(2,J10))912,60,912
60 WRITE(2,200)
200 FORMAT("0-380=1 0-760=2 0-1520=3 CHOIX=4")
ZM=0.
READ(1,*)IZ
GOTO(201,202,303,304),IZ
303 Z=1520.
GOTO 203
201 Z=380.
GOTO 203
304 WRITE(2,800)
800 FORMAT("PROF, MIN,MAX")
READ(1,*)ZM,Z
GOTO 203
202 Z=760.
203 CONTINUE
31 WRITE(2,102)JR,MO,IA,IH,IC,IS
102 FORMAT(3I3,15,16,14/"SIGMA=1 DELTA=2")
READ(1,*)IG
900 CALL CLE(18)
IF(18)401,400,401
401 WRITE(2,402)
402 FORMAT("PLTP")
GOTO 403
400 WRITE(2,105)
105 FORMAT("PLTL")
601 FORMAT(3I3,15,16,14)
403 DO 10 K=1,J10
P=FLOAT(ID(1,K))
T=FLOAT(ID(2,K))/100.
S=FLOAT(ID(3,K))/100.
CALL CLE(18)
IF(18)50,30,30
30 IF(ID(2,K))50,50,11
11 GOTO(21,12,50),IG
21 IY=IFIX(9999.*(SIGMA(T,S)-21.75)/6.25)
GOTO 16
12 A=SIGMA(T,S)*1.E-3
A=(.02736-A/(1.+A))*1.E+5
```

```
      IY=IFIX(9999.*A/500.)
16  IX=IFIX(9999.*(P-ZM)/(Z-ZM))
      IF(K-1)700,700,701
701  WRITE(2,104)IX,IY
      GOTO 10
700  WRITE(2,107)IX,IY
107  FORMAT(I4,I5,1H*)
104  FORMAT(I4,I5)
10  CONTINUE
50  WRITE(2,106)
106  FORMAT("PLTT")
      GOTO 203
      END
      ENDS
```

FICHE TECHNIQUE DU PROGRAMME

NOM: TS02

TYPE	CHAINE			PERIPH.	IN	7
BUT	Tracés groupés pour une campagne des diagrammes TS - J02 - S02.			PERIPH.	OUT	13
Observations				FORMAT	IN	13
Opérations				FORMAT	OUT	13
				ETIQ.	IN	5
				ETIQ.	OUT	6
DATE DE MISE AU POINT				PROGRAMMEUR		
LANGAGE UTILISE	FORTRAN			Malière.		
SSP EMPLOYES	PTAPE					
Starting ADDRESS	2					
Nb Source	Nb Objet		Nb Absolu			
Divers						
Clés employées	10					

FICHE EXPLOITATION

"STATISTICS BORNES, SORTIE"
 "S1, S2, NS"
 1, 5, 20
 "TS=1 TC2=2 CS=3"
 1
 "PLTT"

PROGRAM TS02

```
      DIMENSION ID(4,10),MASTR(6)
100 FORMAT("PLTR")
105 FORMAT("TS=1  T02=0  023=3")
103 FORMAT(14,15)
104 FORMAT("PLTT")
101 FORMAT(3I3,15,15,14)
102 FORMAT("STATIONS BORNES, SORTIE"/"S1, S2, NS")
10  WRITE(2,100)
    READ(1,*)I51,I52,NS
    WRITE(2,105)
    READ(1,*)IC
10  CALL RTAPE(7,1,0)

    READ(7,101)MASTR
    IF(MASTR(6)-I51)1,2,1
20  WRITE(NS,102)
30  READ(7)((ID(I,J)),I=1,4),J=1,10)
    DO 3 I=1,10
      IF(ID(2,I))4,4,5
50  IT=IFIX(9999.*FLOAT(ID(2,I))-400)/2500.)
      IS=IFIX(9999.*FLOAT(ID(3,I))-3300)/380.)
      IO=IFIX(9999.*FLOAT(ID(4,I))/1250.)

20  IX=IS
      IX=IT
      GOTO 30
22  IX=IFIX(9999.*FLOAT(ID(4,I))/950.)
      IX=IT
      GOTO 30
23  IX=IS
      IX=IO
30  WRITE(NS,103)IX,IV
3  CONTINUE
  GOTO 6
4  WRITE(NS,104)
    WRITE(NS,101)MASTR
    WRITE(NS,102)
    CALL RTAPE(7,1,0)
    READ(7,101)MASTR
    IF(MASTR(6)-I52)5,6,7
7  WRITE(NE,104)
    PAUSE
    REWIND 7
    GOTO 10
END
ENFS
```

FICHE TECHNIQUE DU PROGRAMME

NOM: **point** POINT

TYPE	TRACE	CHAINE	B	PERIPH.	IN	17
BUT	trace sur le plotter le plan de la campagne à partir du tableau Meteo, BT, RP (RED + NET)			FORMAT	IN	17
Observations				ETIQ.	IN	17
Opérations					OUT	17
DATE DE MISE AU POINT	AVRIL 1974, modifié Dec 74			PROGRAMMEUR		
LANGAGE UTILISE	FORTRAN			DARBOIS.		
SSP EMPLOYES						
Starting ADDRESS	2025(obj), 2					
Nb Source	Nb Objet		Nb Absolu			
Divers						
Clés employées						

FICHE EXPLOITATION

PCINT
 LAT MIN-MAX(D) N+ S-
 +10, -10
 LON MIN-MAX(D) E+ W-)
 +10, -10
 IN: EM=1 RP=2 RP METEO=3
 2
 PLTP


```
PROGRAM POINT
DIMENSION IR(20)
N=116B
IE=105B
90 WRITE(2,200)
230 FORMAT("POINT"/"LAT MIN-MAX(D) N+ S-")
READ(1,*)AMIN,AMAX
AMIN=AMIN*60.
AMAX=AMAX*60.
WRITE(2,231)
201 FORMAT("LON MIN-MAX(D) E+ W-")
READ(1,*)OMIN,OMAX
OMIN=OMIN*60.
OMAX=OMAX*60.
WRITE(2,501)
501 FORMAT("IN: BM=1 RP=2 RP METEO=3")
READ(1,*)KC
IN=5
IF(KC-1)502,502,506
502 WRITE(2,505)
505 FORMAT("FILE")
READ(1,*)NF
CALL PTAPE(7,NF,0)
IN=7
506 CONTINUE
WRITE(2,202)
202 FORMAT("PLTP")
5 GOTO(503,503,504),KC
503 READ(IN,601)IS,XLAT1,LAT2,XLON1,LON2
GOTO 512
504 READ(IN,602)IS,XLAT1,LAT2,XLON1,LON2
512 CONTINUE
IF(IS-999)703,99,99
703 J1=1
54 J10=J1+9
GOTO(700,701,702),KC
700 READ(7)IR
GOTO 704
701 READ(5,603)IR
704 IF(IR(18))702,702,55
55 J1=J10+1
GOTO 54
702 CONTINUE
XLAT1=XLAT1/100.
XLON1=XLON1/100.
IF(LAT2-N)1,2,1
1 XLAT1=-XLAT1
2 ILA=IFIX(XLAT1)
XLAT1=100.*XLAT1-40.*FLOAT(ILA)
IF(LON2-IE)3,4,3
3 XLON1=-XLON1
4 ILO=IFIX(XLON1)
XLON1=100.*XLON1-40.*FLOAT(ILO)
IX=IFIX(9999.*(XLON1-OMIN)/(OMAX-OMIN))
IY=IFIX(9999.*(XLAT1-AMIN)/(AMAX-AMIN))
WRITE(2,100)IX,IY
100 FORMAT(I4,I5)
GOTO5
99 GO TO 90
601 FORMAT(20X14,F5.0,1XA2,F6.0,1XA2)
602 FORMAT(4X13,10X,F4.0,1XA1,F5.0,1XA1)
603 FORMAT(5(3I4,I3))
END
```

FICHE TECHNIQUE DU PROGRAMME

NOM: LOOK

TYPE	TRACES	CHAINE	B	PERIPH.	IN	7
BUT	Trace les isolignes par points pour T,S,O2,SIGMA,DST d'une radiale on peut choisir la position des stations en mm			OUT	P	
				FORMAT	IN	B
				OUT	P	
Observations				ETIQ.	IN	M/S
Opérations					OUT	L
DATE DE MISE AU POINT	SEPTEMBRE 1974			PROGRAMMEUR		
LANGAGE UTILISE	FORTRAN			NOEL		
SSP EMPLOYES	CLE LEADR PTAPE SIGMA					
Starting ADDRESS	2					
Nb Source	Nb Objet		Nb Absolu			
Divers						
Clés employées						

FICHE EXPLOITATION

STATIONS, POS. ?

1 20 LOOK

2 30

3 40

4 50

5 60

6 70

8 100

9 110

10 120

12 160

13 180

14 200

15 220

27 250

28 280

29 310

30 340

31 370

9999 100 10 100 10

VARIABLE T=2, S=3, O2=4, SIG=5, DST=6

2

PROF MAX ?

200

ECART ? 2 3 4 5 6

~~100~~

50 100 10 100 10 10

PL TL

PL TP

30 6 74 1055 7404 1 2430

PL TP

30 6 74 1320 7404 0 2410

```

PROGRAM LOOK
DIMENSION IR(4,110),INO(51,2),IE(6)
WRITE(2,1)
1 FORMAT("LOOK"//"STATIONS, POS. ?")
N=0
2 N=N+1
INB=N-1
READ(1,*)INO(N,1),INO(N,2)
IF(INO(N,1)-999)2,3,3
3 REWIND 7
WRITE(2,4)
4 FORMAT("VARIABLE T=2, S=3, O2=4, SIG=5, DST=6")
READ(1,*)IV
WRITE(2,5)
5 FORMAT("PROF MAX ?")
READ(1,*)LZ
WRITE(2,6)
6 FORMAT("ECART ?")
READ(1,*)IE(IV)
IE(5)=10
IE(6)=10
LZ=10000/LZ
WRITE(2,10)
10 FORMAT("PLTL")
IDEB=INO(1,2)*25
IFIN=INO(INB,2)*27
IY=9999
WRITE(2,11)IDEB,IY
WRITE(2,12)IFIN,IY
11 FORMAT(2I5," ")
12 FORMAT(2I5)
NZ=10000/LZ
I1=1
I2=50
DO 20 MZ=50,NZ,50
IY=9999-MZ*LZ
WRITE(2,11)I1,IY
20 WRITE(2,12)I2,IY
WRITE(2,13)
13 FORMAT(" PLTT")

```

C
C
C

LECTURE

```

KS=1
30 CALL PTAPE(7,1,0)
READ(7,100)JR,MO,IA,IH,IC,IST
100 FORMAT(3I3,I5,I6,I4)
IF(IST-INO(KS,1))30,31,999
31 WRITE(2,14)
14 FORMAT("PLTP")
J1=1
32 J10=J1+9
READ(7)((IR(I,J),I=1,4),J=J1,J10)
IF(IR(2,J10))33,34,33
33 J1=J10+1
GO TO 32

```

C
C
C

DONNEES

```

34 IF(IV-5)35,36,37
35 DO 40 N=1,110
IF(IR(2,N))40,50,40
40 DO 42 N=IR(IV,N)

```

```

40 IR(2,N)=IR(IV,N)
36 DO 42 N=1,110
   IF(IR(2,N))41,50,41
41 T=FLOAT(IR(2,N))/100.
   S=FLOAT(IR(3,N))/100.
   A=SIGMA(T,S)
42 IR(2,N)=IFIX(A*100.)
37 DO 43 N=1,110
   IF(IR(2,N))44,50,44
44 T=FLOAT(IR(2,N))/100.
   S=FLOAT(IR(3,N))/100.
   A=SIGMA(T,S)
   A=A*1.E-3
   A=(.02736-A/(1.+A))*1.E5
43 IR(2,N)=IFIX(A)

```

C
C
C

RECHERCHE

```

50 IT1=IR(2,1)/IE(IV)
   N=0
   NP=2
53 IF(N-IR(1,NP))51,51,52
51 AN=FLOAT(N-IR(1,NP-1))/FLOAT(IR(1,NP)-IR(1,NP-1))
   IT2=(IR(2,NP-1)+IFIX(FLOAT(IR(2,NP)-IR(2,NP-1))*AN))/IE(IV)
   IF(IT2-IT1)200,201,200
201 IT1=IT2
   N=N+1
   GO TO 53
52 NP=NP+1
   IF(IR(2,NP))500,500,53

```

C
C
C

POINT

```

200 I PROF=9999-N*LZ
   I STA=IFIX(FLOAT(INO(KS,2))*26.3)
   IF(IV-5)202,203,204
202 WRITE(2,12)I STA,I PROF
   IF(IT2-IT1)205,205,206
205 MR=IT1-(IT1/5)*5
   GO TO 207
206 MR=IT2-(IT2/5)*5
207 IF(MR)201,208,201
208 I STA=I STA+25
   WRITE(2,12)I STA,I PROF
   GO TO 201
203 IF(IT2-250)210,202,202
210 JT1=IT1/10
   JT2=IT2/10
   IF(JT2-JT1)220,201,220
220 DO 221 I WW=1,3
   I STA=I STA+(I WW-1-1)*25
221 WRITE(2,12)I STA,I PROF
   GO TO 201
204 IF(IT1-25)202,202,210
500 WRITE(2,13)

```

C
C
C

MAX

```

   IF(IV-3)501,502,501
502 I P MAX=0
   I S MAX=0
   DO 503 K N=2,110
   IF(IR(2,KN))504,505,504
504 IF(IR(2,KN)-I S MAX)503,506,506

```

```
506 ISMAX=IR(2,KN)
    IPMAX=IR(1,KN)
503 CONTINUE
505 WRITE(2,10)
    IPRF=9999-IPMAX*LZ
    I STA=I STA-30
    WRITE(2,12)I STA,IPROF
    I STA=I STA+60
    WRITE(2,12)I STA,IPROF
    WRITE(2,13)
501  WRITE(2,110)JR,MO,IA,IH,IC,IST,IR(2,1)
110  FORMAT(3I3,I5,I6,I4,6X,I5)
    KS=KS+1
    IF(INO(KS,1)-999)30,3,3
999  PAUSE
    END
    ENDS
```

```

----- PROGRAM LOOK          - 'sans bande magnetique' - SA = 3047
                                603-77
DIMENSION IR(4,113),INO(51,2),IE(5)
WRITE(2,1)
1  FORMAT("LOOK"// "STATIONS, POS. ?")
812 FORMAT(5(3I4,13))
N=3
2  N=N+1
   INB=N-1
   READ(1,*)INO(N,1),INO(N,2)
   IF(INO(N,1)-999)2,3,3
3  WRITE(2,4)
4  FORMAT("VARIABLE T=2, S=3, O2=4, S13=5, DST=0")
   READ(1,*)IV
   WRITE(2,5)
5  FORMAT("PROF MAX ?")
   READ(1,*)L7
   WRITE(2,6)
6  FORMAT("ECART ?")
   READ(1,*)IE(IV)
   IE(5)=13
   IE(5)=10
   LZ=10000/L7
   WRITE(2,10)
10  FORMAT("PLTL")
   IDEB=INO(1,2)*25
   IFIN=INO(INB,2)*27
   IY=9999
   WRITE(2,11)IDEB,IY
   WRITE(2,12)IFIN,IY
11  FORMAT(2I5,"")
12  FORMAT(2I5)
   NZ=10000/L7
   I1=1
   I2=53
   DO 20 MZ=50,NZ,50
   IY=9999-MZ*LZ
   WRITE(2,11)I1,IY

23  WRITE(2,12)I2,IY
   WRITE(2,13)
13  FORMAT("      PLTT")

```

C
C
C

LECTURE

```

KS=1
30  READ(5,100)JR,MO,IA,IH,IC,IST
100  FORMAT(3I3,15,16,14)
   IF(IST-INO(KS,1))30,31,999
31  WRITE(2,14)
14  FORMAT("PLTP")
   J1=1
32  J10=J1+9
   READ(5,812)((IR(I,J),I=1,4),J=J1,J10)
   IF(IR(2,J10))33,34,33
33  J1=J10+1
   GO TO 32

```

C

```

34 IF (IV-5) 35, 35, 37
35 DO 40 N=1, 110
    IF (IR(2, N)) 40, 50, 40
40 IR(2, N) = IR(IV, N)
36 DO 42 N=1, 110
    IF (IR(2, N)) 41, 50, 41
41 T = FLOAT(IR(2, N)) / 100.
    S = FLOAT(IR(3, N)) / 100.
    A = SIGMA(T, S)
42 IR(2, N) = IFIX(A * 100.)
37 DO 43 N=1, 110
    IF (IR(2, N)) 44, 50, 44
44 T = FLOAT(IR(2, N)) / 100.
    S = FLOAT(IR(3, N)) / 100.
    A = SIGMA(T, S)
    A = A * 1. E-3
    A = (.32736 - A / (1. + A)) * 1. E5
43 IR(2, N) = IFIX(A)

```

C

C

C

RECHERCHE

```

50 IT1 = IR(2, 1) / IE(IV)
    N = 3
    NP = 2
53 IF (N - IR(1, NP)) 51, 51, 52
51 AN = FLOAT(N - IR(1, NP - 1)) / FLOAT(IR(1, NP) - IR(1, NP - 1))
    IT2 = (IR(2, NP - 1) + IFIX(FLOAT(IR(2, NP) - IR(2, NP - 1)) * AN)) / IE(IV)
    IF (IT2 - IT1) 200, 201, 200
201 IT1 = IT2
    N = N + 1
    GO TO 53
52 NP = NP + 1
    IF (IR(2, NP)) 503, 500, 53

```

```
C
230 IPROF=9999-N*LZ
    ISTA=IFIX(FLOAT(INO(KS,2))*25.3)
    IF(IV-5)202,203,204
232 WRITE(2,12) ISTA, IPROF
    IF(IT2-IT1)205,205,206
235 MR=IT1-(IT1/5)*5
    GO TO 207
236 MR=IT2-(IT2/5)*5
237 IF(MR)201,209,201
238 ISTA=ISTA+25
    WRITE(2,12) ISTA, IPROF
    GO TO 241
233 IF(IT2-250)210,202,202
210 JT1=IT1/10
    JT2=IT2/10
    IF(JT2-JT1)220,201,220
203 DO 201 IVV=1,3
    ISTA=ISTA+(IVV-1)*25
221 WRITE(2,12) ISTA, IPROF
    GO TO 201
234 IF(IT1-25)202,202,210
500 WRITE(2,13)
C
C     MAX
C
    IF(IV-3)501,502,501
502 IPMAX=0
    ISMAX=0
    DO 503 KN=2,110
    IF(IR(2,KN))504,505,504
504 IF(IR(2,KN)-ISMAX)503,506,506
505 ISMAX=IR(2,KN)
    IPMAX=IR(1,KN)
503 CONTINUE
505 WRITE(2,13)
    IPROF=9999-IPMAX*LZ
    ISTA=ISTA-30
    WRITE(2,12) ISTA, IPROF
    ISTA=ISTA+60
    WRITE(2,12) ISTA, IPROF
    WRITE(2,13)
501 WRITE(2,110) JR,MO,IA,IH,IC,IST,IR(2,1)
110 FORMAT(3I3,15,16,14,6X,15)
    KS=KS+1
    IF(INO(KS,1)-999)30,3,3
999 PAUSE
    END
    ENDS
```


FICHE TECHNIQUE DU PROGRAMME

NOM: LOOK2

TYPE	CHAINE				PERIPH.	IN	7	
BUT	Visualise dans le plan de isostries les isolignes de l'un quelconque des paramètres Z, T, S, O.				PERIPH.	OUT	-	
Observations					FORMAT	IN	8	
Opérations					FORMAT	OUT	8	
					ETIQ.	IN	M	
					ETIQ.	OUT	M	
DATE DE MISE AU POINT	Octobre 74				PROGRAMMEUR			
LANGAGE UTILISE	FORTRAN				CITEAU			
SSP EMPLOYES	SIGAL							
Starting ADDRESS	2							
Nb Source	1	Nb Objet	1	Nb Absolu				1
Divers								
Clés employées								

FICHE EXPLOITATION

```
0311 PROGRAM LOOKP
0312 DIMENSION IR(4,11), INOC(5,5), IEC(5)
0313 1 FORMAT('COUPE PAR PARTOUT A DST.??????.FI(0,?)=??.I(0,?)=??.')
0314 WRITE(2,1)
0315 N=1
0316 2 N=N+1
0317 INB=N-1
0318 READ(1,*)INOC(N,1),INOC(N,2)
0319 IF(INOC(N,1)-999)3,3,3
0320 3 REV=1/2
0321 WRITE(2,7)
0322 7 FORMAT('REPRESENTED (0,?)TC(0,?)C(0,?)R(0,?)')
0323 READ(1,*)IV
0324 WRITE(2,5)
0325 5 FORMAT('DST.MAX.??')
0326 READ(1,*)LZ
0327 WRITE(2,6)
0328 6 FORMAT('ECART ??')
0329 READ(1,*)IE(IV)
0330 IE(5)=10
0331 IE(5)=13
0332 LZ=13000/LZ
0333 WRITE(2,10)
0334 10 FORMAT('PLTL')
0335 IDEB=INOC(1,2)*25
0336 IFIN=INOC(INB,2)*27
0337 IY=9999
0338 IFD=0
0339 WRITE(2,11)IFIN,IY
0340 WRITE(2,12)IFD,IY
0341 WRITE(2,12)IFD,IFD
0342 11 FORMAT(2I5,' ')
0343 12 FORMAT(2I5)
0344 N7=10000/LZ
0345 I1=1
0346 I0=51
0347 DO 20 MZ=50,NZ,50
0348 IY=M7*LZ
0349 WRITE(2,11)I1,IY
0350 20 WRITE(2,12)I0,IY
0351 WRITE(2,13)
0352 13 FORMAT(' PLTT')
0353 C
0354 C LECTURE
0355 C
0356 NC=1
0357 31 CALL RTAPE(7,1,0)
0358 READ(7,130)JR,MO,IA,IB,IC,IST
0359 133 FORMAT(3I3,15,13,14)
0360 IF(IST-IDUCK(5,1))33,31,000
0361 31 WRITE(2,14)
0362 14 FORMAT('PLTP')
0363 J1=1
0364 32 J10=J1+9
0365 READ(7)((IR(I,J),I=1,4),J=J1,J10)
0366 IF(IR(2,J10))33,35,33
0367 33 J1=J10+1
0368 GO TO 32
0369 C
```

```
0061 C          DONNEES
0062 C
0063 35 DO 43 N=1,113
0064      IF(IR(2,N)) 44,53,44
0065 44 T=FLOAT(IR(2,N))/133.
0066      S=FLOAT(IR(3,N))/133.
0067      A=5135*(T,S)
0068      A=A*1.E-3
0069      A=(.32735-A/(1.+A))*1.E5
0070      IR(3,N)=I*(I,N)
0071 43 IR(4,N)=IFIX(A)
0072 C
0073 C          RECHERCHE
0074 C
0075 51 IT1=IR(2,1)/IR(1,1)
0076      N=1
0077      NP=2
0078 53 IF(N-IR(1,NP)) 51,51,52
0079 51 AN=FLOAT(N-IR(1,NP-1))/FLOAT(IR(1,NP)-IR(1,NP-1))
0080      IT2=(IR(2,NP-1)+IFIX(FLOAT(IR(2,NP)-IR(2,NP-1))*10))/IR(1,1)
0081      IT3=IR(4,NP-1)+IFIX(FLOAT(IR(4,NP)-IR(4,NP-1))*10)
0082      IF(IT2-IT1) 231,231,233
0083 231 IT1=IT2
0084      N=N+1
0085      GO TO 53
0086 52 NP=NP+1
0087      IF(IR(2,NP)) 533,533,53
0088 C
0089 C          POINT
0090 C
0091 234 IPRUF=IT3*L2
0092      ISTA=IFIX(FLOAT(INUCK(2))*25.3)
0093      IF(IV-5) 232,233,234
0094 232 WRITE(2,12) ISTA, IPRUF
0095      IF(IT2-IT1) 235,235,235
0096 235 MR=IT1-(IT1/5)*5
0097      GO TO 237
0098 236 MR=IT2-(IT2/5)*5
0099 237 IF(MR) 231,236,231
0100 233 ISTA=ISTA+25
0101      WRITE(2,12) ISTA, IPRUF
0102      GO TO 231
0103 233 IF(IT2-253) 210,202,202
0104 213 JT1=IT1/13
0105      JT2=IT2/13
0106      IF(JT2-JT1) 220,221,223
0107 223 DO 221 I=1,3
0108      ISTA=ISTA+(I+1-1)*25
0109 221 WRITE(2,13) ISTA, IPRUF
0110      GO TO 231
0111 234 IF(IT1-25) 212,212,213
0112 211 WRITE(2,13)
0113 C
0114 C          MAX
0115 C
0116      IF(IV-3) 531,532,531
0117 532 IPRAX=M
0118      ISMAX=0
0119      DO 533 KN=2,113
0120      IF(IR(2,KN)) 504,505,534
```

```
0120 504 IF(IRC(2,KND)-ISMAY)503,505,506
0121 505 ISMAX=IRC(2,KND)
0122 IPMAX=IRC(4,KND)
0123 503 CONTINUE
0124 505 WRITE(2,130)
0125 IPROF=IPMAX*ELZ
0126 ISTA=ISTA-3
0127 WRITE(2,130)ISTA,IPROF
0128 ISTA=ISTA+1
0129 WRITE(2,130)ISTA,IPROF
0130 WRITE(2,130)
0131 501 WRITE(2,113)JRMOD,IA,IB,IC,IE,IF,IG,II
0132 113 FORMAT(3I3,15,13,14,3X,10)
0133 KS=KS+1
0134 IF(KS-IND)30,30,3
0135 30 PAUSE
0136 END
0137 ENDS
```

**END-OF-TAPE

*
/E

*END

FICHE TECHNIQUE DU PROGRAMME

NOM: LOOK3

TYPE	CHAINE			PERIPH.	IN	1		
BUT	Visualise sur le Plotteur, la cote Dynamique des isobares.			OUT	2			
Observations	Au choix: Niveau de Reference, echelle, pas entre isob.			FORMAT	IN	21x11		
Opérations				OUT	21x11			
DATE DE MISE AU POINT	Octobre 1974			ETIQ.	IN	A4		
LANGAGE UTILISE				OUT	A4			
SSP EMPLOYES	CLE, SIGNAL			PROGRAMMEUR				
Starting ADDRESS	2			CITEAU				
Nb Source	1	Nb Objet	1				Nb Absolu	1
Divers								
Clés employées	15							

FICHE EXPLOITATION

```
0001 PROGRAM LOOK3
0002 DIMENSION JD(4,110),ID(51),A(51)
0003 1 FORMAT("COTE DYNAMIQUE D'ISOBARES"/"ECART ENTRE ISOBARES?")
0004 20 FORMAT("ERREUR DE REFERENCE")
0005 50 FORMAT("ECHELLE:30MM PAR DEGRE,OK=1,SINON=2")
0006 51 FORMAT("ST A SUIVRE:LEVER(15),DEBUT A GAUCHE=1, DROITE=2")
0007 52 FORMAT("ST. DE DEBUT?, ET DE FIN?")
0008 53 FORMAT("ST. ?")
0009 54 FORMAT("ERREUR D'ETIQUETTE. POSITION **.***,***.***(+NE) ?")
0010 55 FORMAT("***MM PAR DEGRE?")
0011 14 FORMAT("PLTP")
0012 13 FORMAT("PLTT")
0013 102 FORMAT(215,5X2(12,1H/),12,2X14,2XF5.2,A2,2XF5.2,3?)
0014 100 FORMAT(313,15,16,14,F5.2,2X14,F5.2,2X14)
0015 ECH=30.
0016 NU=1160
0017 IE=1050
0018 IV=1070
0019 ISU=1230
0020 NE=7
0021 L=0
0022 4 WRITE(2,1)
0023 READ(1,*)ICAR
0024 WRITE(2,5)
0025 5 FORMAT("H.DYN. MAXI (EN CM.DYN),ET REFERENCE ?")
0026 READ(1,*)LZ,NR
0027 LZ=10000/LZ
0028 10 FORMAT("PLTL")
0029 11 FORMAT(215,"")
0030 12 FORMAT(215)
0031 NZ=10000/LZ
0032 WRITE(2,50)
0033 READ(1,*)I0
0034 GOTO(91,92),I0
0035 92 WRITE(2,55)
0036 READ(1,*)ECH
0037 91 WRITE(2,51)
0038 READ(1,*)IG
0039 IX=9999*(IG-1)
0040 IY=0
0041 WRITE(2,10)
0042 WRITE(2,11)IX,IY
0043 WRITE(2,12)IX,IY
0044 I1=IX+3-2*IG
0045 I2=IX+50*(3-2*IG)
0046 DO 20 MZ=1,NZ,10
0047 IY=(MZ-1)*LZ
0048 WRITE(2,12)I1,IY
0049 WRITE(2,12)I2,IY
0050 WRITE(2,12)I1,IY
0051 20 CONTINUE
0052 IY=9999
0053 WRITE(2,12)I1,IY
0054 WRITE(2,13)
0055 I=1
0056 80 ID(I)=(I-1)*ICAR
0057 IF(ID(I)-NR)81,90,90
0058 81 I=I+1
0059 N=I
```

```
0060      GOTO 80
0061      20 CALL CLE(KLE)
0062      IF(KLE)7,8,8
0063      7 WRITE(2,52)
0064      READ(1,*)IDEB,IFIN
0065      GOTO 9
0066      8 WRITE(2,53)
0067      READ(1,*)IDEB
0068      9 IF(IDEB-9999)39,4) ,41
0069      41 STOP
0070      39 CALL PTAPE(7,1,0)
0071      READ(NE,109)JR,MO,IA,IH,IC,IS,AT,LA,ON,LU
0072      IF(IS-IDEB)39,40,39
0073      40 WRITE(2,100)IC,IS,JR,MO,IA,IH,AT,LA,ON,LU
0074      L=L+1
0075      IF(LA-NO)201,202,201
0076      201 IF(LA-ISU)203,199,203
0077      199 AT=-AT
0078      202 IF(L0-IE)205,206,205
0079      205 IF(L0-IW)203,204,203
0080      204 ON=-ON
0081      GOTO 206
0082      203 WRITE(2,54)
0083      READ(1,*)AT,ON
0084      206 IF(L-1)207,207,208
0085      207 Y1=AT/.6-FLOAT(IFIX(AT))*4./6.
0086      X1=ON/.6-FLOAT(IFIX(ON))*4./6.
0087      208 Y2=AT/.6-FLOAT(IFIX(AT))*4./6.
0088      X2=ON/.6-FLOAT(IFIX(ON))*4./6.
0089      DL=SQRT((X2-X1)**2+(Y2-Y1)**2)
0090      J1=1
0091      32 J10=J1+9
0092      READ(7)((JD(I,J),I=1,4),J=J1,J10)
0093      IF(JD(2,J10))33,35,33
0094      33 J1=J10+1
0095      GO TO 32
0096      35 J=1
0097      D=0
0098      DI=0
0099      I=1
0100      M=0
0101      PI=0
0102      18 IF(JD(2,J))27,27,19
0103      19 IF(M-JD(1,J))200,23,21
0104      200 AN=FLOAT(M-JD(1,J-1))/FLOAT(JD(1,J)-JD(1,J-1))
0105      T=.01*FLOAT(JD(2,J-1))+FLOAT(JD(2,J)-JD(2,J-1))*AN*.01
0106      S=.01*FLOAT(JD(3,J-1))+FLOAT(JD(3,J)-JD(3,J-1))*AN*.01
0107      P=FLOAT(M)
0108      GOTO 250
0109      23 P=FLOAT(JD(1,J))
0110      T=FLOAT(JD(2,J))*01
0111      S=FLOAT(JD(3,J))*01
0112      250 CALL SIGAL(P,T,S,SIGMA,DALFA)
0113      D=D+(DI+DALFA)*(P-PI)*1.E+02/2.
0114      IF(ID(I)-M)100,350,100
0115      350 A(I)=D
0116      I=I+1
0117      100 IF(NR-M)70,24,70
0118      70 PI=P
0119      DI=DALFA
```

```
0120      M=M+1
0121      GOTO 19
0122      21 J=J+1
0123      GO TO 18
0124      24 WRITE(2,14)
0125      ISTA=IFIX((DL*ECH+10.)*26.3)*(3-2*IG)+9999*(IG-1)
0126      D067 I=1,N
0127      IPROF=LZ*IFIX(A(N)-A(I))
0128      IF(ID(I)-100*(ID(I)/100))63,64,63
0129      64 ISTA=ISTA+25
0130      WRITE(2,12)ISTA,IPROF
0131      ISTA=ISTA-25
0132      63 WRITE(2,12)ISTA,IPROF
0133      67 CONTINUE
0134      WRITE(2,13)
0135      CALL CLE(KLE)
0136      IF(KLE)65,8,8
0137      65 IF(IDEB-IFIN)66,90,90
0138      66 IDEB=IDEB+1
0139      GOTO 9
0140      27 WRITE(2,28)
0141      GOTO 4
0142      PAUSE
0143      END
0144      ENDS
```

**END-OF-TAPE

*

/E

*END

FICHE TECHNIQUE DU PROGRAMME

NOM: **GRAD T**

TYPE					CHAINÉ			PERIPH.	IN	7
BUT	Visualise les gradients Thermiques						FORMAT	OUT	2	
Observations							FORMAT	IN	B	
Opérations							FORMAT	OUT	L	
Observations							ETIQ.	IN	M	
Opérations							ETIQ.	OUT	M	
DATE DE MISE AU POINT	9/12/74					PROGRAMMEUR				
LANGAGE UTILISE	FORTRAN					CITEAU				
SSP EMPLOYES	CL6									
Starting ADDRESS										
Nb Source	1	Nb Objet	1	Nb Absolu	1					
Divers										
Clés employées	15									

FICHE EXPLOITATION

CUPE DE GRADIENT THERMIQUE
 PROFONDEUR MAX.?
 500
 ECHELLE: 30MM PAR DEGRE, CK=1, SINCK=2
 1
 ST A SUIVRE: LEVER(15), DEBUT A GAUCHE=1, DROITE=2
 1
 PLTL
 ST. DE DEBUT?, ET DE FIN?
 1, 3
 7606 1 24/ 5/76 1045 2.02W 1.00E
 PLTP
 7606 2 24/ 5/76 1700 1.41W .59E
 PLTP
 7606 3 24/ 5/76 2300 1.20W 1.00E
 PLTP
 ST. DE DEBUT?, ET DE FIN?

```
PROGRAM GRADT
DIMENSION JDC(4,113)
1 FORMAT("COUPE DE GRADIENT THERMIQUE"/"PROFONDEUR (MM) ?")
50 FORMAT("ECHELLE: 30MM PAR DEGRE, OK=1, SINON=2")
51 FORMAT("ST A SUIVRE: LEVER(15), DEBUT A GAUCHE=1, DROITE=2")
52 FORMAT("ST. DE DEBUT?, ET DE FIN?")
53 FORMAT("ST. ?")
54 FORMAT("ERREUR D'ETIQUETTE. POSITION **.*, *.*.* (+NE) ?")
55 FORMAT("***MM PAR DEGRE?")
14 FORMAT("PLTP")
13 FORMAT("PLTT")
102 FORMAT(2I5, 5X2(I2, I4/), I2, 2X14, 2XF5, 2X2, 2XF5, 2X14)
109 FORMAT(6I3, I5, I5, I4, F5.2, 2X11, F5.2, 2X11)
539 FORMAT(5(6I4, I3))
ECH=30.
NO=116B
IE=135B
IV=127B
ISU=123B
NE=5
L=0
4 WRITE(2,1)
READ(1,*)LZ
LZ=13300/LZ
10 FORMAT("PLTL")
11 FORMAT(2I5, " ")
12 FORMAT(2I5)
NZ=10000/LZ
WRITE(2,50)
READ(1,*)I0
GOTO(91,92), I0
92 WRITE(2,55)
READ(1,*)ECH
91 WRITE(2,51)
READ(1,*)IG
IX=9999*(IG-1)
IY=0
WRITE(2,10)
WRITE(2,11)IX, IY
WRITE(2,12)IX, IY
I1=IX+3-2*IG
I2=IX+50*(3-2*IG)
DO 20 MZ=1, NZ, 50
IY=9999-(MZ-1)*LZ
WRITE(2,12)I1, IY
WRITE(2,12)I2, IY
WRITE(2,12)I1, IY
20 CONTINUE
IY=0
WRITE(2,12)I1, IY
WRITE(2,13)
90 CALL CLE(KLE)
IF(KLE)7, 8, 8
7 WRITE(2,52)
READ(1,*)IDEB, IFIN
```

```
GOTO 9
8 WRITE(2,53)
  READ(1,*)IDEB
9 IF(IDEB-9999)39,41,41
41 STOP
39 READ(NE,109)JR,MO,IA,IH,IC,IS,AT,LA,ON,LO
  IF(IS-IDEB)39,40,39
40 WRITE(2,102)IC,IS,JR,MO,IA,IH,AT,LA,ON,LO
  L=L+1
  IF(LA-NO)231,232,231
231 IF(LA-ISU)203,199,233
199 AT=-AT
232 IF(LO-IE)205,206,205
235 IF(LO-IV)203,204,203
204 ON=-ON
  GOTO 206
202 WRITE(2,54)
  READ(1,*)AT,ON
206 IF(L-1)207,207,208
207 Y1=AT/.6-FLOAT(IFIX(AT))*4./6.
  X1=ON/.6-FLOAT(IFIX(ON))*4./6.
208 Y2=AT/.6-FLOAT(IFIX(AT))*4./6.
  X2=ON/.6-FLOAT(IFIX(ON))*4./6.
  DL=SQRT((X2-X1)**2+(Y2-Y1)**2)
  J1=1
32 J10=J1+9
  READ(5,507)((JD(I,J),I=1,4),J=J1,J10)
  IF(JD(2,J10))33,35,33
33 J1=J10+1
  GO TO 32
35 JSTA=IFIX((DL*ECH+10.)*26.3)*(3-2*IG)+9999*(IG-1)
  WRITE(2,14)
  I=1
  5 I=I+1
  ISTA=JSTA
  IF(JD(2,I))42,80,42
42 Z=FLOAT(JD(1,I)+JD(1,I-1))*0.5
  IZ=IFIX(Z)
  DT=FLOAT(JD(2,I-1)-JD(2,I))/FLOAT(JD(1,I)-JD(1,I-1))
  IDT=IFIX(DT+5.)/10
  IF(IDT)5,5,28
28 IPROF=9999-IZ*LZ
  IF(IDT-1)5,6,3
  6 WRITE(2,12)ISTA,IPROF
  GOTO 5
  3 DO 67 M=1,IDT
  ISTA=JSTA+(M-1)*23
  WRITE(2,12)ISTA,IPROF
67 CONTINUE
  GOTO 5
80 WRITE(2,13)
  CALL CLE(KLE)
  IF(KLE)65,8,8
65 IF(IDEB-IFIN)66,90,90
66 IDEB=IDEB+1
  GOTO 9
  END
  ENDS
```

FICHE TECHNIQUE DU PROGRAMME

NOM: EX1TR

TYPE		CHAINE		PERIPH.	IN	5,7
BUT	Calcul des UAO, %O2, SIGMA, DST, HDYN			FORMAT	OUT	2, L
Observations				ETIQ.	IN	M
Opérations					OUT	L
DATE DE MISE AU POINT	AVRIL 1974			PROGRAMMEUR		
LANGAGE UTILISE	FORTRAN			NOEL		
SSP EMPLOYES	SIGMA SIGMZ OXY CLE PTAPE LEADR					
Starting ADDRESS	2					
Nb Source	Nb Objet		Nb Absolu			
Divers						
Clés employées						

FICHE EXPLOITATION

EX1TR
 IN 5 CU 7
 5
 IMMERSIONS?
 0
 1
 2
 3
 9999

* * * * *

CFC SITE: 7606
 STATION : 1
 POSITION : 2.02' 0 N 1.00' 0 E

DATE: 24 5 76
 HEURE: 1045

AIR= 270 233 VENT= 160/ 8 FCND= 4600 NP= 59

PR.	TEMP.	SAL.	O2	C%	UAO	SIGMA	DST	H.DYN
0	28.40	34.14	4.82	108	-.35	21.64	617.9	.000
1	28.40	34.14	4.75	107	-.31	21.64	617.9	.618
2	28.40	34.14	4.74	106	-.27	21.64	617.9	1.235
3	28.40	34.14	4.69	105	-.23	21.64	617.9	1.653

* * * * *

```
PROGRAM EXITR
DIMENSION IDC(4,110),IP(61),MET(12)
WRITE(2,1)
1 FORMAT(/"EXITR"/"IN 5 OU 7")
  READ(1,*) IN
  IF(IN-7)2,3,3
3 WRITE(2,4)
4 FORMAT("FILE /")
  READ(1,*)IFIL
  CALL PTAPE(7,IFIL,0)
2 WRITE(2,5)
5 FORMAT("IMMERSIONS IN=5 OU 7 ?")
  READ(1,*)JN
  DO 6 N=1,61
  READ(JN,*)IP(N)
  IF(IP(N)-9999)6,20,20
6 CONTINUE
20 CALL CLEKLE)
  IF(KLE)2,7,2
7 READ(IN,8)(MET(N),N=1,6),AT,IA1,IA2,ON,I01,I02,(MET(N),N=7,12)
8 FORMAT(3I3,15,16,14,F5.2,11,A2,F5.2,11,A2,2I4,4X,2I3,15,16)
  IF(MET(6)-9999)14,21,21
21 PAUSE
  GO TO 20
14 J1=1
15 J10=J1+9
  IF(IN-7)9,10,10
9 READ(IN,11)((ID(I,J),I=1,4),J=J1,J10)
  GO TO 12
10 READ(7)((ID(I,J),I=1,4),J=J1,J10)
12 IF(ID(2,J10))13,100,13
13 J1=J10+1
  GO TO 15
11 FORMAT(5(3I4,13))
100 WRITE(2,101)
101 FORMAT(7(9X"*"))
  WRITE(2,102)MET(5),MET(1),MET(2),MET(3),MET(6),MET(4),
  IAT,IA1,IA2,ON,I01,I02,(MET(N),N=7,12)
102 FORMAT(2/"CROISIERE:"15,29X"DATE:"3I3/"STATION :":15,
129X"HEURE:"18/"POSITION :":
1F5.2,1H',11,1XA2,3XF6.2,1H',11,1XA2,
2// "AIR="2I4,
29X," VENT="I3"/"I2" FOND="I5" NP="I3)
  WRITE(2,103)
103 FORMAT(// " PR. TEMP. SAL. O2 O% UAU
1SIGMA DST H.DYN"/)
  "
```

```
NP=1
N=0
ND=1
D=0
DI=0
PI=0
205 IF(N-ID(1,ND))200,201,202
200 AN=FLOAT(N-ID(1,ND-1))/FLOAT(ID(1,ND)-ID(1,ND-1))
T=.31*FLOAT(ID(2,ND-1))+FLOAT(ID(2,ND)-ID(2,ND-1))*AN
T=.31*FLOAT(ID(2,ND-1))+FLOAT(ID(2,ND)-ID(2,ND-1))*AN
T=.31*FLOAT(ID(2,ND-1))+FLOAT(ID(2,ND)-ID(2,ND-1))*AN*.31
S=.31*FLOAT(ID(3,ND-1))+FLOAT(ID(3,ND)-ID(3,ND-1))*AN*.31
O=FLOAT(ID(4,ND-1))+FLOAT(ID(4,ND)-ID(4,ND-1))*AN
P=FLOAT(N)
GO TO 250
201 T=.01*FLOAT(ID(2,ND))
S=.01*FLOAT(ID(3,ND))
O=FLOAT(ID(4,ND))
P=FLOAT(N)
GO TO 250
202 ND=ND+1

IF(ID(2,ND))500,500,205
250 CALL SIGAL(P,T,S,SIGM,DALFA)
D=D+(DI+DALFA)*(P-PI)*1.E+32/2.
PI=P
DI=DALFA
IF(N-IP(NP))300,261,300
261 O=O/100.
O100=OXY(T,S,O)
I0100=IFIX(O100+.5)
A=.001*SIGM
DST=(.02736-A/(1.+A))*1.E+5
UAO=O*100./O100-O
WRITE(2,270)IP(NP),T,S,O,I0100,UAO,SIGM,DST,D
270 FORMAT(I6,2(2XF5.2),2XF4.2,2XI3,2XF4.2,2XF5.2,2XF6.1,
12XF8.3)
NP=NP+1
IF(IP(NP)-9999)300,500,500
300 N=N+1
GO TO 205
500 DO 502 K=NP,62
WRITE(2,501)
501 FORMAT(" ")
502 CONTINUE
WRITE(2,101)
CALL PTAPE(7,1,0)
GO TO 20
END
END$
-----
```

FICHE TECHNIQUE DU PROGRAMME

NOM: LISTR

TYPE	SORTIE	CHAINE	B			PERIPH.	IN	5,7
BUT	listing des fautes caractéristiques (profondeurs réduites)					OUT	5	
						FORMAT	IN	A
							OUT	L
Observations						ETIQ.	IN	M
Opérations							OUT	L
DATE DE MISE AU POINT	OCTOBRE 1974					PROGRAMMEUR		
LANGAGE UTILISE	FORTRAN					NOEL		
SSP EMPLOYES								
Starting ADDRESS	2							
Nb Source	1	Nb Objet	1	Nb Absolu	1			
Divers								
Clés employées								

FICHE EXPLOITATION

LIST

METEC 0 CU 1 ?

0
IN ?
5

* * * * *

CRCI SIERE: 7606

DATE: 24 5 76

STATION : 1

HEURE: 1045

0	2840	3414	482	75	1690	3566	300	150	1440	3546	286	250	1110	3509	230
5	2840	3414	461	80	1660	3565	272	155	1440	3545	272	260	1090	3507	223
10	2840	3415	468	85	1640	3563	258	160	1430	3544	244	270	1070	3504	223
15	2840	3416	482	90	1620	3562	286	165	1420	3542	223	260	1040	3501	223
20	2840	3416	482	95	1600	3560	286	170	1390	3538	195	290	1020	3498	230
25	2840	3418	454	100	1570	3559	279	175	1360	3534	181	300	1000	3496	223
30	2290	3422	475	105	1560	3557	279	180	1330	3532	174	325	940	3491	223
35	2150	3574	440	110	1540	3556	258	185	1320	3528	167	350	910	3488	209
40	2100	3597	433	115	1530	3554	265	190	1290	3526	167	375	890	3485	216
45	2090	3600	412	120	1510	3553	272	195	1260	3523	167	400	850	3478	202
50	2080	3598	398	125	1500	3552	265	200	1230	3521	167	425	770	3474	237
55	2040	3586	377	130	1480	3550	258	210	1200	3518	168	450	730	3471	250
60	1980	3576	326	135	1470	3550	286	220	1160	3515	209	475	700	3469	300
65	1800	3572	307	140	1460	3548	307	230	1150	3512	216	500	670	3467	314
70	1720	3568	300	145	1450	3547	300	240	1120	3510	230	0	0	0	64

* * * * *

LISTR

```
PROGRAM LIST
DIMENSION IR(4,110),MAT(12)
WRITE(2,1)
1 FORMAT("LIST"// "METEO 0 OU 1 ?")
READ(1,*)MET
WRITE(2,2)
2 FORMAT("IN ?")
READ(1,*)IN
IF(IN-7)4,3,3
3 REWIND 7
WRITE(2,5)
5 FORMAT("FILE ?")
READ(1,*)IFIL
CALL PTAPE(7,IFIL,0)
4 DO 6 I=1,4
DO 6 J=1,110
6 IR(I,J)=0
IF(MET)10,11,10
10 READ(IN,650)(MAT(I),I=1,6),AT,IA1,IA2,ON,I01,I02,(MAT(I)
1,I=7,12)
GO TO 20
11 READ(IN,650)(MAT(I),I=1,6)
20 IF(MAT(6)-9999)21,22,21
21 J1=1
23 J10=J1+9
IF(IN-7)30,31,31
30 READ(IN,130)((IR(I,J),I=1,4),J=J1,J10)
130 FORMAT(5(3I4,I3))
GO TO 32
31 READ(7)((IR(I,J),I=1,4),J=J1,J10)
32 IF(IR(2,J10))33,500,33
33 J1=J10+1
GO TO 23
500 WRITE(2,50)
50 FORMAT(5/,7(9X"*"))
WRITE(2,200)MAT(5),MAT(1),MAT(2),MAT(3),MAT(6),MAT(4)
200 FORMAT(2/"CROISIERE:"I5,29X"DATE:"3I3/"STATION : "I5,29X
1"HEURE:"I8)
IF(MET)501,502,501
501 WRITE(2,201)AT,IA1,IA2,ON,I01,I02,(MAT(N),N=7,12)
201 FORMAT("POSITION:"F5.2,1H',I1,1XA2,3XF6.2,1H',I1,1XA2, //
1"AI R="2I4,9X," VENT="I3"0/"I2" FOND="I5" VP="I3,/)
650 FORMAT(3I3,I5,I6,I4,F5.2,I1,A2,F6.2,I1,A2,2I4,4X,2I3,I5,I6)
502 NL=(MAT(12)+3)/4
DO 777 KW=1,NL
KY=KW+NL*3
777 WRITE(2,505)((IR(I,J),I=1,4),J=KW,KY,NL)
505 FORMAT(4(I4,2I5,I4))
WRITE(2,50)
700 CALL CLE(K)
IF(K)800,701,800
701 IF(IN-7)4,702,702
702 CALL PTAPE(7,1,0)
GO TO 4
22 PAUSE
GO TO 4
800 END
END$
```


FICHE TECHNIQUE DU PROGRAMME

NOM: VITES 2

TYPE	CHAINE				PERIPH.	IN	7,5	
BUT	Calcul de Vitesses géostrophiques par rapport à une référence à fixer				OUT	2		
					FORMAT	IN	B, A	
Observations	Message d'erreur si le niveau de référence n'est				OUT	4		
Opérations					ETIQ.	IN	M	
					OUT	M		
DATE DE MISE AU POINT	Janvier 74				PROGRAMMEUR			
LANGAGE UTILISE	FORTRAN				Citeau			
SSP EMPLOYES								
Starting ADDRESS								
Nb Source	1	Nb Objet	1	Nb Absolu				1
Divers								
Clés employées								

FICHE EXPLOITATION

- Ce programme travaille sur des données réduites, Binaires sur BM et ASCII sur R.P. Cette version s'appelle VITES2
- Il en existe une version appelée VIGIE qui opère sur les données brutes issues de SONDE

```
0001      PROGRAM VITES
0002      DIMENSION ID(50),A(2,53),JD(4,110),X(2),Y(2),DR(2),JSC(2)
0003      NO=116B
0004      IE=105B
0005      WRITE(2,105)
0006      READ(1,*)NR,JL
0007      I=1
0008      5 READ(JL,*)ID(I)
0009      N=I
0010      IF(ID(I)-1500)3,3,4
0011      3 I=I+1
0012      GOTO 5
0013      105 FORMAT("VITESSE GEOMETRIQUE"/"NEF. 2, DIRECTIVE (ID:1 OF 50)")
0014      4 L=1
0015      WRITE(2,108)
0016      NE=7
0017      108 FORMAT("ST. ?")
0018      7 READ(1,*)NF
0019      IF(NF-9999)39,41,41
0020      41 STOP
0021      39 CALL PTAPE(7,1,0)
0022      READ(NE,109)JR,MO,IA,IH,IC,IS,AT,LA,ON,LO
0023      IF(IS-NF)39,40,39
0024      109 FORMAT(3I3,15,16,14,F5.2,2XA1,F6.2,2XA1)
0025      40 JS(L)=IS
0026      WRITE(2,102)IC,IS,JR,MO,IA,IH,AT,LA,ON,LO
0027      IF(LA-NO)1,2,1
0028      1 AT=-AT
0029      2 Y(L)=AT
0030      IF(LO-IE)11,12,11
0031      11 ON=-ON
0032      12 X(L)=ON
0033      Y(L)=Y(L)/0.6-FLOAT(IFIX(Y(L)))*4./5.
0034      X(L)=X(L)/0.6-FLOAT(IFIX(X(L)))*4./5.
0035      D=0
0036      DI=0
0037      I=1
0038      M=0
0039      PI=0
0040      JI=1
0041      15 JI0=JI+9
0042      READ(NE)((JD(K,J),K=1,4),J=JI,JI0)
0043      IF(JD(2,JI0))16,17,16
0044      16 JI=JI+10
0045      GOTO 15
0046      17 J=1
0047      18 IF(JD(2,J))27,27,19
0048      19 IF(M-JD(1,J))200,23,21
0049      200 AN=FLOAT(M-JD(1,J-1))/FLOAT(JD(1,J)-JD(1,J-1))
0050      T=.01*FLOAT(JD(2,J-1))+FLOAT(JD(2,J)-JD(2,J-1))*AN*.31
0051      S=.01*FLOAT(JD(3,J-1))+FLOAT(JD(3,J)-JD(3,J-1))*AN*.31
0052      P=FLOAT(M)
0053      GOTO 250
0054      23 P=FLOAT(JD(1,J))
0055      T=FLOAT(JD(2,J))/100.
0056      S=FLOAT(JD(3,J))/100.
0057      250 CALL SIGAL(P,T,S,SIGMA,DALFA)
0058      D=D+(DI+DALFA)*(P-PI)*1.E+02/2.
0059      IF(ID(I)-M)10,35,10
```

```
0060 35 A(L,I)=D
0061      I=I+1
0062 10 IF(NR-M)73,24,73
0063 70 P1=P
0064      DI=DALFA
0065      M=M+1
0066      GOTO 19
0067 21 J=J+1
0068      GOTO 18
0069 24 DR(L)=D
0070      IF(L-2)75,87,83
0071 75 WRITE (2,6)
0072      6 FORMAT("ST. SUIVANTE ?")
0073      L=L+1
0074      GOTO 7
0075 80 WRITE(2,143)JS(1),JS(2)
0076 140 FORMAT(1X,/"ST"13" A "13" (+ A DROITE)"/" DR. (CM/5)"
0077      DL=SQRT((X(2)-X(1))**2+(Y(2)-Y(1))**2)
0078      PHI=(Y(1)+Y(2))*3.14159/360.
0079      F=DL*SIN(PHI)*22224.*0.729E-04
0080      I=1
0081 95 V=(DR(1)-DR(2)-A(1,I)+A(2,I))/F
0082      WRITE(2,120)ID(1),V
0083      I=I+1
0084      IF(I-N)95,90,90
0085 90 DR(1)=DR(2)
0086      X(1)=X(2)
0087      Y(1)=Y(2)
0088      JS(1)=JS(2)
0089      NN=N-1
0090      DO 85 J=1,NN
0091      A(1,J)=A(2,J)
0092 85 CONTINUE
0093      L=1
0094      GOTO 75
0095 27 WRITE(2,28)
0096 28 FORMAT("ERREUR DE REFERENCE")
0097      STOP
0098 120 FORMAT(16,4XF6.0)
0099 142 FORMAT(215,5X2(12,1H/),12,2X14,2XF5.2,A2,2XF6.0,10)
0100      END
0101      ENDS
```

**END-OF-TAPE

*

FICHE TECHNIQUE DU PROGRAMME

NOM: *JOE.R*

TYPE	CHAINE		
BUT	<i>Calcul de hauteurs dynamiques rapportées à une référence à fixer, et calcul des fonctions de transport (SAD d₃).</i>		
Observations			
Opérations			
DATE DE MISE AU POINT	<i>Janvier 74</i>		
LANGAGE UTILISE	<i>Fortran</i>		
SSP EMPLOYES			
Starting ADDRESS	<i>2</i>		
Nb Source	<i>1</i>	Nb Objét	<i>1</i> ... Nb Absolu <i>1</i>
Divers			
Clés employées			

PERIPH.	IN	<i>7.5.1</i>
	OUT	<i>2</i>
FORMAT	IN	<i>B</i>
	OUT	<i>A</i>
ETIQ.	IN	<i>M</i>
	OUT	<i>M</i>
PROGRAMMEUR		
<i>Citeau</i>		

FICHE EXPLOITATION

```
0001 PROGRAM JOBR
0002 DIMENSION ID(50),AC(4),T(50),JD(4,10)
0003 WRITE(2,135)
0004 READ(1,*)NR,JL
0005 I=1
0006 5 READ(JL,*)ID(I)
0007 N=I
0008 IF(ID(I)-1577)3,3,2
0009 3 I=I+1
0010 GOTO 5
0011 135 FORMAT("H.D.M. REFERENCE?, DIVISION?(I J 1 10) ")
0012 2 WRITE(2,135)
0013 NR=7
0014 138 FORMAT("ST. ?")
0015 7 READ(1,*)NF
0016 IF(NF-9999)39,41,41
0017 41 STOP
0018 39 CALL PTAPE(7,1,3)
0019 READ(NR,139)JR,MO,IA,IB,IC,IS,AT,LA,ON,LO
0020 IF(IS-NF)39,40,39
0021 139 FORMAT(3I3,15,16,14,F5.2,1XA2,F6.2,1XA2)
0022 40 WRITE(2,102)IC,IS,JR,MO,IA,IB,AT,LA,ON,LO
0023 D=0
0024 TR=0
0025 DI=0
0026 I=1
0027 M=3
0028 PI=0
0029 JI=1
0030 15 JI=JI+9
0031 READ(NR)((JD(K,J),K=1,4),J=JI,JI3)
0032 IF(JD(2,JI3))16,17,15
0033 16 JI=JI+10
0034 GOTO 15
0035 17 J=1
0036 18 IF(JD(2,J))27,27,19
0037 19 IF(M-JD(1,J))200,23,21
0038 200 AN=FLOAT(M-JD(1,J-1))/FLOAT(JD(1,J)-JD(1,J-1))
0039 T=.01*FLOAT(JD(2,J-1))+FLOAT(JD(2,J)-JD(2,J-1))*AN+.41
0040 S=.01*FLOAT(JD(3,J-1))+FLOAT(JD(3,J)-JD(3,J-1))*AN+.41
0041 P=FLOAT(M)
0042 GOTO 250
0043 23 P=FLOAT(JD(1,J))
0044 T=FLOAT(JD(2,J))/100.
0045 S=FLOAT(JD(3,J))/100.
0046 250 CALL SIGAL(P,T,S,SIGMA,DALFA)
0047 D=D+(DI+DALFA)*(P-PI)*1.E+12/0.
0048 TR=TR+D
0049 IF(ID(I)-M)10,35,10
0050 35 A(I)=D
0051 TR(I)=TR-D*.5
0052 I=I+1
0053 10 IF(NR-M)70,24,70
0054 70 PI=P
0055 DI=DALFA
0056 M=M+1
0057 GOTO 19
0058 21 J=J+1
0059 GOTO 18
```

```
0361      24 DREF=D
0361      WRITE(2,104)
0362      I=1
0363      T2(I)=A
0364      57 T2(I)=DREF*FLOAT(I*DC(I))-T2(I)
0365      A(I)=DREF-A(I)
0366      WRITE(2,225)I*DC(I),A(I),T2(I)
0367      I=I+1
0368      IF(I-I)2,2,53
0369      203 FORMAT(15,4NE2.3,3NF1.0)
0370      114 FORMAT(" 22. ", 2.15E)      *2.15E10*
0371      17 WRITE(2,22)
0372      3 FORMAT("F.21E, DE.2E, ", 2.15E)
0373      STOP
0374      122 FORMAT(215,5X(2(15,1E7),15,2NF1.0), 2E, 3NF, 2NF, 3NF)
0375      END
0375      ENDS
```

```
**END-OF-TAPE
*
/E
```

FICHE TECHNIQUE DU PROGRAMME

NOM: PYCNE

TYPE	CHAINE			PERIPH.	IN	7		
BUT	Calcul de P, T, S, O2 sur des valeurs de Δs au σt donnés par l'utilisateur.			OUT	2			
				FORMAT	IN	15		
				OUT	L			
Observations				ETIQ.	IN	5		
Opérations				OUT	L			
DATE DE MISE AU POINT	Oct. 74			PROGRAMMEUR				
LANGAGE UTILISE	FORTRAM			Mulline.				
SSP EMPLOYES	Sigma							
Starting ADDRESS	2							
Nb Source	1	Nb Objet	1				Nb Absolu	1
Divers								
Clés employées								

FICHE EXPLOITATION

Donner : σt par ordre croissant
 ou bien Δs par ordre décroissant

```
0001      PROGRAM PYCNE
0002      DIMENSION NST(100), ISO(10), JD(4,100)
0003      DIMENSION XP(10), XT(10), XS(10), XO(10)
0004      WRITE(2,100)
0005 100  FORMAT(///"PYCNE"// "STATIONS?")
0006      NS=0
0007      1  NS=NS+1
0008      READ(1,*) NST(NS)
0009      IF(NST(NS)-999)1,2,2
0010      2  WRITE(2,101)
0011 101  FORMAT("ISOPYCNES=1 DELTA ST=0 XXXX(MAY=1.0)")
0012      READ(1,*) NC
0013      WRITE(2,500)
0014 500  FORMAT("VALEURS ?")
0015      NT=0
0016      3  NT=NT+1
0017      READ(1,*) ISO(NT)
0018      IF(ISO(NT)-9999)3,4,4
0019      4  NS=NS-1
0020      NT=NT-1
0021      WRITE(2,102)(ISO(J),J=1,NT)
0022 102  FORMAT(10/6X1116)
0023      DO 7 J=1,12
0024      XP(J)=9999.
0025      XT(J)=0.
0026      XO(J)=0.
0027      7  XS(J)=0.
0028      WRITE(2,200)
0029 200  FORMAT(/)
0030      N=1
0031      REVIND7
0032      13 CALL PTAPE(7,1,0)
0033      11 READ(7,103) IS
0034 103  FORMAT(20X14)
0035      IF(NST(N)-IS)10,9,10
0036      9  L=1
0037      J1=1
0038 401  J10=J1+9
0039      READ(7)((JD(J,K),J=1,4),K=J1,J10)
0040      J1=J10+1
0041      IF(JD(2,J10))400,400,401
0042 400  DO 13 K=1,J10
0043      NPA=0
0044      IF(JD(2,K+1))50,50,14
0045      14 IF(L-NT)30,30,50
0046      30 T1=FLOAT(JD(2,K))/100.
0047      T2=FLOAT(JD(2,K+1))/100.
0048      S1=FLOAT(JD(3,K))/100.
0049      S2=FLOAT(JD(3,K+1))/100.
0050      O1=FLOAT(JD(4,K))/100.
0051      O2=FLOAT(JD(4,K+1))/100.
0052      P1=FLOAT(JD(1,K))
0053      P2=FLOAT(JD(1,K+1))
0054 302  P=P1+FLOAT(NPA)
0055      IF(P-P2)300,13,13
0056 300  T=(T1-T2)/(P1-P2)*(P-P1)+T1
0057      S=(S1-S2)/(P1-P2)*(P-P1)+S1
0058      O=(O1-O2)/(P1-P2)*(P-P1)+O1
0059      GOTO(71,72),NC
```



```
0060      71  IF(IFIX(SIGMA(T,S)*100.)-ISU(L))303,15,15
0061      72  A=SIGMA(T,S)*1.E-3
0062      A=(.02736-A/(1.+A))*1.E+5
0063      IF(IFIX(A)-ISU(L))15,15,303
0064      15  XP(L)=P
0065      XT(L)=T
0066      XS(L)=S
0067      XO(L)=0
0068      L=L+1
0069      303  NPA=NPA+1
0070      GOTO 302
0071      13  CONTINUE
0072      54  WRITE(2,105)IT,(XP(N1),N1=1,NT)
0073      WRITE(2,106) (XT(N2),N2=1,NT)
0074      WRITE(2,106) (XS(N2),N2=1,NT)
0075      105  FORMAT(14,1X11F6.0)
0076      106  FORMAT(6X11F6.2)
0077      WRITE(2,106)(XO(N2),N2=1,NT)
0078      WRITE(2,200)
0079      DO25 K=1,12
0080      XP(K)=9999.
0081      XT(K)=0.
0082      XO(K)=0.
0083      25  XS(K)=0.
0084      IF(N-NS)20,21,21
0085      20  N=N+1
0086      CALL PTAPE(7,1,0)
0087      GOTO 11
0088      21  WRITE(2,110)
0089      110  FORMAT(10/)
0090      REWIND7
0091      END
0092      ENDS
```

**END-OF-TAPE

*
/E

*END
*NEXT?
:PAUSE

FICHE TECHNIQUE DU PROGRAMME

NOM: **GRAD.R**

TYPE		CHAINE		PERIPH.	IN	7
BUT	Calcul des gradients de T, S, O, Δst, et des dérivées secondes de T et S.				OUT	2
Observations	le calcul sera effectué à tous les niveaux réduits, entre des bornes à fixer.			FORMAT	IN	B
Opérations					OUT	A
DATE DE MISE AU POINT	Novembre 1974			ETIQ.	IN	M
LANGAGE UTILISE	FORTRAN				OUT	M
SSP EMPLOYES	SIGMA			PROGRAMMEUR		
Starting ADDRESS	2			CITEAU		
Nb Source	1	Nb Objet	1 ...	Nb Absolu		
Divers						
Clés employées						

FICHE EXPLOITATION

** GRADIENT **

LIMITES D'IMMERSION ***?,***?

10,20

STATION ?

1

CRUISIERE: 7701

STATION : 1

POSITION : .24'5 N 3.57'8 W

PROF.*TEMP.*SALIN*OXY.* Z1 * D1T * D1S * D1O *D1DST* Z2 * D2T * D2S *
(X 100) (X 1000)

10 27.47 35.12 .00

13 27.47 35.12 .00 11 0 0 0 .00

15 27.47 35.12 .00 14 0 0 0 .00 12 0 0

18 27.46 35.12 .00 16 0 0 0 -.10 15 -1 0

STATION ?

20

CRUISIERE: 7701

STATION : 20

POSITION : 1.59'5 S 4.00'9 W

PROF.*TEMP.*SALIN*OXY.* Z1 * D1T * D1S * D1O *D1DST* Z2 * D2T * D2S *
(X 100) (X 1000)

16 26.81 35.68 .00

20 26.66 35.76 .00 18 -3 2 0 -2.56

STATION ?

Y

```
PROGRAM GRADR
DIMENSION ID(4,110),Z(2),DT(2),DN(2),ST(3),MET(5)
107 FORMAT("** GRADIENT **"/" LIMITES D'IMMERSION ***?,***?")
121 FORMAT(" STATION ?")
8 FORMAT(3I3,I5,I3,I4,F5.2,I10,F5.2,I10,F5.2)
102 FORMAT(2/"CROISIERE:"I5,29X"DATE:"3I3/"STATION : "I5,
129X"HEURE:"I8/"POSITION : "F5.2,I10,I10,3XF5.2,I10,I10,14A20)
300 FORMAT("PROF.*TEMP.*SALIN*OXY.* ZI * DIT * DIS * DIO *DID*DT*
1 32 * D25 * D25 *"/27X("5X"100"7X")"13X"( " K 1000 ")"/)
303 FORMAT(I4,2XF5.2,1XF5.2,1XF4.1)
302 FORMAT(I4,1X,2(1XF5.2),1XF4.2,1XI4,3(1XI5),F5.2)
301 FORMAT(I4,1X,2(1XF5.2),1XF4.2,1XI4,3(1XI5),F5.2,1XI4,2(1XI5))
50 FORMAT(5(3I4,I3))
WRITE(2,133)
READ(1,*)IM1,IM2
5 WRITE(2,131)
READ(1,*)IS0
10 READ(5,8)(MET(N),N=1,6),AT,IA1,IA2,ON,I01,I02
IF(MET(6)-9999)14,21,21
21 PAUSE
14 IF(MET(6)-IS0)17,11,10
11 WRITE(2,132)MET(5),MET(1),MET(2),MET(3),MET(6),MET(4),
IA1,IA2,ON,I01,I02
WRITE(2,300)
J1=1
15 J10=J1+9
READ(5,57)((ID(I,J),I=1,4),J=J1,J10)
IF(ID(2,J10))13,30,13
13 J1=J10+1
GOTO 15
30 I=1
K=1
L=1
31 IF(ID(1,I)-IM1)32,35,35
32 I=I+1
GOTO 31
35 IF(ID(1,I)-IM2)36,36,5
36 IF(ID(2,I))38,5,38
38 T=FLOAT(ID(2,I))*01
S=FLOAT(ID(3,I))*01
O=FLOAT(ID(4,I))*01
A=.001*SIGMA(T,S)
ST(K)=(.02736-A/(1.+A))*1.E+5
GOTO(41,42,42),K
41 WRITE(2,303)ID(1,I),T,S,0
K=2
I=I+1
GOTO 35
```

```
42 Z(L)=FLOAT(ID(1,I)+ID(1,I-1))/Z
  IZ=IFIX(Z(L))
  DT(L)=FLOAT(ID(2,I)-ID(2,I-1))/FLOAT(ID(1,I)-ID(1,I-1))
  DS(L)=FLOAT(ID(3,I)-ID(3,I-1))/FLOAT(ID(1,I)-ID(1,I-1))
  DO=FLOAT(ID(4,I)-ID(4,I-1))/FLOAT(ID(1,I)-ID(1,I-1))
  DDST=(ST(K)-ST(K-1))/FLOAT(ID(1,I)-ID(1,I-1))
  IDT=IFIX(DT(L))
  IDS=IFIX(DS(L))
  IDO=IFIX(DO)
  IF(K-2)41,44,43
44 WRITE(2,312)ID(1,I),T,S,0,I2,IDT,IDS,IDO,DDST
  K=3
  L=2
  I=I+1

  GOTO 35
43 IZ2=IFIX((Z(1)+Z(2))*5)
  ID2T=IFIX(17.*(DT(1)-DT(2))/(Z(1)-Z(2)))
  ID2S=IFIX(18.*(DS(1)-DS(2))/(Z(1)-Z(2)))
  WRITE(2,331)ID(1,I),T,S,0,IZ2,IDT,IDS,IDO,DDST,IZ2,ID2T,ID2S
  Z(1)=Z(2)
  DT(1)=DT(2)
  DS(1)=DS(2)
  ST(2)=ST(3)
  I=I+1

  GOTO 35
  END
ENDS
```



```

0001 PROGRAM GRADS
0002 DIMENSION ID(4,110),Z(2),DT(2),DS(2),ST(3),MET(6)
0003 100 FORMAT("** GRADIENT **"/" LIMITES D'IMMERSION ***?***?")
0004 101 FORMAT(" STATION ?")
0005 3 FORMAT(3I3,I5,I6,I4,F5.2,I1,A2,F6.2,I1,A2)
0006 102 FORMAT(2/"CROISIERE:"I5,29X"DATE:"3I3/"STATION : "I5,
0007 129X"HEURE:"I8/"POSITION : "F5.2,1I1,I1,1XA2,3XF6.2,1I1,I1,1XA2)
0008 300 FORMAT("PROF.*TEMP.*SALIN*OXY.* Z1 * D1T * D1S * D1C *DIS*
0009 1 Z2 * D2T * D2S *"/27X("5X"X 100"7X")"10X"( X 1000 )"/)
0010 303 FORMAT(I4,2XF5.2,1XF5.2,1XF4.2)
0011 302 FORMAT(I4,1X,2(1XF5.2),1XF4.2,1XI4,3(1XI5),F6.3)
0012 301 FORMAT(I4,1X,2(1XF5.2),1XF4.2,1XI4,3(1XI5),F6.3,1XI4,2(1XI5))
0013 50 FORMAT(5(3I4,I3))
0014 95 FORMAT(3A1)
0015 99 FORMAT("DERIVEE SEC. ?")
0016 WRITE(2,100)
0017 READ(1,*)IM1,IM2
0018 WRITE(2,99)
0019 READ(1,95)IR,KT,KT
0020 5 WRITE(2,101)
0021 READ(1,*)IS0
0022 10 READ(5,8)(MET(N),N=1,6),AT,IA1,IA2,GN,IO1,IO2
0023 IF(MET(6)-9999)14,21,21
0024 21 PAUSE
0025 14 IF(MET(6)-IS0)10,11,10
0026 11 WRITE(2,102)MET(5),MET(1),MET(2),MET(3),MET(6),MET(4),
0027 1AT,IA1,IA2,GN,IO1,IO2
0028 WRITE(2,300)
0029 J1=1
0030 15 J10=J1+9
0031 READ(5,50)((ID(I,J),I=1,4),J=J1,J10)
0032 IF(ID(2,J10))13,30,13
0033 13 J1=J10+1
0034 GOTO 15
0035 30 I=1
0036 K=1
0037 L=1
0038 31 IF(ID(1,I)-IM1)32,35,35
0039 32 I=I+1
0040 GOTO 31
0041 35 IF(ID(1,I)-IM2)36,36,500
0042 36 IF(ID(2,I))500,500,38
0043 38 T=FLOAT(ID(2,I))*01
0044 S=FLOAT(ID(3,I))*01
0045 C=FLOAT(ID(4,I))*01
0046 ST(K)=SIGMA(T,S)
0047 GOTO(41,42,42),K
0048 41 WRITE(2,303)ID(1,I),T,S,0
0049 K=2
0050 I=I+1
0051 GOTO 35
0052 42 Z(L)=FLOAT(ID(1,I)+ID(1,I-1))/2.
0053 IZ=IFIX(Z(L))
0054 DT(L)=FLOAT(ID(2,I)-ID(2,I-1))/FLOAT(ID(1,I)-ID(1,I-1))
0055 DS(L)=FLOAT(ID(3,I)-ID(3,I-1))/FLOAT(ID(1,I)-ID(1,I-1))
0056 DO=FLOAT(ID(4,I)-ID(4,I-1))/FLOAT(ID(1,I)-ID(1,I-1))
0057 DDST=(ST(K)-ST(K-1))/FLOAT(ID(1,I)-ID(1,I-1))
0058 IDT=IFIX(DT(L))
0059 IDS=IFIX(DS(L))

```

```
0060      IDO=IFIX(DO)
0061      IF(K-2)41,44,43
0062  44  WRITE(2,302)ID(1,I),T,S,O,IZ,IDT,IDS,IDO,DDST
0063      K=3
0064      L=2
0065      I=I+1
0066      GOTO 35
0067  43  IF(IR-1175)440,45,440
0068  440  ST(2)=ST(3)
0069      GOTO 44
0070  45  IZ2=IFIX((Z(1)+Z(2))*0.5)
0071      ID2T=IFIX(10.*(DT(1)-DT(2))/(Z(1)-Z(2)))
0072      ID2S=IFIX(10.*(DS(1)-DS(2))/(Z(1)-Z(2)))
0073  67  WRITE(2,301)ID(1,I),T,S,O,IZ,DET,IES,IEO,ELST,IZP,LEDT,LI
0074  70  Z(1)=Z(2)
0075      DT(1)=DT(2)
0076      DS(1)=DS(2)
0077      ST(2)=ST(3)
0078      I=I+1
0079      GOTO 35
0080  500  CALL CLE(KC)
0081      IF(KC)501,501,5
0082  501  IS0=IS0+1
0083      GOTO 10
0084      EVD
0085      EVD$
```

**EVD- CF- TAPE

*

FICHE TECHNIQUE DU PROGRAMME

NOM: DCCDA

TYPE					CHAINE				
BUT	<i>traitement de stations couantométriques AA au mouillage. permet de passer du code FFI-PS à des données en vraie grandeur binaire. EFFECTUE DES MOYENNES.</i>						PERIPH.	IN	S
								OUT	7,4
Observations	<i>Opérations moyenne mobile sur 10 cycles.</i>						FORMAT	IN	K
								OUT	1,1
Observations							ETIQ.	IN	M AA
Opérations								OUT	M AA
DATE DE MISE AU POINT	1976						PROGRAMMEUR		
LANGAGE UTILISE	FORTRAN + SP ASSEMBLER						CITEAU		
SSP EMPLOYES	CLE, ANDER, SAL								
Starting ADDRESS	2655								
Nb Source	/	Nb Objet	/	Nb Absolu	/				
Divers									
Clés employées									

FICHE

EXPLOITATION

voir note technique

DCCDA

CUT, DECL. ?

2, -10

MCY SUR .. CYCLES ?

1

CYCLES: DEBUT, FIN

1, 12

CR..STA**...**HEUR.....N.....E COUR PEND

7701001210177123001000W004000W 2220 30

7701 1 21/ 1/77 1230 100.0'0W 400.00'0W 2220 30 CORRECT ?

CUi

LIBRAIRIE 1 CU 5 ?

5

PAUSE

7701 121 1771230 1000W 4000W2220 30 1

REV. CY. PR. TEMP SAL. U V DIR Vi T COWD

3302 5014 3939 1294 -8561

585 1301 1854 -1 -40

596 1276 1865 -1 -40

607 1257 1874 3 -40

618 1245 1880 4 -34

627 1226 1890 8 -34

641 1201 1902 11 -33

652 1185 1910 12 -32

665 1173 1916 14 -31

676 1163 1920 17 -30

687 1151 1926 16 -24

694 1141 1931 16 -24

MCY SUR .. CYCLES ?

71

*
/L
/F

G-98

SYMECLIC FILE SOURCE DEVICE?

/P

```

0001      PROGRAM DCODA
0002      DIMENSION SGM(5),IF(5),L(15)
0003      1 FORMAT('DCODA"/"OUT,DECL.?')
0004      7 FORMAT('CF..STA**...**HEUF.....N.....E CODE FENI')
0005      99 FORMAT('M CY SUR .. CYCLES ?')
0006      2 FORMAT('CYCLES: DEEUT,FIN')
0007      3 FORMAT(I4,I3,3I2,I4,I4,I1,A1,I5,I1,A1,2I4,I8)
0008      16 FORMAT(I4,I3,3I2,2I4,I1,A1,I5,I1,A1,6(I4))
0009      5 FORMAT(I4,1XI3,1XI2,2(1H/I2),1XI4,1XI5,2,1H/I1,A1,1XI6,2,
0010      11H/I1,A1,2XI4,1XI4,13H      COEFFCT ?)
0011      6 FORMAT(3A1)
0012      IFEF=0
0013      150 WRITE(2,1)
0014      READ(1,*)IOUT,IECL
0015      23 IF(IOUT-4)15,13,13
0016      13 CALL LEADR(4,10)
0017      15 WRITE(2,99)
0018      READ(1,*)NOMER
0019      GMER=FLOAT(NOMER)
0020      36 WRITE(2,2)
0021      READ(1,*)IDEE,IFIN
0022      IF(IFIN+1-IDEE-NOMER)36,4,4
0023      4 WRITE(2,7)
0024      READ(1,16)(L(i),i=1,14)
0025      AL=FLOAT(L(7))/100.
0026      CL=FLOAT(L(10))/100.
0027      DECAL=(FLOAT(IDEE)+0.5*GMER-1.5)*FLOAT(L(14))/60.
0028      TMIN=FLOAT(L(6)-40*(L(6)/100))+IECAL
0029      JH=IFIX(TMIN/60.)
0030      TMIN=TMIN-60.*FLOAT(JH)
0031      52 IF(JH-24)53,51,51
0032      51 L(3)=L(3)+1
0033      JH=JH-24
0034      GOTO 52
0035      53 JM=IFIX(TMIN)
0036      L(6)=JH*100+JM
0037      IF(L(3)-26)69,69,85
0038      65 WRITE(2,555)
0039      555 FORMAT('M C I S DE .. JOURS')
0040      READ(1,*)JL
0041      IF(L(3)-JL)69,69,70
0042      70 L(3)=L(3)-JL
0043      L(4)=L(4)+1
0044      IF(L(4)-12)69,69,86
0045      86 L(4)=1
0046      L(5)=L(5)+1
0047      69 WRITE(2,5)(L(i),i=1,14)
0048      READ(1,6)IF,KT,KT
0049      IF(IF-117E)4,8,4
0050      8 IF(IFEF-L(13))21,17,21
0051      21 WRITE(2,88)
0052      88 FORMAT('L I E P A I R I E 1 O U 5 ?')
0053      READ(1,*)ILIE
0054      65 READ(ILIE,*)IREF,A1,F,C1,I1
0055      READ(ILIE,*)E,F,G,H,CT,DF,FEV
0056      IF(IREF-L(13))65,17,65
0057      17 PAUSE
0058      ICY=0
0059      L(15)=NOMER

```

G-99

```

0060      WRITE(I CUT, 3) L
0061      WRITE(2, 204)
0062      DO 66 I=1, 5
0063      66 SUM(I)=0.
0064      DO 769 J=1, NCMER
0065      15 CALL ANIFF(I 1, I 2, I 3, I 4, I 5, I 6, I FF, I 8)
0066      IF(I 8-1) 19, 60, 60
0067      19 ICY=ICY+1
0068      20 IF(IDEF-ICY) 24, 24, 16
0069      24 IF(ICY-IFIN) 67, 67, 60
0070      67 T=FLCAT(I 2)
0071      T=A1+T*(F+T*(C1+T*11))
0072      PF=(G+H*FLCAT(I 4))/1.087
0073      C=F+F*FLCAT(I 3)
0074      IP(3)=IFIX(SAL(PF, T, C)*100.)
0075      IP(2)=IFIX(T*100.)
0076      IIF=FLCAT(I 5)*DF+FFCL+CT
0077      IF(DIF-360.) 870, 152, 6+0
0078      660 IIF=IIF-360.
0079      GOTO 152
0080      870 IF(DIF) 890, 152
0081      890 IIF=360.+IIF
0082      152 VIT=1.5+FLCAT(I 6)*42.*FEV/FLCAT(L(14))
0083      DI=(90.-IIF)*3.1416/160.
0084      IP(1)=IFIX(PF)
0085      IP(4)=IFIX(VIT*CGS(DI))
0086      IP(5)=IFIX(VIT*SIN(DI))
0087      I3=IFIX(C*100.)
0088      I5=IFIX(DIR)
0089      I6=IFIX(VIT)
0090      CALL CLE(KC)
0091      IF(IIF+KC) 320, 320, 46
0092      46 WRITE(2, 200) I 1, ICY, (IP(I), I=1, 5), I 5, I 6, I 3, I FF
0093      200 FORMAT(11I 5)
0094      204 FORMAT(" REV. CY. PF. TEMP SAL. U V IIF VIT CGS")
0095      320 DO 769 I=1, 5
0096      769 SUM(I)=SUM(I)+FLCAT(IP(I))
0097      IP(I)=IFIX(SUM(I)/CMER)
0098      IF(ICUT-4) 319, 321, 321
0099      319 WRITE(ICUT, 101) IP
0100      GOTO 82
0101      321 WRITE(ICUT) IP
0102      GOTO 82
0103      101 FORMAT(5I 6)
0104      60 IF(ICUT-4) 15, 49
0105      49 I4=0
0106      WRITE(ICUT) I 4, I 4, I 4, I 4, I 4
0107      50 WRITE(2, 100)
0108      100 FORMAT("SUITE=1 FIN=0")
0109      READ(1, *) KC
0110      IF((KC-1)*(3-ICUT)) 150, 15, 10
0111      10 WRITE(ICUT, 31)
0112      31 FORMAT(36(1H9))
0113      CALL LEADF(4, 10)
0114      GO TO 150
0115      END
0116      ENDS

```

SSP ANDER: 1^{me} Version
 à 6 paramètres, 6 test d'erreur
 et 1 test de Synchronisation

G-100

0001		NAM	ANDER	0058		SZA
0002		ENT	ANDER	0059		JMP SUIT1
0003		EXT	.ENTR, ENDIO	0060		LDA ERROR
0004	IAA	BSS	13	0061		STA FAUTE
0005	ANDER	NOP		0062		JSB CARAC
0006		JSB	.ENTR	0063		JMP MOT, I
0007		DEF	IAA	0064	SUIT1	LDA X
0008		JSB	ENDIO	0065		STA PARA
0009		DEF	*+1	0066		JSB CARAC
0010		CLA		0067		LDA CA6
0011		STA	FAUTE	0068		SZA, RSS
0012		STA	SYNC	0069		JMP SUIT3
0013		STA	IAA+12, I	0070		LDA ERROR
0014		LDA	M6	0071		STA FAUTE
0015		STA	CNTR	0072	SUIT3	LDA CA7
0016	BLANC	JSB	CARAC	0073		SZA, RSS
0017		LDA	X	0074		JMP SUIT2
0018		SZA,	RSS	0075		LDA ERROR
0019		JMP	BLANC	0076		STA FAUTE
0020		LDA	ADD	0077		JMP MOT, I
0021		STA	MOT	0078	SUIT2	LDA X
0022		LDA	IAA	0079		ALF, ALS
0023		STA	N	0080		ADA PARA
0024		JMP	MOT+2	0081		STA PARA
0025	AGAIN	JSB	MOT	0082		JMP MOT, I
0026	DEBUT	LDA	PARA	0083	CARAC	NOP
0027		LDB	N	0084		CLF 0
0028		STA	I, I	0085		STC 15B, C
0029		INB		0086		SFS 15B
0030		STB	N	0087		JMP *-1
0031		LDA	FAUTE	0088		LIB 15B
0032		LDB	N	0089		LDA I
0033		STA	I, I	0090		AND =B37
0034		CLA		0091		STA X
0035		STA	FAUTE	0092		LDA I
0036		INB		0093		AND =B40
0037		STB	N	0094		STA CA6
0038		ISZ	CNTR	0095		LDA I
0039		JMP	SYN	0096		AND =B100
0040		LDA	SYNC	0097		STA CA7
0041		SZA		0098		CLC 15B
0042		JMP	FIN	0099		JMP CARAC, I
0043		LDA	ERROR	0100	ADD	DEF DEBUT
0044		STA	IAA+12, I	0101	X	BSS I
0045		JMP	FIN	0102	CA6	BSS I
0046	SYN	LDA	SYNC	0103	CA7	BSS I
0047		SZA		0104	PARA	BSS I
0048		JMP	FIN	0105	CNTR	BSS I
0049		JMP	AGAIN	0106	ERROR	OCT I
0050	FIN	JSB	ENDIO	0107	M6	DEC -6
0051		DEF	*+1	0108	FAUTE	BSS I
0052		JMP	ANDER, I	0109	N	BSS I
0053	MOT	NOP		0110	SYNC	BSS I
0054		JSB	CARAC			
0055		LDA	CA6			
0056		STA	SYNC			
0057		LDA	CA7	0111		END ANDER

NOM: DCOD B

FICHE TECHNIQUE DU PROGRAMME

TYPE	CHAINE			PERIPH.	IN	5
BUT	même chose que DCODA - Pas de moyenne - Fiche antiparasite du type JIVAR			FORMAT	OUT	4, 7, 2
Observations				FORMAT	IN	B
Opérations	effectue un "lissage Antiparasite"			FORMAT	OUT	A, B
DATE DE MISE AU POINT	1976			ETIQ.	IN	#1AA
LANGAGE UTILISE	FORTRAN + SP ASSEMBLER			ETIQ.	OUT	M AA
SSP EMPLOYES	CLE, ANGER, SAL			PROGRAMMEUR		
Starting ADDRESS	2753			CITEAU		
Nb Source	/	Nb Objet	/			
				Nb Absolu	/	
Divers						
Clés employées						

FICHE

EXPLOITATION

VOIR NOTE TECHNIQUE

```
0001      PROGRAM DCOB
0002      DIMENSION JD(5,60),L(14)
0003      1 FORMAT("DCODB"/"OUT,DECL.?" )
0004      7 FORMAT("CR..STA**MO**HEUR.....N.....E COUR PEND")
0005      2 FORMAT("CYCLES: DEBUT,FIN")
0006      3 FORMAT(I4,I3,3I2,I4,I4,I1,A1,I5,I1,A1,2I4)
0007      16 FORMAT(I4,I3,3I2,2I4,I1,A1,I5,I1,A1,2(1X14))
0008      5 FORMAT(I4,1X13,1X12,2(1H/I2),1X14,1XF5.2,1H'I1,A1,1XF5.2,
0009      11H'I1,A1,2(1X14),13H      CORRECT ?)
0010      6 FORMAT(3A1)
0011      200 FORMAT(I3,A1,I4,3(I4,A1),I4,1X,2(I3,A1),2I4,2A1)
0012      101 FORMAT(8X14,1X14,6X14,9X,2I4)
0013      31 FORMAT(38(1H8))
0014      32 FORMAT(38(1H9))
0015      IREF=0
0016      150 WRITE(2,1)
0017      READ(1,*)IOUT,DECL
0018      23 IF(IOUT-4)15,13,14
0019      13 CALL LEADR(4,10)
0020      GOTO 15
0021      14 REWIND 7
0022      WRITE(2,77)
0023      77 FORMAT("FILE ?")
0024      READ(1,*)IFIL
0025      IF(IFIL-1)11,10,11
0026      10 WRITE(7,31)
0027      34 ENDFILE 7
0028      GOTO 15
0029      11 CALL PTAPE (7,IFIL,0)
0030      15 WRITE(2,2)
0031      READ(1,*)IDEB,IFIN
0032      4 WRITE(2,7)
0033      READ(1,16)(L(I),I=1,14)
0034      AL=FLOAT(L(7))/100.
0035      OL=FLOAT(L(10))/100.
0036      DECAL=FLOAT(IDEB)*FLOAT(L(14))/60.
0037      L(6)=60*(L(6)/100)+L(6)-100*(L(6)/100)+IFIX(DECAL)
0038      JH=L(6)/60
0039      JM=L(6)-60*JH
0040      52 IF(JH-24)53,51,51
0041      51 L(3)=L(3)+1
0042      JH=JH-24
0043      GOTO 52
0044      53 L(6)=JH*100+JM
0045      IF(L(3)-28)69,69,85
0046      85 WRITE(2,555)
0047      555 FORMAT("MOIS DE .. JOURS")
0048      READ(1,*)JL
0049      IF(L(3)-JL)69,69,70
0050      70 L(3)=L(3)-JL
0051      L(4)=L(4)+1
0052      IF(L(4)-12)69,69,86
0053      86 L(4)=1
0054      L(5)=L(5)+1
0055      69 WRITE(2,5)(L(I),I=1,14)
0056      READ(1,6)IR,KT,KT
0057      IF(IR-117B)4,8,4
0058      8 IF(IREF-L(13))21,17,21
0059      21 WRITE(2,88)
```

```
0060      88 FORMAT("LIBRAIRIE (5)")
0061      PAUSE
0062      63 READ(5,*)IREF,A,B,C1,D,TR,E,F,G,H,REV
0063      IF(IREF-L(13))63,64,63
0064      64 PAUSE
0065      17 ICY=0
0066      KD=IFIX(TR/FLOAT(L(14)))
0067      WRITE(IOUT,3)(L(I),I=1,14)
0068      WRITE(2,204)
0069      204 FORMAT("REF. CY. PR. TEMP COND SAL. DIR VIT  U  V SYNC")
0070      J=0
0071      J0=2
0072      J1=1
0073      93 J2=50+KD
0074      90 CALL ANDER(I1,L1,I2,L2,I3,L3,I4,L4,I5,L5,I6,L6,L7)
0075      ICY=ICY+1
0076      IF(ICY-IDEB)90,91,91
0077      91 J=J+1
0078      L(1)=L1
0079      L(2)=L2
0080      L(3)=L3
0081      L(4)=L4
0082      L(5)=L5
0083      L(6)=L6
0084      L(7)=L7
0085      IER=0
0086      DO 22 I=1,7
0087      IF(L(I)-1)24,25,25
0088      24 L(I)=040B
0089      GOTO 22
0090      25 L(I)=105B
0091      IER=1
0092      22 CONTINUE
0093      T=FLOAT(I2)
0094      T=A+B*T+C1*T*T+D*T*T*T
0095      PR=(G+H*FLOAT(I4))/1.027
0096      C=E+F*FLOAT(I3)
0097      S=SAL(PR,T,C)
0098      DIR=FLOAT(I5)*0.352+DECL
0099      IF(DIR)300,301
0100      300 DIR=DIR+360.
0101      301 VIT=1.5+FLOAT(I6)*42.*REV/FLOAT(L(14))
0102      DI=(90.-DIR)*3.1416/180.
0103      JD(1,J)=IFIX(PR)
0104      JD(2,J)=IFIX(T*100.)
0105      JD(3,J)=IFIX(S*100.)
0106      JD(4,J)=IFIX(VIT*COS(DI))
0107      JD(5,J)=IFIX(VIT*SIN(DI))
0108      I3=IFIX(C)
0109      I5=IFIX(DIR)
0110      I6=IFIX(VIT)
0111      CALL CLE(KC)
0112      IF(KC)50,98,48
0113      98 IF(IER)80,80,48
0114      48 WRITE(2,200)I1,L(1),ICY,JD(1,J),L(4),JD(2,J),L(2),I3,L(3),
0115      1JD(3,J),I5,L(5),I6,L(6),JD(4,J),JD(5,J),L(7),L(7)
0116      80 IF(IFIN-ICY)99,99,320
0117      99 KC=-1
0118      GOTO 50
0119      320 IF(J-J2)90,50,50
```

↑
attention ceci est la première
version de ANDER avec identification
des erreurs sur chaque paramètre.

```
0120      50 J2=J-1
0121      DO 96 NF=1,2
0122      DO 96 I=1,5
0123      DO 96 J=J0,J2
0124      K=J*(2-NF)+(J2+2-J)*(NF-1)
0125      IG1=JD(I,K)-JD(I,K-1)
0126      IG2=JD(I,K+1)-JD(I,K)
0127      IF(IG1*IG2)97,96,96
0128      97 JD(I,K)=(JD(I,K-1)+JD(I,K+1))/2
0129      96 CONTINUE
0130      IF(KC)81,82
0131      81 J2=J2+1
0132      82 J2=J2-KD
0133      DO 83 J=J1,J2
0134      J3=J+KD
0135      IF(IOUT-4)319,321
0136      319 WRITE(IOUT,101)JD(1,J),JD(2,J3),JD(3,J),JD(4,J),JD(5,J)
0137      GOTO 83
0138      321 WRITE(IOUT)JD(1,J),JD(2,J3),JD(3,J),JD(4,J),JD(5,J)
0139      83 CONTINUE
0140      100 IF(KC)60,61
0141      61 J4=2+KD
0142      DO 62 I=1,5
0143      DO 62 J=1,J4
0144      J3=J+48
0145      62 JD(1,J)=JD(1,J3)
0146      J0=2+KD
0147      J1=2
0148      J=J4
0149      GOTO 93
0150      60 IF(IOUT-4)15,49
0151      49 I4=0
0152      WRITE(IOUT)I4,I4,I4,I4,I4
0153      IF(IOUT-7)30,33
0154      33 ENDFILE 7
0155      30 PAUSE
0156      CALL CLE(KC)
0157      IF(KC)54,15
0158      54 WRITE(IOUT,32)
0159      IF(IOUT-4)15,13,14
0160      END
0161      ENDS
```

**END-OF-TAPE

*

NOM: **DCODP**

FICHE TECHNIQUE DU PROGRAMME

TYPE	CHAINE			PERIPH.	IN	5
BUT	traite "automatiquement" les profils en reconnaissant le début de la descente et l'arrivée sur l'anoteur.			OUT	4, 2	
Observations				FORMAT	IN	B
Opérations	on ouvre que les données enregistrées à la descente des Profils.			OUT	A, B	
DATE DE MISE AU POINT	1976			ETIQ.	IN	M _{AA}
LANGAGE UTILISE	FORTRAN			OUT	M _{AA}	
SSP EMPLOYES	ANDER, CLE, SAL			PROGRAMMEUR		
Starting ADDRESS	5133			CITSAU		
Nb Source	Nb Objet		Nb Absolu			
Divers						
Clés employées						

FICHE

EXPLOITATION

VOIR NOTE TECHNIQUE

- en début d'opération il faut remplir en respectant le format l'étiquette de la station.
- introduire ensuite au lecteur optique le ruban contenant les librairies des courantomètres. L'ordinateur se mettra en PAUSE 7777 quand il aura identifié la bonne librairie. Par la suite si l'on traite une série d'enregistrements du même courantomètre, cette étape sera évitée.
- à la question Z MAX? répondre simplement OUI ou NON car des remontées accidentelles peuvent se produire dans un profil insuffisamment lesté.
- ~~Demander~~ Demander (1) un listing au niveau de référence,
- introduire (2) direction et vitesse pour effectuer la correction.


```

DIMENSION JD(5,300),L(14)
1  FORMAT("DECOD P: OUT,DECL?")
7  FORMAT("CR..STA**MO**HEUR.....N.....TCOUR")

2  FORMAT("LIST=1,DE Z1,A Z2"/"-DERIV=2,DIR,VIT   FIN=3,")
16  FORMAT(I4,I3,3I2,2I4,I1,A1,I5,I1,A1,2I4)
204  FORMAT(I3,1X,I4,4(I4,1X),3(I3,1X),"E")
131  FORMAT(8X14,1X14,6X14,8X,2(I14))
32  FORMAT(38(1H9))
    IREF=0
154  WRITE(2,1)
    READ(1,*)IOUT,DECL
23  IF(IOUT-4)4,13
13  CALL LEADR(4,10)
4  WRITE(2,7)
    READ(1,16)(L(I),I=1,13)
    IF(IREF-L(13))21,17,21
21  WRITE(2,88)
88  FORMAT("LIBRAIRIE(5)")
    PAUSE
63  READ(5,*)IREF,A1,B,C1,D1
    READ(5,*)E,F1,G,H,CT,DR,REV
    IF(IREF-L(13))63,17,63
17  L(14)=30
    WRITE(IOUT,16)L
    PAUSE7777
26  WRITE(2,204)
204  FORMAT("REF. CY. PR. TEMP COND SAL. DIR VIT U V")
    ICY=3
    J=2
97  CALL ANDER(11,12,13,14,15,16,17,18)
    ICY=ICY+1
    PR=(G+H*FLOAT(I4))/1.027
    IF(I8-1)542,777,777
542  I4=IFIX(PR)
    IF(I4)99,893,37
99  IF(J-1)90,90,41
890  IF(J-1)36,36,41
37  IF(J-1)36,39
37  IF(I4-JD(1,J))38,90,38

```

```
38 J=J+1
   IF(301-J)41,41,41
36 J=1
47 T=FLOAT(I2)
   T=A1+T*(B+T*(C1+T*D1))
   C=E+F1*FLOAT(I3)
   JD(4,J)=IFIX(FLOAT(I5)*DR+DECL+CT)
   JD(5,J)=IFIX(1.5+1.4*REV*FLOAT(I5))
   IF(JD(4,J)-363)87,93,86
86 JD(4,J)=JD(4,J)-363
   GO TO 93
777 NP=J
   GO TO 421
87 IF(JD(4,J))89,93
89 JD(4,J)=363+JD(4,J)
93 JD(1,J)=I4
   JD(2,J)=IFIX(T*133.)
   JD(3,J)=IFIX(133.*SAL(PR,T,C))
   I3=IFIX(133.*C)
94 CALL CLE(KC)
   IF(KC+17)92,92,48
48 WRITE(2,200)I1,ICY,JD(1,J),JD(2,J),I3,JD(3,J),JD(4,J),
 1JD(5,J),I7
82 IF(J-3)93,35
35 IF(JD(1,J)-JD(1,J-2))41,93
41 IF(JD(1,J-2)-JD(1,J-1))80,42,42
80 NP=J-1
   GO TO 421
42 NP=J-2
421 MAXZ=JD(1,NP)
   WRITE(2,33)MAXZ
33 FORMAT("7 MAX="I4"??")
   READ(1,55)IR,KT,KT
55 FORMAT(3A1)
   IF(IR-117B)90,53,90
53 WRITE(2,2)
   READ(1,*)IREP,D,V
   GO TO(95,6,61),IREP
96 D=FLOAT(MAXZ)-10.
95 DO 137 J=1,NP
   IF(JD(1,J)-IFIX(D))107,47
47 IF(IFIX(V)-JD(1,J))53,48
48 WRITE(2,103)(JD(1,J),I=1,5)
103 FORMAT(8X14,1X14,6X14,214)
107 CONTINUE
49 IF(IFIX(D)-MAXZ)53,96
5 IF(D-180.)410,420
410 D=D+180.
```

```
GO TO 43
42* D=D-180.
43 U=V*COS((90.-D)*.0174532)
V=V*SIN((90.-D)*.0174532)
92 DO 91 J=1, NP
DI=(90.-FLOAT(JD(4,J)))*.0174532
JD(4,J)=IFIX(U+FLOAT(JD(5,J))*COS(DI))
JD(5,J)=IFIX(V+FLOAT(JD(5,J))*SIN(DI))
53* IF(IOUT-4) 312, 321
312 WRITE(IOUT,131)(JD(I,J), I=1,5)
GO TO 84
321 WRITE(IOUT)(JD(I,J), I=1,5)
84 IF(JD(1,J)-MAY7) 91, 31, 91
91 CONTINUE
51 IF(IOUT-4) 33, 46
46 I4=0
WRITE(IOUT) I4, I4, I4, I4, I4
30 WRITE(2,193)
173 FORMAT("FIN=1, ST. SUIVANTE SEPARÉE=2, A CHERCHER =3")
READ(1,*) KC
GO TO(54,4,540), KC
540 CALL ANDER(I1, I2, I3, I4, I5, I6, I7, I8)
IF(I8-1) 541, 30, 33
541 IPP=IFIX(G+H*FLOAT(I4))
IF(IPP) 4, 4, 540
54 IF(IOUT-4) 150, 543
543 WRITE(IOUT, 32)
GO TO 23
END
ENDs
```

NOM: **FILER**

FICHE TECHNIQUE DU PROGRAMME

TYPE	CHAINE			PERIPH.	IN	5, 7		
BUT	<i>visualiser les mesures faites à l'aide d'un AAudena monté en "Profilet".</i>			OUT	P			
				FORMAT	IN	B		
Observations				OUT				
Opérations	<i>utiliser les données issues de DCOD E</i>			ETIQ.	IN	MAA		
DATE DE MISE AU POINT	1976			OUT				
LANGAGE UTILISE	FORTRAN			PROGRAMMEUR				
SSP EMPLOYES				MORLIERE				
Starting ADDRESS	2			<i>Modifié 02.77 pour utilisation dans Base Majestique - lecture de toutes les données et mise en mémoire - CHUCKLA -</i>				
Nb Source	1	Nb Objet	...				Nb Absolu	✓
Divers								
Clés employées								

FICHE EXPLOITATION

VOIR NOTE TECHNIQUE

```
PROGRAM FILER
DIMENSION MASTR(14),IPARA(5,200)
10 IPLOT(1,MN,MA)=IFIX(9999.*FLOAT(1-MN)/FLOAT(MA-MN))
10 WRITE(2,300)
300 FORMAT("TRACE PROFIL"/"IN, STATION?")
   READ(1,*)IN,IS
   5 READ(IN,200)MASTR
200 FORMAT(I4,I3,3I2,2I4,I1,A1,I5,I1,A1,2I4,I2)
   F(MASTR(2)-IS)1,2,1
   1 IF(IN-7)10,4,4
   4 CALL PTAPE(7,1,0)
   GOTO 5
   2 WRITE(2,200)MASTR
   DO 701 N=1,5
   DO 701 J=1,200
701 IPARA(N,J)=0
   J=3
702 J=J+1
   READ(IN)(IPARA(N,J),N=1,5)
   IF(IPARA(2,J))13,13,700
13 WRITE(2,302)
302 FORMAT("X MIN,MAX")
   READ(1,*)MNI,MA1
   3 J=0
   WRITE(2,301)
301 FORMAT("T=2 S=3 PROJ.AXE=4 VECT=5 ELLIPSE=6")

   READ(1,*)IS
   IEL =0
   IF(IS-6)75,78,10
78 IEL=1
   IS=5
75 N3=0
   WRITE(2,303)
303 FORMAT("Y MIN,MAX")
   READ(1,*)MN2,MA2
   IF(IS-4)401,404,404
401 WRITE(2,305)
   GOTO 403
404 WRITE(2,305)
305 FORMAT("DIR AXE/N")
   READ(1,*)IALFA
   IF(IS-5)401,405,405
405 WRITE(2,306)
306 FORMAT("PLTL")
   IX=7000
   IY=7000
   WRITE(2,100)IX,IY
100 FORMAT(I4,I5,1H+)
   ALFA=FLOAT(IALFA)*3.1416/180.
   IU=IFIX(40.*SIN(ALFA))
   IV=IFIX(40.*COS(ALFA))
   IX=IX+IPL0T(IU,0,380)
   IY=IY+IPL0T(IV,0,250)
   WRITE(2,101)IX,IY
101 FORMAT(I4,I5)
```

```
400 J=J+1
      IF(IPARA(2,J))7,7,54
54  IF(IS-4)38,39,39
      38 IY=IPL0T(IPARA(1S,J),MN2,MA2)
43  IY=IPL0T(IPARA(1,J),MNI,MA1)
      IF(N3)55,55,56
55  WRITE(2,103)IX,IY
      GOTO 400
56  WRITE(2,131)IX,IY
402  CALL CLE(1)
      IF(I)16,16,7
      16 N3=1
          GOTO 430
      7  WRITE(2,337)
337  FORMAT("PLTT")
      IF(IN-7)3,21,21
21  CALL RTAPE(7,-1,0)
      GOTO 3
39  U=FLOAT(IPARA(4,J))
      V=FLOAT(IPARA(5,J))
      R=V/U
      X=U*U+V*V
      D=93.-180.*ATAN(R)/3.1416
      IF(U)71,72,72
71  D=D+180.
72  Y=SQRT(X)
      IF(IS-5)86,85,85
36  D=(FLOAT(IALFA)-D)*3.1416/180.
      IY=IFIX(X*COS(D)+.5)
      IY=IPL0T(IY,MN2,MA2)
      GOTO 43
35  D=D*3.1416/180.+ALFA
      IPARA(4,J)=IFIX(X*SIN(D)+.5)
      IPARA(5,J)=IFIX(X*COS(D)+.5)
      IF(IEL)77,77,79
79  IPARA(1,J)=MA1/2
77  IX=IPL0T(IPARA(1,J),MNI,MA1)
      IY=IPL0T(3,MN2,MA2)
      WRITE(2,100)IX,IY
      IX=IX+IPL0T(IPARA(4,J),0,300)
      IY=IY+IPL0T(IPARA(5,J),0,250)
      GOTO 56
      END
      ENDS
-----
```

NOM: MOTRA

FICHE TECHNIQUE DU PROGRAMME

TYPE	CHAINE				PERIPH.	IN	5,7	
BUT	tracé en fonction du temps Z, T, S, vecteur courant, la projection du vecteur courant sur un axe quelconque, l'ellipse de courant et l'hodographe intégral -				OUT	P		
					FORMAT	IN	B	
Observations					OUT			
Opérations	utiliser les données issues de SCODA, SCOB				ETIQ.	IN	MAA	
DATE DE MISE AU POINT	1976				OUT			
LANGAGE UTILISE	FORTRAN				PROGRAMMEUR			
SSP EMPLOYES					MORLIERE			
Starting ADDRESS	2							
Nb Source	/	Nb Objet	/	Nb Absolu				/
Divers								
Clés employées								

FICHE

EXPLOITATION

VOIR NOTE TECHNIQUE

```
0001      PROGRAM MOTRA
0002      DIMENSION MASTR(14),IPARA(5),XM(5)
0003      IPLOT(1,MN,MA)=IFIX(9999.*FLOAT(1-MN)/FLOAT(MA-MN))
0004      10 WRITE(2,300)
0005      300 FORMAT("TRACE AA"/"IN,STATION?")
0006      READ(1,*)IN,IS
0007      5 READ(IN,200)MASTR
0008      200 FORMAT(I4,I3,3I2,2I4,1I,1A,1I5,1I,1A,2I4,I2)
0009      IF(MASTR(2)-IS)1,2,1
0010      1 IF(IN-7)10,4,4
0011      4 CALL PTAPE(7,1,0)
0012      GOTO 5
0013      2 WRITE(2,200)MASTR
0014      WRITE(2,320)
0015      320 FORMAT("NB VAL A MOY")
0016      READ(1,*)NVM
0017      N=IFIX(FLOAT(MASTR(14))/60.*FLOAT(NVM))
0018      WRITE(2,301)N
0019      ICY=0
0020      IEL=0
0021      301 FORMAT("T BASE="I6// "P=1 T=2 S=3 PROJ.AXE=5 VECT=6 PROG=7
0022      1 ELLIPSE=8")
0023      READ(1,*)IS
0024      IF(IS-8)75,78,78
0025      78 IEL=1
0026      IS=6
0027      75 N=1
0028      XM(1)=0.
0029      WRITE(2,303)
0030      303 FORMAT("Y MIN,MAX")
0031      READ(1,*)MN2,MA2
0032      IF(IS-7)36,37,37
0033      37 WRITE(2,306)
0034      306 FORMAT("X MIN,MAX")
0035      READ(1,*)MN1,MA1
0036      WRITE(2,304)
0037      304 FORMAT("PLTL")
0038      IX=0
0039      IY=IPLOT(0,MN2,MA2)
0040      WRITE(2,100)IX,IY
0041      100 FORMAT(I4,I5,1H')
0042      16 N=2
0043      GOTO 30
0044      36 WRITE(2,302)
0045      302 FORMAT("NB CYCLES MAX")
0046      READ(1,*)MA1
0047      MN1=1
0048      IF(IS-5)29,40,40
0049      40 WRITE(2,310)
0050      310 FORMAT("DIR AXE/N")
0051      READ(1,*)IALFA
0052      29 WRITE(2,304)
0053      IF(IS-6)30,88,88
0054      88 IX=7000
0055      IY=7000
0056      WRITE(2,100)IX,IY
0057      ALFA=FLOAT(IALFA)*3.1416/180.
0058      IU=IFIX(40.*SIN(ALFA))
0059      IV=IFIX(40.*COS(ALFA))
```



```
0060      IX=IX+IPL0T(IU,0,380)
0061      IY=IY+IPL0T(IU,0,250)
0062      WRITE(2,101)IX,IY
0063  30 DO 69 I=1,5
0064  69 XM(I)=0.
0065      N9=0
0066      DO 51 I=1,NVM
0067      READ(IN)(IPARA(N),N=1,5)
0068      IF(IPARA(2))55,55,53
0069  55 IF(I-1)7,7,57
0070  57 N9=1
0071      NVM=I-1
0072      GOTO 58
0073  53 DO 51 K=1,5
0074  51 XM(K)=XM(K)+FLOAT(IPARA(K))
0075  58 DO 52 K=1,5
0076  52 IPARA(K)=IFIX(XM(K)/FLOAT(NVM)+.5)
0077      ICY=ICY+1
0078      IF(IS-5)38,39,42
0079  39 U=FLOAT(IPARA(4))
0080      V=FLOAT(IPARA(5))
0081      IF(U)251,252,251
0082  252 X=ABS(V)
0083      IF(V)253,254,254
0084  253 D=180.
0085      GO TO 255
0086  254 D=0.
0087      GO TO 255
0088  251 R=V/U
0089      X=U*U+V*V
0090      D=90.-180.*ATAN(R)/3.1416
0091      IF(U)71,72,72
0092  71 D=D+180.
0093  72 X=SQRT(X)
0094  255 IF(IS-6)86,85,85
0095  85 D=D*3.1416/180.+ALFA
0096      IPARA(4)=IFIX(X*SIN(D)+.5)
0097      IPARA(5)=IFIX(X*COS(D)+.5)
0098      GOTO 44
0099  86 D=(FLOAT(ALFA)-D)*3.1416/180.
0100      IY=IFIX(X*COS(D)+.5)
0101      IY=IPL0T(IY,MN2,MA2)
0102      GOTO 43
0103  38 IY=IPL0T(IPARA(IS),MN2,MA2)
0104  43 IX=IPL0T(ICY,MN1,MA1)
0105  46 IF(N-1)91,91,92
0106  91 WRITE(2,100)IX,IY
0107      GOTO 47
0108  92 WRITE(2,101)IX,IY
0109  101 FORMAT(I4,I5)
0110  47 IF(N9)54,54,7
0111  54 CALL CLE(I)
0112      IF(I)16,16,7
0113  42 IF(IS-6)39,39,45
0114  44 IF(IEL)77,77,79
0115  79 ICY=MA1/2
0116  77 IX=IPL0T(ICY,MN1,MA1)
0117      IY=IPL0T(0,MN2,MA2)
0118      CALL CLE(I)
0119      IF(I)45,98,98
```

```
0120      98 WRITE(2,100)IX,IY
0121      45 IX=IX+IPL0T(IPARA(4),0,380)
0122          IY=IY+IPL0T(IPARA(5),0,250)
0123          GOTO 92
0124          7 WRITE(2,305)
0125      305 FORMAT("PLTT")
0126          IF(IN-7)10,21,21
0127          21 CALL PTAPE(7,-1,0)
0128          IF(I)10,2,2
0129          GOTO 2
0130          END
0131          ENDS
```

**END-OF-TAPE

*

/E

*END

FICHE TECHNIQUE DU PROGRAMME

NOM: **VECOR**

TYPE					CHAINE			PERIPH.	IN	5					
BUT	Permet, d'obtenir des données en format type "AAndraa" à partir des données Ekman.						PERIPH.	OUT	4						
							FORMAT	IN	A						
Observations								FORMAT	OUT	B					
							Opérations							ETIQ.	IN
DATE DE MISE AU POINT						PROGRAMMEUR								OUT	M _{RA}
						Jun 77						CHUCHLA			
LANGAGE UTILISE						FORTRAN									
SSP EMPLOYES						CLE									
Starting ADDRESS						2									
Nb Source		1	Nb Objet		1	Nb Absolu		1							
Divers															
Clés employées						clef 15 pour une fin de ruban									

FICHE EXPLOITATION

VECCR: IN, i CUT?
 5, 4
 7506 15 7 75 2
 7506 15 7 75 3
 7506 16 7 75 4
 7506 16 7 75 5
 7506 16 7 75 6
 7506 16 7 75 7
 STCP

```
0001      PROGRAM VECOF
0002      DIMENSION IC(5,100),MET(16)
0003      1 FORMAT('VECOF:IN,ICUT?')
0004      WRITE(2,1)
0005      READ(1,*)IN,ICUT
0006      CALL LEADR(4,10)
0007      4 READ(IN,100)(MET(L),L=1,16),NF
0008      IF(MET(6))2,4,2
0009      15 FORMAT(i4,3i5,i5)
0010      2 WRITE(2,15)MET(5),(MET(L),L=1,3),MET(6)
0011      MET(13)=0
0012      MET(14)=0
0013      WRITE(ICUT,5)MET(5),MET(6),(MET(L),L=1,3),MET(4),MET(7),
0014      1(MET(L),L=8,14)
0015      DC 10 I=1,5
0016      DC 10 J=1,100
0017      10 IC(I,J)=0
0018      J1=1
0019      101 J2=J1+4
0020      READ(5,3)((IC(I,J),I=1,4),J=J1,J2)
0021      IF(IC(2,J2))110,105,110
0022      110 J1=J2+1
0023      GO TO 101
0024      105 J2=J2-1
0025      DC 50 J=1,J2
0026      IC(5,J)=IC(4,J)
0027      IC(4,J)=IC(3,J)
0028      50 IC(3,J)=0
0029      DC 60 J=1,J2
0030      DIR=FLCAT(IC(4,J))
0031      VIT=FLCAT(IC(5,J))
0032      DI=((90.-DIR)*3.1416/180.)
0033      IC(4,J)=IFIX(VIT*CCS(DI))
0034      IC(5,J)=IFIX(VIT*SIN(DI))
0035      60 CONTINUE
0036      DC 30 J=1,J2
0037      30 WRITE(ICUT)(IC(I,J),I=1,5)
0038      100 FORMAT(3i3,i5,i6,i4,i5,i1,A2,i6,i1,A2,3i4,2i3,i5,i1)
0039      5 FORMAT(i4,i3,3i2,2i4,i1,A1,i5,i1,A1,2i4)
0040      3 FORMAT(5(3i4,i3))
0041      CALL CLE(K)
0042      IF(K)28,4,4
0043      28 WRITE(ICUT,9)
0044      9 FORMAT(70(1H9))
0045      END
0046      ENDS
```

**END- CF- TAPE

*

FICHE TECHNIQUE DU PROGRAMME

NOM: **CORAN**

TYPE	CHAINE			PERIPH.	IN	5		
BUT	Edition du type "EXIT" pour un profil de courant.			OUT	2			
				FORMAT	IN	B		
Observations				OUT				
Opérations				ETIQ.	IN	MAA		
				OUT				
DATE DE MISE AU POINT	1976			PROGRAMMEUR				
LANGAGE UTILISE	FORTRAN			CITEAU				
SSP EMPLOYES	CLE							
Starting ADDRESS	2							
Nb Source	1	Nb Objet	1				Nb Absolu	1
Divers								
Clés employées								

FICHE EXPLOITATION

- 1 Rappel : du Ruban perforé issu du décodeur AANDERAA, on passe à un ruban "archive" par l'intermédiaire du programme DECODP
- 2 On fait un listing de ce ruban "archive" par l'intermédiaire de CORAN. Les immersions voulues peuvent être entrées soit à la teletype (4) soit une fois pour toutes sur un ruban perforé que l'on place dans le lecteur optique (5).
- L'édition se fait alors en chaîne pour toute une croisière.

```
PROGRAM CORAN
  DIMENSION ID(5),MASTR(14),JD(5,5)
  3 FORMAT("CORAN"/"IMMERSIONS 1 OU 5?")
  6 FORMAT(I4,I3,3I2,2I4,I1,A1,I5,I1,A1,2I4)
  320 FORMAT("ENTREE =5"/"STATION ?")
  102 FORMAT(2/"CROISIERE*"I5,26X"DATE   : "3I3/"STATION   : "I5,
  126X"HEURE   : "I8/"POSITION : "F5.2,IH',I1,1XA2,3XF6.2,IH'I1,
  21XA2/"COURANTOMETRE:"I6,21X"PENDULE:"I6"SEC")
  271 FORMAT(I4,F6.2,6X,1X2I4,2I5)
  272 FORMAT(I4,2F6.2,1X2I4,2I5,F6.2,F6.1,F7.3)
  101 FORMAT(7("*",9X))
  WRITE(2,3)
  READ(1,*)IN
  I=0
  303 I=I+1
  READ(IN,*)ID(I)
  IF(ID(I)-9999)303,301
  301 WRITE(2,327)
  IN=5
  READ(1,*)IS
  306 CALL CLE(KC)
  IF(KC)600,5,301
  5 READ(IN,6)MASTR
  IF(MASTR(2)-IS)5,2,5
  2 AT=FLOAT(MASTR(7))/100.
  ON=FLOAT(MASTR(10))/100.
  WRITE(2,101)
  WRITE(2,102)MASTR(1),(MASTR(I),I=3,5),MASTR(2),MASTR(5),
  IAT,MASTR(8),MASTR(9),ON,(MASTR(I),I=11,14)
  WRITE(2,103)
  103 FORMAT(//"PROF.TEMP.  SAL  DIR VIT  U  V  S1  S2  S3
  IDST  H.DYN"/)
  N=0
  D=0.
  DI=0.
  P1=0.
  N7=1
  J=1
  READ(IN)(JD(I,J),I=1,5)
  IF(JD(1,1))20,20,22
  22 DO 23 I=1,5
  23 JD(I,2)=JD(I,1)
  JD(1,1)=0
  J=2
  20 J=J+1
  READ(IN)(JD(I,J),I=1,5)
  IF(JD(2,J))24,24,25
  25 IF(J-50)20,24,24
  24 J=1
```

```
235 IF(N-JD(1,J))231,231,231
237 AN=FLOAT(N-JD(1,J-1))/FLOAT(JD(1,J)-JD(1,J-1))
T=.71*(FLOAT(JD(2,J-1))+AN*FLOAT(JD(2,J)-JD(2,J-1)))
S=.71*(FLOAT(JD(3,J-1))+AN*FLOAT(JD(3,J)-JD(3,J-1)))
U=FLOAT(JD(4,J-1))+AN*FLOAT(JD(4,J)-JD(4,J-1))
V=FLOAT(JD(5,J-1))+AN*FLOAT(JD(5,J)-JD(5,J-1))
P=FLOAT(N)
GOTO 250
241 P=FLOAT(N)
T=.71*FLOAT(JD(2,J))
S=.71*FLOAT(JD(3,J))
U=FLOAT(JD(4,J))
V=FLOAT(JD(5,J))
GOTO 250
242 J=J+1
IF(J-50)32,32,31
32 IF(JD(2,J))500,500,205
31 DO 7 I=1,5
7 JD(I,1)=JD(I,J-1)
J=1
GOTO 23
250 CALL SIGAL(P,T,S,SIBM,DALFA)
D=D+(DI+DALFA)*(P-P1)*1.E+20/2.
P1=P
DI=DALFA
IF(N-ID(N7))300,261,300
261 A=.001*SIBM
DST=(.02736-A/(1.+A))*1.E+5
IVIT=IFIX(SQRT(U*U+V*V))
IF(IFIX(U))500,701,500
701 IF(V)703,705
703 IDIR=130
GOTO 530
705 IDIR=7
GOTO 530
500 DI=ATAN(V/U)*57.295827
IF(U)510,520
520 IDIR=20-IFIX(DI)
GOTO 530
510 IDIR=270-IFIX(DI)
530 IU=IFIX(U)
IV=IFIX(V)
IF(S)631,631,632
631 WRITE(2,271)N,T,IDIR,IVIT,IU,IV
GOTO 200
632 WRITE(2,272)N,T,S,IDIR,IVIT,IU,IV,SIBM,DST,D
200 N7=N3+1
IF(ID(N7)-9999)300,500
300 N=N+1
GOTO 205
500 WRITE(2,501)
501 FORMAT(///)
WRITE(2,101)
GOTO 306
600 STOP
END
ENDS
```


FICHE TECHNIQUE DU PROGRAMME

APPROXIMATION
 NOM: POLYNOMIALE
 FOURIER + LAGRANGE

TYPE	CALCUL divers	CHAINE		PERIPH.	IN	I
BUT	Approximation ds courbes par des polynomes de LAGRANGE et des Series deFOURIER (2Programmes)				OUT	2
				FORMAT	IN	A
Observations					OUT	L
Opérations				ETIQ.	IN	' -
					OUT	-
DATE DE MISE AU POINT				PROGRAMMEUR		
LANGAGE UTILISE	BASIC			MORLIERE		
SSP EMPLOYES						
Starting ADDRESS						
Nb Source	Nb Objet		Nb Absolu			
Divers						
Clés employées						

FICHE EXPLOITATION

PTA

READY
PLIST

READY
LIST

```

1  DIM A(12), B(12), X(12), Y(12), C(12)
2  PRINT
3  PRINT
4  PRINT "APPROXIMATION PAR SERIE DE FOURIER"
5  PRINT
10  MAT B=ZER
15  MAT A=ZER
20  READ N, P
30  READ T0, T
40  MAT READ X(N)
50  MAT READ Y(N)
60  READ M
70  MAT READ C(M)
80  FOR I=1 TO N
82  LET X(I)=2*3.1416/T*(X(I)-T0)
85  LET B(I)=B(I)+Y(I)/N
90  NEXT I
95  FOR K=2 TO P+1
100  FOR I=1 TO N
102  LET B(K)=B(K)+Y(I)*COS((K-1)*X(I))*2/N
103  LET A(K)=A(K)+Y(I)*SIN((K-1)*X(I))*2/N
105  NEXT I
110  NEXT K
115  PRINT
116  PRINT "COEFFICIENTS"
117  PRINT "  B  ", "  A"
120  FOR I=1 TO P+1
125  PRINT B(I), A(I)
130  NEXT I
131  PRINT
132  PRINT "VALEURS CALCULEES"
133  PRINT "  X", "  Y"
135  FOR K=1 TO M
136  LET W=C(K)
137  LET C(K)=2*3.1416/T*(C(K)-T0)
140  LET R=B(1)
145  FOR I=2 TO P+1
150  LET R=R+B(I)*COS((I-1)*C(K))+A(I)*SIN((I-1)*C(K))
155  NEXT I
160  PRINT W, R
170  NEXT K
175  PRINT
500  DATA 12, 3
501  DATA 0, 12
502  DATA 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11
503  DATA 0, 1, 4, 5, 4, 3, 3.5, 3.5, 3, 1, .5, .2
504  DATA 5
505  DATA 2, 3, 6, 8, 8.5
800  END

```

READY

READY
PLIST

READY
LIST

```
5 PRINT
6 PRINT
7 PRINT
10 PRINT "APPROXIMATION POLYNOMIALE - MOYENNES CARREES"
20 DIM A(10), B(10), C(10), D(10)
21 DIM M(10,10), I(10,10)
30 MAT M=ZER
35 MAT I=ZER
40 MAT D=ZER
50 READ N, P
60 MAT READ A(N)
70 MAT READ B(N)
80 PRINT
90 PRINT "PT. MESURE"
100 PRINT
110 FOR I=1 TO N
112 PRINT A(I), B(I)
114 NEXT I
120 LET P1=P+1
130 FOR I=1 TO P1
135 FOR J=1 TO P1
140 FOR K=1 TO N
150 LET L=2*P-(J-1)-(I-1)
160 LET M(I,J)=M(I,J)+A(K)*L
170 NEXT K
180 NEXT J
190 NEXT I
200 FOR I=1 TO P1
210 FOR K=1 TO N
220 LET D(I)=D(I)+B(K)*A(K)*(P-(I-1))
230 NEXT K
240 NEXT I
241 PRINT
242 PRINT
243 PRINT "SYSTEME LINEAIRE"
244 PRINT
245 FOR I=1 TO P1
246 FOR J=1 TO P1
247 PRINT M(I,J),
248 NEXT J
249 PRINT D(I)
250 NEXT I
251 GOSUB 400
252 LET N=A0
260 MAT C=I*D
270 LET E1=0
280 FOR K=1 TO N
290 LET S=0
300 FOR J=1 TO P1
310 LET S=S+C(J)*A(K)*(P-J+1)
320 NEXT J
330 LET F1=E1+(B(K)-S)*2
340 NEXT K
350 LET F1=F1/N
360 PRINT
370 PRINT "COEFF. POLY DEGRE"; P
380 PRINT
390 FOR I=1 TO P1
391 PRINT " DEGRE "; P+1-I; C(I)
392 NEXT I
400 PRINT
410 PRINT "EGART QUADRATIQUE MOYEN ="; F1
```

```
421 PRINT "GRAPH"
422 PRINT "X0,X1,Y0,Y1,PAS"
423 INPUT X0,X1,Y0,Y1,I6
424 PRINT "COEFF. MULTIPLICATIVE LFS X";
425 INPUT C9
426 PRINT "PLTL"
427 LET X=X0
430 IF X >= X1 THEN 460
432 LET A1=)))*(X-X0)/(X1-X0)
434 LET A2=0
436 FOR I=1 TO P1
437 IF P1-I=0 THEN 440
438 LET A2=A2+C[I]*(X*C9)^(P1-I)
439 GOTO 441
440 LET A2=A2+C[I]
441 NEXT I
442 LET A2=)))*(A2-Y0)/(Y1-Y0)
444 PRINT INT(A1),INT(A2)
448 LET X=X+I6
450 GOTO 430
460 PRINT "PLTL"
470 STOP
493 LET A)=N
499 LET N=P1
500 FOR I=1 TO N
501 LET M[I,I]=M[I,I]-1
502 LET I[I,I]=1
503 NEXT I
510 FOR M=1 TO N
511 FOR I=M+1 TO N
512 LET X[I]=M[I,M]/M[M,M]
513 LET E[I]=M[M,I]
514 NEXT I
520 FOR I=1 TO N
521 LET S=0
522 LET S1=0
523 FOR K=M TO N
524 LET S=S+I[I,K]*M[K,M]
525 LET S1=S1+M[M,K]*I[K,I]
526 NEXT K
530 LET U[I]=S
531 LET V[I]=S1
535 NEXT I
540 LET S=0
541 FOR J=M TO N
542 LET S=S+V[J]*M[J,M]
543 NEXT J
550 LET L=S+M[M,M]
560 FOR I=1 TO N
565 FOR J=1 TO N
566 IF L=0 THEN 571
570 LET I[I,J]=I[I,J]-U[I]*V[J]/L
571 NEXT J
572 NEXT I
580 IF M=N THEN 600
582 FOR I=M+1 TO N
583 FOR J=M+1 TO N
585 LET M[I,J]=M[I,J]-X[I]*E[J]
590 NEXT J
595 NEXT I
596 NEXT M
600 RETURN
699 DATA 5,2
700 DATA .2,.7,1.1,1.3,1.5
702 DATA 14.35,15.35,16.62,17.7,18.5
999 END
```

FICHE TECHNIQUE DU PROGRAMME

NOM: curfit
CURFIT CURFIT2

TYPE	CALCUL divers	CHAINE		PERIPH.	IN	II
BUT	Determination des courbes passant par des points de mesures - Methode des moindres carrees				OUT	2
Observations				FORMAT	IN	A
Opérations					OUT	1
DATE DE MISE AU POINT				ETIQ.	IN	
LANGAGE UTILISE	BASIC				OUT	
SSP EMPLOYES				PROGRAMMEUR MORLIERE		
Starting ADDRESS						
Nb Source	Nb Objet		Nb Absolu			
Divers						
Clés employées						

FICHE EXPLOITATION

SCP

READY
PTA

READY

LIST

```

3 DIM X(14),Y(14),U(14),V(14),A(6),B(6),S(6),R(6)
4 MAT F=CON
5 READ N
6 PRINT
7 FOR I=1 TO N
8 READ X(I),Y(I)
9 NEXT I
10 PRINT
11 PRINT
12 PRINT " ", "APPROXIMATION COURBE PAR MOINDRES CARRÉS"
13 PRINT
14 PRINT "TYPE DE COURBE", " INDEX DE", " A", " B"
15 PRINT " ", "DETERMINATION"
16 PRINT
17 FOR I=1 TO 6
18 MAT S=ZER
19 GOSUB 120
20 IF (I-5)*(I-6)=0 THEN 35
21 IF (I-2)*(I-3)=0 THEN 23
22 FOR J=1 TO N
23 LET V(J)=Y(J)
24 GOSUB 98
25 NEXT J
26 IF I=1 THEN 45
27 GOTO 56
28 FOR J=1 TO N
29 IF Y(J) <= 0 THEN 42
30 LET V(J)=LOG(Y(J))
31 GOSUB 98
32 NEXT J
33 IF I=3 THEN 50
34 GOTO 45
35 FOR J=1 TO N
36 IF Y(J)=0 THEN 42
37 LET V(J)=1/Y(J)
38 GOSUB 98
39 NEXT J
40 IF I=6 THEN 56
41 GOTO 45
42 PRINT "MAUVAISE APPROXIMATION"
43 LET F(I)=0
44 GOTO 63
45 FOR J=1 TO N
46 LET U(J)=X(J)
47 GOSUB 101
48 NEXT J
49 GOTO 61
50 FOR J=1 TO N
51 IF X(J) <= 0 THEN 42
52 LET U(J)=LOG(U(J))
53 GOSUB 101
54 NEXT J
55 GOTO 61
56 FOR J=1 TO N
57 IF X(J)=0 THEN 42
58 LET U(J)=1/X(J)

```

```
59 GOSUB 101
60 NEXT J
61 GOSUB 161
62 PRINT C(I),A(I),B(I)
63 NEXT I
64 GOSUB 105
65 PRINT
66 PRINT
67 PRINT
68 PRINT "TYPE DE COURBE A PRECISE (1 A 6, 0=FI)";
69 INPUT I
70 IF I=0 THEN 136
71 LET K=I
72 IF F(I)=1 THEN 76
73 GOSUB 120
74 PRINT "APPROXIMATION IMPOSSIBLE"
75 GOTO 65
76 GOSUB 138
77 IF (I-1)*(I-5)*(I-6)≠0 THEN 37
78 FOR J=1 TO N
79 LET Y=A(I)+B(I)*X(J)
80 IF I=1 THEN 84
81 LET Y=1/Y
82 IF I=5 THEN 84
83 LET Y=X(J)*Y
84 GOSUB 176
85 NEXT J
86 GOTO 65
87 FOR J=1 TO N
88 IF I=2 THEN 94
89 IF I=3 THEN 92
90 LET Y=A(4)+B(4)/X(J)
91 GOTO 95
92 LET Y=A(3)*(X(J)+B(3))
93 GOTO 95
94 LET Y=A(2)*EXP(B(2)*X(J))
95 GOSUB 176
96 NEXT J
97 GOTO 65
98 LET S(5)=S(5)+V(J)+2
99 LET S(3)=S(3)+V(J)
100 RETURN
101 LET S(1)=S(1)+U(J)
102 LET S(2)=S(2)+U(J)+2
103 LET S(4)=S(4)+U(J)*V(J)
104 RETURN
105 FOR I=1 TO N-1
106 LET M=I
107 FOR J=I+1 TO N
108 IF X(M) <= X(J) THEN 110
109 LET M=J
110 NEXT J
111 IF M=I THEN 118
112 LET P=X(M)
113 LET Q=Y(M)
114 LET X(M)=X(I)
115 LET Y(M)=Y(I)
116 LET X(I)=P
117 LET Y(I)=Q
118 NEXT I
119 RETURN
120 LET K=I
121 IF K=1 THEN 136
122 IF I=2 THEN 134
123 IF K=3 THEN 132
```

```
124 IF K=4 THEN 130
125 IF K=5 THEN 123
126 PRINT "6. Y=X/(A+B*X) ";
127 RETURN
128 PRINT "5. Y=1/(A+B*X) ";
129 RETURN
130 PRINT "4. Y=A+B/X ";
131 RETURN
132 PRINT "3. Y=A*(X+B)";
133 RETURN
134 PRINT "2. Y=A*EXP(B*X)";
135 RETURN
136 PRINT "1. Y=A+(B*X)";
137 RETURN
138 PRINT " ";
139 GOSUB 121
140 PRINT "C'EST UNE FONCTION ";
141 IF K=1 THEN 146
142 IF K=2 THEN 148
143 IF K=3 THEN 150
144 PRINT "HYPERBOLIQUE";
145 GOTO 151
146 PRINT "LINEAIRE";
147 GOTO 151
148 PRINT "EXPONENTIELLE";
149 GOTO 151
150 PRINT "POISSANCE";
151 PRINT " LES RESULTATS "
152 IF K=1 THEN 154
153 PRINT "DE L'APPROXIMATION MOINDRES CARRÉS DE LA TROUSSE DE 4 EN LIG. 1."
154 PRINT " (PAR X CROISSANTS)"
155 PRINT " SONT LES SUIVANTS : "
156 PRINT
157 PRINT "X-ACTUEL", "Y-ACTUEL", " Y-CALC", " PCT DIFFER"
158 PRINT
159 RETURN
160 PRINT
161 LET B=(N*SC(4)-SC(11)*SC(3))/(N*SC(2)-(SC(11)+2))
162 LET A=(SC(3)-B*SC(1))/N
163 LET S1=SC(5)-(SC(3)+2)/N
164 LET S2=(B+2)*(SC(2)-(SC(11)+2)/N)
165 LET C(I)=S2/S1
166 IF (I-1)*(I-4)*(I-5)=0 THEN 173
167 IF (I-2)*(I-3)=0 THEN 171
168 LET A(I)=B
169 LET B(I)=A
170 RETURN
171 LET A(I)=EXP(A)
172 GOTO 174
173 LET A(I)=A
174 LET B(I)=B
175 RETURN
176 PRINT X(J),Y(J),Y,
177 LET D=Y(J)-Y
178 LET D=.1*SGN(D)*INT(1000*ABS(D/Y))
179 IF D<0 THEN 184
180 IF D>0 THEN 183
181 PRINT " 0"
182 RETURN
183 PRINT " ";
184 PRINT D
185 RETURN
186 STOP
900 DATA 6
901 DATA 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12
999 END
```


PTA

BFADY
PLIST
A

CURFIT 2

LIST

```
2 PRINT
3 DIM X(14),Y(14),U(14),V(14),AC(6),BC(6),SC(6),EC(6)
4 MAT F=CON
5 BFAD N
6 PRINT
7 FOR I=1 TO N
8 BFAD X(I),Y(I)
9 NEXT I
10 PRINT
11 PRINT
12 PRINT " ", "APPROXIMATION COURBE PAR MOINDRES CARRÉS"
13 PRINT
14 PRINT "TYPE DE COURBE", " INDEX " "DE", " " "A", " " "J"
15 PRINT " ", "DETERMINATION"
16 PRINT
17 FOR I=1 TO 6
18 MAT S=ZER
19 GOSUB 120
20 IF (I-5)*(I-6)=0 THEN 35
21 IF (I-2)*(I-3)=0 THEN 28
22 FOR J=1 TO N
23 LET V(J)=Y(J)
24 GOSUB 98
25 NEXT J
26 IF I=1 THEN 45
27 GOTO 56
28 FOR J=1 TO N
29 IF Y(J) <= 0 THEN 42
30 LET V(J)=LOG(Y(J))
31 GOSUB 98
32 NEXT J
33 IF I=3 THEN 50
34 GOTO 45
35 FOR J=1 TO N
36 IF Y(J)=0 THEN 42
37 LET V(J)=1/Y(J)
38 GOSUB 98
39 NEXT J
40 IF I=6 THEN 56
41 GOTO 45
42 PRINT "MAUVAISE APPROXIMATION"
43 LET F(I)=0
44 GOTO 63
45 FOR J=1 TO N
46 GOSUB 900
47 GOSUB 101
48 NEXT J
49 GOTO 61
50 FOR J=1 TO N
51 IF X(J) <= 0 THEN 42
52 LET U(J)=LOG(U(J))
53 GOSUB 101
54 NEXT J
55 GOTO 61
56 FOR J=1 TO N
57 IF X(J)=0 THEN 42
58 LET U(J)=1/X(J)
59 GOSUB 101
```

```

      GOSUB 161
62  PRINT C(I),A(I),B(I)
63  NEXT I
64  GOSUB 105
65  PRINT
66  PRINT
67  PRINT
68  PRINT "TYPE DE COURBE A PRECISER (1 A 6, 0=FIN)";
69  INPUT I
70  IF I=0 THEN 136
71  LET K=I
72  IF F(I)=1 THEN 76
73  GOSUB 120
74  PRINT "APPROXIMATION IMPOSSIBLE"
75  GOTO 65
76  GOSUB 138
77  IF (I-1)*(I-5)*(I-6)#0 THEN 87
78  FOR J=1 TO N
79  GOSUB 905
80  IF I=1 THEN 84
81  LET Y=1/Y
82  IF I=5 THEN 84
83  LET Y=X(J)*Y
84  GOSUB 176
85  NEXT J
86  GOTO 65
87  FOR J=1 TO N
88  IF I=2 THEN 94
89  IF I=3 THEN 92
90  LET Y=A(4)+B(4)/X(J)
91  GOTO 95
92  LET Y=A(3)*(X(J)+B(3))
93  GOTO 95
94  LET Y=A(2)*EXP(B(2)*X(J))
95  GOSUB 176
96  NEXT J
97  GOTO 65
98  LET S(5)=S(5)+V(J)+2
99  LET S(3)=S(3)+V(J)
100  RETURN
101  LET S(1)=S(1)+U(J)
102  LET S(2)=S(2)+U(J)+2
103  LET S(4)=S(4)+U(J)*V(J)
104  RETURN
105  FOR I=1 TO N-1
106  LET M=I
107  FOR J=I+1 TO N
108  IF X(M) <= X(J) THEN 110
109  LET M=J
110  NEXT J
111  IF M=I THEN 113
112  LET P=X(M)
113  LET Q=Y(M)
114  LET X(M)=X(I)
115  LET Y(M)=Y(I)
116  LET X(I)=P
117  LET Y(I)=Q
118  NEXT I
119  RETURN
120  LET K=I
121  IF K=1 THEN 136
122  IF I=2 THEN 134
123  IF K=3 THEN 132
124  IF K=4 THEN 130
125  IF K=5 THEN 128
126  PRINT "6. Y=X/(A+B*X) ";
127  RETURN
```

```
RETURN
130 PRINT "4. Y=A+B/X ",
131 RETURN
132 PRINT "3. Y=A*(X+B)",
133 RETURN
134 PRINT "2. Y=A*EXP(B*X)";
135 RETURN
136 PRINT "1. Y=A+B*X+2",
137 RETURN
138 PRINT " ";
139 GOSUB 121
140 PRINT "C'EST UNE FONCTION ";
141 IF K=1 THEN 146
142 IF K=2 THEN 148
143 IF K=3 THEN 150
144 PRINT "HYPERBOLIQUE";
145 GOTO 151
146 PRINT "PARABOLIQUE";
147 GOTO 151
148 PRINT "EXPOONENTIELLE";
149 GOTO 151
150 PRINT "PUISSANCE";
151 PRINT " LES RESULTATS "
152 IF K=1 THEN 154
153 PRINT "DE L'APPROXIMATION MOINDRES CARRÉS DE LA TRANSFORMÉE LIN."
154 PRINT " (PAR X CROISSANTS)"
155 PRINT " SONT LES SUIVANTS : "
156 PRINT
157 PRINT "X-ACTUEL", "Y-ACTUEL", " Y-CALC", " PCT DIFFER"
158 PRINT
159 RETURN
160 PRINT
161 LET B=(N*SC(4)-SC(1)*SC(3))/(N*SC(2)-(SC(1)+2))
162 LET A=(SC(3)-B*SC(1))/N
163 LET S1=SC(5)-(SC(3)+2)/N
164 LET S2=(B+2)*(SC(2)-(SC(1)+2)/N)
165 LET C(I)=S2/S1
166 IF (I-1)*(I-4)*(I-5)=0 THEN 173
167 IF (I-2)*(I-3)=0 THEN 171
168 LET A(I)=B
169 LET B(I)=A
170 RETURN
171 LET A(I)=EXP(A)
172 GOTO 174
173 LET A(I)=A
174 LET B(I)=B
175 RETURN
176 PRINT X(J),Y(J),Y,
177 LET D=Y(J)-Y
178 LET D=.1*SGN(D)*INT(1000*ABS(D/Y))
179 IF D<0 THEN 184
180 IF D>0 THEN 183
181 PRINT " 0"
182 RETURN
183 PRINT " ";
184 PRINT D
185 RETURN
186 STOP
```

```
900  END
901  IF I=1 THEN 903
902  LET UCJ]=X[CJ]
903  RETURN
904  LET UCJ]=X[CJ]+2
905  RETURN
906  IF I=1 THEN 903
907  LET Y=AC[I]+E[CJ]+X[CJ]
908  RETURN
909  LET Y=AC[I]+E[CJ]*X[CJ]+2
910  RETURN
950  DATA 3
951  DATA 0, 14.42, 40, 14.75, 70, 15.4, 100, 16.42, 150, 18.45
952  DATA 20, 14.5, 120, 17.3, 50, 14.9
999  END
```

READY

FICHE TECHNIQUE DU PROGRAMME

NOM: ETOILE

TYPE		CHAINE		PERIPH.	IN	
BUT	<i>Permet le calcul du point étoile.</i>			FORMAT	OUT	
Observations	<i>Les caractéristiques des étoiles sont à changer tous les ans.</i>			ETIQ.	IN	
Opérations					OUT	
DATE DE MISE AU POINT				PROGRAMMEUR		
LANGAGE UTILISE	<i>BASIC</i>			<i>Matière</i>		
SSP EMPLOYES						
Starting ADDRESS						
Nb Source	Nb Objet		Nb Absolu			
Divers						
Clés employées						

FICHE EXPLOITATION

DATE
 ?12,7,,77
 POINT VERNAL NON PREVU POUR 1977
 METTRE EN 701 LE PT VERNAL POUR 1977

READY
 FIN

DATE
 ?12,10,74
 LATITUDE ESTIMEE (DEG, MIN)
 ?5,4
 LONGITUDE ESTIMEE (DEG, MIN)
 ?4,24
 ETOILE, HEURE, MINUTE, SECONDE, HO (DEG, MIN)
 ?12,18,33,43,34.2,35.4
 ETOILE, HEURE, MINUTE, SECONDE, HO (DEG, MIN)
 ?99,0,0,0,0,0

POINT ETOILE DU 12 10 1974

ESTIMEE LATITUDE 5 4 N
 LONGITUDE 4 24 N

F	H	M	S	HO	MI	SE
12	18	33	43	34.2	35.4	2443.31

```
1 DIM E(20,2),R(12,8),V(12)
9 MAT READ E
10 MAT READ V
28 PRINT
29 PRINT
30 PRINT "DATE"
40 INPUT J,M1,A
41 IF A=74 THEN 50
42 PRINT " POINT VERNAL NON PREVU POUR";1900+A
43 PRINT "METTRE EN 701 LE PT VERNAL POUR";1900+A
45 STOP
50 LET A1=V(M1)
51 PRINT "LATITUDE ESTIMEE (DEG,MIN)"
52 INPUT F1,F2
53 LET F3=F2+F1*60
54 PRINT "LONGITUDE ESTIMEE(DEG,MIN)"
55 INPUT G1,G2
56 LET G=G2+G1*60
57 LET I9=0
60 PRINT "ETOILE, HEURE, MINUTE, SECONDE, MO (DEG, MIN)"
62 LET I9=I9+1
65 INPUT N,H,M,S,O1,O2
67 IF N=99 THEN 450
71 LET R(I9,1)=N
80 LET R(I9,2)=H
90 LET R(I9,3)=M
91 LET R(I9,4)=S
160 LET X=A1+59.14*(J-1)+902.46*H+15.04*M+.25*S-G+E(N,1)
170 IF X<21600 THEN 200
190 LET X=X-21600
200 IF X <= 10800 THEN 230
210 LET P=21600-X
220 GOTO 240
230 LET P=X
240 LET D1=E(N,2)
241 LET F=F3*3.14159/10800
242 LET D1=D1*3.14159/10800
243 LET P=P*3.14159/10800
250 LET Y=SIN(F)*SIN(D1)+COS(F)*COS(D1)*COS(P)
260 LET H0=ATN(Y/SQR(1-Y*Y))
270 LET H0=H0*10800/3.14159
280 LET O=O1*60+O2
290 LET I=O-(4.71024+3490.6/O)-H0
320 LET R(I9,5)=O1
330 LET R(I9,6)=O2
340 LET R(I9,7)=I
430 GOTO 60
450 GOSUB 800
460 PRINT "POINT ETOILE DU ";J;M1;1900+A
461 PRINT
462 PRINT
465 PRINT "ESTIME LATITUDE ";
470 IF F1<0 THEN 490
472 IF F2<0 THEN 490
480 PRINT F1;F2;"N"
485 GOTO 500
490 PRINT -F1;-F2;"S"
```

```
500 PRINT "          LONGITUDE";
510 IF G1<0 THEN 530
512 IF G2<0 THEN 530
515 PRINT G1;G2;"V"
520 GOTO 540
530 PRINT -G1;-G2;"E"
540 PRINT
550 PRINT
555 PRINT " E      H      M      S          HO          1"
556 LET I=0
557 LET I=I+1
558 IF I=19 THEN 568
560 PRINT
564 PRINT R(1,1);R(1,2);R(1,3);
565 PRINT R(1,4);R(1,5);R(1,6);R(1,7)
567 GOTO 557
568 GOSUB 800
570 STOP
600 REM ACHERNAR 1
601 DATA 20148.5,-3442
602 REM ACUX 2
603 DATA 10422.1,-3777.7
604 REM ALDEBARAN 3
605 DATA 17482.7,287.6
606 REM ALPHERATZ 4
607 DATA 21493.7,1737
608 REM ALTAIR 5
609 DATA 3756.6,528.1
610 REM ANTARES 6
611 DATA 6782.1,-1582.6
612 REM ARCTURUS 7
613 DATA 8782.5,1158.7
614 REM BETELGEUSE 8
615 DATA 16292.7,444.2
616 REM CANOPUS 9
617 DATA 15849.1,-3150.9
618 REM CHEVRE 10
619 DATA 16877.5,2758.5
620 REM DENEb 11
621 DATA 2991.5,2711.4
622 REM LEPI 12
623 DATA 9542.1,-661.8
624 REM FOMALHAUT 13
625 DATA 956.1,-1785.4
626 REM POLLUX 14
627 DATA 14643.2,1585.3
628 REM PROCIOn 15
629 DATA 14730.1,317.4
630 REM REGULUS 16
631 DATA 12494.3,725.4
632 REM RIGEL 17
633 DATA 16900.1,-493.8
634 REM RIGIL KENT 18
635 DATA 8431.9,-3643.9
636 REM SIRIUS 19
637 DATA 15539.2,-1000.9
638 REM VEGA 20
639 DATA 4858.9,2325.5
700 REM POINT VERNAL 1974 PAR MOIS
701 DATA 6015.9,7849.2,9505.1,11338.4,13112.5,14945.8,16723,18553.3
702 DATA 20386.6,560.8,2394,4168.2
```

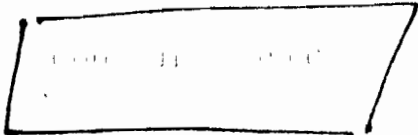
```
801 LET I=I+1
802 PRINT
803 IF I<10 THEN 801
804 RETURN
900 END
```


FICHE TECHNIQUE DU PROGRAMME

NOM: COURANT

TYPE		CHAINE		PERIPH.	IN			
BUT	Calcul de camautometrie				OUT			
Observations				FORMAT	IN			
Opérations					OUT			
DATE DE MISE AU POINT				ETIQ.	IN			
LANGAGE UTILISE	BASIC				OUT			
NSP EMPLOYES				PROGRAMMEUR				
Starting ADDRESS				MORC...				
Nb Source	1	Nb Objet	X				Nb Absolu	X
Divers								
Clés employées	X							

FICHE EXPLOITATION



```

5  PI 4 AC 2.5, 31
10  PRINT "STATION?"
20  INPUT S0
30  PRINT "DATE (JOUR, MOIS) ?"
40  INPUT J0
50  PRINT "DEFIURE * MIT, BUR ?"
51  INPUT M0, P1
52  IF D1 < 130 THEN 55
53  LET D0 = D1 - 130
54  GOTO 60
55  LET D0 = 130 + P1
60  PRINT "DONNEES ? (FIN TAPER 000, 05, 3 !!!!)"
70  LET I = 1
80  INPUT Z, U, D
100  LET AC I, 11 = Z
110  LET AC I, 21 = U
120  LET AC I, 31 = D
130  LET I = I + 1
131  LET N = I
140  INPUT Z, U, P
150  IF Z = 000 THEN 130

```

```

170  GOTO 100
180  LET X0 = 90 * COS(3.14159 * (D0 - 10) / 130)
181  LET Y0 = 90 * SIN(3.14159 * (D0 - 10) / 130)
182  PRINT
183  PRINT
184  PRINT
185  PRINT
186  PRINT
187  PRINT
188  PRINT
189  PRINT
190  PRINT
191  PRINT
192  PRINT
193  PRINT
194  PRINT
195  PRINT "STATION: "; S0
197  PRINT "DATE: "; J0
198  PRINT
199  PRINT
200  PRINT "      Z      MIT      GAR      T-R      S-J"
201  PRINT
202  PRINT
205  LET I = 1
220  LET Z = AC I, 11
230  LET U = AC I, 21
240  LET P = AC I, 31
250  LET X = 90 * COS(3.14159 * (D0 - 10) / 130)
260  LET Y = 90 * SIN(3.14159 * (D0 - 10) / 130)
270  LET X1 = X0 + X
280  LET Y1 = Y0 + Y
290  LET A0 = SQRT(X1 * X1 + Y1 * Y1)
300  LET I1 = ATN(Y1 / X1)
310  IF X1 < 0 THEN 340
320  LET D1 = 130 * (3.14159 / 2 - P1) / 3.14159
330  GOTO 350
340  LET D1 = 130 * (3.14159 * 3 / 2 - P1) / 3.14159
350  LET D1 = INT(D1)
360  LET A1 = INT(A0)
370  LET X1 = INT(X1)
380  LET Y1 = INT(Y1)
390  PRINT "      "; Z; A0; D1; X1; Y1
400  IF I = 3 - 1 THEN 400

```

410 LET I = I + 1
420 GOTO 220
430 PRINT
440 PRINT
460 PRINT
470 PRINT
480 END

FICHE TECHNIQUE DU PROGRAMME

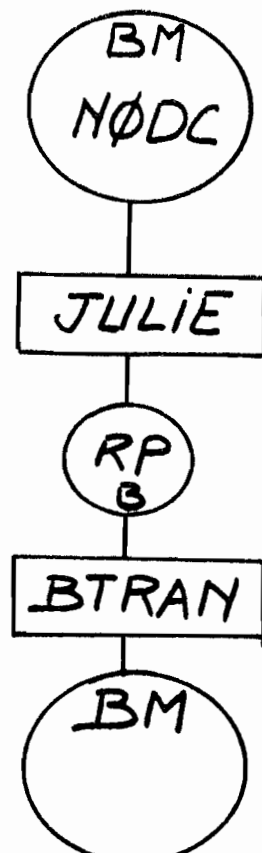
NOM: JULIE

TYPE	CHAINE	BT	PERIPH.	IN	7
BUT	Traduction de la bande BT du NODC Décodage et transcription sortie sur RP en extraire, ou catalogue sur TTY		OUT	4,2	
Observations			FORMAT	IN	
Opérations			OUT		
			ETIQ.	IN	
			OUT		
DATE DE MISE AU POINT	16/10/74 modifié 15/11/76		PROGRAMMEUR		
LANGAGE UTILISE	ASSEMBLER		Moline		
SSP EMPLOYES					
Starting ADDRESS	3341				
Nb Source	Nb Objet	Nb Absolu			
Divers					
Clés employées	15 pour fin de perf.				

FICHE EXPLOITATION

Les rubans perforés issus de Julie nécessitent
l'emploi de BTRAN pour être mis sur bande magnétique
au format ORSTOM.

- a) SA = 3341 PRESET RUN
- b) HALT T = 102077
lever la clé 2 ↑ (OUT = 4)
RUN
- c) HALT T = 102007
A. Nb d'regist. BM lus
clés: = nb d'regist. à sauter (total)
- d) RUN HALT T = 102000
baisser les clés



```
0001      NAM JULIE
0002  NBRCD OCT 0
0003  N1200 OCT 0
0004  UNIT  BSS 1
0005  C1200 BSS 1
0006  C120  BSS 1
0007  A1200 BSS 1
0008  A120  BSS 1
0009  DATA BSS 600
0010  I120  BSS 60
0011  IT     BSS 21
0012  CONT  BSS 1
0013  CNTR  BSS 1
0014  N     BSS 1
0015  X     BSS 1
0016  Y     BSS 1
0017  Z     BSS 1
0018  CMTR  BSS 1
0019  RRE   OCT 23
0020  ADDR  DEF DATA
0021  ADI   DEF I120
0022  ADIT  DEF IT
0023  NORD  ASC 1,N
0024  SUD   ASC 1,S
0025  EST   ASC 1,E
0026  WEST  ASC 1,W
0027  FMTT  ASC 9,(4I2,I4,A1,I5,A1)
0028  FMT1  ASC 28,(4A2,1X2A2,1X4I2,5X14,I5,7X2A2,27X12,9XA1,18XA1,6X2I2
0029        ASC 1,)
0030  FMT3  ASC 5,(10X20I3)
0031        ENT SORT1,BCDAS,CAR,JULIE,FMBT1,FMBT3,DOVER
0032        EXT .DIO.,.IOI.,.IOR.,.IAR.,.DTA.,CLRIO,ENDIO
0033        EXT .BIO.
0034  COM   ASC 15,("UNITE SORTIE,RCD A SAUTER")
0035  FMTRD ASC 9,(/"RCD LUS ="15)
0036  JULIE NOP
0037        CLA
0038        STA NBRCD
0039        STA N1200
0040        LDA =D2
0041        CLB
0042        JSB .DIO.
0043        DEF COM
0044        DEF F1
0045        JSB .DTA.
0046  F1    LDA =B1
0047        CLB,INB
0048        JSB .DIO.
0049        OCT 0
0050        DEF F2
0051        JSB .IOI.
0052        STA UNIT
0053        JSB .IOI.
0054        CMA,INA
0055        STA NBRCD
0056  F2    JSB ENDIO
0057        DEF *+1
0058        JMP X1200
0059  JULE  LDA =B2
```

```
0060      CLB
0061      JSB .DIO.
0062      DEF FMTRD
0063      DEF F4
0064      LDA NI200
0065      JSB .IOI.
0066      JSB .DTA.
0067  F4   JSB ENDIO
0068      DEF *+1
0069      HLT
0070      JMP JULIE
0071  SAUT LDA NBRCD
0072      INA
0073      STA NBRCD
0074      JSB ENDIO
0075      DEF *+1
0076  X1200 LIA 01
0077      SSA,RSS
0078      JMP X4
0079      LDA =D99
0080      STA IT+20
0081      LDA UNIT
0082      CPA =D4
0083      JSB ECRIT
0084      JMP JULE
0085  X4   NOP
0086      *
0087      *   LECTURE DE 1200 CARACTERES SUR 2020
0088      *
0089      CLC 0,C
0090      LDA =D-600
0091      STA CONT
0092      LDB ADDR
0093      STB X
0094      LDA RRE
0095      OTA 11B
0096  BIS  SFS 10B
0097      JMP *-1
0098      LIA 10B,C
0099      AND =B77
0100      CPA =B77
0101      JMP BIS
0102      ALF,ALF
0103      SFS 10B
0104      JMP *-1
0105      LIB 10B,C
0106      ADA 1
0107      STA X,I
0108      ISZ X
0109      ISZ CONT
0110      JMP BIS
0111      SFS 11B
0112      JMP *-1
0113      CLC 10B
0114      LDA NI200
0115      INA
0116      STA NI200
0117      LDA NBRCD
0118      SZA
0119      JMP SAUT
```

```
0120 *
0121 * APPEL DECODAGE, RANGEMENT DANS DATA
0122 *
0123 LDA =D-600
0124 STA CNTR
0125 LDB ADDR
0126 S2 LDA 1,1
0127 STB N
0128 JSB BCDAS
0129 LDB N
0130 STA 1,1
0131 INB
0132 ISZ CNTR
0133 JMP S2
0134 JSB CLR10
0135 DEF *+1
0136 *
0137 * DECOUPAGE PAR BLOC 120
0138 *
0139 LDA =D-10
0140 STA C1200
0141 LDA ADDR
0142 STA A1200
0143 B120 LDA =D-50
0144 STA C120
0145 LDA ADI
0146 STA A120
0147 RANG LDA A1200,I
0148 STA A120,I
0149 LDB A1200
0150 INB
0151 STB A1200
0152 LDB A120
0153 INB
0154 STB A120
0155 ISZ C120
0156 JMP RANG
0157 JSB SORTI FIN BLOC 120
0158 ISZ C1200
0159 JMP B120
0160 JMP X1200 FIN BLOC 1200
0161 SORTI NOP IMPRESSION BLOC 120
0162 LDA I120+39 RECHERCHE TYPE
0163 CPA =B32160 50
0164 JMP SORTI,I
0165 CPA =B30062
0166 JMP SORTI,I 02
0167 CPA =B30061
0168 JMP TYPI 01
0169 JSB FMBT3 03,04,05,06,07
0170 JMP SORTI,I
0171 TYPI JSB FMBTI
0172 JMP SORTI,I
0173 *
0174 * DECODAGE BCD-ASCII DANS A
0175 *
0176 BCDAS NOP
0177 STA X
0178 AND =B77
0179 JSB CAR
```

```
0180      STA Y
0181      LDA X
0182      ALF,ALF
0183      AND =B77
0184      JSB CAR
0185      ALF,ALF
0186      ADA Y
0187      JMP BCDAS,I
0188 CAR   NOP
0189      STA Z
0190      LDB =D-64
0191      STB CMTR
0192      LDB ASCII
0193 SUITE LDA Z
0194      CPA I,I
0195      JMP TRAD
0196      INB
0197      ISZ CMTR
0198      JMP SUITE
0199      LDA =B40
0200      JMP CAR,I
0201 TRAD  LDA ASCII
0202      CMA,INA
0203      ADB 0
0204      ADB =B40
0205      LDA I
0206      JMP CAR,I
0207 ASCII DEF CODE
0208 CODE  OCT 20,52,37,13,53,34,60,36,75,55,54,60,33,40,73,21
0209      OCT 12,1,2,3,4,5,6,7,10,11
0210      OCT 15,56,76,35,16,72,14
0211      OCT 61,62,63,64,65,66,67,70,71
0212      OCT 41,42,43,44,45,46,47,50,51
0213      OCT 22,23,24,25,26,27,30,31
0214      OCT 75,55,20,77,32
0215 *
0216 *      MISE AU FORMAT DES DONNEES BT TYPE 01 03
0217 *
0218 FMBTI NOP
0219      LDA I120+13      LATITUDE
0220      LDB 0
0221      AND =B100
0222      SZA
0223      JSB DOVER
0224      STB I120+13
0225      LDA I120+15      LONGITUDE
0226      LDB 0
0227      AND =B100
0228      SZA
0229      JSB DOVER
0230      STB I120+15
0231      LDA I120+42      LAT OVERP
0232      AND =B77
0233      CPA =B40
0234      JMP *+3
0235      LDA SUD
0236      JMP *+2
0237      LDA NORD
0238      ALF,ALF
0239      STA I120+42
```

```
0240 LDA I120+52 LON OVERP
0241 ALF,ALF
0242 AND =B77
0243 CPA =B40
0244 JMP *+3
0245 LDA EST
0246 JMP *+2
0247 LDA WEST
0248 STA I120+52
0249 CLA CONVERSION BINAIRE
0250 CLB,INB
0251 JSB .D10.
0252 DEF I120
0253 DEF FMT1
0254 DEF *+4
0255 LDA =D20
0256 LDB ADIT
0257 JSB .IAR.
0258 LDA IT+14
0259 CLB,INB
0260 CMB
0261 ADA I
0262 STA IT+14
0263 LDA =D1
0264 STA IT+20
0265 JSB ECRIT
0266 JMP FMBT1,I
0267 FMBT3 NOP
0268 CLA
0269 CLB,INB
0270 JSB .D10.
0271 DEF I120
0272 DEF FMT3
0273 DEF *+4
0274 LDA =D20
0275 LDB ADIT
0276 JSB .IAR.
0277 LDA =D3
0278 STA IT+20
0279 JSB ECRIT
0280 JMP FMBT3,I
0281 ECRIT NOP
0282 LDA UNIT
0283 CLB ECRITURE
0284 CPA =D2
0285 JMP TTY
0286 JSB .B10.
0287 LDA =D21
0288 LDB ADIT
0289 JSB .IAR.
0290 JSB .DTA.
0291 FIN3 JSB ENDI0
0292 DEF *+1
0293 JMP ECRIT,I
0294 TTY LDA IT+20
0295 CPA =D3
0296 JMP ECRIT,I
0297 JSB .D10.
0298 DEF FMTT
0299 DEF F3
```



```
0300 LDA IT+6
0301 JSB .IOI.
0302 LDA IT+7
0303 JSB .IOI.
0304 LDA IT+8
0305 JSB .IOI.
0306 LDA IT+9
0307 JSB .IOI.
0308 LDA IT+10
0309 JSB .IOI.
0310 LDA IT+11
0311 JSB .IOI.
0312 LDA IT+16
0313 JSB .IOI.
0314 JSB .DTA.
0315 F3 JMP FIN3
0316 DOVER NOP
0317 STB N
0318 LDA N
0319 AND =B177
0320 LDB =B31
0321 CMB, INB
0322 ADB 0
0323 LDA N
0324 AND =B177400
0325 ADB 0
0326 JMP DOVER, I
0327 END JULIE
```

11.309
LDA IT+15
JSB .IOI.

FICHE TECHNIQUE DU PROGRAMME

NOM: BTMAP

TYPE	PLOTAGE		CHAINE	BT	PERIPH.	IN	5,7	
BUT	fait une carte des positions des BT pour le ou les mois choisis.				PERIPH.	OUT	2	
Observations	M(12)				FORMAT	IN	(B)NODC	
Opérations					FORMAT	OUT	P	
DATE DE MISE AU POINT	NOV 74				ETIQ.	IN	3.11	
LANGAGE UTILISE	FORTRAN				ETIQ.	OUT		
SSP EMPLOYES					PROGRAMMEUR			
Starting ADDRESS	2042				DARBOIS			
Nb Source	1	Nb Objet	1	Nb Absolu				
Divers								
Clés employées	↑ stop le plotage certains mois?							

FICHE EXPLOITATION

*
/L
/E

SYMBOLIC FILE SOURCE DEVICE?
/P

```
0001          PROGRAM BTMAP
0002          DIMENSION IR(20),M(12)
0003          N=1168
0004          IE=1058
0005          WRITE(2,200)
0006          200 FORMAT("BTMAP"/"LAT MIN-MAX(D) N+ S-")
0007          READ(1,*)AMIN,AMAX
0008          AMIN=AMIN*60.
0009          AMAX=AMAX*60.
0010          WRITE(2,201)
0011          201 FORMAT("LON MIN-MAX(D) E+ W-")
0012          READ(1,*)OMIN,OMAX
0013          OMIN=OMIN*60.
0014          OMAX=OMAX*60.
0015          WRITE(2,501)
0016          501 FORMAT("IN")
0017          READ(1,*)IN
0018          600 WRITE(2,500)
0019          500 FORMAT("PLTT"/"MOIS:99,99,99,...")
0020          DO 520 I=1,12
0021          520 M(I)=0
0022          READ(1,*)M
0023          WRITE(2,202)
0024          202 FORMAT("PLTP")
0025          5 CALL CLE(KC)
0026          IF(KC)600,25,600
0027          25 READ(IN)IR
0028          IF(IR(18)-999)30,999,999
0029          30 IF(IR(8))5,5,31
0030          31 IF(IR(8)-12)5,26,5
0031          26 DO 20 I=1,12
0032          IF(IR(8)-M(I))20,21,20
0033          20 CONTINUE
0034          GO TO 5
0035          21 CONTINUE
0036          12 XLAT1=FLOAT(IR(11))/100.
0037          LAT2=IR(16)
0038          XLON1=FLOAT(IR(12))/100.
0039          LON2=IR(17)
0040          IF(LAT2-N)1,2,1
0041          1 XLAT1=-XLAT1
0042          2 ILA=IFIX(XLAT1)
0043          XLAT1=100.*XLAT1-40.*FLOAT(ILA)
0044          IF(LON2-IE)3,4,3
0045          3 XLON1=-XLON1
0046          4 ILO=IFIX(XLON1)
0047          XLON1=100.*XLON1-40.*FLOAT(ILO)
0048          IX=IFIX(9999.*(XLON1-OMIN)/(OMAX-OMIN))
0049          IY=IFIX(9999.*(XLAT1-AMIN)/(AMAX-AMIN))
0050          WRITE(2,100)IX,IY
0051          100 FORMAT(I4,I5)
0052          999 REWIND 7
0053          GOTO 600
0054          GOT05
0055          END
0056          ENDS
```

FICHE TECHNIQUE DU PROGRAMME

NOM: BTRAN

TYPE	CHAINE			BT	PERIPH.	IN	5
BUT	Copie sur BM du ruban perforé issue de JULIE. liste en option des étiquettes déjà enregistrées qualification des la 1 ^{re} station en (15) transfert avec listing des obj en option fin : fin de RP en (15)↑			EFORMAT	OUT	7	
Observations					IN	B	
Opérations				ETIQ.	OUT	B	
DATE DE MISE AU POINT				PROGRAMMEUR	IN	MODC	
LANGAGE UTILISE	FORTRAN			Mullier.	OUT	NOOC	
SSP EMPLOYES							
Starting ADDRESS	2042						
Nb Source	1	Nb Objet	1		Nb Absolu	0	
Divers							
Clés employées	15 ↑ stop 999 ↓ m>0↑ ecriteur etiquette						

FICHE EXPLOITATION

```
0001      PROGRAM BTRAN
0002      DIMENSION IR(20),ID(20)
0003 1000  WRITE(2,502)
0004      IE=0
0005      IST=0
0006      REWIND 7
0007      1  READ(7)IR
0008      IE=IE+1
0009      IF(IR(13)-999)3,2,2
0010      3  IF(IR(13))1,1,4
0011      4  IF(IR(3)-12)6,6,1
0012      6  IST=IST+1
0013      DO 9 I=1,20
0014      9  ID(I)=IR(I)
0015      CALL CLC(KC)
0016      IF(KC)99,1,10
0017      10 WRITE(2,500)IST,IE
0018      GOTO 1
0019      2  BACKSPACE 7
0020      IK=2
0021      WRITE(2,503)IE
0022 503  FORMAT('NE D'ENRESETEMENTS=',I3)
0023      WRITE(2,500)IST,IE
0024      5  READ(5)IR
0025      IF(IR(13))5,5,3
0026      3  IF(IR(3)-12)100,100,5
0027 100  IF(IR(3)-IR(13))5,7,5
0028      7  WRITE(2,500)IK,IE
0029      GOTO(11,14,20),IK
0030      14 WRITE(2,501)
0031      READ(1,*)IK
0032      GOTO(11,5),IK
0033      11 WRITE(7)IE
0034      IST=IST+1
0035      NE=1
0036      20 READ(5)IR
0037      IF(IR(4)-ID(4))30,31,30
0038      30 IF(IR(13))24,24,21
0039      21 IF(IR(3)-12)22,22,24
0040      22 IST=IST+1
0041      NE=0
0042      CALL CLC(KC)
0043      IF(KC)9,24,12
0044      12 WRITE(2,500)IST,IE
0045      24 WRITE(7)IR
0046      NE=NE+1
0047      DO 23 I=1,20
0048      23 ID(I)=IR(I)
0049      GOTO 20
0050      31 DO 33 I=1,NE
0051      33 BACKSPACE 7
0052      99 DO 32 I=1,20
0053      32 IR(I)=99)
0054      WRITE(7)IE
0055      REWIND 7
0056 999  GOTO 1000
0057 500  FORMAT(I5,2X6A2,I3,2(I4/I2),I4,2(I6,1X02),2(I7/I5,1X13,1X14),
0058      1,4I3)
0059 501  FORMAT('TRANSFERT (UI=1,NOV=2')
```

```
PROGRAM TCLIN
  DIMENSION IMET(20), IR(20), JD(100), IM(12), ISM(140)
100 FORMAT("**TCLIN**"/"CHOIX DU MARS DEN ?")
101 FORMAT("DES SOUS MARS DEN...,FIN 999")
102 FORMAT("DES MOIS...,FIN 99")
103 FORMAT("/10X"THERMOCLINE"9X"*TEMP* COUCHE * DATE HEURE
  1 POSITION"/"PROF. MOY.-EPAISSEUR-INTENSITE*SURF*BOMME
  JENE*"/)
10 FORMAT("PLTL")
11 FORMAT("PLTT")
12 FORMAT("CLE: FIN, RESULT, RESULT+GRAPH"/"IN 5 OU 7 ?")
104 FORMAT(/,3X13,8X13,5XF6.2,3XF4.1,2(3X13),2(CIN/10),13,14B,
  1F6.2,A1,F6.2,A1)
105 FORMAT(3X13,8X13,5XF6.2)
106 FORMAT(2I5)
  NPT=0
  WRITE(2,100)
  READ(1,*)IMA
  WRITE(2,101)
  I=0
1  I=I+1
  READ(1,*)ISM(I)
  IF(ISM(I)-999)1,15,15
15 WRITE(2,102)
  I=0
2  I=I+1
  READ(1,*)IM(I)
  IF(IM(I)-99)2,20,20
20 WRITE(2,12)
  READ(1,*)IN
  WRITE(2,103)
250 IW=1
25 READ(IN)(IR(I),I=1,20)
  CALL CLE(KC)
  IF(KC)33,37,37
33 PAUSE
  GOTO 25
37 IF(IR(18)-999)22,23,23
23 IF(IN-7)22,33,22
22 ND=IR(15)
  IF(IR(8))25,25,24
24 IF(IR(8)-12)26,26,25
26 IF(IR(20)-IMA)25,27,25
27 I=0
28 I=I+1
  IF(ISM(I)-999)29,25,25
29 IF(ISM(I)-IR(19))28,30,28
30 I=0
31 I=I+1
  IF(IM(I)-99)32,25,25
32 IF(IM(I)-IR(8))31,40,31
40 DO 5 I=1,20
  5 IMET(I)=IR(I)
  K=0
45 READ(IN)(IR(I),I=1,20)
  DO 50 J=1,20
  L=J+K
  JD(L)=IR(J)
50 CONTINUE
  NT=L
  K=K+20
  ND=ND-1
```

```
      IF(ND)60,60,45
60 IF(JD(NT),59,59,61
59 NT=NT-1
   GOTO 60
61 I=0
   IT=JD(1)-10
   TS=0.1*FLOAT(JD(1))
62 I=I+1
   IF(I-NT)63,65,65
63 IF(JD(I)-IT)64,65,62
64 IHOM=5*(I-2+(JD(I-1)-IT)/(JD(I-1)-JD(I)))
   GOTO 66
65 IHOM=(I-1)*5
66 I=1
75 IE=3
   IDT=0
   GRAD=0.
   IP=0
67 I=I+1
   IF(NT-I)70,68,68
68 IF(JD(I-1)-JD(I)-5)67,69,69
69 IE=IE+5
   IDT=IDT+JD(I-1)-JD(I)
   IP=(I-1)*5-IE/2
   GRAD=FLOAT(IDT)/(FLOAT(IE)*13.)
   IF(JD(I)-JD(I+1)-5)70,67,67
70 GOTO (71,72),IW
71 AT=FLOAT(IMET(11))*0.81
   ON=FLOAT(IMET(12))*0.81
   WRITE(2,104)IP,IE,GRAD,TS,IHOM,IMET(7),IMET(8),IMET(9),
   IMET(10),AT,IMET(15),ON,IMET(17)
   IW=2
   GOTO 74
72 IF(IDT-5)74,73,73
73 WRITE(2,105)IP,IE,GRAD
74 IF(NT-I)80,80,75
80 CALL CLE(KC)
   IF(KC)999,250,90
90 NPT=NPT+1
   IPAS=IFIX(9999./25.)
   IF(NPT-15)91,91,92
91 MINT=(NPT-1)*IPAS
   MINZ=0
   GOTO95
92 IF(NPT-30)93,93,94
94 PAUSE
   NPT=0
   GOTO 90
93 MINT=(NPT-15)*IPAS
   MINZ=IFIX(9999.*20./38.)
95 WRITE(2,10)
   DO 200 I=1,NT
   IY=(JD(I)-100)*IFIX(9999./537.)+MINT
   IX=(I-1)*IFIX(9999./152.)+MINZ
   WRITE(2,105)IX,IY
200 CONTINUE
   WRITE(2,11)
   GOTO 250
999 PAUSE
   END
   ENDS
```

FICHE TECHNIQUE DU PROGRAMME

NOM: TEMP

TYPE		CHAINE		PERIPH.	IN	5		
BUT	Conction de thermomètres protégés. Immersion réelle à l'aide de thermomètres non protégés			OUT	2			
				FORMAT	IN	A		
Observations				OUT	A			
Opérations				ETIQ.	IN	5		
DATE DE MISE AU POINT	DEPUIS...			OUT	5			
LANGAGE UTILISE	FORTRAN			PROGRAMMEUR				
SSP EMPLOYES				Morlière				
Starting ADDRESS	2							
Nb Source	1	Nb Objet	1				Nb Absolu	1
Divers								
Clés employées								

FICHE EXPLOITATION

TEMP

```
FUNCTION CALIB(M,X,Y,T)
DIMENSION X(1),Y(1)
I=1
IF(T-X(1))4,4,2
2 DO 10 I=2,M
IF(T-X(I))3,4,10
3 A=(Y(I)-Y(I-1))/(X(I)-X(I-1))
B=Y(I)-A*X(I)
CALIB=A*T+B
GOTO 25
10 CONTINUE
4 CALIB=Y(I)
25 RETURN
END
FUNCTION CORT(TLU,TAUX,TAB,TW,N)
DIMENSION TAB(1),Y(7),TP(7),TA(5)
TA(1)=15.
TP(1)=0.
DO 2 I=2,5
2 TA(I)=TA(I-1)+5.
DO 20 J=10,14
L=J-9
20 Y(L)=TAB(J)
TAX=TAUX+CALIB(5,TA,Y,TAUX)
DO 15 J=3,9
L=J-2
15 Y(L)=TAB(J)
IF(N-1)10,10,12
10 DO 1 I=2,7
1 TP(I)=TP(I-1)+5.
C=(TLU+TAB(2))*(TLU-TAX)/(TAB(15)-2.*TLU-TAB(2)+TAX)
GOTO 25
12 DO 3 I=2,7
3 TP(I)=TP(I-1)+10.
C=(TLU+TAB(2))*(TW-TAX)/(TAB(15)-TW+TAX)
25 CORT= TLU+C+CALIB(7,TP,Y,TLU)
RETURN
END
PROGRAM TEMP
DIMENSION IC(110,16),CARAC(16),TAB(15)
WRITE(2,500)
500 FORMAT("PROGRAMME TEMP"/"PLACER CARAC. DANS LECTEUR")
DO 400 K=1,110
READ(5,*)(CARAC(I),I=1,16)
IC(K,1)=IFIX(CARAC(1))
IC(K,2)=IFIX(CARAC(2))
DO 401 L=3,9
401 IC(K,L)=IFIX(100.*CARAC(L))
DO 402 L=10,14
402 IC(K,L)=IFIX(10.*CARAC(L))
IC(K,15)=IFIX(CARAC(15))
400 IC(K,16)=IFIX(CARAC(16))
WRITE(2,99)
99 FORMAT(" TAPER IMM.,NUMERO,PRINC.,AUX.,"/)
```

```
1 READ(1,*)IMM,NUM,TPRIN,TAUX
  DO 3 J=1,110
    IF(IC(J,1)-NUM)3,4,3
  4 TAB(1)=FLOAT(IC(J,1))
    TAB(2)=FLOAT(IC(J,2))
    DO 600 K=3,9
  600 TAB(K)=FLOAT(IC(J,K))/130.
    DO 601 K=10,14
  601 TAB(K)=FLOAT(IC(J,K))/10.
    TAB(15)=FLOAT(IC(J,15))
    Q=FLOAT(IC(J,16))/100000.
    GOTO 6
  3 CONTINUE
  WRITE(2,104)NUM
  104 FORMAT(I3," INCONNU")
  GOTO 15
  6 IF(Q)7,7,2
  7 T=CORT(TPRIN,TAUX,TAB,0.,1)
  WRITE(2,101)IMM,NUM,TPRIN,TAUX,T
  101 FORMAT(25X,I4,1X,I5,F6.2,F5.1,F8.2)
  15 WRITE(2,202)
  202 FORMAT("?")
  GOTO 1
  2 WRITE(2,103)
  103 FORMAT(" N-P    TAPER TW")
  READ(1,*)TW
  T=CORT (TPRIN,TAUX,TAB,TW,2)
  Z=(T-TW)/(1.02726*Q)
  WRITE(2,105)IMM,NUM,TPRIN,TAUX,T,TW,Z
  105 FORMAT(25X,I4,1X,I5,F6.2,F5.1,F8.2,F6.2,F6.3)
  GOTO 15
  END
EN
  END3
```


FICHE TECHNIQUE DU PROGRAMME

NOM:

ORG 13142 B

TYPE	trait.	CHAINE	PERIPH.	IN	5
BUT	Lecture sur Lecteur rapide 5 en basic		OUT	2	
Observations			FORMAT	IN	
Opérations			OUT		
DATE DE MISE AU POINT	14 Dec 74		ETIQ.	IN	
LANGAGE UTILISE	ASMB		OUT		
SSP EMPLOYES			PROGRAMMEUR		
Starting ADDRESS	13142		NOEL		
Nb Source	Nb Objet				
Divers					
Clés employées					

FICHE EXPLOITATION

```

ORG 13142B
SBTLB OCT 401
DEF CLE
ENDTB EQU *
CLE NOP
AD1 DE
AD1 DEF 0,I
LDA AD1,I

LEC STC 15B
LIB 15B
SZB
JMP SUI
JUM LEC
JMP LEC
SUI REP 13B
BLS
REP 13B
BRS
STB 0,I
JMP CLE,I
LSTWD EQU *
ORG 110B
DEF LSTWD
ORG 121
ORG 121B
DEF SBTLB
DEF ENDTB
END
    
```

en BASIC

CALL (1,K)

K sur en 5 est transmis

observation préalablement LET K = 20

voir programme de traitement d'une station en BASIC

6-158

FICHE TECHNIQUE DU PROGRAMME

NOM: **ANDER**

TYPE	CHAINE			PERIPH.	IN	
BUT	permet de passer du code FFI-PS à des nombres compris entre 0 - 1024			FORMAT	OUT	
Observations				ETIQ.	IN	
Opérations					OUT	
DATE DE MISE AU POINT				PROGRAMMEUR		
LANGAGE UTILISE				Morlière		
SSP EMPLOYES						
Starting ADDRESS						
Nb Source	Nb Objet		Nb Absolu			
Divers						
Clés employées						

FICHE EXPLOITATION

- Decode les rubans perforés de Courantométrie AANDERAA
 - lecture des bandes issues du Decodeur AANDERAA
- et donne pour chacun des 6 canaux 1 nombre compris entre 0 et 1024, la 7^e valeur fait le cumul des erreurs, la 8^e identifie la fin de bande perforée.

Note - La version ANDB permet de passer directement du decodeur AANDERAA à l'ordinateur avec l'interface appropriée

- Dans cette version c'est l'ordinateur qui commande la lecture du ruban magnétique par le DECODEUR

*
 /L
 /E

G-159

SYMBOLIC FILE SOURCE DEVICE?
 /P

0001		NAM	ANDER		
0002		ENT	ANDER		
0003		EXT	.ENTR, ENDI C		
0004	IAA	ESS	8		
0005	ANDER	NOP		0060	ADA IAA+6, i
0006		JSE	.ENTR	0061	STA IAA+6, i
0007		DEF	IAA	0062	JSE CARAC
0008		JSE	ENDI C	0063	JMP MCT, i
0009		DEF	*+1	0064	SUI T1 LDA X
0010		CLA		0065	STA PARA
0011		STA	IAA+6, i	0066	JSE CARAC
0012		STA	IAA+7, i	0067	LDA CA6
0013		LDA	M15	0068	SZA, RSS
0014		STA	CNTEL	0069	JMP SUI T3
0015		LDA	M6	0070	LDA ERROR
0016		STA	CNTR	0071	ADA IAA+6, i
0017	ELANC	JSE	CARAC	0072	STA IAA+6, i
0018		LDA	X	0073	SUI T3 LDA CA7
0019		SZA		0074	SZA, RSS
0020		JMP	NOVEL	0075	JMP SUI T2
0021		ISZ	CNTEL	0076	LDA ERROR
0022		JMP	ELANC	0077	ADA IAA+6, i
0023		LDA	ERROR	0078	STA IAA+6, i
0024		STA	IAA+7, i	0079	JMP MCT, i
0025		JMP	FIN	0080	SUI T2 LDA X
0026	NOVEL	LDA	M15	0081	ALF, ALS
0027		STA	CNTEL	0082	ADA PARA
0028		LDA	ADD	0083	STA PARA
0029		STA	MCT	0084	JMP MCT, i
0030		LDA	IAA	0085	CARAC NOP
0031		STA	N	0086	CLF 0
0032		JMP	MCT+2	0087	STC 15E, C
0033	AGAIN	JSE	MCT	0088	SFS 15E
0034	DEEUT	LDA	PARA	0089	JMP *-1
0035		LDE	N	0090	LIE 15E
0036		STA	1, i	0091	LDA 1
0037		INE		0092	AND =E37
0038		STE	N	0093	STA X
0039		ISZ	CNTR	0094	LDA 1
0040		JMP	SYN	0095	AND =F40
0041		LDA	SYNC	0096	STA CA6
0042		SZA		0097	LDA 1
0043		JMP	FIN	0098	AND =E100
0044		LDA	ERROR	0099	STA CA7
0045		ADA	IAA+6, i	0100	CLC 15E
0046		STA	IAA+6, i	0101	JMP CAFAC, i
0047		JMP	FIN	0102	ADD DEF DEEUT
0048	SYN	LDA	SYNC	0103	X ESS 1
0049		SZA		0104	CA6 ESS 1
0050	FIN	JMP	ANDER, i	0105	CA7 ESS 1
0051		JMP	AGAIN	0106	PARA ESS 1
0052	MCT	NOP		0107	CNTR ESS 1
0053		JSE	CARAC	0108	ERROR CCT 1
0054		LDA	CA6	0109	M6 DEC -6
0055		STA	SYNC	0110	M15 DEC -45
0056		LDA	CA7	0111	CNTEL ESS 1
0057		SZA		0112	N ESS 1
0058		JMP	SUI T1	0113	SYNC ESS 1
0059		LDA	ERROR	0114	END ANLER

Λ
/E

G-160

SYMBOLIC FILE SOURCE DEVICE?
/P

```
0001      NAM ANDRC
0002      ENT ANDRC
0003      EXT .ENTR, ENDI C
0004  PARA  ESS 14
0005  ANDRC NCP
0006      JSE .ENTR
0007      DEF PARA
0008      JSE ENDI C
0009      DEF *+1
0010      CLA
0011      STA PARA+12,i
0012      STA PARA+13,i
0013      STA SYNCR
0014      LDA M6
0015      STA CNTR
0016      LDA PARA
0017      STA W .
0018  LECT  JSE AA1
0019      SSA
0020      JMP ECR
0021      LDE W
0022      STA 1,i
0023      INB
0024      STE W
0025      ISZ CNTR
0026      JMP LECT
0027      JSE FAUTE
0028      JSE AA1
0029      SSA, RSS
0030      JMP *-2
0031      JMP ANDRC,i
0032  ECR   AND =E77777
0033      LDE W
0034      STA 1,i
0035      ISZ CNTR
0036      JSE FAUTE
0037      JMP ANDRC,i
0038  AA1   NCP
0039      STC 14E,C
0040      SFS 14E
0041      JMP *-1
0042      LIA 14E
0043      LIE 14E
0044      CMA
0045      CME
0046      SSE
0047      JMP NOREC
0048      SLE
0049      JSE FAUTE
0050  CUT   CLC 14E,C
0051      JMP AA1,i
0052  NOREC LDE ERROR
0053      STE PARA+13,i
0054      CLC 14E,C
0055      JMP ANDRC,i
0056  FAUTE NCP
0057      LDE ERROR
0058      ADE PARA+12,i
0059      STE PAFA+12,i
```

G-161

```
0060          JMP FAUTE,i
0061 N        ESS 1
0062 SYNCR   ESS 1
0063 CNTR    ESS 1
0064 ERROR   CCT 1
0065 M6      DEC -12
0066          END ANDRC
```


FICHE TECHNIQUE DU PROGRAMME

 SSP
 NOM: CLE

TYPE	CHAINE			PERIPH.	IN	
BUT	Chargement du registre de clef dans l'argument du sous programme.			FORMAT	OUT	
Observations				ETIQ.	IN	
Opérations					OUT	
DATE DE MISE AU POINT				PROGRAMMEUR		
LANGAGE UTILISE	ASSEMBLEUR			MORLIERE		
SSP EMPLOYES						
Starting ADDRESS						
Nb Source	Nb Objet		Nb Absolu			
Divers						
Clés employées	toutes					

FICHE EXPLOITATION

 ISW
 CLE

 NAM CLE
 ENT CLE
 EXT .ENTR
 BSS I
 NOP
 JSB .ENTR
 DEF ISW
 LIA 01
 STA ISW, I
 JMP CLE, I
 END

CLF

FICHE TECHNIQUE DU PROGRAMME

NOM: SSP OMEGA

TYPE	CHAINE	PERIPH.	IN	OUT
BUT	Horloge ,l'argument egale/10seconde		FORMAT	IN
Observations			ETIQ.	IN
Opérations				OUT
DATE DE MISE AU POINT			PROGRAMMEUR	
LANGAGE UTILISE	ASSEMBLEUR		MORLIERE	
SSP EMPLOYES				
Starting ADDRESS				
Nb Source	Nb Objet	Nb Absolu		
Divers				
Clés employées				

FICHE EXPLOITATION

```

-      NAM OMEGA
      ENT OMEGA
      EXT .ENTR
NB     DEC 0
N      BSS 1
-      OMEGA NOP
      JSB .ENTR
      DEF N
      LDA N, I
      CMA, INA
      STA NB
      LDA DS
      CMA, INA
      STA CONT
      ISZ CONT
      JMP *-1
      ISZ NB
      JMP *-6
      JMP OMEGA, I
CONT  DEC 0
DS    DEC 15384
      END
**END-OF-TAPE
*

```

```

**END-OF-TAPE
*

```

SSP

NOM:

OSCAR

FICHE TECHNIQUE DU PROGRAMME

TYPE		CHAINE	SONDE	PERIPH.	IN	
BUT	Controle des entrees -sorties (LABEL / FINBM / NEUF / OUT10 / COPIE) sous programmes de SONDE			FORMAT	OUT	
Observations					IN	
Opérations				ETIQ.	OUT	
DATE DE MISE AU POINT	AVRIL 1974			PROGRAMMEUR		
LANGAGE UTILISE	ASSEMBLEUR			MORLIERE		
SSP EMPLOYES						
Starting ADDRESS						
Nb Source	Nb Objet		Nb Absolu			
Divers						
Clés employées						

FICHE EXPLOITATION

```
0001      NAM OSCAR
0002      EXT .IOC., .ENTR, .PAUS
0003      EXT STDC
0004      ENT OSCAR, STAT2, LFEED, BACON, SORTI
0005      FVT LABEL, STAT7, STAT4, NEUF, FINB4, COPIE
0006      ENT OUT40
0007  COM1  ASC 6, ** SONDE **
0008  COM2  ASC 8, CONTROLE LIAISON
0009  COM3  ASC 15, DATE, HEURE, CPOISIERE, STATION
0010  COM4  ASC 12, XX XX XX XXXX XXXX XXX
0011  ENTRP BSS 12
0012  I      BSS 1
0013  JP     BSS 1
0014  JT     BSS 1
0015  JS     BSS 1
0016  JO     BSS 1
0017  JC     BSS 1
0018  OSCAR NOP
0019      JSB .IOC.
0020      OCT 0
0021      JSB LFEED
0022      JSB LFEED
0023      JSB .IOC.
0024      OCT 20002
0025      JMP *-2
0026      DEF COM1
0027      DEC 6
0028      JSB LFEED
0029      JSB .IOC.
0030      OCT 20002
0031      JMP *-2
0032      DEF COM2
0033      DEC 8
0034      JSB STAT2
0035      JSB STDC
0036      DEF FIN1
0037      DEF JP
0038      DEF JT
0039      DEF JS
0040      DEF JO
0041      DEF JC
0042  FIN1  LDA JP
0043      JSB SORTI
0044      LDA JT
0045      JSB SORTI
0046      LDA JS
0047      JSB SORTI
0048      LDA JO
0049      JSB SORTI
0050      JSB LFEED
0051      JSB .IOC.
0052      OCT 20002
0053      JMP *-2
0054      DEF COM3
0055      DEC 15
0056      JSB .IOC.
0057      OCT 20002
0058      JMP *-2
0059      DEF COM4
```

```
0060      DFC -24
0061      JSB STAT2
0062      JSB .IOC.
0063      OCT 10401
0064      JMP *-2
0065      DEF ENTRP
0066      DEC -24
0067      JSB .IOC.
0068      OCT 40001
0069      SSA
0070      JMP *-3
0071      LDA =B7
0072      JSB .PAUS
0073      JMP OSCAL, I
0074  STAT2  NOP
0075      JSB .IOC.
0076      OCT 40002
0077      SSA
0078      JMP *-3
0079      JMP STAT2, I
0080  SORTI  NOP
0081      JSB BACON
0082      JSB .IOC.
0083      OCT 20002
0084      JMP *-2
0085      DEF N2
0086      DEC 2
0087      JSB STAT2
0088      JMP SORTI, I
0089  LFEED  NOP
0090      JSB .IOC.
0091      OCT 20002
0092      JMP *-2
0093      DEF LF
0094      DEC 1
0095      JSB STAT2
0096      JMP LFEED, I
0097  LF     OCT 212
0098  * SOUS PROGRAMME DE CONVERSION BINAIRE DECIMAL (ASCII)
0099  * NB EN A, RESULTAT EN N1, N2, N3 EN 3 MOTS
0100  BACON  NOP
0101      CLB
0102      DIV =D10000
0103      STA N1
0104      LDA 1
0105      CLB
0106      DIV =D1000
0107      STA N2
0108      LDA 1
0109      CLB
0110      DIV =D100
0111      STA N3
0112      LDA 1
0113      CLB
0114      DIV =D10
0115      STA N4
0116      STE N5
0117      LDA N1
0118      ADA =B60
0119      STA N1
```

```
0120      LDA N2
0121      ADA =B60
0122      ALF, ALF
0123      LDB N3
0124      ADB =B60
0125      ADA 1
0126      STA N2
0127      LDA N4
0128      ADA =B60
0129      ALF, ALF
0130      LDB N5
0131      ADB =B60
0132      ADA 1
0133      STA N3
0134      JMP BACON, I
0135  N1    BSS 1
0136  N2    BSS 1
0137  N3    BSS 1
0138  N4    BSS 1
0139  N5    BSS 1
0140  * ECRITURE ENTETE BM OU RP
0141  N      BSS 1
0142  LABEL  NOP
0143      JSB .ENTR
0144      DEF N
0145      LDA =D4
0146      CPA N, I
0147      JMP PERFO
0148      JSB .IOC.
0149      OCT 20007
0150      JMP *-2
0151      DEF ENTRP
0152      DEC 12
0153      JSB STAT7
0154  FIN   JMP LABEL, I
0155  PERFO JSB .IOC.
0156      OCT 20004
0157      JMP *-2
0158      DEF ENTRP
0159      DEC 12
0160      JSB STAT4
0161      JMP FIN
0162  STAT4  NOP
0163      JSB .IOC.
0164      OCT 40004
0165      SSA
0166      JMP *-3
0167      JMP STAT4, I
0168  STAT7  NOP
0169      JSB .IOC.
0170      OCT 40007
0171      SSA
0172      JMP *-3
0173      JMP STAT7, I
0174  NEUF   NOP
0175      JSB .ENTR
0176      DEF N
0177      LDA =D4
0178      CPA N, I
0179      JMP PERF9
```



```
0240      JSB .I OC.  
0241      OCT 20107  
0242      JMP *-2  
0243      DEF ID40  
0244      DEC 41  
0245      JSB STAT7  
0246  FIN40 JMP OUT40,I  
0247  PERF4 JSB .I OC.  
0248      OCT 20104  
0249      JMP *-2  
0250      DEF ID40  
0251      DEC 41  
0252      JSB STAT4  
0253      JMP FIN40  
0254      COM ID40(41)  
0255      END OSCAR
```

**END- OF- TAPE

*

FICHE TECHNIQUE DU PROGRAMME

NOM: SSP : SUITE

TYPE	CHAINE			PERIPH.	IN	
BUT	Appel IPL(XCCR1,XCCR2,XCCR3) <i>permet de chaîner les programmes à l'aide de B.M.</i>			FORMAT	OUT	
Observations				ETIQ.	IN	
Opérations					OUT	
DATE DE MISE AU POINT				PROGRAMMEUR		
LANGAGE UTILISE	ASSEMBLEUR			MORLIERE		
SSP EMPLOYES						
Starting ADDRESS						
Nb Source	Nb Objet		Nb Absolu			
Divers						
Clés employées						

FICHE EXPLOITATION

```

0001      JAM SUITE
0002      ENT XCCR2,XIVAR
0003  XCCF2  NOP
0004      LDA M3
0005      JSB .106B,I
0006      ASC 5,CCR2
0007  XIVAR  NOP
0008      LDA M3
0009      JSB .106B,I
0010      ASC 5,XIVAR
0011  M3     DEC -3
0012  .106B  ABS 100106B
0013      FND XCCR2
    
```

**EVD-OF-TAPE

*

ALPHA
SIGMA
SIGAL
OXY
SIGMZ

FICHE TECHNIQUE DU PROGRAMME

NOM: SSP.

TYPE		CHAINE		PERIPH.	IN
BUT	Calcul de Sigma et delta - alpha ... pourcentage d'oxygène dissous			FORMAT	OUT
Observations				ETIQ.	IN
Opérations					OUT
DATE DE MISE AU POINT				PROGRAMMEUR	
LANGAGE UTILISE	FORTRAN			MORLIERE	
SSP EMPLOYES					
Starting ADDRESS					
Nb Source	Nb Objet		Nb Absolu		
Divers					
Clés employées					

FICHE EXPLOITATION

SIGAL

SUBROUTINE SIGAL(P,UT,US,SIGMA,DALFA)

N=0

T=UT

S=US

3 CL=S/1.80655

SIGMZ=(((3.98E-05)*CL-(1.57E-03))*CL+1.4708)*CL-.069

AT=(((1.0843E-03)*T-.098185)*T+4.7867)*T*(1.E-03)

BT=(((0.01667*T-.8164)*T+18.03)*T*1.E-06

ST=-((T-3.98)*(T-3.98))*(T+283.)/(503.57*(T+67.26))

SIG=ST+(SIGMZ+.1324)*(1.-AT+BT*(SIGMZ-.1324))

ALPHZ=1./(1.+.001*SIG)

SU=(4886./(1.+1.83E-05*P))-((.004*T-.551)*T+28.33)*T+227.+((-1.58
1*T+9.5)*T+105.5)*P*1.E-04-T*P*P*1.5E-08-.1*(SIGMZ-28.)*(((.04*T
2-2.72)*T+147.3)-P*((.02*T-.87)*T+32.4)*1.E-04)+.01*(SIGMZ-28.)
3*(SIGMZ-28.)*(4.5-.1*T-P*(1.8-.06*T)*1.E-04)

SU=SU*1.E-09

ALPHA=ALPHZ*(1.-SU*P)

IF(N)1,1,2

1 N=N+1

D=ALPHA

SIGMA=SIG

T=0.

S=35.

GOTO 3

2 DALFA=D-ALPHA

RETURN

END

OXY

FUNCTION OXY(U,S,02)

T=U+273.16

A=EXP(-7.424+4417./T-2.927*ALOG(T)+.04238*T-(S/1.80655)*(-.1074+
153.44/T-.04442*ALOG(T)+7.145E-04*T))

PV1=1.-EXP(26.1205*(1.-T/373.16))

PV2=1.-EXP(8.03945*(1.-373.16/T))

PV=EXP(18.1973*(1.-373.16/T)+(3.1813E-07)*PV1-1.8726E-08*PV2+.1134
102*ALOG(373.16/T))*(1.-9.701E-04*(S/1.80655))

O200=.2094*A*(1.-PV)

OXY=(O2/O200)*100.

RETURN

END

SIGMZ

FUNCTION SIGMZ(S)

CL=S/1.80655

SIGMZ=(((3.98E-05)*CL-(1.57E-03))*CL+1.4708)*CL-.069

RETURN

END

SIGMA

FUNCTION SIGMA(T,S)

AT=(((1.0843E-03)*T-.098185)*T+4.7867)*T*(1.E-03)

BT=(((0.01667*T-.8164)*T+18.03)*T*1.E-06

ST=-((T-3.98)*(T-3.98))*(T+283.)/(503.57*(T+67.26))

SIGMA=ST+(SIGMZ(S)+.1324)*(1.-AT+BT*(SIGMZ(S)-.1324))

RETURN

END

ALPHA

FUNCTION ALPHA(T,P,S)

ALPHZ=1./(1.+.001*SIGMA(T,S))

SU=(4886./(1.+1.83E-05*P))-((.004*T-.551)*T+28.33)*T+227.+((-1.58
1*T+9.5)*T+105.5)*P*1.E-04-T*P*P*1.5E-08-.1*(SIGMZ(S)-28.)*(((.04*T
2-2.72)*T+147.3)-P*((.02*T-.87)*T+32.4)*1.E-04)+.01*(SIGMZ(S)-28.)
3*(SIGMZ(S)-28.)*(4.5-.1*T-P*(1.8-.06*T)*1.E-04)

SU=SU*1.E-09

ALPHA=ALPHZ*(1.-SU*P)

RETURN

FICHE TECHNIQUE DU PROGRAMME

NOM: SSP COR

TYPE	ACQUISITION	CHAINE	C	PERIPH.	IN	
BUT	Entree du coef. de correction dans SONDB				OUT	
				FORMAT	IN	
					OUT	
Observations				EIIQ.	IN	
Opérations					OUT	
DATE DE MISE AU POINT	AVRIL 1974			PROGRAMMEUR		
LANGAGE UTILISE	assembleur			MORLIERE		
SSP EMPLOYES						
Starting ADDRESS						
Nb Source	Nb Objet		Nb Absolu			
Divers						
Clés employées						

FICHE EXPLOITATION

```
NAM COR
EXT .ENTR, .IOC, FLOAT
ENT COR, LIBRE, CHAR
COM ASC 11, COEF. CORRECTION*1.E+5
C BSS 2
COR NOP
JSB .ENTR
DEF C
JSB .IOC.
OCT 0
JSB .IOC.
OCT 20002
JMP *-2
DEF COM
DEC 11
JSB .IOC.
OCT 40002
SSA
JMP *-3
LECT CLF 0
LDA =B160000
OTA 17B
JSB LIBRE
JSB FLOAT
FMP =F.00001
DST C, I
JMP COR, I
LIBRE NOP
AGAIN CLA
STA DATA
LIRE JSB CHAR
CPA =B377
JMP RUB
CPA =B254
JMP SEPAR
CPA =B240
JMP SEPAR
CPA =B215
JMP CR
AND =B17
STA CA
LDA DATA
MPY =D10
ADA CA
STA DATA
RUB JMP LIRE
JSB CHAR
JSB CHAR
JMP AGAIN
CR JSB CHAR
SEPAR LDA DATA
JMP LIBRE, I
CHAR NOP
STC 17B, C
SFS 17B
JMP *-1
LIA 17B
JMP CHAR, I
DATA BSS 1
CA BSS 1
END COR
```

NOM: *Sos*

FICHE TECHNIQUE DU PROGRAMME

TYPE		CHAINE		PERIPH.	IN	5
BUT	<i>Correction des rubans mal perforés par une TTY factieuse.</i>				OUT	4
Observations				FORMAT	IN	/
Opérations					OUT	/
				ETIQ.	IN	/
					OUT	/
DATE DE MISE AU POINT				PROGRAMMEUR		
LANGAGE UTILISE	<i>ASSEMBLER.</i>			<i>Muliere.</i>		
SSP EMPLOYES						
Starting ADDRESS						
Nb Source	Nb Objet		Nb Absolu			
Divers						
Clés employées						

FICHE EXPLOITATION

```
NAM SOS
ENT START
CNTR DEC 0
M25 DEC -75
START NOP
  CLF 0
  JSB READ
    SZA, RSS
    JMP *-2
  LDB M25
  STB CNTR
  JSB PUNCH
  JSB READ
  SZA
  JMP *-5
  ISZ CNTR
  JMP *-5
  HLT 77B
  JMP START+1
READ NOP
  STC 15B, C
  SFS 15B
  JMP *-1
  LIA 15B
  CPA =B0
  JMP READ, I
  CPA =B212
  JMP READ, I
  CPA =B215
  JMP READ, I
  AND =B137677
  IOR =B20040
  JMP READ, I
PUNCH NOP
  OTA 16B
  STC 16B, C
  SFS 16B
  JMP *-1
  JMP PUNCH, I
  END START
```

FICHE TECHNIQUE DU PROGRAMME

NOM: **STDC**

TYPE	<i>Acquisition SSP</i>	CHAINE	<i>A</i>
BUT	<i>Contrôle du hardware d'acquisition de données, décodage, tri et mise en forme des signaux bruts du 8200.</i>		
Observations			
Opérations			
DATE DE MISE AU POINT			
LANGAGE UTILISE	<i>ASSEMBLER</i>		
SSP EMPLOYES			
Starting ADDRESS			
Nb Source	Nb Objet		Nb Absolu
Divers			
Clés employées	<i>0 et 1 (0 entrée = acquisition - 1 = fin de l'acquisition)</i>		

PERIPH.	IN	<i>13</i>
	OUT	
FORMAT	IN	
	OUT	
ETIQ.	IN	
	OUT	
PROGRAMMEUR		
<i>Malière.</i>		

FICHE EXPLOITATION

HED LIAISON SONDE-ORDINATEUR :LECTURE 6200
 * PROGRAMME LECTURE DOUVVEES SONDE (SORTIE SUR 6200)
 * APPEL FORTRAN CALL STDC(JP, JT, JS, JO, JCLE)
 *

```

      NAM STDC
      EVT STDC, CONV
      EXT .EWTR
JP     ESS 1
JT     ESS 1
JS     ESS 1
JO     ESS 1
JCLE  ESS 1
STDC  NOP
      JSE .EWTR
      DEF JP
DEBUT LIA 01      TEST, SUR, CLEFS
      STA JCLE, I
      SZA, RSS      CLEFS DOWN
      JMP *-3
      JSE LECT
      SSA
      JSE LECT
      STA D1
      LDA 1
      STA W
      JSE CONV
      STA JS, I
      LDA D1
      STA W
      JSE CONV
      STA JT, I
      JSE LECT
      STA D1
      LDA 1
      STA W
      JSE CONV
      STA JP, I
      LDA D1
      AND #B7777
      STA W
      JSE CONV
      STA JO, I
      LDA JP, I
      CMA, IWA

```

```
      ADA =D1500      A=1500-P
      JMP DEBUT      B POSITIVE
      JMP STDC,i
      SPC 1
* LECTURE CARTE DS1 EN 13
      SPC 1
LECT  WOP
      STC 13B,C
      SFS 13B
      JMP *-1
      LIA 13B
      LIB 13B
      CMA
      CMB
      CLC 13B
      JMP LECT,i
      SPC 1
* SCUS PROGRAMME DE CONVERSION
* DONNEE EN A ET N, RESULTAT DANS A
      SPC 1
COJV  WOP
      AND =B17
      STA N1
      LDA N
      AND =E360
      ALF,ALF
      ALF
      MPY =D10
      ADA N1
      STA N1
      LDA N
      AND =B7400
      ALF,ALF
      MPY =D100
      ADA N1
      STA N1
      LDA N
      AND =B70000      ELIMINATION BIT 15
      ALF
      MPY =D1000
      ADA N1
      JMP COJV,i
      SPC 2
N1    ESS 1
N     ESS 1
D1    ESS 1
      END
      ENDS
```

NOM: TESTD

FICHE TECHNIQUE DU PROGRAMME

TYPE		CHAINE		PERIPH.	IN			
BUT	Contrôle la liaison (sonde) → 8200 — ORDINATEUR			FORMAT	OUT			
Observations				FORMAT	IN			
Opérations				FORMAT	OUT			
DATE DE MISE AU POINT	1999			ETIQ.	IN			
LANGAGE UTILISE	FORTRAN			ETIQ.	OUT			
SSP EMPLOYES	STDC			PROGRAMMEUR				
Starting ADDRESS				Moulière				
Nb Source	/	Nb Objet	/				Nb Absolu	/
Divers								
Clés employées	EXPLICITES: sur TTY							

FICHE EXPLOITATION

levé 1 clé pour transmission sur TTY les données affichées au 8200

```

PROGRAM TEST D
50 CALL STDC (JP, JT, JS, JO, JC)
WRITE (2, 100) JP, JT, JS, JO
100 FORMAT (4I5)
GO TO 50
END
END

```

FICHE TECHNIQUE DU PROGRAMME

NOM: MTU

TYPE	CHAINE			PERIPH.	IN	
BUT	permet de passer du repère MTU au repère géographique en φ et G				OUT	
				FORMAT	IN	
Observations					OUT	
Opérations				ETIQ.	IN	
					OUT	
DATE DE MISE AU POINT	Fevrier 1977			PROGRAMMEUR		
LANGAGE UTILISE	BASIC			CITEAU		
SSP EMPLOYES						
Starting ADDRESS						
Nb Source	Nb Objet		Nb Absolu			
Divers						
Clés employées						

FICHE EXPLOITATION

il faut donner les informations concernant le point pris pour repère . exemple le Phare de

Port Bouet } $x = 393.7925$ en MTU
 $y = 579.9236$

pris en φ et G $\varphi = 5^{\circ} 14' 8$
 $G = 3^{\circ} 59' 4$

Pendant ce la les autres couples x, y seront convertis en φ, G

```
1 PRINT " REPERE EN MTU(X,Y) "
2 INPUT X0,Y0
3 PRINT "PHI=(DEG,MIN,SEC) "
4 INPUT D2,M2,S2
5 PRINT "G=(DEG,MIN,SEC) "
6 INPUT D1,M1,S1
7 LET M1=SGN(D1)*(M1+60*AES(D1)+S1/60)
8 LET M2=SGN(D2)*(M2+60*AES(D2)+S2/60)
10 PRINT
15 PRINT "X=";
20 INPUT X
25 PRINT "Y=";
26 INPUT Y
30 LET R=(X-X0)/1.852+M1
32 LET R1=SGN(R)
34 LET R=AES(R)
96 LET D=INT(R/60)
97 LET M=INT(R-D*60)
98 LET D9=INT((R-D*60-M)*10)
99 IF R1<0 THEN 150
120 PRINT D;M;"";"D9;"E";
125 GOTO 153
150 PRINT D;M;"";"D9;"W";
153 PRINT "***";
155 LET R=(Y-Y0)/1.852+M2
156 LET R1=SGN(R)
157 LET R=AES(R)
160 LET D=INT(R/60)
165 LET M=INT(R-D*60)
170 LET D9=INT((R-D*60-M)*10)
175 IF R1<0 THEN 180
176 PRINT D;M;"";"D9;"N";
177 GOTO 185
180 PRINT D;M;"";"D9;"S";
185 GOTO 185
900 END
```

FICHE TECHNIQUE DU PROGRAMME

NOM:

POINT

TYPE		CHAINE		PERIPH.	IN	
BUT	Trace les trajectoires de flotteurs à chaque			FORMAT	IN	
Observations				ETIQ.	IN	
Opérations					OUT	
DATE DE MISE AU POINT	Fevrier 1977			PROGRAMMEUR		
LANGAGE UTILISE	BASIC			CITEAU		
SSP EMPLOYES						
Starting ADDRESS						
Nb Source	Nb Objet		Nb Absolu			
Divers						
Clés employées						

FICHE EXPLOITATION

- le balisage d'une zone marine (exemple le littoral de Vlissi à Port Bouet) est réalisé par des amers dont la position est connue dans un repère MTU
- Ces informations sont rentrées dans les instructions DATA
- chaque flotteur est repéré à un instant déterminé par deux angles. Exemple $\left\{ \begin{array}{l} \text{PHARE avec C} : 28^{\circ}53' \\ \text{E avec D} : 58^{\circ}50' \end{array} \right.$
- les deux arcs capables ainsi déterminés se coupent en C et sur la position du flotteur repéré.
- les deux solutions sont imprimées sur la teletype. La première est la solution cherchée. Cette position est pointée sur la table tracante.
- Si elle est hors-limite, la seconde solution est alors retenue.

```
1 DIM A(2,13),R(2,2),E(2),Z(2),V(2),W(2),G(2),P(2)
5 MAT READ A
6 PRINT "POINT PAR 2 ARCS. TRACE ?(OU=1)"
7 INPUT R4
8 IF R4#1 THEN 13
9 PRINT "X MIN,MAX,Y MIN,MAX?"
10 INPUT X1,X2,Y1,Y2
13 PRINT "ENTRER R1, AVEC R2, ANGLE (DEG, MIN)"
17 PRINT
18 INPUT N1,N2,D,M
20 GCSUE 300
22 MAT C=Z
24 LET R1=R9
28 INPUT N1,N2,D,M
30 GCSUE 300
32 MAT P=Z
34 LET R2=R9
36 LET T=(P(2)-G(2))/(P(1)-G(1))
38 LET T1=ATN(T)
40 MAT E=O
42 GCSUE 400
44 MAT C=Z
46 MAT E=P
48 GCSUE 400
50 MAT P=Z
52 LET D1=P(1)-G(1)
54 LET C=(R2*R2-D1*D1-R1*R1)/(-2*R1*D1)
56 LET E(1)=R1*C+G(1)
58 LET E(2)=G(2)-R1*SQR(1-C*C)
59 LET E3=G(2)+R1*SQR(1-C*C)
60 LET T1=-T1
62 GCSUE 400
63 LET Z1=Z(1)
65 LET Z2=Z(2)
67 LET E(2)=E3
70 GCSUE 400
72 LET Z3=Z(1)
74 LET Z4=Z(2)
76 PRINT "X1='Z 1; 'Y 1='Z 2; ** X2='Z 3; 'Y 2='Z 4
100 IF R4#1 THEN 17
102 LET X=INT((Z1-X1)*9999/(X2-X1))
104 LET Y=INT((Z2-Y1)*9999/(Y2-Y1))
106 PRINT "PLTP"
107 PRINT X;Y
108 PRINT X;Y
109 PRINT "PLTT"
110 IF (X*(X-9999))>0 THEN 115
111 IF (Y*(Y-9999))>0 THEN 115
112 GOTO 18
115 IF Z1=Z3 THEN 18
1116 LET Z1=Z3
1177 LET Z2=Z4
```

```
118 PRINT "SQ.2"
119 GOTO 102
300 LET T=(A[2,N2]-A[2,N1])/(A[1,N2]-A[1,N1])
302 LET T1=ATN(T)
305 LET E[1]=A[1,N1]
308 LET E[2]=A[2,N1]
310 GOSUB 400
312 MAT V=Z
314 LET E[1]=A[1,N2]
316 LET E[2]=A[2,N2]
318 GOSUB 400
320 MAT W=Z
322 LET D=(D+M/60)*3.14159/180
324 LET E[1]=(V[1]+W[1])/2
326 LET R9=ABS(W[1]-V[1])/(2*SIN(D))
328 LET E[2]=V[2]-R9*COS(D)
330 LET T1=-T1
332 GOSUB 400
338 RETURN
400 LET R[1,1]=COS(T1)
401 LET R[1,2]=SIN(T1)
403 LET R[2,1]=-SIN(T1)
404 LET R[2,2]=COS(T1)
406 MAT Z=R*B
410 RETURN
500 DATA 385.932, 391.752, 394.778, 396.592, 400.799, 388.875
501 DATA 388.783, 387.882, 388.215, 392.481, 393.792, 396.327
502 DATA 389.165, 579.285, 580.411, 579.455, 579.136, 578.492
503 DATA 579.45, 579.815, 580.193, 580.566, 581.038, 579.924
504 DATA 580.108, 581.285
999 END
```


G - 136

FICHE TECHNIQUE DU PROGRAMME

NOM:

AABM / SRBM

TYPE		CHAINF		PERIPH.	IN	
BUT	Mise sur Bande Magnétique de rubans perforés ssus de courantomètres AANDERRA			FORMAT	OUT	
Observations				ETIQ.	IN	
Opérations					OUT	
DATE DE MISE AU POINT	Juin 1977			PROGRAMMEUR		
LANGAGE UTILISE				MORLIERE		
SSP EMPLOYES						
Starting ADDRESS						
Nb Source	Nb L: jet		Nb Absol..			
Divers						
Clés employées						

FICHE EXPLOITATION

Messages explicites - respecter les formats d'écriture demandés.

<p>But : TRANSFERT DE DONNEES SONDE REDUITES (ruban), sur Bande magnetique</p> <p>Utilisation : Messages explicites ,respecter les formats demandés</p>	<p>NOM :</p> <p>SRBM</p> <p>Programmeur :</p> <p>MORLIERE</p>
---	---

```

0001      CRG 2000E
0002      JMP DEB
0003 MSG   ASC 26,**** MISE SUR BANDE 600 EPI DES RUEANS AANDERAA ****
0004 MSG1  ASC 18,FIN FAIRE CLES#0 SINON FAIRE CLES#0
0005 MSG2  ASC 5,FAIRE RUN
0006 MSG3  ASC 11,STATUS CONTROLER 7970
0007 MSG4  ASC 3,FIN EM
0008 MSG5  ASC 20,INSCRIPTION TERMINEE NE ENREGISTREMENTS
0009 MSG6  ASC 23,INSCRIPTION A SUIVRE FAIRE CLES#0 SINON CLES#0
0010 MSG7  ASC 11,FILE (<100) FORMAT XX
0011 MSG8  ASC 14,NUMERO ENREGISTREMENT XXXXXX
0012 A0    DEF MSG
0013 A1    DEF MSG1
0014 A2    DEF MSG2
0015 A3    DEF MSG3
0016 A4    DEF MSG4
0017 A5    DEF MSG5
0018 A6    DEF MSG6
0019 A7    DEF MSG7
0020 A8    DEF MSG8
0021 LG    DEC 52
0022 LG1   DEC 36
0023 LG2   DEC 10
0024 LG3   DEC 22
0025 LG4   DEC 6
0026 LG5   DEC 40
0027 LG6   DEC 46
0028 LG7   DEC 22
0029 LG8   DEC 2
0030 LG8A  DEC 26
0031 LG9   DEC 6
0032 ANF   DEF NF
0033 NF    ESS 1
0034 E17   CCT 17
0035 C10   ESS 1
0036 ACR   DEF CR
0037 CR    CCT 12
0038 N6    ESS 1
0039 M6    ESS 1
0040 AELCC DEF ELCC
0041 ELCC  ESS 390
0042 M390  DEC -390
0043 C390  ESS 1
0044 M30   DEC -30
0045 C30   ESS 1
0046 M26   DEC -13
0047 C26   ESS 1
0048 D760  DEC 760
0049 ADATA DEF DATA
0050 IATA  ESS 13
0051 E377  CCT 377
0052 NER   ESS 1
0053 D4    DEC 4
0054 AN1   DEF N1
0055 N1M   ESS 1
0056 ANUM  DEF NUM
0057 LIGNE NOP
0058      CLA
0059      INA

```

```

0060      LDE ACR
0061      JSE 102E,i
0062      JMP LIGNE,i
0063 DEE   CLA
0064      STA NBR
0065      JSE LIGNE
0066      JSE LIGNE
0067      LDA LG
0068      LDE A0
0069      JSE 102E,i
0070      JSE STATU
0071      JSE 107E,i
0072      CCT 3
0073      JSE LIGNE
0074      LDA LG7
0075      LDE A7
0076      JSE 102E,i
0077      LDA LG8
0078      LDE ANF
0079      JSE 104E,i
0080      LDA NF
0081      SZ A, RSS
0082      JMP DEPAR
0083      LDA LIX
0084      CMA, INA
0085      STA C10
0086      LDA NF
0087      ALF, ALF
0088      AND E17
0089      STA 1
0090      LDA NF
0091      AND E17
0092 H    ADA 1
0093      ISZ C10
0094      JMP H
0095      CLE
0096      JSE 107E,i
0097      CCT 4
0098      JMP ECT
0099      NCP
0100 DEPX  LDA M390
0101      STA C390
0102      JSE LIGNE
0103      LDA LG8A
0104      LDE A8
0105      JSE 102E,i
0106      LDA LG9
0107      LDE ANUM
0108      JSE 104E,i
0109      LDE AELCC
0110      CLA
0111 W    STA 1,i
0112      INE
0113      ISZ C390
0114      JMP W
0115      LDA NUM
0116      STA ELCC
0117      JSE CITEM
0118 DEPAF LDA M390
0119      STA C390

```

```

0120      LDE AELCC
0121      CLA
0122  ZERC  STA 1,i
0123      INE
0124      ISZ C390
0125      JMP ZERC
0126      LDA M30
0127      STA C30
0128      LDA AELCC
0129      STA W8
0130  LFCT  LDA ADATA
0131      STA M6
0132      LDA M26
0133      STA C26
0134      JSE ANDEF
0135      ISZ WER
0136      JSE ASCII
0137  D     LDA M6,i
0138      STA W6,i
0139      LDA M6
0140      INA
0141      STA M6
0142      LDA W8
0143      INA
0144      STA W6
0145      ISZ C26
0146      JMP D
0147      LDA AA6
0148      SZA
0149      JMP ECR
0150      ISZ C30
0151      JMP LECT
0152      JSE CUTEM
0153      JMP DEPAR
0154  CUTEM NCP
0155      LDA D780
0156      LDE ALLCC
0157      JSE 107E,i
0158      CCT 1
0159      JMP ECT
0160      NCP
0161      JMP CUTEM,i
0162  ECR   JSE CUTEM
0163      JSE LIGNE
0164      LDA LG1
0165      LDE A1
0166      JSE 102E,i
0167      LDA LG2
0168      LDE A2
0169      JSE 102E,i
0170      HLT 55E
0171      LDA 01
0172      SZA
0173      JMP DEPAR
0174      JSE 107E,i
0175      CCT 2
0176      JMP ECT
0177      NCP
0178      LDA LG5
0179      LDE A5

```

```

0180      JSE 102E,i
0181      LDA WER
0182      JSE BINAS
0183      LDA L4
0184      LDE AW1
0185      JSE 102E,i
0186      JSE LIGNE
0187      LDA LG6
0188      LDE A6
0189      JSE 102E,i
0190      HLT
0191      LIA 01
0192      SZA
0193      JMP NEW
0194      JSE 107E,i
0195      CCT 5
0196      HLT 77E
0197      JMP DEE
0198  NEW   CLA
0199      STA WER
0200      JMP DEPX
0201  STATU WCP
0202      JSE 107E,i
0203      CCT 7
0204      SZE,RSS
0205      JMP STATU,i
0206      ELF,ELF
0207      SSE,RSS
0208      JMP STATU,i
0209      LDA LG3
0210      LDE A3
0211      JSE 102E,i
0212      HLT 44E
0213      JMP STATU,i
0214  ECT   LDA LG4
0215      LDE A4
0216      JSE 102E,i
0217      JSE 107E,i
0218      CCT 5
0219      HLT 77E
0220  iAA   DEF AA
0221  AA    ESS 6
0222  AA7   ESS 1
0223  AA8   ESS 1
0224  ANI ER WCP
0225      CLA
0226      STA AA7
0227      STA AA8
0228      LDA M15
0229      STA CNTEL
0230      LDA M6
0231      STA CNTR
0232  ELANC JSE CARAC
0233      LIA X
0234      SZA
0235      JMP WQWEL
0236      iSZ CNTEL
0237      JMP ELANC
0238      LIA FRRGR
0239      STA AA8

```

```

0240          JMP FIN
0241  NGWEL   LDA M15
0242          STA CNTEL
0243          LDA ADD
0244          STA MCT
0245          LDA IAA
0246          STA W
0247          JMP MCT+2
0248  AGAIN   JSE MCT
0249  DEFUT   LDA PARA
0250          LDE W
0251          STA 1,i
0252          INE
0253          STE W
0254          ISZ CNTR
0255          JMP SYN
0256          LDA SYNC
0257          SZA
0258          JMP FIN
0259          LDA ERRCP
0260          ADA AA7
0261          STA AA7
0262          JMP FIN
0263  SYN     LDA SYNC
0264          SZA
0265  FIN     JMP ANDER,i
0266          JMP AGAIN
0267  MCT     WCP
0268          JSE CARAC
0269          LDA CA6
0270          STA SYNC
0271          LDA CA7
0272          SZA
0273          JMP SUI T1
0274          LDA ERRCP
0275          ADA AA7
0276          STA AA7
0277          JSE CARAC
0278          JMP MCT,i
0279  SUI T1  LDA X
0280          STA PAPA
0281          JSE CARAC
0282          LDA CA6
0283          SZA,FSS
0284          JMP SUI T3
0285          LDA ERRCP
0286          ADA AA7
0287          STA AA7
0288  SUI T3  LDA CA7
0289          SZA,FSS
0290          JMP SUI T2
0291          LDA ERRCP
0292          ADA AA7
0293          STA AA7
0294          JMP MCT,i
0295  SUI T2  LDA X
0296          ALF,ALS
0297          AIA PARA
0298          STA PAPA
0299          JMP MCT,i

```

```

0300 CAPAC WCP
0301 CLF 0
0302 STC 15E,C
0303 SFS 15E
0304 JMP *-1
0305 LIE 15E
0306 LDA 1
0307 AND E37
0308 STA X
0309 LDA 1
0310 AND E40
0311 STA CA6
0312 LDA 1
0313 ANI E100
0314 STA CA7
0315 CLC 15E
0316 JMP CAPAC,i
0317 ADI DEF DEEUT
0318 X ESS 1
0319 CA6 ESS 1
0320 CA7 ESS 1
0321 PARA ESS 1
0322 CNTR ESS 1
0323 EFBOP CCT 1
0324 M6 DEC -6
0325 M15 DEC -45
0326 CNTEL ESS 1
0327 E37 CCT 37
0328 E40 CCT 40
0329 E100 CCT 100
0330 N FSS 1
0331 SYNC ESS 1
0332 ASCII WCP
0333 LDA M6
0334 STA CW
0335 LIA ADATA
0336 STA M
0337 LDA IAA
0338 STA N
0339 E LDA N,i
0340 JSE EINAS
0341 LIA N1
0342 STA M,i
0343 LIE M
0344 INE
0345 LIA N2
0346 STA 1,i
0347 INE
0348 STE M
0349 LDE N
0350 INE
0351 STE N
0352 ISZ CW
0353 JMP E
0354 LDA AA7
0355 ADA E60
0356 AND E377
0357 ALF,ALF
0358 LIE 0
0359 LDA AA8

```

```

0360      ADA E60
0361      ANL E377
0362      ADA 1
0363      STA M,i
0364      JMP ASCi,i
0365 M     ESS 1
0366 CN    ESS 1
0367 *BINAIRE-ASCII DONNEE EN A<10000
0368 *RESULTATS N1,N2
0369 FINAS NCP
0370      LDE MILLE
0371      JSE DIVIS
0372      STA N1
0373      LDA 1
0374      LDE CENT
0375      JSE DIVIS
0376      STA N2
0377      LDA 1
0378      LDE DIX
0379      JSE DIVIS
0380      STA N3
0381      STE N4
0382      LDA N1
0383      ADA E60
0384      ALF,ALF
0385      LDE N2
0386      ADE E60
0387      ADA 1
0388      STA N1
0389      LDA N3
0390      ADA E60
0391      ALF,ALF
0392      LDE N4
0393      ADE E60
0394      ADA 1
0395      STA N2
0396      JMP FINAS,i
0397 N1    ESS 1
0398 N2    ESS 1
0399 N3    ESS 1
0400 N4    ESS 1
0401 DIX   DEC 10
0402 CENT  DEC 100
0403 MILLE DEC 1000
0404 E60   CCT 60
0405 *DIVISION ENTIERS EN A PAR E
0406 *RESULTATS EN A DEXTE EN E
0407 DIVIS NCP
0408      STE Y
0409      CLE
0410      STE N5
0411      LIB Y
0412      CME,INE
0413 C     ISZ N5
0414      ADA 1
0415      SSA,FSS
0416      JMP C
0417      ADA Y
0418      STA 1
0419      LDA M1
0420      ADA N5
0421      JMP DIVIS,i
0422 M1    DEC -1
0423 Y     ESS 1
0424 N5    ESS 1
0425      END

```



```

0001      CRG 2000E
0002      JMP DEPAR
0003 M1    ASC 26,**** MISE SUR BANIE 800 EPI DES STATIONS SCNIE ****
0004 M2    ASC 4,FILE XX
0005 M3    ASC 23,FIN CAMPAGNE FAIRE CLES=1  SINON FAIRE CLES=0
0006 M4    ASC 10,PLACER RUEAN PERFCEP
0007 M5    ASC 10,STATUS CCNTRCLEP LM
0008 M6    ASC 4,FIN EM**
0009 A1    DEF M1
0010 A2    DEF M2
0011 A3    DEF M3
0012 A4    DEF M4
0013 A5    DEF M5
0014 A6    DEF M6
0015 L1    DEC 52
0016 L2    DEC 8
0017 L3    DEC 46
0018 L4    DEC 20
0019 L5    DEC 2
0020 LM    DEC 70
0021 LD    DEC 75
0022 METEC ESS 38
0023 DATA ESS 36
0024 AM    DEF METEC
0025 AD    DEF DATA
0026 AFILE DEF FILE
0027 FILE  ESS 1
0028 C10   ESS 1
0029 E17   CCT 17
0030 ACR   DEF CR
0031 CR    CCT 12
0032 A99   ASC 1,99
0033 ZERC  CCT 7417
0034 E100  CCT 100
0035 D33   DEC 33
0036 E34   DEC 34
0037 C2    ESS 1
0038 LIGNE NCP
0039 C36   ESS 1
0040 M36   DEC -36
0041 A00   ASC 1,00
0042      CLA
0043      INA
0044      LDE ACR
0045      JSE 102F,i
0046      JMP LIGNE,i
0047 ECT   LDA L2
0048      LDE A6
0049      JSE 102E,i
0050      JSE 107L,i
0051      CCT 5
0052      HLT 77E
0053 DEPAR JSB LIGNE
0054      JSE LIGNE
0055      LDA M36
0056      STA C36
0057      LIA AM
0058      LIE A00
0059 ZZ   STE 0,i

```

```

0060      INA
0061      ISZ C38
0062      JMP ZZ
0063      LDA L1
0064      LDE A1
0065      JSB 102E,i
0066      E   JSE 107E,i
0067      CCT 7
0068      SZE,FSS
0069      JMP A
0070      CPE E100
0071      JMP A
0072      LDA L4
0073      LDE A5
0074      JSE 102E,i
0075      HLT 44E
0076      JMP E
0077      A   JSE 107E,i
0078      CCT 3
0079      JSE LIGNE
0080      LDA L2
0081      LDE A2
0082      JSE 102E,i
0083      LDA LG
0084      LDE AFILE
0085      JSE 104E,i
0086      LDA FILE
0087      SZA,FSS
0088      JMP LECT
0089      LDA DIX
0090      CMA,INA
0091      STA C10
0092      LDA FILE
0093      ALF,ALF
0094      AND 17
0095      STA 1
0096      LDA FILE
0097      AND E17
0098      ii  ALA 1
0099      ISZ C10
0100      JMP H
0101      CLE
0102      JSE 107E,i
0103      CCT 4
0104      JMP ECT
0105      NOP
0106      LECT LDA LM
0107      LDE AM
0108      JSE 101E,i
0109      LIA MFTEC
0110      CPA A99
0111      JMP FINC
0112      LIA 01
0113      SSA
0114      JMP E
0115      LIA LM
0116      LDE AM
0117      JSE 102E,i
0118      B   LDA LD
0119      LDE AM

```

```

0120      JSE 107E,i
0121      CCT 1
0122      JMP ECT
0123      NCP
0124  SUIT  LDA LG
0125      CMA,INA
0126      STA C2
0127  I     LIA LD
0128      LDE AD
0129      JSE 101E,i
0130      LIA LD
0131      LDE AD
0132      JSE 107E,i
0133      CCT 1
0134      JMP ECT
0135      NCP
0136      ISZ C2
0137      JMP D
0138      LDE AD
0139      ADE I33
0140      LDA 1,i
0141      AND ZERC
0142      SZA
0143      JMP SUIT
0144      LDE AD
0145      ADE D34
0146      LDA 1,i
0147      AND ZERC
0148      SZA
0149      JMP SUIT
0150      JMP LECT
0151  FINC  LDA L3
0152      LDE A3
0153      JSE 102E,i
0154      HLT
0155      LIA 01
0156      SZA,FSS
0157      JMP SUIV
0158      JSE 107L,i
0159      CCT 2
0160      JMP ECT
0161      NCP
0162      JSE 107E,i
0163      CCT 5
0164      HLT 77E
0165      JMP DEPAR
0166  SUIV  LDA L4
0167      LDE A4
0168      JSE 102E,i
0169      HLT 55E
0170      JMP LECT
0171  FIX   DEC 10
0172      END

```

**FNL-CF-TAPE

*

/F

*END