NOTICE BIBLIOGRAPHIQUE

Auteur(s)	Compagnie des machines Bull
Titre	Gamma 3, calculateur électronique
Adresse	Paris : Compagnie des machines Bull, [1952]
Collation	1 vol. (95 p - 41 pl) : ill.; 28 cm
Nombre de vues	284
Cote	CNAM-MUSEE CM0.4-BUL
Sujet(s)	Ordinateurs Calculatrices Cartes perforées, Systèmes de
Thématique(s)	Catalogues de constructeurs Technologies de l'information et de la communication
Typologie	Ouvrage
Note	Fonds Bécu
Langue	Français
Date de mise en ligne	21/10/2020
Date de génération du PDF	16/01/2023
Permalien	http://cnum.cnam.fr/redir?M14926

F.D.BECU.

GAMMA 3

■

CALCULATEUR ÉLECTRONIQUE

COMPAGNIE DES MACHINES BULL
École d'Application
94, avenue Gambetta — PARIS 20e

N° 1657

Prix : 2.900 Francs

Sommaire

Chapitre premier

GÉNÉRALITÉS

I. FONCTION DU GAMMA 3

Le Gamma 3 est un calculateur électronique connectable à une machine à cartes perforées, Tabulatrice ou Reproductrice. Cette machine transmet au Calculateur les données du calcul et en reçoit les résultats qu'elle imprime, perfore ou cumule.

Un tableau de connexion amovible de 720 plots, s'adaptant à l'intérieur de l'armoire, permet de déterminer pour chaque travail les opérations confiées au programme électronique du Calculateur. Ce programme se superpose, en le complétant, au programme de la machine connectée qui reste entièrement disponible pour les fonctions ne faisant pas intervenir le Calculateur.

Le Calculateur connecté à une Reproductrice, remplit toutes les fonctions d'une Calculatrice classique, avec cependant les avantages apportés par les deux pistes, les 5 brosses et la perforation de la Reproductrice, ainsi que l'extrême rapidité du calcul électronique. Connecté à une Tabulatrice, il constitue une large extension de la Tabulatrice à Multiplication, permettant de supprimer, dans de nombreux cas, le calcul préalable des cartes par une Calculatrice, sans ralentir pour cela les opérations de la Tabulatrice.

II. LA NUMÉRATION EN GAMMA 3

Dans la numération décimale, les nombres sont décrits à l'aide de 10 figures d'écriture que l'œil est habitué à reconnaître. Lorsque l'on désire faire reconnaître à une machine ces 10 figures, il est nécessaire d'établir un mécanisme ou son équivalent, capable de se positionner sur 10 espaces différenciés que l'on explore ensuite systématiquement. C'est ainsi que les totalisateurs BULL possèdent 10 barrettes horizontales, chacune affectée à l'un des chiffres de la numération décimale.

Il existe un système de numération qui ne requiert pas ces organes. C'est le système binaire où il n'existe que deux figures d'écriture des nombres. Ces deux signes peuvent être

matérialisés sur un conducteur électrique par la présence d'une tension positive ou négative.

L'avantage de la numération binaire est donc évident lorsqu'il s'agit d'une machine. Son inconvénient tient en particulier au fait que ce n'est pas le système habituel d'expression des nombres. C'est pourquoi le Gamma 3 allie les avantages des deux systèmes : il utilise la numération décimale codée en binaire, où chacun des chiffres a pour code son expression binaire.

III. ÉCRITURE DES NOMBRES EN DÉCIMAL ET EN BINAIRE

Tout système d'écriture de nombres utilise, outre les signes d'écriture appelés chiffres, des positions dans lesquelles on place les chiffres. Ces positions déterminent le rang des unités constituant le nombre, alors que le chiffre indique le nombre de fois qu'il les contient. En numération décimale, les divers rangs d'unités ont reçu les noms d'unités, dizaines, centaines, etc...

Ainsi, le nombre 2.384 a la structure suivante :

Milliers	Centaines	Dizaines	Unités
pris	prises	prises	prises
deux fois	trois fois	huit fois	quatre fois
2	3	8	4

et il serait lu, si la langue était en parfaite correspondance avec cette structure :

— Deux milliers,
— Trois centaines,
— Huit dizaines et
— Quatre unités.

Les ordres d'unités sont des puissances de la base de numération. Dans le système décimal :

le 1er ordre, unités, est égal à $10^0 = 1$,
le 2e ordre, dizaines, est égal à $10^1 = 10$,
le 3e ordre, centaines, est égal à $10^2 = 100$,
le 4e ordre, milliers, est égal à $10^3 = 1.000$, etc.

Dans le système binaire, on distingue également des ordres d'unités qui sont les puissances de base 2 :

le 1er ordre, égal à 2^0, est appelé « poids 1 »,
le 2e ordre, égal à 2^1, est appelé « poids 2 »,
le 3e ordre, égal à 2^2, est appelé « poids 4 »,
le 4e ordre, égal à 2^3, est appelé « poids 8 ».

Pour fixer les idées (*fig.* 1), on peut traduire le nombre 228 en décimal et en binaire.

IV. ÉCRITURE DES NOMBRES EN DÉCIMAL CODE BINAIRE

La base d'écriture est 10. C'est chaque chiffre de 0 à 10 qui est écrit en binaire dans chaque ordre décimal.

Les caractéristiques de cette écriture sont :

a) de conserver aux nombres leur articulation décimale,
b) de limiter les nécessités du transcodage décimal-binaire et vice-versa aux 9 chiffres usuels.

Chaque chiffre sera dès lors figuré par un groupe indivisible de quatre impulsions, présentes ou absentes, selon la valeur du chiffre représenté. Le tableau suivant (*fig.* 2) indique la codification en binaire des dix figures d'écriture en système décimal.

Décimal Base 10		
10^2	10^1	10^0
100	10	1
2	2	8

Binaire Base 2							
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	64	32	16	8	4	2	1
1	1	1	0	0	1	0	0

Reconstitution

8 × 1	= 8	128
2 × 10	= 20	64
		32
2 × 100	= 200	4
	228	228

Fig. 1

	2^3	2^2	2^1	2^0
0				
1				X
2			X	
3			X	X
4		X		
5		X		X
6		X	X	
7		X	X	X
8	X			
9	X			X

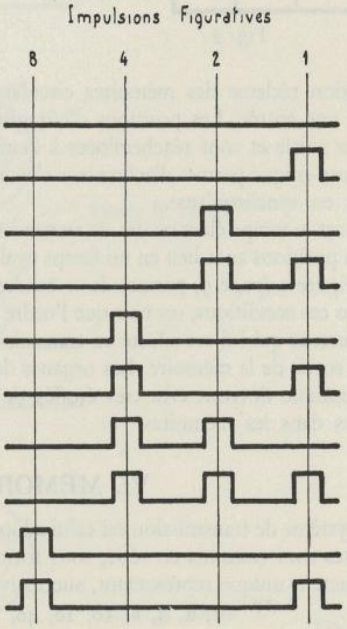


Fig. 2

V. TRANSMISSION PARALLÈLE - TRANSMISSION SÉRIE

Quel que soit le mode de représentation adopté pour un nombre, on dispose de deux possibilités de le transmettre d'un organe à l'autre d'une machine.

1. Transmission parallèle.

Chaque position d'écriture est transmise par un conducteur distinct (*fig.* 3). C'est le système adopté sur les machines électro-mécaniques de la gamme BULL.

Les transmissions de tous les chiffres du nombre transmis se font simultanément le long de chacun des conducteurs. Si t est le temps élémentaire de transmission de l'une des positions de numération, la transmission de toutes les positions aura donc lieu au cours du temps t.

2. Transmission série.

L'ensemble des positions d'écriture est transmis par un seul conducteur (*fig.* 4). Cette

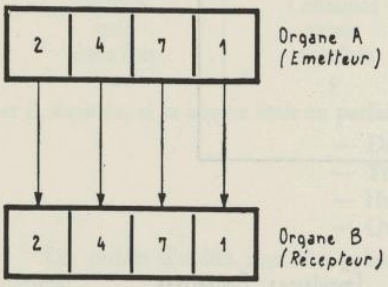


Fig. 3

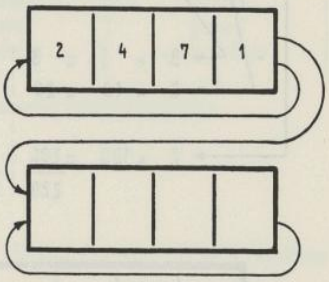


Fig. 4

organisation réclame des mémoires circulantes, bouclées sur elles-mêmes, présentant une sortie et une entrée. Les positions d'enregistrement des nombres se présentent successivement à la sortie et sont réacheminées à l'entrée de la mémoire par la boucle, tandis qu'un conducteur unique permet d'acheminer chacune des positions à l'entrée d'une autre mémoire tournant en synchronisme.

Si t est le temps élémentaire de transmission d'une position de numération, la transmission de n positions aura lieu en un temps égal à $n\,t$.

La figure 5 (page 9) permet de se rendre compte des transmissions au cours du temps.

Dans ces conditions, on voit que l'ordre d'unité transmis sera défini non pas par le rang du conducteur qui lui est affecté en transmission parallèle, mais par le temps de son apparition à la sortie de la mémoire. Les organes définiteurs du temps dans une machine adoptant cette technique devront être très étoffés de façon à pouvoir repérer les diverses valeurs contenues dans les mémoires.

VI. MÉMOIRES GAMMA

Ce système de transmission est celui adopté dans le Calculateur Gamma 3 où les nombres représentés sont transmis en série, sous forme d'une suite d'impulsions, se propageant sur un conducteur unique représentant, successivement, les valeurs :

$$1,\ 2,\ 4,\ 8,\ 10,\ 20,\ 40,\ 80,\ 100,\ 200,\ 400,\ 800\ldots$$
$$10^{11},\ 2 \times 10^{11},\ 4 \times 10^{11},\ 8 \times 10^{11}.$$

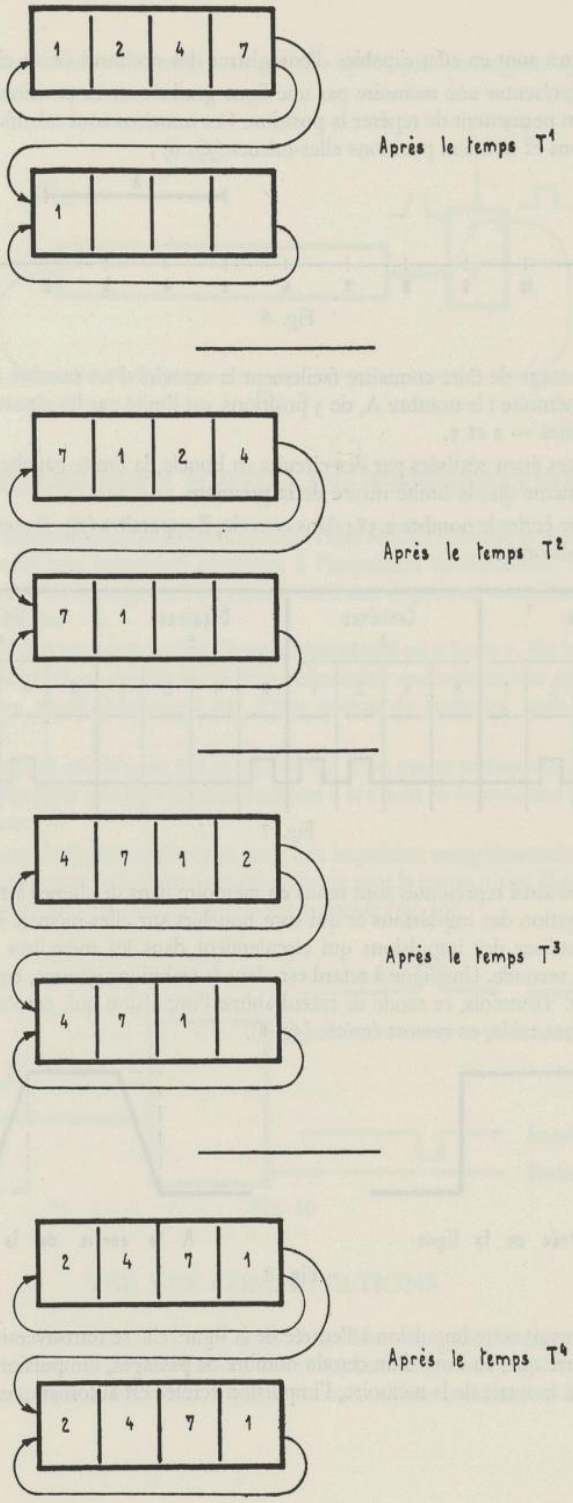


Fig. 5

Ces mémoires sont en effet capables d'enregistrer des nombres de 12 chiffres.

On peut représenter une mémoire par une ligne graduée en 12 positions. Des numéros sur la graduation permettent de repérer la position. Ces numéros sont relatifs aux séparations entre les positions et non aux positions elles-mêmes (*fig.* 6) :

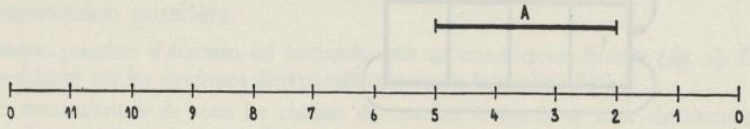


Fig. 6

Ceci a l'avantage de faire connaître facilement la capacité d'un nombre en énonçant ses limites dans la mémoire : le nombre A, de 3 positions, est limité par les séparations — encore appelées intervalles — 2 et 5.

Les mémoires étant réalisées par des circuits en boucle, la limite gauche de la douzième position est la même que la limite droite de la première.

Si l'on désire écrire le nombre 2.384 dans ce code, il apparaîtra (*fig.* 7) comme la séquence suivante d'impulsions :

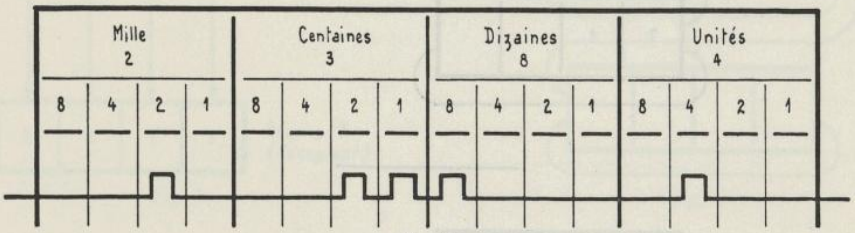


Fig. 7

Les nombres ainsi représentés sont tenus en mémoire dans des lignes à retard qui ralentissent la propagation des impulsions et qui sont bouclées sur elles-mêmes. En effet, il n'est pas possible d'utiliser des impulsions qui circuleraient dans les mémoires à la vitesse de 300.000 km. à la seconde. Une ligne à retard est, dans la technique gamma, constituée par des selfs et capacités. Toutefois, ce mode de retard abîme l'impulsion qui, entrée fraîche dans la ligne avec un front raide, en ressort écrêtée (*fig.* 8).

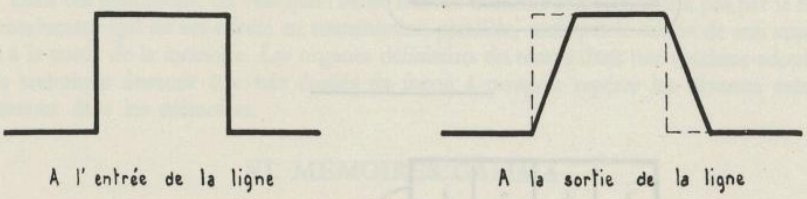


Fig. 8

Si l'on renvoyait cette impulsion à l'entrée de la ligne, elle se retrouverait encore écrêtée à la sortie, de sorte que, au bout d'un certain nombre de passages, l'impulsion serait annulée. C'est pourquoi, à la sortie de la mémoire, l'impulsion écrêtée est automatiquement remplacée

par une impulsion fraîche. Ce remplacement est opéré par un organe appelé régénérateur (*fig.* 9).

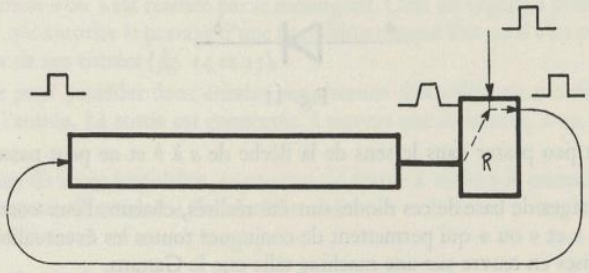


Fig. 9

VII. LES DEUX SORTIES DU RÉGÉNÉRATEUR

Les régénérateurs sont dotés de deux sorties : l'une est dite normale et fournit, en échange de l'impulsion usée une impulsion identique à l'impulsion déclencheuse. L'autre est dite « barre » et fournit en échange de l'impulsion usée une impulsion complémentaire de l'impulsion déclencheuse.

Il importe de préciser cette notion de complémentarité ou « barre ». En logique moderne (notation de Bourbaky), on appelle barre d'un phénomène quelconque tout ce qui n'est pas ce phénomène. Il ne s'agit absolument pas d'une notion de contraire, mais simplement de complémentarité.

Si un phénomène est désigné par *a*, tout ce qui n'est pas ce phénomène, (phénomène *b*, *c*. *d*... *x*,) sera désigné par $\bar{a}$ ce qui s'énonce « a barre » et s'écrit en surmontant la lettre désignatrice par une barre.

La sortie barre du régénérateur émet donc une impulsion complémentaire de l'impulsion de déclenchement sortant de la ligne : elle est présente tout le temps où est absente l'impulsion de déclenchement et absente lorsque celle-ci est présente (*fig.* 10).

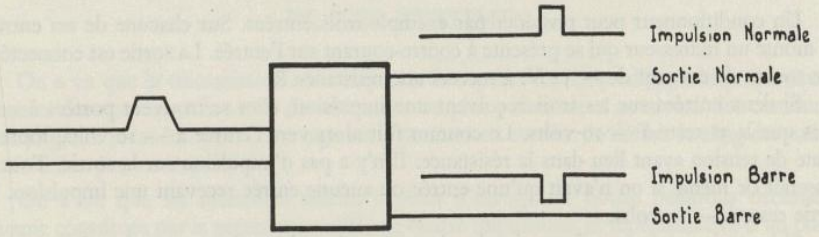


Fig. 10

VIII. LES COMMUTATIONS

Les diverses transmissions d'organe de mémoire à autre organe de mémoire sont contrôlées par des commutations réalisées par des diodes à cristal de germanium. Ces diodes sont des redresseurs, analogues aux détecteurs à galène ou aux cellules redresseuses de cuivre oxyde.

Elles présentent dans un sens une résistance faible au passage et dans l'autre sens une résistance très forte. On les représente sous la forme suivante (*fig.* 11).

Fig. 11

Le courant peu passer dans le sens de la flèche de *a* à *b* et ne peut passer dans le sens inverse, de *b* à *a*.

Deux montages de base de ces diodes ont été réalisés, chacun d'eux correspondant aux conditions « et » et « ou » qui permettent de conjuguer toutes les éventualités logiques qui peuvent être mises en œuvre sur une machine telle que le Gamma.

1. Conditionneurs.

La conjonction « et » est réalisée par le conditionneur. C'est un organe à plusieurs entrées et une seule sortie qui n'autorise le passage d'une impulsion que lorsque toutes ses entrées reçoivent simultanément une impulsion (*fig.* 12 et 13).

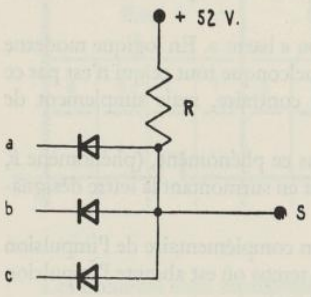


Fig. 12

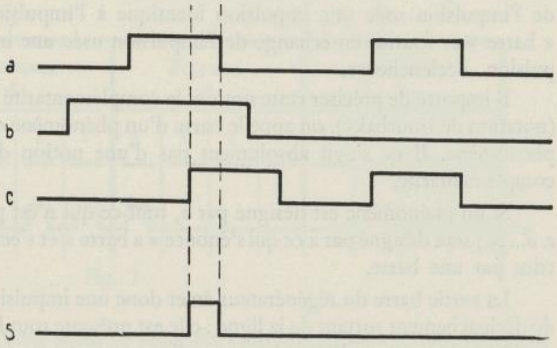


Fig. 13

Un conditionneur peut posséder par exemple trois entrées. Sur chacune de ses entrées est monté un redresseur qui se présente à contre-courant sur l'entrée. La sortie est connectée à une source de courant de + 52 V. à travers une résistance R.

Si deux entrées sur les trois reçoivent une impulsion, elles se trouvent portées à 0 V. alors que la 3e reste à — 10 volts. Le courant fuit alors vers l'entrée à — 10 volts, toute la chute de tension ayant lieu dans la résistance. Il n'y a pas d'impulsion sur la sortie. Tout se passerait de même si on n'avait qu'une entrée ou aucune entrée recevant une impulsion. La sortie reste à — 10 volts.

Si les trois entrées reçoivent simultanément une impulsion, le courant fuit alors vers les entrées qui se trouvent à 0 volt. La sortie acquiert également ce potentiel : il y a alors une impulsion à la sortie.

Un tel montage ne laisse passer une impulsion que si les trois entrées reçoivent simultanément une impulsion. Ce qui arrive sur *a*, *b*, et *c* peut être un signal continu ou discontinu ou une suite d'impulsions. La durée de l'impulsion de sortie est égale au temps pendant lequel les trois entrées reçoivent simultanément une impulsion.

Le résultat est identique à celui que l'on obtient en montant trois contacts en série.

2. Mélangeurs.

La conjonction « ou » est réalisée par le mélangeur. C'est un organe à plusieurs entrées et une seule sortie qui autorise le passage d'une impulsion chaque fois qu'il s'en présente une sur une quelconque de ses entrées (*fig.* 14 et 15).

Cet organe peut posséder deux entrées sur chacune desquelles est monté un redresseur passant depuis l'entrée. La sortie est connectée, à travers une résistance, à un potentiel négatif (— 56 V).

En l'absence de toute impulsion, le courant se ferme à travers R entre les entrées et le — 56 volts. La chute de tension a lieu dans la résistance et S est au potentiel de *a* et *b*, soit — 10 volts.

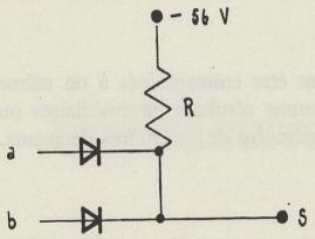


Fig. 14

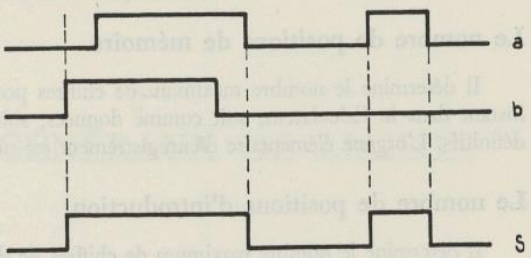


Fig. 15

Si *a* reçoit une impulsion, ce qui le porte à 0 volt, le courant se ferme entre *a* et le — 56 V. S est alors au potentiel de *a* soit 0 volt. Le redresseur de *b* est pris à contre-courant, ce qui laisse *b* à — 10 volts (effet anti-retour). Il existe alors une impulsion sur la sortie. Si les deux entrées sont à 0 volt, S est également à 0 volt.

Le mélangeur mélange les impulsions qu'il reçoit sur ses entrées. La durée de l'impulsion de sortie est égale au temps pendant lequel une impulsion se présente à l'une quelconque des entrées. Ce résultat est analogue à celui obtenu en montant plusieurs contacts en parallèle.

IX. LES RYTHMES

On a vu que la transmission série imposait de repérer les temps d'apparition des divers éléments des nombres à la sortie des mémoires ou plus précisément au niveau des commutations joignant les mémoires. Ces temps seuls permettent en effet de définir le rang des unités représentées par les impulsions.

On a vu que les mémoires étaient capbles d'enregistrer douze positions décimales, chacune constituée par la présence possible de quatre impulsions, et représentative d'un poids binaire. Le temps qui s'écoule pour que passe une de ces impulsions au niveau des commutations est appelé Θ. Chaque position décimale passera donc en 4 Θ. Ce temps est appelé τ. Un nombre complet de 12 positions défilera en 12 τ correspondant à un tour de mémoire. Ce temps n'a pas reçu de nom.

Pour repérer ces différents temps, le Gamma est doté de bases de temps électroniques, appelées « rythmeurs » qui définissent les principaux temps et rangs suivants :

a) Le temps élémentaire Θ et le rang dans chaque τ des 4 poids binaires.

b) Le temps élémentaire τ et le rang dans chaque tour des 12 positions décimales. Ce rythmeur est appelé rythmeur décimal.

c) Le temps de circulation complet d'une mémoire, égal à 170 micro-secondes.

Ces rythmeurs peuvent être comparés, dans le raisonnement de compréhension, à des horloges, et nous les représenterons dans la suite sous la forme d'un cadran.

X. CARACTÉRISTIQUES COMMERCIALES

Le Calculateur Gamma 3 existe en plusieurs modèles se caractérisant par :

Le nombre de positions de mémoire.

Il détermine le nombre maximum de chiffres pouvant être emmagasinés à un même instant dans le Calculateur, soit comme données, soit comme résultats intermédiaires ou définitifs. L'organe élémentaire d'enregistrement est une mémoire de 12 chiffres décimaux.

Le nombre de positions d'introduction.

Il détermine le nombre maximum de chiffres de données pouvant être enregistré dans le Calculateur en introduction cinématique au cours d'un cycle de machine connectée. L'organe élémentaire d'introduction est un introducteur de 12 chiffres décimaux.

Le nombre de positions d'extraction.

Il détermine le nombre maximum de chiffres de résultats pouvant être extraits au cours d'un cycle de machine connectée. L'organe élémentaire d'extraction est un extracteur de 12 chiffres décimaux.

Le nombre de lignes de programme.

Il détermine le nombre d'opérations élémentaires pouvant être affichées au tableau du Calculateur.

Ces quatre caractéristiques sont résumées par la formule MIEL. Le Calculateur standard est un 432 — 32, ce qui signifie :

M (nombre de mémoires)	4 (48 positions).
I (nombre d'introducteurs)	3 (36 —).
E (nombre d'extracteurs)	2 (24 —).
L (nombre de lignes de prog.)	32

Le nombre de lignes de programme peut atteindre 48 ou 64. Le nombre de mémoires peut être porté, unité par unité, jusqu'à 7. Le nombre d'introducteurs peut être porté, unité par unité, jusqu'à 6. Le nombre d'extracteurs peut être porté, unité par unité, jusqu'à 4. Le Gamma 3 peut donc atteindre la capacité 764 — 64.

Chapitre deuxième

LES CIRCUITS GÉNÉRAUX DE LIAISON

I. ROLES RESPECTIFS DE LA MACHINE CONNECTÉE ET DU CALCULATEUR GAMMA 3

Le Calculateur Gamma 3 est un calculateur autonome qui reçoit ses données de calcul depuis une machine à cartes perforées qui lui est connectée, élabore à partir de ces données les résultats que l'on cherche et transmet ces résultats à la machine connectée qui les exploite.

Les organes émetteurs de la machine connectée sont dans ce but capables d'être mis en relation avec les organes récepteurs du Calculateur, appelés introducteurs.

Les organes récepteurs de la machine connectée sont capables d'être mis en relation avec les organes émetteurs du Calculateur, appelés extracteurs.

La synchronisation entre les deux machines est réalisée à partir des organes définiteurs du temps dans la machine connectée (rupteur et cames) mis en relation, du fait de la connexion au Gamma, avec des organes du Gamma 3 observateurs de l'évolution du travail de la machine connectée (unifieur et mémoire code point).

Un schéma de ces relations (voir pl. 4) est donné en annexe.

II. LES TEMPS DE TRAVAIL DES DEUX MACHINES

Trois stades sont donc à distinguer dans le déroulement d'un calcul effectué sur une machine connectée au Gamma 3 :

a) Le stade d'introduction des données fournies par la machine connectée. Dans bien des cas, les données sont fournies au Calculateur au cours même des points d'exploration cinématique caractéristique des machines électro-mécaniques.

Ce mode d'introduction, cinématique, réclame que le Calculateur soit capable, à chaque point du cycle de la machine connectée, d'interpréter les émissions du rupteur qui lui parviennent.

b) Lorsque les données ont été introduites, le Calculateur peut commencer ses calculs, soit à la fin du cycle de la machine connectée. Il les effectue à une vitesse suffisante pour que ceux-ci soient terminés avant le point 9 du cycle suivant.

c) Au cours de ce cycle peuvent être extraits les résultats de ces calculs. Pour la machine connectée, cette extraction a lieu nécessairement pendant les points du cycle correspondant aux valeurs à extraire.

D'où le diagramme (*fig.* 16) concernant les trois stades de confection d'un résultat à partir des données :

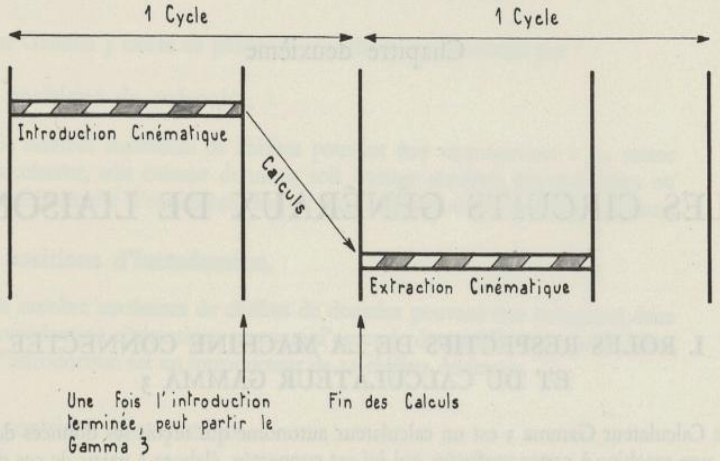


Fig. 16

Comme on choisit nécessairement, dans une telle organisation, des positions de mémoires distinctes pour recevoir d'une part les données de calculs et d'autre part leurs résultats, il s'ensuit que lors de l'introduction des données d'un calcul dans les mémoires d'introduction, les résultats du calcul précédent sont disponibles dans les mémoires d'extraction.

On peut être conduit à effectuer des calculs à tous les cycles, en superposant à chaque cycle introduction et extraction, ou un cycle sur deux en distinguant à chaque cycle introduction et extraction (*fig.* 17).

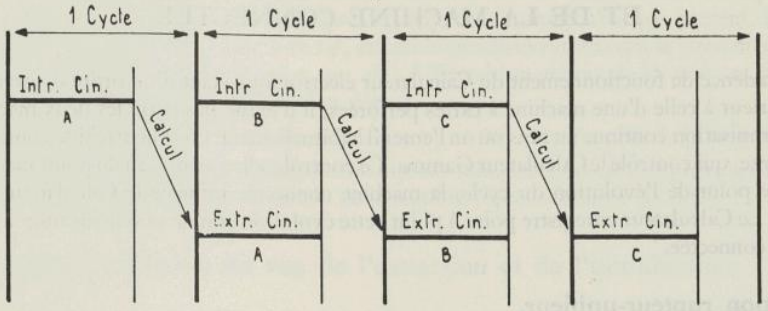
De toutes manières, on ne pourra jamais obtenir les résultats au même cycle que l'introduction des données. Ce point est important pour la présentation des états en tabulatrice. Un chapitre spécial lui est consacré.

Au sujet de la longueur des calculs, on a intérêt à se ménager, depuis leur début jusqu'à la fin du cycle, le maximum de durée. La fin du calcul ne peut pas être reculée au-delà de 15, 14 sur la tabulatrice et de 18, 14 sur la P. R. D.

Le début des calculs est fonction de la durée de l'introduction et de l'extraction. Le Calculateur peut être réglé, par connexion sur son tableau propre, de façon à ce que la machine connectée lui indique à quel point du cycle il doit commencer son calcul. Ce réglage est constitué par l'opération VPM (variante point machine) qui sera étudiée lors des opérations élémentaires du Calculateur.

Il n'existe pas de contrôle de départ de la machine connectée après le calcul Gamma :

Dans le premier cas :



Dans le deuxième cas :

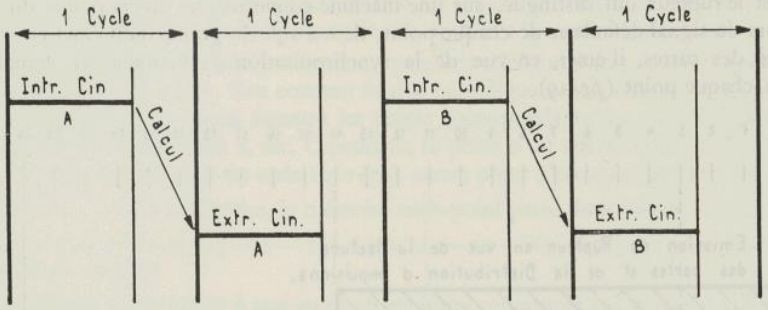


Fig. 17

cette machine tourne continuellement et la longueur du calcul doit être évaluée afin de déterminer s'il sera achevé avant le point 9 dans tous les cas.

Le diagramme (*fig.* 18) montre le rôle de la VPM :

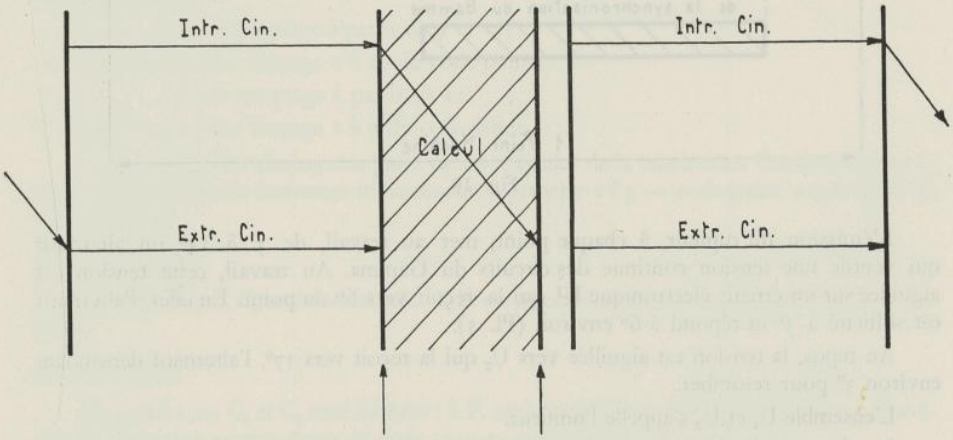


Fig. 18

III. LA SYNCHRONISATION DU CALCULATEUR ET DE LA MACHINE CONNECTÉE

La cadence de fonctionnement du Calculateur électronique étant d'un ordre de grandeur très supérieur à celle d'une machine à cartes perforées, il n'existe pas entre les deux machines de synchronisation continue au sens où on l'entend habituellement. C'est la machine connectée, la plus lente, qui contrôle le Calculateur Gamma. Ce contrôle a lieu au niveau du point machine. A chaque point de l'évolution du cycle, la machine connectée informe le Calculateur de sa position. Le Calculateur enregistre point à point cette évolution afin de se synchroniser avec la machine connectée.

1. Relation rupteur-unifieur.

C'est le rupteur qui distingue, sur une machine connectée, les divers points du cycle. En dehors du signal définiteur de chaque point, de 0 à 14° du point, qu'il émet en vue de la lecture des cartes, il émet, en vue de la synchronisation du Gamma, un signal de 3 à 14° de chaque point (*fig.* 19).

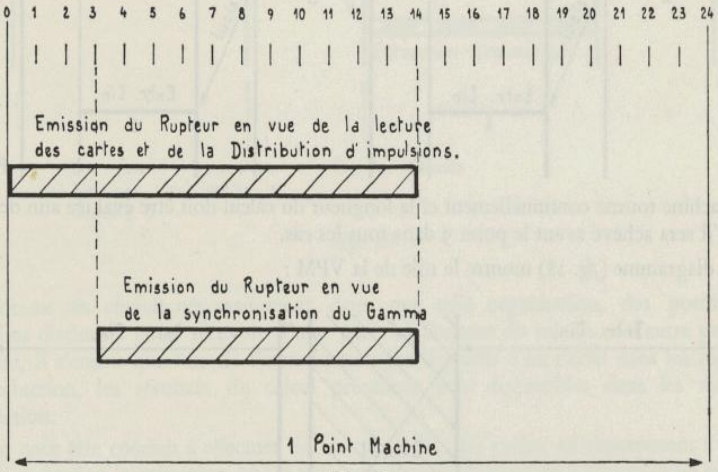


Fig. 19

L'émission du rupteur, à chaque point, met au travail, de 3 à 14° un alternatif qui ventile une tension continue des circuits du Gamma. Au travail, cette tension est aiguillée sur un circuit électronique U^1 qui la reçoit vers 6° du point. En effet, l'alternatif est sollicité à 3° et répond à 6° environ (Pl. 5).

Au repos, la tension est aiguillée vers U_2 qui la reçoit vers 17°, l'alternatif demandant environ 3° pour retomber.

L'ensemble U_1 et U_2 s'appelle l'unifieur.

Les deux parties U_1 et U_2 reçoivent tous les 96 τ une tension brève en provenance des rythmes du Gamma.

La tension Gamma aiguillée par l'alternatif valide la première impulsion de rythme se présentant après son établissement.

A partir de 6^0, arrive sur U_1 la tension Gamma. Si à ce moment arrive l'impulsion de rythme sur U_1, l'impulsion est émise par U_1 à 6^0. Si le rythme n'est pas présent, la tension Gamma étant permanente entre 6 et 14^0, attend le prochain rythme qui se présentera au plus tard 1 ms, 36 après 6^0, soit $7^0 5$ ($96\ \tau = 1^0 5$). L'impulsion unique de U_1 arrivera donc entre 6^0 et $7^0 5$. On l'appelle I. D. (impulsion début).

De même pour U_2 où elle apparaîtra entre 17 et $18^0 5$. On l'appelle I. F. (impulsion fin).

On pourra se reporter au « diagramme rupteur-unifieur », donné en annexe, pour étudier ce qui se passe dans le temps au niveau de la relation rupteur-unifieur (Pl. 6).

2. La synchronisation en vue de l'extraction et de l'introduction.

Le signal I. F. de l'unifieur est conduit sur une mémoire spéciale du Calculateur appelée mémoire « code-point ». Son rôle fondamental est d'informer en permanence le Calculateur de la position, c'est-à-dire du point du cycle, où se trouve la machine connectée.

Elle est constituée par une boucle de 1 τ permettant d'enregistrer quatre impulsions soit les codes binaires de 0 à 15. Son contenu évolue périodiquement à chaque cycle, en passant successivement par les codes figurant les points correspondants du cycle : code 1001 au point 9, code 1000 au point 8, etc. Cependant, le point 0 est affecté non pas du code 0000 (absence d'impulsion), mais du code 1010 (10), aucun point n'étant désigné par 10.

Dans le cas de la tabulatrice, la mémoire code-point passe donc par les valeurs :

9 — 8 — 7 — 6 — 5 — 4 — 3 — 2 — 1 — 10 — 11 — 12 — 13 — 14 — 15; — 9 — 8 — etc., exprimées en binaire.

Le passage d'une valeur à une autre est toujours commandé par le signal I. F. de l'unifieur vers 17^0 de chaque point.

Il est réalisé par un compteur-décompteur incorporé dans la mémoire Code-point.

Il est facile de voir que l'évolution du code-point présente au cours d'un cycle de tabulatrice 5 phases :

a) de 9 à 1, il y a décomptage à partir de 9 ;

b) au point 0, il y a « forçage » à 10 du code-point;

c) de 10 à 15, il y a comptage à partir de 10;

d) au point 9, il y a « forçage » à 9 du code-point.

Ces 4 phases sont distinguées par 2 cames spéciales de la tabulatrice. Ces deux cames C_3 et C_4 ont les temps de fermeture indiqués sur la planche « l'3 — code-point » qui figure en annexe (Pl. 7).

A. 1re *PHASE*

De 15,20 à 2,20, C_3 seule est fermée : I. F. est interprétée en vue du décomptage du code-point.

B. 2e *PHASE*

De 2,20 à 1,20. C_3 et C_3 sont fermées : I. F. est interprétée en vue du forçage à 10, c'est-à-dire de l'injection systématique du code 1010 au code-point.

C. 3e *PHASE*

De 1,20 à 14.20, C_4 seule est fermée : I. F. est interprétée en vue du comptage du code-point.

D. 4e *PHASE*

De 14,20 à 15,20, C_3 et C_4 sont ouvertes : I. F. est interprétée en vue du forçage à 9, c'est-à-dire de l'injection systématique du code 1001 au code-point.

Il est très important de remarquer que le code-point est en avance sur les points machine connectée. En effet, c'est au moment de I. F., c'est-à-dire de 17 à $18^0 5$ du point que le code-point évolue pour contenir le code du point machine qui apparaîtra sur la machine connectée à 24^0. De toute façon, il est visible que le code-point contient le code du point que définit le rupteur. Le code du point en cours est présent de 0 à 14^0 pendant l'émission du rupteur. Cette avance du code-point est utilisée lors de l'extraction, comme on le verra plus loin.

Cette organisation permet de comprendre le principe de l'introduction et le principe de l'extraction.

L'introduction cinématique consiste à envoyer, dans une mémoire du Calculateur le code binaire du chiffre lu sur la machine connectée. Ce chiffre apparaît sous la forme d'une impulsion du rupteur et sa valeur est définie par le temps de son apparition. Il suffit de valider le transfert du code-point, qui est la traduction binaire de la valeur du rupteur, dans la mémoire où l'on introduit. Ce transfert est déclenché par l'impulsion du rupteur arrivant sur l'introducteur (voir Pl. 8).

Le diagramme de la figure 20 précise la planche 8 : il s'agit de l'introduction d'un 7 lu sur une carte.

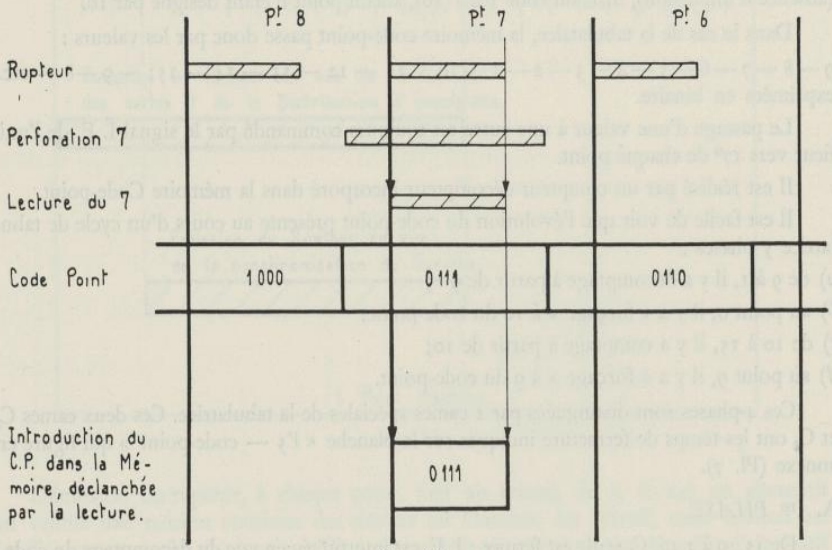


Fig. 20

L'extraction est le symétrique de l'introduction. Elle consiste, un code étant dans une mémoire, à le traduire en impulsion du rupteur au point déterminé qui définit la valeur de cette impulsion.

Il faut donc détecter le code à extraire au moment où le rupteur passe par le point correspondant. Si l'on compare la mémoire à extraire au code-point qui est au courant de l'évolution du rupteur, l'égalité entre la position à extraire et le code-point, déclenchera la validation du rupteur.

La figure 21 illustre l'extraction d'un 7 :

Les deux fonctions d'introduction cinématique et d'extraction seront étudiées en détail plus loin.

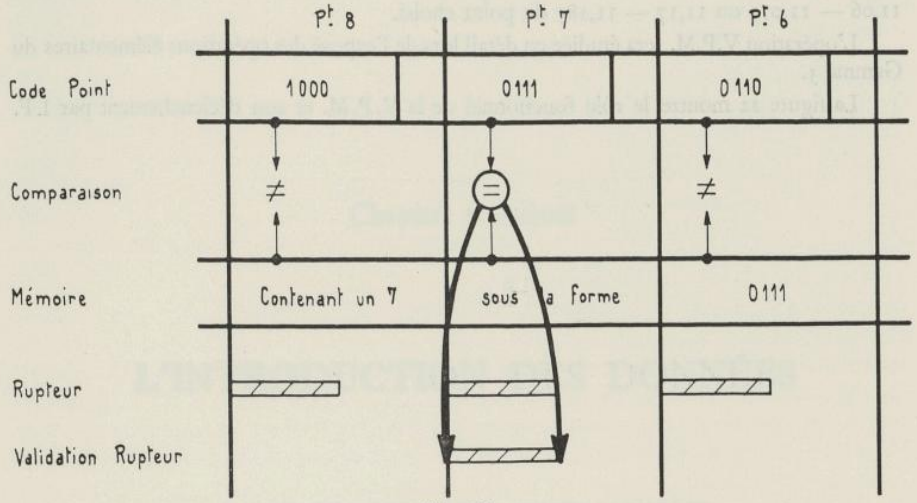


Fig. 21

3. La synchronisation en vue du départ des calculs.

La V.P.M. apparaît comme un réglage du Calculateur en vue du démarrage des calculs à un point déterminé du cycle. Ce réglage se fait sur le tableau de connexions du Gamma. Le déclenchement par la machine connectée est assuré par les impulsions de sortie de l'unifieur.

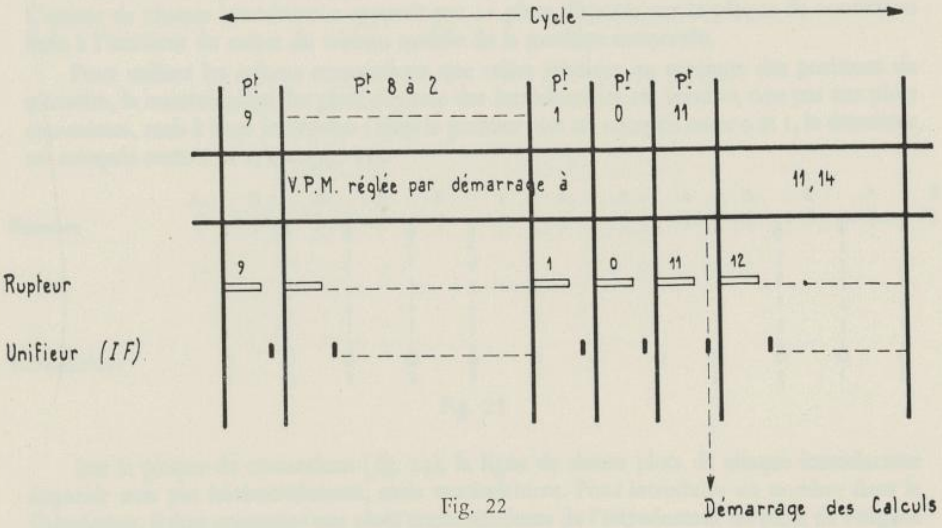


Fig. 22

Comme on a deux impulsions par point, I. D. au début, I. F. à la fin, il est possible de régler le Calculateur pour que le démarrage soit possible au début du point ou à la fin du point que l'on a choisi, par exemple à 11,3 ou à 11,14. En fait, ces appellations 11,3 ou 11,4 ne tiennent pas compte des retards dus au temps de réponse des circuits. Ces démarrages ont lieu entre 11,06 — 11,075 ou 11,17 — 11,185 du point choisi.

L'opération V.P.M. sera étudiée en détail lors de l'exposé des opérations élémentaires du Gamma 3.

La figure 22 montre le rôle fonctionnel de la V. P. M. et son déclenchement par I. F.

Chapitre troisième

L'INTRODUCTION DES DONNÉES

I. L'INTRODUCTION CINÉMATIQUE

1. Les introducteurs.

On désigne ainsi l'enregistrement, dans les mémoires du Calculateur, de données émises point par point par la machine connectée. Ces données peuvent provenir de la lecture d'une carte, de l'exploration d'un totalisateur ou d'une émission de constantes par le distributeur.

L'introduction dans les 12 positions d'une mémoire est assurée par un introducteur. L'entrée de chaque introducteur apparaît par 12 plots disposés sur la plaque de connexion fixée à l'intérieur du carter du tableau mobile de la machine connectée.

Pour utiliser les mêmes conventions que celles relatives au repérage des positions de mémoire, la numérotation des plots d'entrée des introducteurs est relative, non pas aux plots eux-mêmes, mais à leurs intervalles : ainsi le premier plot est compris entre 0 et 1, le deuxième est compris entre 1 et 2, etc. (*fig.* 23).

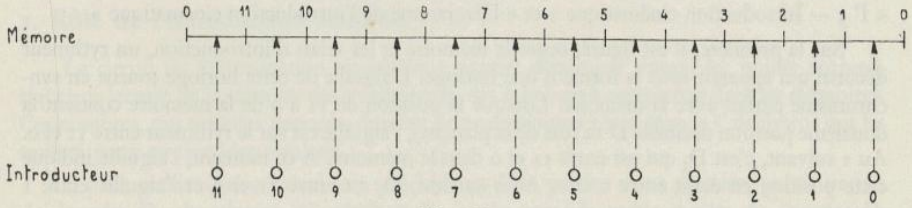


Fig. 23

Sur la plaque de connexions (*fig.* 24), la ligne de douze plots de chaque introducteur apparaît non pas horizontalement, mais verticalement. Pour introduire un nombre dans le Calculateur, il faut connecter aux plots correspondants de l'introducteur les plots de l'organe

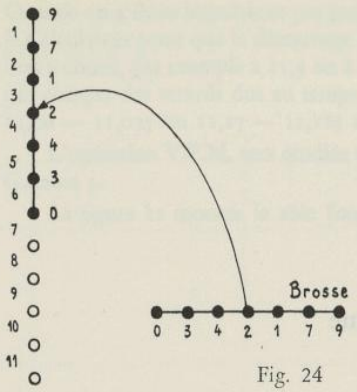


Fig. 24

émetteur : brosse, sortie de totalisateur, distributeur, en disposant les ordres décimaux en croissant, de haut en bas.

Il existe sur la machine standard 3 introducteurs liés aux mémoires 2, 3 et 4. Dans les machines à 7 mémoires, 3 introducteurs supplémentaires sont liés aux mémoires 5, 6 et 7. La mémoire 1 peut également recevoir des données en cinématique : l'introducteur de la mémoire 3 permet d'introduire, de la façon qui sera étudiée plus loin, soit dans la mémoire 3, soit dans la mémoire 1, soit dans les deux simultanément. Dans ce dernier cas, les données sont enregistrées en Y dans les deux mémoires.

2. Circuits de l'introduction.

Le principe de l'introduction a été étudié lors de la synchronisation du Gamma 3 et de la machine connectée. Elle consiste à transférer le contenu de la mémoire code-point dans les positions de mémoire du Calculateur, ce transfert étant déclenché à partir d'une impulsion de rupteur apparaissant sur les plots de l'introducteur. Cette impulsion a été préalablement filtrée soit par la carte, soit par les cames du distributeur, soit par les cames du distributeur et les roues des totalisateurs.

Pour cela, chaque plot d'entrée aboutit à un petit relais (voir Pl. Γ 3 — Introduction cinématique), qui, lorsqu'il est excité, autorise le transfert du code-point dans la boucle de mémoire. Les impulsions reçues au point 9 provoquent l'enregistrement du code 1001 et ainsi de suite jusqu'au point 0 où une impulsion provoque l'enregistrement d'un code 1010.

Cependant, au niveau de l'introducteur a lieu le passage de la transmission parallèle propre à la machine connectée, à la transmission série propre au Calculateur. Dans les mémoires, les nombres doivent présenter successivement leurs positions décimales à l'entrée de la mémoire où n'entrera qu'une seule position décimale en 1 τ. Dans ces conditions, choisir la position de mémoire revient donc à choisir l'instant du transfert au moment où la position désirée défile devant l'entrée de la mémoire. C'est pourquoi les 12 contacts de relais relatifs aux 12 positions d'un introducteur sont explorés successivement par les rythmes décimaux, en synchronisme avec le défilement des nombres dans la boucle de mémoire.

Pour illustrer ce qui se passe, il est nécessaire de se reporter à la fois aux deux planches « Γ 3 — Introduction cinématique » et « Diagramme de l'introduction cinématique ».

Sur la première, il est figuré, outre la mémoire et les relais d'introduction, un rythmeur décimal qui apparaît sous la forme d'une horloge. L'aiguille de cette horloge tourne en synchronisme parfait avec la mémoire. Lorsque la position de 11 à 0 de la mémoire contient la douzième position décimale D 12 (cas de la planche), l'aiguille est sur le rythmeur entre 11 et 0. Au τ suivant, c'est D_1 qui est entre 11 et 0 dans la mémoire. A ce moment, l'aiguille indique cette position en étant entre 0 et 1. Au τ suivant, D_2 est entre 11 et 0 et l'aiguille entre 1 et 2, etc.

A chacune de ces positions, l'aiguille, qui agit comme le commun d'un sélecteur rotatif, fait communiquer le contact de sortie du relais correspondant à la position décimale qui se trouve à l'entrée de la boucle avec le commutateur de validation du transfert du code-point. Si ce relais est excité du fait de l'arrivée d'une impulsion, le code-point est transféré dans la

position correspondant au relais excité. C'est ainsi qu'est établie la correspondance entre le rang du plot d'entrée et la position de mémoire recevant le transfert du code point.

Il est facile de voir dans ces conditions qu'à chaque point de la machine connectée, les 12 positions de la mémoire sont « balayées » par le rythmeur décimal (Pl. 9).

3. Les commandes d'introduction.

En plus de la validation du transfert code-point-mémoire qui vient d'être étudiée, existe une autre validation d'entrée, en série avec la première : chaque introducteur est en effet doté de deux plots strappés de commande qui apparaissent à la plaque de connexion, au droit des positions d'entrée. Tant que ces plots ne sont pas alimentés, les impulsions pouvant parvenir aux 12 positions d'entrée sont sans effet.

Les commandes des introducteurs ne doivent jamais être faites par des commandes de cycle ou sélection. On utilise, pour introduire de 9 à 1 et de 9 à 0 les plots du connecteur Gamma intitulés 9 à 0 et 9 à 11. Dans le cas d'un introducteur 1_7 à 1_5 connecté à une zone de brosse ou à un totalisateur, la validation pourra être faite par une commande de cycle de 9 à 1 ou 9 à 11. Dans ce cas les impulsions arrivant sur les entrées (depuis les brosses ou les totalisateurs...) doivent être incluses entièrement dans la durée de la commande, ou lui être totalement extérieures.

Les commandes du connecteur Gamma 9 à 1 — 9 à 0 — 9 à 11 — *ps.*, — 15. etc..., ne doivent jamais être mises en parallèle avec des commandes de la machine connectée. De même, ces commandes ne doivent jamais servir à alimenter des organes de la machine connectée.

Les commandes d'introduction agissent immédiatement et ne sont pas maintenues. Elles permettent de conditionner aisément un enregistrement, par un code de carte, par exemple. Cependant, leur action porte systématiquement sur les 12 positions. Au cas où l'on voudrait faire sur un introducteur, une introduction cinématique sélectionnée, il y aurait lieu d'utiliser pour cette sélection des alternatifs, conformément à la technique habituelle.

L'introducteur de la mémoire 3 présente la particularité d'avoir deux plots de commande I_3 et I_1, permettant d'introduire sur l'une ou l'autre ou les deux mémoires 3 et 1.

Les commandes d'introduction, dans les mémoires 1 et 2 n'autorisent que l'enregistrement des codes de 9 à 1, même s'ils sont alimentés en dehors de ces points. Les raisons de cette particularité seront indiquées plus loin lors de l'étude de la codification des zéros.

II. L'INTRODUCTION STATIQUE

1. Rôle de l'introduction statique.

Le circuit de l'introduction cinématique permet d'envoyer durant les points arithmétiques de lecture de la carte ou des totalisateurs, des nombres à enregistrer dans les mémoires. Ces nombres, qui sont des données, doivent pouvoir trouver place dans le Calculateur qui les traitera pour en tirer des résultats.

Il peut se faire qu'un calcul soit scindable en deux parties : une première partie partira de données D_1 aboutissant aux résultats R_1 ; une seconde partie partira de données D_2 conjuguées ou non aux résultats R_1, pour aboutir aux résultats R_2. Pendant la deuxième partie du calcul, on n'aura plus besoin de D_1. S'il faut introduire toutes les données D_1 et D_2 au début du calcul, il faudra prévoir an Calculateur à grande capacité d'enregistrement.

C'est pour réduire cette capacité d'enregistrement qu'a été conçue l'introduction statique.

Elle consiste en une possibilité d'introduire, au cours des calculs, des données pour la suite des calculs. Ces données pourront prendre la place qu'occupaient les précédentes données introduites avant tous calculs, en cinématique, et qui n'ont plus d'utilité.

La figure 25 montre comment est organisé ce procédé.

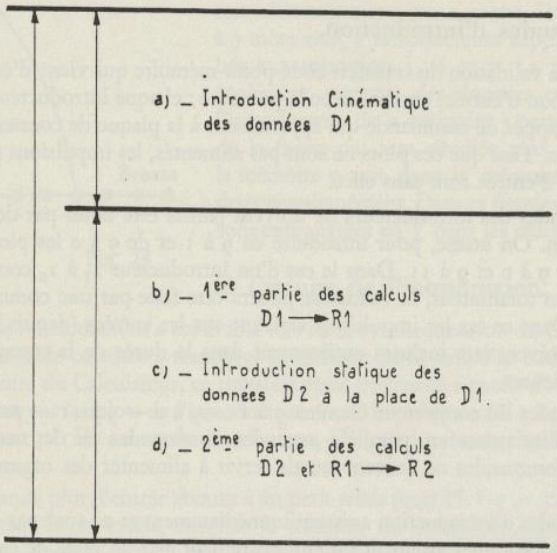


Fig. 25

Bien entendu, si les données D_1 et D_2 proviennent de la même carte, il faudra au cours du cycle d'enregistrement cinématique de D_1 stocker provisoirement dans un totalisateur les données D_2, totalisateur qu'on devra pouvoir lire à la fin de la première partie des calculs (*fig.* 25).

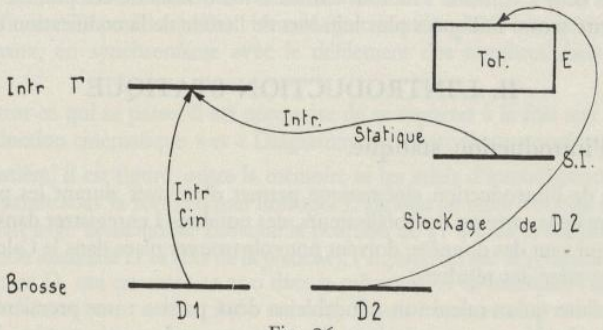


Fig. 26

Le schéma ci-dessus (*fig.* 26) est purement destiné à préciser le rôle de l'introduction statique car le regroupement sur l'introducteur doit faire l'objet de précautions qui seront exposées plus loin.

2. Principe de l'introduction statique.

Pour atteindre le but proposé, les 3 premiers introducteurs, qui sont ceux existant sur la machine standard, sont munis, en plus des circuits précédemment décrits, de circuits spéciaux. Ces circuits leur permettent de recevoir, pendant les points 12 et 13 du cycle, des données contenues dans des totalisateurs.

Pour ce mode particulier d'introduction, le Calculateur électronique émet, de 11,22 à 13,22, des impulsions spéciales qui sont transmises à la tabulatrice. Le Calculateur se substitue alors aux cames du distributeur, qui sont à cet instant toutes ouvertes, et émet ces impulsions en amont des relais d'émission des totalisateurs. En commandant ces relais, il est donc possible d'explorer les totalisateurs par ces impulsions. La planche 10 indique le principe de ce circuit.

On voit que le circuit utilisé est celui du distributeur : il s'ensuit que des constantes sont émises par le distributeur pendant les points 11,22 à 13,22, qui peuvent être utilisées en introduction statique.

L'émission d'introduction statique est codée. Elle a lieu en quatre temps, durant lesquels le Calculateur envoie successivement des poids 1 — 2 — 4 — 8 sur le circuit du distributeur. Ainsi, les barrettes horizontales, ainsi que les plots du distributeur, sont explorés de la façon suivante :

1er temps (poids 1).	barrettes 1 — 3 — 5 — 7 — 9
2e temps (poids 2).	barrettes 2 — 3 — 6 — 7
3e temps (poids 4).	barrettes 4 — 5 — 6 — 7
4e temps (poids 8).	barrettes 8 — 9

Les codes ainsi constitués et filtrés sur les totalisateurs sont utilisables directement sur les 3 premiers introducteurs, moyennant certaines précautions qui seront exposées plus loin.

3. Circuit de commande de l'I.S.

Il est constitué de deux catégories de commandes.

a) un commutateur d'admission de l'émission codée sur le circuit du distributeur ;

b) des commutateurs alternatifs, appelés relais « cinématique — statique », qui au repos, assurent l'introduction cinématique précédemment décrite et, au travail, l'acheminement vers le rythmeur décimal en vue de l'introduction, des émissions d'I. S. filtrées par les totalisateurs ou provenant du distributeur.

La deuxième catégorie de commandes apparaît au connecteur sous la forme de 4 plots référencés I. S. qui émettent à tous les cycles de 0,18 à 13,22. Ces commandes ont pour rôle de mettre au travail les divers commutateurs des circuits de position d'I. S. pendant ces instants.

Ces plots sont à considérer comme des commandes de cycles et doivent être l'objet des mêmes précautions que les commandes de cycles pour éviter tout retour. C'est pour cela qu'il est prévu 4 commandes distinctes, isolées en dehors des points 11, 12 et 13.

Ces 4 commandes émettent à partir de 0,18, afin de mettre au travail les relais T, B, *t* ou alternatifs, faisant partie du circuit de position de l'I. S., un peu avant qu'ils soient explorés par les émissions d'I. S.

On ne doit pas mettre une commande de cycle, de 9 à 1, ou de 9 à 11, en parallèle avec une commande d'I. S. En effet, les contacts des commandes de cycles subiraient une étincelle de rupture à leur retombée, du fait de leur alimentation, à partir de 0, 18 par les plots I. S.

4. Principes généraux d'utilisation de l'I.S.

A. *Précautions générales.*

L'introduction statique, du fait qu'elle utilise, pour transmettre des impulsions électroniques, les circuits électriques de la machine connectée, réclame certaines précautions d'utilisation.

Pendant l'I. S. (11,20 — 13,20 en B. S. ; 11,20 — 17,20 en P.R.D.) les entrées d'un introducteur commandé en I. S. ne doivent jamais être en liaison avec un organe quelconque de la machine connectée autre que les organes émetteurs d'impulsions I. S. Cette protection doit être appliquée aux 12 positions de l'introducteur, même si toutes ne sont pas utilisées en I. S.

Le tableau donné en annexe (Pl. 11) représente par E et R les organes émetteurs et récepteurs de l'I. S. et de la machine connectée.

On voit sur ce tableau que seules les relations E à R verticales sont normales. Toutes les relations horizontales ou obliques sont à proscrire. L'élimination de ces relations ne peut se faire qu'en instituant l'isolement absolu, par alternatifs, des deux ordres de circuits E — R machine connectée (de 13,20 à 11,20) et E — R introduction statique (de 11,20 à 13, 20).

Pendant la durée de l'I. S, les organes émetteurs d'impulsions I. S. ne doivent jamais être reliés à d'autres organes de la machine connectée.

Cet isolement doit se faire par alternatifs du fait que l'exploration en I. S. d'un totalisateur réutilisant le circuit du distributeur, c'est sur les mêmes plots de la tabulatrice qu'en exploration cinématique on recueillera les impulsions codées résultant de l'émission d'introduction statique. Il en est de même pour le distributeur dont les plots sont émetteurs successivement des émissions cinématiques et des émissions statiques.

Les conséquences suivantes en découlent :

Aucun organe récepteur ne doit être relié pendant le temps de l'introduction statique :

— aux plots du distributeur de constantes,

— aux sorties des totalisateurs dont les T ou B et *t* sont commandés pendant l'I.S.,

— aux entrées des introducteurs I_2-$I_{3\cdot1}$-I_4,

Du fait de la séparation du cycle en deux temps distincts quant à l'introduction, une position d'introducteur peut être utilisée deux fois au cours d'un même cycle, en cinématique et en statique. Dans ce cas, l'entrée liste du totalisateur exploré en I.S. permet de regrouper les données introduites de 9 à 0.

Quelques exemples illustreront ces règles :

a) Soit à introduire sur deux mémoires différentes un chiffre du distributeur en cinématique dans l'une, en statique dans l'autre. D'autre part, le même chiffre du distributeur sera

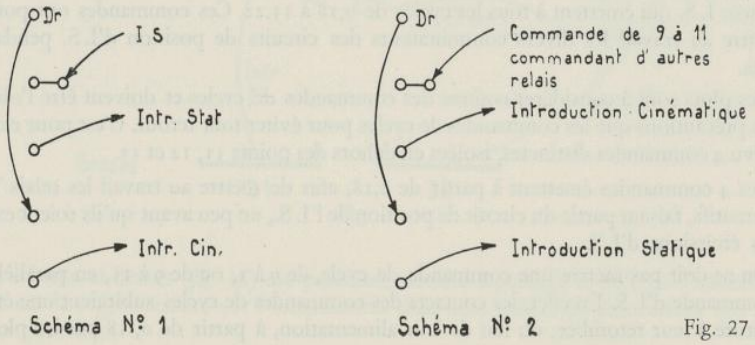


Schéma N° 1 Schéma N° 2 Fig. 27

conduit à une position de comparaison recevant sur son autre entrée une lecture de Brosse. Il faut en premier lieu (*fig.* 27) ventiler l'émission du distributeur sur un alternatif commandé par une commande I. S. (schéma 1). On a intérêt à commander l'alternatif par I.S. et

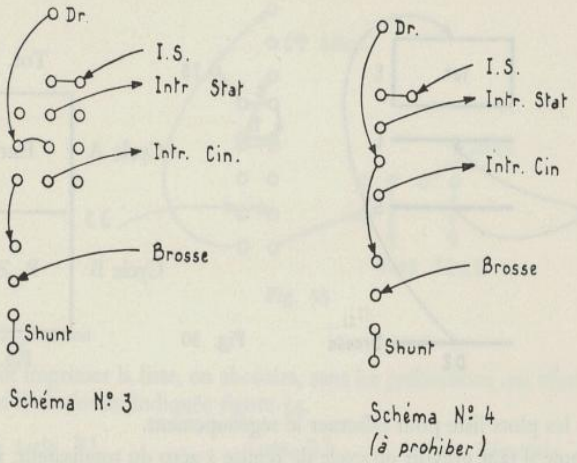


Fig. 28

non par une commande de cycle ou de sélection dont la charge sur d'autres relais peut prolonger la durée (schéma 2).

En second lieu (*fig.* 28), il faut isoler pendant l'I.S. le récepteur que constitue la position de comparaison de l'introducteur, alimenté en statique. Le schéma 3 montre la connexion valable. Le schéma 4 est à prohiber.

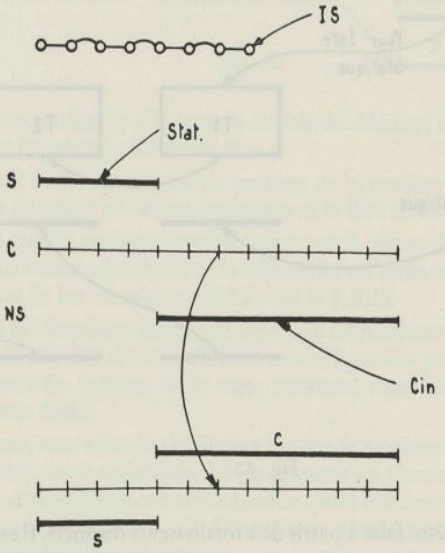


Fig. 29

b) Soit à introduire dans un introducteur cinématique-statique 8 positions en cinématique et 4 positions en statique (*fig.* 29), les positions cinématique et statique étant distinctes. Il est nécessaire d'utiliser un groupe de 12 positions d'alternatifs selon le schéma ci-contre :

c) Soit à effectuer la connexion réalisant l'exemple ayant servi à l'exposé du rôle de l'introduction statique. Ici, il y a à effectuer un regroupement sur le même introducteur d'impulsions d'introduction cinématique et d'introduction statique (*fig.* 30).

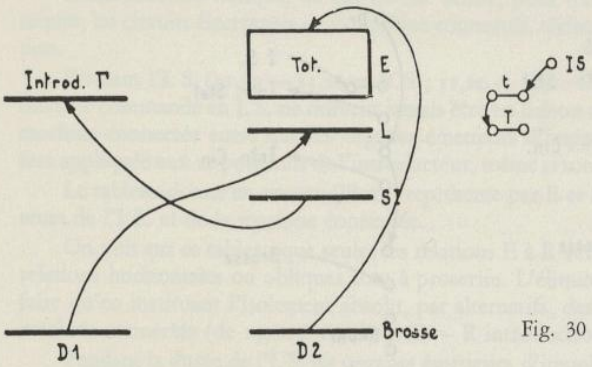


Fig. 30

	Tot. 1	Tot. 2
Cycle A	Enr.	R. Z.
Cycle B	R. Z.	Enr.

Fig. 31

On utilisera les plots liste pour effectuer le regroupement.

En fait, comme il faut prévoir un cycle de remise à zéro du totalisateur, il faut, lorsque l'on veut travailler à 150 introductions statiques minute, prévoir deux totalisateurs montés en bascule double. Le programme de cette bascule est donné par la figure 31.

La sélection d'enregistrement est facile, les relais z étant au travail un cycle sur deux, celui où l'on ne doit pas enregistrer. Le regroupement se fera par les plots liste (*fig.* 32) :

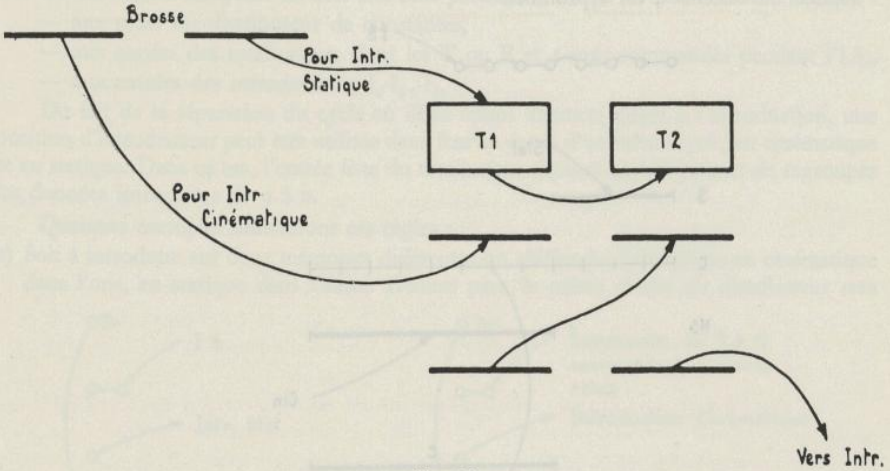


Fig. 32

L'I. S. en bascule doit être faite à partir de 2 totalisateurs distincts. Il est interdit d'utiliser 2 groupes de 6 roues d'un même totalisateur, car on ne peut commander à la fois T et Z sur un même totalisateur sans opérer une commande dépourvue de signification.

La régulation de la bascule se fera à partir d'une commande de sélection qui, alternati-

vement, admettra ou interdira l'accès du compte-carte ou d'un chiffre du distributeur (avant 11) sur une commande de sélection de solde dont le temps permet l'aiguillage des commandes I.S. (*fig.* 33).

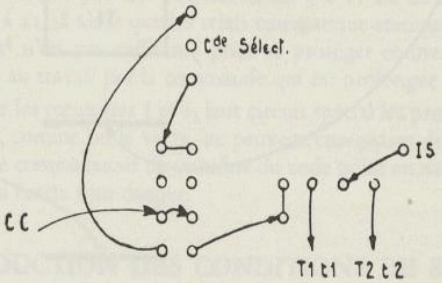


Fig. 33

B. *Forme de l'impression*

Si l'on veut imprimer la liste, on aboutira, sans les précautions qui vont être exposées, à une impression de la forme indiquée figure 34.

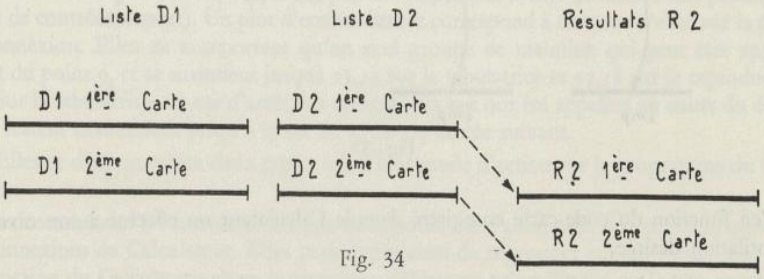


Fig. 34

Les résultats seront décalés d'un espace du fait du décalage d'un cycle entre l'introduction cinématique et l'extraction des résultats.

On peut aboutir à une représentation correcte de la manière suivante (*fig.* 35), en lisant les données en vue du calcul en brosse supérieure et la liste en brosse inférieure :

Cette solution appelle quelques considérations sur la nature des sélections de lecture que l'on doit assurer. La brosse supérieure de la tabulatrice est alors utilisée en brosse de lecture. De même, lorsqu'on lit les données en B.S.P. sur la P.R.D.

a) Bien souvent, le fichier comporte, par groupe, une ou plusieurs cartes maîtresses et une ou plusieurs cartes détail. Les cartes maîtresses seront signalées par un code qui viendra trop tard pour éliminer les lectures de la carte maîtresse dans les zones correspondant aux données des cartes détail.

On enregistre, sans sélection d'élimination sur la machine connectée, dans le calculateur, ces zones des cartes maîtresses. Le programme du Gamma, modifié par le code des C.M. sera réglé, comme on le verra par la suite, pour éliminer les calculs sur ces données sans signification quant à ces calculs. D'une façon générale, toute sélection d'élimination pourra être traitée de cette manière.

b) Si l'on doit ventiler une zone de carte détail en plusieurs directions, on enregistrera cette zone dans une seule mémoire du Gamma et on règlera le programme du Calculateur pour

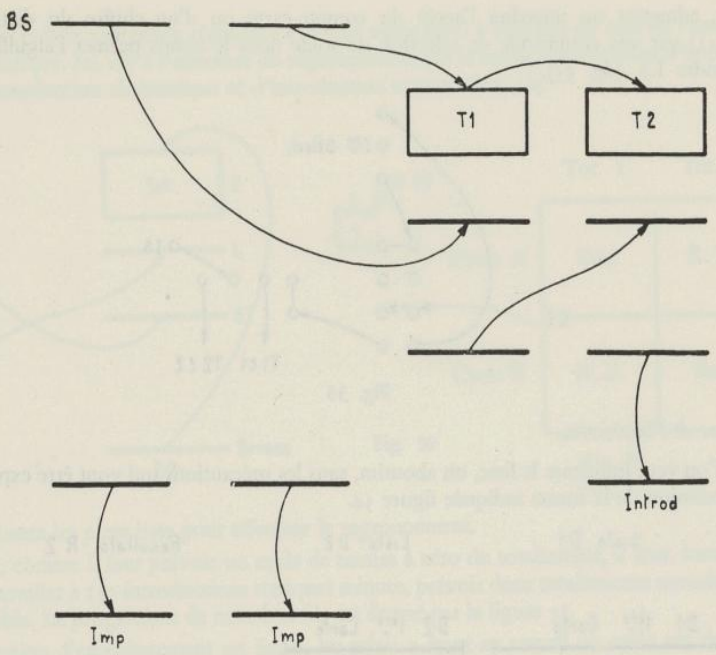


Fig. 35

qu'en fonction du code-carte enregistré dans le Calculateur on effectue à son niveau la ventilation désirée.

c) Si l'on a affaire à une sélection de regroupement, on peut également la faire au niveau du programme du Calculateur électronique en enregistrant toutes les zones à regrouper dans le Gamma et en choisissant, lors du calcul, celle indiquée par le code-carte.

C. *Enregistrement des totalisateurs explorables en I. S.*

Un totalisateur ne pouvant être exploré aux instants où il est susceptible d'enregistrer un nombre, on ne peut explorer en I.S. un totalisateur qui peut exécuter des reports au moment de l'I.S. Il est donc possible d'explorer en I.S., soit un totalisateur qui a enregistré avec report lors d'un cycle précédent, soit un totalisateur ayant enregistré au cycle d'I.S. à condition que cet enregistrement n'ait pu déterminer aucun report.

5. Exécution de l'introduction statique.

Les connexions de position et de commande précédemment décrites constituent une préparation de l'I.S. Cette opération est exécutée grâce au programme interne du Calculateur sur le tableau duquel on l'affiche. Cette exécution sera décrite lors de l'étude des opérations élémentaires du Calculateur.

Il est bon de remarquer qu'en introduction statique les commandes d'introducteurs ne doivent pas être excitées.

6. Remarque.

Il a été dit plus haut qu'on ne devait en aucun cas commander un introducteur pouvant recevoir des impulsions d'I.S. par une commande de 9 à 11 ou de sélection. En effet, ces commandes retombent à 11,18 alors que les relais cinématique-statique travaillent à partir de 11,20. La sécurité de 2^0 n'est pas suffisante pour se protéger contre les effets de selfs des bobines de relais mises au travail par la commande qui est prolongée au-delà de 11,18.

En ce qui concerne les mémoires 1 et 2, leur circuit spécial les protège contre cette particularité. Ces mémoires, comme on le verra, ne peuvent enregistrer de codes égaux ou supérieurs à 10. Pour cela, le commutateur de transfert du code point en mémoire n'est validé que jusqu'au point 1, ce qui écarte tout danger.

III. INTRODUCTION DES CONDITIONS DE SÉLECTION

1. Rôle et description.

Le Gamma 3 est doté de commandes de sélection permettant de modifier son programme selon les éventualités observées sur la machine connectée. Ces commandes sont constituées de petits relais et peuvent être appelées, par une impulsion courte (lecture d'une perforation, shunt de contrôle, report). Un plot d'entrée directe correspond à chacune d'elles sur la plaque de connexion. Elles ne comportent qu'un seul groupe de maintien qui peut être appelé à partir du point 9, et se maintient jusquà 15,14 sur la tabulatrice et 17,18 sur la reproductrice.

Sur la tabulatrice, en cas d'arrêt, les commandes qui ont été appelées au cours du dernier cycle restent maintenues jusqu'à la fin du cycle de lancée suivant.

Elles se distinguent en deux types, selon leur mode d'action sur le programme du Calculateur :

A. Les commandes S contrôlent chacune deux positions d'alternatifs situées sur le tableau de connexions du Calculateur. Elles permettent ainsi de regrouper sur un même plot d'instruction du Calculateur deux instructions différentes selon l'appel ou le non-appel de la commande S, ainsi qu'il sera vu plus loin.

B. Les commandes V agissent sur les circuits internes du Calculateur, en fonction d'instructions affichées une fois pour toutes au tableau de connexions.

La machine standard possède 8 commandes S et 8 commandes V. Le nombre des unes et des autres peut être porté à 16, soit au maximum 32 commandes de sélection.

Les commandes V_0, V_1, V_2, V_3, S_0, S_1 (standard) et V_8, V_9, V_{10}, V_{11}, S_8, S_9 (supplément) sont munies d'un plot de sortie alimenté par le maintien de la commande. Ces plots permettent de commander des organes de la tabulatrice lorsque celle-ci est également affectée par la sélection et que la durée du maintien est convenable.

2. Utilisation des commandes V. et S.

A. Les commandes V et S ne comportent pas d'entrée spéciale. Pour sélectionner la perforation d'appel, on a recours à des relais alternatifs rapides appelés « alternatifs standard ». Ces alternatifs peuvent être commandés, position par position par des chiffres du distributeur. Leur nombre est de 8 sur la machine standard. Il peut être porté à 16.

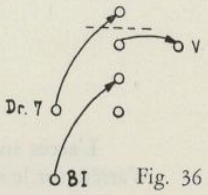


Fig. 36

La connexion de la figure 36 est relative à l'appel d'une commande V par une perfora-

tion 7 lue en B.I. Il ne faut pas s'autoriser de ce caractère pour faire la connexion consistant à exciter un de ces alternatifs par une impulsion de rupteur (ou par l'impulsion à contrôler elle-même) si l'impulsion ainsi filtrée doit être introduite dans un mémoire Gamma. L'Introduction ayant lieu entre les degrés 4 et 6, l'impulsion filtrée risque en effet d'arriver trop tard.

B. Les commandes V et S sont très sensibles et doivent faire l'objet de précautions dans leur emploi.

a) Il est absolument prohibé d'appeler en parallèle une commande V ou S et un relais à double enroulement. En effet, le relais à double enroulement constitue un transformateur.

Lors de l'appel, avec le montage de la figure 37, rien d'anormal n'aura lieu. Mais à 15, la came fera retomber la sélection de solde et la variation de potentiel qui en résultera

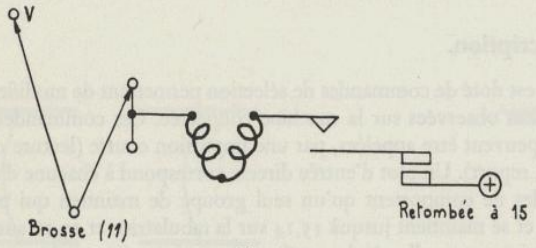


Fig. 37

créera un courant induit sur la bobine d'appel du double enroulement. On appellera ainsi à nouveau la commande V. Il en est de même pour les commandes de sélection lorsque l'on utilise l'entrée directe. Dans ce cas, la retombée du premier groupe est antérieure à 15,14, mais on rappelle néanmoins la commande V par le courant induit qui peut durer deux points.

Ici encore on utilisera les alternatifs standard afin de constituer une entrée filtrée sur V et S. Le montage sera conforme au schéma de la figure 38.

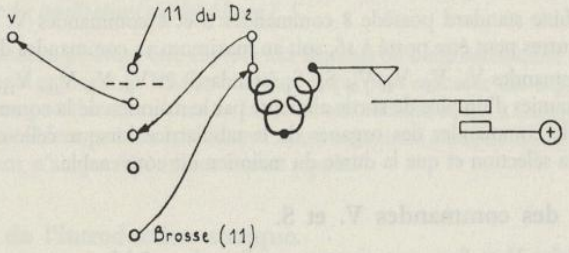


Fig. 38

L'accès sur la commande V est ainsi validé à 11 et le retour du courant induit s'arrête sur le commun.

Le montage de la figure 39 est à prohiber, car le courant induit risquerait de mettre au travail l'alternatif et de valoriser le retour :

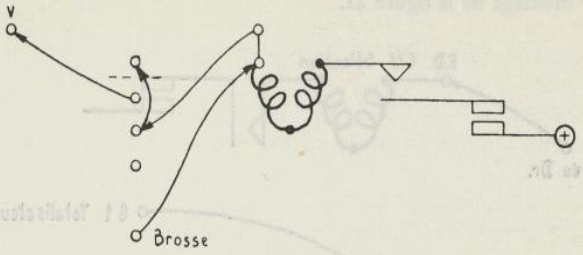


Fig. 39

b) On doit également rejeter, lorsque les colonnes d'appel des deux commandes sont distinctes, le montage de la figure 40.

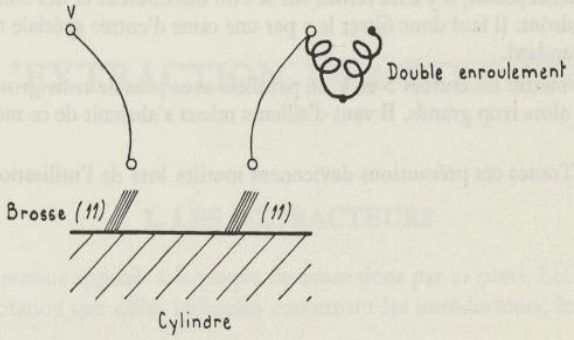


Fig. 40

En effet, malgré la distinction des plots d'appel sur la brosse, lorsque la carte est lue, les balais sont shuntés par le cylindre. En cas de courants induits pendant les points de shuntage des balais, le retour est validé. Le montage de la figure 41 doit donc être fait :

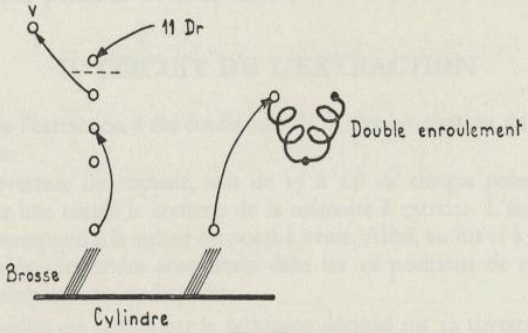


Fig. 41

c) Cette nécessité de filtrer tout appel des commandes de sélection quand il intervient des relais à double enroulement est générale.

Soit le montage de la figure 42.

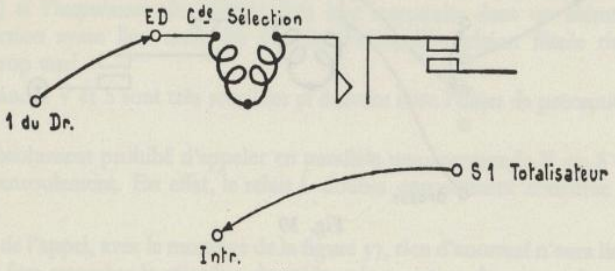


Fig. 42

Lors du courant induit, il y aura retour sur le 1 du distributeur et des anomalies à l'introduction se produiront. Il faut donc filtrer le 1 par une came d'entrée spéciale réglée à 1 ou par un alternatif standard.

On ne peut mettre les entrées S et V en parallèle avec plus de trois gros relais, les effets de self devenant alors trop grands. Il vaut d'ailleurs mieux s'abstenir de ce montage rarement nécessaire.

Remarque : Toutes ces précautions deviennent inutiles lors de l'utilisation des machines à petits relais.

Chapitre quatrième

L'EXTRACTION DES DONNÉES

I. LES EXTRACTEURS

Chaque extracteur apparaît à la plaque de connexions par 12 plots. Les mêmes conventions de numérotation que celles indiquées concernant les introducteurs, leur ont été appliquées.

Ces plots émettent, en fonction des nombres contenus dans les mémoires, des impulsions qui proviennent du rupteur même de la machine connectée, à travers des petits relais situés dans le Calculateur.

Les extracteurs peuvent donc être utilisés sans restriction, comme tout autre organe émetteur. Ils peuvent être connectés à l'imprimante, au magasin de perforation, aux entrées des totalisateurs, aux positions de comparaison.

II. CIRCUIT DE L'EXTRACTION

Le principe de l'extraction a été étudié lors de la synchronisation du Gamma 3 et de la machine connectée.

A chaque ouverture du rupteur, soit de 17 à 24° de chaque point, un comparateur d'égalité reçoit sur une entrée le contenu de la mémoire à extraire. L'autre entrée reçoit le code-point, qui correspond à la valeur du point à venir. Ainsi, au fur et à mesure de la circulation d'une mémoire, les codes enregistrés dans les 12 positions de cette mémoire sont successivement comparés au code-point.

Le signal d'égalité est ventilé par le rythmeur décimal sur 12 thyratrons correspondant aux 12 positions de mémoire et aux 12 plots d'extracteur. L'allumage du thyratron sélectionné par les rythmes décimaux provoque l'excitation d'un relais qui restera maintenu pendant la fermeture du rupteur suivant immédiatement, et dont le contact, alimenté par le rupteur, émettra une impulsion sur un plot de sortie (voir Pl. 12).

Le code-point passant à 9, à 15,17, les codes 9 existant dans la mémoire sont repérés par la comparaison, à partir de 15,17, et les relais d'extraction des positions correspondantes sont excités jusqu'à 9,17. Les contacts de ces relais, alimentés par le rupteur, émettent une impulsion 9 à leur plot de sortie.

A 9,17, le maintien des relais est coupé. Le code-point passe à 8, à 9,17 et le comparateur repère les codes 8 afin de positionner les relais pour le point 8, etc. (*fig.* 43).

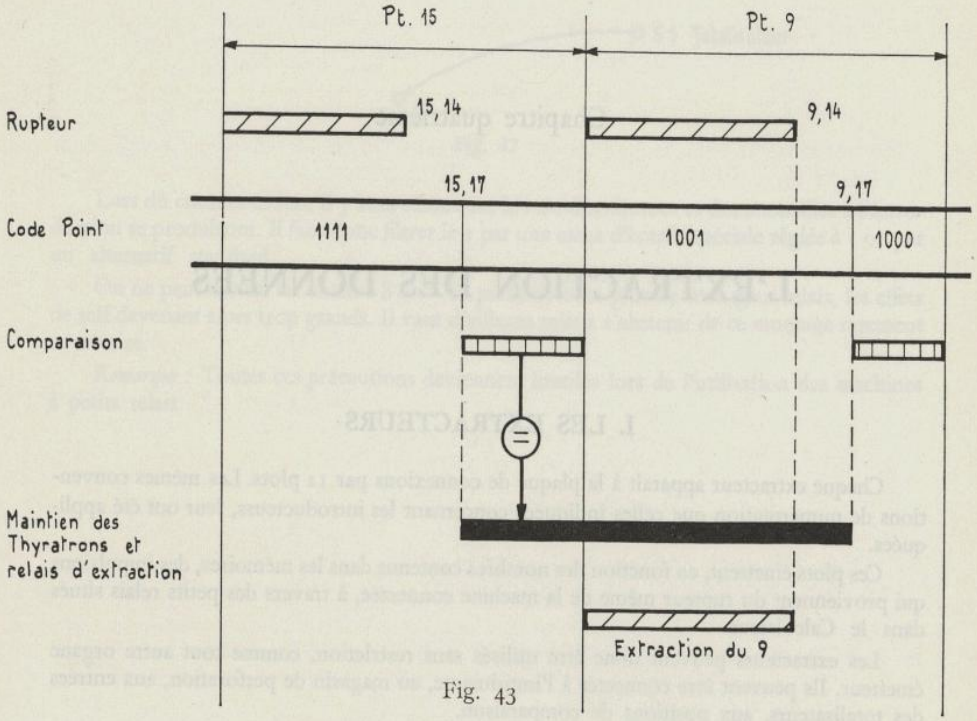


Fig. 43

III. LES COMMANDES D'EXTRACTION

Contrairement aux introducteurs, les extracteurs ne sont pas liés aux mémoires.

Ils sont conçus pour extraire l'une des deux mémoires qui leur sont affectées et sont dotés pour cela de deux plots de commande qui apparaissent à la plaque de connexion au droit des positions de sorties.

Ces commandes ont un double rôle :

— aiguiller l'une ou l'autre des deux mémoires sur le comparateur d'extraction.

— valider la comparaison.

L'organisation générale du circuit est telle que les commandes d'extraction ne doivent changer leur état de travail ou de repos que pendant la fermeture du rupteur. Or, toutes les commandes longues des machines connectées changent leur état de travail ou de repos en dehors de cette fermeture. C'est pourquoi les commandes d'extraction ont un fonctionnement spécial.

Elles comportent un circuit de maintien à petits relais qui ne peut être appelé qu'à 15,4, 9,4 8,4,..... 0,4, 11,4 de tout cycle et qui retombe, sauf nouvel appel, à 4° du point suivant. De plus, les commandes sont figées aux points 12, 13 et 14, de sorte qu'une commande appelée à 11,4 ne retombe qu'à 15,4 (*fig.* 44).

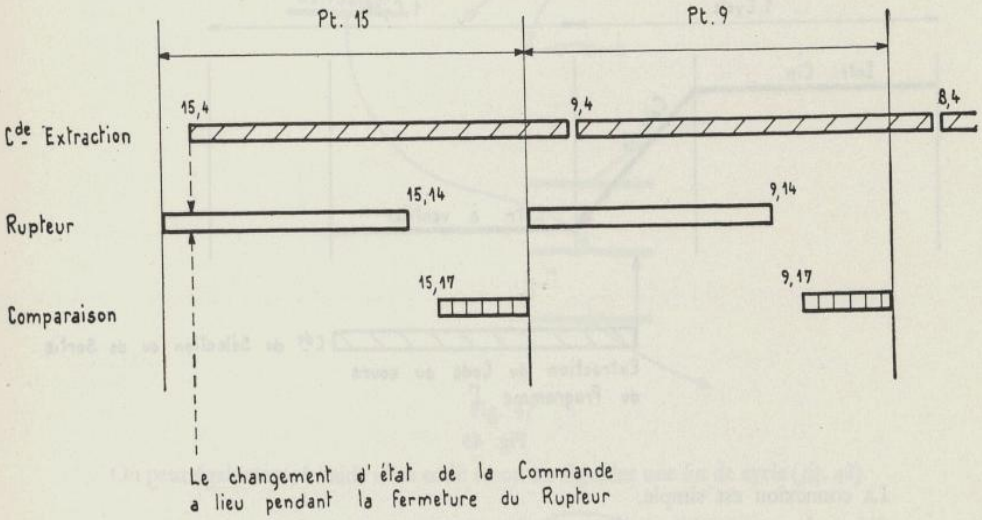


Fig. 44

IV. LES EXTRACTIONS QUALITATIVES

On désigne ainsi les extractions de codes non arithmétiques : 11, 12, 13, 14, 15. De tels codes peuvent en effet être introduits dans les mémoires du Calculateur par le programme du Calculateur (opération KB étudiée plus loin) pour tester une éventualité observée par le Calculateur : sens d'une comparaison, signe d'un solde, etc.

Les codes sont extraits aux points correspondants du cycle puisque le code-point passe par ces valeurs. Ces extractions peuvent soit être utilisées directement, pour commander une perforation hors texte (11 ou 12), soit pour appeler un circuit de maintien chargé d'opérer une sélection. Ce circuit peut être une commande de sélection (11 ou 12), une sélection de solde, ou une des commandes existant sur le Gamma 3 spécialement dans ce but : les commandes de sortie.

1. Utilisation directe du code (11 ou 12).

Le code sera perforé ou imprimé sous la forme d'un 11 ou d'un point ou d'un 12 ou d'un zéro, au point 11 ou 12 du cycle d'extraction. Il ne peut dans ce cas être question de ventiler le nombre en fonction du code extrait ou non extrait, puisque celui-ci est extrait après l'extraction du nombre.

2. Utilisation du code en vue d'une sélection.

Le code devra alors être extrait sur une commande de sélection de la machine connectée ou sur une commande de sortie et le nombre sera extrait au cycle d'extraction, une fois positionnée la commande (*fig.* 45).

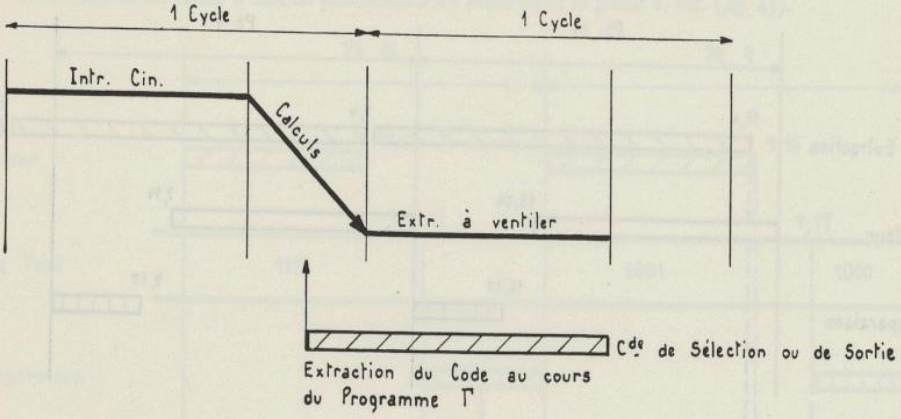


Fig. 45

La connexion est simple.

Si le code est un 11 ou un 12, on prendra une commande de sélection (*fig.* 46).

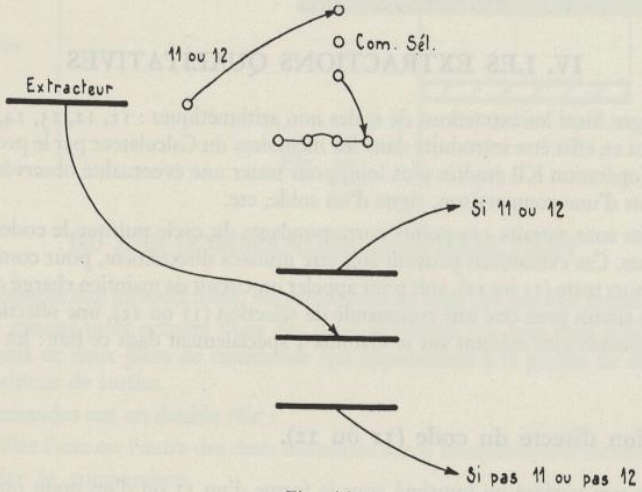


Fig. 46

Si le code est supérieur à 12, on prendra une commande de sortie (*fig.* 47). Ces commandes sont en effet constituées d'un seul groupe de maintien qui peut être appelé à partir du point 13 et se maintient jusqu'à 11,18 du cycle suivant.

Chaque commande apparaît à la plaque de connexions par un plot d'entrée directe et un plot de sortie, alimenté par le circuit de maintien. Il existe 4 commandes CS sur la machine standard. Ce nombre peut être porté à 6.

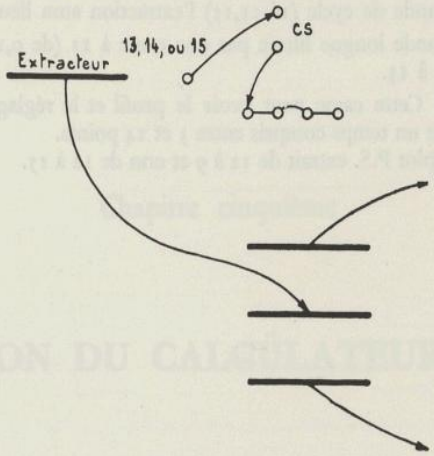


Fig. 47

On peut également, à l'aide d'un code 11 ou 12 aiguiller une fin de cycle (*fig.* 48).

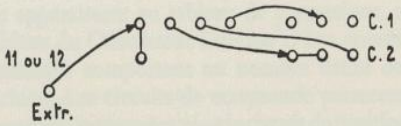


Fig. 48

Bien entendu, lorsque l'on désire extraire, au cours du programme, des codes qualitatifs, il faut organiser le programme du Calculateur et choisir les codes de telle sorte que l'état de la mémoire d'où on extrait ce code ne change pas pendant le point d'extraction du code.

V. COMMANDE DES EXTRACTEURS

Les commandes d'extraction peuvent être validées par des commandes de cycle ou de sélection. Ces commandes peuvent être mises en parallèle avec d'autres organes de la machine connectée sans qu'il en résulte d'inconvénient.

Etant donné le fonctionnement des commandes d'extraction, une commande de cycle de 9 à 1 extrait de 9 à 0 et une commande de cycle de 9 à 11 ou de sélection extrait de 9 à 15.

Si l'on désire extraire de 9 à 1, la commande de cycle de 9 à 1 est trop longue et l'on doit se servir du plot 9 à 1 du connecteur Gamma.

Si l'on désire extraire un point déterminé on ne peut commander l'extracteur par une impulsion du distributeur (dont l'émission cesse avant qu'on soit assuré d'avoir accroché les maintiens nécessaires). Il faut pour cela prendre une commande de cycle filtrée par une came — $6^0 + 18^0$ du point précédant le point que l'on désire extraire.

En règle générale il faut couvrir les degrés 16 à 20 du point précédent.
Extraction de 12 à 15.
Les extracteurs doivent être commandés :

— par une commande de cycle (16-11,15) l'extraction aura lieu de 9 à 15 ;
— par une commande longue filtrée par une came à 11 (de 0,18 à 11,18) l'extraction aura lieu de 12 à 15.
— par la came β. Cette came peut avoir le profil et le réglage désirés et peut ainsi émettre pendant un temps compris entre 3 et 14 points.
A noter que le plot P.S. extrait de 12 à 9 et non de 12 à 15.

Chapitre cinquième

DESCRIPTION DU CALCULATEUR GAMMA 3

I. CIRCUITS DE POSITION ET DE COMMANDE

Contrairement aux machines électromécaniques de la gamme BULL, où les circuits de position et de commande apparaissent au tableau de connexions, seuls les circuits de commande apparaissent au tableau du Calculateur Gamma 3. Les circuits de position font l'objet d'une organisation systématique comportant un nombre défini de commutateurs, prévues dans le câblage de la machine. Les circuits de commande permettent de choisir, au tableau de connexions, les commutations intéressées pour le circuit de position que l'on veut constituer.

II. CIRCUITS DE POSITION

Cette organisation repose sur l'attribution à l'une des mémoires, la mémoire M_1, du rôle d'organe de calcul. Cette mémoire est dite « mémoire opérateur ». Toutes les autres mémoires servent uniquement à enregistrer des données et des résultats ; elles sont dites « mémoires banales » ((Pl. 14).

Que la mémoire 1 soit dotée du pouvoir de calcul ne veut pas dire qu'elle soit capable de faire seulement des calculs : elle est également dotée de toutes les propriétés des mémoires banales et on peut s'en servir comme d'une simple mémoire banale dans certaines opérations.

D'autre part, l'une des mémoires banales, la mémoire 2, tout en ne devenant pas capable de calculer, est annexée, dans certaines opérations, à la mémoire 1 dont elle double ainsi la capacité.

La mémoire opérateur est dotée de la fonction calcul, grâce à un « additionneur-soustracteur » qui constitue sa douzième position. Cet organe comporte deux entrées sur lesquelles sont amenés les deux termes, et une sortie qui émet la somme ou la différence.

L'une des entrées est alimentée en permanence par la sortie de la mémoire opérateur et la sortie est systématiquement connectée à l'entrée de cette mémoire.

Il en résulte que dans toute addition ou soustraction, l'un des facteurs doit être préala-

blement enregistré dans la mémoire opérateur et que le résultat de l'opération se forme toujours dans la mémoire opérateur à la place du facteur qui s'y trouvait.

La deuxième entrée de l'additionneur-soustracteur est alimentée par un circuit particulier dit « canal-donnée ». Il peut recevoir la sortie de l'une quelconque des mémoires du Calculateur et transmettre son contenu pour l'ajouter, ou le soustraire, à celui de la mémoire opérateur.

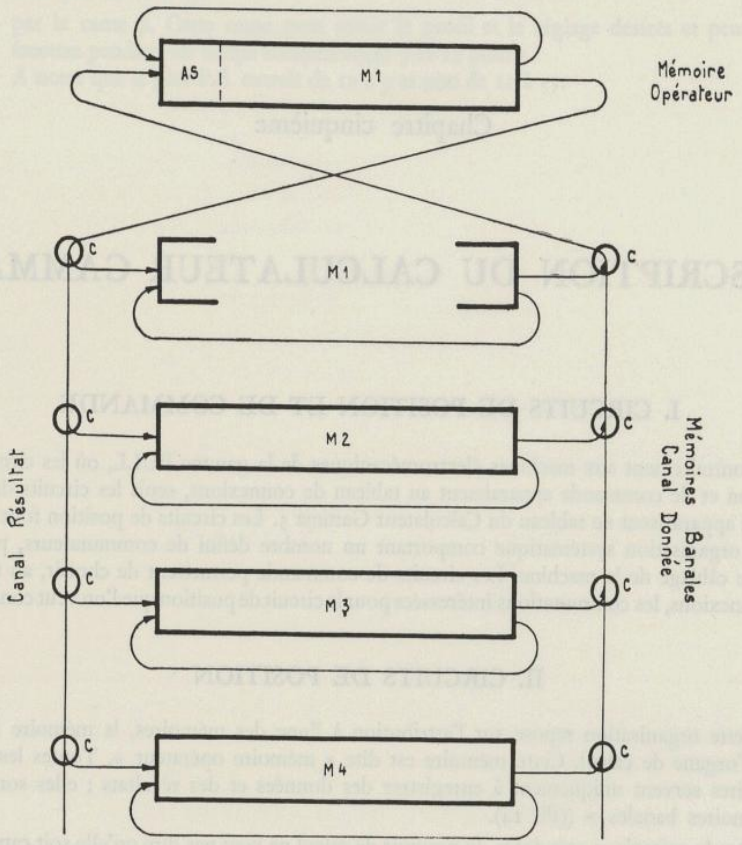


Fig. 49

Un circuit symétrique, dit « canal-résultat » est alimenté par la sortie de la mémoire opérateur et permet de transférer son contenu dans l'une quelconque des mémoires du Calculateur.

On a figuré sur un schéma général (*fig.* 49) les relations entre les mémoires, deux fois la mémoire 1, afin de montrer son double rôle : opérateur et banale.

En effet, le contenu lui-même de la mémoire opérateur peut être considéré comme donnée de calcul ou comme résultat à renvoyer dans la mémoire 1, comme on le verra par la suite. Néanmoins, pour rappeler le caractère spécial d'une telle mémoire banale, on a figuré cette mémoire sous la forme d'un rectangle inachevé.

L'accès des sorties de mémoire sur le canal-donnée ainsi que l'accès du canal-résultat sur les entrées des mémoires est contrôlé par des commutations C mises au travail par les circuits de commandes validés par le programme. Ainsi le programme du Calculateur a pour rôle de valider le passage des nombres par l'un ou l'autre des deux canaux, les deux canaux n'étant jamais validés simultanément. Le programme a également pour rôle de choisir la mémoire dont la sortie est à connecter au « canal-donnée » et l'entrée au « canal-résultat ». Ces deux connexions sont opérées simultanément, mais seule, celle qui correspond au canal utilisé est validée.

III. CIRCUITS DE COMMANDE

1. Organisation générale.

Les circuits de commande du Calculateur Gamma 3 sont organisés de la façon suivante :

a) Un tableau de connexions permet d'afficher des opérations élémentaires d'effacement, de transferts et arithmétiques.

b) Le Calculateur prend connaissance, ligne à ligne d'instructions ainsi affichées, des opérations qu'il doit effectuer. A partir de cette connaissance, un ensemble d'organes déclenche des circuits de commandes, appelés trains d'impulsions, qui, en les validant, agissent sur les commutations que fait intervenir l'opération affichée.

Le tableau de connexions constitue le « programme externe ». Il est variable pour chaque problème et dépend de l'affichage que l'on a effectué.

La séquence des trains d'impulsions faisant intervenir opportunément les commutations constitue le « programme interne ».

Il est déterminé pour chaque type d'opération que l'on affiche au programme externe et ne peut être varié par aucun moyen.

On appelle « trains » des circuits électroniques de maintien constitués d'une mémoire pouvant enregistrer une seule impulsion binaire. Lorsque celle-ci est présente, elle tourne dans la mémoire à la cadence de répétition des positions d'impulsion dans les mémoires de nombres. Un régénérateur placé dans sa circulation émet alors une suite continue d'impulsions ou « train » d'impulsions. Ces trains jouent le rôle de tension de commande pour contrôler les commutations.

Prenons pour montrer le rôle de ces trains l'exemple le plus simple : celui du transfert d'une mémoire banale à la mémoire opérateur. Ce transfert utilise le canal-donnée et s'effectue de la façon suivante (*fig.* 50). Sur le canal-donnée, se trouve une commutation qui est un

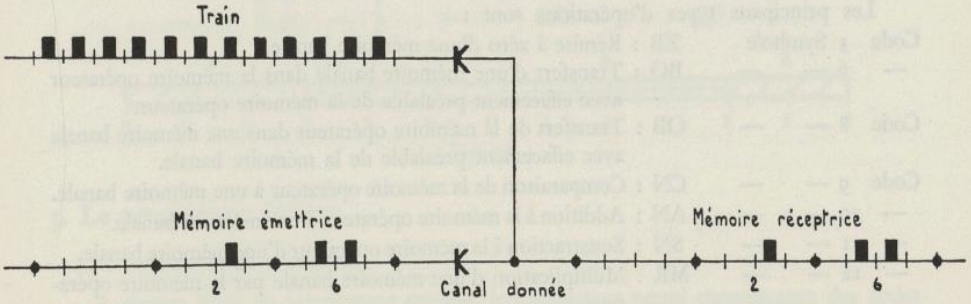


Fig. 50

conditionneur, dont l'une des entrées reçoit la mémoire émettrice et l'autre entrée le train qui contrôle le transfert.

A l'entrée de la mémoire réceptrice, se présentera une impulsion si la mémoire émettrice présente une impulsion au niveau de la commutation, donc à sa sortie, et si le train en présente une. Le train en présente une systématiquement à chaque Θ tout le temps de sa présence. La mémoire en émet une si elle en contient une. Dans ces conditions, toute impulsion présente dans la mémoire sera transmise si le train est présent : le Calculateur le rend présent au moment où il y a lieu d'effectuer le transfert. On comprend facilement que pour transférer une mémoire de 12 τ, il faille un train de 48 impulsions, donc de 12 τ de durée.

2. Les instructions du programme.

Les opérations constituant une suite de calculs peuvent être classées en opérations à 1, 2 ou 3 adresses, suivant que leur exécution fait intervenir, 1, 2 ou 3 organes de mémoire, qu'il faut préciser au programme en donnant leur adresse. Ce sont :

— Opération à une adresse : remise à zéro d'une mémoire.

— Opération à deux adresses : transfert d'une mémoire à une autre mémoire.

— Opération à trois adresses : les 4 opérations arithmétiques, pour lesquelles il faut deux mémoires pour fournir les données et une mémoire pour recevoir le résultat.

Le Calculateur électronique Gamma 3 procède par opérations élémentaires successives dont chacune est définie par un petit nombre d'instructions constituant une « ligne d'instruction ».

Chaque ligne d'instruction ne peut donner l'adresse que d'une seule mémoire banale ; il en résulte que toute opération de transfert doit utiliser la mémoire opérateur comme mémoire émettrice ou réceptrice, et que toute opération arithmétique doit utiliser la mémoire opérateur pour fournir l'une des données et pour recevoir le résultat. Ces conditions correspondent exactement à celles qui découlent de l'organisation des circuits de position.

Une ligne d'instruction est constituée de 4 instructions, qui sont définies par des codes compris entre 0 et 15.

A. *L'instruction « type d'Opération »* (*T.O.*).

Cette instruction a une action sur un grand nombre de circuits internes de la machine, ses effets essentiels sont :

— Validation du canal-donnée ou du canal-résultat.

— Commande de l'additionneur-soustracteur en addition ou en soustraction.

— Contrôle du programme interne de l'opération qu'elle définit.

Les principaux types d'opérations sont :

Code	3	Symbole	ZB : Remise à zéro d'une mémoire banale.
—	6	— —	BO : Transfert d'une mémoire banale dans la mémoire opérateur avec effacement préalable de la mémoire opérateur.
Code	8	— —	OB : Transfert de la mémoire opérateur dans une mémoire banale avec effacement préalable de la mémoire banale.
Code	9	— —	CN : Comparaison de la mémoire opérateur à une mémoire banale.
—	10	— —	AN : Addition à la mémoire opérateur d'une mémoire banale.
—	11	— —	SN : Soustraction à la mémoire opérateur d'une mémoire banale.
—	12	— —	MR : Multiplication d'une mémoire banale par la mémoire opérateur.
—	13	—	DR : Division de la mémoire opérateur par une mémoire banale.

B. *L'instruction « Adresse » (AD).*

Sauf quelques cas particuliers, cette instruction a pour unique effet de mettre l'une des mémoires en communication avec les canaux. Ces mémoires sont numérotées de 1 à 4 sur la machine standard, de 1 à 7 sur la machine complète, et de 1 à 15 sur la machine avec dispositif de mémoires supplémentaires.

Le code d'adresse est fourni par le numéro de la mémoire banale devant intervenir dans l'opération définie par TO :

— dans le cas d'une opération BO, l'AD définit la mémoire émettrice,

— dans le cas d'une opération OB, l'AD définit la mémoire réceptrice,

— dans le cas d'une opération AN, l'AD définit la mémoire devant s'ajouter à la mémoire opérateur,

etc.

La mémoire 1, ayant pour adresse 1, est la mémoire opérateur considérée comme mémoire banale.

L'adresse 0 est utilisée pour distribuer des facteurs constants. L'utilisation des adresses 0 et 1 sera décrite dans chaque cas de type d'opération.

C. *Les instructions « ordre début » et « ordre fin » (OD—OF).*

Ces deux instructions viennent compléter l'AD pour limiter aux positions voulues celles des 12 positions décimales de la mémoire qui doivent être transférées ou effacées. Elles permettent de « splitter » les mémoires position par position, et de découper différemment une même mémoire en cours de programme.

Les séparations entre les 12 positions d'une mémoire sont numérotées de 0 à 12 (ainsi qu'on l'a déjà dit) : l'ordre début d'un nombre enregistré dans une mémoire est égal au numéro de sa limite droite et son ordre fin est égal au numéro de sa limite gauche. Les mémoires étant réalisées par des circuits en boucle, la limite gauche de la douzième position est la même que la limite droite de la première, c'est pourquoi l'OF de la douzième position est généralement pris égal à zéro.

La différence OF — OD est égal au nombre de chiffres.

Exemple (*fig.* 51) :

Le nombre	A (2 chiffres)	est défini par	OD = 0	OF = 2
—	B (4 —)	—	OD = 2	OF = 6
—	C (2 —)	—	OD = 6	OF = 8
—	D (1 —)	—	OD = 8	OF = 9
—	E (3 —)	—	OD = 9	OF = 0

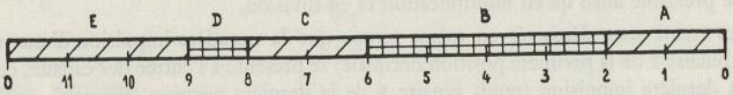


Fig. 51

3. Le programme interne.

A. *L'introduction du programme.*

A chaque ligne de programme externe, le Calculateur prend connaissance des codes TO, AD, OD, OF définissant l'opération à effectuer. Cette prise en charge des codes s'appelle

« l'introduction du programme ». Pour cela les codes affichés sont introduits, en particulier, dans 4 mémoires de programme correspondant aux 4 types d'instruction et portant les mêmes noms TO, AD, OD, OF.

Ces 4 mémoires, de même structure technologique que les mémoires banales, sont constituées chacune d'une boucle de 1 τ capable d'enregistrer les poids binaires 1, 2, 4, 8.

Les commutations qui valident cette introduction sont contrôlées par deux trains d'impulsions. L'un se charge de l'effacement des codes d'instruction de la ligne précédente, l'autre se charge de l'introduction des codes de la ligne qui vient en service.

Le train d'effacement s'appelle Tz. Il émet 4 impulsions qui effacent, dans les 4 mémoires programme, les codes de l'opération qui s'est achevée.

Le train d'introduction s'appelle Tp. Il émet 128 impulsions qui valident le transfert dans les mémoires programme des codes affichés sur la ligne à exécuter. On pourrait penser que le train Tp devrait logiquement durer 4 Θ. En effet, il s'agit de contrôler un transfert dans une mémoire de 4 Θ. Toutefois, les codes affichés au tableau le sont par le moyen de fiches de connexions et de contacts électriques multiples qui ne saurait véhiculer des impulsions d'aussi courtes durées que les impulsions Θ. Il faut que ces impulsions soient plus longues et soient transformées à l'introduction dans les mémoires en impulsions de 1 Θ. C'est pourquoi la mémoire programme doit faire 32 tours ($4 \times 32 = 128$) pour être successivement garnie des poids binaires qu'elle peut contenir.

Les deux trains Tp et Tz sont démarrés simultanément : l'effacement d'une mémoire consiste en effet à ne pas maintenir sa boucle de façon que le code sorti ne se présente plus à l'entrée. Il est donc possible d'entrer dans la mémoire sur une position tandis qu'on efface la position qui sort.

4. L'exécution du programme interne.

Les codes enregistrés dans les mémoires programme ne font que *préparer* les commutations des circuits de nombre qui seront nécessaires pour faire l'opération désirée. Mais ces commutations n'agissent que sous le contrôle de trains qui jouent un double rôle : décomposer l'opération en plusieurs phases et valider chaque commutation pendant une durée bien déterminée, correspondant exactement au temps de passage des nombres dans les commutateurs. Les commutations de nombre sont contrôlées par trois trains :

— le *train* T_1 valide les transferts par le canal-donnée ou le canal-résultat, suivant le code du type d'opération. Il assure également l'effacement intégral de la mémoire opérateur dans l'opération BO,

— le *train* T_2 assure l'effacement des mémoires banales,

— le *train* T_3 commande les décalages de la mémoire opérateur intervenant dans le cadrage préalable ainsi qu'en multiplication et en division.

Ces trains sont démarrés en même temps que la première impulsion d'une mémoire (poids binaire 1 de la première position décimale) se présente à l'entrée des canaux, et cessent avec la dernière impulsion (poids binaire 8 de la dernière position décimale), après avoir émis systématiquement 48 impulsions.

5. La séquence des trains.

L'évolution des différentes phases d'une opération est surveillée en permanence par deux comparateurs : le comparateur de cadrage, principalement chargé des décalages, et le comparateur général, principalement chargé des transferts.

Ces comparateurs sont dotés de mémoires qui conservent le souvenir des comparaisons faites. Ils ont pour rôle de déterminer, à la fin de chaque tour de mémoire, en fonction du type d'opération et de l'état d'avancement de l'opération, le départ d'un des trains T_1 T_2 ou T_3, ou l'émission du signal de fin d'opération démarrant Tp.

Les trains agissent eux-mêmes sur les comparateurs, soit indirectement en faisant évoluer le contenu des mémoires reliées aux comparateurs, soit directement afin de faire conserver par les comparateurs le souvenir de leur exécution..

6. La sélection des lignes de programme.

Les instructions affichées au tableau sont normalement exécutées ligne par ligne dans l'ordre croissant des lignes de programme. Pour cela, à tout instant, une seule ligne parmi les 64 est sélectionnée. Cette sélection est assurée par une mémoire spécialisée dite « numéro de ligne » (NL) qui contient en permanence au cours du calcul le code binaire du numéro de la première des lignes à exécuter.

Le contenu de cette mémoire est indiqué par 6 voyants lumineux situés à droite du tableau de connexion et représentant respectivement, de haut en bas : 1, 2, 4, 8, 16, 32. La somme des valeurs des voyants allumés donne le numéro de la ligne sélectionnée.

Les 64 plots TO sont regroupés sur une sortie unique par un circuit constitué de diodes au germanium et appelé « pyramide ». Seul le code connecté au plot TO de la ligne sélectionnée par la mémoire NL apparaît sur cette sortie. Trois autres pyramides identiques assurent le regroupement des 64 plots AD, OD, OF, sur 3 autres sorties.

Les sorties des 4 pyramides aboutissent à 4 mémoires spécialisées dites « mémoires programme » : TO, AD, OD, OF. Ces mémoires enregistrent les codes affichés sur la ligne sélectionnée et déterminent l'exécution par le Calculateur des instructions correspondantes.

Le changement de ligne de programme est assuré par la progression de la mémoire NL contrôlé par le train Tl. Il faut remarquer qu'après les 128 impulsions de Tp, il reste 16 Θ pour terminer les 36 τ qui encadrent la durée de Tp. Ces 16 Θ sont utilisés par le train Tl qui prépare la sélection de la ligne suivante avant même que les opérations de la ligne en cours ne soient commencées.

Une ligne de programme se présente donc comme la séquence de la forme représentée figure 52.

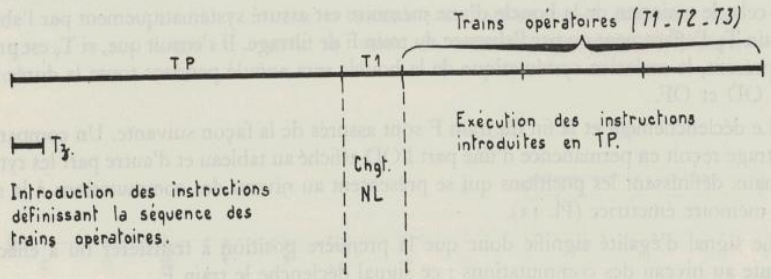


Fig. 52

Comme il a été dit plus haut, la séquence propre à chaque opération est contrôlée par deux comparateurs qui, se tenant constamment au courant de l'état d'avancement de la séquence déclenchée, font démarrer systématiquement les trains qui la composent sans qu'il soit possible d'agir sur elle de l'extérieur de quelque manière que ce soit.

IV. LE FILTRAGE

Le rôle du filtrage est de distinguer les différents nombres qui peuvent se trouver dans une mémoire banale. C'est l'organe qui permet de splitter les mémoires position par position.

Dans le cours d'un transfert ou d'un effacement, ce résultat est obtenu en ne transférant on en n'effaçant qu'une partie de la mémoire, celle contenant un nombre déterminé par son ordre début (OD), c'est-à-dire l'ordre décimal de son chiffre le plus à droite, et son orde fin (OF), c'est-à-dire l'ordre décimal de son chiffre le plus à gauche. Ce résultat est obtenu à l'aide d'un train de filtrage F établi entre OD et OF limitant le nombre.

Ce train est conjugué par une condition « et » avec le train T_1 dans le cas d'un transfert le long du canal-donnée ou du canal-résultat (*fig.* 53).

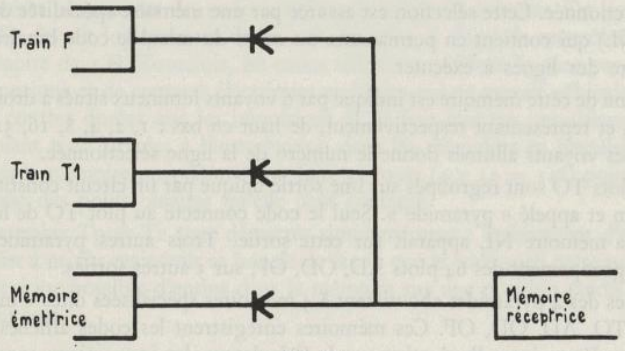


Fig. 53

Dans ces conditions il faut, pour qu'un code présent dans la mémoire émettrice soit transféré dans la mémoire réceptrice que le train de transfert soit présent et que le train F le soit également. Or, le train F est déclenché à partir de l'OD définissant le début du nombre jusqu'à l'OF définissant la fin de ce nombre. Le transfert n'aura donc lieu que pendant le temps de défilement du nombre défini par OD et OF.

Le filtrage permet également d'effacer une mémoire seulement entre un OD et un OF. Pour cela, le maintien de la boucle d'une mémoire est assuré systématiquement par l'absence du train T_2 d'effacement ou par l'absence du train F de filtrage. Il s'ensuit que, si T_2 est présent et F présent, le maintien systématique de la boucle sera annulé pendant toute la durée de F, entre OD et OF.

Le déclenchement et la fin du train F sont assurés de la façon suivante. Un comparateur de filtrage reçoit en permanence d'une part l'OD affiché au tableau et d'autre part les rythmes décimaux définissant les positions qui se présentent au niveau des commutations à la sortie de la mémoire émettrice (Pl. 13).

Le signal d'égalité signifie donc que la première position à transférer ou à effacer se présente au niveau des commutations : ce signal déclenche le train F.

Ce même comparateur de filtrage reçoit également en permanence l'OF affiché au tableau et les rythmes décimaux.

Le deuxième signal d'égalité signifie donc que la dernière position à transférer et à effacer s'est présentée au niveau des commutations : ce signal invalide le train F.

Si l'OF est nul, ou inférieur ou égal à l'OD, le filtrage émet depuis l'OD jusqu'à la

douzième position incluse. Il s'ensuit qu'il faut avoir soin, lors d'un transfert ou d'un effacement de choisir la position du nombre dans la mémoire de façon à ce qu'il y apparaisse entre un OD et un OF tels que OF soit plus grand que OD. Si un nombre est coupé en deux, on ne transférera ou on n'effacera que ses poids faibles compris entre OD et la position 12 incluse (*fig.* 54).

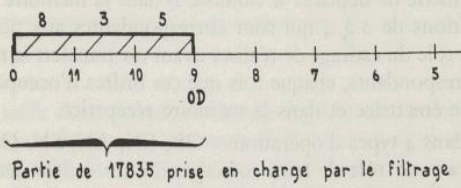


Fig. 54

En résumé, le filtrage a trois actions :

— Il valide le canal-donnée uniquement pendant le passage des positions décimales comprises entre OD et OF et interdit le transfert dans la mémoire opérateur des chiffres éventuellement contenus dans la mémoire banale, à droite et à gauche du nombre filtré.

— Il valide le canal-résultat dans les mêmes conditions et interdit le transfert dans une mémoire banale des chiffres éventuellement contenus dans la mémoire opérateur, à droite et à gauche du nombre filtré.

— Il conditionne l'effacement des mémoires banales afin de n'effacer que la zone définie par le filtrage et conserver les chiffres enregistrés à droite et à gauche de cette zone.

V. LE CADRAGE

Le splittage des mémoires position par position conduit à la possibilité de placer les nombres de telle sorte qu'on utilise la capacité maximum des mémoires. Mais dans ces conditions, il arrive fréquemment que des nombres enregistrés dans deux mémoires et qui doivent être conjugués position par position décimale correspondante (cas de l'addition) soient enregistrés entre des OD et OF n'assurant pas cette correspondance.

Un exemple d'addition montre le problème qui se pose. Soient deux nombres A et B dans une mémoire banale. On désire en faire, en mémoire opérateur, la somme A + B (*fig.* 55).

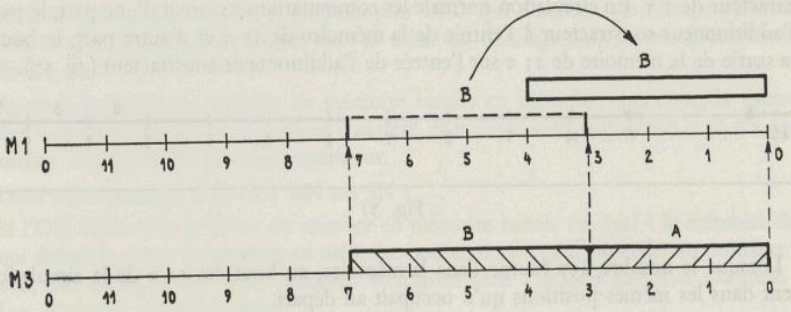


Fig. 55

Une première opération (BO) permettra d'amener B en mémoire opérateur. Du fait du synchronisme des mémoires, B s'y placera de 3 à 7 ainsi qu'il se trouve en mémoire 3. Lors de l'addition, du fait du même synchronisme, A se placera en M_1 de 0 à 3 et le nombre résultant en M_1 ne sera pas la somme A + B mais la somme 10^n B + A, n étant égal à la différence entre les OD des deux nombres : ici $n = 3$.

Il faut donc qu'un dispositif permette de déplacer le nombre B dans la mémoire opérateur jusqu'à ce qu'il occupe les positions de 0 à 4 qui sont correspondantes aux positions de A dans la mémoire banale. C'est le rôle du cadrage de réaliser avant un transfert la mise en coïncidence des ordres décimaux correspondants, chaque fois que ces ordres n'occupent pas les mêmes positions dans la mémoire émettrice et dans la mémoire réceptrice.

Le cadrage préalable intervient dans 4 types d'opérations : OB, CN, AN, SN. Dans les opérations CN, AN, SN, le cadrage a pour rôle de faire correspondre les ordres décimaux des deux facteurs de l'opération. Dans l'opération OB, le cadrage a pour rôle de faire coïncider le nombre contenu en mémoire opérateur avec les positions de mémoire banale destinées à le recevoir en transfert : soit, en effet, en M_1 un nombre enregistré de 3 à 9 que l'on désire transférer en M_1 de 0 à 6. Il faut d'abord déplacer le nombre en M_1 afin qu'il se place de 0 à 6 avant ce transfert (*fig.* 56).

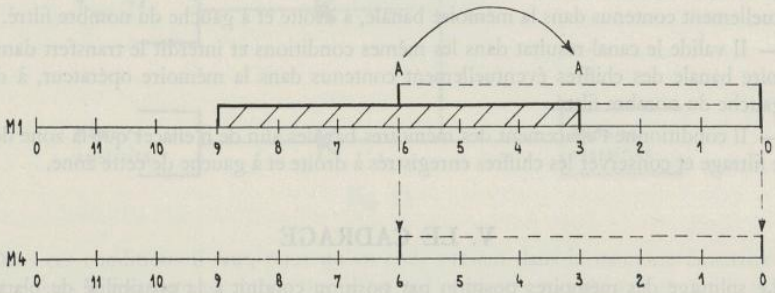


Fig. 56

Cette mise en coïncidence est toujours réalisée par déplacement du nombre contenu en mémoire opérateur. Ce déplacement est opéré par décalages successifs, à raison d'une position vers la droite par tour de mémoire. Les décalages sont accompagnés d'une permutation circulaire qui ramène en douzième position le chiffre contenu en première position.

La mémoire opérateur est composée d'une mémoire de 11 τ et d'un additionneur-soustracteur de 1 τ. En circulation normale les commutations assurent d'une part, le passage de l'additionneur-soustracteur à l'entrée de la mémoire de 11 τ et d'autre part, le bouclage de la sortie de la mémoire de 11 τ sur l'entrée de l'additionneur-soustracteur (*fig.* 57).

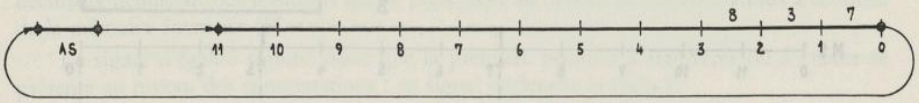


Fig. 57

Lorsque le nombre 837 tourne dans la mémoire, au bout des 12 τ de la circulation, il revient dans les mêmes positions qu'il occupait au départ.

Si l'on décompose la circulation du nombre, on voit qu'au bout du premier τ le nombre

837 occupe les positions suivantes, décalées d'une position vers la droite par rapport à l'implantation initiale (*fig.* 58).

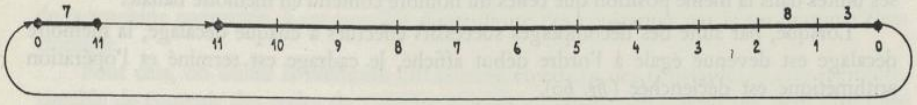


Fig. 58

Il suffit donc de maintenir cette implantation pour que le décalage cherché soit effectué. Il reste 11 τ de temps à parcourir. Si l'on fait donc circuler séparément les deux parties de la mémoire, celle de 11 τ pendant 11 τ, le nombre 83 reviendra à sa position initiale après un tour, celle de 1 τ pendant 11 τ, le nombre 7 reviendra à sa position initiale après 11 tours (*fig.* 59).

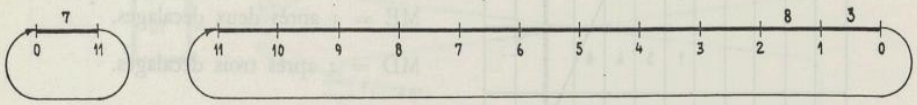


Fig. 59

C'est ainsi qu'est réalisé le décalage en vue du cadrage. Un train T_2 est démarré en même temps que la mémoire. Durant le premier τ la commutation normale est assurée, et à partir de cet instant la circulation se fait sur deux boucles relatives aux deux parties de la mémoire opérateur. Les commutations de boucles sont contrôlées par le train T_3. Il faut donc démarrer autant de trains T_3 qu'on désire opérer de positions de décalage.

Le cadrage ayant pour effet de déplacer le contenu de la mémoire opérateur, il est nécessaire de repérer à tout instant la position de son contenu. C'est le rôle d'une mémoire annexe de la mémoire opérateur, qui est appelée « mémoire décalage » MD. Cette mémoire peut contenir un code compris entre 0 et 11 qui, en principe, indique à tout instant la position début du nombre enregistré en M_1. Lors de tout décalage de la mémoire opérateur, la mémoire décalage est tenue à jour par soustraction d'une unité à chaque décalage. Pour tenir compte de la permutation circulaire, le décomptage de MD est agencé pour donner $0 - 1 = 11$.

Le fonctionnement du cadrage est le suivant :

Dans une opération BO, le nombre introduit depuis une mémoire banale dans la mémoire opérateur se retrouve dans les mêmes positions de la mémoire opérateur que celles qu'il occupe dans la mémoire banale. C'est pourquoi l'OD qui détermine le filtrage pendant le transfert est simultanément introduit dans la mémoire décalage dont le précédent contenu est effacé.

Après le transfert du nombre de mémoire banale en mémoire opérateur, la mémoire décalage contient donc l'OD du nombre transféré, et indique ainsi la position la plus à droite du nombre enregistré en mémoire opérateur.

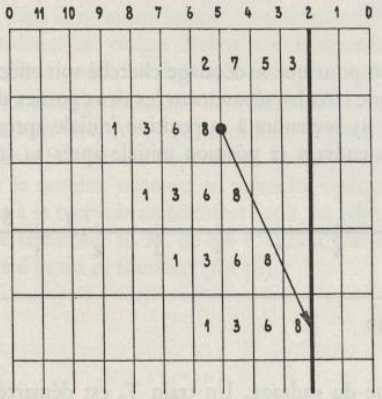
Dans une opération OB, CN, AN ou SN :

Si l'OD définissant le début du nombre en mémoire banale est égal à la mémoire décalage qui définit le début du nombre en mémoire opérateur, on peut procéder au transfert : en addition, par exemple, les deux nombres viendront bien s'ajouter position décimale sur position décimale correspondante.

Si l'OD est différent du contenu de la mémoire décalage, il y a lieu, avant d'effectuer le

transfert, de placer les deux nombres en coïncidence, c'est-à-dire de les « cadrer ». Le cadrage provoque alors automatiquement des décalages du nombre contenu en M_1, jusqu'à ramener ses unités dans la même position que celles du nombre contenu en mémoire banale.

Lorsque, par suite des décomptages successifs effectués à chaque décalage, la mémoire décalage est devenue égale à l'ordre début affiché, le cadrage est terminé et l'opération arithmétique est déclenchée (*fig.* 60).



Mémoire banale :
OD = 2.

MD = 5 au début de l'opération.

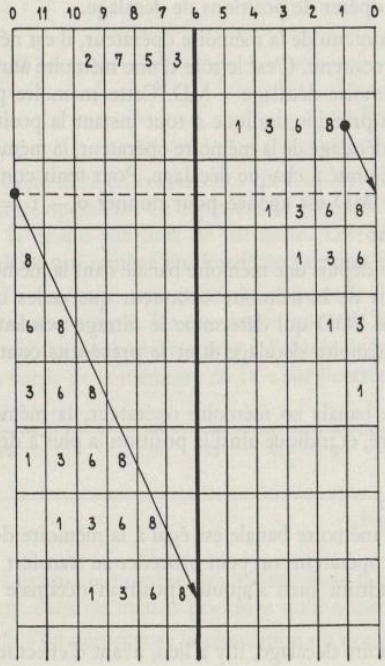
MD = 4 après un décalage.

ME = 3 après deux décalages.

MD = 2 après trois décalages.

Fig. 60

Si, au début de l'opération, le nombre en mémoire opérateur est à droite de celui en mémoire banale, le cadrage s'effectue avec permutation circulaire (*fig.* 61).



Mémoire banale :
OD = 6.

M_1 au début de l'opération :
MD = 1.

Après 1 décalage MD = 0.

— 2 — MD = 11.

— 3 — MD = 10.

— 4 — MD = 9.

— 5 — MD = 8.

— 6 — MD = 7.

— 7 — MD = 6.

Fig. 61

VI. DISTRIBUTION DE CONSTANTES

Indépendamment des introductions cinématique et statique, à partir du distributeur de la machine connectée, il est possible d'introduire des constantes par le programme électronique.

Pour cela, on utilise la mémoire OF comme émettrice des constantes. Cette mémoire, capable de contenir des codes de 1 chiffre affichés au tableau de connexion, peut être mise en rapport avec le canal-donnée ou avec le canal-résultat (*fig.* 62).

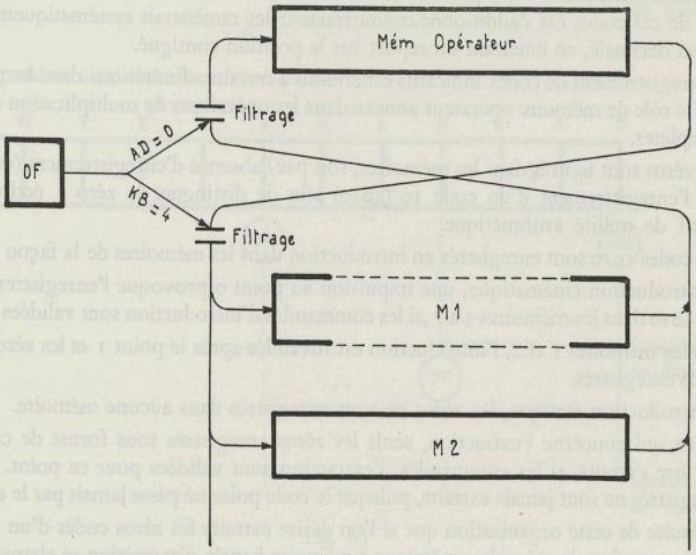


Fig. 62

L'accès de la mémoire OF sur le canal-donnée est conditionné par l'indication d'une adresse, l'adresse 0, qui est celle de la mémoire OF.

L'accès de la mémoire OF sur le canal-résultat est conditionné par indication d'un type d'opération, qui est l'opération KB (code 4).

La mémoire OF ne pouvant enregistrer qu'un seul chiffre décimal, il n'est possible de distribuer qu'un seul chiffre de constante par ligne de programme.

Le chiffre de constante est défini par le code OF qui perd sa signification d'ordre fin. Le filtrage est alors entièrement déterminé par OD, car il ne valide qu'une seule position décimale définie par :

Ordre début : OD.

Ordre fin : OD + 1.

Pratiquement OD indique la position décimale du chiffre de constante que représente OF.

L'adresse 0 supprime le cadrage préalable. En effet, celui-ci ne permettrait d'envoyer en opération CN, AN, SN que des constantes représentant des ordres d'unités égaux à celui le plus à droite de M_1. Sa suppression permet de faire occuper au chiffre de constante une position décimale quelconque, par rapport au nombre auquel il est combiné.

VII. CODIFICATION DES ZÉROS

1. Organisation générale.

Parmi les 16 codes pouvant être enregistrés dans une position de mémoire, on distingue les 10 codes arithmétiques (0 à 9) et 6 codes indicatifs n'ayant pas de signification décimale (10 à 15).

Seules les mémoires 2 à 7 peuvent contenir des codes indicatifs. La mémoire 1 ne peut contenir de ces codes car l'additionneur-soustracteur les ramènerait systématiquement à une expression décimale, en émettant un report sur la position contiguë.

L'enregistrement de codes indicatifs est soumis à certaines limitations dans la mémoire 2, qui joue le rôle de mémoire opérateur annexe dans les opérations de multiplication et de division complètes.

Les zéros sont figurés dans les mémoires, soit par l'absence d'enregistrement (code 0000), soit par l'enregistrement d'un code 10 (1010) afin de distinguer le zéro à écrire du zéro significatif de nullité arithmétique.

Ces codes 1010 sont enregistrés en introduction dans les mémoires de la façon suivante :

A. En introduction cinématique, une impulsion au point 0 provoque l'enregistrement d'un code 1010 dans les mémoires 3 à 7 ,si les commandes d'introduction sont validées à ce point

Dans les mémoires 1 et 2, l'introduction est invalidée après le point 1 et les zéros ne sont jamais enregistrés.

B. En introduction statique, les zéros ne sont enregistrés dans aucune mémoire.

En ce qui concerne l'extraction, seuls les zéros enregistrés sous forme de codes 1010 peuvent être extraits, si les commandes d'extraction sont validées pour ce point. Les zéros non enregistrés ne sont jamais extraits, puisque le code point ne passe jamais par le code 0000.

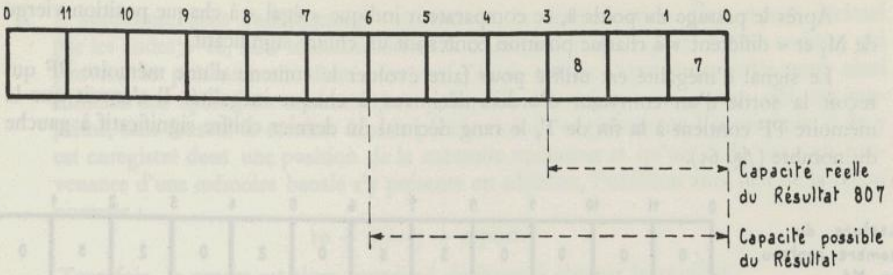
Il résulte de cette organisation que si l'on désire extraire les zéros codés d'un résultat, il faut que le transfert de mémoire opérateur à mémoire banale d'extraction se charge de coder les zéros à la place des zéros de nullité arithmétique contenus en mémoire opérateur. Il faut que, le long du canal résultat, se trouve une possibilité d'injection de codes 1010 entre les chiffres significatifs en provenance de la mémoire opérateur.

D'autre part, il est nécessaire que les zéros codés introduits, soit de cette façon dans une mémoire banale, soit par le jeu de l'introduction cinématique, soient supprimés lorsqu'un résultat intermédiaire ou une donnée sont transférés de mémoire banale en mémoire opérateur. Il faut donc, le long du canal-donnée, un organe capable de supprimer les zéros codés ainsi que les codes indicatifs qui pourraient se trouver dans la zone transférée.

2. Principe de la codification des zéros.

C'est lors de l'opération OB que se fait cette codification. Il s'agit de remplacer les positions d'enregistrement nul de la mémoire opérateur par un code 1010 à envoyer dans la mémoire banale. Le problème serait très facile si l'on ne devait pas coder uniquement les zéros à droite des chiffres significatifs et non pas ceux qui se trouvent à gauche.

Soit en mémoire opérateur un nombre pouvant atteindre 6 chiffres (*fig.* 53). On désire le transférer en mémoire banale 3 par une opération OB. Supposons que le nombre soit égal à 807. Il se trouvera ainsi dans la mémoire opérateur :



Il devra apparaître ainsi dans la mémoire banale :

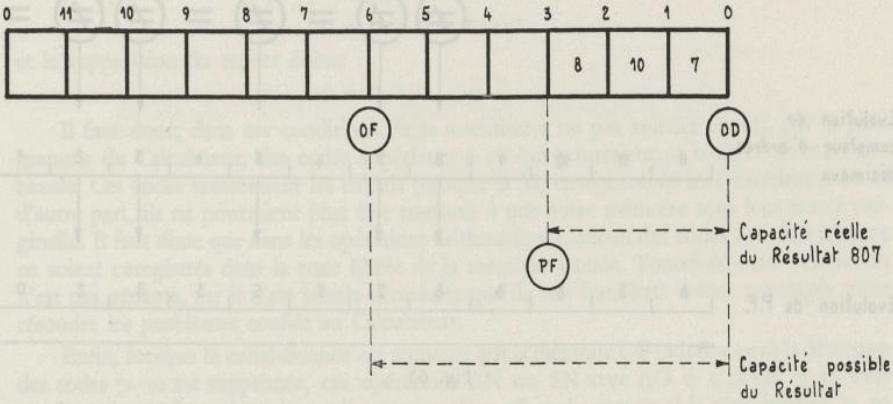


Fig. 63

La capacité possible du résultat à transférer est définie par OD = 0 et OF = 6. La capacité réelle de 807 devrait être définie par OD = 0 et OF = 3. Si on pouvait substituer à l'OF = 6 affiché un OF = 3 quand le résultat a 3 positions, un OF = 4 quand il a 4 positions, etc., il suffirait alors de procéder à une injection pure et simple de codes 1010 dans les positions nulles, le long du canal résultat.

Il faut donc pouvoir « mesurer » la capacité réelle de chaque nombre dans la mémoire opérateur.

Cette mensuration se fait de la façon suivante : lors de l'effacement par T_2 du contenu de la mémoire banale qui va recevoir le transfert, un comparateur à « 0000 » reçoit successivement les positions contenues dans la mémoire opérateur (*fig.* 64).

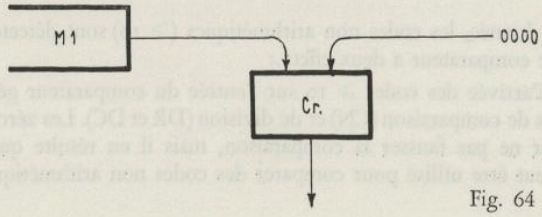


Fig. 64

Après le passage du poids 8, ce comparateur indique « égal » à chaque position vierge de M_1 et « différent » à chaque position contenant un chiffre significatif.

Le signal d'inégalité est utilisé pour faire évoluer le contenu d'une mémoire PF qui reçoit la sortie d'un compteur d'ordres décimaux à chaque inégalité. Il s'ensuit que la mémoire PF contient à la fin de T_2 le rang décimal du dernier chiffre significatif à gauche du nombre (*fig.* 65).

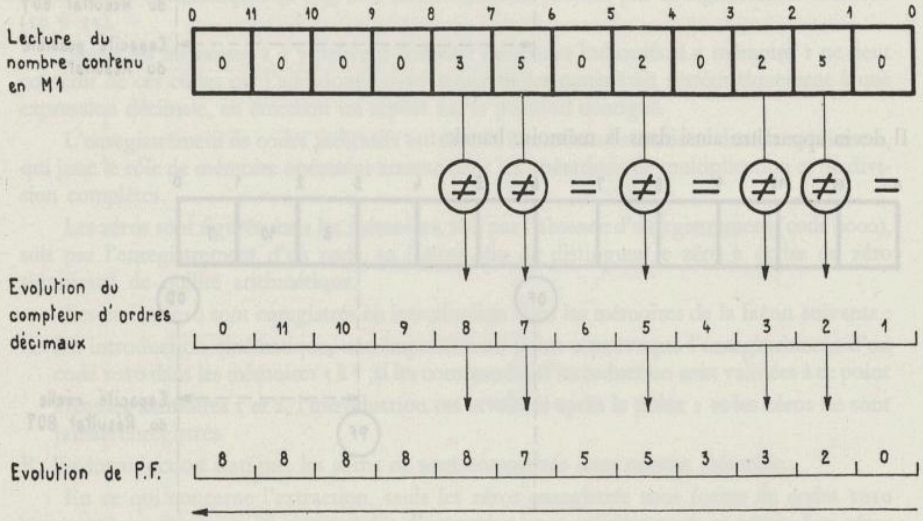


Fig. 65

La mémoire PF est alors transférée sur la mémoire OF en même temps que celle-ci est effacée. Ainsi, dans l'exemple donné plus haut, on a substitué à l'OF = 6 un PF = 3. Le filtrage pendant le transfert est donc validé entre l'OD et le nouvel OF avec injection de code 1010 dans les positions vierges à nouveau repérées par le comparateur.

En résumé, pendant T_2 de l'opération OB, on efface, entre OD et OF définissant la capacité possible du résultat, la mémoire banale qui va recevoir le transfert et on mesure le nombre qui sera transféré. On substitue alors à l'OF affiché le rang du dernier chiffre significatif à gauche du résultat réel contenu en M_1. Pendant T_1, on valide le transfert entre l'OD affiché et le nouvel OF, en continuant de détecter les positions vierges afin d'y injecter des codes 1010.

3. Élimination des zéros codes.

Sur le canal-donnée, les codes non arithmétiques ($\geqslant$ 10) sont détectés par un comparateur spécial. Ce comparateur a deux effets :

A. Il supprime l'arrivée des codes $\geqslant$ 10 sur l'entrée du comparateur général utilisé dans les opérations de comparaison (CN) et de division (DR et DC). Les zéros codés sont ainsi éliminés pour ne pas fausser la comparaison, mais il en résulte que le comparateur général ne peut être utilisé pour comparer des codes non arithmétiques enregistrés en mémoire.

B. Il supprime, dans l'additionneur-soustracteur, la détermination d'un report décimal par les codes $\geqslant$ 10, sans toutefois altérer la transmission des reports sur les codes 9 qui peuvent se présenter simultanément sur l'autre entrée. Les zéros codés sont ainsi éliminés des opérations arithmétiques puisqu'ils déterminent un report, qui est supprimé, sans fausser le résultat dans la position où ils se présentent. Par exemple, si un 7 est enregistré dans une position de la mémoire opérateur et qu'un code 1010 en provenance d'une mémoire banale s'y présente en addition, l'addition aura lieu de la façon normale :

$$10 + 7 = 7 + \text{report.}$$

Toutefois, le report est alors supprimé de façon à obtenir le résultat :

$$10 + 7 = 7.$$

Si la position de la M_1 était vierge, on aura :

$$10 + 0 = 0 + \text{report.}$$

et la suppression du report donne :

$$10 + 0 = 0.$$

Il faut donc, dans ces conditions, faire attention à ne pas amener en M_1, par le programme du Calculateur, des codes supérieurs à 10 qui pourraient se trouver en mémoire banale. Ces codes fausseraient les calculs puisque la M_1 enregistrerait leur excédent à 10 et d'autre part, ils ne pourraient plus être transmis à une autre mémoire sous leur forme originelle. Il faut donc que dans les opérations arithmétiques, aucun des codes 11, 12, 13, 14, 15 ne soient enregistrés dans la zone filtrée de la mémoire banale. Toutefois, cette restriction n'est pas gênante, car il n'est jamais nécessaire que de tels transferts soient envisagés pour résoudre les problèmes confiés au Calculateur.

Enfin, lorsque le canal-donnée est alimenté par la mémoire OF (adresse = 0) la détection des codes $\geqslant$ 10 est supprimée, ces opérations AN ou SN avec AO = 0 et OF $\geqslant$ 10 sont possibles; le résultat est correct et l'opération s'est effectuée comme si le code binaire $\geqslant$ 10 avait été écrit en semi-binaire sur 2 τ.

Chapitre sixième

LES OPÉRATIONS ÉLÉMENTAIRES DU CALCULATEUR GAMMA 3

I. LES OPÉRATIONS SANS TRANSFERT

1. L'opération variante.

V - Code 0.

A. *Définition et caractères généraux.*

On a vu que les instructions affichées au tableau de connexions sont prises en charge successivement par le Calculateur qui les sélectionne à partir du contenu d'une mémoire numéro de ligne NL. Normalement cette mémoire progresse de 0 à 63, unité par unité, et retourne à Z à Z0 à partir de 63. Toutefois, le contenu de la mémoire NL peut évoluer différemment grâce à l'opération « variante ».

La progression spéciale de la mémoire NL peut être provoquée à partir des éventualités suivantes :

a) A partir d'éventualités observées sur la machine connectée. Les variantes prennent alors le nom de « variantes à relais » (VR).

b) A partir d'éventualités observées sur le Calculateur Gamma lui-même. Les variantes prennent alors le nom de « variantes comparaison » (VC).

c) A partir d'une nécessité systématique. Les variantes prennent alors le nom de « variantes systématiques » (VS).

d) A partir de la position de l'arbre de la machine connectée en vue du contrôle du départ des calculs. Elles prennent alors le nom de « variantes point machine » (VPM).

L'instruction TO = 0 est commune à ces quatre catégories de variantes. Pour les distinguer entre elles, il est fait recours aux deux instructions AD et OF. Le type réel d'opération pour une variante définie est donc constitué par l'ensemble de trois instructions TO, AD, OF.

D'autre part, les variantes agissent sur la mémoire NL de telle sorte qu'elles reçoivent le programme non sur la ligne suivant celle sur laquelle elles sont affichées, mais sur une ligne définie par la formule :

$$NL = 4\ OD + \frac{OF}{4}.$$

On voit que les 4 instructions TO, AD, OD, OF sont utilisées à ces deux fins de telle sorte que OF sert aux deux fins (*fig.* 66).

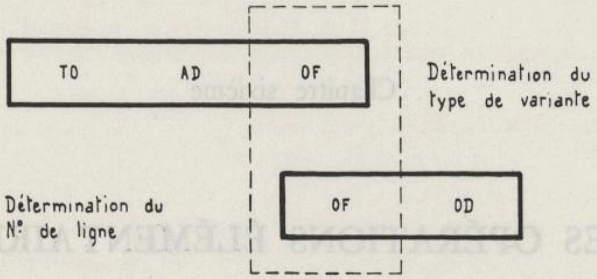


Fig. 66

On voit aussi que si l'on connaît les deux paramètres OD et OF, on peut déterminer le numéro de ligne. Mais en réalité, le problème se pose de la façon inverse : on désire déterminer un OD et un OF pour obtenir tel numéro de ligne. La relation

$$NL = 4\ OD + \frac{OF}{4}$$

est alors indéterminée.

On a recours à un tableau intitulé « tableau de codification Gamma » dont la disposition est en équerre (*fig.* 67).

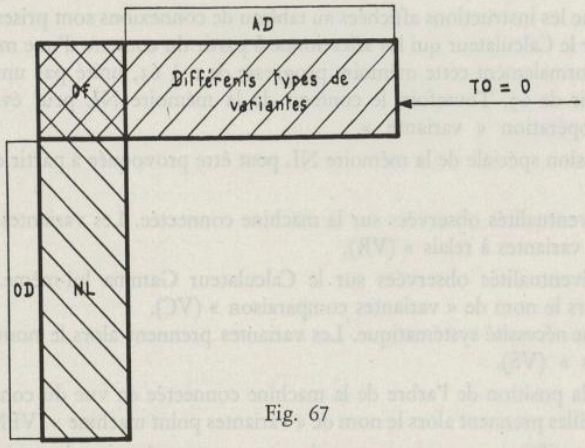


Fig. 67

Les deux branches de l'équerre comportent :

— l'une, sur le thème TO = O, les différents types de variantes et verticalement une graduation en OF, horizontalement une graduation en AD,

— l'autre, les différents numéros de ligne, avec horizontalement une graduation en OF, verticalement une graduation en OD.

Le tableau s'emploie ainsi (*fig.* 68).

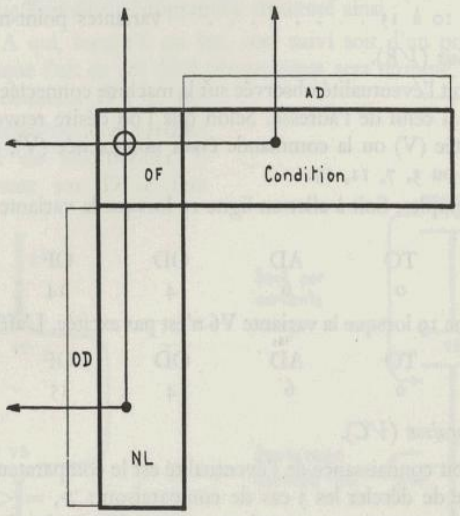


Fig. 68

A partir de la condition de variante choisie, on trace deux axes de coordonnées qui coupent la graduation en AD à l'adresse qui doit être indiquée et la graduation en OF sur quatre OF possibles.

D'autre part, à partir du numéro de la ligne choisie, on trace un deuxième système d'axes qui coupent la graduation en OD sur l'OD à indiquer, et la graduation en OF sur quatre OF possibles.

A l'intersection des deux systèmes d'axes se trouve l'OF à choisir pour définir l'opération.

Les différentes variantes, sur le thème TO = 0, sont définies par les codes OF et AD de telle sorte que la branche horizontale de l'équerre est divisée ainsi (*fig.* 69) :

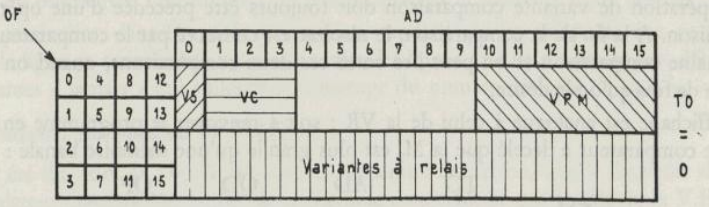


Fig. 69

Les variantes à relais se trouvent dans la région définie par les adresses de 0 à 15 et les OF 2, 6, 10, 14 — 3, 7, 11, 15.

Les autres types de variantes se trouvent dans la région au-dessus, définie par les OF 0, 4, 8, 12 — 1, 5, 9, 13 où elles sont distinguées par l'adresse :

AD = 0 . . . . . variantes systématiques,
AD de 1 à 3 . . . . . variantes comparaison,
AD de 10 à 15 . . . . . variantes point-machine.

B. *Les variantes à relais* (*VR*).

L'organe qui reçoit l'éventualité observée sur la machine connectée est une commande V. Son rang correspond à celui de l'adresse. Selon que l'on désire renvoyer le programme la commande étant excitée (V) ou la commande étant non-excitée ($\overline{V}$), on choisit des OF de la forme 2, 6, 10, 14 ou 3, 7, 11, 15.

Prenons deux exemples. Soit à aller en ligne 19 lorsque la variante V_6 est excitée. L'affichage sera le suivant :

TO	AD	OD	OF
0	6	4	14

Soit à aller en ligne 19 lorsque la variante V6 n'est pas excitée. L'affichage sera le suivant :

TO	AD	OD	OF
0	6	4	15

C. *Les variantes comparaison* (*VC*).

L'organe qui reçoit connaissance de l'éventualité est le comparateur général du Gamma. Cet organe est capable de déceler les 3 cas de comparaison : >, = <. La variante comparaison sert à lire le comparateur, en normal ou en complémentaire. Les 6 cas obtenus sont définis sur le tableau comme suit :

				0	1	2	3
0	4	8	12		>	=	⩾
1	5	9	13		⩽	≠	<
2	6	10	14				
3	7	11	15				

L'opération de variante comparaison doit toujours être précédée d'une opération de comparaison. A la fin de la comparaison, le résultat est conservé par le comparateur jusqu'à la prochaine comparaison et on peut lire entre ces deux comparaisons, quand on le désire et autant de fois qu'on le désire.

L'affichage est analogue à celui de la VR : soit à renvoyer le programme en ligne 19 quand le comparateur a décelé que la M_1 est plus grande qu'une mémoire banale :

	TO	AD	OD	OF
CN	9			
. . .	. . .	. . .	. . .	. . .
. . .	. . .	. . .	. . .	. . .
. . .	. . .	. . .	. . .	. . .
VC	0	1	4	12

Si la comparaison a donné un résultat différent de celui défini par AD et OF la succession normale des lignes est maintenue.

D. *Les variantes systématiques* (*VS*).

Il est fréquent que l'on ait un programme structuré ainsi :

Un programme A qui, lorsqu'il est fini, sera suivi soit d'un programme B soit d'un programme C et lorsque l'un de ces deux programmes sera terminé, regroupement sur un second programme commun D (*fig.* 70).

Une variante à relais ou une variante comparaison permet de sélectionner B et C. Le regroupement sur D se fera

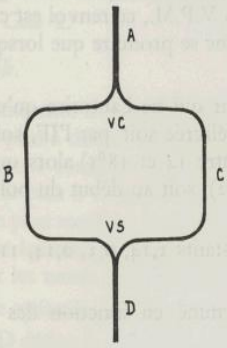


Fig. 70

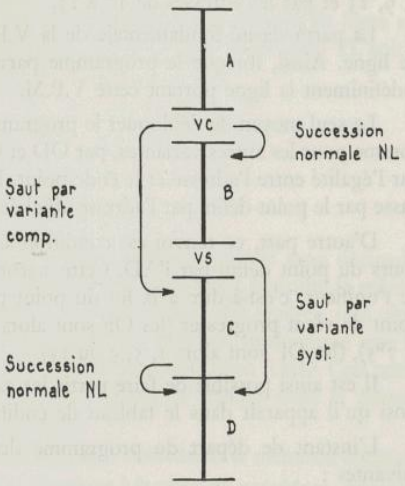


Fig. 71

grâce à une variante systématique « toujours ». En effet, les programmes étant constitués de successions normales de lignes, on devra mettre les sous-programmes A B C D à la suite les uns des autres et accéder soit par « saut » contrôlé par variante de l'un à l'autre, soit par succession normale des numéros de ligne (*fig.* 71).

Lorsque AD = 0 et OF = 0 (OF = 4 — 8 ou 12, n'ayant alors aucun sens ainsi que OD), le programme passe normalement à la ligne suivante (variante « jamais »). La variante « jamais » sert en particulier dans certains cas d'analyse de signe. Elle est également prévue pour le cas des lignes inutilisées et ne recevant aucune connexion. Pour ces lignes, les codes TO, AD, OD et OF sont nuls et déterminent une variante « jamais » qui permet au programme de poursuivre l'exploration du tableau jusqu'à la première ligne utilisée.

En particulier, dans le cas de la machine standard qui ne comporte que 32 lignes au tableau, la ligne 31 renvoie à la ligne 0. En effet, les lignes 32 à 63 n'existant pas, déterminent des variantes « jamais » qui valident le comptage du numéro de ligne de 32 à 64 = 0.

E. *Les variantes point-machine* (*V.P.M.*).

Il a été dit, lors de l'étude des circuits généraux de liaison, que le contrôle du départ du Calculateur Gamma était le fait de la machine connectée et que le rôle de la V.P.M. était de régler le Calculateur pour qu'il parte au point du cycle choisi lorsqu'il apparaît sur la machine connectée. C'est le premier rôle de la V.P.M.

Elle en a un second : la succession des lignes de programme, par comptage ou par variante, est automatique. Elle est uniquement contrôlée par le temps nécessaire pour exé-

cuter les instructions affichées sur ces lignes. Ainsi le comptage faisant normalement passer de la ligne 63 à la ligne 0, les calculs faisant l'objet du programme seraient répétés plusieurs centaines de fois par seconde s'il n'était pas prévu un blocage. C'est le second rôle de la V.P.M. qui peut ainsi se définir : la V.P.M. est une opération qui, à partir d'un signal de position de l'arbre de la machine connectée, déclenche un programme et un seul à un instant que l'on peut choisir.

La région des V.P.M. sur le tableau de codification est définie par les OF 0, 4, 8, 12 — 1, 5, 9, 13 et par les adresses de 10 à 15.

La particularité fondamentale de la V.P.M. est de supprimer le comptage du numéro de ligne. Ainsi, lorsque le programme parvient sur une V.P.M., il se bloque en répétant indéfiniment la ligne portant cette V.P.M.

Le seul moyen de débloquer le programme est alors de le renvoyer à une ligne définie, comme pour les autres variantes, par OD et OF. Dans les V.P.M., ce renvoi est conditionné par l'égalité entre l'adresse et le code-point. Il ne peut donc se produire que lorsque le cycle passe par le point défini par l'adresse affichée à la V.P.M.

D'autre part, ce renvoi est conditionné par l'unifieur qui ne l'autorise qu'une fois au cours du point défini par l'AD. Cette autorisation est délivrée soit par l'IF, soit par l'ID de l'unifieur, c'est-à-dire à la fin du point précédent (entre 14 et 18^05) alors que le code-point de vient progresser (les OF sont alors 0, 4, 8 ou 12), soit au début du point (entre 3 et 7^05), (les OF sont alors 1, 5, 9 ou 13).

Il est ainsi possible de faire partir les calculs aux instants 1,14, 0,3, 0,14, 11,3... 14,14, ainsi qu'il apparaît dans le tableau de codification.

L'instant de départ du programme doit être déterminé en fonction des remarques suivantes :

— Une valeur arithmétique introduite en cinématique peut être utilisée par le programme à partir de 1,14 afin d'attendre l'enregistrement des 1. Il est inutile d'enregistrer les zéros qui sont éliminés par le calcul.

— La consultation d'une commande V ou S par le programme (VR ou sélection d'instructions) ne doit être autorisée qu'à partir de 1,14, 0,14, 11,14, ou 12,14 suivant que l'impulsion d'appel parvient au plus tard à 1, 0, 11 ou 12.

— Une introduction statique ne doit être exécutée par le programme qu'à partir de 12,3.

— Une mémoire qui enregistre en introduction cinématique ne peut être réutilisée par le programme (ZB - KB - OB) qu'à partir de 1,14 ou 0,14 suivant que l'introduction est validée de 9 à 1 ou de 9 à 0.

— Une position de mémoire utilisée en extraction ne peut être réutilisée par le programme (ZB - KB - OB) qu'à partir de 1,14, 0,3, 11,3, suivant que l'extraction de cette position est validée, par la commande générale ou par sélection individuelle, pour les points 1, 0 ou 11.

Ces conditions de départ ne sont pas valables pour les positions de mémoires qui sont utilisées pour extraire des codes de 11 à 15 *avant* l'extraction des codes 9 à 0.

Si, par exemple, on désire perforer un 11 lorsqu'un résultat est négatif, l'effacement du signe du résultat précédent et l'enregistrement du signe du résultat à extraire doivent se faire après 11,3.

Si, par contre, on désire ventiler l'extraction du résultat en deux zones « débit » et « crédit » à l'aide d'une commande de sélection de la machine connectée, cette commande doit être appelée par une impulsion 11 (ou 12) apparaissant au cycle précédent ; dans ce cas,

l'effacement du signe du résultat précédent et l'enregistrement du signe du résultat à extraire doivent se faire avant 0,14 (ou 11,14), le calcul de détermination du signe se plaçant alors entre 1,14 et 0,14 (ou 1,14 et 11,14).

Si la ventilation est commandée par une commande de sortie, le signe peut être extrait aux points 13,14 ou 15.

F. *Séquence des trains.*

Les opérations variante ne consistent qu'en un changement du numéro de ligne après une introduction des instructions : il suffit donc d'un train Tp suivi d'un train Tl, soit une durée totale de 3 tours.

2. Altération mémoire décalage.

AMD - *Code* 2.

A. *Rôle.*

La mémoire décalage a pour fonction de contenir en permanence le rang décimal du chiffre le plus à droite enregistré en mémoire opérateur. Le Calculateur consulte la mémoire décalage pour connaître la position de ce nombre. Si l'on désire que soit repérée, non plus la position la plus à droite effectivement enregistrée, mais une position quelconque que l'on désire, on peut modifier le contenu de la mémoire décalage par l'opération AMD. Ceci permet de définir la mémoire décalage comme l'organe qui permet de repérer dans la mémoire opérateur les unités du nombre que l'on désire traiter.

Cette opération n'altère, ni en valeur, ni en position, le contenu de la mémoire opérateur.

L'OD définit le code à transférer en mémoire décalage à la place du précédent. AD et OF ne sont pas utilisés dans cette opération.

B. *Séquence des trains.*

L'opération comporte simplement une introduction de programme (transfert de OD dans MD), donc elle ne requiert qu'un train Tp suivi d'un train Tl, soit une durée totale de 3 tours.

C. *Utilisation.*

Cette opération joue un rôle préparatoire aux opérations où la mémoire décalage intervient dans la détermination du nombre de décalages.

Ces opérations sont :

a) Les opérations avec cadrage préalable (OB, CN, AN, SN). La MD indique à tout instant le rang de l'ordre décimal le plus à droite du nombre contenu en M_1, donc le rang de la position de M_1 à mettre en correspondance avec la position de la mémoire banale définie par OD. On peut être amené à vouloir modifier cette correspondance dans deux cas :

α) Négliger des décimales consiste, en effet, à altérer la MD de façon à ce qu'elle contienne le rang du premier chiffre à droite que l'on veut considérer. L'opération AMD ne modifiant pas le contenu de la mémoire opérateur, les décimales négligées sont conservées dans la mémoire opérateur et peuvent servir dans un calcul qui suit (*fig.* 72).

β) Enregistrement de deux nombres distincts en M_1. Bien que, théoriquement, la mémoire opérateur ne soit pas splittable, on peut éventuellement y enregistrer deux nombres. Lorsqu'après avoir traité un des nombres, on désire traiter l'autre, il faut modifier la MD pour donner son début (*fig.* 73).

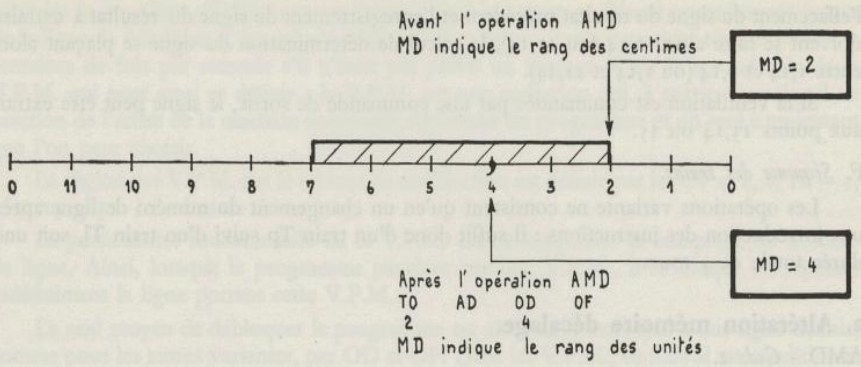


Fig. 72

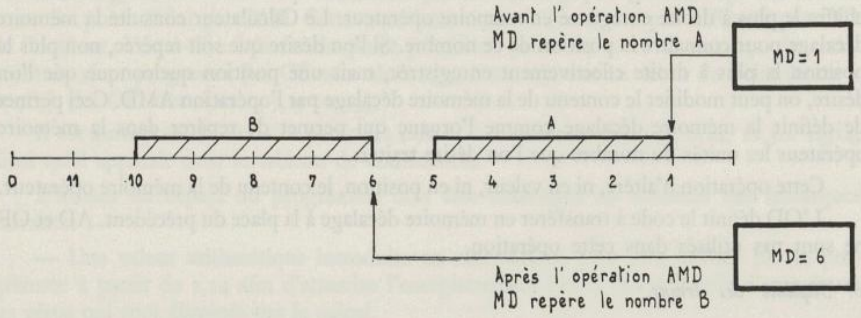


Fig. 73

Il faut bien faire attention que lorsque la MD = 1 repère le nombre A, en fait, elle repère A + 10^n B où *n* est égal à la différence entre les positions début de A et B. Il convient donc de tenir compte de ce fait pour que le nombre B n'intervienne pas afin de n'être pas altéré dans le calcul sur A. Ainsi, si A intervient en addition, on laissera, si la capacité peut augmenter de ce fait, une position vierge entre A et B.

b) Les opérations récurrentes. Dans les opérations de multiplication et division complètes, le nombre de décalages est déterminé par la valeur que l'opération précédente a laissé dans MD. Dans tous les cas où cette valeur n'est pas convenable, il y a lieu de positionner la MD au nombre de décalages voulus. L'opération AMD peut servir à ce résultat.

3. Effacement mémoire banale.

ZB - *Code* 3.

A. *Rôle.*

Cette opération permet d'effacer totalement ou partiellement une mémoire banale. L'adresse définit la mémoire à effacer. L'OD et l'OF définissent les positions à effacer en cas d'effacement partiel.

B. *Séquence des trains.*

Après les trains Tp et Tl, démarre un train T_2 affecté à l'effacement des mémoires banales. Durée totale : 4 tours.

C. *Utilisation.*

Toute introduction dans une mémoire, du fait du programme du Gamma, assure automatiquement l'effacement préalable des positions devant recevoir le transfert. Seule l'introduction cinématique est incapable de provoquer cet effacement préalable. Il faut donc, avant de procéder à une introduction cinématique, effectuer l'effacement préalable par le moyen de l'opération ZB. Cette opération prend donc sa place habituelle en fin de programme. Il est très rare qu'on ait à l'utiliser en cours de programme et il est bon, si on en envisage une dans ces conditions, de considérer que la plupart du temps son inclusion au cours du programme pourrait être évitée en adoptant un programme plus logique.

L'opération ZB peut être utilisée avec l'adresse 1. La M_1 est alors utilisée comme mémoire banale. C'est pourquoi le maintien de MD à son contenu précédent est assuré en opération ZB.

II. LES OPÉRATIONS AVEC TRANSFERT

1. Introduction de constantes.

KB - *Code* 4.

A. *Rôle.*

L'opération KB permet de donner accès à la mémoire OF sur les mémoires banales par le canal [redacted] résultats

Cette opération permet donc d'introduire un code compris entre 0 et 15 dans une position de mémoire, le précédent contenu de cette position étant automatiquement effacé.

La mémoire émettrice étant la mémoire OF, la valeur à transférer est définie par la valeur de l'OF affiché. La mémoire réceptrice est définie par l'adresse. La position de mémoire banale dans laquelle on transfère est définie par l'OD.

La mémoire 1 peut recevoir un code en KB. Elle est, dans cette opération, considérée comme banale, donc seule la position définie par OD est effacée et cet OD n'est jamais introduit en MD qui se maintient à son précédent contenu. D'autre part, l'OF ne doit jamais être supérieur à 9, la mémoire opérateur ne pouvant pas enregistrer des codes non arithmétiques.

B. *Séquence des trains.*

Après les trains Tp et Tl démarrent simultanément les trains T_2 d'effacement de mémoire banale et T_1 de transfert. Durée totale : 4 tours.

C. *Utilisation.*

Cette opération est utilisée pour :

a) Introduire des constantes de calcul, surtout lorsqu'elles comportent peu de chiffres. En effet, il faut une ligne de KB pour chaque code introduit. Pour les constantes d'un seul chiffre, il est préférable d'utiliser l'AD = 0 qui permet d'effectuer le calcul et l'introduction de la constante en une seule ligne.

b) Transmettre à la machine connectée le résultat d'une comparaison. On place alors une KB dans une séquence de programme conditionnée par une variante comparaison.

L'extraction du code introduit par la KB atteste de l'exécution de la séquence, donc du résultat de la comparaison (sens d'un solde, etc.).

Le choix de la constante permet de choisir le point du cycle auquel le test ainsi constitué sera extrait. Il faut, à ce sujet, bien étudier à quel moment par rapport à ce point sera constitué éventuellement le test au cours du programme.

Si le test est constitué avant son point d'extraction, il sera extrait au cours de la fin du programme, avant l'extraction des résultats (*fig.* 74). Il pourra alors servir, par une commande

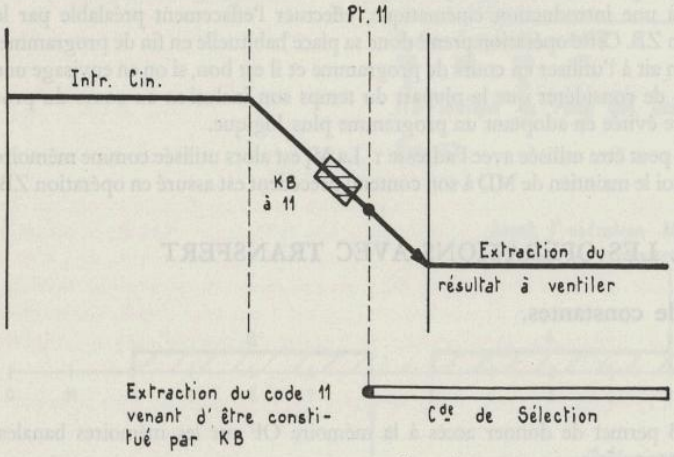


Fig. 74

CS ou de sélection, à faire une ventilation sur les organes récepteurs de la machine connectée. L'opération KB d'introduction de ce code dans la mémoire à extraire doit être terminée au moins 4⁰ avant le point à extraire (ex. 11,20 pour « 12 »).

Si le test est constitué après son point d'extraction, il sera extrait au cycle suivant, après l'extraction des résultats (*fig.* 75). Il ne pourra pas servir à les sélectionner, mais à imprimer ou perforer le code indicateur (si c'est un 11 ou un 12).

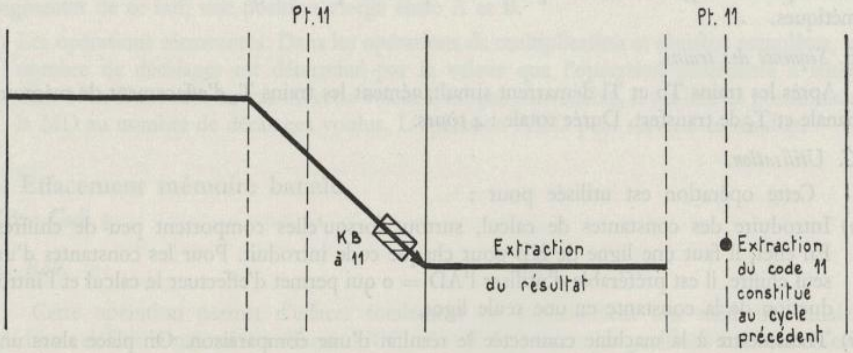


Fig. 75

4. Introduction statique.

IS - *Code* 5.

A. *Rôle.*

L'opération IS valide le transfert effectif dans une mémoire de données contenues dans un totalisateur ou composées au distributeur. Ce transfert doit être préparé au tableau de la machine connectée ainsi qu'il a été dit.

L'adresse définit le numéro de la mémoire banale sur laquelle on introduit. Elle ne permet donc d'introduire que dans une seule mémoire, soit 12 chiffres au maximum par ligne de programme. Elle est conditionnée par le filtrage, et il n'y a pas lieu de prévoir le splittage des positions d'entrée.

Le contenu précédent de la mémoire est automatiquement effacé dans la zone définie par le filtrage.

L'introduction statique peut être commandée au profit de la mémoire opérateur. La mémoire décalage prend alors la valeur de OD, afin de permettre de repérer la position du nombre transféré en M_1.

B. *Séquence des trains.*

Après les trains Tp et Tl, 8 trains T_1 de transfert sont démarrés. Simultanément au premier train T_1 est lancé un train T_2 d'effacement de la mémoire. La durée totale est donc de 11 tours. Cette durée de 11 tours est nécessitée par le fait que bien qu'il s'agisse de garnir une mémoire banale, qu'un seul train T_1 pourrait contrôler, il faut allonger la durée des impulsions définitrices des nombres qui cheminent dans les circuits électriques de la machine connectée.

C. *Utilisation.*

L'opération IS devant faire l'objet d'une préparation de 0,18 à 13,22 sur le tableau de la machine connectée, et les émissions IS ne durant que de 11,22 à 13,22, il y a lieu de s'assurer que la ligne ou les lignes de programme qui portent des IS auront bien lieu entre 12 et 13,20.

Comme on n'est généralement pas sûr de cette position dans le temps de l'IS affichée, il y a lieu de n'autoriser le départ du programme qu'à 12,3 ou, dans le cas où l'on désire profiter au maximum du temps de calcul disponible, d'afficher, après une première séquence de calcul, une VPM validée à 12,3 qui précède immédiatement la première IS.

5. Transfert d'une mémoire banale en mémoire opérateur.

BO - *Code* 6.

A. *Rôle.*

Transférer dans la mémoire opérateur un nombre contenu en mémoire banale. La mémoire opérateur est préalablement effacée dans sa totalité. La mémoire banale est inaltérée.

L'adresse désigne la mémoire banale que l'on transfère.

L'OD et l'OF définissent la place dans cette mémoire du nombre à transférer. L'ordre début affiché est transféré en mémoire décalage.

B. *Séquence des trains.*

Après les trains Tp et Tl démarre un train T_1 de transfert qui assure l'effacement total de la mémoire opérateur. Durée totale : 4 tours.

C. *Utilisation.*

a) Transfert d'un nombre en M_1 depuis MB. C'est l'utilisation classique décrite au paragraphe « rôle » ;

b) *BO avec AD* = 1. Dans ce cas, la mémoire émettrice est la M_1 considérée comme banale. La mémoire réceptrice est également M_1 considérée comme mémoire opérateur.

Le train T_1 remet donc à zéro toute la mémoire opérateur et assure le transfert du nombre défini par le filtrage.

Il en résulte un maintien filtré de M_1 (*fig.* 76).

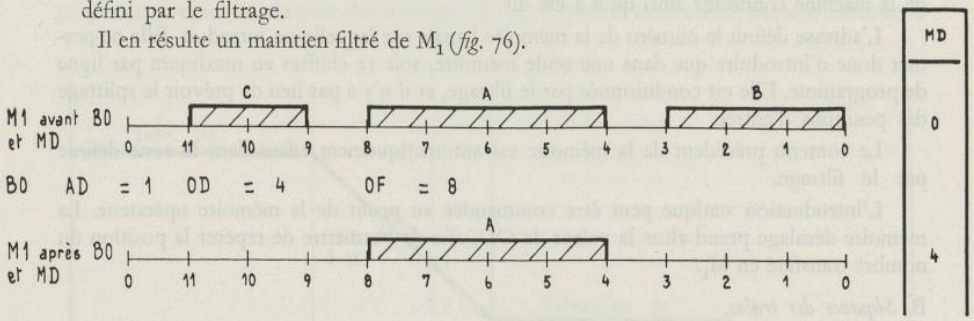


Fig. 76

c) *BO* avec *AD* = 0. Dans ce cas, la mémoire d'origine est OF et l'opération permet d'introduire une constante d'un chiffre défini par OF dans une position de M_1 définie par OD. Les autres positions de M_1 *sont effacées* et l'OD est transféré en MD, ce qui distingue cette opération d'une KB avec AD = 1.

d) *BO avec AD* = 0 et *OF* = 0. La M_1 est totalement effacée et la MD est positionnée à la valeur de l'OD. Cette opération équivaut donc à une ZB avec AD = 1 suivie d'une AMD. Elle est souvent utilisée pour effacer la M_1 en fin de programme et positionner la MD en vue du début du programme suivant. Elle est également utilisée avant une MC, pour libérer la M_1 en vue de la constitution du produit et déterminer le nombre de décalages.

III. LES OPÉRATIONS AVEC CADRAGE PRÉALABLE

1. Transfert de la mémoire opérateur sur mémoire banale.

OB - *Code* 8.

A. *Rôle.*

Cette opération permet de transférer dans une mémoire banale un nombre contenu dans la mémoire opérateur. L'adresse définit le numéro de la mémoire banale. OD et OF définissent les positions de cette mémoire qui sont affectées à l'enregistrement du nombre transféré. Le précédent contenu de ces positions est automatiquement effacé.

Le transfert est précédé éventuellement d'un cadrage de la mémoire opérateur afin de placer le nombre à transférer dans les mêmes positions de cette mémoire que celles qui lui sont réservées dans la mémoire banale. La MD contient donc toujours à la fin de l'opération l'OD affiché. La mémoire opérateur n'est pas altérée en valeur, mais peut l'être en position par le cadrage.

B. *Séquence des trains.*

Après les trains Tp et Tl démarrent un ou plusieurs trains T_3 de décalage, si le contenu de la MD est différente de l'OD affiché. Dès l'égalité de MD et de OD, on démarre un train T_2

d'effacement de la mémoire banale au cours duquel on repère le poids fort en vue de la codification des zéros. A la suite de T_2, démarre un train T_1, de transfert. Durée totale : 5 tours plus les tours de cadrage.

C. *Utilisation.*

a) Cette opération sert à mettre en réserve dans une mémoire banale une quantité qui vient d'être calculée, afin de pouvoir en disposer lors d'un calcul ou lors de l'extraction ;

b) Il est intéressant, bien que cette utilisation n'ait pas une grande utilité pratique, d'examiner les propriétés de cette opération avec l'AD = 1. Cette utilisation est, en effet, une bonne illustration du fonctionnement de la mémoire opérateur.

Soit deux nombres A et B dans la mémoire opérateur (*fig.* 77).

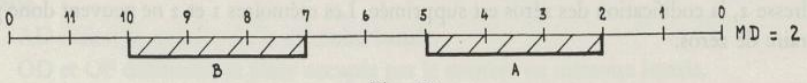


Fig. 77

La mémoire décalage est à 2. On affiche une opération OB avec les codes :

TO	AD	OD	OF
8	1	0	3

Après les trains T_3 de cadrage le contenu de M_1 se présente ainsi que le montre la figure 78.

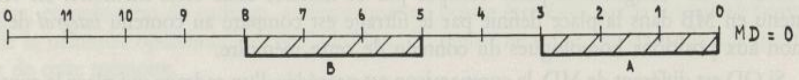


Fig. 78

Puis le train T_2 efface A de 0 à 3 (*fig.*79).

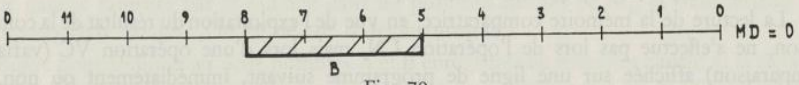


Fig. 79

Le train T_1 transfère par le canal-résultat le contenu nul de 0 à 3, tandis que B est retransmis par la boucle de M_1 (*fig.* 80).

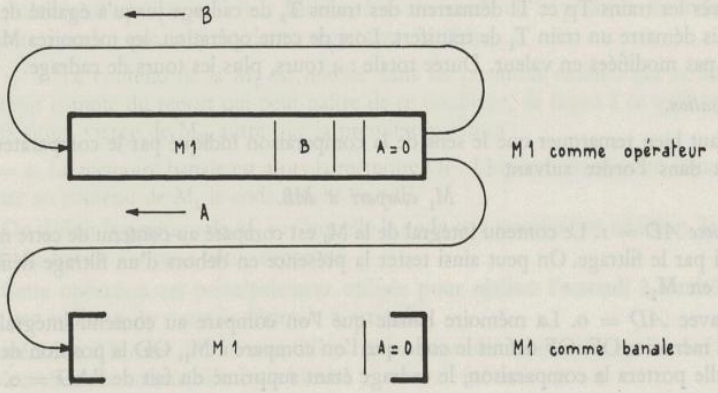


Fig. 80

D'où le contenu (*fig.* 81) :

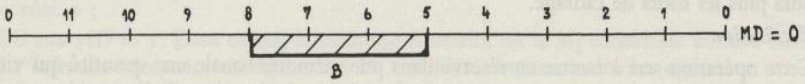


Fig. 81

Cette opération a permis de décaler B en effaçant A mais la mémoire décalage n'indique pas le début de B.

Il faut remarquer que lorsqu'une opération OB est commandée avec l'adresse 1 ou l'adresse 2, la codification des zéros est supprimée. Les mémoires 1 et 2 ne peuvent donc pas extraire de zéros.

2. Comparaison.

CN - *Code* 9.

A. *Rôle.*

Comparer le nombre contenu en M_1 à un nombre contenu dans une mémoire banale.

AD désigne la mémoire banale.

OD et OF désignent la place occupée par ce nombre dans cette mémoire. Le terme contenu en MB dans la place définie par le filtrage est comparé au contenu *intégral* de M_1, et non aux positions homologues du contenu de cette mémoire.

Si OD est différent de MD, la comparaison est précédée d'un cadrage qui place le contenu de M_1 en correspondance avec le terme à comparer en mémoire banale.

Le résultat de la comparaison est enregistré dans une mémoire spéciale du Calculateur qui le conserve jusqu'à ce qu'une nouvelle opération CN soit effectuée.

La lecture de la mémoire comparatrice, en vue de l'exploitation du résultat de la comparaison, ne s'effectue pas lors de l'opération CN, mais lors d'une opération VC (variante comparaison) affichée sur une ligne de programme suivant, immédiatement ou non, la ligne portant la CN. Il est évident que si l'on peut surseoir à l'exploitation d'une CN, il importe de procéder à la VC avant une nouvelle comparaison.

B. *Séquence des trains.*

Après les trains Tp et Tl démarrent des trains T_3 de cadrage jusqu'à égalité de MD et OD. Puis démarre un train T_1 de transfert. Lors de cette opération, les mémoires M_1 et MB ne sont pas modifiées en valeur. Durée totale : 4 tours, plus les tours de cadrage.

C. *Utilisation.*

Il faut bien remarquer que le sens de la comparaison indiqué par le comparateur se lit toujours dans l'ordre suivant :

M_1 comparé à MB.

a) *CN avec AD* = 1. Le contenu intégral de la M_1 est comparé au contenu de cette mémoire défini par le filtrage. On peut ainsi tester la présence en dehors d'un filtrage défini, d'un code en M_1.

b) *CN* avec *AD* = 0. La mémoire banale que l'on compare au contenu intégral de M_1 est la mémoire OF. OF définit le code que l'on compare à M_1, OD la position de M_1 sur laquelle portera la comparaison, le cadrage étant supprimé du fait de l'AD = 0.

Cette opération peut être utilisée pour comparer un nombre contenu en M_1 à une constante, toutes les fois que cette constante ne comporte qu'un seul chiffre significatif, ce qui est fréquent (tranche, plafond, etc.). Dans les autres cas, la constante doit être enregistrée en mémoire banale.

3. Addition.

AN - *Code* 10.

A. *Rôle.*

Additionner au nombre contenu en mémoire opérateur un nombre contenu en mémoire banale.

AD définit le numéro de la mémoire banale.

OD et OF définissent la place occupée par le nombre en mémoire banale.

B. *Séquence des trains.*

Après les trains Tp et Tl démarrent les trains T_3 de cadrage puis, après cadrage, un train T_1 de transfert. Durée totale : 4 tours, plus les tours de cadrage.

C. *Utilisation.*

a) Dans tous les cas d'addition, les termes doivent être placés dans les mémoires de façon que, compte tenu du cadrage, aucun report ne puisse être émis par la douzième position de la mémoire opérateur lors de l'addition. Ce report ne serait pas transmis à la position 1 de cette mémoire.

Par conséquent, le nombre préalablement enregistré en M_1 ne doit pas, *après le cadrage*, atteindre la douzième position, sauf s'il ne peut y avoir augmentation de capacité du fait de l'addition. Dans tous les cas, il ne doit jamais avoir un ou plusieurs de ses chiffres de gauche ramenés par le cadrage dans les positions droites de M_1, la transmission des reports étant coupée. On en déduit la règle suivante :

Règle : pour additionner deux nombres A et B enregistrés en mémoire banale, si la capacité (OFb-ODb) de l'un des nombres est supérieure au complément à 11 de l'OD (ODa) de l'autre nombre A, ce nombre A doit être transféré en premier par la BO et le nombre B en second par l'AN. Dans les autres cas, l'ordre des transferts est indifférent.

Si un nombre en mémoire banale doit s'ajouter à un résultat constitué en M_1, il n'est plus possible de choisir et il faut vérifier, lors du choix de la position du nombre en mémoire banale, que la précédente règle sera vérifiée.

b) $AD = 1$. Le contenu de la M_1 est doublé dans les positions définies par le filtrage. Il faut tenir compte du report qui peut naître de ce doublage, de façon à ce qu'il arrive sur une position vierge de M_1, autre que la première position :

c) $AD = 0$. La mémoire banale est alors la mémoire OF. L'opération consiste donc à additionner au contenu de M_1 le code affiché en OF.

On définit la position de M_1 sur laquelle le code sera transféré en addition, le cadrage étant, du fait de l'adresse zéro, supprimé.

Cette opération est principalement utilisée pour réaliser l'arrondi à l'unité la plus proche, par addition de 5 sur la décimale à arrondir.

L'opération AN avec AD = 0 et OF $\geqslant$ 10 est possible. Le résultat est correct et l'opération s'est effectuée comme si le code binaire $\geqslant$ 10 avait été écrit sur 2 τ.

4. Soustraction.

SN- *Code* 11.

A. *Rôle.*

Retrancher du nombre contenu en M_1 un nombre contenu en MB. Ce résultat, quel que soit son signe, est toujours exprimé en valeur absolue. Le signe de ce résultat n'est pas exprimé en mémoire opérateur, et il y a lieu de l'analyser soit par une CN préalable, soit grâce au dispositif spécial d' « opérateur de signe ».

B. *Séquence des trains.*

Après les trains T_2 de cadrage, démarrent trois trains T_1 de transfert, au cours desquels ont lieu la soustraction, une comparaison interne des termes et le redressement éventuel du résultat. Durée totale : 5 tours, plus les tours de cadrage et le tour de redressement.

C. *Utilisation.*

a) De même que pour AN il y a lieu de déterminer le cadrage des termes de façon que le nombre contenu en M_1 ne dépasse pas la douzième position après cadrage, l'augmentation de capacité n'étant évidemment jamais à prévoir

b) $AD = 1$. On réalise alors une remise à zéro totale ou partielle de M_1.

c) $AD = 0$. L'opération consiste donc à soustraire au contenu de M_1 la constante affichée en OF.

L'opération SN avec $AD = 0$ et $OF \geqslant 10$ est également possible de la même façon que l'AN.

IV. LES OPÉRATIONS RECURRENTES

Le Calculateur Gamma 3 effectue automatiquement, par programme interne, la multiplication et la division qui sont dites opérations récurrentes. La multiplication s'opère en effet par additions successives et la division par soustractions successives. Ces opérations d'addition et de soustraction s'effectuent dans la mémoire opérateur.

Chacune de ces deux opérations peut être réalisée sur le mode réduit ou sur le mode complet. Cette distinction ne porte pas sur le procédé de calcul, identique dans les deux cas, mais sur les capacités mises en œuvre par ce procédé.

La figure 82 montre les capacités pouvant être mises en œuvre :

	Mode réduit	Mode complet
Multiplication.	Produit de 11 chiffres au maximum	Produit de 23 chiffres au maximum
	MR. Code 12	MC. Code 14°
Division	Dividende de 11 chiffres au maximum	Dividende de 23 chiffres au maximum
	DR Code 13	DC Code 15°

Fig. 82

1. Multiplication réduite.

MR - *Code* 12.

A. *Rôle.*

Effectuer la multiplication de deux nombres :

Le multiplicateur est placé dans les positions droites de M_1 ; les positions de M_1 non occupées par lui doivent être effacées avant l'opération. Le multiplicateur est détruit par l'opération.

Le multiplicande est placé dans une mémoire banale et est défini par AD (2 à 7) OD et OF. L'OD doit être au moins égal à la position fin du multiplicateur dans M_1, c'est-à-dire que les facteurs ne doivent pas se chevaucher. L'OF doit être au plus égal à 11. La somme des nombres de chiffres des deux facteurs ne peut donc excéder 11. Le multiplicande reste inaltéré à la fin de l'opération.

Le produit est constitué dans la M_1; son nombre de chiffres est au plus égal à 11.

B. *Principe de l'opération.*

a) *Multiplication par un chiffre de multiplicateur.* La multiplication est réalisée par additions successives. Pour constituer le produit du multiplicande par un chiffre du multiplicateur, il faut additionner le multiplicande à la M_1 autant de fois qu'il y a d'unités dans ce chiffre.

Pour cela, au cours de chaque addition, le Calculateur soustrait une unité au chiffre du multiplicateur, qui est toujours contenu dans la première position à droite de M_1. Le produit est effectué lorsque ce chiffre a été amené à zéro.

Exemple : $236 \times 3 = 708$.

M_1 avant l'opération	3
M_1 après 1 addition 236	2
M_1 après 2 additions 472.	1
M_1 après 3 additions 708.	0

b) *Décalage.* Le chiffre de droite dans M_1 est constamment envoyé à un comparateur (comparateur général) qui détecte son passage à zéro. Le comparateur arrête alors les séquences des trains T_1 qui assurent les additions et fait partir un train T_3.

Ce train provoque un décalage d'une position vers la droite de M_1, comme pour le cadrage; cependant la permutation circulaire ne joue pas, puisque le décalage n'intervient que lorsque la première position à droite est vide.

Le décalage a trois actions sur le contenu de la M_1.

— Il décale les chiffres restant du multiplicateur : après décomptage des unités, le premier décalage amène le chiffre des dizaines à l'extrême droite de M_1; après décomptage des dizaines, le deuxième décalage amène le chiffre des centaines à l'extrême droite de M_1, etc. Les chiffres de multiplicateur sont donc traités successivement dans l'ordre croissant des unités décimales.

Exemple : si le multiplicateur est 213, les positions à droite de M_1 contiennent successivement :

— avant l'opération. . . .	213
— — le 1er décalage . .	210
— après le 1er —	21
— avant le 2e —	20
— après le 2e —	2
— avant le 3e —	0
— après le 3e —	(fin d'opération).

— Il décale le produit partiel constitué à gauche du multiplicateur; ainsi, après le premier décalage, les unités du multiplicande s'ajoutent aux dizaines du produit partiel, tandis que les dizaines de multiplicateur sont décomptées, etc.
— Il substitue, dans les positions rendues libres par le décalage du multiplicateur, les chiffres de droite du produit, au fur et à mesure que ces chiffres sont constitués définitivement. Ainsi, le produit s'étend progressivement au détriment du multiplicateur.

Le comparateur reste en service pendant les trains T_3, afin d'enchaîner les décalages pour les zéros du multiplicateur.

Exemple (*fig.* 83) : soit à multiplier 736 par 4 013.

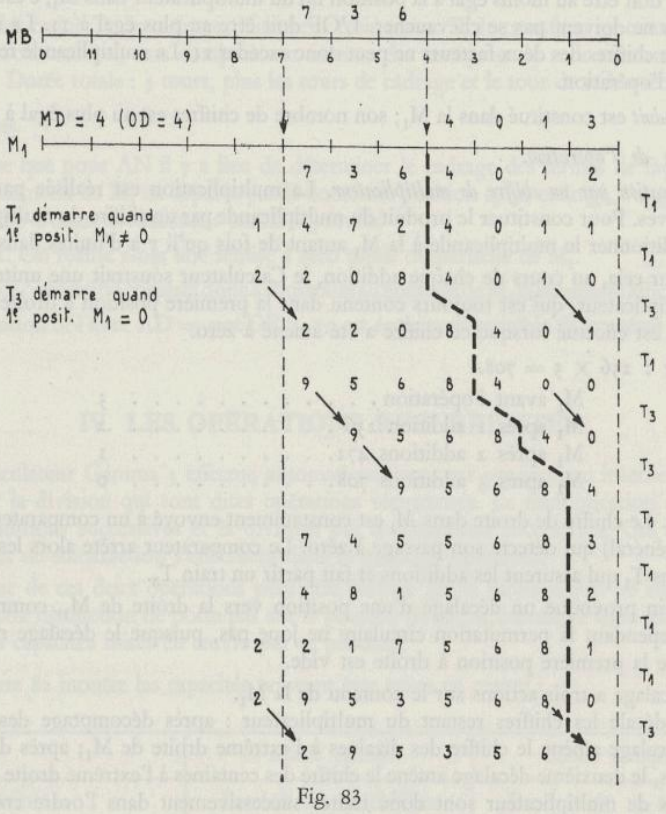


Fig. 83

c) *Fin de l'opération.* L'opération est terminée lorsque le multiplicateur a été entièrement décompté, c'est-à-dire après le décalage qui amène le chiffre des unités du produit à l'extrême droite de M_1. Ce résultat est obtenu lorsque le nombre des décalages effectués est égal à l'OD du multiplicande.

C'est pourquoi la fin d'opération est déterminée par le processus suivant :
—Au début de l'opération (pendant Tl) l'OD est transféré dans la MD dont le précédent contenu est effacé.
— A chaque train T_3, la MD est décomptée d'une unité, comme dans le cadrage.

— Le contenu de MD est constamment envoyé à un comparateur (comparateur de cadrage) qui le compare à zéro.

— Lorsque ce comparateur indique MD = 0, il détermine le signal de fin d'opération provoquant l'exécution de l'opération suivante.

C. *Séquence des trains.*

Après les trains Tp et Tl, la séquence des trains est entièrement déterminée par les deux comparateurs de cadrage et général :

Si MD > 0 et Mr > 0 : départ T_1
Si MD > 0 et Mr = 0 : départ T_2
Si MD = 0 : Fin d'opération.

Durée totale :

Tp et Tl : 3 tours.

Additions : Nombre de tours égal à la somme des chiffres du multiplicateur.

Décalages : Nombre de tours égal à l'OD.

Pratiquement, il faut considérer le cas le plus défavorable où tous les chiffres de multiplicateur sont des 9. La durée totale est alors donnée par la formule :

$$3 + (10 \times OD) \text{ tours.}$$

D. *Utilisation.*

a) *Cadrage de base des termes.* La multiplication réduite utilise deux mémoires. Les termes doivent être cadrés de telle sorte qu'ils ne puissent être confondus lors des transferts. Il s'ensuit que le cadrage de base est analogue à celui de la figure 84.

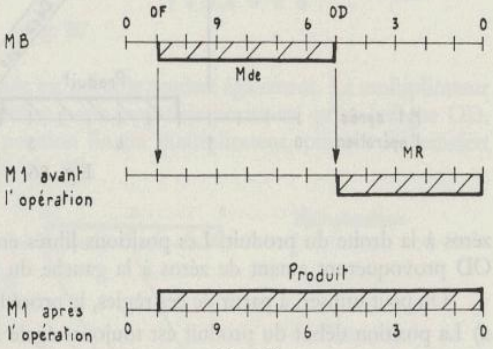


Fig. 84

L'OD du multiplicande est transféré en MD. Il s'ensuit que si le multiplicande est calé en MB au-delà vers la gauche du multiplicateur, l'opération s'effectue comme si le multiplicateur comportait autant de zéros à gauche qu'il y a de positions d'intervalle entre les deux facteurs (*fig.* 85) :

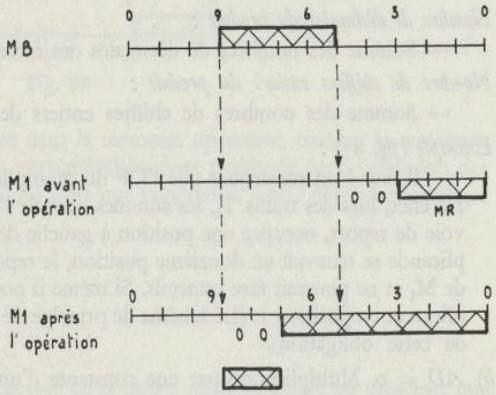


Fig. 85

Le produit comprend ici 7 chiffres significatifs (3 Mr. + 4 Mde = 7 Prod.). Mais il est pratique, pour comprendre le cadrage, de le considérer de 9 chiffres provenant d'un multiplicateur de 5 chiffres (dont 2 zéros à gauche) et d'un multiplicande de 4 chiffres.

D'une façon générale, pour toute étude de cadrage, il y a lieu de se repérer sur le chiffre des

unités du produit. Ce chiffre se constitue, lors du premier T_1 dans la position définie par OD. Puis il est décalé d'autant de positions qu'en indique cet OD. Exemple (*fig.* 86) :

On voit que le cadrage de base est respecté. La façon la plus pratique de procéder pour cadrer le produit est donc la suivante : l'OF du multiplicande définit la position fin du produit en M_1. Le multiplicande est défini par OD et OF. Le multiplicateur est compris entre o et OD. Les positions libres entre O et les unités du multiplicateur provoqueront autant de

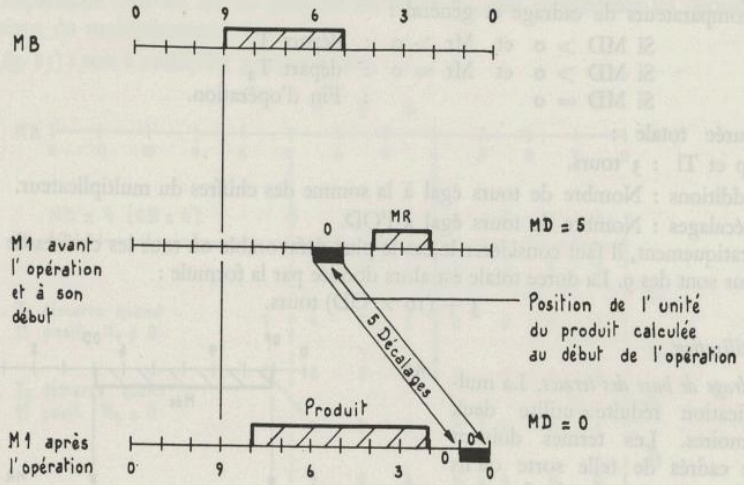


Fig. 86

zéros à la droite du produit. Les positions libres entre les poids forts du multiplicateur et OD provoqueront autant de zéros à la gauche du produit.

On peut utiliser, à partir de ces règles, le procédé suivant qui en découle :

α) La position début du produit est toujours égale à celle du multiplicateur.

β) A partir de cette position, la virgule et la fin du produit sont déterminées par les règles suivantes :

Nombre de décimales du produit :

— Somme des nombres de décimales des facteurs.

Nombre de chiffres entiers du produit :

— Somme des nombres de chiffres entiers des facteurs.

Exemple (*fig.* 87) :

Il faut bien remarquer que l'OF du multiplicande doit être au maximum égal à 11. En effet, lors des trains T_1, les sommes nées de l'addition du multiplicande peuvent, par voie de report, occuper une position à gauche de plus que le multiplicande. Si le multiplicande se trouvait en douzième position, le report serait émis par la douzième position de M_1 et ne pourrait être transmis. Si même il pouvait l'être, il viendrait altérer le multiplicateur en position 1. (Le schéma de principe de l'opération permet de se rendre compte de cette obligation).

b) $AD = 0$. Multiplication par une constante d'un seul chiffre.

La constante intervient en multiplicande; sa valeur est définie par OF.

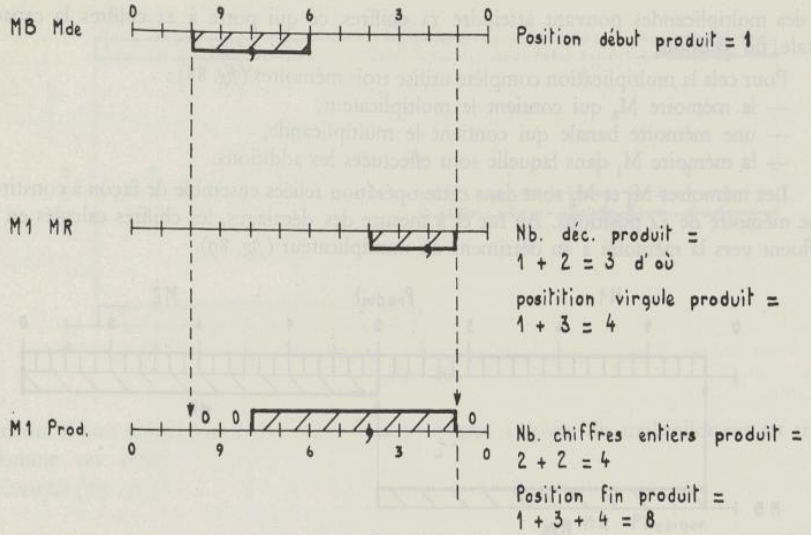


Fig. 87

c) *SI AD* = 0 et *OF* = 0, le multiplicande est nul et le produit également. Le multiplicateur sera donc décompté position par position pour autant de positions qu'en indique OD. Ici OD peut être plus petit que la position fin du multiplicateur, puisque le transfert d'un zéro ne saurait affecter ce facteur (*fig.* 88).

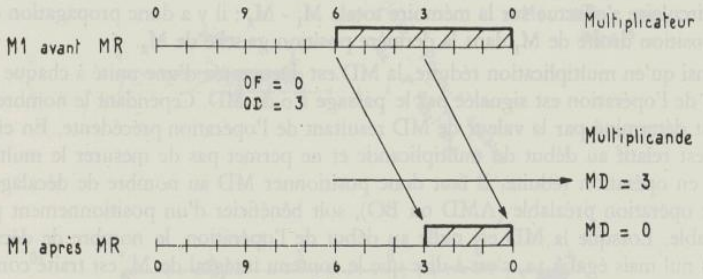


Fig. 88

On peut ainsi décaler un nombre dans la mémoire opérateur, tout en la remettant partiellement à zéro. Cette opération est particulièrement employée dans l'abandon des décimales. Il s'agit bien là d'un abandon, alors que l'opération AMD permet de négliger les décimales en M_1 sans les effacer.

2. Multiplication complète.

MC - *Code* 14.

A. *Rôle*.

La multiplication complète, qui s'effectue de la même manière, quant au principe, que la multiplication réduite, permet de traiter des multiplicateurs pouvant atteindre 12 chiffres

et des multiplicandes pouvant atteindre 11 chiffres, ce qui porte à 23 chiffres la capacité totale du produit.

Pour cela la multiplication complète utilise trois mémoires (*fig.* 88) :

— la mémoire M_2 qui contient le multiplicateur,

— une mémoire banale qui contient le multiplicande,

— la mémoire M_1 dans laquelle sont effectuées les additions.

Les mémoires M_1 et M_2 sont dans cette opération reliées ensemble de façon à constituer une mémoire de 24 positions. Au fur et à mesure des décalages, les chiffres calculés en M_1 refluent vers la mémoire 2 au détriment du multiplicateur (*fig.* 89).

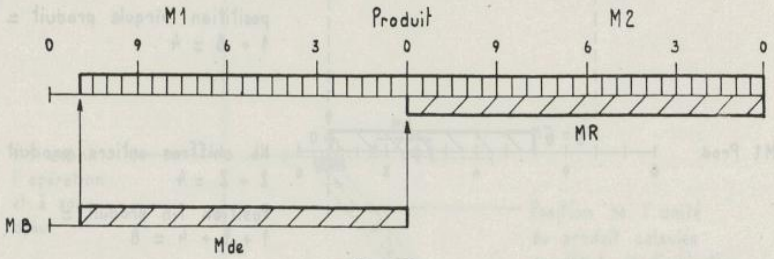


Fig. 89

On voit donc que le thème général est identique à celui de la multiplication réduite en substituant à la seule M_1 la mémoire double M_1 - M_2. Le décomptage du multiplicateur a lieu sur la position d'extrême droite de cette mémoire, soit sur la première position de M_2. Lorsqu'un décalage se produit, il a lieu simultanément dans les deux mémoires et la permutation circulaire s'effectue sur la mémoire totale M_1 - M_2 : il y a donc propagation de la première position droite de M_1 dans la dernière position gauche de M_2.

Ainsi qu'en multiplication réduite, la MD est décomptée d'une unité à chaque décalage, et la fin de l'opération est signalée par le passage à 0 de MD. Cependant le nombre de décalages est déterminé par la valeur de MD résultant de l'opération précédente. En effet, l'OD affiché est relatif au début du multiplicande et ne permet pas de mesurer le multiplicateur comme en opération réduite. Il faut donc positionner MD au nombre de décalages voulu, par une opération préalable (AMD ou BO), soit bénéficier d'un positionnement précédent convenable. Lorsque la MD est nulle au début de l'opération, le nombre de décalages est non pas nul mais égal à 12, c'est-à-dire que le contenu intégral de M_2 est traité comme multiplicateur.

B. *Séquence des trains.*

Elle est identique à celle de la multiplication réduite.

C. *Cadrage de base.*

Les mêmes principes que ceux énoncés pour la MR sont valables en MC (*fig.* 90).

Pour toute recherche de positionnement du produit on définira d'abord la position des unités du produit. Elle est définie au début de l'opération par l'OD du multiplicande. Puis, on étudiera sa position après opération : elle sera décalée d'autant de positions par rapport à celle d'origine que MD a indiqué de décalages. Ainsi, MD définit toujours la capacité du multiplicateur que traite la machine, même s'il ne correspond pas au multiplicateur physique enregistré. Ayant déterminé la position des unités du produit on calculera sa capacité. Là

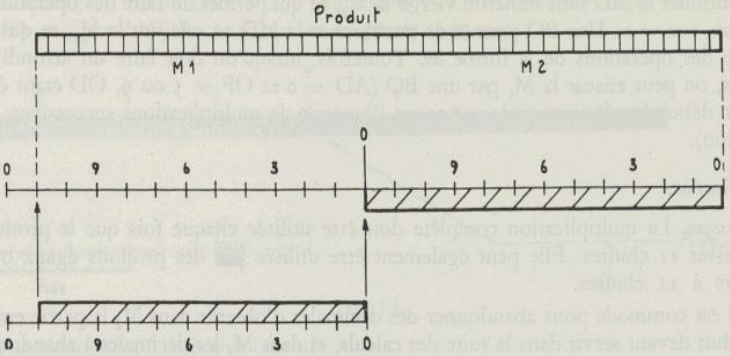


Fig. 90

encore il faut tenir compte que toute position vierge à droite du multiplicateur est traitée comme un zéro.

Exemple (*fig.* 91).

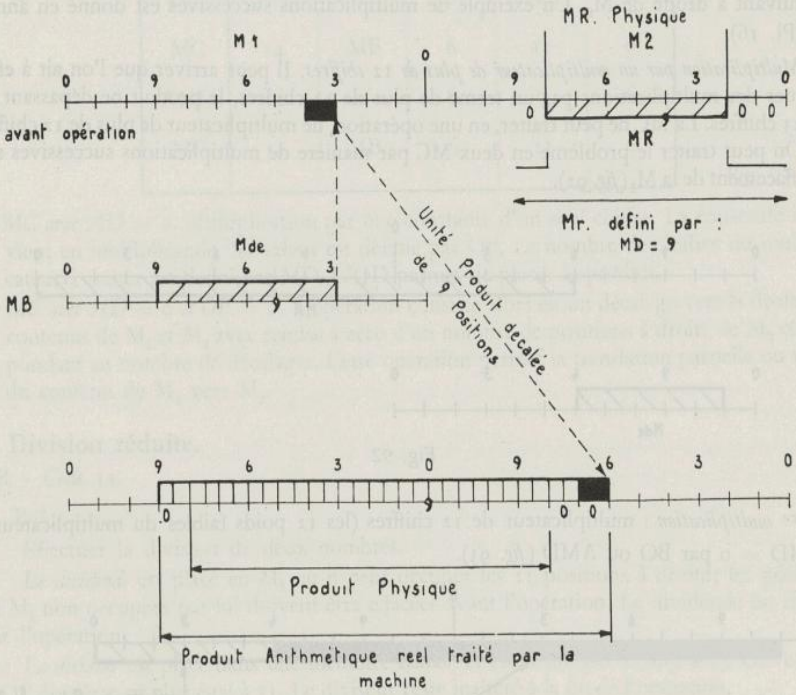


Fig. 91

D. *Positionnement de MD.*

Le positionnement, avant l'opération, de la MD peut, s'il n'est pas convenable, être effectué soit par une AMD, soit par une BO (avec AD = o et OF = o). Une AMD permet

de positionner la MD sans remettre vierge la M_1, ce qui permet de faire des opérations de la forme $ab + c = x$. Une BO permet de positionner la MD en effaçant la M_1, ce qui permet de faire des opérations de la forme ab. Toutefois, lorsqu'on doit faire un arrondi sur les décimes, on peut effacer la M_1 par une BO (AD = 0 et OF = 5 ou 9, OD étant égal à la position début des décimes (voir en annexe l'exemple de multiplications successives, 3e multiplication).

E. *Utilisation.*

a) *Classique.* La multiplication complète doit être utilisée chaque fois que le produit peut dépasser 11 chiffres. Elle peut également être utilisée pour des produits égaux ou inférieurs à 11 chiffres.

α) Il est commode pour abandonner des décimales d'obtenir dans M_1 la partie entière du produit devant servir dans la suite des calculs, et dans M_2 les décimales à abandonner. Ce résultat est obtenu par un MC avec un positionnement convenable des facteurs et de la mémoire décalage. L'exemple donné plus haut (§ cadrage de base) illustre cette possibilité.

β) Dans le cas de multiplications successives, il est possible d'enregistrer plusieurs multiplicateurs en M_2. En positionnant, avant chaque MC, la mémoire décalage au nombre de chiffres du multiplicateur à traiter, l'opération décale les autres facteurs en amenant le suivant à droite de M_2. Un exemple de multiplications successives est donné en annexe (Pl. 16).

b) *Multiplication par un multiplicateur de plus de* 12 *chiffres.* Il peut arriver que l'on ait à effectuer des multiplications par un terme de plus de 12 chiffres, le produit ne dépassant pas 23 chiffres. La MC ne peut traiter, en une opération, de multiplicateur de plus de 12 chiffres. On peut traiter le problème en deux MC par manière de multiplications successives sans effacement de la M_1 (*fig.* 92).

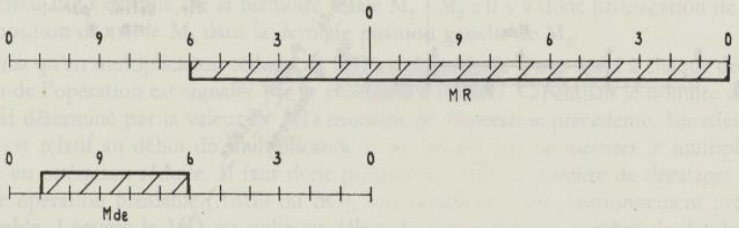


Fig. 92

1re *multiplication* : multiplicateur de 12 chiffres (les 12 poids faibles du multiplicateur) : MD = 0 par BO ou AMD (*fig.* 93).

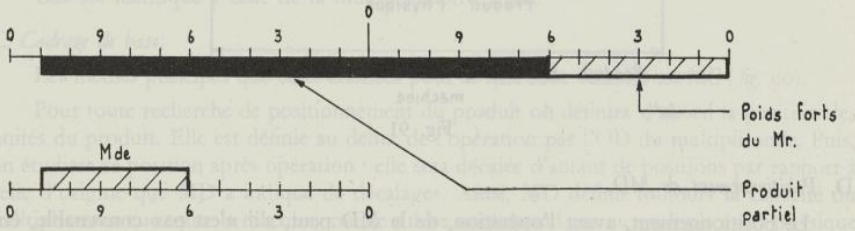


Fig. 93

2e *multiplication* : multiplicateur de 6 chiffres (les 6 derniers poids forts du multiplicateur) — MD = 6 par AMD sans effacement du produit (*fig.* 94).

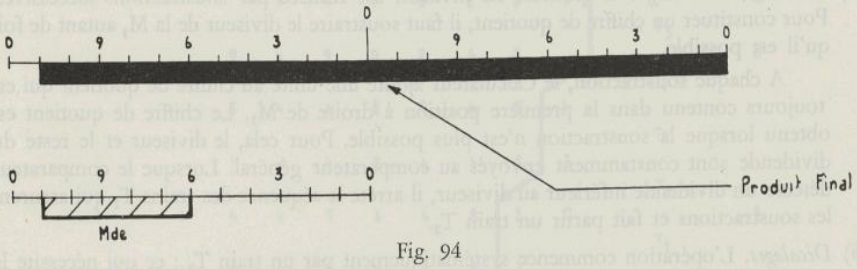


Fig. 94

La séquence sera la suivante (*fig.* 95) :

	TO	AD	OD	OF	MD
BO	6	1	0	0	0
MC	14	MB	6	11	0
AMD	2	0	6	0	6
MC	14	MB	6	11	0

Fig. 95

c) *MC avec AD* = 0. Multiplication par une constante d'un seul chiffre. La constante intervient en multiplicande. Sa valeur est définie par OF. Le nombre de chiffres du multiplicateur à traiter est défini par MD. L'OD permet de placer le produit.

d) *MC avec AD* = 0 *et OF* = 0. L'opération consiste alors en un décalage vers la droite des contenus de M_1 et M_2 avec remise à zéro d'un nombre de positions à droite de M_2 correspondant au nombre de décalages. Cette opération permet la translation partielle ou totale du contenu de M_1 vers M_2.

3. Division réduite.

DR - *Code* 13.

A. *Rôle.*

Effectuer la division de deux nombres.

Le dividende est placé en M_1 où il peut occuper les 11 positions à droite; les positions de M_1 non occupées par lui doivent être effacées avant l'opération. Le dividende est détruit par l'opération.

Le diviseur est placé dans une mémoire banale et est défini par AD (2 à 7), OD et OF. L'OF doit être au plus égal à 11. Le diviseur reste inaltéré à la fin de l'opération.

Le quotient est constitué dans la partie droite de M_1. Son nombre de chiffres est déterminé par l'OD du diviseur. Celui-ci doit donc être placé à gauche de la mémoire banale, et son nombre de chiffres, augmenté du nombre de chiffres de quotient à calculer, ne peut excéder 11.

Le reste se retrouve dans M_1, à gauche du quotient.

B. *Principe de l'opération.*

a) *Formation d'un chiffre de quotient.* La division est réalisée par soustractions successives. Pour constituer un chiffre de quotient, il faut soustraire le diviseur de la M_1 autant de fois qu'il est possible.

A chaque soustraction, le Calculateur ajoute une unité au chiffre de quotient qui est toujours contenu dans la première position à droite de M_1. Le chiffre de quotient est obtenu lorsque la soustraction n'est plus possible. Pour cela, le diviseur et le reste du dividende sont constamment envoyés au comparateur général. Lorsque le comparateur détecte un dividende inférieur au diviseur, il arrête la séquence des trains T_1 qui assurent les soustractions et fait partir un train T_3.

b) *Décalages.* L'opération commence systématiquement par un train T_3 : ce qui nécessite la position libre à l'extrême gauche de la mémoire.

Ce train, ainsi que les autres trains T_3 qui suivent la formation de chaque chiffre de quotient, provoquent un décalage vers la gauche. Ce décalage a trois actions :

— il décale le reste du dividende afin de pouvoir de nouveau lui soustraire le diviseur,

— il susbtitue dans les positions rendues libres du dividende les chiffres de gauche du quotient au fur et à mesure que ces chiffres sont constitués définitivement.

— il libère la première position à droite de M_1, afin de pouvoir y constituer un nouveau chiffre de quotient.

Ce comparateur reste en service pendant les trains T_3, afin d'enchaîner les décalages si le quotient contient des zéros, c'est-à-dire tant que le reste du dividende est inférieur au diviseur.

Exemple : soit à diviser 9 490 268 par 40 213 (*fig.* 96, *page* 87).

c) *Fin de l'opération.* L'opération est terminée lorsque le dernier chiffre du quotient est entièrement constitué, c'est-à-dire lorsque le nombre de décalages effectués est égal à l'OD du diviseur. La fin d'opération est déterminée par le processus suivant :

— A début de l'opération (pendant Tl) l'OD est transféré dans la MD dont le précédent contenu est effacé.

— A chaque train T_3, la MD est décomptée, comme dans la MR.

— Le contenu de MD est constamment envoyé au comparateur de cadrage qui le compare à zéro.

— Lorsque le comparateur général indique dividende < diviseur, et lorsque simultanément le comparateur de cadrage indique MD = 0, le signal de fin d'opération est déterminé et provoque l'exécution de l'opération suivante.

C. *Séquence des trains.* Les trains Tp et Tl sont systématiquement suivis d'un T_3. Ensuite la séquence des trains est entièrement déterminée par les deux comparateurs de cadrage et général.

Si dividende > diviseur :		départ de T_1.
Si MD > 0 et dividende < diviseur	:	— de T_3.
Si MD = 0 et dividende < diviseur	:	fin d'opération.

Durée totale : Tp et Tl : 3 tours.

Soustraction : nombre de tours égal à la somme des chiffres du quotient.

Décalages : nombre de tours égal à l'OD.

Pratiquement, il faut considérer que tous les chiffres du quotient sont des 9.

MB		4	0	2	1	3					
			9	4	9	0	2	6	8	T_3	3
		9	4	9	0	2	6	8		T_1	2
		5	4	6	8	9	6	8	1	T_1	
		1	4	4	7	6	6	8	2	T_3	
	1	4	4	7	6	6	8	2	0	T_1	1
	1	0	4	5	5	3	8	2	1	T_1	
		6	4	3	4	0	8	2	2	T_1	
		2	4	1	2	7	8	2	3	T_3	
	2	4	1	2	7	8	2	3	0	T_1	0
	2	0	1	0	6	5	2	3	1	T_1	
	1	6	0	8	5	2	2	3	2	T_1	
	1	2	0	6	3	9	2	3	3	T_1	
		8	0	4	2	6	2	3	4	T_1	
		4	0	2	1	3	2	3	5	T_1	
Reste →		0	0	0	0	0	2	3	6	Fin d' Opération	
							Quotient				

Fig. 96

La durée totale est alors donnée par la formule :

$$3 + (10 \times \text{OD}) \text{ tours.}$$

D. Utilisation.

a) *Cadrage de base des termes.* Il convient d'abord de remarquer qu'en toutes choses la division est symétrique de la multiplication. Elle s'effectue par soustractions successives et par décalages vers la gauche alors que la multiplication a lieu par additions successives et par décalages à droite. Au début d'une multiplication, on possède en M_1 le multiplicateur, à la même place où apparaîtra le quotient en division. Le produit se constitue à la fin de la multiplication à la place même où se trouve le dividende au début de la division.

La raison de cette symétrie se trouve dans le fait que les deux opérations ont trait au calcul de la formule :

$$ab + c = d$$

En multiplication a est le multiplicateur, b le multiplicande, c est nul et d est le produit qui joue le rôle d'inconnue. En division d est le dividende, b est le diviseur, a le quotient et c est le reste qui est toujours compris entre 0 et le diviseur :

$$0 \geqslant c < b$$

Il s'ensuit que la formule de capacité des divers facteurs en division est identique à celle de la multiplication :

N.b chiffres Dde = Nb. Chiffres Divr. + Nb. Chiffres quotient.

De sorte que le cadrage de base est le suivant (*fig.* 97).

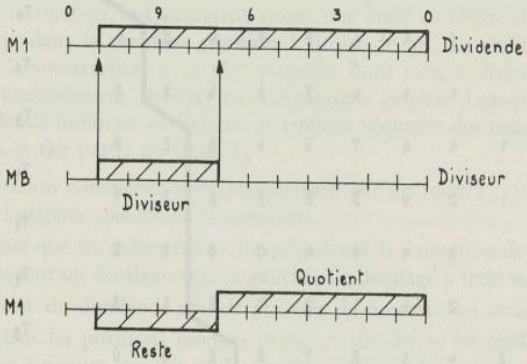


Fig. 97

Ce cadrage vérifie bien la formule de capacité énoncée plus haut.

Le dividende ne doit pas dépasser 11 chiffres et la dernière position de M_1 doit rester vierge. En effet, la séquence des trains, de la forme T_3 T_1, inverse de celle de la multiplication T_1 T_3, permet de faire démarrer l'opération par un T_3 systématique qui amène ainsi le dividende en M_1 (*fig.* 98).

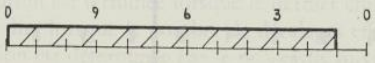


Fig. 98

Ainsi, est libérée la première position de M_1 dans laquelle commencera à se décompter le quotient.

Il s'ensuit que le premier T_1 a lieu sur des facteurs cadrés ainsi (*fig.* 99).

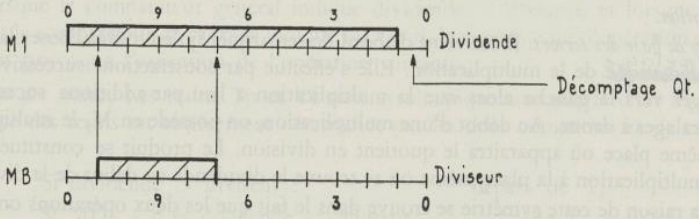


Fig. 99

Dans ces conditions, il est nécessaire d'être certain que le cadrage de base, avant T_3, assure l'impossibilité de la première soustraction. Il est facile de démontrer qu'il en est bien toujours ainsi. En effet, si cette première soustraction était possible, on décompterait, à partir d'elle, une position de quotient. Puis on ferait autant de décalages qu'en indique la

MD garnie par l'OD, ce qui porterait le quotient à un nombre de chiffres égal à OD + 1. Si l'on porte cette valeur dans la formule de capacité, on a -:

$$\begin{aligned} \text{Dde.} &= \text{Dr.} + \text{Quotient} \\ \text{OF (Dr.)} &= (\text{OF}-\text{OD}) + \text{OD} + 1 \\ \text{d'où OF} &= \text{OF} + 1, \text{ ce qui est absurde.} \end{aligned}$$

L'hypothèse de la première soustraction possible est donc absurde.

Si cette démonstration paraît abstraite, il est facile de se rendre compte de la justesse du cadrage de base de la division à partir de la constatation suivante. Dressons une table de Pythagore (*fig.* 100).

	1	2	3	4	5	6	7	8	9
1	1	2	3	4	5	6	7	8	9
2	2	4	6	8	10	12	14	16	18
3	3	6	9	12	15				
4	4	8	12	16					
5	5	10	15						
6	6	12							
7	7	14							
8	8	16							
9	9	18							81

Fig. 100

Il semble que l'on puisse la diviser en deux régions, l'une où le produit comporte deux chiffres, où se vérifie la formule de capacité :

$$\text{Mr.} + \text{Mde.} = \text{Produit.}$$

L'autre où le produit comporte un seul chiffre, où ne se vérifierait pas cette formule.

En réalité dans ce cas se vérifie aussi la formule de capacité, car le chiffre des dizaines est absent et remplacé par un 0 : le produit $2 \times 2 = 04$ et non $2 \times 2 = 4$. L'habitude de ne pas écrire les zéros à gauche n'est qu'une habitude esthétique. Elle ne veut pas signifier que les zéros à gauche soient moins significatifs que les zéros à droite : tout zéro est la matérialisation écrite de l'absence d'une puissance de la base de numération dans le décomptage d'un nombre. Il s'ensuit que le dividende 04 se trouve ainsi dans la mémoire M_1 par rapport au diviseur 2 et au quotient 2 (*fig.* 101, page 90).

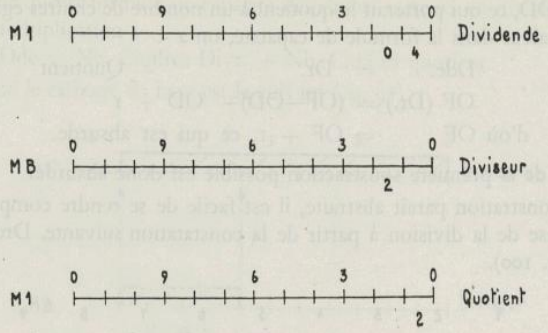


Fig. 101

et non pas ainsi (*fig.* 102) :

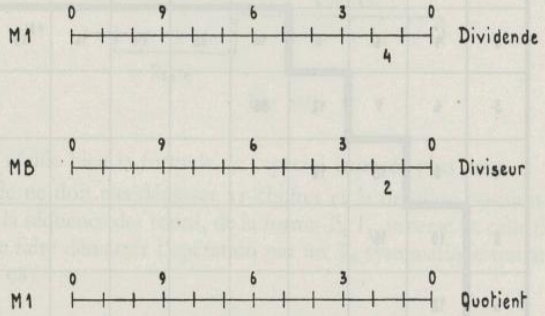


Fig. 102

Il est donc nécessaire, pour cadrer les termes de procéder ainsi :

α) Placer le quotient. C'est lui que l'on cherche et l'on connaît toujours le nombre de ses chiffres.

β) Appliquer la formule de capacité. Pour cela, on connaît toujours la capacité maximum de l'un des facteurs. On la porte dans la formule de capacité ainsi que la capacité maximum du quotient. On résout l'équation. On obtient la capacité maximum de l'autre facteur. On place alors la capacité réelle de ce facteur à l'intérieur de cette capacité théorique.

Exemple :

Quotient cherché 4 chiffres
Diviseur maximum 6 —
Dividende *réel* 8 — maximum.

$$\text{Dde} = \text{Dr.} + \text{Qt.}$$
$$x = 6 + 4$$
$$x = 10$$

D'où le cadrage (*fig.* 103).

Lorsque les facteurs apportés aux mémoires seront :

Dde = 8 chiffres.
Dr. = 6 —

le quotient s'écrira avec 2 chiffres significatifs à la droite de deux zéros, sous la forme :

00 a b

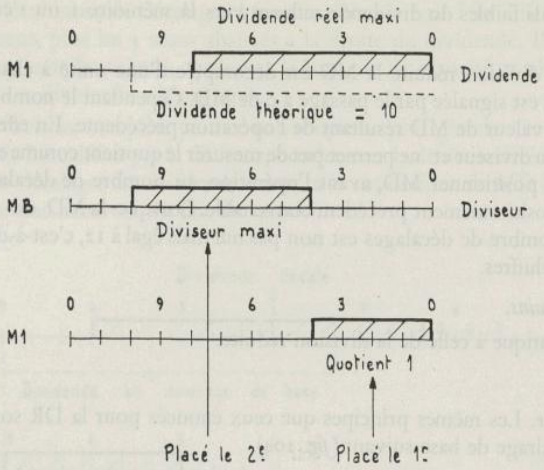


Fig. 103

b) *Sécurité de diviseur non nul.* Si le diviseur est nul, après le T_3 de départ, partiront indéfiniment des trains T_1, la soustraction étant alors toujours possible, la machine resterait alors bloquée sur cette ligne de programme. Il s'ensuit que l'on doit toujours, avant une division, s'assurer de ce que le diviseur n'est pas nul. Il faut donc faire précéder la division d'une comparaison relative au diviseur et d'une variante afin de sauter la division en cas de diviseur nul. Il faut à ce sujet faire attention que si les données proviennent des brosses, le diviseur ne sera lu que sur la première ou la deuxième brosse, de sorte que les premiers cycles d'alimentation (le premier ou les deux premiers selon qu'on lit en première ou seconde brosse), feraient, en démarrant le Gamma, se bloquer le programme sur la première division.

c) *DR classique.* La DR peut être utilisée avec les AD 2 à 7 chaque fois que l'on a quotient maximum + diviseur maximum $\leqslant$ 11.

d) *DR avec AD* + 0. Le diviseur est alors défini par la valeur de la constante affichée en OF. Cette valeur ne doit jamais être nulle. Le nombre de chiffres du quotient est alors égal à OD.

Division complète.

DC — *Code* 15

A. *Rôle.*

La division complète, qui s'effectue de la même manière, quant au principe, que la division réduite, permet de traiter des dividendes de 23 chiffres au maximum, le quotient pouvant atteindre au maximum 12 chiffres et le diviseur, la différence, soit 11 chiffres.

Pour cela la division complète utilise trois mémoires :

— La mémoire M_1-M_2 qui contient le dividende au début de l'opération — M_1 contenant, à la fin, le reste, M_2 contenant alors le quotient.

— Une mémoire banale qui contient le diviseur.

Comme dans la multiplication complète, les mémoires M_1-M_2 sont dans cette opération reliées ensemble de façon à constituer une mémoire de 24 positions. Au fur et à mesure des

décalages les poids faibles du dividende refluent vers la mémoire 1 où s'effectuent les soustractions.

Ainsi qu'en division réduite la MD est décomptée d'une unité à chaque décalage et la fin de l'opération est signalée par le passage à 0 de MD. Cependant le nombre de décalages est déterminé par la valeur de MD résultant de l'opération précédente. En effet, l'OD affiché est relatif au début du diviseur et ne permet pas de mesurer le quotient comme en division réduite. Il faut donc soit positionner MD, avant l'opération, au nombre de décalages voulus — soit bénéficier d'un positionnement précédent convenable. Lorsque la MD est nulle au début de l'opération, le nombre de décalages est non pas nul mais égal à 12, c'est-à-dire que le quotient calculé aura 12 chiffres.

B. *Séquence des trains.*

Elle est identique à celle de la division réduite.

C. *Utilisation.*

a) *Cadrage de base.* Les mêmes principes que ceux énoncés pour la DR sont valables en DC à partir du cadrage de base suivant (*fig.* 104).

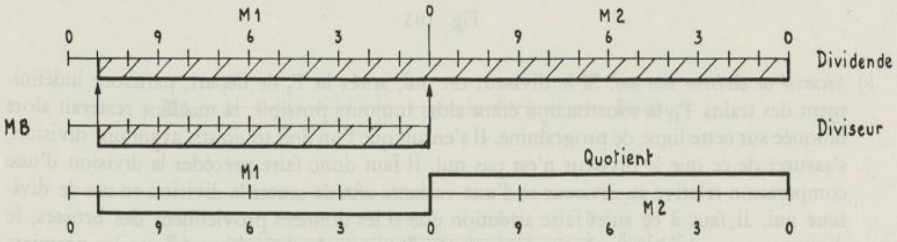


Fig. 104

On définira d'abord la place qu'occupera le quotient en M_2. Le nombre de décalages sera égal au moins à la position fin du quotient ainsi placé.

On appliquera la formule de capacité, et on placera le diviseur et le dividende de façon à ce que les poids forts de ces deux facteurs soient alignés sur le cadrage de base. Avec ce cadrage, le contenu de MD sera égal à la position fin du quotient.

Exemple (*fig.* 105) :

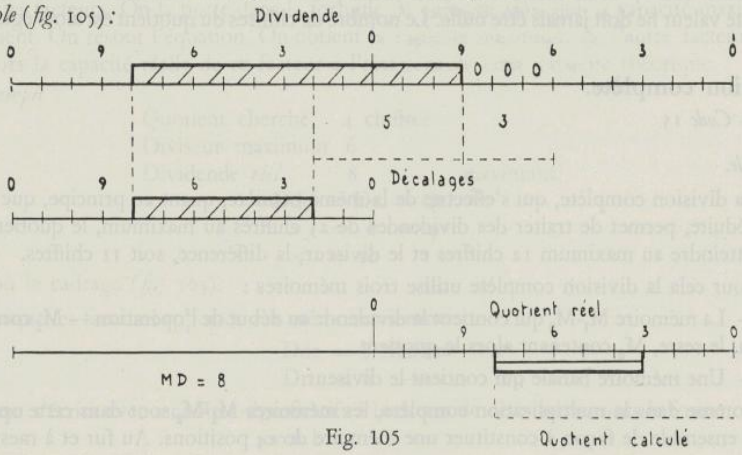


Fig. 105

On voit que le nombre de décalages à commander est égal à la différence de capacité entre Dde et diviseur, plus les 3 zéros abaissés à la droite du dividende. En somme, il faut prendre le quotient complet calculé, appliquer la formule de capacité et placer le dividende réel à l'intérieur de ce dividende arithmétique.

Toutefois, la division complète permet une souplesse plus grande. On peut fort bien, à partir du cadrage de base défini par le diviseur, décaler vers la droite le dividende. Le nombre de ces positions de décalages du dividende donnera la majoration de MD à prévoir à partir du nombre qu'elle doit recevoir du fait du quotient.

Exemple (*fig.* 106) :

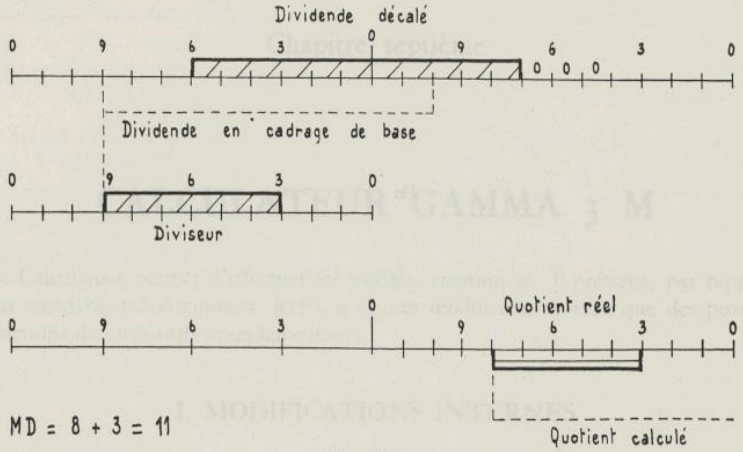


Fig. 106

En ce qui concerne les virgules, il est facile de voir qu'il suffit de traiter le problème comme s'il n'y avait pas de virgule et appliquer la formule de capacité pour placer les facteurs ainsi multipliés par des puissances de 10. On placera ensuite les virgules en appliquant la formule de capacité sur les parties des nombres traités. Ainsi dans l'exemple précédent, si le quotient possède 2 décimales avant la virgule et le diviseur 4 décimales avant la virgule, le dividende en comportera 6 avant la virgule qui se trouvera à la jonction de M_1 et M_2.

Une formule peut être proposée pour résoudre ces problèmes :

$$\text{Virg. Dde.} = \text{Virg. Dr.} + \text{Virg. Qt.} + 12 - \text{MD}$$

Cette formule respecte la numérotation normale de M_1 et M_2. Si l'on considère M_1-M_2 comme une seule mémoire numérotée de 0 à 23, la formule devient plus pratique et s'écrit :

$$\text{Virg. Dde.} = \text{Virg. Dr.} + \text{Virg. Qt.} - \text{MD}.$$

b) *DC classique*. La division complète, avec les AD de 3 à 7, doit être utilisée chaque fois que la somme des nombres de chiffres du diviseur et du quotient est supérieure à 11.

En raison de la souplesse de cadrage examinée au paragraphe précédent, supérieure à celle présentée par la D.R., cette opération peut être avantageusement utilisée à la place d'une DR. lorsqu'elle facilite les cadrages.

On doit prendre les mêmes précautions qu'en DR quant au diviseur nul.

c) *DC avec AD* = 0. Le diviseur d'un chiffre est défini par OF qui ne doit jamais être nul. Le nombre de chiffres du quotient est défini par la valeur de MD, OD définit la position de la virgule du quotient.

Chapitre septième

CALCULATEUR GAMMA 3 M

Ce Calculateur permet d'effectuer les calculs scientifiques. Il présente, par rapport au Gamma standard, précédemment décrit, quelques modifications, ainsi que des possibilités d'adjonctions de dispositifs complémentaires.

I. MODIFICATIONS INTERNES

1. Interrupteurs.

Trois interrupteurs (représentés ci-dessous) placés sur le côté gauche du tableau de connexions sont ajoutés :

A. *Interrupteur de droite « cartes »*.

Il permet (pour la connexion PRD) de modifier le décomptage du « code point », afin que les cartes puissent être alimentées soit à l'endroit, soit à l'envers.

Lorsque l'interrupteur est placé à gauche (« 9 en tête »), les cartes sont alimentées normalement dans le magasin, c'est-à-dire ligne des 9 en tête : lorsqu'il est placé vers la droite (« 12 en tête »), les cartes sont alimentées à l'envers, c'est-à-dire ligne des 12 en tête.

Il est impossible d'alimenter les cartes à l'envers dans le cas de connexion double PL-Gamma 3.

B. *Interrupteur du centre « Codification des zéros »*.

Il peut occuper trois positions :

— à gauche « Aucun », aucun zéro n'est codifié en mémoire ;

— au centre « A droite », les zéros sont codifiés à droite du premier chiffre significatif dans la limite du filtrage ;

— à droite « Tous », tous les zéros sont codifiés dans la limite du filtrage.

C. *Interrupteur de gauche « Signe »*.

— en position « Avec », il autorise l'action du dispositif « opérateur de signe »,

— en position « Sans », il supprime l'action de ce dispositif.

Pour les machines non munies du dispositif « opérateur de signe », cet interrupteur doit obligatoirement se trouver en position « sans ».

— L'utilisation de l'opérateur de signe est incompatible avec la position de l'interrupteur sur « Tous zéros ».

SIGNES	ZÉROS	CARTES
Avec — Sans	Aucun — A droite — Tous	9 en tête — 12 en tête

2. Commandes S et V.

Les commandes S et V ne sont plus à retombée systématique comme il a été dit dans la description du Gamma standard. Elles sont de la catégorie des commandes à retombée commandée : une fois appelées, elles se maintiennent jusqu'à ce qu'on commande positivement leur retombée au connecteur. Sur le connecteur Gamma M (Pl. 17), figurent des plots strappés « V » qui contrôlent la retombée des commandes V_0 à V_3. Les plots strappés « S_{03} » et « $S_{4\text{-}7}$ » contrôlent respectivement la retombée des commandes S_0 à S_3 et S_4 à S_7. De même, sur le connecteur supplémentaire, des plots strappés permettent la retombée des commandes V_4 à V_7 et S_8 à S_{11} et S_{12} à S_{15}.

Ces plots peuvent être alimentés par une commande quelconque de la machine connectée : la retombée n'a pas lieu au début de l'émission de la commande, mais à la coupure de cette commande. Il est ainsi possible d'agir sur le programme au cours de plusieurs cycles successifs à partir d'éventualités observées par les commandes S et V. Le plot PS permet de faire retomber systématiquement les commandes à la fin de chaque cycle.

Il est évident que les impulsions d'appel de V ou S d'un groupe ne doivent jamais être présentes lors de la fin de l'impulsion de retombée de ce groupe.

Le nombre des commandes V est réduit par rapport à la machine standard : il est de 4 sur le connecteur normal et de 4 sur le connecteur supplémentaire.

Le nombre des commandes S est inchangé. Toutefois, 4 seulement émettent leur maintien au connecteur : S_0, S_1, S_8, S_9.

3. Plots PS. - 9 à 1 - 9 à 0 - 9 à 11, etc...

Ces plots ne doivent jamais être utilisés pour contrôler des organes de la machine connectée. Ils ne doivent pas non plus être mis en parallèle avec des commandes de la machine connectée.

En connexion P.R.D. le plot PS n'émet pas : son rôle est assuré par les commandes IS qui émettent de 0,20 à 13,20.

4. Introducteur 2.

L'introduction cinématique, par l'introducteur 2, est possible pour les points de 9 à 0 au lieu de 9 à 1 dans le Gamma standard. Il y a donc lieu de veiller lorsque le multiplicateur d'une MC est introduit en cinématique dans M_2 à ce que l'introduction soit contrôlée de 9 à 1 seulement.

5. Extracteur.

On peut commander la validation des extracteurs par :

1°) — des commandes de cycle ou de sélection,

2°) — des commandes Gamma.

Il est interdit de le faire :

1°) — par un chiffre du distributeur (dont l'émission cesse avant qu'on soit assuré d'avoir accroché les maintiens nécessaires).

2°) — en règle générale par toute commande dont le début n'est pas situé entre les degrés 16 et 20 et risque de commander l'extraction.

Nota : — pour commander l'extraction de 12 à 15 on peut utiliser :

a) — Une commande de cycle longue (14 à 11,15). L'extraction aura lieu de 9 à 15.

b) — Une commande longue filtrée par une came à 11 (de 0,18 à 11,18). L'extraction aura lieu de 12 à 15.

c) — La came β. Cette came peut avoir le profil et le réglage désirés par le client et peut ainsi émettre pendant un temps compris entre 3 et 14 points. Le plot correspondant est le troisième à partir de la droite sous coupure BI. (Cette came est incompatible avec le dispositif de 3 cames filtre). A noter que le plot PS extrait de 12 à 9 et non de 12 à 15).

d) — les plots émetteurs IS (0,20 à 13, 20).

Pour commander l'extraction d'un 11, on peut soit utiliser la came β, soit le plot O du sélecteur alimenté par une commande de cycle.

6. VPM.

Le calcul scientifique, en programme par carte, impose de posséder l'opération VPM non seulement pour les points entre cycles mais à tous les points machines. Le nombre des VPM a donc été augmenté. Il existe :

a) Pour chaque point, deux VPM à + 3° et 14°;

b) Une VPM validable à + 3° de chaque point.

Ces VPM occupent les adresses de 8 à 15 et sont distinguées par les codes OF (voir le tableau de code Gamma M (Pl. 18). Les VPM Pt. 3 n'autorisent le départ du programme qu'au degré 3 des 12 points de lecture. Dans l'entre cycle, elles bloquent le déroulement du programme.

7. Opération SN.

Il est possible d'effectuer dans tous les cas la soustraction de deux termes occupant la douzième position.

8. Opération K.B.

Lorsque l'on doit extraire un code indicatif, l'opération KB d'introduction de ce code dans la mémoire à extraire doit être terminée au moins 4° avant le point à extraire (ex. 11,20 pour 12).

II. DISPOSITIFS COMPLÉMENTAIRES

Le Calculateur Gamma M peut être muni de dispositifs complémentaires qui seront décrits :

1. Le programme par cartes (PPC).

Les instructions du programme sont perforées dans les cartes et lues par la machine connectée. Elles sont transmises au dispositif « PPC » par l'intermédiaire d'un connecteur supplémentaire sur le tableau de connexions.

2. Armoires de mémoires supplémentaires.

Il peut être adjoint au Gamma M 3 armoires contenant 8 mémoires supplémentaires portant à 31 le nombre des mémoires. Ces mémoires ne comportent ni introducteurs ni extracteurs et ne sont donc accessibles que depuis la mémoire opérateur.

3. Point décimal flottant.

Ce dispositif introduit dans le programme deux opérations spéciales « BD » et « DCC ». Ces opérations facilitent l'utilisation des opérations standard avec point décimal flottant.

4. Opérateur de signe.

Il permet de traiter les nombres avec leurs signes : il détermine automatiquement l'opération à effectuer et le signe du résultat.

III. LE PROGRAMME PAR CARTES (P.P.C.)

Ce dispositif est destiné aux calculs comportant de longues chaînes pour un petit nombre de groupes de données. Il consiste en un connecteur et un câble supplémentaire donnant accès, à partir du tableau de la machine connectée à un panneau de 64 relais supplémentaires situé dans le Gamma.

Il permet de fournir les instructions de programme non plus seulement au moyen du tableau de connexions, mais aussi à partir de cartes. Les instructions TO, AD, OD, OF sont perforées dans la carte en binaire, à raison de 4 colonnes pour une instruction correspondant aux poids binaires 1, 2, 4, 8. Une poinçonneuse spéciale (poinçonneuse cartes-programme) est prévue à cet effet.

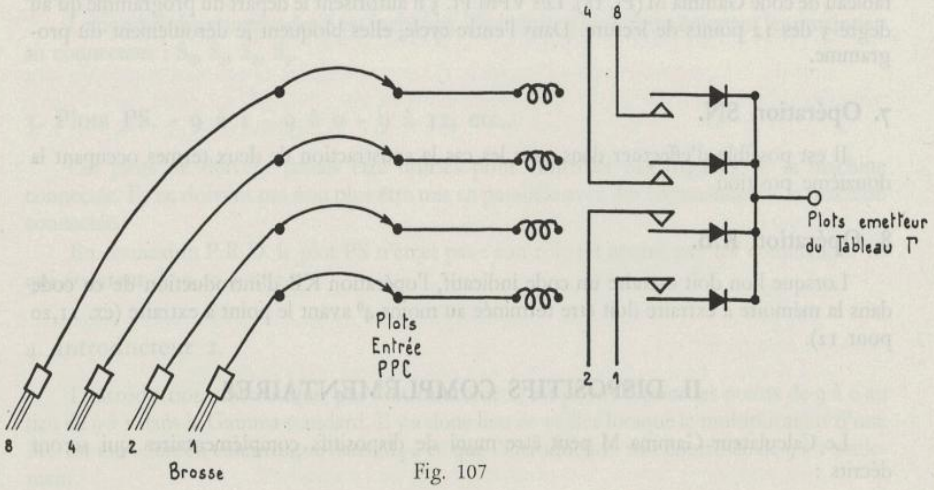


Fig. 107

Les colonnes de brosse correspondant à la zone programme contrôlent les 64 relais du PPC par l'intermédiaire du connecteur. Chaque relais correspond donc à un poids binaire et un ensemble de 4 relais à un code Gamma. Avec 64 relais, il est possible d'afficher simultanément 16 codes, soit 4 lignes de programme.

La présence d'une perforation positionne le relais correspondant au travail pendant le temps de fermeture du rupteur. Les rythmes « *e* » explorent les relais PPC. et, combinés 4 à 4 constituent des codes binaires analogues à ceux des distributeurs du tableau Gamma. Chacun de ces codes apparaît sur un plot du tableau de connexions Gamma. Les 16 plots utilisés à cet effet sont ceux du distributeur inférieur. affecté normalement aux alternatifs. Ce distributeur est donc inutilisable en standard sur tous les Gamma qui possèdent le dispositif PPC.

Ces plots, reliés par connexion aux plots TO, AD, OD, OF, de 4 lignes de programme, agiront donc comme des codes-distributeurs dont la valeur est à chaque point, fonction des perforations qui se trouvent dans la carte.

Sur chaque ligne de cartes on a la possibilité d'utiliser 16 codes, comme bon semble, par exemple, 4 lignes au point 9, 4 lignes au point 8, etc., soit en tout par carte 48 lignes de programme.

Il faut donc pouvoir démarrer un calcul à chaque point machine : c'est la raison d'être de la VPM validée au degré 3 de tous les points (VPM Pt. 3) qui renvoie à chaque point sur la première des 4 lignes du tableau affecté au PPC.

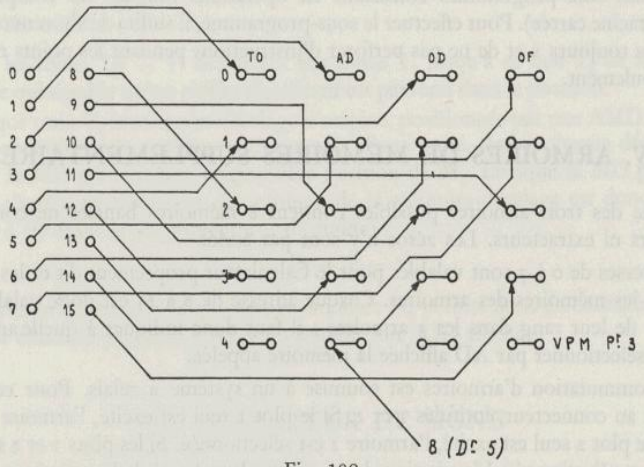


Fig. 108

Les codes sont donc émis sur les mémoires de programme sélectionnées par le NL de 3 à 14° de chaque point. Il faut donc que les codes de la dernière opération soient encore émis lorsque le NL parviendra à sa ligne, ce qui revient à dire que cette ligne doit être sélectionnée par le NL avant 14°. Une fois le train Tp de la dernière opération effectué, les codes correspondants se trouvent enregistrés dans les mémoires programme et l'opération peut se dérouler après le degré 14.

En pratique, on a constaté que le temps disponible depuis le degré 3 jusqu'au degré 14 permettait d'effectuer 37 tours maximum pour les 3 premières opérations d'une ligne de carte (ou les 5 premières si l'on est en six opérations). On ne doit jamais évidemment dépasser ce nombre sur aucune machine.

Quant à la durée de la quatrième opération, elle peut comprendre les 123 tours maximum d'une opération récurrente complète avant la VPM Pt. 3 de la ligne suivante. On peut donc afficher une série d'instructions par ligne de carte.

La zone de programme coïncide presque toujours avec des zones données ; les colonnes correspondantes sont envoyées en fiches doubles de la brosse aux positions d'introducteurs et aux commandes des relais PPC qui apparaissent au connecteur PPC. Il faut donc empêcher les perforations des cartes données d'être considérées comme des instructions de programmes. A cet effet, il existe sur le Gamma M un plot « suppression calcul » (situé sur le connecteur standard à la place du plot inférieur « IS ») qui bloque le déroulement du programme aussi longtemps qu'il est alimenté.

Ce plot reçoit donc une commande de cycle filtrée par un alternatif contrôlé par une commande de sélection appelée en brosse de sélection par l'indicatif des cartes données.

Remarques :

1. Les codes de PPC étant valables indifféremment pour TO, AD, OD, OF, on peut les utiliser selon d'autres organisations que celles consistant à afficher 4 lignes avec TO, AD, OD, OF. Par exemple, on peut afficher 6 lignes PPC, les premières comportant seulement TO, AD.
2. La plus grande partie des lignes du tableau de connexions reste libre. Il est possible d'y afficher des sous-programmes consistant en opérations longues ou compliquées (par exemple racine carrée). Pour effectuer le sous-programme, il suffira de s'y renvoyer par une variante « toujours » et de ne pas perforer d'instructions pendant les points nécessaires à son déroulement.

IV. ARMOIRES DE MÉMOIRES SUPPLÉMENTAIRES

Chacune des trois armoires possibles contient 8 mémoires banales ne comportant ni introducteurs ni extracteurs. Les zéros n'y sont pas codés.

Les adresses de 0 à 7 sont valables pour le Calculateur proprement dit et les adresses de 8 à 15 pour les mémoires des armoires. Chaque adresse de 8 à 15 est donc valable pour les 3 mémoires de leur rang dans les 3 armoires : il faut donc indiquer à quelle armoire on se réfère pour sélectionner par AD affichée la mémoire appelée.

Cette commutation d'armoires est soumise à un système à relais. Pour cela, 2 plots apparaissent au connecteur, intitulés 1 et 2. Si le plot 1 seul est excité, l'armoire 1 est sélectionnée. Si le plot 2 seul est excité, l'armoire 2 est sélectionnée. Si les plots 1 et 2 sont excités, l'armoire 3 est sélectionnée. L'excitation de ces plots a lieu depuis la lecture de perforations à la ligne de carte où la commutation d'armoire doit avoir lieu.

Cette commutation n'est pas instantanée : il faut un point machine pour qu'elle ait lieu. Durant ce point, on ne peut utiliser aucune armoire mais on peut utiliser le Calculateur.

L'appel d'une armoire doit donc avoir lieu à la ligne précédant la 1re opération utilisant une mémoire de cette armoire.

V. POINT DÉCIMAL FLOTTANT

Deux opérations sont prévues pour les problèmes traités en point décimal flottant : l'opération BD et l'opération DCC.

1. Transfert de mémoire banale en mémoire décalage.

BD - *Code* 7.

A. *Rôle.*

L'opération consiste dans le transfert du contenu d'une position de la mémoire banale 2 dans la MD. L'adresse est 0, ainsi que l'OF. L'OD définit le rang de la position de M_2 que l'on transfère. Il est ainsi possible de transférer le résultat d'un calcul dans la mémoire décalage, lorsque celui-ci ne dépasse pas 9.

B. *Séquence des trains.*

Seuls les trains Tp-Tl sont démarrés.

2. Division complète de cadrage.

DCC - *Code* 15.

A. *Rôle.*

L'opération consiste en une division complète de cadrage. Elle permet de décaler les mémoires M_1-M_2 vers la gauche jusqu'à ce qu'un chiffre significatif parvienne dans la position définie par OD. L'affichage en est le suivant : TO = 15, AD = 0, OD = à fixer, OF = 1. C'est donc une division par 1 qui s'arrête lorsque la division pourrait avoir lieu.

B. *Séquence des trains.*

Après les trains Tp — Tl démarrent des trains T_3 jusqu'à ce que T_1 ait tendance à se produire, ce qui signifie qu'un chiffre significatif est parvenu dans la position.

A chaque train T_3 la mémoire décalage d'origine, positionnée par une AMD préalable ou tout autre opération convenable, est décomptée d'une unité. Avant chaque décomptage, le contenu de la MD est transféré en première position de M_2. Lorsque la MD parvient à 0, l'opération s'arrête normalement. Le contenu de la mémoire décalage est donc égal, après δ décalages à (D étant le positionnement d'origine) :

$$x = D - (\delta - 1).$$

L'analyse du contenu de la première position de M_2 permet donc de connaître le nombre de décalages effectués jusqu'au blocage du nombre sur l'OD affiché.

VI. OPÉRATEUR DE SIGNE

C'est un dispositif complémentaire qui permet de programmer les 4 opérations algébriques aussi aisément que sur des nombres arithmétiques. Le signe d'un nombre est enregistré en mémoire banale dans la dernière position décimale à gauche de la zone filtrée. Les transferts de signes sont contrôlés par un « filtrage-signe » (FS) défini par l'intervalle OF — 1, OF.

Le signe « — » est représenté par un zéro codé (code 10) dans cette position. Si le nombre est positif, le signe « + » n'est pas symbolisé par un code spécial, et la position réservée au signe peut être occupée par un blanc, ou même par un code arithmétique (< 10), et représentant l'ordre décimal le plus élevé du nombre.

L'opérateur de signe est constitué :

— d'une mémoire signe (MS_1) qui contient à tout instant le signe du nombre situé en mémoire opérateur ;

— d'une autre mémoire signe (MSB) qui reçoit le signe du second terme utilisé au cours de l'opération algébrique et défini par AD-OD-OF;

— et d'un opérateur de signes proprement dit, qui détermine automatiquement le signe du résultat, c'est-à-dire le nouveau contenu de MS_1.

1. Introduction des signes en mémoire.

A. *Introduction en mémoire signe.*

La MS_1 et la MSB ne peuvent recevoir de signe que pendant les opérations du programme et dans les conditions décrites dans les paragraphes suivants.

B. *Introduction cinématique.*

Le signe « — » représenté par un code 10 peut être introduit dans une *mémoire banale* à l'aide d'une introduction cinématique, mais ne peut l'être en mémoire opérateur. Cette dernière n'enregistre que les codes arithmétiques. Lorsqu'un terme introduit en *mémoire opérateur* est négatif, son signe, provenant de la machine connectée, peut appeler une commande S qui agit par connexion sur les instructions du programme lors de la première utilisation du terme. Pour la suite des calculs, l'opérateur de signes agit seul.

Dans le Gamma mathématique, la *mémoire* 2 peut recevoir des zéros codés au cours d'une introduction cinématique. Cependant, ces codes 10 ne doivent pas entrer en mémoire opérateur par suite de décalages vers la gauche (DC), ni être décomptés comme des chiffres du multiplicateur (MC).

C. *Introduction statique.*

Le signe « — », comme tout code non arithmétique, ne peut être introduit dans une mémoire quelconque à l'aide de l'introduction statique. Il doit être enregistré en mémoire par une autre méthode : par exemple à l'aide d'une opération KB sélectionnée par une commande S ou une variante.

2. Transfert du signe de mémoire banale vers MS_1.

A. *Cas général* (*fig.* 109).

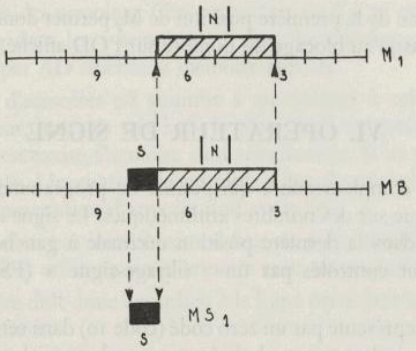


Fig. 109

Soit le nombre |N| en mémoire banale, son signe S étant sur sa gauche.
L'opération BO permet de transférer |N| en M_1 et le signe en MS_1.
On l'affiche ainsi :

TO	AD	OD	OF
6	B	3	8

De OD à OF-1, soit de 3 à 7 le transfert de |N| a lieu au profit de M_1; de OF — 1 à OF, il a lieu au profit de MS_1 pour le signe et au profit de M_1 qui élimine le 10 sur son entrée. Il s'ensuit que si la position de OF — 1 à OF comporte un chiffre arithmétique (cas d'un nombre positif ordinaire) la MS_1 ne reçoit rien puisque ce n'est pas un code 1010 qui est filtré, tandis que M_1 reçoit le chiffre arithmétique pour la même raison.

B. *Maintien filtré de la mémoire opérateur — BO avec adresse* 1.

Au cours de l'opération BO avec adresse 1 (maintien filtré de la mémoire opérateur) le signe du nombre situé en mémoire opérateur est conservé en MS_1.

C. *Remise à zéro de mémoire opérateur et MS_1 — BO avec adresse* 0.

L'adresse zéro ne distribuant que des chiffres positifs, une opération BO avec adresse 0 efface le contenu de MS_1.

Cette propriété et celles déjà connues font de cette opération la mieux indiquée pour réaliser *l'effacement simultanée* de la *mémoire opérateur* et de la *MS_1* avec ou sans altération de la MD.

Remarque : l'opération ZB avec adresse 1 remet à zéro la mémoire opérateur sans altérer les contenus de la MS_1 ni de la MD. C'est, en effet, comme banale qu'est alors utilisée la M_1.

3. Transfert de signe MS_1 vers mémoire banale.

A. *Cas général* (*fig.* 110).

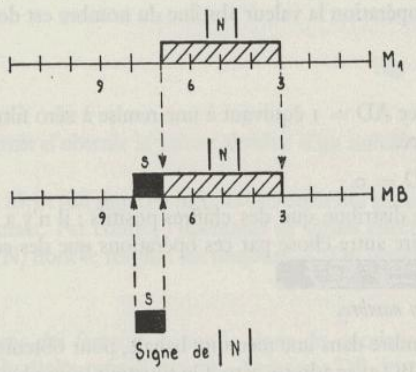


Fig. 110

Lorsque le nombre situé en M_1 est négatif, l'opérateur de signes commande automatiquement, au cours de l'opération OB, le transfert d'un code 1010 dans la MB dont l'adresse est affichée, et dans la position définie par le filtrage signe FS, entre OF — 1 et OF.

L'affichage est le suivant :

TO	AD	OD	OF
8	B	3	8

Lorsque le signe de |N| est positif, la position de MB entre OF — 1 et OF ne reçoit rien.

Si par suite d'une erreur d'affichage de OF la position de M_1 correspondant au filtrage signe comprend un chiffre significatif, ce chiffre a priorité sur le signe : il est transféré en MB à la place du signe qui n'est pas transféré.

B. *Effacement filtré de M_1 (OB avec AD = 1).*

Au cours de cette opération le contenu de MS_1 n'est pas altéré.

4. Opération comparaison.

La comparaison reste arithmétique : le signe n'y intervient pas.

5. Addition et soustraction.

A. *Cas général.*

Lors de l'addition ou de la soustraction de deux nombres algébriques, A en mémoire opérateur, et B en mémoire banale, l'opération arithmétique est déterminée non plus par le TO seul (AN ou SN), mais par le TO et les signes de A et B afin d'effectuer :

$$(\pm A) \pm (\pm B) = \pm C.$$

A la fin de l'opération, le résultat est toujours exprimé en valeur absolue et son signe est en MS_1. La soustraction de deux nombres égaux en valeur absolue entraîne l'apparition d'un signe négatif : le zéro est dit « négatif ».

Pour le décompte du nombre de trains, l'AN doit, comme la SN, compter pour 6, le train de redressement pouvant avoir lieu en AN.

— Après l'opération [redacted] AN si les termes sont égaux en valeur absolue et de signes contraires, la mémoire MS_1 peut indiquer indifféremment « plus » ou « moins » sans qu'il soit possible de le savoir au préalable.

B. *AN avec AD* = 1 (*Multiplication par* 2).

Au cours de cette opération la valeur absolue du nombre est doublée, sans altération du signe.

C. *SN avec AD* = 1.

L'opération SN avec AD = 1 équivaut à une remise à zéro filtrée de M_1 avec inversion du signe du terme effacé.

D. *AN ou SN avec AD* = 0.

La mémoire OF ne distribue que des chiffres positifs : il n'y a donc aucune possibilité d'ajouter ou de soustraire autre chose par ces opérations que des constantes positives, [redacted]

E. *Inversion du signe d'un nombre.*

Étant donné un nombre dans une mémoire banale, pour obtenir son opposé, on remet à zéro la M_1 à l'aide d'une BO avec adresse zéro. On soustrait le nombre contenu en MB de la M_1 ainsi remise à zéro, puis on renvoie par une OB le résultat en MB.

6. Multiplication réduite.

A. *Cas général.*

L'opérateur de signes effectue le produit du signe du multiplicateur, contenu en MS_1, avec celui du multiplicande, transmis en MSB durant l'opération. Ce signe du produit se trouve en MS_1 à la fin de l'opération.

Toutes les règles énoncées pour le cadrage sont valables quant aux valeurs absolues du multiplicateur et du multiplicande : les signes n'interviennent pas dans les capacités. En effet, les signes subissent le filtrage signe entre OF et OF — 1 qui apparaît comme l'OF propre aux valeurs absolues des facteurs. Il est donc possible de faire, en apparence, chevaucher d'une position (celle du signe) les filtrages du multiplicande et du multiplicateur et de donner au multiplicande un OF nul, à condition bien entendu que les positions de ces facteurs, réservées

aux signes ne contiennent que des codes 1010 ou 0000 : le signe qui se trouve dans la dernière position du filtrage n'entre pas en M_1 (*fig.* 111).

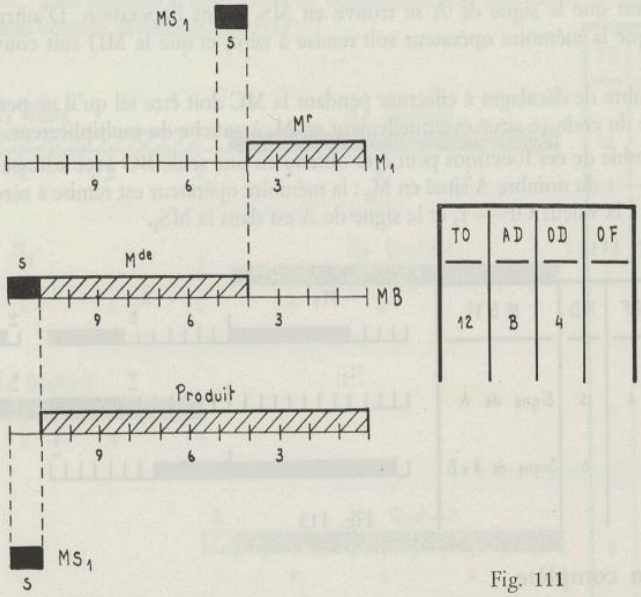


Fig. 111

B. *MR avec adresse* 1.

Cette opération permet d'obtenir la valeur absolue d'un nombre. L'OD et l'OF affichés sont alors nuls.

Le nombre N est en M_1 et son signe en MS_1. L'OD étant nul la multiplication n'a pas lieu : le nombre N est alors intact. Par contre, l'opérateur de signes effectue le produit des signes (signe de N $\times$ signe de N) dont le résultat est toujours $+$. On obtient donc bien $|N|$.

7. Division réduite.

Le produit des signes des facteurs donne le signe du quotient en MS_1. Pour la même raison qu'en MR, l'OF du diviseur peut être nul. Après une DR, le reste est situé en mémoire opérateur immédiatement à gauche du quotient. Avant de transférer le quotient en mémoire banale, il faut effacer ce reste à l'aide d'une opération BO avec adresse 1, de façon que la position de la mémoire opérateur correspondant au FS donné au cours de l'OB ne contienne plus de chiffre significatif. D'autre part, l'OF affiché à la ligne de l'OB doit être supérieur d'au moins une unité à celui du quotient (*fig.* 112).

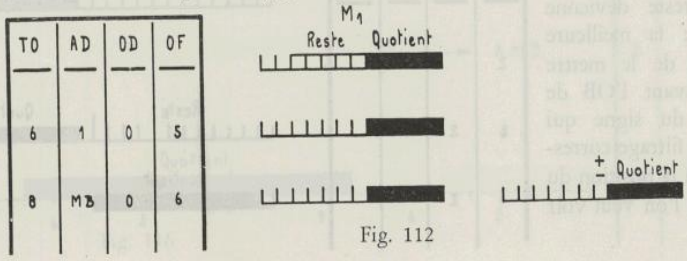


Fig. 112

8. Multiplication complète.

Dans la multiplication complète de deux nombres, A en mémoire 2, et B en mémoire banale, il faut que le signe de A se trouve en MS_1 avant l'opération. D'autre part, il est nécessaire que la mémoire opérateur soit remise à zéro, et que la MD soit convenablement positionnée.

Le nombre de décalages à effectuer pendant la MC doit être tel qu'il ne permette pas le décomptage du code 10 situé éventuellement en M_2 à gauche du multiplicateur.

L'ensemble de ces fonctions peut être obtenu en une seule BO avec adresse 2 et un OD égal à l'OF — 1 du nombre A situé en M_2 : la mémoire opérateur est remise à zéro, la MD est positionnée à la valeur OF — 1, et le signe de A est dans la MS_1.

Exemple (*fig.* 113) :

TO	AD	OD	OF	MD	MS1
6	2	5	6	5	Signe de A
14	B			0	Signe de A x B

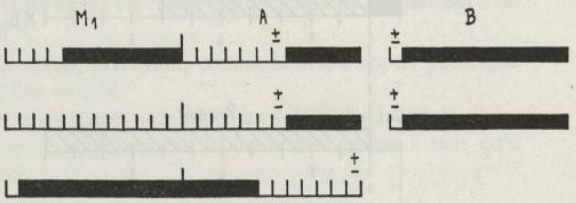


Fig. 113

9. Division complète.

Après la division complète de deux nombres, le quotient est en M_2 exprimé en valeur absolue, tandis que son signe est en MS_1. Le transfert BO du quotient en M_1 effacerait le signe de MS_1 et donnerait en M_1 la valeur absolue du quotient.

Deux procédés peuvent être employés pour regrouper la valeur absolue du quotient et son signe.

A. *Par transferts.*

Le signe est regroupé avec le quotient en M_2 par une OB. Puis une BO transfère le quotient et son signe en M_1 et MS_1. L'OB a pour rôle de transférer uniquement le signe. Si l'on désire le mettre en douzième position de M_2, il n'y a pas de difficulté, car une AMD à 11 permettra à une OB avec OD — 11 et OF = 0 de transférer le signe sans que joue la priorité des chiffres significatifs de M_1. En effet, le reste va au maximum jusqu'en onzième position de M_1.

D'où la séquence de la figure 114.

Si on veut le mettre ailleurs, il peut se faire que le reste devienne gênant : la meilleure façon est de le mettre à zéro avant l'OB de transfert du signe qui recevra le filtrage correspondant à la position du signe que l'on veut voir en M_2.

TO	AD	OD	OF
2		11	
8	2	11	
6	2		

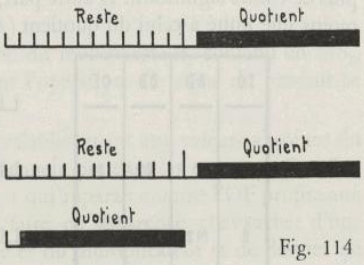
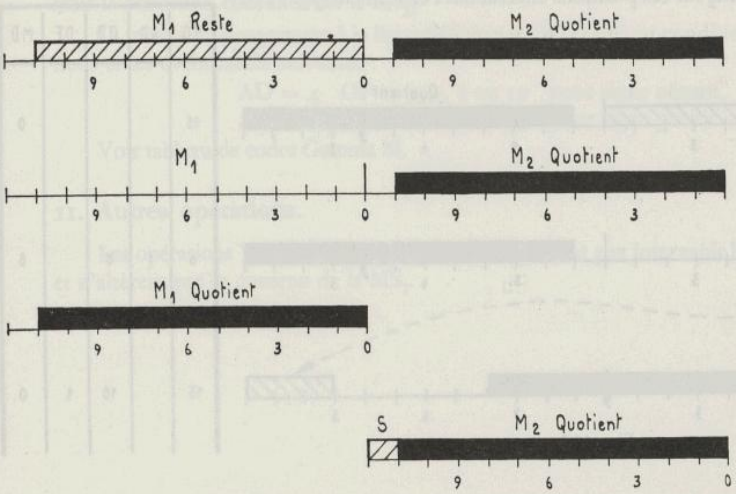


Fig. 114

B. *Par décalages* $M_2 - M_1$.

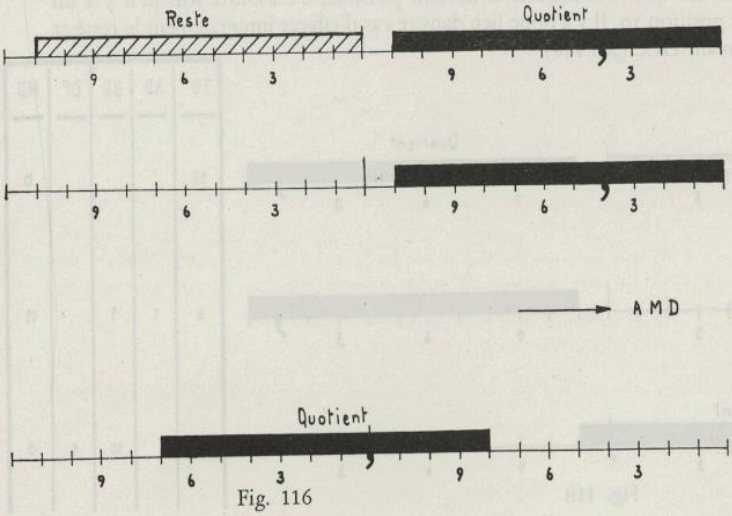
Un décalage, obtenu par division complète, permet de faire venir en M_1 le quotient contenu en M_2. Bien entendu il faut, avant de commander une telle division, effacer le reste qui serait traité comme dividende (*fig.* 115).



TO	AD	OD	OF	MD
15				0
3	1			0
15		10	1	0
8	2			0

Fig. 115

Si le quotient possède une virgule et que l'on désire n'amener en M_1 que les décimales à gauche de la virgule, on fera en sorte de positionner la MD pour qu'elle n'amène que cette partie en M_1 (*fig.* 116).

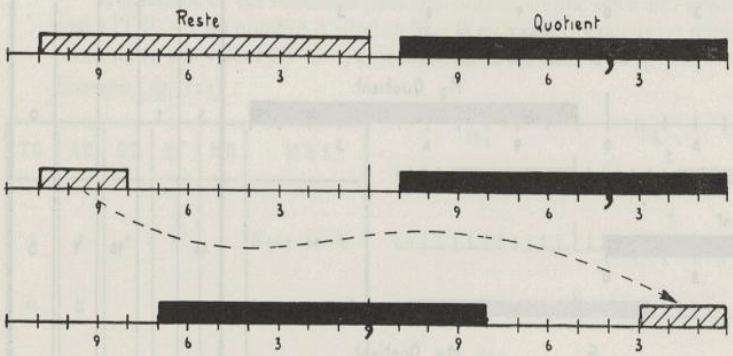


TO	AD	OD	OF	MD
15				0
3	1			0
2		8		8
15		10	1	0

Fig. 116

La ZB a été remplacée par une BO AD = 1 afin d'effacer le reste sans effacer le signe en MS_1.

On voit que si les décimales après la virgule étaient plus nombreuses, il faudrait afficher sur la BO un OD plus faible qui conserverait en M_1 une partie du reste, qui, au cours de la DC serait divisé par 1 et arriverait en M_2 : ce n'est, en général, pas gênant puisque tout nouveau nombre arrivant en M_2 sur ces positions effectuerait l'effacement (*fig.* 117).



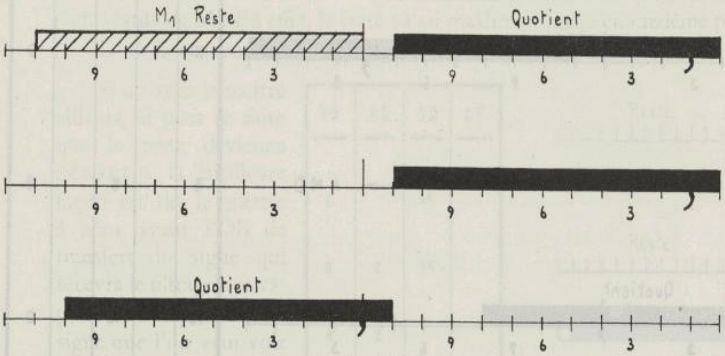
TO	AD	OD	OF	MD
15				0
6	1	8		8
15		10	1	0

Fig. 117

Au cours de cette division de regroupement en M_1, il n'y a pas de danger de changer le signe en MS_1 : l'AD zéro ne traite que des nombres positifs et le signe reste inchangé en MS_1.

Ces affichages 15, 0, 10, 1 sont valables aussi bien sur un Gamma mathématique pourvu de la DCC que sur une machine dépourvue de ce dispositif.

Toutefois, la dernière possibilité, celle où le reste demeure en partie en M_2, est exclue : en effet, la DCC s'arrête dès que la soustraction devient possible, c'est-à-dire lorsqu'il y a un chiffre significatif en position 10. Il y a donc lieu dans ce cas d'effacer intégralement le reste et de faire une AMD avant DCC (*fig.* 118).



TO	AD	OD	OF	MD
15				0
6	1	1		11
15		10	1	0

Fig. 118

Cette séquence est valable en DC si l'on est gêné par l'arrivée de la partie forte du reste en M_2.

10. Exploration de la MS_1 : Variantes MS + et MS —.

La variante mémoire signe (VMS) a la même action sur le programme que les VC eu VR, mais la condition consultée est la MS_1.

Le renvoi du programme à la ligne définie par OD et OF est conditionné par les instructions et les éventualités suivantes :

AD = 4 OF = 0, 4, 8 ou 12 pour signe négatif,
AD = 4 OF = 1, 5, 9 ou 13 pour signe positif.

Voir tableau de codes Gamma M.

11. Autres opérations.

Les opérations VC, VR, VPM, AMD, ZB, IS ne font pas intervenir l'opérateur de signes et n'altèrent pas le contenu de la MS_1.

Chapitre huitième

CALCULATEUR GAMMA 3 B

I. DÉFINITION

Le Gamma 3 B, dérivé des types précédemment décrits, présente les caractéristiques suivantes :

— unification du type (scientifique et comptable),
— extension de la capacité en mémoires.
— extension du programme,
— plus grande souplesse d'utilisation.

Le Gamma 3 B peut être utilisé indifféremment pour les problèmes comptables et les problèmes scientifiques. Il peut recevoir tous les dispositifs propres au Gamma 3 M.

Il est possible de lui connecter une armoire d'extension programme (A. E. P.) et de 1 à 3 armoires supplémentaires, afin d'étendre ses capacités en lignes de programme et en mémoires.

La description de ses connecteurs va permettre d'énumérer les autres caractéristiques de ce Calculateur (Voir Pl. 19 : connecteur Gamma 3 B).

1. Commandes d'introduction et d'extraction.

A. *Commandes d'introduction cinématique.*

Le connecteur standard possède deux plots de commande intitulés :

I : 9 à 1 permettant d'introduire de 9 à 1.
I : 9 à 0 permettant d'introduire de 9 à 0.

Remarque : l'introduction des données est invalidée par circuit interne au cours des cycles à vide.

B. *Commandes d'introduction statique.*

Ces commandes sont au nombre de 4, 2 sur chaque connecteur.

C. *Commandes d'extraction.*

Le connecteur standard possède 1 plot de commande intitulé : E : 9 à 1 permettant d'extraire de 9 à 1 (pour extraire de 9 à 11, il est possible d'utiliser les commandes de cycles de 9 à 11).

Les extracteurs peuvent être validés par n'importe quel émetteur. En particulier il est possible de valider un point en utilisant le distributeur au point précédent.

Remarque : L'extraction des données est invalidée par circuit interne au cours des cycles à vide.

2. Introduction des conditions de sélection.

Le connecteur standard possède :

— 8 commandes S : 0 à 7, dont 2 (S_0 et S_1) ont leur maintien apparent,

— 4 commandes V : 0 à 3, dont 2 (V_0 et V_1) ont leur maintien apparent.

Le connecteur supplémentaire possède :

— 8 commandes S : 8 à 15, dont 2 (S_8 et S_9) ont leur maintien apparent,

— 4 commandes V : 4 à 7, dont 2 (V_4 et V_5) ont leur maintien apparent.

Toutes ces commandes sont à retombée commandée. Elles émettent dès l'appel et cessent d'émettre à la fin de l'impulsion de commande de la retombée. Il est évident que les impulsions d'appel de V ou S d'un groupe ne doivent jamais être présentes lors de la fin de l'impulsion de retombée de ce groupe. Le maintien de ces commandes est ainsi splitté :

— connecteur standard :

la paire de plots référencée R : 0 à 3 permet de commander la retombée des V : 0 à 3 et des S : 0 à 3 ;

la paire de plots référencée R : 4 à 7 permet de commander la retombée S = 4 à 7.

— connecteur supplémentaire :

la paire de plots référencée R : 4 à 7, 8 à 11 permet de commander la retombée des V : 4 à 7 et des S : 8 à 11,

la paire de plots référencée R : 12 à 15 permet de commander la retombée des S : 12 à 15.

3. S.R.C.

Le Gamma 3 B possède 8 SRC : 4 sur le connecteur standard (0 à 3), 4 sur le connecteur supplémentaire (4 à 7). Chacune comprend : 1 plot « entrée », 1 plot « sortie » et un plot « retombée ». Ces commandes émettent dès l'appel jusqu'à la fin de l'impulsion de retombée.

4. Plot 15.

En calcul comptable, ce plot permet de faire retomber les commandes de sélection en fin de programme.

Son temps d'émission BS : 14,18 — 15,18

PRD : 14,18 — 16,04 permet le maintien des commandes de sélection à l'arrêt de la machine.

L'émission de ce plot est annulée par circuit interne en PPA et en PIA afin de maintenir les commandes de sélection pendant les cycles de mise au point en pas à pas automatique.

Sur BS, elle est également annulée lorsqu'il n'a pas été lancé de cycle électrique (arrêt-début de cycle à vide).

En calcul scientifique (PPC), le plot 15 peut être utilisé pour appeler les commandes entre la lecture des cartes.

Sur BSB ce plot émet pendant l'arrêt de la machine.

5. Plot 12.

Ce plot (11,18-12, 18) permet de commander la retombée des commandes de sélection en PPC .Il n'émet pas au cours du cycle de lancée BS.

6. Plot SC.

Ce plot récepteur permet la suppression du calcul en suspendant le déroulement des trains. Son action dure le temps de l'impulsion de commande qui peut être une impulsion courte ou une impulsion longue.

En problème comptable, passage d'une carte Nom, par exemple, il suffit d'alimenter ce plot pendant le point correspondant à la VPM pour annuler le programme. En PPC, il permet d'éviter le calcul pendant la lecture des cartes données.

7. Plots AS.

Ces deux plots récepteurs permettent de commander la commutation des armoires supplémentaires.

8. Alternatifs standard.

Le connecteur standard et le connecteur supplémentaire possèdent chacun 8 alternatifs « petits relais » à 1 position.

9. Opération KB.

Lorsque l'on doit extraire un code indicatif, l'opération KB d'introduction de ce code dans la mémoire à extraire doit être terminé au moins 10^0 avant le point à extraire (ex. 11,14 pour 12).

II. L'ARMOIRE D'EXTENSION PROGRAMME

1. Organisation.

L'armoire d'extension programme (AEP) est connectée au Gamma par l'intermédiaire d'un câble souple aboutissant à un connecteur, fixé sur le bâti du Calculateur.

L'armoire est dotée d'un tableau de connexion permettant d'afficher les 64 lignes de programme supplémentaires (Pl. 20).

Ce tableau comprend :

A. 64 lignes de programme numérotées de 0 à 63. Pour chaque groupe de 16 lignes, un distributeur de codes de 1 à 15 est prévu comme sur le tableau du Gamma 3.

B. 72 alternatifs « petits relais », avec leurs plots de commande. Ces alternatifs, réservés à la sélection des codes affichés, sont ainsi splittés :

24 à 1 position = 24
4 à 2 positions = 16
4 à 4 — = 8
4 à 6 — = 24.

C. 4 distributeurs de 0 à 15, affectés aux alternatifs suivants :

Distributeurs	1er	2e	3e	4e
Alternatifs . . .	0 à 15	16 à 27	28.29.32.33	30.31.34.35

D. Les sorties de 16 commandes S et des 8 commandes de sélections à retombée commandée (SRC) du Gamma. Chacune de ces sorties ne doit pas contrôler plus de 20 positions d'alternatifs.

2. Utilisation.

A. *Lignes de programme.*

Les lignes de programme de l'armoire extension calcul sont numérotées de 0 à 63. Le type d'opération variante est de fait double :

a) TO = 0 définit les variantes renvoyant aux lignes du tableau normal du Gamma.

b) TO = 1 définit les variantes renvoyant aux lignes du tableau de l'AEP.

Les codes AD, OD, OF, se déterminent de la même façon à partir des mêmes conditions, dans les deux cas, au TO près.

La ligne 63 du Tableau normal du Gamma renvoie systématiquement à la ligne 0 de l'armoire extension et la ligne 63 de l'armoire extension renvoie à la ligne 0 du tableau normal du Gamma.

B. *Mémoires.*

Il n'est pas prévu d'introducteur pour les mémoires 8 à 15. Par contre, il est possible d'extraire des mémoires 8, 9 et 10. L'affectation des extracteurs, variable selon le type de machine est précisée dans la note d'information nº 179 (capacités intermédiaires — affectation des introducteurs et des extracteurs).

3. Capacités intermédiaires.

Deux capacités de programme sont prévues : 32 ou 64 lignes venant s'ajouter aux 64 lignes de l'armoire standard.

Le nombre de mémoires peut varier entre 0 et 8 s'il y a extension du programme (32 ou 64 lignes), et entre 2 et 8 s'il n'y a pas extension du programme.

Remarques :

a) A chacun des 4 distributeurs de 0 à 15 a été affecté un certain nombre d'alternatifs dont les 4 zones sont délimitées sur le tableau de connexion. La sélection des codes de chacun de ces distributeurs ne doit se faire qu'à l'intérieur du groupe qui lui est affecté.

b) *Connexions prohibées.* Le tableau de connexion de l'AEP est séparé en deux parties par un « gêneur ». Pour des raisons technologiques, il est rigoureusement interdit de relier les deux parties du tableau, quelle que soit la connexion envisagée. Une telle connexion risque de provoquer une détérioration du matériel.

III. ÉTABLISSEMENT DES CONNEXIONS

1. Affichage.

Les codes définiteurs des opérations et des nombres qu'elles traitent sont émis par quatre distributeurs du tableau Gamma à raison d'un distributeur par quadrant de 16 lignes. Ces 4 distributeurs sont référencés de 1 à 15 quant à leurs plots : il n'y a pas de plot « zéro ».

Il est indispensable de ne pas afficher les codes « zéro ». En effet, les mémoires programme du Gamma sont antiparasitées jusqu'aux plots du tableau sous lesquels elles apparaissent et toute fiche qui serait plantée dans un de ces plots sans aboutir à un distributeur de code réel se comporterait un peu comme une antenne capable d'introduire des codes sans signification pour la machine. Comme il n'y a pas de distributeur de « zéro », il est évident qu'une fiche, posée sur des plots devant enregistrer un zéro, se trouverait constituer l'une de ces sortes d'antenne.

Toutefois, et pour la même raison, cette règle de ne pas afficher les codes zéro ne s'applique qu'aux connexions ne subissant pas de sélection sur les alternatifs S.

Soit, en effet, une adresse, égale à 5 dans le cas où la commande S_0 est utilisée, et égale à 0 quand la commande S_0 n'est pas utilisée. Si l'on n'affiche pas le zéro, la connexion est celle de la figure 119.

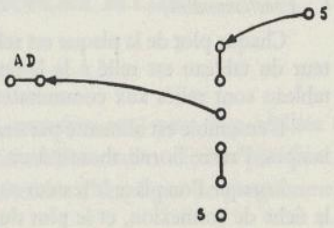


Fig. 119

Dans ces conditions, lorsque l'alternatif sera au travail, il établira la liaison entre le 5 du distributeur et le plot d'AD. Mais quand il sera au repos, il n'y aura plus que la fiche qui réunit le commun de l'alternatif aux plots d'AD, qui constituera une antenne.

Pour cela, il existe, sur le distributeur propre aux alternatifs, un plot intitulé « 0 » qui émet une tension appropriée que l'on doit connecter au plot de programme par l'intermédiaire, ici, du non-sélectionné de S_0 (*fig.* 120).

On voit ainsi que toute sélection de programme par les alternatifs des commandes S doit être opérée par manière de regroupement et jamais par manière de ventilation. En effet, en ventilation, il y aurait toujours l'une des mémoires, celle qui n'est pas sélectionnée, qui serait parasitée (*fig.* 121).

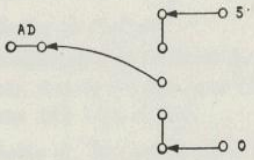


Fig. 120

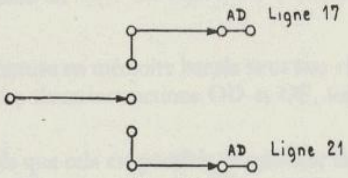


Fig. 121

2. Contrôleurs de tableaux gamma.

A. *But.*

Cet appareil est destiné à contrôler les connexions réalisées sur un tableau Gamma.

Ce contrôle a un double but :

— Vérifier que les connexions sont conformes au tableau de développement résultant de l'étude du problème.

— Vérifier que les connexions n'établissent aucun court-circuit entre les codes provenant du distributeur, de telles connexions ayant pour effet une détérioration du Calculateur.

B. *Description.*

L'appareil se présente sous la forme d'un pupitre au dos duquel se fixe le tableau à vérifier (Pl. 21).

La face avant (voir Pl. 22) est constituée d'une plaque à plots représentant l'image de la plaque gravée du tableau. Les alternatifs sont remplacés par des commutateurs. Un commutateur correspond aux deux positions d'alternatifs de chacune des commandes S.

Deux rangées de lampes représentent les distributeurs de codes ; la rangée supérieure affectée aux distributeurs standard, la rangée inférieure affectée au distributeur spécialisé pour les sélections par alternatifs.

C. *Fonctionnement.*

Chaque plot de la plaque est relié au plot du tableau correspondant. Chaque plot distributeur du tableau est relié à la lampe portant le même numéro. Les plots des alternatifs du tableau sont reliés aux commutateurs.

L'ensemble est alimenté par une prise 48 v. Une borne de la prise alimente le commun des lampes, l'autre borne aboutit à un testeur.

Lorsque l'on place le testeur sur un plot, un circuit s'établit par ce plot, le plot du tableau, la fiche de connexion, et le plot du distributeur pour arriver à la lampe.

— Si la connexion est normale, une lampe rouge s'allume, son numéro indique le code affiché pour l'instruction considérée.

— Si la connexion est sélectionnée, une lampe verte s'allume ; en modifiant la position du commutateur sélectionnant cette instruction, une autre lampe verte s'allume. Les numéros de ces deux lampes indiquent les codes regroupés par cet alternatif.

Dans ces deux cas, les codes indiqués par les lampes doivent être identiques à ceux inscrits sur le tableau de développement.

Si la connexion est erronée, c'est-à-dire si deux ou plusieurs codes sont en court-circuit, l'alimentation d'un plot provoque l'allumage de deux ou plusieurs lampes.

Dans ce cas, l'erreur est signalée, mais la cause doit en être retrouvée dans les circuits du tableau qui sont à reconsidérer.

Chapitre neuvième

LES FONCTIONS PRINCIPALES

Chaque fonction du programme (sélection, calcul, transfert, etc.) peut être réalisée :
— soit par la simple application de l'une des opérations élémentaires décrites plus haut,
— soit par une combinaison particulière des quatre codes d'une ligne de programme,
— soit par une succession de plusieurs opérations.

L'objet du présent chapitre est de regrouper par fonctions principales toutes les combinaisons caractéristiques de codes ou de lignes. L'application de ces séquences facilitera la résolution des problèmes, dont l'analyse aboutit toujours, comme on le verra, à la distinction de fonctions successives à mettre en œuvre.

I. LA SÉLECTION

1. La sélection des instructions d'une ligne.

La sélection d'une instruction TO, AD, OD, ou OF d'une ligne de programme est effectuée à l'aide des alternatifs liés aux commandes S.

A. *Modification de l'adresse.*

La modification de l'adresse du terme contenu en mémoire banale peut être effectuée par la sélection, soit de l'instruction AD, soit des deux instructions OD et OF, soit des trois instructions AD, OD, et OF.

a) *Modification de l'instruction AD :* chaque fois que cela est possible, il convient de conserver les termes à sélectionner dans des mémoires différentes, avec des OD semblables et des OF semblables. Il suffit alors de sélectionner la seule instruction AD.

Soit, par exemple, un montant M introduit en mémoire 1, à additionner aux cumuls D ou C, selon que A est positif ou négatif (ventilation débit-crédit) :

D est en M_2 OD = 0 OF = 10
C est en M_3 OD = 0 OF = 10

Le type d'opération est AN = 10, OD = 0, et OF = 10.

L'adresse est définie par le signe de A :

— si A est négatif, l'adresse est AD = 3,

— si A est positif, l'adresse est AD = 2.

Lorsque la perforation symbolisant le signe négatif de A est lue, l'impulsion correspondante appelle une commande S (par exemple S_1).

L'un des deux alternatifs contrôlés par cette commande peut être utilisé pour regrouper les codes 3 et 2 sur le plot AD de la ligne utilisée pour cette opération (*fig.* 122).

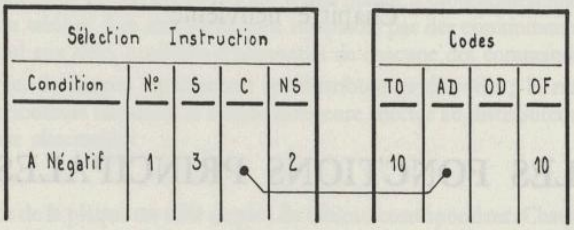


Fig. 122

b) *Modification des deux instructions OD et OF :* lorsque la somme des nombres de chiffres des termes à sélectionner est inférieure ou égale à 12, il peut être préférable de grouper ces termes dans la même mémoire. Il suffit alors de sélectionner les codes OD et OF.

Soit, par exemple, trois prix unitaires (P_1, P_2, P_3) de quatre chiffres chacun, lus dans une carte barème et introduits en mémoire 3. Le choix de l'un de ces prix est fonction d'un code de la carte détail :

S'il y a code 1,	choisir P_1	OD = 0	OF = 4
— — 2,	— P_2	OD = 4	OF = 8
— — 3,	— P_3	OD = 8	OF = 0

(Il y a toujours 3 codes).

Le regroupement des 3 codes OD (0, 4, 8) est réalisé par 2 alternatifs en cascade, contrôlés chacun par une commande de sélection. Les 3 codes OF sont regroupés par les deux autres alternatifs contrôlés par ces commandes (fig. 123).

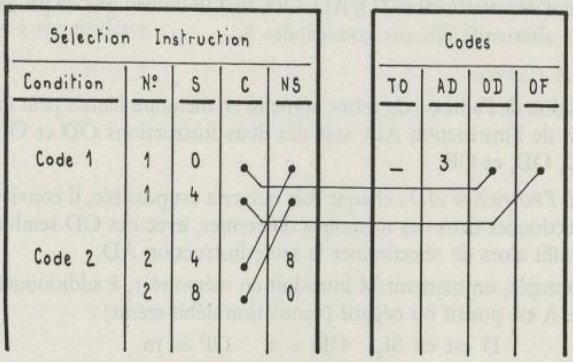


Fig. 123

c) *Modification des 3 instructions AD, OD et OF.* Lorsque les deux solutions précédentes ne sont pas possibles, il faut sélectionner les 3 instructions AD, OD, et OF.

Cette solution, nécessitant des positions d'alternatif pour chaque instruction AD, OD et OF, peut souvent être évitée par une implantation des termes bien étudiée.

B. *Modification de la fonction d'une ligne.*

La fonction d'une ligne de programme peut être modifiée en agissant :

— soit sur le TO.

— soit sur l'AD ou l'OF pour les variantes.

a) *Modification de l'instruction TO.* Une application courante de cette possibilité de sélection est le choix entre les types d'opérations AN et SN :

Soit à effectuer l'addition algébrique A $\pm$ B (A est toujours positif).

Le type d'opération est défini par le signe de B :

— si B est négatif, le type d'opération est SN = 11,

— si B est positif, le type d'opération est AN = 10.

Lorsque la perforation symbolisant le signe négatif de B est lue, l'impulsion correspondante appelle une commande S (par exemple S_0).

L'un des deux alternatifs contrôlés par cette commande peut être utilisé pour regrouper les codes 11 et 10 sur le plot TO de la ligne consacrée à l'opération A $\pm$ B (*fig.* 124).

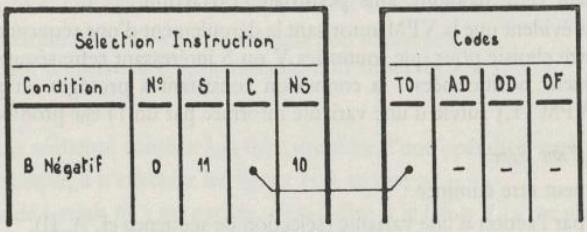


Fig. 124

b) *Modification de l'instruction AD d'une variante.* Une variante comparaison est transformée en variante systématique lorsqu'à l'AD 1, 2 ou 3 est substitué l'AD = 0.

Soit, par exemple, une séquence dont l'utilisation est subordonnée au sens d'une comparaison, sauf si une condition C est lue dans la carte, auquel cas la séquence n'est jamais à faire.

— Si la condition C est remplie, la variante est « systématique » et l'AD = 0.

— Si la condition C n'est pas remplie, la variante consulte normalement le comparateur et l'AD est égale à 1, 2 ou 3, suivant le cas.

Le regroupement des codes adresse 0 et 1, 0 et 2, ou 0 et 3 est réalisé comme précédemment à l'aide d'un alternatif excité lorsque la condition C est présente.

c) *Modification de l'instruction OF d'une variante.* Le code OF complétant le code AD pour définir le conditionnement de la variante, la sélection de ce code peut modifier l'action de cette variante.

Soit, par exemple, une variante systématique (TO = 0, AD = 0), qui envoie à la ligne définie par les codes OD et OF :

— « toujours » lorsque le code OF est *impair*,

— « jamais » lorsque le code OF est *pair*.

d) *Cas particulier de la VPM.* Il peut être intéressant de sélectionner la ligne de renvoi d'une *VPM* en fonction d'un code carte. Cette sélection est possible mais elle ne peut porter que sur le code OD.

En effet, ainsi qu'on l'a vu relativement à la VPM, le type de variante est défini par l'ensemble des 3 codes TO = 0, AD et OF. Seul le code OD n'est pas définiteur du type de variante et c'est à ce titre qu'il peut seul, ici, être sélectionné.

En effet, il est impossible de sélectionner une instruction quelconque pendant que l'on se sert de l'opération en résultant. Or, la seule opération du programme dont on se serve pendant les points de lecture est précisément l'opération VPM dont le rôle est de bloquer le programme pendant ces points.

Si l'on considère d'autre part la formule de renvoi en variante :

$$NL = 4\,OD + \frac{OF}{4},$$

on peut voir que la modification d'une unité de la mémoire OD entraîne une modification du numéro de ligne de renvoi de 4 unités. Si donc une VPM, en absence d'excitation de la commande S, par exemple renvoie en ligne 1, elle pourra renvoyer, en cas d'excitation de S, par un code, sur l'une des lignes 5, 9, 13, 17, etc.

— En cas de VPM 1, 14 si l'extraction d'une mémoire est validée pour l'extraction des zéros, il est évident qu'il est impossible d'effectuer des opérations OB-KB ou ZB affectant cette mémoire sans perturber l'extraction.

— Il est évident que la VPM autorisant le déroulement d'une séquence de programme *doit* être choisie pour que toutes les V ou S intéressant cette séquence soient préalablement positionnées : la connexion consistant à protéger un programme par une VPM 11,3 suivie d'une variante informée par un 11 est prohibée.

C. *Élimination d'une ligne.*

Une ligne peut être éliminée :

— soit par l'action d'une variante (sélection de séquence cf. A, II),

— soit par la sélection de l'une de ses instructions TO ou AD.

a) *Modification de l'instruction TO.* L'élimination est obtenue par la substitution au code TO prévu d'un code d'une opération inexistante (1 ou 7), ou d'une opération qui n'altère pas le contenu des mémoires, ni n'affecte la suite des calculs (exemple : CN).

b) *Modification de l'instruction AD.* L'élimination est obtenue par la substitution au code AD prévu du code d'une mémoire inexistante dans la machine, ou inutilisée dans le problème.

— Les opérations avec AD = 8 sur Gamma sans Armoire supplémentaire sont autorisées. Ces opérations sont formellement interdites si le Gamma possède une armoire même incomplète.

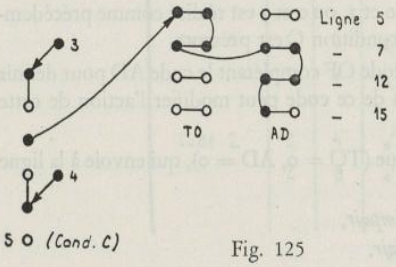


Fig. 125

D. *Réutilisation des alternatifs.*

Lorsqu'une même condition commande le regroupement de 2 codes sur différentes instructions du programme, il suffit d'une seule position d'alternatif contrôlé par cette condition.

Par exemple : si les codes TO des lignes 5, 7 et AD des lignes 7, 12 et 15 sont :

— tantôt 3, lorsque la condition C est présente,

— tantôt 4, lorsque la condition C est absente,
les codes 3 et 4 sont regroupés par un seul alternatif, dont le commun est relié aux plots TO des lignes 5 et 7, et AD des lignes 7, 12 et 15 (*fig.* 125).

2. La sélection des séquences de programme.

La sélection d'une séquence est effectuée à l'aide d'une opération variante (TO = 0) qui subordonne la progression normale du numéro de ligne à l'observation d'un certain fait « e ».

On définit un fait « $\overline{e}$ » (lire « e barre ») par :

— si « e » est réalisé, « $\overline{e}$ » ne l'est pas; si « e » n'est pas réalisé, « $\overline{e}$ » l'est.

On dit que « e » et « $\overline{e}$ » sont deux faits complémentaires.

Il convient aussi de dire que la variante affectée à la ligne L observe :

— le fait « e », si le code OF définissant la ligne L est pair;

— le fait « $\overline{e}$ », si le code OF définissant la ligne L est impair.

Cette variante envoie à la ligne dont le numéro est défini par les codes OD et OF, lorsque le fait qu'elle observe est réalisé.

Et la progression du NL est normale lorsqu'est réalisé le fait complémentaire du fait observé par la variante.

Il résulte que si l'utilisation d'une séquence est subordonnée :

— à la réalisation d'un fait « e », la variante doit observer « $\overline{e}$ »,

— à l'absence d'un fait « e », la variante doit observer « e ».

A. *Élimination d'une séquence.*

Éliminer une séquence consiste à la faire précéder d'une opération variante.

Soit, par exemple, à n'effectuer les lignes 15 à 20 que :

a) Si la commande à relais n° 5 est excitée, c'est-à-dire si le fait « V_5 » est réalisé.

L'opération VR est affichée à la ligne 14 : TO = 0, AD = 5.

La variante envoie à la ligne 21 définie par les codes OD et OF de la ligne 14, quand le fait complémentaire de V_5, soit $\overline{V}_5$ est observé.

Le tableau de correspondance des NL et des codes OD-OF permet de définir les codes OD et OF : OD = 5, OF relatif à $\overline{V}_5$ est impair et égal à 7 (*fig.* 126).

Sélection Lignes	NL	Codes			
		TO	AD	OD	OF
	14		5	5	7
	15				
	20				
	21				

Fig. 126

Sélection Lignes	NL	Codes			
		TO	AD	OD	OF
	14		5	5	6
	15				
	20				
	21				

Fig. 127

B. *Sélection de plusieurs séquences.*

a) *Méthode générale.* Soit à effectuer, par exemple, les séquences A, B, C..., lorsque les faits correspondants *a*, *b*, *c*, sont réalisés.

Chaque séquence est précédée d'une variante qui l'élimine lorsque le fait complémentaire correspondant est réalisé :

— $\overline{V}_a$ précède A, l'élimine si *a* est absent et envoie à $\overline{V}_b$,

— $\overline{V}_b$ précède B, l'élimine si *b* est absent et envoie à $\overline{V}_c$,

— etc...

La figure 128 représente ces alternatives :

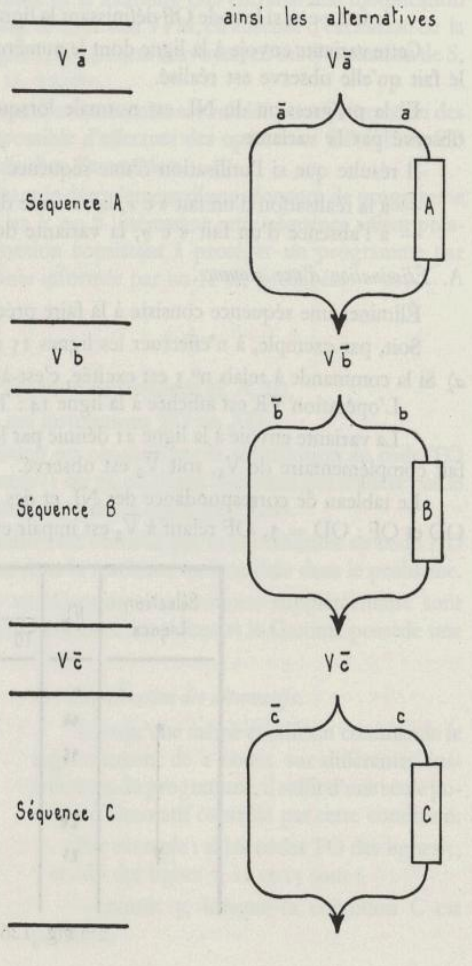
b) *Priorité d'une séquence sur les précédentes.* Lorsque chaque séquence annule l'effet de la précédente, c'est-à-dire lorsque les résultats se trouvent eux-mêmes placés dans les mêmes mémoires à la fin de chacune des

b) Si la commande à relais nº 5 n'est pas excitée, c'est-àdire si le fait $\overline{V}_5$ est réalisé.

L'opération VR affichée à la ligne 14 (TO = 0) consulte la commande à relais nº 5 (AD = 5) pour envoyer à la ligne 21, quand le fait complémentaire de $\overline{V}_5$, est observé, OD = 5. L'OF relatif à V_5 est pair et égal à 6 (*fig.* 127).

Fig. 128

On peut représenter ainsi les alternatives



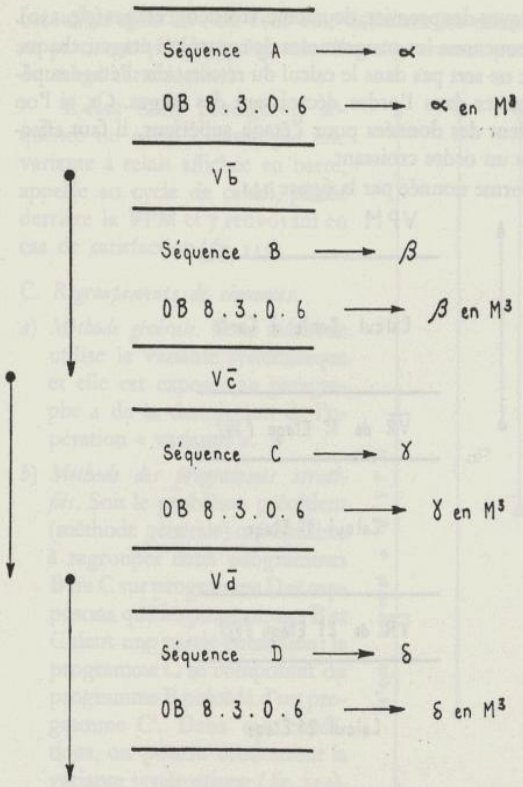


Fig. 129

séquences, on peut appliquer la méthode générale : toutefois, la première variante est superflue.

Soient les séquences A, B, C, D, subordonnées au fait *a*, *b*, *c*, *d* calculant respectivement les résultats α, β, γ, δ, tous à extraire de la M_3 de 0 à 6 (*fig.* 129).

On voit que α sera toujours calculé. Si l'on a ensuite $\bar{b}$, $\bar{c}$, $\bar{d}$, les séquences B, C, D, seront sautées et α restera en M_3. Si l'une des séquences B, C ou D a lieu, le terme β, γ ou δ sera calculé et viendra se substituer à α précédemment calculé.

c) *Sélection de séquences hiérarchisées.* Soit à sélectionner des séquences relatives à des étages interdépendants et utilisées dans l'ordre décroissant de ces numéros d'étages (par exemple séquences de remise à zéro).

Une seule VS placée au début de cette série de séquences suffit à choisir la première des séquences à utiliser. Les codes OD et OF qui définissent le NL sont sélectionnés à l'aide de commandes S. Ces commandes S_1, S_2, S_3... sont respectivement ap-

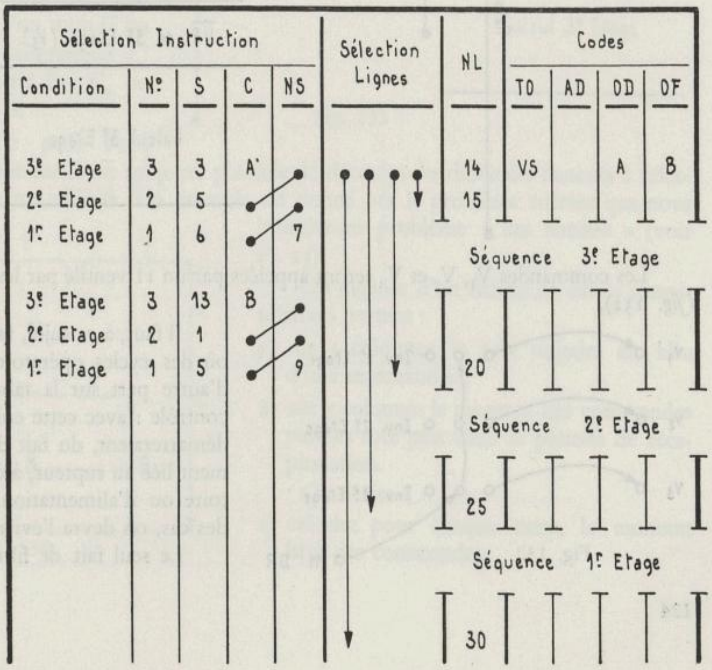


Fig. 130

pelées par les impulsions caractéristiques des premier, deuxième, troisième étages (*fig.* 130).

Cette séquence est convenable pour tous les programmes de contrôle à étages, chaque fois que le résultat calculé à un étage ne sert pas dans le calcul du résultat du l'étage supérieur. En effet, on effectue les séquences dans l'ordre décroissant des étages. Or, si l'on veut que les résultats d'un étage soient des données pour l'étage supérieur, il faut effectuer les séquences hiérarchisées dans un ordre croissant.

Le programme sera alors de la forme donnée par la figure 131.

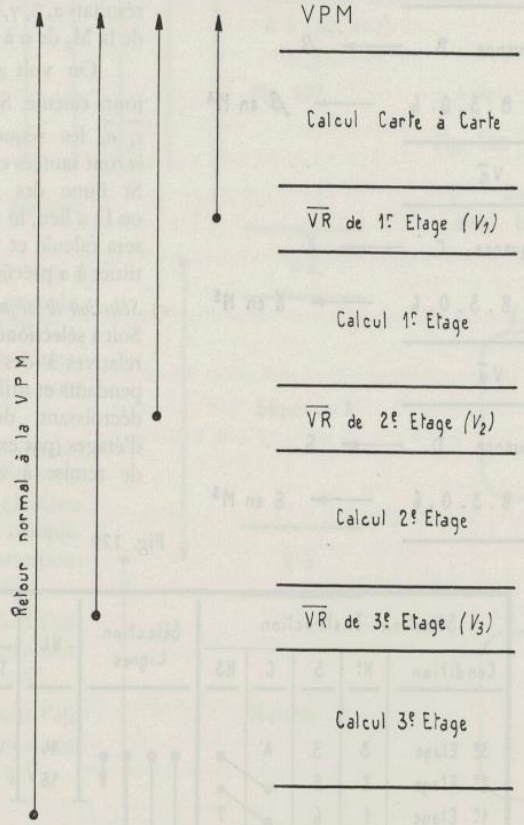


Fig. 131

Les commandes V_1, V_2 et V_3 seront appelées par un 11 ventilé par les inverseurs d'étage (*fig.* 132).

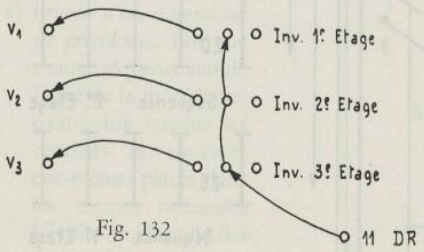


Fig. 132

Il faut, à ce sujet, faire attention au cas où des cycles opératoires sont déclenchés d'autre part sur la tabulatrice du fait du contrôle : avec cette connexion les calculs démarreraient, du fait de la VPM uniquement liée au rupteur, à chaque cycle opératoire ou d'alimentation. Dans la plupart des cas, on devra l'éviter.

Le seul fait de filtrer le 11 du distri-

buteur au cycle où l'on veut voir démarrer les calculs d'étages assurera que ces séquences ne partiront qu'à ce cycle, mais elles n'assureront pas le démarrage pour un cycle seulement du calcul carte à carte.

Il faut donc protéger la séquence de carte à carte par une variante à relais affichée en barre, appelée au cycle de calcul, placée derrière la VPM et y renvoyant en cas de satisfaction (*fig.* 133).

C. *Regroupements de séquences.*

a) *Méthode générale.* Cette méthode utilise la variante systématique et elle est exposée au paragraphe 4 de la description de l'opération « variante ».

b) *Méthode des programmes stratifiés.* Soit le problème précédent (méthode générale) qui consiste à regrouper deux programmes B ou C sur programme D et supposons que les programmes B et C aient une partie commune : le programme C se composant du programme B précédé d'un programme C'. Dans ces conditions, on pourra économiser la variante systématique (*fig.* 134).

Selon que l'on aura C ou $\overline{C}$, on fera le programme C' suivi de B, c'est-à-dire C, ou bien seulement B.

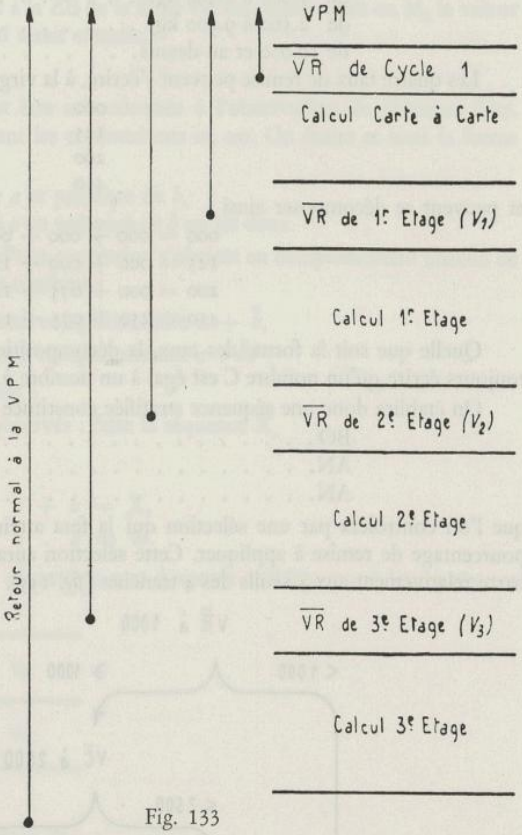


Fig. 133

Cette méthode peut s'appliquer fréquemment car il est toujours possible de décomposer différents facteurs à sélectionner en constituants additifs. Un exemple est donné par le problème suivant que nous intitulerons problème « des remises » (voir Pl. 23).

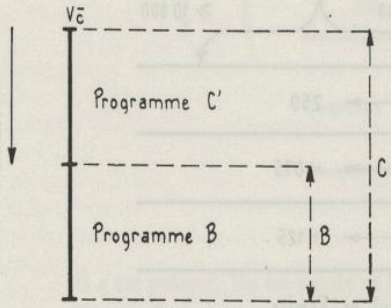


Fig. 134

On dispose d'un fichier de cartes récapitulatives portant :

α) sur 3 colonnes le prix unitaire au kilo d'une marchandise,

β) sur 5 colonnes le montant des commandes passées à ce prix dans la période de récapitulation.

On désire :

α) calculer pour chaque carte, le montant brut des commandes,

β) calculer la remise à effectuer pour chaque carte. La loi d'établissement de la remise est la suivante :

de	0 à 999 kg.	Pas de remise,
de	1.000 à 2.499 kg.	12,50 % de remise,
de	2.500 à 9.900 kg.	20 % de remise,
de	10.000 et au-dessus.	45 % de remise.

Les quatre taux de remise peuvent s'écrire, à la virgule près :

000
125
200
450

et peuvent se décomposer ainsi :

$$000 = 000 + 000 + 000,$$
$$125 = 000 + 000 + 125,$$
$$200 = 000 + 075 + 125,$$
$$450 = 250 + 075 + 125.$$

Quelle que soit la forme des taux, la décomposition est toujours possible car on peut toujours écrire qu'un nombre C est égal à un nombre *b* plus leur différence *a*.

On établira donc une séquence stratifiée constituée par une BO et deux AN :

BO.	250,
AN.	+ 75,
AN.	+ 125,

que l'on contrôlera par une sélection qui la fera atteindre plus ou moins haut suivant le pourcentage de remise à appliquer. Cette sélection aura la forme VR, VC, VR affichées en barre relativement aux 3 seuils des 4 tranches (*fig.* 135).

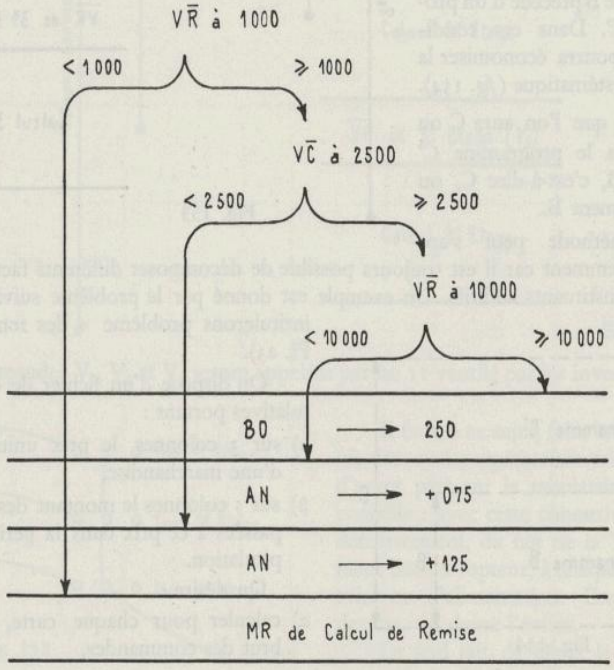


Fig. 135

Il faut, bien entendu, effacer préalablement la mémoire opérateur (ligne 14).

Il faut également remarquer que le cas de la remise nulle doit faire l'objet d'une séquence propre qui peut se réduire à l'OB de la ligne 12 ou comprendre également la MR de la ligne 11 : en effet, si l'on sautait immédiatement à la ZB de la ligne 13, on conserverait en M_3 la valeur de la remise de la carte précédente qui serait extraite.

D. *Conditions combinées.*

L'utilisation d'une séquence peut être subordonnée à l'observation de plusieurs faits. Ces faits se combinent toujours suivant les conjonctions et, ou. On écrira et sous la forme $\times$ et ou sous la forme $+$.

$\bar{a} . b$ s'énonce absence de a et présence de b,

$\bar{a} + b$ s'énonce absence de a ou présence de b ou les deux.

D'autre part, la complémentaire d'une expression s'obtient en complémentant chacun de ses facteurs logiques et le signe qui les combine :

$\bar{a} . b$ a pour complémentaire $a + \bar{b}$,

$\bar{a} + b$ a pour complémentaire $a . \bar{b}$.

a) *Combinaison « ou »*. Soit le problème :

— Si a ou b, ou les deux, sont observés : faire la séquence X,

— Sinon, faire la séquence W.

Ce qui s'écrit :

$$a + b = X,$$
$$\bar{a} . \bar{b} = W.$$

Deux variantes consécutives observant a et b seront validées pour leur présence (*fig.* 136).

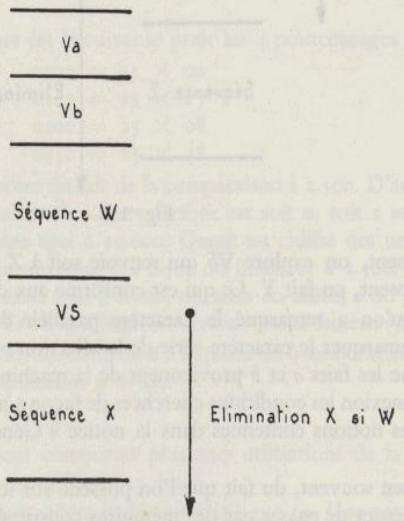


Fig. 136

Si a est présent, Va est validé et l'on fait X.

Si a est absent, on explore Vb qui si b est présent, on fait également X.

Seul le cas simultané d'absence de *a* et de *b* assure la séquence W, ce qui est conforme aux données.

Il est bon de remarquer que les deux renvois en séquence X sont assurés depuis V*a* et V*b* en parallèle, ce qui est conforme aux notions logiques générales exposées dans la notice « Généralités ».

b) *Combinaison « et »*. Soit le problème :

— si *a* et *b* sont observés, faire la séquence Z.

— sinon, faire la séquence Y.

Ce qui s'écrit :

$$a \cdot b = Z,$$
$$\bar{a} + \bar{b} = Y.$$

Deux variantes consécutives $\overline{V}a$, V*b* contrôlent cette sélection (*fig* 137

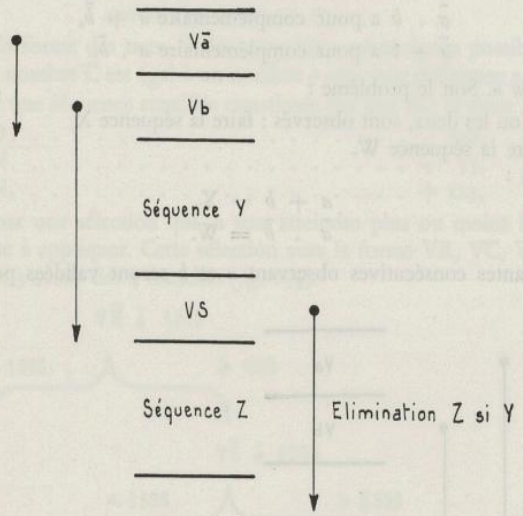


Fig. 137

Si *a* est présent, on explore V*b* qui renvoie soit à Z si *b* est présent, soit à Y si *b* est absent. Si *a* est absent, on fait Y. Ce qui est conforme aux données.

De même qu'on a remarqué le caractère parallèle de la sélection pour la condition « ou » on peut remarquer le caractère série de la sélection pour la condition « et ».

Remarque : lorsque les faits *a* et *b* proviennent de la machine connectée, il est préférable de composer par connexion les conditions cherchées de façon à n'utiliser qu'une seule commande à relais. Toutes les notions contenues dans la notice « Généralités » (logique des contacts) s'appliquent.

Toutefois, bien souvent, du fait que l'on possède sur le Gamma des alternatifs rapides, on pourra se dispenser de passer par des mémoires constituées par des commandes de sélection ou de sélection de solde et explorer par un 11 du distributeur des alternatifs mis au travail par les codes 11 observés. Mais il convient alors de remarquer que les alternatifs peuvent ne pas répondre tous au même degré du point et il est nécessaire de prendre la précaution suivante : l'impulsion d'appel doit toujours être contrôlée par au moins un sélec-

tionné d'alternatif, ce qui assure que la validation se fera toujours lorsque l'on est sûr que tous les contacts intéressés sont au travail.

Les deux montages suivants, relatifs aux conditions « et », « ou » sont à faire (*fig.* 138).

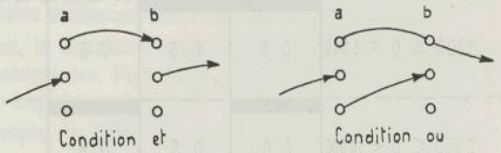


Fig. 138

E. *Sélection de séquences combinées avec sélection d'instruction.*

Il est toujours économique en lignes de confier les sélections aux commandes S chaque fois que les éventualités proviennent de la machine connectée. Mais, il peut se faire qu'une série de faits *a*, *b*, *c* soit telle que deux faits *a* et *c* par exemple, soient faciles à mettre en observation sur des commandes alors que le fait *b* n'est facile à mettre en œuvre que sur le Gamma par manière de variante comparaison. Il faut alors combiner les deux procédés.

L'exemple du problème des remises illustrera bien cette combinaison. Sa deuxième solution (voir Pl. 23) utilise ce procédé. On a confié à des commandes S l'étude des seuils à 1.000 et 10.000 et à une variante comparaison l'étude du seuil 2.500.

D'autre part, afin de profiter d'une conjugaison heureuse des valeurs des pourcentages de remise, on les a décomposés en facteurs communs. Il faut bien remarquer à ce sujet que cette méthode arithmétique ne présente pas le caractère général de celle qui décompose en éléments additifs. Elle est applicable pour des grandeurs, physiques par exemple, qui ne changent pas.

La décomposition en facteurs est la suivante pour les 4 pourcentages :

$$0000 = 25 \times 00$$
$$0125 = 25 \times 05$$
$$0200 = 25 \times 08$$
$$0450 = 25 \times 18$$

Le terme 25 est déjà en mémoire du fait de la comparaison à 2.500. D'autre part, on peut remarquer que le chiffre des dizaines de l'autre facteur est soit 0, soit 1 selon que l'on est plus petit que 10.000 ou au moins égal à 10.000. Quant au chiffre des unités, il est soit 8 soit 0 ou 5 suivant que l'on est au moins égal à 2.500 ou inférieur à 2.500. Enfin, selon que l'on est inférieur à 1.000 ou au moins égal à 1.000, on aura en unités 0 ou 5, ce 5 devra être éventuellement transformé en 8. D'où la sélection suivante sur ce facteur (*fig.* 139).

La substitution de 8 à 5 et l'introduction de 8 à la place du 0 sera obtenue par une KB à 8 contrôlée par une variante comparaison à 2.500.

F. *Remontée de lignes.*

Un programme de calcul peut comporter plusieurs utilisations de la même séquence. Soit à utiliser deux fois la séquence S (ligne 15 à 20).

a) *Avec les mêmes instructions.* On accédera deux fois à la ligne 15. Une première fois depuis la ligne 13 qui porte une VS toujours renvoyant en ligne 15. Une deuxième fois depuis la ligne 14.

On accède en ligne 13 depuis le programme précédent la séquence, et en ligne 14 depuis la fin de la séquence faite une première fois.

Q < 1000	0 0	0 0	0 0
1000 ≤ Q < 2500	0 5	0 5	0 5
2 500 ≤ Q < 10 000	0 8	0. 5	0 8
Q ≥ 10 000	1 8	1 0	1 8
	Facteurs à obtenir	Facteurs obtenus après sélection par les C^{des} S	Facteurs obtenus après KB à 8 (VC 2500)

Fig. 139

Lorsque la séquence a été faite à nouveau, on ne remonte pas en ligne 14 mais on effectue le programme suivant la séquence. Pour cela, la ligne 20 porte une variante comparaison qui exploite la CN de la ligne 19 dont le rôle est de tester que la ligne 14 s'est faite (2e séquence) ou pas faite (1re séquence). A cet effet, la ligne 14 porte une KB qui garnit une position de mémoire d'un code quelconque, ici 5. La présence du code 5 introduit en M_3 lors de la 2e séquence est décelée par l'opération VC. Le comparateur indique ≠ et la ligne 21 succède à la ligne 20 (*fig.* 140).

NL		TO	AD	OD	OF
13	VS			3	13
14	KB	4	3	11	5
15					
19	CN	9	3	11	
20	VCc		2	3	8
21					

Fig. 140

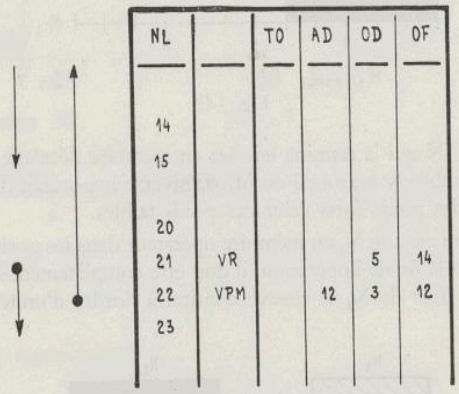
b) *Avec des instructions différentes*. Les instructions sont modifiées au moyen d'alternatifs excités après la première utilisation de la séquence S. Il faut introduire un temps de sécurité entre chaque utilisation de S.

D'une part, l'appel des commandes S contrôlant les alternatifs ne devra se faire que lorsque la première utilisation de la séquence sera terminée.

D'autre part, la deuxième utilisation ne devra être autorisée que lorsque les sélections seront positionnées. Pour cela, une VPM définit l'instant de départ qui doit être postérieur aux impulsions d'appel.

Dans l'exemple, la VPM affichée à la ligne 22 doit observer l'instant 11,14 si les commandes S sont appelées au point 11.

Enfin, il faut éliminer cette VPM lors de la seconde utilisation de la séquence, en affichant une variante à la ligne 21. Cette variante peut consulter, par exemple, une commande à relais appelée au point 11 (*fig.* 141).



NL		TO	AD	OD	OF
14					
15					
20					
21	VR			5	14
22	VPM		12	3	12
23					

Fig. 141

II. LES TRANSFERTS

1. Mémoire banale vers mémoire opérateur.

A. *Cas général.*

Transfert d'un nombre en M_1.

Le type d'opération BO résout ce problème (voir description de cette opération).

B. *Regroupement de plusieurs nombres en M_1.*

Ce problème se pose toutes les fois qu'il y a lieu de regrouper en M_1 plusieurs nombres ou plusieurs membres d'un même nombre répartis dans différentes mémoires. La difficulté du problème tient au fait que l'opération BO n'est valable que pour le transfert du premier nombre puisque T_1 efface le contenu de la mémoire opérateur. Il est donc nécessaire d'effectuer le second transfert par manière d'addition. Or, l'addition se fait avec cadrage préalable, et pour éviter d'additionner purement et simplement les deux nombres, il faut prévoir une AMD entre la BO et l'AN.

Soient : N_1 les poids faibles et N_2 les poids forts d'un nombre N, introduits respecti-

vement dans les mémoires M_2 (OD = 8, OF = 0) et M_3 (OD = 7, OF = 0) (*fig.* 142).

Au cours d'une première ligne, on effectue une BO qui amène N_2 (poids forts) en mémoire opérateur. Il est important, en effet, de réserver le transfert des poids faibles à la

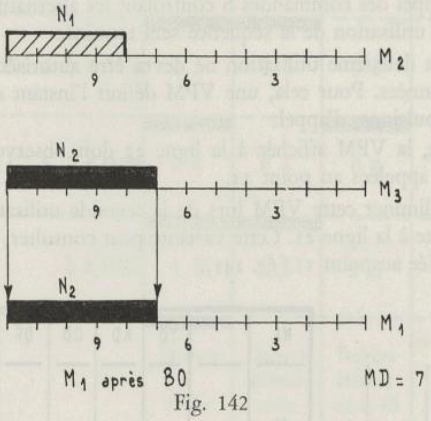


Fig. 142

fin de la séquence : l'AN qui la termine laissera en mémoire décalage un OD qui doit être celui des unités du nombre N regroupé en M_1. L'inverse imposerait de faire une AMD qui substituerait à l'OD des poids forts celui des poids faibles.

L'addition amènera ensuite N_1 en mémoire opérateur dans les positions de 7 à 0 de cette mémoire. Quant à N_2 à la fin de l'opération, il doit être complètement à droite, de façon que l'ordre d'unité le plus fort de N_1 se trouve contigu à l'ordre d'unité le plus faible de N_2 (*fig.* 143).

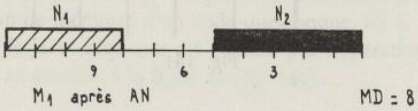


Fig. 143

Le nombre, ainsi constitué en M_1, se trouve fractionné, mais la MD indiquant le début du nombre, celui-ci se reformera convenablement en M_1 au cours du cadrage préalable de l'opération suivante qui doit être OB, CN, AN ou SN.

TO	AD	OD	OF	Opération	MD
6	3	7		BO	7
2		3		AMD	3
10	2	8		AN	8

Fig. 144

Il sera donc nécessaire que le cadrage de l'AN réalise 7 décalages du nombre N et que le train T_1 démarre lorsque le décomptage de MD l'aura amené à 8, qui est l'OD de N_1. La position initiale de MD avant AN sera donc :

x = MD finale + N b de décalages (à 12 près),

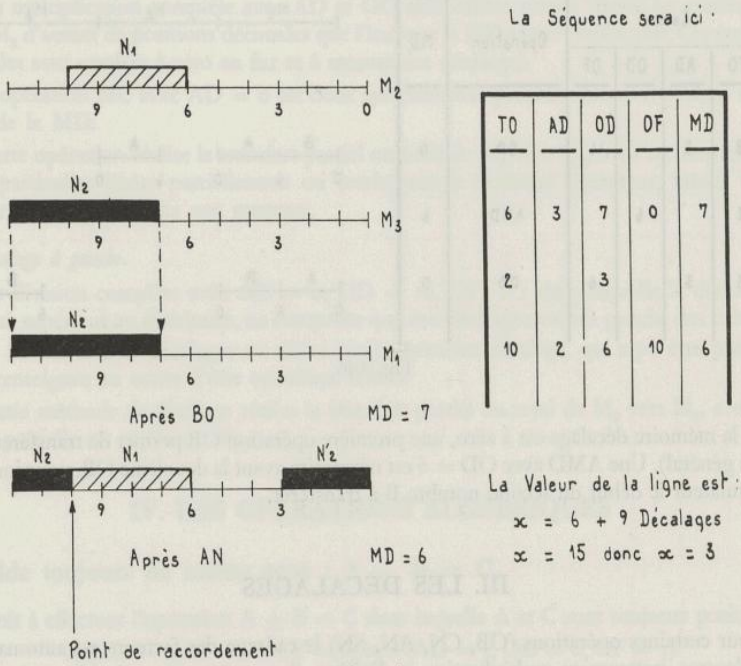
soit :

$x = 8 + 7 = 15$ (à 12 près),

$x = 15 - 12 = 3$.

D'où la séquence de la figure 144.

Il est essentiel de bien voir que l'impératif le plus important de cette séquence est qu'on doit amener en continuité les deux parties du nombre qui se trouvait en quelque sorte écartelé. Une variante de l'exemple donné plus haut illustre ce point (*fig.* 145).



TO	AD	OD	OF	MD
6	3	7	0	7
2		3		
10	2	6	10	6

Fig. 145

La valeur de la ligne AMD est :

$$x = 6 + 9 \text{ décalages}$$
$$x = 15 \text{ donc } x = 3.$$

On peut remarquer que l'OD est alors identique à celui de l'exemple précédent pour les mêmes capacités.

C'est très normal puisque les décalages doivent permettre à N_2 de rattraper N_1 qui, s'il a ses poids forts plus à droite de *n* positions, a inévitablement, du fait de sa capacité, ses poids faibles de *n* positions plus à droite : le nombre des décalages est augmenté de *n*, mais la MD à obtenir après décomptage est diminuée de *n*.

2. Mémoire opérateur vers mémoire banale.

A. *Cas général*

Transfert d'un nombre en MB.

Le type d'opération (voir description OB) résoud ce problème.

B. *Transfert partiel.*

L'opération OB doit être précédée d'une AMD convenable, lorsque le contenu de la mémoire décalage ne correspond pas au début du nombre à transférer.

Soient : 2 nombres A (OD = 0, OF = 6) et B (OD = 6 ,OF = 0) introduits en mémoire opérateur, à transférer respectivement en M_2 et M_3 (OD = 0, OF = 6) (*fig.* 146).

Codes				Opération	MD
TO	AD	OD	OF		
8	2		6	OB	0
2		6		AMD	6
8	3		6	OB	0

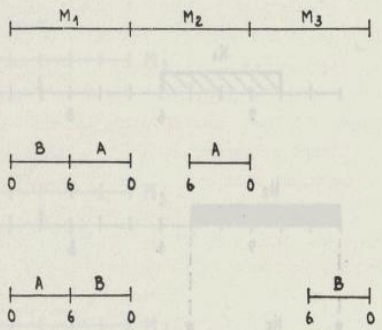


Fig. 146

Si la mémoire décalage est à zéro, une première opération OB permet de transférer A en M_2 (cas général). Une AMD avec OD = 6 est nécessaire avant la deuxième OB, pour indiquer au Calculateur le début du second nombre B à transférer.

III. LES DÉCALAGES

Pour certaines opérations (OB, CN, AN, SN) le cadrage des facteurs est automatique. Pour d'autres, notamment multiplication et division, il peut être nécessaire de programmer la mise en place convenable des facteurs.

1. Décalages en mémoire opérateur.

A. *Décalage de M_1 sans altération de son contenu.*

Toute opération avec cadrage préalable peut être utilisée pour décaler le contenu de la M_1 et l'amener finalement à droite ou à gauche de sa position initiale.

Soit à décaler de 4 positions vers la droite un nombre N contenu en M_1 (MD = 4).

L'opération OB avec l'adresse d'une mémoire non utilisée au cours du problème (AD = 15) par exemple, et un ordre début 0, réalise ce décalage en amenant N dans la position : OD = 0, OF = 4 (MD = 0).

B. *Décalage à droite avec abandon des poids faibles.*

La multiplication réduite avec AD et OF nuls décale vers la droite le contenu de M_1 d'autant de positions décimales que l'indique l'OD affiché. Ces positions décimales sont remises à zéro au fur et à mesure des décalages.

Au cours d'une seule ligne de programme, cette opération peut permettre, par exemple, l'abandon des décimales d'un résultat partiel (ainsi que BO avec AD = 1) et sa mise en place à droite de M_1 en vue de la MR suivante.

2. Décalages simultanés des mémoires 1 et 2.

A. *Décalages à droite avec abandon des poids faibles de* M_2.

La multiplication complète avec AD et OD nuls décale vers la droite les contenus de M_1 et M_2 d'autant de positions décimales que l'indique la MD avant l'opération. Ces positions décimales sont remises à zéro au fur et à mesure des décalages.

L'opération MC avec AD = 0 est donc généralement précédée d'une altération convenable de la MD.

Cette opération réalise le transfert partiel ou total de M_1 vers M_2, mais ce transfert d'un genre particulier libère partiellement ou totalement la mémoire opérateur, tandis que le transfert OB n'altère pas son contenu.

B. *Décalage à gauche.*

La division complète avec AD = 0, OD = 10, OF = 1 dans laquelle le diviseur est toujours supérieur au dividende, ne comporte que des décalages vers la gauche des mémoires 1 et 2. Le nombre de décalages est défini par la mémoire décalage, qui a pu être préalablement renseignée au cours d'une opération AMD.

Cette méthode de décalage réalise le transfert partiel ou total de M_2 vers M_1, avec libération partielle ou totale de M_2.

IV. LES OPÉRATIONS ALGÉBRIQUES

1. Solde toujours de même sens : A ± B = C.

Soit à effectuer l'opération A ± B = C dans laquelle A et C sont toujours positifs.

A (AD = 2, OD = 0, OF = 8) et B (AD = 3, OD = 0, OF = 7).

A. *Le signe de B est donné par la machine connectée.*

L'impulsion électrique caractérisant le sens négatif de B appelle une commande de sélection S_0. Le type d'opération est SN ou AN, selon que S_0, est excité ou non (*fig.* 147).

Codes				Opération	MD
TO	AD	OD	OF		
6	2		8	BO	0
11/10	3		7	SN ou AN	0

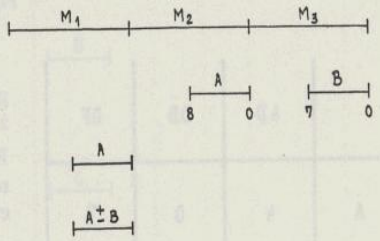


Fig. 147

Ainsi le TO de la ligne unique affecté à l'opération A ± B reçoit du commun de l'alternatif S_0 :

— le code 11 (SN) si S_0 est excité,

— le code 10 (AN) si S_0 n'est pas excité.

B. *Le signe de B est donné par le Gamma.*

Ce signe est déterminé par exemple par le sens d'une comparaison :

positif lorsque le comparateur indique $>$,
négatif lorsque le comparateur indique $\leqslant$.

Le type d'opération est AN ou SN, selon que B est positif ou négatif. Le signe de B est analysé par une variante comparaison (*fig.* 148).

Codes				Opération	MD	Sélection
TO	AD	OD	OF			
9		–	–	CN		
6	2		8	BO	0	
	–	–	–	VC $\leqslant$	0	
10	3		7	AN	0	
		–	–	V systémat.	0	
11	3		7	SN Transfert du Résultat	0	

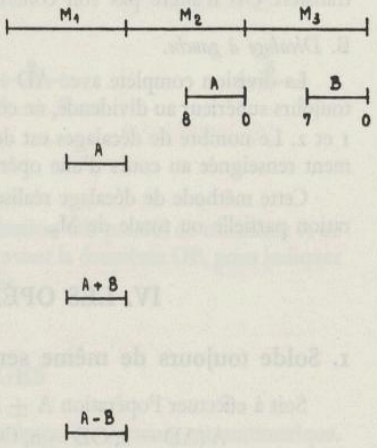


Fig. 148

	AD	OD	OF
A	4	0	7
B	2	0	7
C	3	0	8

Fig. 149

Cette VC sélectionne les deux lignes de programme affectées respectivement à l'addition et à la soustraction.

Remarque : le résultat d'une soustraction étant toujours exprimé en valeur absolue, on peut effectuer indifféremment A $\pm$ B ou B $\pm$ A.

2. Solde positif ou négatif : A $\pm$ B $=$ $\pm$ C.

Soit à effectuer l'opération A $\pm$ B $=$ $\pm$ C, dans laquelle A est toujours positif (*fig.* 149).

A. *Le signe de B est donné par la machine connectée.*

Le type d'opération est sélectionné par une commande S_0 excitée pour B négatif.

Si S_0 est au repos, le type d'opération est AN et le signe est toujours positif. Si S_0 est excitée, le type d'opération est SN et le signe est fonction des valeurs absolues de A et B :

$$\text{Si } |B| < |A|,\ C > 0$$
$$|B| > |A|,\ C < 0$$
$$|B| = |A|,\ C \text{ est nul.}$$

Ce dernier cas sera étudié par la suite. Ici, le cas $C = 0$ est lié au cas $C > 0$.

Le signe négatif de C peut être représenté par un code quelconque introduit dans une position déterminée d'une mémoire banale. Soit par exemple, un code 11 introduit en M_3 (OD = 8) à l'aide d'une opération KB.

La comparaison préalable des termes A et B permet de sélectionner cette KB à l'aide d'une variante comparaison.

Lorsque S_0 n'est pas excité, il n'y a pas lieu de tenir compte du résultat de la comparaison et la variante doit « toujours » éliminer la KB.

Lorsque S_0 est excité, il faut consulter le comparateur et éliminer KB quand $B \leqslant A$.

Le fait observé est donc soit $\leqslant$ (AD = 1), soit « toujours » (AD = 0), selon que S_0 est excité ou non. Les codes AD 1 et 0 sont regroupés à l'aide d'un alternatif commandé par S_0.

BO amène B en mémoire opérateur;
CN compare B à A.

B étant alors en M_1, il est possible d'effectuer l'opération $B \pm A$ dont le résultat est identique à l'opération $A \pm B$.

OB transfère C en M_3. L'OF affiché à cette ligne provoque la remise à zéro de la position de M_3 affectée au signe (*fig.* 150).

Sélection Lignes	Codes TO	AD	OD	OF	Opération
	6	2		7	BO
	9	4		7	CN
	11/10	4		7	SN ou AN
	8	3		9	OB
		1/0			V ≤ ou Tjs
	4	3	8	11	KB

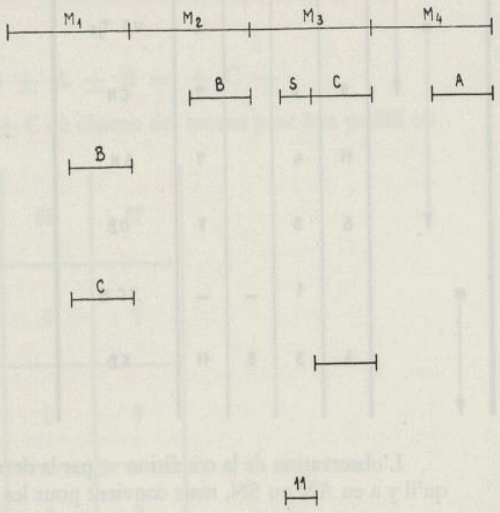


Fig. 150

VC élimine KB quand C est positif ou nul;

KB introduit un code 11 en M_3 lorsque C est négatif.

B. *Le signe de B est donné par le Gamma.*

Le signe négatif de B est symbolisé par exemple par un code 5 introduit en mémoire 2 (OD = 7).

Le type d'opération dépend de la nature de ce signe. Une première comparaison entre la mémoire opérateur remise à zéro et la position de M_2 occupée par le signe de B est exploitée ensuite par une variante pour sélectionner les lignes d'addition et de soustraction (cf ci-dessus D, II, 2°).

Une seconde comparaison entre B et A est nécessaire avant la soustraction pour préparer la détermination du signe de C effectuée plus loin (*fig.* 151).

Sélection Lignes	Codes TO	AD	OD	OF	Opération
	9	2	7	8	CN
	6	2		7	BO
		2	–	–	VC ≠
	10	4		7	AN
			–	–	VS Tjs
	9	4		7	CN
	11	4		7	SN
	8	3		9	OB
		1	–	–	VC ≤
	4	3	8	11	KB

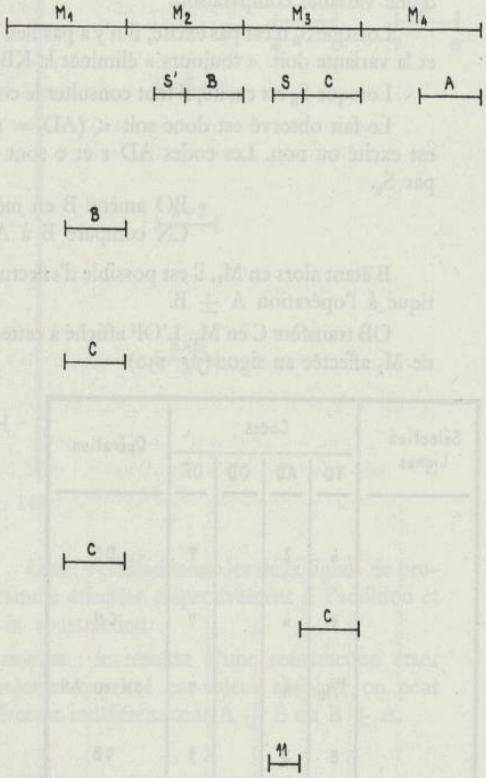


Fig. 151

L'observation de la condition ⩽ par la dernière VC a une signification différente suivant qu'il y a eu AN ou SN, mais convient pour les deux cas.

S'il y a eu SN, la seconde comparaison a eu lieu :

⩽ signifie B ⩽ A.

S'il y a eu AN, la seconde comparaison n'a pas lieu :

$\leqslant$ signifie signe de B positif ou nul.

Dans les deux cas, il faut éliminer KB par VC $\leqslant$.

C. *Détection du solde nul.* Faux

La comparaison des deux termes de la soustraction sert, dans les exemples précédents, à déterminer le sens du solde. Elle peut aussi servir à détecter les soldes nuls. Il suffit de donner l'AD = 2 à une VC pour lire le comparateur en = ou $\neq$.

Le problème A $\pm$ B = $\pm$ C (cf. D, III, 1°) étant complété de la détection du solde nul, la séquence devient par exemple :

CN
SN
OB
VC =
KB code 10 pour solde nul
VC $\leqslant$
KB code 11 pour solde négatif.

D. *Extraction du signe du solde.*

Le signe d'un solde peut être symbolisé par un code quelconque, extrait vers la machine connectée au point correspondant.

S'il doit être perforé au-dessus de la zone de la carte qui reçoit le solde, ce code peut être 10, 11 ou 12. Il est extrait au point 0, 11 ou 12 après ce solde.

S'il doit appeler une commande de sélection, destinée elle-même à ventiler le solde, il faut l'extraire avant le point 9, et lui donner les valeurs 11, 12, 13, 14 ou 15 (cf. opération KB).

S'il doit être repris au cours des séquences suivantes, il faut pouvoir l'utiliser en opération CN, donc lui donner une valeur arithmétique (inférieure à 10). Si besoin est, ce code peut être traduit en une perforation 11, à l'aide d'une connexion simple au tableau de la machine connectée.

3. Somme algébrique de deux termes $\pm$ A $\pm$ B = $\pm$ C —

Soit à effectuer l'opération $\pm$ A $\pm$ B = $\pm$ C où chacun des termes peut être positif ou négatif (*fig.* 152).

	AD	OD	OF
A	4	0	7
B	2	0	7
C	3	0	8

Fig. 152

A. *Les signes de A et de B sont donnés par la machine connectée.*

Le type d'opération dépend des signes des termes. Le signe du résultat est fonction, d'une part du type d'opération, donc des termes, et, d'autre part, de la valeur relative de A et de B, lorsqu'il y a soustraction.

Les huit combinaisons possibles sont réunies dans le tableau de la figure 153.

Signe B	B ≷ A	Signe A	Signe C	TO	Conditions		Type Variante
+	≤	+	+	AN	$\overline{S_0}$	$\overline{S_1}$	VS Tjs
+	>	+	+	AN	$\overline{S_0}$	$\overline{S_1}$	VS Tjs
+	≤	−	−	SN	S_0	S_1	VC >
+	>	−	+	SN	S_0	S_1	VC >
−	≤	+	+	SN	S_0	$\overline{S_1}$	VC ≤
−	>	+	−	SN	S_0	$\overline{S_1}$	VC ≤
−	≤	−	−	AN	$\overline{S_0}$	S_1	VS Jamais
−	>	−	−	AN	$\overline{S_0}$	S_1	VS Jamais

Fig. 153

Ce tableau fait apparaître que :

— le type d'opération est AN lorsque les signes sont identiques, et SN lorsque les signes sont différents,

— le signe de C est le même que celui de A, sauf dans deux cas déterminés, caractérisés par « signes différents » et « B > A ».

Il faut donc appeler deux commandes S :

— S_0 lorsque les signes des termes sont différents, pour déterminer le TO,

— S_1 lorsque le signe de A est négatif, pour déterminer le signe de C.

On voit que l'état de S_0 définit deux régions : celle des AN et celle des SN et que l'état de S_1 découpe en deux chacune de ces régions en fonction des signes de C et A. Les régions. $\overline{S}_0\ S_1$ et $\overline{S}_0\ \overline{S}_1$, ne font pas difficulté : les signes de A et C sont toujours identiques. En $\overline{S}_0\ \overline{S}_1$ il y a lieu de toujours sauter la KB de signe de C : il y a donc lieu à VS « toujours ». En $\overline{S}_0\ S_1$ il y a lieu de ne jamais sauter cette KB : il y a donc lieu à VS « jamais ».

Dans la région des 4 cas SN (S_0), il y aura lieu de faire les variantes comparaisons affichées sur le tableau : le travail de S_1 validera un VC > et son repos une VC ⩽ qui feront sauter la KB dans les deux cas positifs.

On voit donc que S_0 contrôlera le type de variante : VS en $\overline{S}_0$ et VC en S_0, et S_1, le mode d'exploitation de ces variantes. D'où la séquence de la figure 154.

Sélection Instant.				Sélection Lignes	NL	Codes				Opération
Nº	S	C	NS			TO	AD	OD	OF	
					1	6	2		7	BO
					2	9	4		7	CN
0	11	A	10		3	A	4		7	SN ou AN
					4	B	3		9	OB
0	1	B	0		5		B	1	C	Variante
1	12	C	13		6	4	3	8	11	KB
					7					

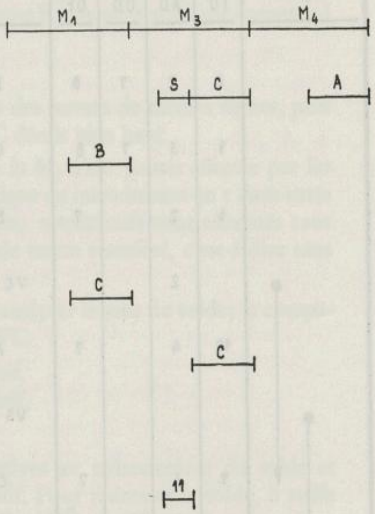


Fig. 154

B. *Les signes de A et B sont donnés par le Gamma.*

Les signes négatifs des termes peuvent être représentés par un code 5 introduit immédiatement à gauche de chacun de ces termes, c'est-à-dire en M_4, OD_7 pour A et en M_2, OD_7 pour B.

L'étude des 8 cas possibles met en évidence que :

— le TO est AN ou SN selon que les signes sont identiques ou différents,

— le signe du résultat C est le même que celui de A sauf dans deux cas déterminés, caractérisés par « signes différents » et « $B > A$ ».

Deux comparaisons sont donc nécessaires :

— la comparaison des signes pour déterminer le type d'opération,

— la comparaison des termes pour déterminer le signe du résultat lorsqu'il y a soustraction.

La séquence de programme ressemble à celle déjà décrite pour le problème $A \pm B = \pm C$. Elle ne diffère de celle-ci que par la détermination du signe (*fig.* 155, page 142).

4. Solde progressif.

Un solde progressif est une somme algébrique verticale de deux termes. L'un des termes est donné par une carte, l'autre terme est le solde précédent que l'on reprend :

$$\pm A_1 \pm B = \pm A_2$$
$$\pm A_2 \pm B = \pm A_3$$
$$\pm A_3 \pm B = \pm A_4, \text{ etc.}$$

Sélection Lignes	Codes TO	AD	OD	OF	Opération
	6	2	7	8	BO
	9	3	7	8	CN
	6	2		7	BO
		2			VC
	10	4		7	AN
					VS Tjs
	9	4		7	CN
	11	4		7	SN
	8	3		9	OB
	6	4	7	8	BO
		1			VC
	10		7	5	AN
	8	3	8	9	OB

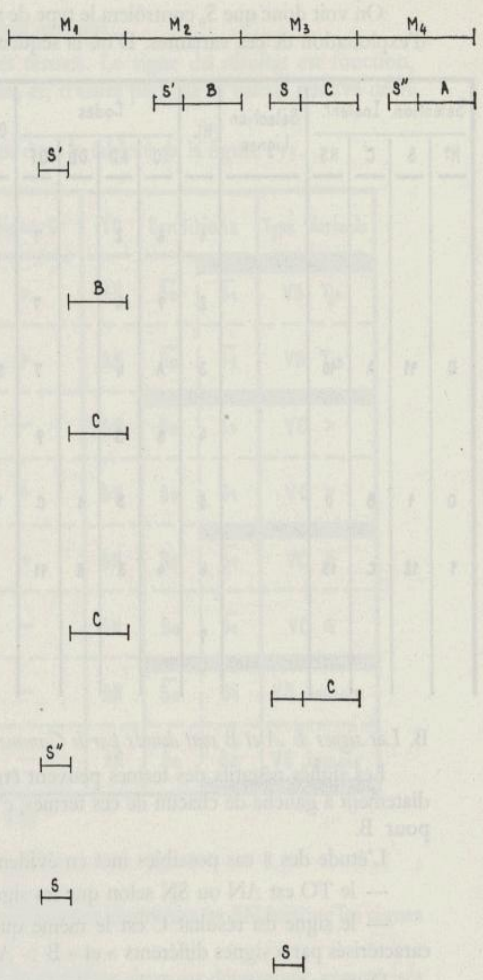


Fig. 155

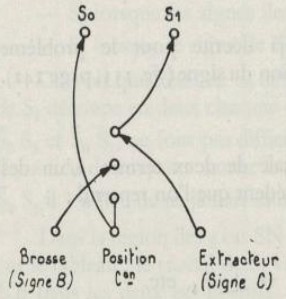


Fig. 156

Le signe de C doit être extrait à trois fins :

— pour être comparé au signe de B de la carte suivante et appeler S_0 si les signes sont différents,

— pour appeler S_1 s'il est négatif,

— pour être perforé, imprimé, ou exciter une sélection de débit-crédit s'il y a lieu.

Le schéma de connexions donné ici indique le principe de la connexion. Bien entendu, les précautions habituelles d'appel des commandes S sont à respecter (*fig.* 156).

5. Sommes algébriques de plusieurs termes.

La méthode ci-dessus (3) pourrait être appliquée. Il est préférable d'utiliser d'autres méthodes plus économiques en lignes de programme, lorsque la somme algébrique comprend plus de 2 termes.

Soit par exemple :

— *Première méthode :* effectuer séparément les cumuls des termes de mêmes signes, puis agir avec les 2 totaux comme dans le cas $A - B = \pm C$ décrit plus haut.

— 2e *méthode :* lorsque la 11e position décimale de la M_1 n'est jamais affectée par les soldes, il est possible d'empêcher le redressement automatique en introduisant un 1 dans cette 11e position au début du calcul. Les transferts additifs ou soustractifs sont effectués sans tenir compte de la valeur relative du solde progressif et du terme transféré, c'est-à-dire sans analyser le sens du solde après chaque transfert.

Une seule opération CN suffit à la fin du calcul pour analyser le sens du solde; la comparaison avec AD 0, OD 10 et OF 1 est exploitée par une VC.

Si $M_1 > 1$ solde positif,
Si $M_1 < 1$ solde négatif,
Si $M_1 = 1$ solde nul.

La VC sélectionne deux lignes de programme relatives au redressement du solde et l'introduction d'un code-signe, lorsque le solde est négatif. Pour redresser le solde, il suffit de le soustraire à 10^{10} à l'aide d'une opération SN avec AD 0, OD 10, OF 1.

6. Nombre en complément.

Le résultat C d'une soustraction A—B est exprimé en valeur absolue. Pour le transférer en complément dans un totalisateur, il faut l'avoir préalablement soustrait de 10^{11} au cours d'une opération spéciale.

Les codes TO = 11, AD = 0, OD = 10, OF = 10, définissent cette opération; exceptionnellement un code non arithmétique (code 10) provenant de OF est soustrait du contenu de la mémoire opérateur.

Par suite de la comparaison du terme C et du code 10 introduit en 11e position, la soustraction $C - 10^{11} = C - 0 = C$ est suivie d'un redressement automatique commandé par le comparateur et consistant en l'opération $0 - C = 10^{11} - C$.

Remarque : le complément à 10^{11} en exposant 10^{11} n'est pas identique au complément à 9 obtenu en tabulatrice. Si l'on transfère ce complément à 10^{11} dans un totalisateur, il ne faut pas connecter le retour du report sur ce totalisateur.

V. LES OPÉRATIONS RECURRENTES

1. Choix du type d'opération.

Le choix entre les opérations complètes et réduites n'est possible que lorsque le produit ou le dividende sont au plus égaux à 11 chiffres. Dans le cas contraire, l'opération est obligatoirement complète.

Lorsqu'il est possible, le choix est soumis à différents impératifs qui tiennent à la forme du problème et à la souplesse de base plus grande des opérations complètes, tempérée par la nécessité de recourir à 3 mémoires au lieu de 2 et de positionner préalablement la mémoire

décalage. Tout ceci a été exposé lors de la description des opérations. Rappelons simplement que les opérations complètes comportent 3 degrés de liberté : la position des 2 facteurs et le positionnement de MD alors que les opérations réduites n'en comportent que 2, le positionnement de la MD étant lié à la position d'un des facteurs.

On exposera donc simplement quelques exemples type donnant la préférence tantôt à l'une, tantôt à l'autre forme d'opération.

2. Applications.

A. *Suite de produits.*

Cet exemple a été traité lors de la description de la MC.

B. *Addition d'un nombre à un produit.*

a) *Cas général* $AB + C = X$. La multiplication est réalisée par addition successive du multiplicande. Le produit est donc constitué dans la mémoire opérateur. Si cette mémoire contient un nombre avant l'opération, le produit peut s'ajouter à ce nombre.

Toutefois, les produits partiels sont décalés vers la droite au cours de l'opération.

Pour effectuer l'opération AB + C au cours d'une seule ligne de programme affectée à la multiplication de A par B, il faut préalablement introduire C en M_1 en tenant compte du décalage qu'il subira au cours de la multiplication; l'origine de C (avant l'opération) est égale à l'origine du produit (après l'opération) majorée du nombre de décalages.

b) *Cas particulier : arrondi.* L'arrondi peut être obtenu selon la méthode précédente : la correction désirée étant introduite dans la position convenable de M_1 avant la multiplication.

Soit à effectuer le produit de A (AD = 2, OF = 0, OD = 5) par B (AD = 3, OD = 1, OF = 10) arrondi à l'unité la plus proche (A et B ayant chacun 2 décimales).

Un 5 est introduit préalablement en M_1 (OD = 4) (*fig.* 157).

Codes				Opération	MD
TO	AD	OD	OF		
2		5		AMD	5
14	3	1	10	MC	0

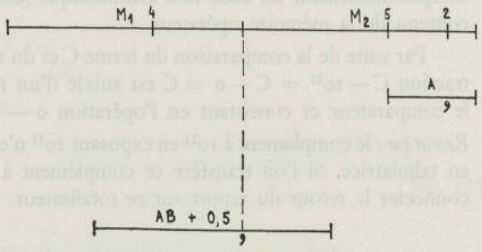


Fig. 157

Remarque : l'arrondi peut aussi être obtenu par addition au produit d'une constante convenable, c'est-à-dire AN avec AD 0. Cette méthode est utilisée notamment après une multiplication réduite.

C. *Somme de produits.*

Soit à effectuer le calcul suivant :

$$Aa + Bb + Cc = X$$

chacun des produits ayant 8 chiffres et la somme 9 chiffres. La capacité des facteurs permet de choisir MR. Les facteurs sont ainsi mis en place. (*fig.* 158).

	AD	OD	OF		AD	OD	OF
A	2	3	8	a	1	0	3
B	4	3	8	b	4	0	3
C	5	3	8	c	5	0	3

Fig. 158

On calcule le premier produit A*a* (*fig.* 159).

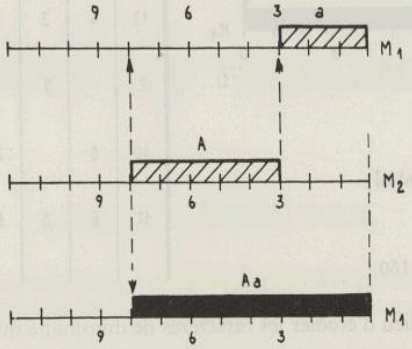


Fig. 159

Lorsque ce produit est calculé, il faut le conserver en M_1 tout en introduisant le nouveau multiplicateur *b*. Une AN permet de faire cette introduction sans effacer A*a*. D'autre part, il est nécessaire de faire venir A*a* avec une position début égale à l'ordre fin de *b* : c'est le problème exposé plus haut du regroupement de deux nombres en M_1. On affichera donc une AMD avec un OD égal à : $x = 9 + 0$ (9 = nombre de décalages de A*a*, 0 valeur finale de MD en fin d'AN) (*fig.* 160, page 146).

On continuera de la même façon pour amener C et multiplier C par *c*.

D. *Multiplication ou division par une constante.*

Un multiplicande ou un diviseur constant ayant un seul chiffre significatif peut être distribué par la mémoire OF au cours d'une opération récurrente avec AD = 0.

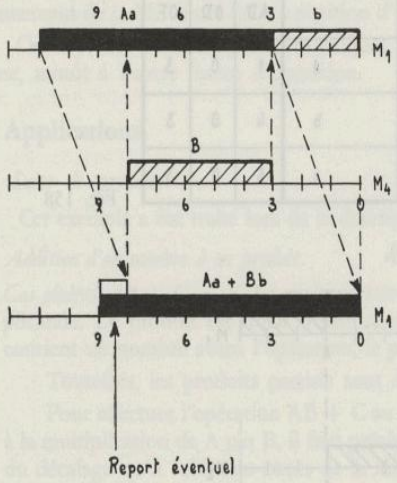
Si la constante a plusieurs chiffres significatifs, il est souvent préférable de l'introduire en mémoire banale. Cependant, on pourra se ramener au cas précédent lorsque la constante est telle que :

$$C = C_1 \times C_2, \; C_1 \text{ et } C_2 \text{ n'ayant qu'un seul chiffre } (35 = 7 \times 5),$$

ou, ce qui est identique :

$$c = \frac{C_1}{C_2} \qquad C_1 \text{ et } C_2 \text{ n'ayant qu'un seul chiffre } \left(0{,}75 = \frac{3}{4}\right)$$

Il est nécessaire de remarquer que cette méthode n'est applicable dans la division par $C = C_1 \times C_2$ que si la première division par C_1 ou C_2 se fait avec un reste nul. Dans l'exemple ou C = 35, la première division donnera un reste nul si le dividende se termine par un 5



TO	AD	OD	OF	Opération	MD
12	2	3	8	MR	0
2		9		AMD	9
10	4		3	AN	0
12	4	3	8	MR	0
2		9		AMD	9
10	5		3	AN	0
12	5	3	8	MR	0

Fig. 160

ou un o. Il y aura donc lieu d'étudier les caractères de divisibilité du dividende par les constantes C_1 C_2 possibles.

E. *Diviseur nul.*

Dans tous les cas où le diviseur peut être nul, il y a lieu d'éliminer la division, sinon les transferts soustractifs s'effectueraient indéfiniment, ce qui provoquerait le blocage de la machine sur la ligne de programme correspondante.

Lorsque le diviseur est lu dans la carte, il doit être considéré comme pouvant être nul (début et fin de travail). La division est validée à l'aide d'une commande S appelée par les unités du diviseur.

Lorsque le diviseur provient de calculs précédemment effectués dans le Calculateur, il doit être comparé à zéro. La division est éliminée par une variante comparaison observant l'égalité.

Il est quelquefois plus simple de comparer entre eux le dividende et le diviseur lorsque ces facteurs sont en place pour la division. Il faut en effet que la partie du dividende contenue en M_1 soit $<$ au diviseur pour que la division puisse se faire convenablement. Et la variante comparaison qui succède à cette comparaison doit observer $\geqslant$ pour éliminer la division lorsque le diviseur est nul ou mal cadré.

Toutefois, la comparaison ne doit pas entraîner un déplacement du contenu de la M_1 par suite du cadrage préalable. Ce cadrage peut être évité :

— soit en altérant convenablement la MD avant la comparaison (et éventuellement après s'il s'agit d'une DC).

— soit en donnant au diviseur un OD égal à la valeur de la MD avant la comparaison (si les positions à droite du diviseur sont libres afin de ne pas fausser la comparaison).

Chapitre dixième

ÉTUDE DES PROBLÈMES

I. EN CONNEXION P.R.D.

En dehors des liaisons de base qui ont été décrites dans la première partie de cette notice, les problèmes qui se posent lors de l'étude de cette connexion ont trait à la vitesse et à la capacité.

Le tableau qui suit donne toutes les combinaisons possibles entre ces deux facteurs. On analysera ensuite l'opportunité de choix pratique à faire parmi elles :

Soient :

D = nombre de chiffres de données,
R = — — de résultats,
M = — de positions de mémoires,
I = — — d'introduction,
E = — — d'extraction.

1. $D > I$ ou $R > E$. Il est évident qu'il y a impossibilité à traiter le problème, soit de calcul, soit de contrôle, en un seul passage.

2. $D + R > M$. Calcul possible en 1 passage à 60 cartes par minute.
Contrôle possible en 2e passage à 60 cartes minute sur la piste de lecture.

3. $D + R < M$. Le calcul est possible en un seul passage à 120 cartes minute.
Le contrôle est possible en 2e passage à 120 cartes minute sur la piste de lecture.
Le calcul et le contrôle sont possibles simultanément (sur les 2 pistes en un passage à 60 cartes minute).

4. $D + 2R < M$ et $D + R < 1$. Le calcul et le contrôle sont possibles en un passage à 60 cartes minute sur la piste de perforation seule.

5. $2D + 2R < M$ et $2D < I$ et $2R < E$. Le calcul et le contrôle sont possibles en un seul passage à 120 cartes minute sur les deux pistes.

6. $2D + 2R < M$ et $2D + R < I$ et $R < E$. Le calcul et le contrôle sont possibles en un seul passage à 120 cartes minute sur piste de perforation seulement.

L'écriture de ces formules est faite pour l'hypothèse où le programme de calcul et de contrôle sont identiques, c'est-à-dire où l'on vérifie par exemple la multiplication $A \times B = C$ en la faisant une seconde fois. Dans le cas où l'on adopte des programmes différents pour le calcul et pour le contrôle, on devra les écrire sous une forme où apparaîtront les capacités dans les deux cas :

$$2\ D \text{ s'écrira } D + D',\ 2R \text{ s'écrira } R + R'.$$

Concernant les capacités requises à 120 cartes minute lorsque l'on désire contrôler, il y a lieu de remarquer que l'immobilisation de mémoires et de positions d'introduction et d'extraction est assez considérable. Il faut donc retenir comme capacité directrice, lors du choix d'un Gamma, celle du § 3 où $D + R < M$. Cette capacité assure une vitesse comparable aux § 2 et 4 et elle n'immobilise pas les capacités importantes des § 5 et 6. Ces deux dernières possibilités seront utilisées pour traiter, de préférence, les problèmes de peu de capacité sur un Gamma déjà existant choisi pour des problèmes de plus grande exigence en capacité selon la méthode du § 3.

A. Le principe de la connexion pour le cas 5 est le suivant : soit le calcul et le contrôle de $A \times B = C$, par division, sur les 2 pistes à 120 c/m (*fig.* 161).

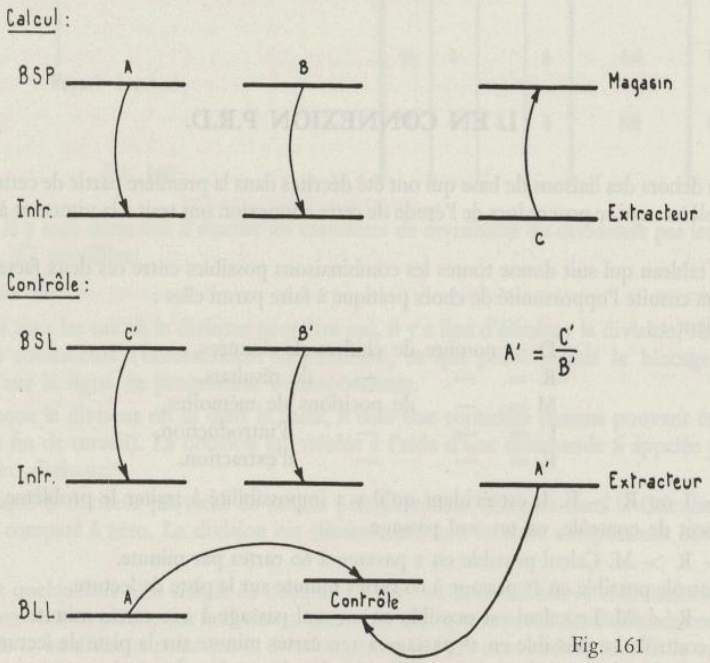


Fig. 161

Les cartes une fois sorties de la piste de perforation sont reprises en piste de lecture pour le contrôle.

B. Pour le cas 6, le principe est le suivant : soit le calcul de $A \times B = C$ et le contrôle par division sur la piste de perfo seule, à 120 c/m (*fig.* 162).

La comparaison de contrôle aura lieu sur le Gamma : on calculera A'' à partir de C' et B' : $A'' = \frac{C'}{B'}$.

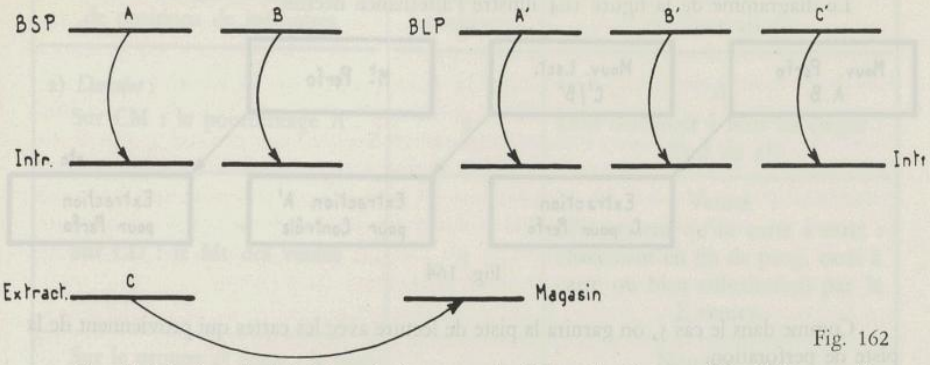


Fig. 162

Or, A″ doit être égal à A′ perforé et enregistré dans le Gamma. Une VC contrôlera une KB validée en cas d'erreur : l'extraction du code arrêtera la PRD. La carte erronée se trouvera au-dessus du paquet.

C. Pour le cas 3, au lieu d'utiliser un Gamma capable de faire côte à côte, simultanément, les deux calculs $A \times B$ et $\frac{C}{B}$, on les fera alternativement.

Un cycle sur deux, la piste Perfo alimentera une carte et le Calculateur calculera $A \times B$. Au cycle suivant, la piste de lecture alimentera une carte et on extraira C du Gamma sur le magasin, calculé au cycle précédent, pendant qu'on calculera $\frac{C}{B}$.

La lecture de C et B aura lieu en BSL : au cycle suivant où la piste de Perfo alimentera une autre carte, la carte lue en BSL passera en BLL du fait du mouvement général et enverra A aux comparateurs de contrôle qui recevront l'extraction du calcul $\frac{C}{B}$ (*fig.* 163).

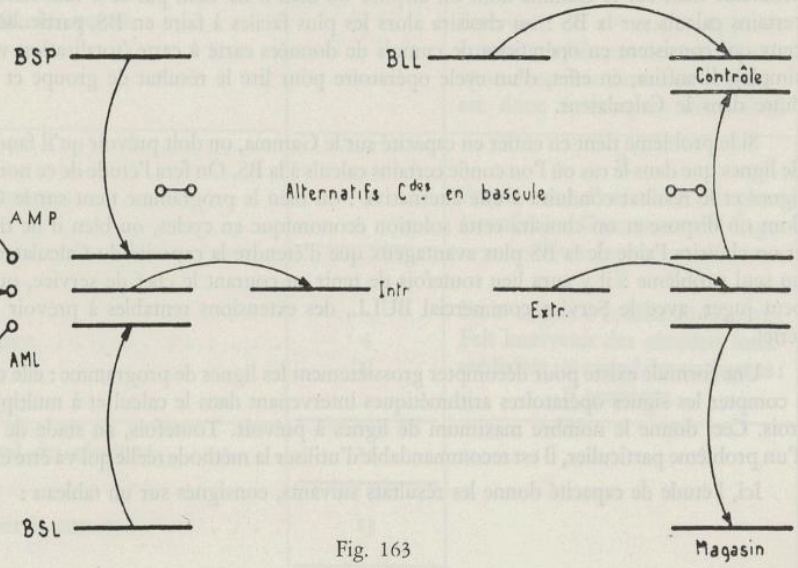


Fig. 163

Le diagramme de la figure 164 illustre l'alternance décrite.

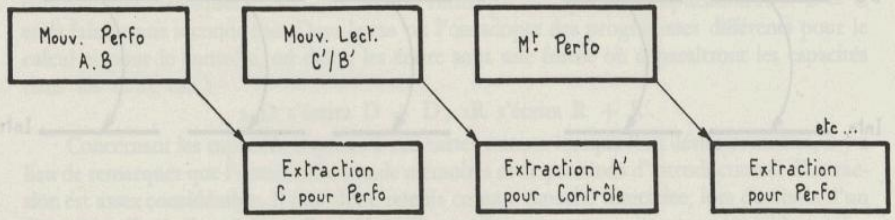


Fig. 164

Comme dans le cas 5, on garnira la piste de lecture avec les cartes qui proviennent de la piste de perforation.

II. EN CONNEXION BS

Soit le problème « Calcul de bonification » (Pl. 24-25-26). On le prendra comme exemple pour dégager un processus d'étude applicable aux divers problèmes qui peuvent se poser.

1. Rôle des deux machines.

Le premier point à étudier est le rôle à donner à chacune des machines. Pour cela, on examine la valeur en capacité des données et des résultats par rapport au Gamma : il est bon, afin d'éviter au maximum les cycles opératoires, de confier au Gamma le plus possible des calculs à faire. On détermine la capacité du Gamma. De deux choses l'une alors : ou bien le problème tient sur le Gamma dont on dispose ou bien il ne tient pas et il faut faire passer certains calculs sur la BS : on choisira alors les plus faciles à faire en BS, particulièrement ceux qui consistent en opérations de cumuls de données carte à carte (totalisation verticale simple). Il suffira, en effet, d'un cycle opératoire pour lire le résultat de groupe et l'introduire dans le Calculateur.

Si le problème tient en entier en capacité sur le Gamma, on doit prévoir qu'il faudra plus de lignes que dans le cas où l'on confie certains calculs à la BS. On fera l'étude de ce nombre de lignes et le résultat conduira à une alternative : ou bien le programme tient sur le Gamma dont on dispose et on choisira cette solution économique en cycles, ou bien il ne tient pas et on choisira l'aide de la BS plus avantageux que d'étendre la capacité du Calculateur pour un seul problème : il y aura lieu toutefois de tenir au courant le chef de service, qui, seul, peut juger, avec le Service commercial BULL, des extensions rentables à prévoir au matériel.

Une formule existe pour décompter grossièrement les lignes de programme : elle consiste à compter les signes opératoires arithmétiques intervenant dans le calcul et à multiplier par trois. Ceci donne le nombre maximum de lignes à prévoir. Toutefois, au stade de l'étude d'un problème particulier, il est recommandable d'utiliser la méthode réelle qui va être exposée.

Ici, l'étude de capacité donne les résultats suivants, consignés sur un tableau :

Nature des immobilisations de positions de mémoires	Nombre de positions	Libellé et observations
a) *Données* : Sur CM : le pourcentage A . .	2	A Doit demeurer à tous les étages : effacé au 2e.
Sur CD : le Mt des ventes . .	5	Ventes N'intervient qu'au carte à carte : effacement en fin de prog. carte à carte ou bien substitution par le Σ ventes.
Sur le groupe 2e étage : le nombre d'agents	2	Nombre Ag. Intervient en 2e étage. Effacé à ce stade
Constantes	0	Toutes les constantes ont un seul chiffre significatif. Elles peuvent être distribuées en AD = 0 et KB.
Total des données.	9	

Nature des immobilisations de positions de mémoires	Nombre de positions	Libellé et observations
b) *Résultats* : Total des ventes	5	ΣV Intervient en 1er étage. Apparaît comme un élément des calculs suivants et comme devant être extrait : est donc à compter deux fois.
Ristournes	4	R Proviennent d'un calcul complexe à confier au Gamma en 1er étage.
Primes.	4 6	P En 2e étage : à confier au Gamma, Fait intervenir des résultats intermédiaires : le cumul des ristournes : 6 positions.
Total des résultats	24	
Dont à extraire.	13	
Total positions mémoire. . . .	33	

Le Gamma 432 apparaît donc comme suffisant : extracteurs et introducteurs sont très larges. Quant aux positions de mémoire à immobiliser, il faut les majorer de la M_1 que l'on doit toujours réserver du fait de son caractère spécial. Cette majoration tient lieu de sécurité de capacité en mémoires : ici, on aura donc $33 + 12 = 45$, ce qui est admissible.

On peut d'ailleurs faire quelques observations qui permettront d'alléger les immobilisations de mémoires. En effet, à chaque rupture 1er étage a lieu, sur une ligne spéciale, l'impression des résultats de 1er étage, consistant dans le total des ventes : il faudra donc un cycle spécial opératoire sur la BS (commun aux 1er et 2e étages) au cours duquel il est facile de lire un totalisateur ayant enregistré le montant des ventes. Ce totalisateur sera remis à zéro à l'indication, puisque la CM agent est une carte de 1er étage. Ainsi, on économisera 5 positions de mémoire et les lignes de calcul sur Gamma du Σ ventes.

Seul ce calcul peut être confié à la BS sans augmenter le nombre des cycles opératoires : en effet, le cumul des agents est un élément de calcul et il faudrait un cycle pour l'introduire, différent du cycle nécessaire à l'impression des 1er et 2e étages. (L'IS ne peut pas convenir à cause du report en cycle 1). On confiera donc provisoirement tous les calculs au Gamma, sauf le Σ ventes donné à la BS.

2. Étude du programme Gamma.

A. *Régulation générale.*

Il y a trois programmes à prévoir :
— le programme carte à carte,
— le programme de 1er étage,
— le programme de 2e étage,

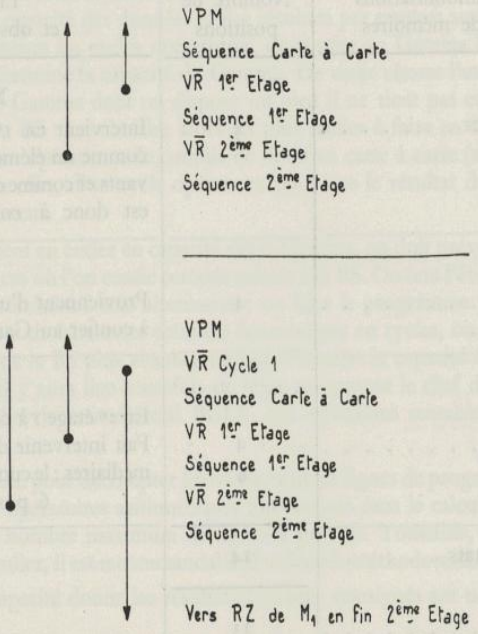


Fig. 165

On remarque que le dessin des cartes permet de traiter la CM sans sélection du programme carte à carte. En effet, la zone 76 à 80 n'est pas perforée en CM. D'autre part, les relations entre ces trois programmes doivent tenir compte du fait que les résultats des étages inférieurs servent dans le programme supérieur. Il s'agit donc d'une relation de la forme donnée par la figure 165.

Il faut aussi remarquer qu'un cycle opératoire est déclenché et que le calcul ne doit pas partir pour la CM. D'où la nécessité d'introduire une $\overline{VR}$ de cycle 1 après la VPM et invalidant les programmes en cycles 2 et 3 (le cycle 3 doit être fait à cause de l'espace). D'où la séquence ci-contre. La régulation générale du programme demandera donc 4 lignes.

B. *Le programme carte à carte.*

Le programme carte à carte est validé pour les cartes vente. Afin de n'avoir pas à stocker deux fois le total des ventes, d'une part pour l'extraction, d'autre part pour le calcul carte à carte, on imprimera le total des ventes depuis un totalisateur. On pourrait prendre l'IS, en lisant les cartes en brosse supérieure, mais l'économie de lignes est faible en regard des servitudes générales de l'IS qui doit être utilisée plutôt dans le cas général exposé plus haut lors de son étude.

Le programme sera donc celui de la figure 166.

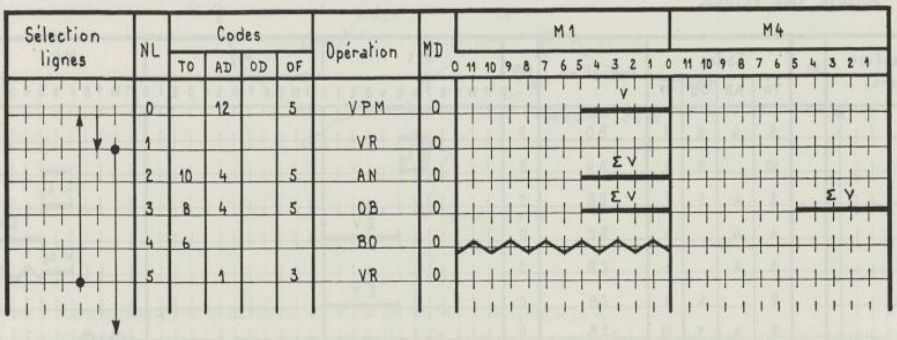


Fig. 166

C'est un programme de cumul vertical absolument classique. Toutefois, il faut remarquer qu'il ne doit démarrer que pour les cartes vente, au cas où des perforations seraient présentes dans les cartes maîtresses dans la zone 76-80. Cette précaution ne demande pratiquement aucune immobilisation d'organes. En effet, la commande V_0 de contrôle de départ du programme au cycle 1 seulement peut être contrôlée pour invalider le départ en cas d'indication (*fig.* 167).

La variante contrôlée par V_0 renverra, non pas à la VPM, mais à la dernière ligne du programme 2e étage, affectée à la remise à zéro de M_1. Cette remise à zéro sera effectuée par une BO AD = 0. En effet, lors de l'établissement du courant continu sur la machine, la M_1 et la MD sont garnies par des codes sans signification qu'il y a lieu d'effacer préalablement à tout travail et seule une BO peut y pourvoir.

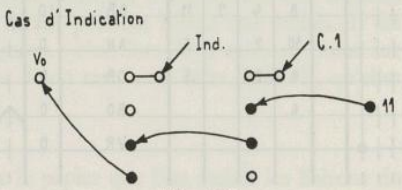


Fig. 167

C. *Le programme* 1er *étage.*

On y fait trois calculs :

— le calcul du nombre d'agents,

— le calcul de la remise R,

— le calcul du total des remises Σ R.

Le calcul du nombre d'agents se fait par décomptage des cartes maîtresses. Il suffit donc de distribuer un 1 en addition à chaque rupture de contrôle 2e étage pour réaliser ce cumul.

Le calcul de la remise doit être précédé d'une étude de tranche : une comparaison à 30.000 sera exploitée pour déclencher trois programmes qui se regrouperont par des VS : aucune partie commune, en effet, ne peut être décelée entre les trois programmes. Il convient d'autre part de rejeter le calcul de la remise à 25 % à la fin du programme de calcul de remise, afin de ne le trouver en M_1 que si elle a dû être calculée.

Le calcul du total des remises suit celui de remise et il est classique (*fig.* 168).

La remise calculée lors du précédent 1er étage doit être effacée par une ZB, parce que l'OB de renvoi de la remise n'a lieu qu'au cas où le Σ V est supérieur à 30.000. En effet, il ne peut être question de la valider pour les deux autres programmes : la M_1 contient la somme des ventes.

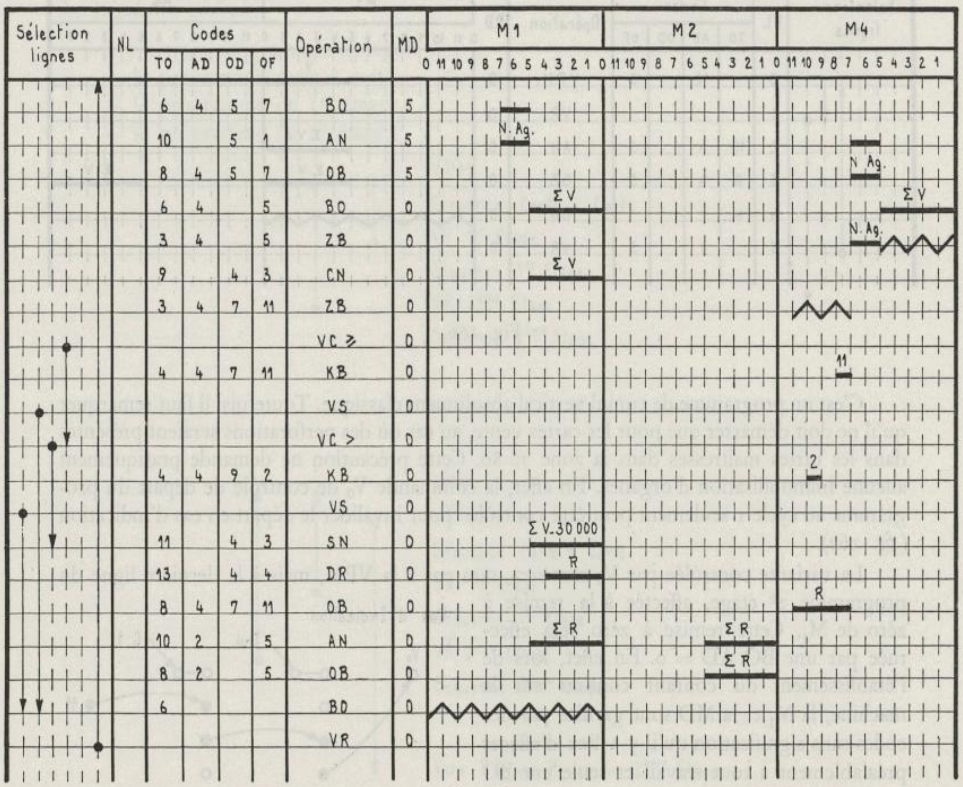


Fig. 168

La multiplication par 0,25 est obtenue par division par 4. Le diviseur n'est donc jamais nul.

Le regroupement des trois programmes se fait sur une BO de remise à zéro de la M_1.

D. *Programme de* 2e étage

Il débute par le calcul de la ristourne moyenne RM. La somme des ristournes est en M_1, le nombre d'agents en M_4. On effectue alors une DR qui donne la ristourne moyenne sur laquelle on effectue deux CN avec ADo pour comparaison à 500 puis à 1.000 exploitées par des VC de façon classique.

Le programme si RM $\leqslant$ 500 consistera alors en une remise à zéro de M_1. Le programme de prime fixée à 500 consistera en une distribution par BO de la constante 5 et celui de la remise en cas de RM $>$ 1.000 consistera dans le calcul de la remise.

On regroupe les 3 programmes par 2 VS ainsi qu'il a été exposé pour les séquences d'élimination.

Le diviseur, N*b*. agents, ne sera nul que s'il n'y a pas de contrôle 1er étage : cela n'arrivera qu'au 1er cycle 1. La sécurité sera obtenue facilement en validant le renvoi en VPM par la commande V_0 qui subira le contrôle indiqué figure 169.

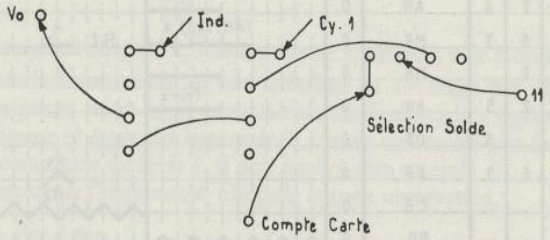


Fig. 169

Le programme sera donc celui de la figure 170 (page 156).

Il faut remettre à zéro le reste de la DR afin de pouvoir faire valablement les comparaisons à 1.000 et 500.

E. *Regroupement du développement.*

C'est lors du regroupement que l'on vérifie les séquences de remise à zéro d'étages : cette méthode de découpage fait souvent oublier les effacements à commander. C'est ainsi que l'on peut être amené à ajouter certaines lignes ou à en modifier certaines autres, ou encore à permuter des termes dans les mémoires pour obtenir plus de souplesse.

Il est important de remarquer qu'on n'a pas voulu donner ici l'explication d'une résolution ayant donné satisfaction sur la machine : on a voulu montrer la suite du raisonnement que fait un technicien étudiant son problème. C'est pourquoi il est possible que d'autres solutions soient plus élégantes ou que certains détails puissent mieux être traités. Il est possible aussi que certaines fautes aient été commises : c'est lors de la mise au point sur machine que l'on fera les rectifications nécessaires. Il est rare, avec cette méthode, qu'elles présentent de grandes difficultés.

F. *Liaisons BS-Gamma.*

C'est une fois le développement terminé sur le papier que l'on établit les liaisons du Calculateur et de la machine connectée.

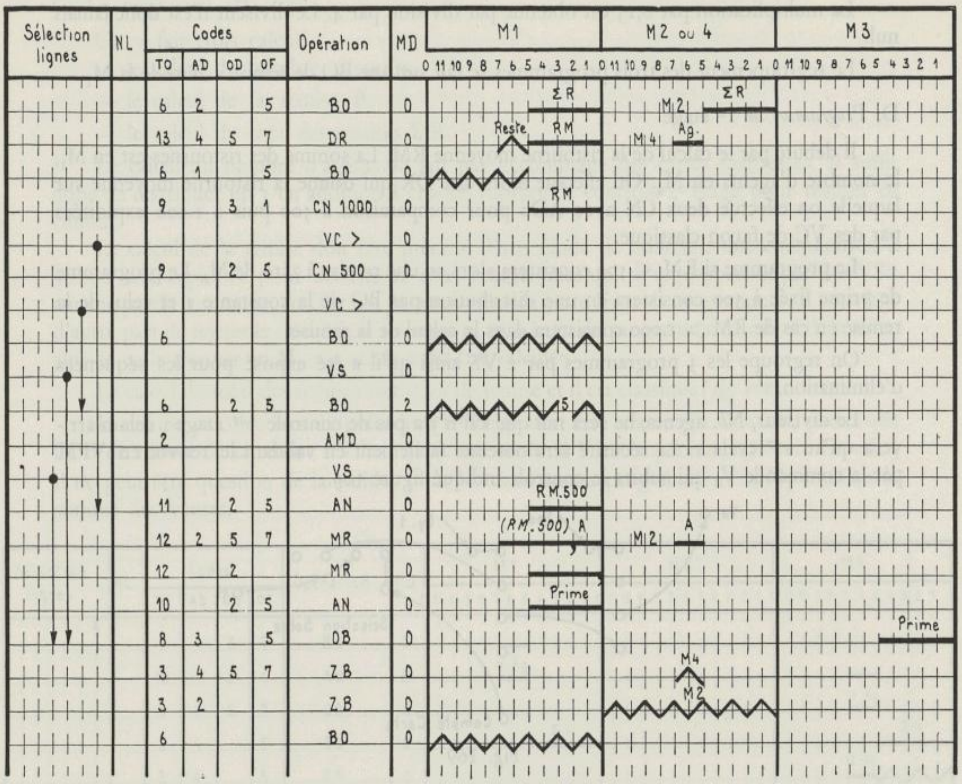


Fig. 170

Les commandes V_0, V_1, V_2 seront appelées comme il a été dit (*fig.* 171).

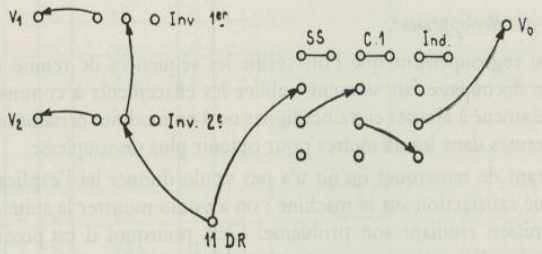


Fig. 171

Les connexions de contrôle sur BS sont classiques.

Les données seront prises en BI, la forme de l'impression le permettant : les extractions Gamma se trouvent au-dessous de la liste. Les introductions cinématiques seront contrôlées de 9 à 1. Celle de A, perforée sur la première carte maîtresse du secteur sera validée à l'indication. Celle de V sera contrôlée à tous les cycles 1 sauf celui d'indication. Les mémoires

étant différentes, il n'y aura pas à introduire d'alternatifs de sélection à l'enregistrement (*fig.* 172).

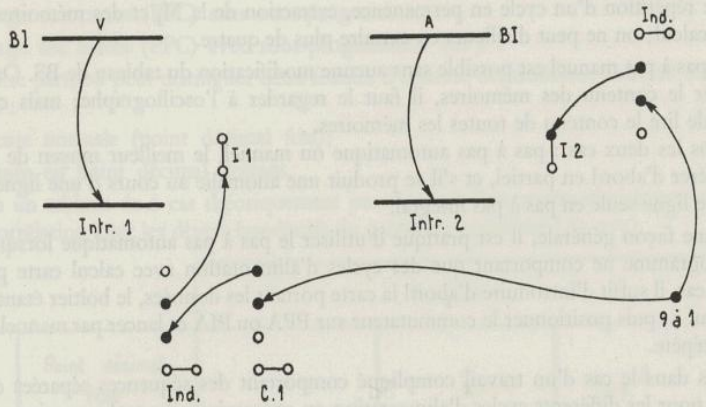


Fig. 172

L'extraction aura lieu au cycle 2, suivie du cycle 3, validé à vide à cause de l'espace.

L'extraction de la ristourne en M_4 sera contrôlée en 1er étage par une commande de cycle 2, celle de la prime en 2e étage par une commande de cycle 2 également validée sur le travail de l'inverseur 2e étage. La lecture de la somme des ventes sera faite sur le totalisateur et par une commande du cycle 2. Il sera remis à zéro au cycle 3.

Le regroupement sur l'imprimante n'appelle aucune observation.

G. *Mise au point - Pas à pas.*

La mise au point peut être effectuée à partir du raisonnement sur les anomalies constatées à l'impression, comme on le fait couramment en tabulatrice. Toutefois, il arrive souvent que l'on soit obligé de vérifier, ligne à ligne, le contenu des mémoires. C'est à cette fin qu'a été créé le dispositif de pas à pas.

Le dispositif consiste en un boîtier muni d'un commutateur à six positions et d'un bouton poussoir (« titilleur »). Ce boîtier est relié au Gamma par un câble souple. En fonctionnement normal, le connecteur du boîtier est remplacé par un bouchon. Pour connecter le boîtier à la place du bouchon, il est nécessaire de couper l'interrupteur 48 V de la BS.

Les différentes positions du commutateur correspondent chacune à un fonctionnement différent.

a) N - Calcul normal; dans cette position, le programme se déroule normalement;

b) O - Calcul zéro; dans cette position, le programme est bloqué;

c) P.P.M. - Pas à pas partiel manuel; dans cette position, chaque appui sur le bouton-poussoir autorise le départ d'une ligne de programme;

d) P.I.M. - Pas à pas intégral manuel; chaque appui sur le bouton-poussoir autorise le départ d'un train.

e) P.P.A. - Pas à pas partiel automatique; à chaque cycle de la machine connectée, le Gamma effectue une ligne de programme;

f) P.I.A. - Pas à pas intégral automatique; à chaque cycle de la machine connectée, le Gamma effectue un train de programme.

Le pas à pas automatique permet l'impression du contenu des mémoires après chaque ligne de programme ou après chaque train. Il nécessite des modifications du tableau de connexion : répétition d'un cycle en permanence, extraction de la M_1 et des mémoires utilisées dans le calcul; on ne peut d'ailleurs en extraire plus de quatre.

Le pas à pas manuel est possible sans aucune modification du tableau de BS. On ne peut imprimer le contenu des mémoires, il faut le regarder à l'oscillographe; mais ce moyen permet de lire le contenu de toutes les mémoires.

Dans les deux cas : pas à pas automatique ou manuel, le meilleur moyen de procéder est d'opérer d'abord en partiel, et s'il se produit une anomalie au cours d'une ligne, d'effectuer cette ligne seule en pas à pas intégral.

D'une façon générale, il est pratique d'utiliser le pas à pas automatique lorsqu'il s'agit d'un programme ne comportant que des cycles d'alimentation avec calcul carte par carte. Dans ce cas, il suffit d'introduire d'abord la carte portant les données, le boîtier étant en position calcul o; puis positionner le commutateur sur PPA ou PIA et lancer par manuel un cycle qui se répète.

Mais dans le cas d'un travail compliqué comportant des séquences séparées ou sélectionnées pour les différents cycles d'alimentation ou opératoires, il est plus pratique d'opérer en pas à pas manuel.

Le processus est alors le suivant :

a) Arrêter le déroulement des cycles, juste avant le départ de celui dont on veut examiner le programme;

b) Faire retomber les commandes S et V (Gamma M), (dans le cas de Gamma standard, il est nécessaire que les cames de maintien S et V soient réglées de façon à retomber avant le point d'arrêt);

c) Positionner le commutateur sur calcul o et lancer le cycle par manuel;

d) Positionner le commutateur sur PPM et effectuer les lignes de programme par appui sur le bouton-poussoir;

N. B. — L'opération IS ne peut être effectuée qu'en PPA en connexion tabulatrice; elle peut être effectuée en PPA ou en PPM en connexion PRD.

III. UTILISATION MATHÉMATIQUE

1. Généralités.

La nature des calculs mathématiques conduit à adopter des méthodes d'utilisation différentes en bien des cas de celles adoptées pour les calculs comptables. Le programme peut être affiché de deux façons différentes :

a) Soit directement sur le tableau du Gamma;

b) Soit indirectement sur ce tableau par l'intermédiaire de cartes programme grâce au dispositif PPC.

Dans ce dernier cas, l'affichage sur le tableau peut revêtir deux aspects :

α) Soit que les codes perforés dans les cartes soient significatifs d'opérations arithmétiques à raison d'une ligne Gamma par opération;

β) Soit que les codes perforés dans les cartes soient capables de renvoyer le programme sur des séquences préfabriquées affichées au tableau. Ces séquences s'appellent des « sous-programmes ».

On distingue donc trois cas :

a) Affichage sur tableau Gamma sans PPC. On l'appellera programme tableau (P.T.);

b) Affichage sur cartes (PPC) sans sous-programmes;

c) Affichage sur cartes (PPC) avec sous-programmes (SP).

D'autre part, on peut distinguer deux formes d'écriture des nombres pouvant être mises en œuvre :

a) L'écriture normale (point décimal fixe);

b) L'écriture en point décimal flottant.

D'où un tableau de 6 cas théoriquement possibles où l'on pourra choisir une organisation en corrélation avec les divers impératifs du problème considéré (*fig.* 173).

		Programme		
Ecriture	Point décimal fixe	1	2	3
	Point décimal flottant	4	5	6
		a) Tableau PT	b) PPC sans SP	c) PPC avec SP

Fig. 173

2. L'écriture des nombres.

Le mode d'écriture habituel des nombres conduit à affecter dans les mémoires les mêmes positions aux mêmes ordres d'unités, ce qui fait que la virgule se trouve toujours à la même place (*fig.* 174).

Par exemple

+	5	4	3	2	,	6	5			
−			3	2	,	9	0			
−				0	,	0	7	0	1	
+				0	,	0	0	0	1	7

Fig. 174

Ceci conduit à utiliser des capacités de mémoires calculées sur les limites droite et gauche extrêmes de l'ensemble des nombres traités et à laisser vierges pour chacun d'eux un certain nombre de positions.

En dehors de cet inconvénient, ce mode d'écriture rend difficile, lorsque les résultats de calculs varient considérablement en ordre de grandeur, de conserver pour chacun d'eux la même erreur relative.

Enfin, la programmation relative à ce mode d'écriture est généralement économique. mais assez longue et délicate, du fait des cadrages à respecter.

Une autre façon d'écrire les nombres, permet de simplifier la programmation qui peut devenir rigoureusement standard, valable pour tous problèmes, tout en assurant l'identité des erreurs relatives et en utilisant au maximum la capacité de la machine. Cette écriture est à la base de l'organisation de calcul dite « point décimal flottant ».

Ce mode d'écriture consiste à considérer chaque nombre comme un produit. L'un des termes du produit est une puissance de 10 caractéristique de l'ordre de grandeur du nombre. L'autre terme est le nombre lui-même, abstraction faite de la position réelle de sa virgule, écrit de telle sorte qu'il se présente sous la forme 0, *a*, *b*, *c*, *d*..., *a* étant le premier chiffre significatif. La valeur de l'exposant indique la position de la virgule :

$$+\,5.432{,}65 \;\ldots\ldots\ldots\; +\,0{,}543265 \times 10^{4},$$
$$-\,32{,}90 \;\ldots\ldots\ldots\; -\,0{,}329000 \times 10^{2},$$
$$-\,0{,}0701 \;\ldots\ldots\ldots\; -\,0{,}701000 \times 10^{-1},$$
$$+\,0{,}00017 \;\ldots\ldots\ldots\; +\,0{,}170000 \times 10^{-3}.$$

Le signe est facile à écrire : le dispositif d'opérateur de signe admet le code 1010 comme signe — écrit à la gauche du nombre. Pour l'exposant, il est donc impossible de repérer son signe.

C'est pourquoi, pour l'écrire dans le Calculateur, on le majore de 50, ce qui fait que 10^4 s'écrit 54, 10^2 s'écrit 52, 10^{-1} s'écrit 49 et 10^{-3} s'écrit 47. Il reste donc, par mémoire de 12 positions 9 positions significatives, de sorte que — 56.739, 4732 s'écrira ainsi que l'indique la figure 175.

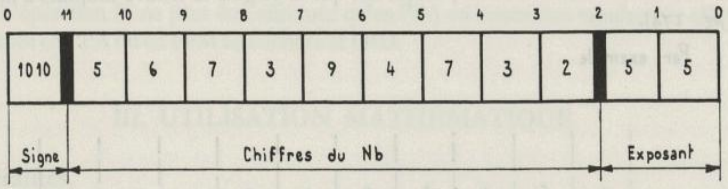


Fig. 175

En effet, le zéro à gauche ne présente aucun intérêt et il peut être sous-entendu sans dommage.

Il y a lieu d'étudier les répercussions de ce mode d'écriture dans chacune des opérations arithmétiques :

A. *En addition.*

Soit l'addition :

$$70 + 0{,}8 = 70{,}8$$

où les nombres sont écrits en décimal flottant sous la forme :

$$7.52 \text{ et } 8.50.$$

Ces deux nombres seront inscrits dans les mémoires selon la figure 176.

Ils ne peuvent être additionnés dans ces positions. Il est nécessaire de déplacer le nombre qui a le plus petit exposant, d'autant de positions vers la droite qu'il y a de différence entre les deux exposants, de façon à ce qu'ils soient placés de la façon indiquée figure 177.

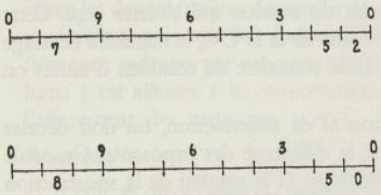


Fig. 176

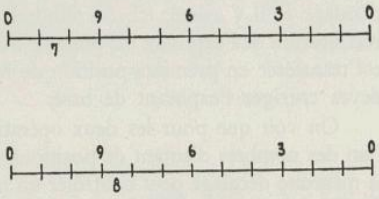


Fig. 177

Ainsi, la somme apparaîtra sous une forme correcte : 708. Mais il y a lieu de se préoccuper de la valeur de l'exposant de cette somme : ici c'est simple. Il est égal à l'exposant du plus grand nombre : 52.

Soit maintenant l'addition :

$$70 + 80 = 150$$

où les nombres sont écrits sous la forme 7.52 et 8.52. Ces deux nombres sont bien cadrés pour l'addition : leurs exposants sont égaux. Il n'y a donc pas lieu de les décaler avant addition. De sorte que la somme apparaîtra en M_1 comme l'indique la figure 178.

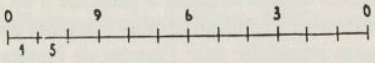


Fig. 178

Mais c'est maintenant la somme qui est mal cadrée : le report l'a fait progresser en 12e position, réservée au signe. Il faut donc ramener le nombre une position vers la droite, ce qui a pour effet de le diviser par 10. Il faut donc prendre comme exposant, non plus 52 mais $52 + 1 = 53$, ce qui donne l'ordre de grandeur exact du nombre 150.

B. *En soustraction.*

Pour la soustraction, les mêmes problèmes de décalage du nombre le plus petit en exposant se posent, du fait qu'addition et soustraction sont deux opérations connexes. Quant à la valeur de l'exposant, elle dépend du résultat de la soustraction qui peut être très différent en ordre de grandeur de l'ordre de grandeur des termes. Soient les deux exemples suivants qui illustreront respectivement ces deux points :

— Soit à faire $70 - 0{,}8 = 69{,}2$

où les termes sont écrits 7.52 et 8.50. Il faut décaler le plus petit 8.50 de deux positions, différence entre les deux exposants. La différence aura évidemment pour exposant celui du plus grand nombre, soit 52.

— Soit $70 - 69{,}991 = 0{,}009$

où les termes sont écrits 70.000.52 et 69.991.52. Il n'y a pas lieu de faire de décalages, les exposants égaux manifestent que les nombres sont de même ordre de grandeur. D'où la soustraction :

$$\begin{array}{rr} 70.000 & 52 \\ -69.991 & 52 \\ \hline 00{,}009 & 52 \end{array}$$

Il est évident que la différence ne satisfait pas à la règle d'écriture énoncée qui veut que la première position gauche soit un chiffre significatif. Il faut donc décaler le résultat vers la gauche jusqu'à apparition du chiffre significatif à l'extrême gauche. L'opération D C C seule peut s'en charger puisque, seule, la mémoire opérateur connaît son contenu qu'on ne connaît pas *a priori*. Mais en faisant ce décalage, on multiplie par 10 le résultat à chaque décalage et il y a lieu, ici, de minorer de 4 l'exposant du résultat qui s'écrira 9.48. Cette rectification de l'exposant est possible du fait que, au cours de la D C C, la mémoire décalage est transférée en première position de M_2, On peut ainsi connaître de combien d'unités on devra corriger l'exposant de base.

On voit que pour les deux opérations d'addition et de soustraction, on doit décaler l'un des nombres d'autant de positions qu'en indique la différence des exposants. Or, seule la mémoire décalage peut contrôler un nombre de décalages et le résultat de la soustraction de deux nombres est inévitablement contenu soit en M_1, soit en mémoire banale après OB.

C'est le rôle de l'opération BD, précédemment décrite, d'introduire en MD la différence des exposants, depuis M_2, afin de contrôler les décalages à effectuer avant l'opération arithmétique.

C. *Multiplication.*

Le calcul de l'exposant de base est très simple : en effet, l'exposant du produit est égal à la somme des exposants des facteurs minorée de 50, puisque chaque exposant a été majoré de 50.

Mais cet exposant de base doit subir éventuellement une correction. En effet, malgré la présence systématique d'un chiffre significatif à l'extrême gauche des facteurs, il peut se faire que la valeur arithmétique de ces chiffres entraîne la présence d'un zéro à gauche du produit : $2 \times 2 = 04$. On pourra en trouver l'explication plus complète dans l'exposé des caractéristiques de la table de Pythagore introduite dans l'exposé de l'opération division.

La D C C se chargera sans difficulté de décaler le premier chiffre significatif du produit à l'extrême gauche. Mais on aura ainsi multiplié le produit par 10 : on devra donc minorer l'exposant de base de 1 unité.

D. *Division.*

Le calcul de l'exposant de base est évidemment inverse de celui de la multiplication : on fait la différence en valeur absolue des deux exposants des facteurs que l'on majore de 50.

Les zéros à gauche qui peuvent se produire au quotient seront éliminés par la D C C et l'exposant de base sera corrigé à partir du nombre de décalages effectués et tenu en mémoire en M_2.

E. *Remarque.*

Ces considérations sont liées à la forme des opérations arithmétiques et n'étudient que les répercussions de l'écriture décimal flottant sur leur nature : c'est lors de l'étude concrète des applications que l'on étudiera les séquences opératoires réelles.

3. Étude du cas 1 : Point décimal fixe - Programme tableau.

Ce cas ne diffère en rien des calculs comptables. De sorte que tout ce qui a été dit relativement à ces calculs s'applique.

4. Étude du cas 2 : Point décimal fixe - Programme PPC sans SP.

Un exemple très simple est donné par la Pl. 27. Soit à calculer $B = \frac{UB.10^3}{V.\sqrt{3}}$.

UB et V sont les données, introduites en M^4 et M^5. Le calcul tient sur les lignes de 9 à 5 de la carte programme. La ligne 9 est destinée à la distribution de la constante $\sqrt{3} = 1{,}732$ par 4 KB successives sur M_2. La ligne 8 ne porte que 4 opérations : une BO d'accès de V en M_1 et une MR qui calcule le dénominateur en M_1. Sur la ligne 7, une BO avec adresse 1 efface les deux dernières décimales de ce dénominateur et une OB le place en M_2 en position de diviseur, puis une BO donne accès en M_1 à UB que l'on multiplie par 10^3 par une opération avec cadrage, par exemple une CN. Sur la ligne 6, on place une DR qui calcule B. La ligne 5 est affectée à la conservation des seules décimales de B, à son transfert en M_3 et à l'effacement des mémoires 1 et 2.

Remarques :

a) Il faut se préoccuper du nombre de tours, ainsi qu'il a été dit déjà. On ne peut rien faire après les opérations récurrentes qui doivent figurer sur la dernière ligne du tableau Gamma en relation avec le PPC. Rappelons que les trois premières opérations ne doivent pas excéder 40 tours, y compris les T*p* des 4 opérations.

b) On ne peut utiliser les variantes qu'à l'intérieur des 4 lignes du programme PPC. Si on doit sauter une séquence assez longue, on utilise une VPM sur laquelle on se renvoie par une VC conditionnée par une CN. On sautera ainsi les lignes de cartes antérieures à celle où l'on reprend les calculs. Si la séquence dure sur plusieurs cartes, la VPM ayant une période de un cycle, on répète la CN et la VC en tête de chacune des cartes, sur la ligne des 9.

c) On peut faire l'opération IS sur la ligne des 12.

5. Étude du cas 3 : Point décimal fixe - PPC avec SP.

Chaque fois que des séquences se répètent dans une chaîne de calculs, on a intérêt à utiliser les lignes disponibles du tableau pour afficher des sous-programmes exécutant ces séquences. Les codes des cartes renvoient alors par VS toujours au début de ces séquences chaque fois qu'il convient, la fin de ces séquences renvoyant aux lignes de PPC par leur VPM (*fig.* 179).

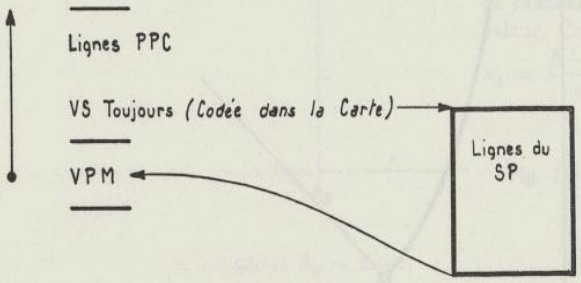


Fig. 179

On prendra un exemple très simple afin de montrer la structure de l'organisation plus qu'un calcul compliqué qui n'apporte rien de plus à la compréhension. Soit l'équation du 2e degré.

$$ax^2 + bx + c = 0$$

où les racines sont données par la formule :

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

On supposera que $b^2 - 4\,ac$ est toujours positif et compris entre les limites A_1 et A_2.

Ce calcul comporte une séquence qui revient à chaque résolution et qui correspond à l'extraction de la racine. On affichera donc la séquence de racine sur le tableau en sous-programme alors que le calcul de $b^2 - 4\,ac$ et le calcul de la fraction seront affichés sur les cartes programme. Ces cartes renverront au sous-programme après le calcul de $b^2 - 4\,ac$. Ce sous-programme sera suivi du calcul de la fraction. Bien entendu, on pourrait tout mettre sur le tableau et se ramener au cas 1, mais il s'agit ici d'illustrer une méthode plus que de résoudre un problème concret et complet que l'utilisateur saura d'autant mieux organiser qu'il aura mieux saisi les différentes méthodes qui s'offrent à son choix.

Nous étudierons à part le sous-programme et le programme par cartes.

A. *Etude du sous-programme.*

La racine sera extraite à partir de la formule de Newton. Soit un nombre A sous radical et x sa racine carrée. Soit une valeur quelconque x_n. La formule suivante donne une autre valeur x, plus proche de x que ne l'était x_0.

$$x_1 = \frac{1}{2}\left(x_0 + \frac{A}{x_0}\right).$$

En portant x_1, à la place de x_0 dans cette formule, on obtient une valeur x_2 plus approchée de x que x_1. Par itérations, on obtiendra donc x_n aussi approché que l'on voudra de x. A chaque itération, le nombre de chiffres exacts de la racine est doublé.

Pour illustrer cette formule, on peut tracer la courbe de la fonction $x = \frac{A}{x'}$ (*fig.* 180).

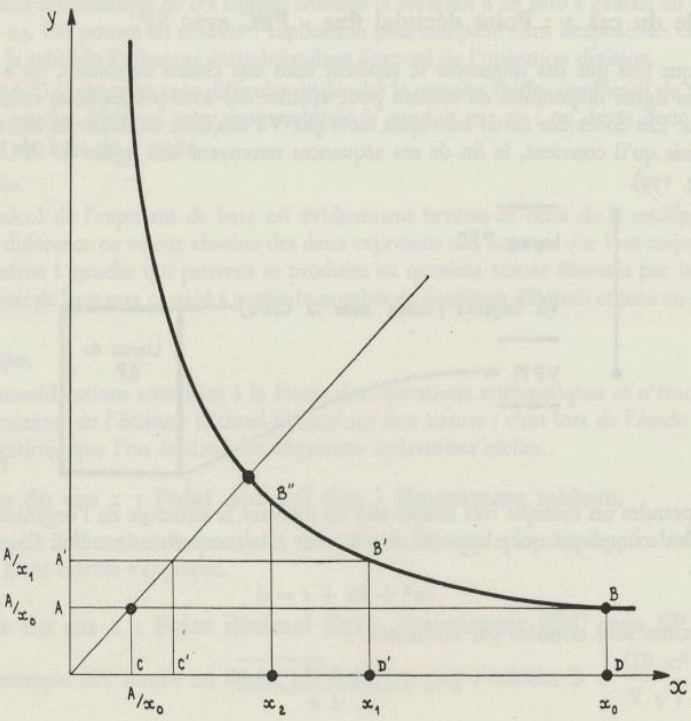


Fig. 180

C'est une hyperbole équilatère dont le sommet a pour ordonnée, égale à l'abscisse, la valeur de $\sqrt{A}$. Soit une valeur x_0 sur l'axe horizontal. Il lui correspond sur l'axe vertical la valeur $\frac{A}{x_0}$ qui figure dans la formule. La droite AB coupe la bissectrice des axes en un point dont la projection sur les abscisses se trouvent à la distance $x_0 - \frac{A}{x_0}$ de x_0. Le milieu de ce segment CD a pour abscisse x_1 égal à $\frac{1}{2}\left(x_0 + \frac{A}{x_0}\right)$. En effet x_1 est égal à :

$$\frac{A}{x_0} + \frac{1}{2}\left(x_0 - \frac{A}{x_0}\right) = \frac{A}{x_0} + \frac{x_0}{2} - \frac{A}{2x_0} = \frac{1}{2}\left(x_0 + \frac{A}{x_0}\right).$$

A x_1, correspond sur les ordonnées le point A' dont l'ordonnée est égale à $\frac{A}{x_1}$. La droite A'B' coupe la bissectrice en un point dont la projection est égale en abscisse à $\frac{A}{x_1}$. Le milieu du segment C'D' donne une valeur x_2, etc. On voit que les valeurs x_3, x_4... x_n, seront de plus en plus proches du sommet, donc de la racine.

Il y a lieu, d'autre part, de choisir la première valeur approchée x_0 : plus elle sera proche de la racine, moins sera grand le nombre d'itérations. On peut représenter la courbe $x = \sqrt{A}$ (*fig.* 181). Entre les limites A_1 et A_2 du discriminant, on confondra la courbe avec la droite $a_1 a_2$. On calcule à la main les valeurs α et β de son équation :

$$A = \alpha x_0 + \beta$$

Les valeurs α et β ainsi trouvées seront introduites dans le Gamma en constantes et chaque valeur de A permettra de calculer x_0 lié à cette valeur. On aura en effet

$$x_0 = \frac{A - \beta}{\alpha}$$

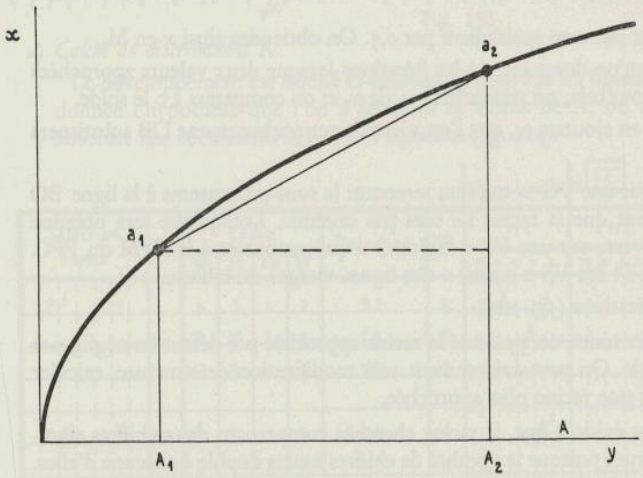


Fig. 181

Soient :

$$A_1 = 900 \text{ et } A_2 = 2.500.$$

On aura :

$$900 = 30\,\alpha + \beta$$
$$2.500 = 50\,\alpha + \beta$$

puisque $\sqrt{900} = 30$ et $\sqrt{2.500} = 50$. On a, en effet, intérêt à prendre comme limites de A des carrés parfaits proches des vraies valeurs limites. Le système écrit ci-dessous donne les valeurs de α et β :

$$\alpha = 80$$
$$\beta = -1.500$$

On peut remarquer que dans l'hypothèse de A toujours positif, β sera toujours négatif, la droite $a_1\, a_2$ coupant toujours l'axe des abscisses sur sa partie négative. Il serait égal à o si A était égal à o. On calculera donc x_0 selon la formule $x_0 = \dfrac{A + \beta}{\alpha}$

D'où la séquence de calcul de x_0 (*fig.* 182).

NL	Codes TO	AD	OD	OF	Opération	MD
	10	3		4	AN	0
	13	3	4	6	DR	0
	8	2	4	6	OB	4

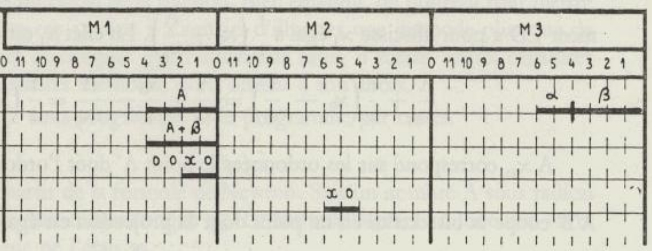


Fig. 182

On commencera ensuite les itérations en calculant $\dfrac{A}{x_0}$. On a mis x_0 en réserve et on réintroduira A en M_1. On effectuera la division réduite qui formera $\dfrac{A}{x_0}$ en M_1 et on additionnera x_0. Une MR divisera ensuite par 2 en multipliant par 0,5. On obtiendra ainsi x en M_1.

On sait, d'autre part, qu'on devra arrêter les itérations lorsque deux valeurs approchées successives seront égales. Pour cela, on retranchera x_0 de x_1 et on comparera à o le solde.

Une AN reformera x_1 en ajoutant x_0 que l'on vient de retrancher et une OB substituera x_1 à x_0.

La CN sera exploitée par une VC > qui fera remonter le sous-programme à la ligne BO de transfert de A en M_1 tant que la racine ne sera pas obtenue. Lorsqu'elle sera obtenue $x_n - x_{n-1} = 0$: la VC renverra sur une VS « toujours » qui remontera à la VPM du PPC. Cette VS économise les Tp Tl des VS « jamais » des lignes vierges du tableau.

D'où la séquence d'itérations (*fig.* 183).

Dans l'exemple choisi et traité, on a calculé la racine approchée par défaut en négligeant les décimales après la virgule. On peut évidemment sans modification de structure, calculer ces décimales afin d'obtenir une racine plus approchée.

Dans cet exemple, il est évident que, la racine cherchée comprenant deux chiffres significatifs, deux itérations suffiront puisque le nombre de chiffres exacts double à chacune d'elles. Cette évaluation du nombre des itérations présente toujours de l'intérêt lorsqu'on travaille avec des sous-programmes, afin de leur laisser le temps de se faire entre deux traitements en PPC.

Il est à noter qu'avant le S. P. la valeur A se trouve en mémoire 2 dans les positions de o à 6 et qu'après le S. P le $\sqrt{}$ se trouve en mémoire 2 dans les positions de 4 à 6.

B. *Programme PPC.*

Ce programme se divise en deux parties :

Une première partie doit calculer le discriminant A et renvoyer au sous-programme de racine carrée, une deuxième partie calculera les deux racines x' et x'' de l'équation.

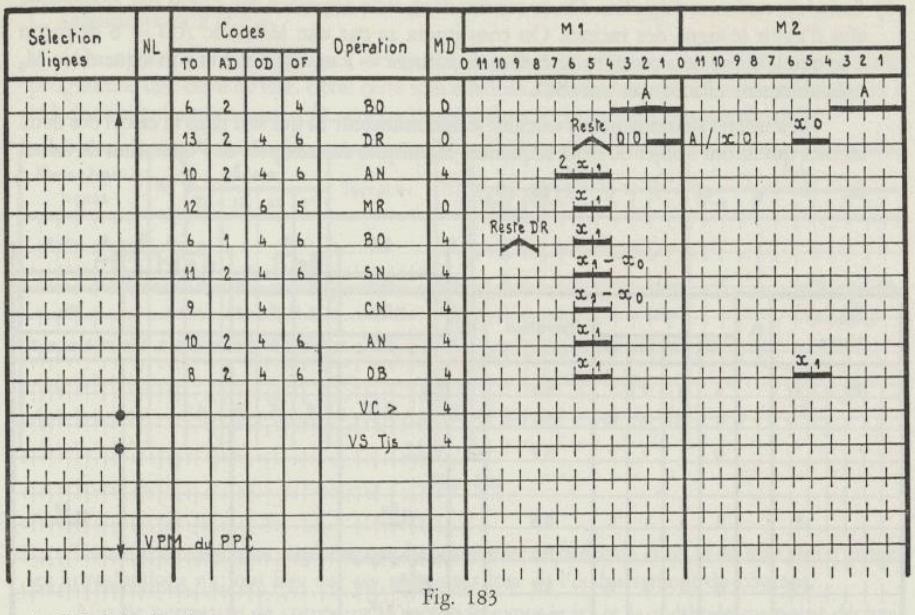


Fig. 183

a) *Calcul du discriminant* A.

Le développement est donné ci-dessous. Il tient sur les lignes de 9 à 6 de la carte. Étant donnée l'hypothèse que l'on a faite sur la valeur de A, on ne traitera que les valeurs absolues des coefficients *a*, *b*, *c* de l'équation (*fig.* 184).

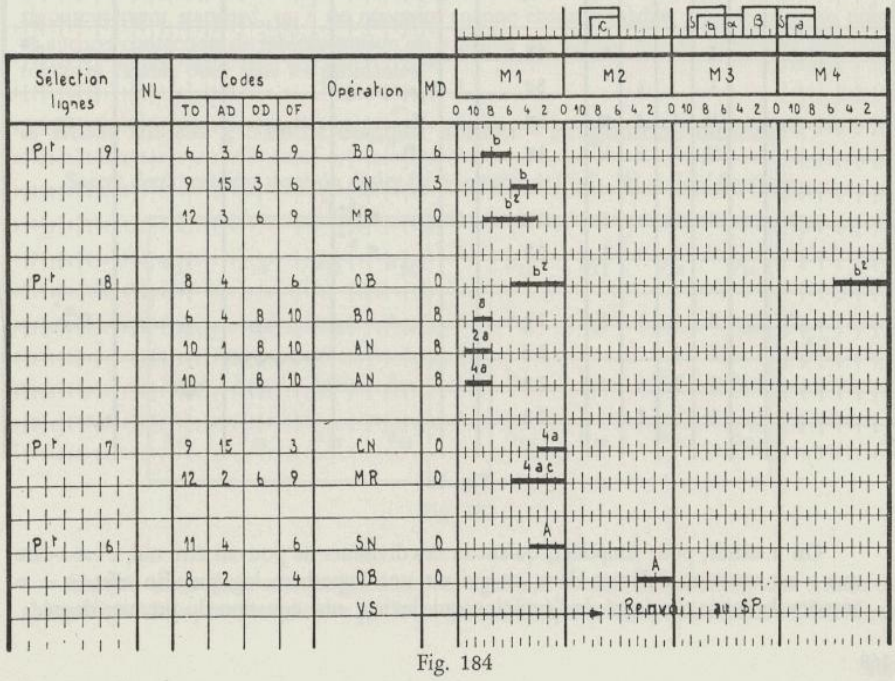


Fig. 184

b) *Calcul des racines de l'équation.* On se servira dans cette séquence des signes des coefficients afin d'avoir le signe des racines. On constituera $2a$ par une MR avec AD = 0 de façon à ce que le signe de a reste inaltéré. On constituera $-b$ en retirant b et son signe de la M_1 préalablement effacée par une BO.

On a intérêt à commencer à calculer le dénominateur $2a$ qui sert dans le calcul des deux racines qui feront l'objet de deux séquences identiques au sens près de l'opération de calcul de $-b-\sqrt{}$ et $-b+\sqrt{}$. (*fig.* 185).

Sélection lignes	NL	Codes TO	AD	OD	OF	Operation	MD	M1	M2	M3	M4
Pt 14		6	4	9		BO	9	a			
		9	15	7	10	CN	7	a			
		12		9	2	MR	0	2a			
13		8	4	4	8	OB	4	2a			2a
		6		6		BO	6				
		11	3	6	10	SN	6	-b			
		11	2	4	6	SN	4	-b-√			
12		9	15		4	CN	0	-b-√			
		13	4	4	8	DR	0	x'			
11		8	4		4	OB	0				x
		6		6		BO	6				
		11	3	6	10	SN	6	-b			
		10	2	4	6	AN	4	-b+√			
10		9	15		5	CN	0	-b+√			
		13	4	4	8	DR	0	x''			
11		8	4	4	9	OB	4				x''
		3	2			ZB	4				
		3	3	6	10	ZB	4				
		3	4	9		ZB	4				

Fig. 185

On a admis, pour simplifier, qu'aucun des diviseurs ne pouvait être nul, c'est-à-dire que a est toujours différent de 0, ce qui est une hypothèse logique. En effet $a = 0$ entraîne la baisse du degré de l'équation qui devient une équation du premier degré.

C. *Introduction de* α *et* β.

On mettra devant le paquet des cartes données interclassées avec leur unique carte-programme une carte de tête. Cette carte sera destinée à effacer les précédents coefficients α et β et à distribuer les nouveaux. Elle sera codée ainsi dans le cas qu'on a exposé (*fig.* 186).

Sélection lignes	NL	Codes TO	AD	OD	OF	Opération	MD	M1	M4
P r 9		6				BO	0		
		3	4		6	ZB	0		
		4	4	2	5	KB	0		5
		4	4	3	1	KB	0		1
8		4	4	5	8	KB	0		8

Fig. 186

On a mis devant cette séquence une BO de remise en ordre de la M_1 et de MD afin d'effacer les introductions qui ont lieu sur ces mémoires lors de l'établissement du courant.

Afin de permettre de comparer la méthode exposée ici et la méthode du point décimal flottant, on prendra comme exemple d'application du cas 6 la résolution de la même équation du 2e degré. On verra que les cartes programme sont plus nombreuses : de 1 carte, on passe à 5 cartes programme. On résoudra donc 25 équations au lieu de 75 à la minute. Toutefois, il faut remarquer que cette baisse de vitesse est compensée par le fait que le travail devient rigoureusement standard, qu'il ne nécessite aucune étude préalable, aucune mise au point et aucune confection de tableau puisqu'on y utilise un tableau mis au point une fois pour toutes et valable pour tous les problèmes.

6. Étude du cas 4 : Point décimal flottant - Programme tableau.

Soient deux matrices dont on désire faire le produit : $[a] \cdot [b] = [c]$ (*fig.* 187).

$$\begin{vmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{24} & a_{43} & a_{44} \end{vmatrix} \times \begin{vmatrix} b_{11} & b_{12} & b_{13} & b_{14} \\ b_{21} & b_{22} & b_{23} & b_{24} \\ b_{31} & b_{32} & b_{33} & b_{34} \\ b_{41} & b_{42} & b_{43} & b_{44} \end{vmatrix} = [c]$$

Fig. 187

Chaque terme de la matrice produit sera égal à la somme des produits des termes de chaque ligne de la première matrice par les termes de chaque colonne de la seconde. Il faut donc

pour calculer chaque terme de [*c*], associer les termes de [*a*] portant le même numéro de ligne que lui et les termes de [*b*] portant le même numéro de colonne (*fig.* 188).

$$[c] = \begin{array}{|cc|} \hline a_{11}b_{11} + a_{12}b_{21} + a_{13}b_{31} + a_{14}b_{41} & a_{11}b_{12} + a_{12}b_{22} + a_{13}b_{32} + a_{14}b_{42} \\ a_{21}b_{11} + \text{———————————} & \text{———————————} \\ \text{———————————} & \text{———————————} \\ \text{———————————} & \text{———————————} \\ \text{———————————} & a_{41}b_{14} + a_{42}b_{24} + a_{43}b_{34} + a_{44}b_{44} \\ \hline \end{array}$$

Fig. 188

A. *Mode opératoire.*

On perfore les matrices [*a*] et [*b*] à raison de 1 élément par carte avec indication du numéro de ligne et du numéro de colonne. La matrice [*a*] est triée par lignes, [*b*] est triée par colonnes. On place [*a*] en piste perforation et intercalant une carte vierge après chaque ligne et [*b*] en piste de lecture de la PRD. On fait défiler les deux pistes en même temps et on obtient la matrice produit par diagonales.

Lorsque la piste de lecture est vide, on replace le paquet de cartes [*b*] sans en modifier l'ordre. Lorsque la piste de perforation est vide, on replace le paquet de cartes [*a*] dans le magasin de perforation après avoir mis la première ligne en fin de paquet et intercalé une carte vierge après chaque ligne.

Il faut passer la matrice [*a*] autant de fois qu'il y a de colonnes dans [*b*], puis trier les cartes résultats si on désire les avoir dans un ordre normal ligne par ligne ou colonne par colonne.

On peut, pour contrôler, ajouter à [*b*] une colonne contrôle égale à la somme des éléments de chaque ligne. On obtiendra dans [*c*] une colonne de contrôle qui devra être égale à la somme des éléments ligne par ligne.

B. *Programme Gamma.*

Les deux éléments conjugués A et B sont introduits dans les mémoires M_5 et M_6. La somme cherchée Σ sera extraite de la M_7. Tous ces éléments sont écrits en point décimal flottant (Pl. 28-29).

On commence par calculer l'exposant de $A \times B$, égal comme on l'a déjà vu à $a + b - 50$. Une « VMS » renvoie le programme aux remises à zéro des mémoires 5 et 6. En effet, $a + b - 50 < 0$ signifie que l'un au moins des deux termes est à considérer comme nul. Après la VMS est un BO qui stocke l'exposant obtenu en M_5 à la place de *b*. D'où la séquence :

Calcul de l'exposant de base (*fig.* 189).

On remarque que la ligne 1 porte une CN dont le rôle sera expliqué par la suite.

Ayant calculé l'exposant de base, on procède à la multiplication complète de A par B, puis on remet à zéro la M_3 pour des raisons qui seront examinées plus loin. D'où la séquence :

Calcul de $A \times B$ (*fig.* 190).

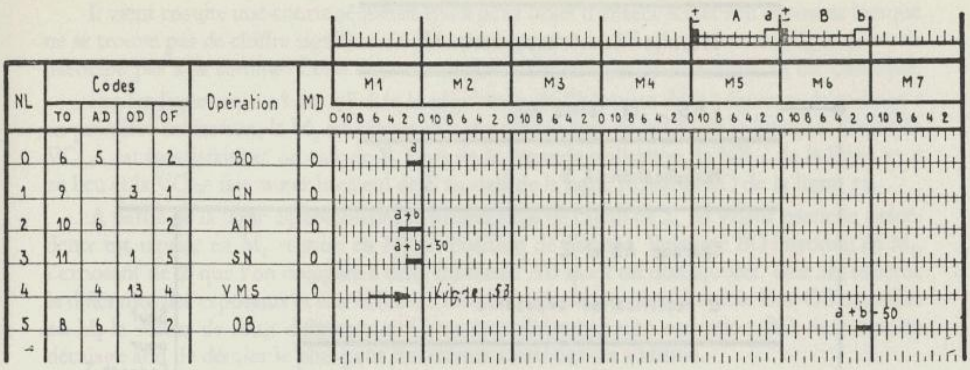


Fig. 189

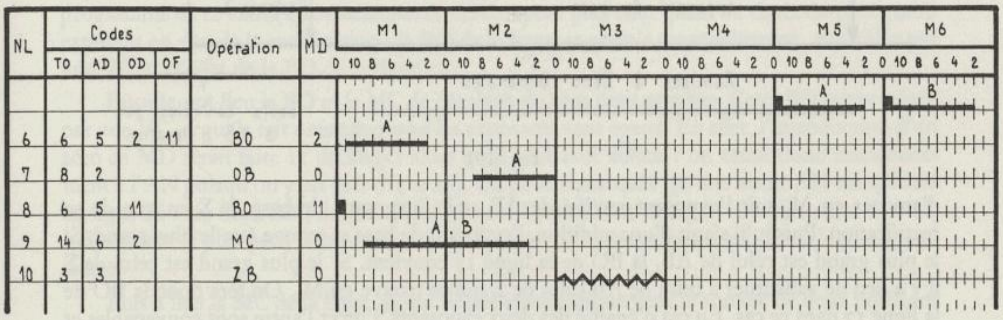


Fig. 190

La séquence qui suit débute par une AMD préalable à une DCC afin de cadrer correctement AB et de rectifier l'exposant de base $a + b - 50$. Cette séquence de cadrage et de rectification sera réutilisée pour cadrer Σ et pour rectifier son exposant. On retrouvera donc le contrôle par KB de répétition de séquence exposé plus haut (fonctions principales du Gamma). On aura donc une structure de cette forme pour le programme qui suit (*fig.* 191).

Il devient difficile de suivre l'exposé si l'on se tient à l'examen de séquences partielles : on se reportera donc au développement général de la planche 28.

Sur la ligne 11 se trouve une AMD dont le rôle est de positionner la MD à 9 afin de contrôler 9 décalages au maximum. La DCC qui la suit cadre A.B tandis que se transfère en première position de M_2 le contenu de cette mémoire avec un retard de 1 unité sur elle comme il a été dit : M_2 contient donc $9 - (\delta - 1)$, δ étant le nombre de décalages effectués.

Une BO à la ligne 13 maintient en M_1 la partie de AB à conserver et une OB la transfère en M_6. Le cadrage de AB est terminé. Il aura lieu de la même manière pour Σ lorsque l'on remontera de la ligne 50.

La rectification de l'exposant de base, valable pour AB et Σ, débute par une BO qui l'amène en M_1. La séquence devra subir une petite variante selon que l'on traite l'exposant de AB ou celui de Σ.

En effet, la BO de la ligne 15 amène en M_1, en première utilisation, l'exposant de base de AB ($a + b - 50$) en vue de le rectifier, alors que lors de la deuxième utilisation, il y a lieu

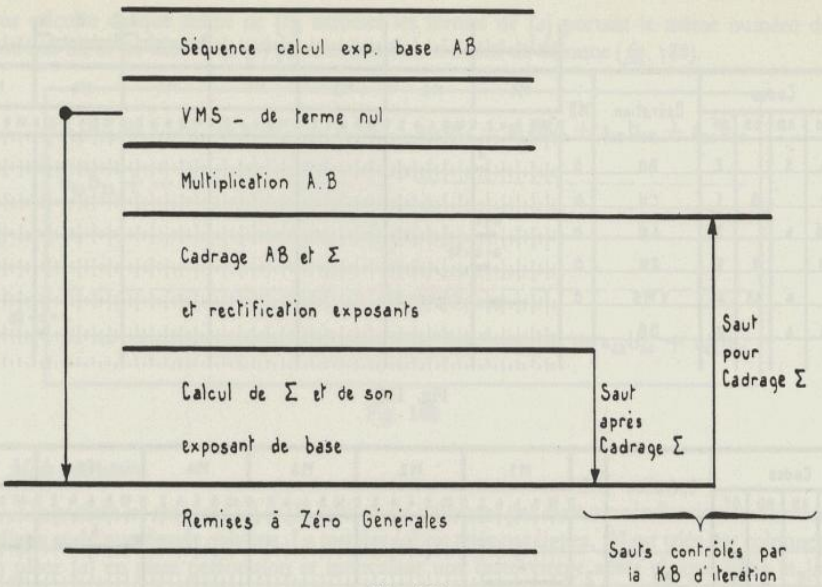


Fig. 191

d'amener en M_1 soit l'exposant rectifié de AB, soit l'exposant de base de Σ en vue de sa rectification. Puisqu'il s'agit d'une addition, l'exposant de base se trouve être le plus grand. Si le plus grand est celui de AB, la BO de la ligne 15 convient. Si le plus grand est celui de Σ il y a lieu de substituer à celui de AB celui de Σ qui se trouve en M_5. On fera donc la BO de la ligne 17 dans ce cas. En cas d'égalité des deux exposants, l'un et l'autre sont convenables et il n'y a pas lieu de faire de substitution. C'est pourquoi la VC de la ligne 16 est affichée en $\leqslant$.

Cette VC exploite, lors de la première utilisation, la CN de la ligne 1 qui force le comparateur en $<$ et lors de la seconde utilisation, elle exploite la CN de la ligne 32 où l'on compare les exposants de AB et Σ.

Pour rectifier les exposants ainsi introduits, on leur ajoute la quantité $9 - (\delta - 1)$ contenue en M_2. Cette quantité est le complément à 10 du nombre de décalages à gauche de la DCC. Elle mesure donc, sous cette forme, les multiplications par 10 du nombre ainsi opérées. Comme il s'agit d'un complément à 10, on le rectifie en retirant, en principe 10, afin que l'on retrouve la valeur δ mesurant les décalages. Or, δ est négatif, donc, une fois retiré cet excédent venant du complément à 10, on minorera l'exposant de la valeur δ pour compenser les multiplications par 10 opérées par la DCC.

Il est bien dit « en retirant en principe 10 ». En fait, c'est 8 que l'on retire. En effet, on a cadré la MC de la ligne 9 pour que le chiffre de gauche du produit soit au maximum en position OF = 9, donc deux positions trop à droite. L'une de ces positions compense le T_3 systématique de début de DCC. L'autre est justifiée pat la nature de l'addition en vue de Σ, opération qui peut faire apparaître un report sur la 12e position. Il faudra donc en ligne 49 afficher une MC de décalage à droite de la somme, qui la décale d'une position à cause de ce report, puis d'une seconde à cause du T_3 systématique de la DCC.

C'est donc 8 et non 10 qu'il faut retirer pour obtenir l'exposant exact : c'est le rôle de la *SB* de la ligne 19. Une OB en ligne 20 envoie l'exposant rectifié en M_5.

Il vient ensuite une courte séquence qui a pour objet d'effacer AB et son exposant lorsque ne se trouve pas de chiffre significatif à l'extrême gauche de AB. Il est alors négligeable et ne participe pas à la somme. Cette séquence, affichée de la ligne 21 à la ligne 24 est classique.

Sur les lignes 25 à 28 est affichée la régulation de l'itération de la séquence de cadrage : en première utilisation, la M_3 est vierge en 12e position lors de la CN de la ligne 26. Donc, la VC $\neq$ est insatisfaite et on calcule Σ. Lors de la seconde utilisation, la KB 2 de la ligne 27 a eu lieu et la VC $\neq$ fait sauter le calcul de Σ au-delà de la ligne 50, sur la BO de la ligne 51.

A partir de la ligne 29 commence le programme de calcul de Σ. Sa valeur partielle précédente est reprise en M_5 et mise en M_7 en position de donnée. Puis une BO introduit en M_1, l'exposant de Σ que l'on compare à celui du terme AB que l'on doit ajouter. Une SN effectue la différence des exposants et une MR avec AD 1 en calcule la valeur absolue. Une OB transfère en M_2 la valeur de cette différence qui sera introduite par une opération BD en mémoire décalage afin de décaler le plus petit des termes à associer en addition.

Auparavant sur la ligne 32 a été faite la comparaison des deux exposants afin de garnir la M_1 avec le terme qui comporte le plus petit exposant puisque c'est lui qui doit être décalé. Le programme de ce choix est symétrique de celui exposé plus haut quant au choix du plus grand exposant en vue de la rectification : la BO de la ligne 40 annule éventuellement, contrôlée par une VC $<$, l'effet de la BO de la ligne 38.

Ensuite ont lieu la BD et la MC de décalage du plus petit nombre. Cette BD est protégée par une VC $=$ qui la fait omettre quand les exposants sont égaux. En effet, l'introduction d'un zéro en MD ferait faire 12 décalages alors qu'il n'en faut aucun : on saute donc immédiatement à l'AN puisqu'on y est prêt (ligne 48). On peut remarquer qu'une autre AN est affichée en ligne 46 : ces deux AN, dont on fait l'une ou l'autre, sont chargées d'additionner à la M_1 qui contient le plus petit terme la M_5 ou la M_6 selon qu'elles contiennent ou non le plus grand terme.

Puis a lieu la MC de la ligne 49 qui décale systématiquement la somme de deux positions selon qu'il a déjà été dit.

Il reste à examiner le rôle de la SN de la ligne 36. Elle a pour rôle de retirer 8 à la valeur absolue de la différence des exposants et le signe de ce solde est exploité par une VMS $+$ qui fait omettre, lorsqu'elle est validée, le choix du plus petit terme et son décalage. On aboutit directement à la ligne 44 où une BO efface l'exposant qui se trouve en M_1 afin de préparer l'addition des termes.

On voit donc que M_1 sera garnie si le signe résultant de la SN est négatif et vierge si ce signe est positif. En effet, si un terme est négligeable, on aura $|a - b| \geqslant 9$. Dans ce cas, on aura $|a - b| - 8 \geqslant 1$ donc positif : c'est pourquoi on soustrait 8.

C. *Extraction de* Σ.

L'extraction de Σ envoyé en M_7 est déclenché au cycle suivant le calcul décrit. Il suffit après cette extraction d'inclure dans le programme un effacement de M_7 contrôlé par l'affichage de la ligne 63 à laquelle renvoie une VPM 11,14. Cette ligne efface M_7 en cas de détection d'un 11 colonne 80 en BLL.

7. Étude du cas 5 : Point décimal flottant - Programme PPC.

L'exemple choisi est celui de l'extraction de la racine carrée dont on a vu la méthode arithmétique et la résolution en point décimal fixe et sous-programme. La méthode arithmétique d'itération est la même. Seuls, sont différents le choix de x_0 et la programmation (Pl. 30).

A. *Calcul de l'exposant de la racine.*

Sur la ligne 9 de la première carte sont affichées : une BO qui amène l'exposant *e* de la quantité sous radical, une AN qui lui ajoute 1, une seconde qui lui ajoute 50, une DR qui divise par 2, enfin une OB qui envoie l'exposant calculé en M_6. Cet exposant est entièrement calculé et correct.

En effet, si *e* est l'exposant d'un nombre sous radical écrit sous la forme $a.10^e$, on peut écrire :

$$\sqrt{a.\,10^e} = \sqrt{a.\,10^2}^{\,\frac{a}{2}}.$$

Toutefois, il faut ajouter 50 à *e* pusique malgré la division par 2 on doit retrouver cette majoration systématique. Il faut aussi ajouter 1 afin que l'exposant, quand il est impair, soit correct, le Gamma ne pouvant pas traiter d'exposants fractionnaires.

Un tableau fera comprendre cet impératif (*fig.* 192).

	Quantité sous radical	Ordre de grandeur racine
a	$1 = 10^0$	$1 = 10^0$
b	$10 = 10^1$	$0{,}3 \,.\, 10^1$
c	$100 = 10^2$	$1 \,.\, 10^1$
d	$1000 = 10^3$	$0{,}3 \,.\, 10^2$
e	$10000 = 10^4$	$1 \,.\, 10^2$
f	$100000 = 10^5$	$0{,}3 \,.\, 10^3$

Fig. 192

On voit que, entre deux nombres sous radicaux figure, pour leur racine, un coefficient égal à 1 ou 0, 3 selon que l'exposant sous radical était pair ou impair et que l'exposant de la quantité sous radical doit être majoré de 1 en cas d'imparité pour que l'exposant de la racine soit correct.

D'autre part, l'ordre de grandeur 0,3 ou 1 pourra être choisi en fonction de l'imparité ou de la parité de l'exposant d'origine. Il est plus simple d'étudier le reste de la division par 2, égal à 0 lorsque l'exposant d'origine est impair et à 1 lorsqu'il est pair, puisque l'addition de 1 a renversé les parités.

Cette analyse a lieu sur la deuxième opération du point 8 qui est une CN. Cette CN est exploitée par une VC $\geqslant$ postérieure à une BO d'accès de A, quantité sous radical, en M_1. Or, la VC $\geqslant$ contrôle une AMD 3 de sorte que l'OB qui suit décalera ou ne décalera pas A suivant que l'exposant était impair ou pair.

B. *Calcul de* x_0.

Ce décalage permettra à l'AN du point 6 d'additionner 2 soit sur la 2e, soit sur la 3e position de A en partant de la gauche. Cette addition donne donc soit A + 0,2 soit A + 0,02, cette quantité étant retenue pour la valeur de x_0 ainsi différente selon la parité de l'exposant.

Ce cadrage correspond en quelque sorte à la division du nombre en tranches de deux chiffres du procédé manuel d'extraction.

Une deuxième CN est affichée au point 6, destinée à comparer à 1 la quantité x_0. En effet, si $x_0 > 1$, on prend A lui-même comme valeur approchée pour l'itération. Une VC $\geqslant$ permet alors d'omettre l'OB qui substituait, en M^9, x_0 à A. Donc la BO suivante donnera accès soit à x_0 soit à A.

C. *Les itérations.*

Elles ne présentent aucune difficulté et il ne paraît pas utile de les décrire : l'examen du développement donné en annexe suffit à se rendre compte de la consistance du programme. Le choix de x_0, tel qu'il a été décrit, permet d'être certain que 6 itérations sont suffisantes.

8. Étude du cas 6 : Point décimal flottant - PPC avec SP.

L'application de cette méthode a permis de lui donner une forme standard qui permet de ne se préoccuper que de coder des opérations simples d'accès dans les mémoires de travail, sur la base d'un tableau Gamma standard, d'un tableau BS — Γ également standard, et de dessin de cartes standard, valables universellement.

A. *Affectation des mémoires.*

Les mémoires du Calculateur Gamma sont affectées à des rôles bien précis (*fig.* 193) :

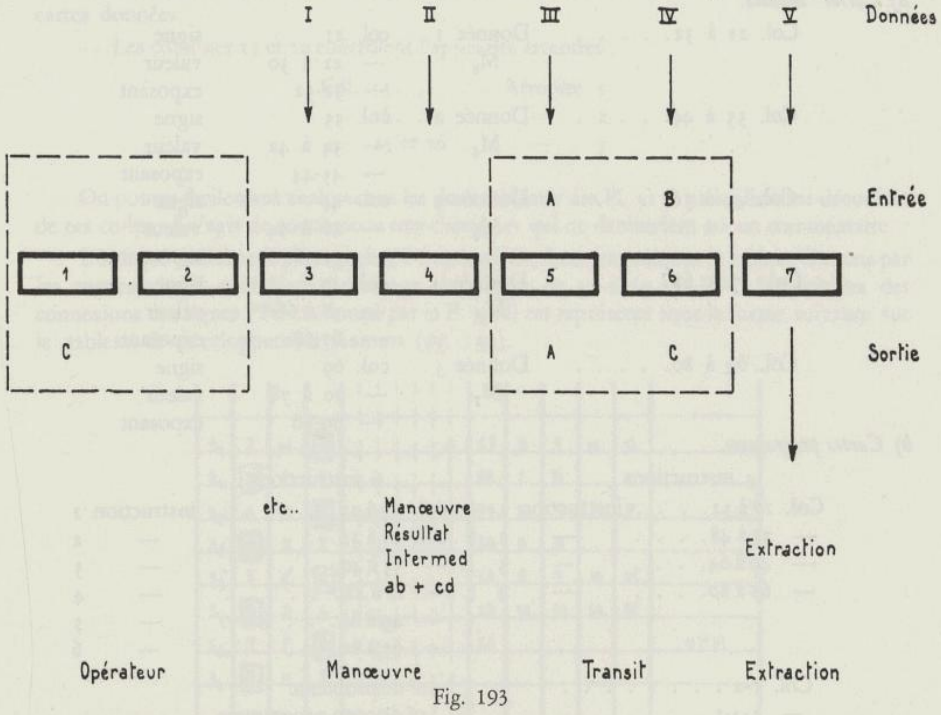


Fig. 193

— les mémoires 1 et 2 servent de mémoires opérateur,

— les mémoires 5 et 6 servent de mémoires données pour les sous-programmes,

— les mémoires 3 et 4 restent disponibles pour toutes manœuvres intermédiaires à coder sur les cartes programmes.

— la mémoire 7 sert à l'extraction des résultats,

— les mémoires supplémentaires servent à recevoir les données depuis les cartes, par l'intermédiaire des introducteurs et de la M_1 ainsi qu'à tous stokages intermédiaires à coder dans les cartes programme.

On devra donc, avant un sous-programme, positionner les données dans les mémoires 5 et 6. Chacune de ces introductions demandera une BO d'accès en M_1 et une OB de renvoi en M_5 ou M_6. D'où la nécessité de coder par ligne de carte plus de 4 opérations. En effet, il faudra ensuite afficher au moins la VS de renvoi au sous-programme intéressé devant conjuguer les deux nombres amenés en position de données. C'est chose facile puisque les OD et OF ne sont pas nécessaires pour compléter les AD des BO et OB qui traitent des nombres écrits sur 12 positions.

A la fin d'un sous-programme le résultat de l'opération se trouve à la fois contenu en M_1 et en M_6, ce qui permet les calculs en chaîne. On n'accède à la M_7 pour l'extraction que par une OB affichée sur les cartes programme.

B. *Dessin des cartes.*

A cette affectation systématique des mémoires correspond un dessin systématique des cartes :

a) Cartes données.

Col. 21 à 32.	Donnée 1	col. 21	signe
	M_3	— 22 à 30	valeur
		— 31-32	exposant
Col. 33 à 44.	Donnée 2	col. 33	signe
	M_4	— 34 à 42	valeur
		— 43-44	exposant
Col. 45 à 56.	Donnée 3	col. 45	signe
	M_5	— 46 à 54	valeur
		— 55-56	exposant
Col. 57 à 68.	Donnée 4	col. 57	signe
	M_6	— 58 à 66	valeur
		— 67-68	exposant
Col. 69 à 80.	Donnée 5	col. 69	signe
	M_7	— 70 à 78	valeur
		— 79-80	exposant

b) Cartes programme.

4 instructions		6 instructions	
Col. 17 à 32. . . .	instruction 1	(col. 17 à 24.	instruction 1
— 33 à 48. . . .	— 2	(col. 25 à 32.	— 2
— 49 à 64. . . .	— 3	(— 33 à 40.	— 3
— 65 à 80. . . .	— 4	(— 11 à 48.	— 4
		(— 49 à 64.	— 5
		(— 65 à 80.	— 6

Col. 1-2		n° de problème
— 3-4-5		n° de carte programme
— 6.	Perf.	9 = Extraction-Impression-Espace.
		8 = Introduction de donnée
		7 = Carte programme 4 opérations
		3 = Saut de papier

Afin d'accroître la puissance de stockage de la machine, on peut utiliser les totalisateurs de la tabulatrice, montés deux à deux en bascule. Les transferts dans les deux sens sont contrôlés par les colonnes 7, 8, 9 et 13 :

Colonne	7.	de M_3 à tot. 1 et 2 en IS
—	8.	de M_4 à tot. 3 et 4 en IS
—	9.	de M_5 à tot. 6 et 7 en IC
—	13.	de M_6 à tot. 8 et 9 en IC

Dans ces colonnes, la valeur de la perforation donne le sens du transfert, le rang du totalisateur et le contrôle de sa remise à zéro :

Perfor. 11 totalisateurs impairs
9 extractions du Gamma
8 introductions (statique ou cinématique) dans Gamma
0 R. A. Z. totalisateur.

— Les colonnes 10 à 12 portent le libellé.

— La colonne 14 est libre : elle peut porter un indice de séparation cartes programme-cartes données.

— Les colonnes 15 et 16 contrôlent l'appel des armoires :

Col.	15.	Armoire	1
—	16.	—	2
—	15 et 16 . .	—	3

On pourra facilement analyser sur les deux tableaux des Pl. 31 et 32 les liaisons découlant de ces codes : il s'agit de connexions très classiques qui ne demandent aucun commentaire.

En ce qui concerne le passage de 4 à 6 opérations, il est contrôlé sur le tableau Gamma par les commandes S contrôlant l'affichage des lignes de 57 à 63 du PPC. Un schéma des connexions des lignes PPC est donné par la P. 33. Il est représenté sous la forme suivante sur le tableau de développement Gamma (*fig.* 194).

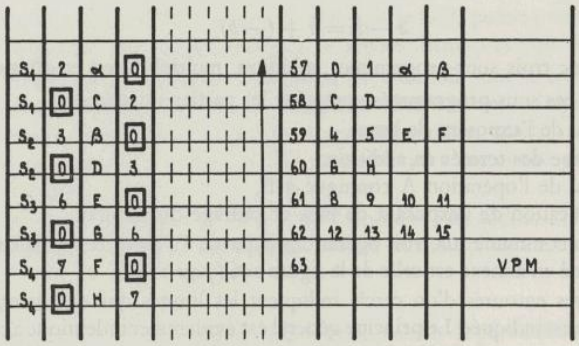


Fig. 194

Les codes « zéro » encadrés sont des zéros en provenance du distributeur de codes Gamma, alors que les autres codes non encadrés proviennent des plots du PPC.

Il résulte de cette sélection que selon le code de la carte des alternatifs sont au travail ou au repos et que la répartition sur les lignes est la suivante dans les deux cas (*fig.* 195).

	TO	AD	OD	OF		TO	AD	OD	OF
57	0	1	$\boxed{0}$	$\boxed{0}$	57	0	1	2	3
58	2	3	—	—	58	$\boxed{0}$	$\boxed{0}$	—	—
59	4	5	$\boxed{0}$	$\boxed{0}$	59	4	5	6	7
60	6	7	—	—	60	$\boxed{0}$	$\boxed{0}$	—	—
61	8	9	10	11	61	8	9	10	11
62	12	13	14	15	62	12	13	14	15
63		$\boxed{8}$	$\boxed{14}$	$\boxed{7}$	63		$\boxed{8}$	$\boxed{14}$	$\boxed{7}$

Fig. 195

C. *Étude des sous-programmes.*

Les sous-programmes standard qui seront exposés ici sont ceux d'addition, de soustraction, de multiplication et de division. Il était logique de ramener la soustraction à l'addition en écrivant :

$$a - b = a + (-b)$$

Il reste donc trois sous-programmes, addition, multiplication et division.

Chacun de ces sous-programmes comporte les parties suivantes :

α) Calcul de l'exposant de base.
β) Cadrage des termes en addition,
γ) Calcul de l'opération A conjugué à B,
δ) Rectification de l'exposant de base et cadrage du résultat.

On a rendu commune aux trois opérations la partie δ), de sorte que la structure du développement donné en annexe est celle de la figure 196 (*page* 179).

Les nombres entourés d'un cercle indiquent les limites qui définissent les séquences dont la fonction est indiquée. Le principe général est évidemment identique à celui indiqué plus haut lors du programme relatif à la matrice : on a fait figurer sur le développement de la Pl. 34 les rôles des diverses lignes les plus importantes.

Toutefois, certaines remarques doivent être faites concernant les obligations qu'amenait la communauté du programme de cadrage et de rectification de l'exposant.

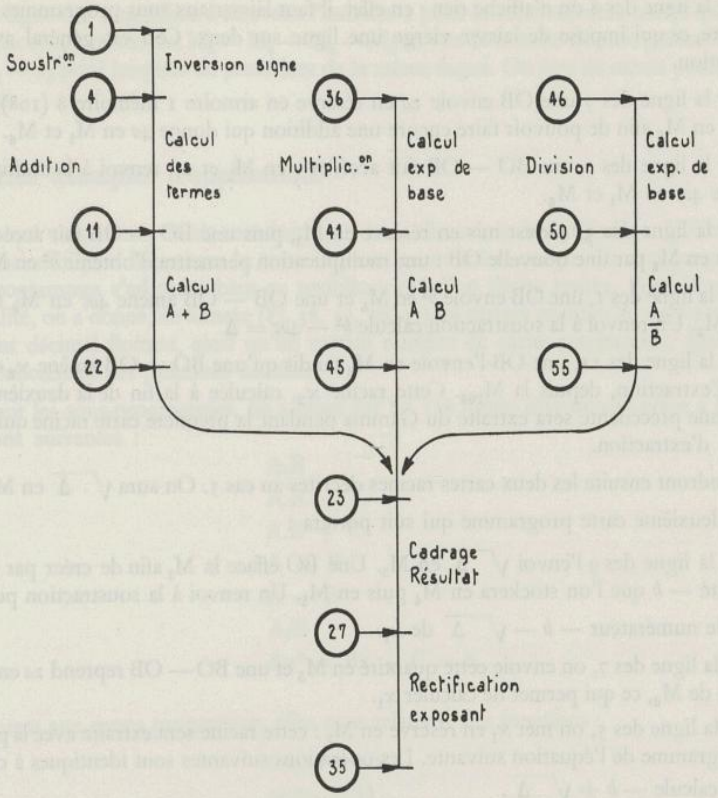


Fig. 196

En premier lieu, on peut constater que, à la fin de chaque calcul, la virgule des lignes 22, 45 et 54 est alignée en position 9. Il convient, en effet, que la DCC puisse prendre, en vue de la rectification de l'exposant, des nombres alignés de façons identiques pour les 3 opérations.

En second lieu la CN de la ligne 5, nécessaire en addition pour le choix du plus petit nombre, est exploitée par les VC incluses dans la partie commune de cadrage et rectification. Il était donc nécessaire de faire des forçages du comparateur en multiplication et division pour pouvoir se servir de la séquence commune. Ces forçages, qui n'ont aucun sens arithmétique, sont affichés sur les lignes 37 et 51.

Enfin, la BO de la ligne 27 amène en M_1 soit l'exposant *b* en addition (auquel on substituera éventuellement *a*), soit l'exposant de base calculé en multiplication et division.

D. *Application.*

On a pris l'équation du second degré comme exemple afin de permettre une comparaison avec la même étude en point décimal fixe (Pl. 35).

Les données *c b a* sont introduites dans les mémoires 3, 4, 5, lors de la lecture de la carte donnée. La première carte programme porte sur la ligne des 9 une BO — OB d'accès de *a* en M_6 et un renvoi en sous-programme d'addition : on a ainsi 2*a* en M_1 et M_6.

Sur la ligne des 8 on n'affiche rien : en effet, il faut laisser aux sous-programmes le temps de se faire, ce qui impose de laisser vierge une ligne sur deux. Ceci est général avec cette organisation.

Sur la ligne des 7 une OB envoie $2a$ en réserve en armoire 1 mémoire 8 (108), et une seconde en M_5 afin de pouvoir faire encore une addition qui donne $4a$ en M_1 et M_6.

Sur la ligne des 5 une BO — OB fait accéder c en M_5 et un renvoi à la multiplication constitue $4ac$ en M_1 et M_6.

Sur la ligne des 3, $4ac$ est mis en réserve en M_3, puis une BO — OB fait accéder b en M_5, puis en M_6 par une nouvelle OB : une multiplication permettra d'obtenir b^2 en M_1 et M_6.

Sur la ligne des 1, une OB envoie b^2 en M_5 et une OB — OB amène $4ac$ en M_6 depuis la réserve M_3. Un renvoi à la soustraction calcule $b^2 - 4ac = \Delta$

Sur la ligne des 11, une OB l'envoie en M_5 tandis qu'une BO — OB amène x_2 en M_7 en vue de l'extraction, depuis la M_{109}. Cette racine x_2, calculée à la fin de la deuxième carte-programme précédente sera extraite du Gamma pendant la première carte racine qui portera un code d'extraction.

Viendront ensuite les deux cartes racines décrites au cas 5. On aura $\sqrt{\Delta}$ en M_1 et M_6.

La deuxième carte programme qui suit portera :

Sur la ligne des 9 l'envoi $\sqrt{\Delta}$ en M_3. Une BO efface la M_1 afin de créer par une SN la quantité $-b$ que l'on stockera en M_4 puis en M_5. Un renvoi à la soustraction permet de calculer le numérateur $-b - \sqrt{\Delta}$ de x_1.

Sur la ligne des 7, on envoie cette quantité en M_5 et une BO — OB reprend $2a$ en M_{108} en direction de M_6, ce qui permet de calculer x_1.

Sur la ligne des 5, on met x_1 en réserve en M_7 : cette racine sera extraite avec la première carte programme de l'équation suivante. Les opérations suivantes sont identiques à ceci près que l'on calcule $-b + \sqrt{\Delta}$.

Sur la ligne des 1 sont affichées les ZB des mémoires qui recevront a b c de la carte-donnée suivante.

E. *Introduction des données.*

La nécessité de passer par les introducteurs du Gamma et de prendre comme mémoire de transit la M_1 impose d'établir des cartes-programme d'introduction des données.

Soit à introduire les données suivantes :

$$A - B - C - D, a_1 - a_2 - a_3 - a_4 - a_5, b_1 - b_2 - b_3 - b_4 - b_5, c_1 - c_2 - c_3, d_1 - d_2.$$

On introduira les termes de la forme a en armoire 1, ceux de la forme b en armoire 2, ceux de la forme c et d en armoire 3.

Le développement donné Pl 36 donne le programme.

Sur une première carte-donnée figureront A B C D qui entreront en $M_3 - M_4 - M_5$ M_6. Sur la ligne des 9, on appellera l'armoire 1. Sur la ligne des 8 une BO — OB enverra A en M_{108}. Sur la ligne des 7, on appellera l'armoire 2. Sur la ligne des 6, on enverra B en M_{208}. De même pour C et D en M_{308} et M_{309}.

La ligne des 3 appellera à nouveau l'armoire 1 pour la carte suivante tandis que la ligne des 2 effacera les mémoires $M_3 - M_4 - M_5 - M_6$ en vue de la carte-donnée portant $a_1 - a_2 - a_3 - a_4 - a_5$ pour lesquels on procédera de la même façon. On fera de même pour les autres données.

9. Autres exemples d'application.

Ce qui vient d'être décrit constitue les thèmes directeurs de l'utilisation du Gamma 3 en calculs scientifiques. Bien entendu sur ces thèmes peuvent être conçus d'autres programmes et sous-programmes s'adaptant bien au problème que l'on désire traiter. Pour illustrer cette possibilité, on a donné, en annexe (Pl. 38, 39, 40, 41) un certain nombre de sous-programmes en point décimal flottant, ainsi qu'un certain nombre de cartes-programme, également en PDF, concernant des fonctions usuelles.

Pour les sous-programmes, on a placé sur les 128 lignes disponibles grâce à l'AEP, les fonctions suivantes :

A.B
A.B. — C
A.B + C
A + B
A — B
A/B
$Ax^2 + Bx + C = 0$.

Quant aux cartes programme, elles sont relatives aux fonctions :

sin. et cos x
arctg N/D

	AVANT L'OPÉRATION							APRÈS L'OPÉRATION				
Opération	Durée opération	Variante	M_1	M_4	M_5	M_6	M_7	M_1	M_4	M_5	M_6	M_7
AB	1 pt	0 0 2 5			A	B		AB	inchangée	A	AB	inchangée
AB — C	2 pt	0 0 2 9			A	B	C	AB — C	inchangée	A	AB	AB — C
AB + C	2 pt	0 0 2 13			A	B	C	AB + C	inchangée	A	AB	AB + C
A + B	1 pt	0 0 7 13				A	B	A + B	inchangée	inchangée	A	A + B
A — B	1 pt	0 0 7 1				A	B	A — B	inchangée	inchangée	A	A — B
A / B	1 pt	0 0 15 13			x	A	B	A / B	inchangée	inchangée	A	A / B
$Ax_2 + Bx + C$ (CD av. 4 col. 6)	4 pt	0 0 2 13		C		A	B	$Ax^2 + Bx + C$	0 chang.	x inchangée	0 sin. x	0 cos. x
sin. et cos. x ..	4 cartes	en ligne 11			inchangée	x						
arctg N/D	4 cartes				inchangée	N	D	arctg	chang.	inchangée	chang.	arctg

N.B. — Les sous-programmes Sin. cos. x et arctg N/D altèrent la mémoire 3.

Fig. 197

Le nombre de facteurs rentrant en jeu dans ces programmes imposent un nombre de BO — OB d'accès plus grand que deux, suffisant quand il n'est conjugué que deux facteurs. D'où la nécessité, selon les formules, de travailler en 4, 6 ou 8 opérations. En effet, 3 facteurs réclament trois séries de BO — OB d'accès. Chacune de ces opérations, sauf la variante de renvoi ne réclamera l'affichage que du TO et de l'AD.

Les tableaux de développement mis en annexe permettront de suivre les programmes. Les mêmes principes que ceux précédemment exposés y sont mis en œuvre : il serait sans intérêt de les répéter.

Toutefois, pour la facilité d'utilisation, le tableau de la figure 197, *page* 181, permettra de suivre l'état des mémoires, de se rendre compte des durées d'opérations à prévoir, et de coder les variantes de renvoi.

10. Mise au point en programme par cartes.

On n'utilise pas le dispositif « pas à pas ». Il est, en effet, beaucoup plus simple d'utiliser une carte-programme dont la fonction sera de provoquer l'extraction de toutes les mémoires derrière chaque carte-programme ou derrière des groupes de cartes-programme.

On effectue les calculs de vérification à la main suivant le même processus que le Gamma.

S'il y a erreur, on pourra redécouper le programme jusqu'au stade où on détectera les instructions erronées.

11. Reproduction des cartes-programme (Pl. 37).

A. *Problème.*

La duplication des cartes-programme n'est pas réalisable selon les procédés courants, vu le grand nombre de perforations pouvant figurer sur une carte. Il est nécessaire de procéder en plusieurs passages afin de ne pas soumettre le bloc poinçons-matrice à des efforts trop considérables.

La réalisation du problème nécessite l'implantation, sur le carter du tableau, d'un tableau auxiliaire.

B. *Méthode.*

L'exploration de la carte est réalisée en deux phases de trois passages chacune, soit 6 passages pour la duplication, plus un septième passage de contrôle.

a) *Duplication.*

1re *phase* — Lecture et duplication des colonnes d'ordre impair :

1er passage : duplication des lignes 11-2-5-8.
2e — — — 12-1-4-7.
3e — : — — 0-3-6-9.

2e *phase*. Lecture et duplication des colonnes d'ordre pair, les lignes reproduites aux 4e, 5e et 6e passages sont respectivement identiques à celles des 1er, 2e et 3e passages.

b) *Contrôle*. Un seul passage permet de contrôler l'ensemble des perforations.

C. *Connexions.*

a) *Lecture-perforation.* La lecture des colonnes de rang pair, d'une part, et impair d'autre part, est validée par deux groupes d'alternatifs (1 à 13 et 14 à 27) représentant les deux phases. *Remarque :* l'envoi au mag. de la col. 80 est validé par C et S de l'alternatif 14 contrôlé lors des passages impairs.

L'inverseur 3 autorise l'alimentation par deux sélections, appelées systématiquement en parallèle, des commandes de l'un ou l'autre des deux groupes d'alternatifs.

Le courant rupté alimentant le cylindre de BLL est ventilé par les deux inverseurs 1 et 2 vers trois groupes de plots du sélecteur (11, 2, 5 et 8 — 12, 1, 4, et 7 — 0, 3, 6 et 9). Le plot commun du sélecteur permettant alors l'alimentation du cylindre par connexion avec le plot récepteur du cavalier BLL. Cette connexion est réalisée à travers le tableau auxiliaire, cavalier en position « reproduction ».

b) *Contrôle.* Les colonnes de BLP sont comparées aux 80 col. de BCL; le contrôle est validé à travers le tableau auxiliaire, cavalier en position « contrôle ».

D. *Manipulation.*

L'inverseur 3 commande les phases :
— en haut : 1re phase,
— en bas : 2e phase.

Les inverseurs 1 et 2 commandent les passages au cours de chaque phase :
— 1 en haut : 1er et 4e passages,
— 2 en haut : —
— 1 en bas : 2e et 5e passages,
— 2 en haut : —
— 1 en bas : 3e et 6e passages,
— 2 en bas : —

La position du cavalier sur le tableau auxiliaire spécialise les passages « reproduction » et le passage « contrôle ».

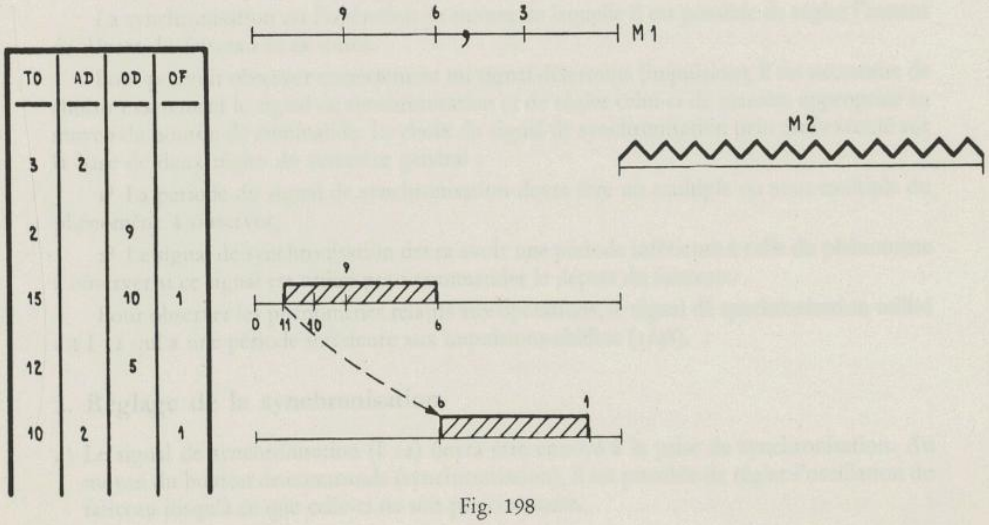


Fig. 198

12. Point décimal flottant réduit.

Chaque mémoire est découpée en deux : dans chaque moitié, on place l'exposant sur une position et 5 chiffres significatifs. Dans ces conditions, il est évident que le programme PPC devient :

	TO	AD	OD	OF		
57	—	—	—	—	—	PPC
58	—	—	—	—	—	PPC
59	8	5	—	—	—	OB
60	—	—	—	—	—	PPC
61	8	6		—	—	OB
62	—	—	—	—	—	PPC
63	VPM	Pt.3.				

Il faudra donc 2 lignes pour la séquence OB — BO — OB — BO — OB — VS.

On voit que le signe ne peut être traité.

On peut passer du point décimal fixe au point décimal flottant réduit :

Pour cela, on amène la virgule du nombre en M_1 sur la position 5 et on efface M_2 (*fig.* 198).

Une DCC précédée d'une AMD 9 permet de placer en position 10 le premier chiffre significatif et une MR permet de décaler de 5 positions le nombre ainsi cadré. Puis on transfère par AN le contenu de la mémoire décalage se trouvant en M_2.

Chapitre onzième

UTILISATION DE L'OSCILLOSCOPE

Le Calculateur Gamma est pourvu de certains circuits destinés à rendre simple le contrôle des connexions affichées au tableau de connexion. Grâce à ces circuits, il est possible d'explorer le contenu de toutes les mémoires au moyen d'un oscilloscope. Ainsi, peuvent être vérifiés non seulement le contenu des mémoires de nombre mais encore le contenu des mémoires de programme.

MANŒUVRE DE L'OSCILLOSCOPE

1. Principe de la synchronisation.

La synchronisation est l'opération au moyen de laquelle il est possible de régler l'instant de départ du faisceau et sa durée.

Pour pouvoir observer correctement un signal déterminé (impulsion), il est nécessaire de choisir exactement le signal de synchronisation et de régler celui-ci de manière appropriée au moyen du bouton de commande. Le choix du signal de synchronisation peut être exécuté sur la base de deux règles de caractère général :

1⁰ La période du signal de synchronisation devra être un multiple ou sous-multiple du phénomène à observer;

2⁰ Le signal de synchronisation devra avoir une période inférieure à celle du phénomène à observer si ce signal est utilisé pour commander le départ du faisceau.

Pour observer les phénomènes relatifs aux opérations, le signal de synchronisation utilisé est I 12 qui a une période inférieure aux impulsions-chiffres (1/48).

2. Réglage de la synchronisation.

a) Le signal de synchronisation (I 12) devra être envoyé à la prise de synchronisation. Au moyen du bouton de commande (synchronisation), il est possible de régler l'oscillation du faisceau jusqu'à ce que celle-ci ne soit pas constante.

b) Au moyen du commutateur de droite, il est possible de choisir la position qui sélectionne la gamme de fréquences.

c) Le bouton de commande placé sous le commutateur permet d'obtenir la plus grande image compatible avec le nombre de spots à observer.

3. Luminosité et focalisation.

Le réglage est exécuté au moyen des deux boutons de commande situés sur la droite de l'écran. Le premier bouton commande la mise sous tension de l'oscillographe. Il est à conseiller de maintenir à une force moyenne la luminosité afin de ne pas détériorer la surface fluorescente de l'écran.

4. Subdivision du faisceau (marquage).

Le choix du signal de subdivision du faisceau s'obtient en fonction des images qui devront être observées. Au cas où le phénomène à observer est représenté par des impulsions-chiffres, le signal est I 1. Ce signal devra être envoyé à la prise de l'oscillographe marquée « marquage ».

L'extinction ou l'atténuation du signal de subdivision est obtenue au moyen du bouton de commande situé au-dessus de la prise « marquage ».

5. Encadrement latéral.

L'encadrement (la déflexion) de l'image dans le sens horizontal s'obtient en manœuvrant le bouton de commande central situé entre les deux commutateurs.

6. Augmentation de l'ampleur (étendue) des images.

L'augmentation verticale de l'étendue des images s'obtient en manœuvrant le premier bouton de commande situé à la gauche de l'écran.

7. Mesure des tensions et de l'étendue des signaux.

Le commutateur de gauche qui porte une aiguille mobile longue et un cadran gradué de + 10 à — 20 volts, permet de mesurer les tensions comprises entre ces valeurs en utilisant comme référence le fil métallique qui traverse horizontalement l'écran. L'oscillographe devra être réglé (taré) préalablement de manière qu'en mettant le commutateur en position 0, le faisceau mis à la masse coïncide avec le fil métallique de référence.

Table des Matières

Chapitre premier

GÉNÉRALITÉS

Chapitre deuxième

LES CIRCUITS GÉNÉRAUX DE LIAISON

Chapitre troisième

L'INTRODUCTION DES DONNÉES

Chapitre quatrième

L'EXTRACTION DES DONNÉES

Chapitre cinquième

DESCRIPTION DU CALCULATEUR GAMMA 3

Chapitre sixième

LES OPÉRATIONS ÉLÉMENTAIRES DU CALCULATEUR GAMMA 3

Chapitre septième

CALCULATEUR GAMMA 3 M

Chapitre huitième

CALCULATEUR GAMMA 3 B

Chapitre neuvième

LES FONCTIONS PRINCIPALES

Chapitre dixième

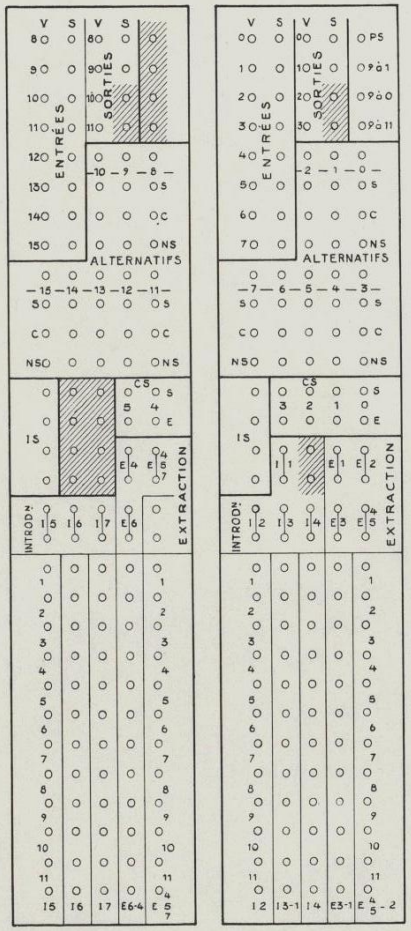
ÉTUDE DES PROBLÈMES

Chapitre onzième

UTILISATION DE L'OSCILLOSCOPE

N° 836 Imprimé en France. — GOTSCHO — Paris

GAMMA 3



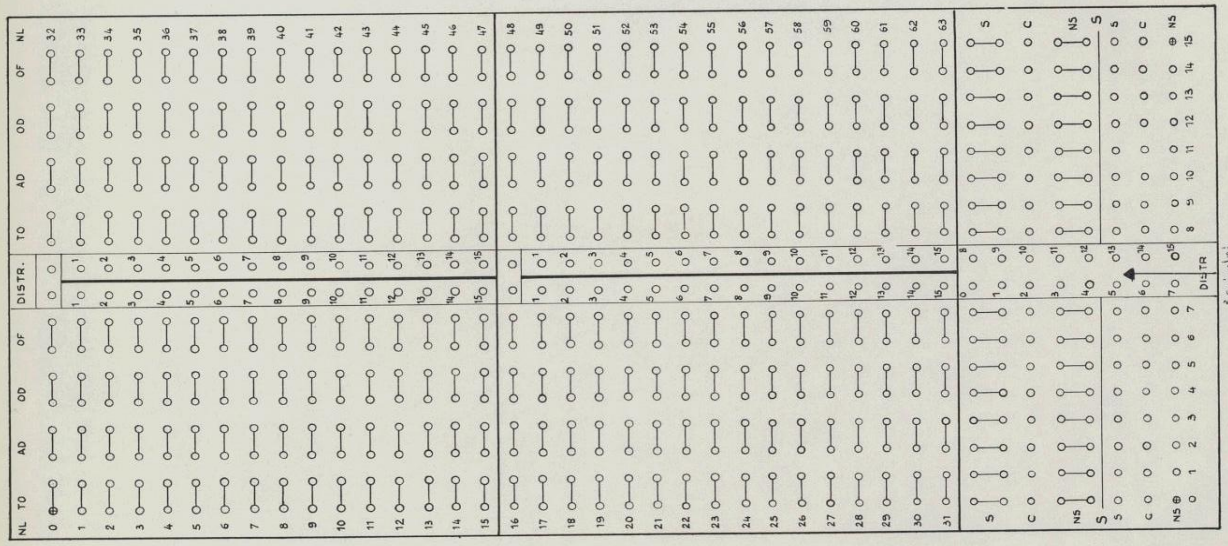


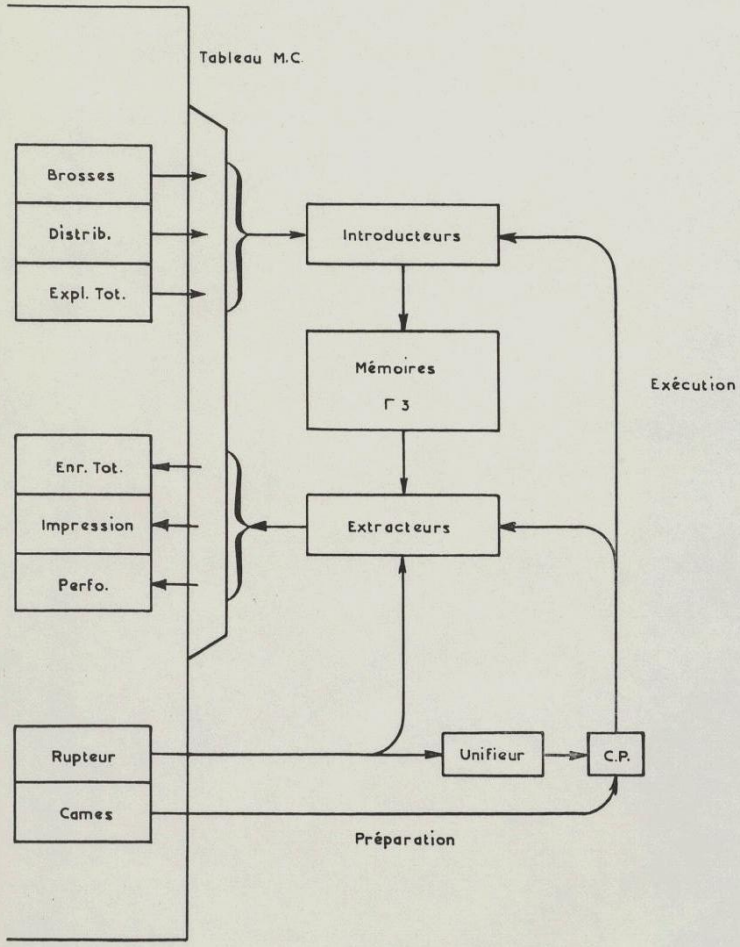
TABLEAU DE CODE GAMMA

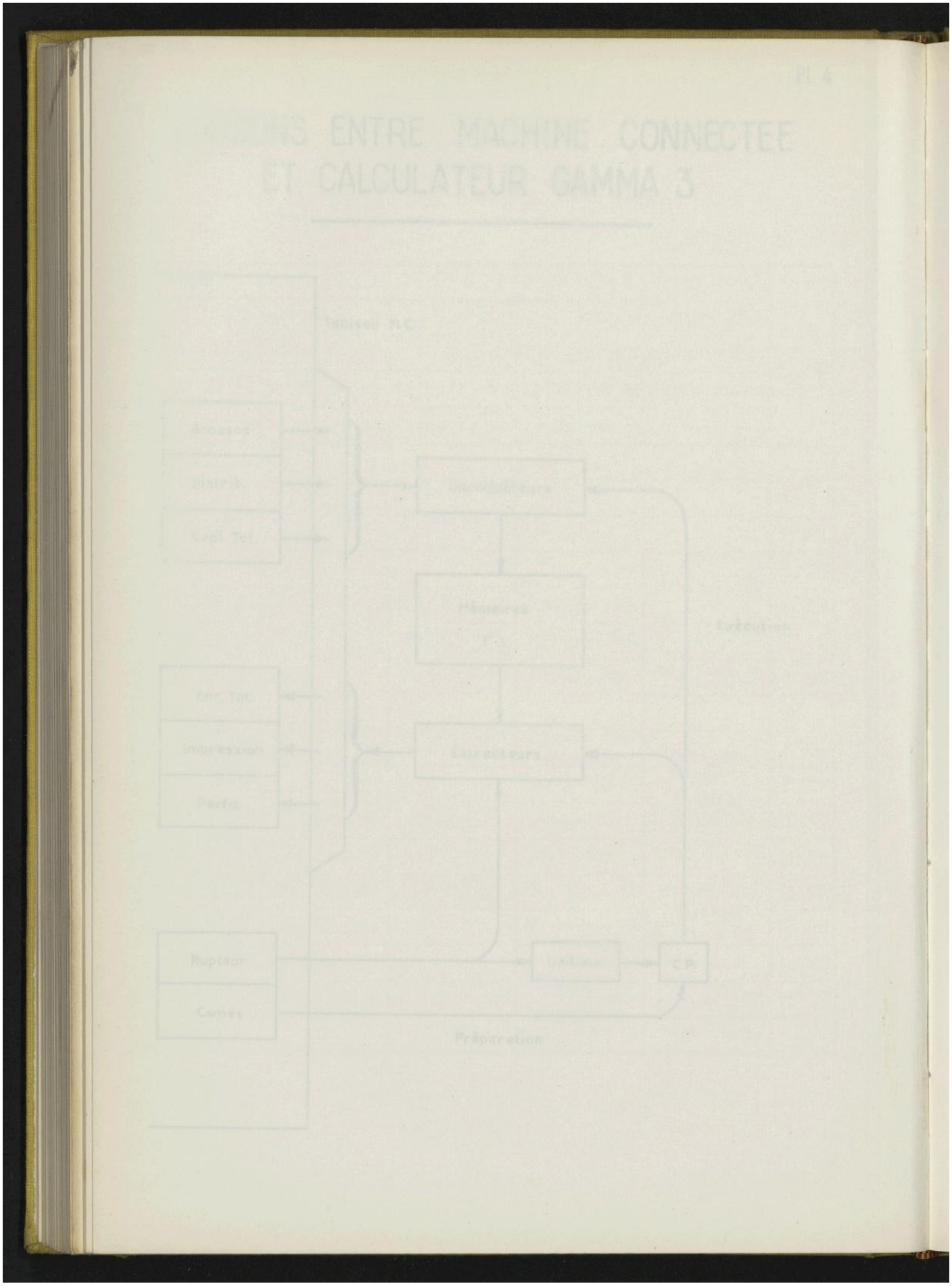
OF ↘	AD →				0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	← AD	
OD ↓	0	4	8	12	Jamais	>	=	≥							1,14	0,14	11,14	12,14	13,14	14,14	0	V
	1	5	9	13	Toujours	≤	≠	<					Pt, 3		0,3	11,3	12,3	13,3	14,3			
	2	6	10	14	V0	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15		
	3	7	11	15	$\overline{V}_0$	$\overline{V}_1$	$\overline{V}_2$	$\overline{V}_3$	$\overline{V}_4$	$\overline{V}_5$	$\overline{V}_6$	$\overline{V}_7$	$\overline{V}_8$	$\overline{V}_9$	$\overline{V}_{10}$	$\overline{V}_{11}$	$\overline{V}_{12}$	$\overline{V}_{13}$	$\overline{V}_{14}$	$\overline{V}_{15}$		
0	0	1	2	3	Condition de transfert en NL																	
1	4	5	6	7																	1	
2	8	9	10	11																	2	AMD
3	12	13	14	15					R à Z Filtrée				R à Z Filtrée				Maintien MD				3	ZB
4	16	17	18	19						OF → M1					OF → MB						4	KB
5	20	21	22	23					M1	Intr. → M1 OD = MD			MB		Intr. → MB						5	IS
6	24	25	26	27	OF → M1				Maintien f. M1				MB → M1				R à Z Totale M1 OD → MD				6	BO
7	28	29	30	31																	7	
8	32	33	34	35					R à Z f. M1				R à Z f. MB M1 → MB				Opérations avec cadrage préalable → MD = OD				8	OB
9	36	37	38	39	Suppression	M1 ≷ OF			M1 t. ≷ M1 f.				M1 ≷ MB								9	CN
10	40	41	42	43	Cadrage préalable — Maintien MD	M1 + OF			M1 x 2				M1 + MB								10	AN
11	44	45	46	47		M1 − OF			R à Z f. M1				M1 − MB								11	SN
12	48	49	50	51	M1 x OF = M1								M1 x MB = M1				Nombre de décalages de M1 = OD				12	MR
13	52	53	54	55	M1 ÷ OF = M1								M1 ÷ MB = M1								13	DR
14	56	57	58	59	M2 x OF = M1M2										M2 x MB = M1M2		Nombre de décalages de M1M2 = MD				14	MC
15	60	61	62	63	M1M2 ÷ OF = M2										M1M2 ÷ MB = M2						15	DC
OD ↑	Numéro de Ligne NL				0				1				2		3 à 15		← AD				TO ↑	

TABLEAU DE CODE GAMMA

OF ↘	AD →				0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	← AD	
OD ↓	0	4	8	12	Jamais	>	=	⩾							1,14	0,14	11,14	12,14	13,14	14,14	0	V
	1	5	9	13	Toujours	⩽	≠	<					Pt,3		0,3	11,3	12,3	13,3	14,3			
	2	6	10	14	V_0	V_1	V_2	V_3	V_4	V_5	V_6	V_7	V_8	V_9	V_{10}	V_{11}	V_{12}	V_{13}	V_{14}	V_{15}		
	3	7	11	15	$\bar{V}_0$	$\bar{V}_1$	$\bar{V}_2$	$\bar{V}_3$	$\bar{V}_4$	$\bar{V}_5$	$\bar{V}_6$	$\bar{V}_7$	$\bar{V}_8$	$\bar{V}_9$	$\bar{V}_{10}$	$\bar{V}_{11}$	$\bar{V}_{12}$	$\bar{V}_{13}$	$\bar{V}_{14}$	$\bar{V}_{15}$		
0	0	1	2	3	Condition de transfert en NL																	
1	4	5	6	7																	1	
2	8	9	10	11																	2	AMD
3	12	13	14	15					R à Z Filtrée				R à Z Filtrée				Maintien MD				3	ZB
4	16	17	18	19						OF → M1				OF → MB							4	KB
5	20	21	22	23					M1	Intr.r → M1 OD = MD			MB	Intr.r → MB							5	IS
6	24	25	26	27	OF → M1				Maintien f. M1				MB → M1				R à Z Totale M1 OD → MD				6	BO
7	28	29	30	31																	7	
8	32	33	34	35					R à Z f. M1				R à Z f. MB M1 → MB				Opérations avec cadrage préalable → MD = OD				8	OB
9	36	37	38	39	Suppression cadrage préalable Maintien MD		M1 ⩾ OF		M1 t. ⩾ M1 f.				M1 ⩾ MB								9	CN
10	40	41	42	43			M1 + OF		M1 x 2				M1 + MB								10	AN
11	44	45	46	47			M1 − OF		R à Z f. M1				M1 − MB								11	SN
12	48	49	50	51	M1 x OF = M1								M1 x MB = M1				Nombre de décalages de M1 = OD				12	MR
13	52	53	54	55	M1 ÷ OF = M1								M1 ÷ MB = M1								13	DR
14	56	57	58	59	M2 x OF = M1M2										M2 x MB = M1M2		Nombre de décalages de M1M2 = MD				14	MC
15	60	61	62	63	M1M2 ÷ OF = M2										M1M2 ÷ MB = M2						15	DC
OD	Numéro N'L de Ligne				0				1				2		3 à 15		← AD				TO	

LIAISONS ENTRE MACHINE CONNECTEE ET CALCULATEUR GAMMA 3





SYNCHRONISATION MACHINE CONNECTEE GAMMA 3

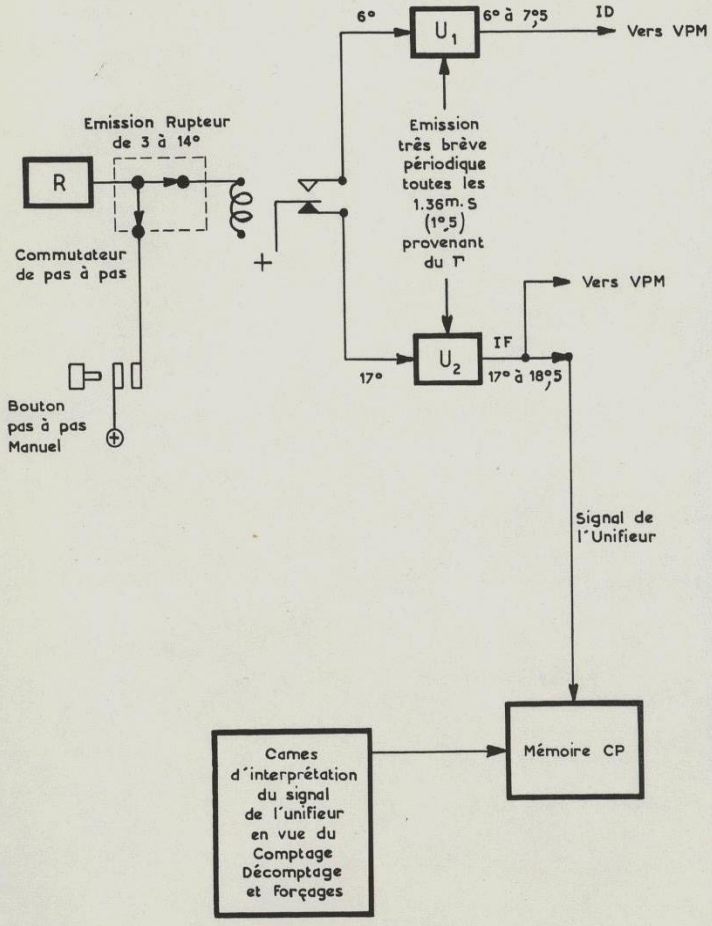
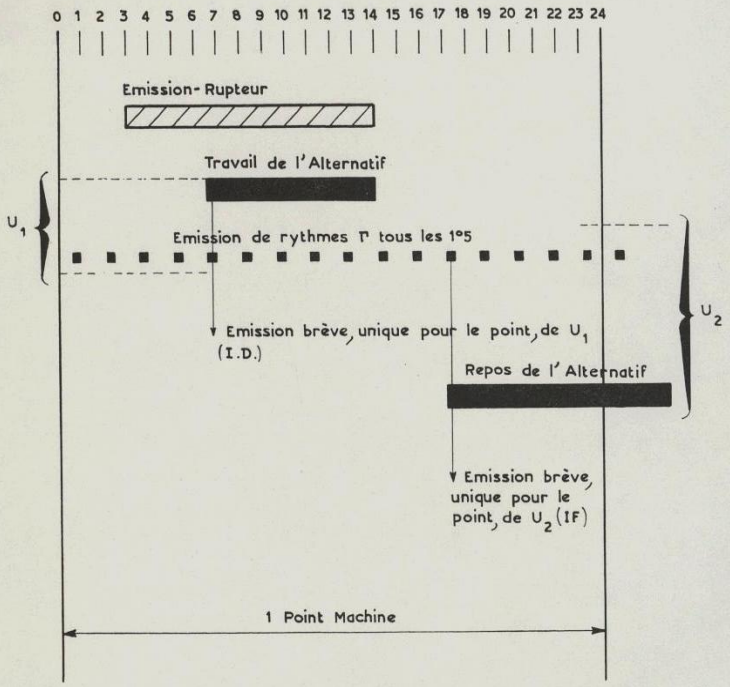
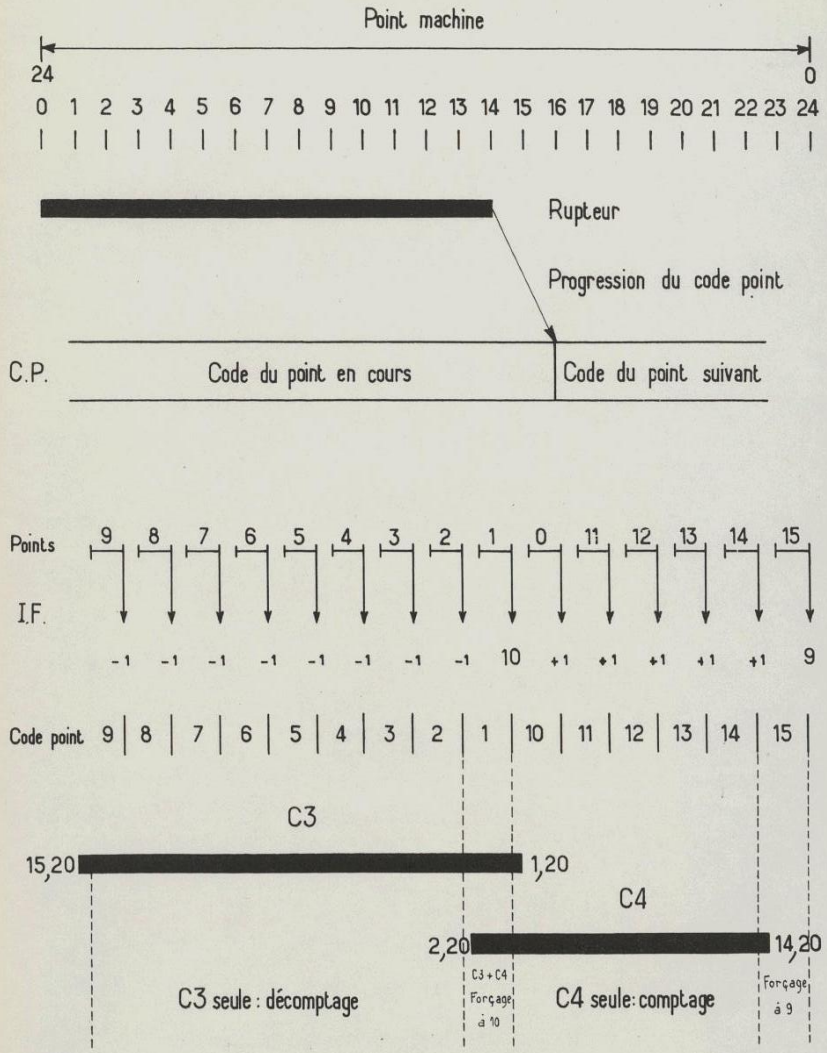


DIAGRAMME RUPTEUR UNIFIEUR



Γ 3 CODE POINT



Γ3 _ INTRODUCTION CINEMATIQUE

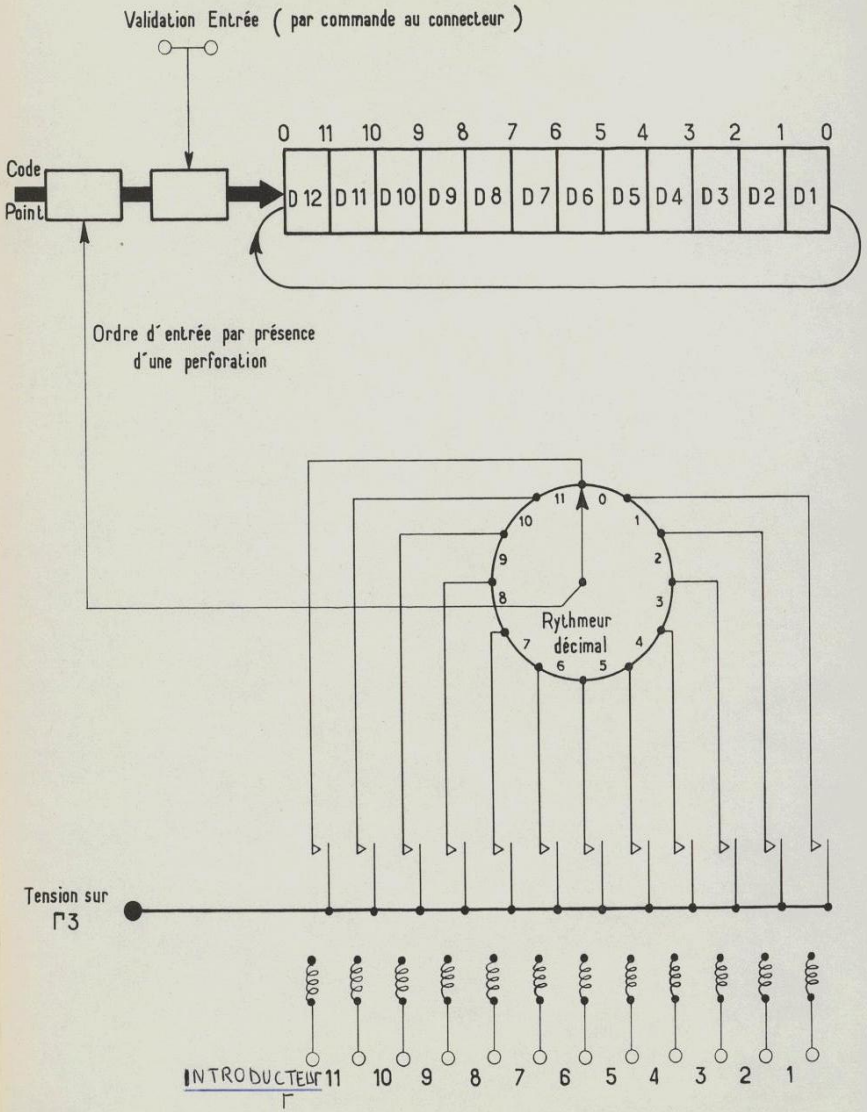
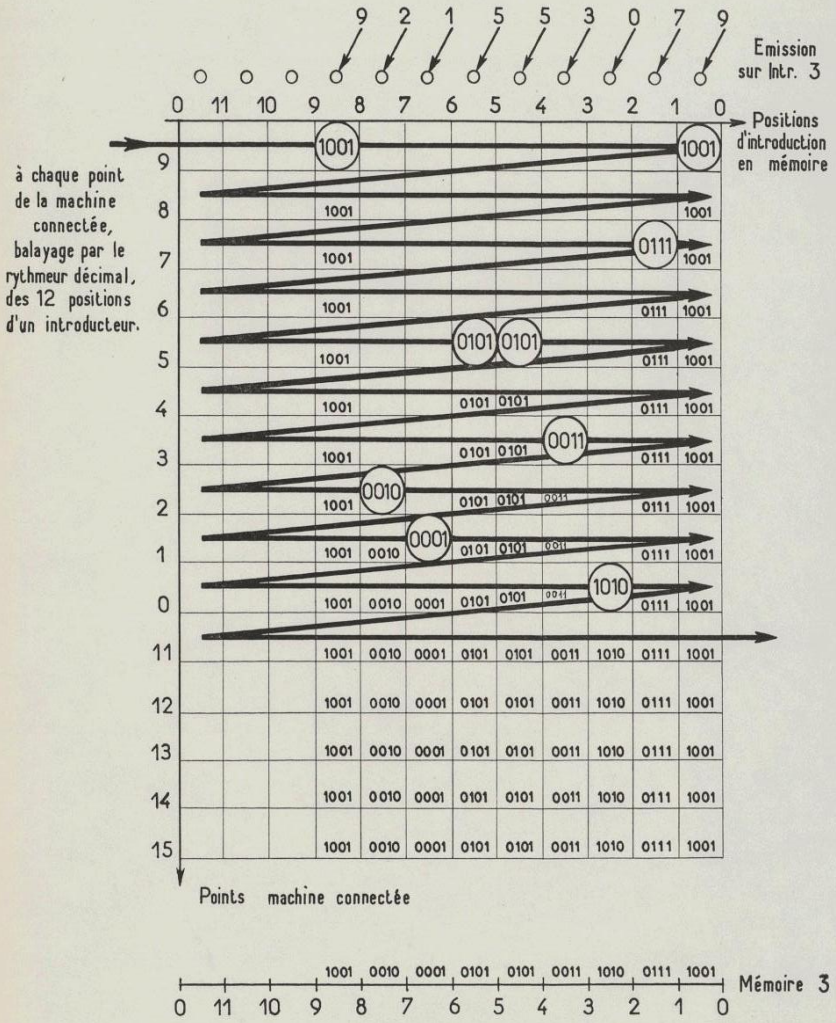
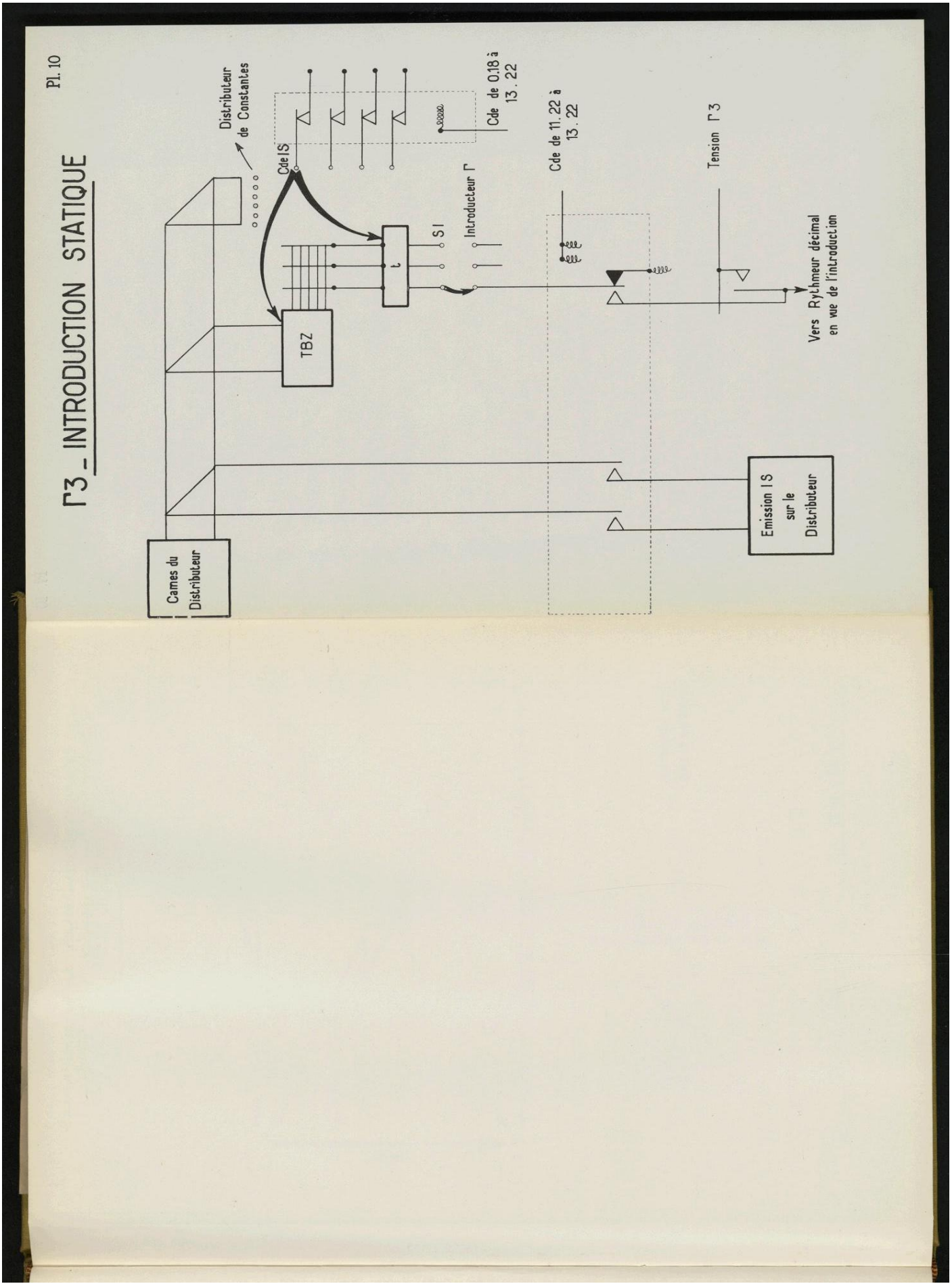
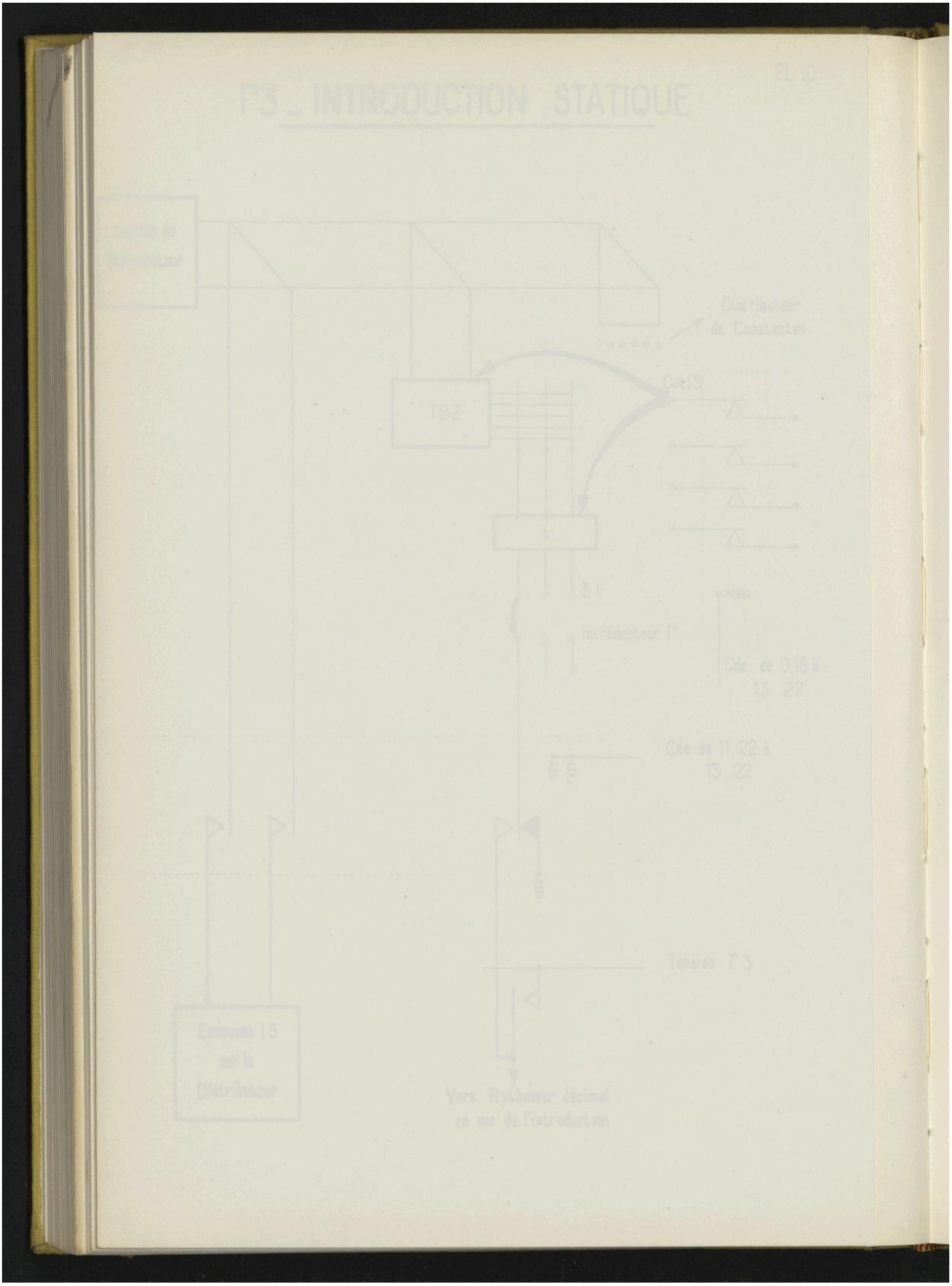


DIAGRAMME DE L'INTRODUCTION CINEMATIQUE

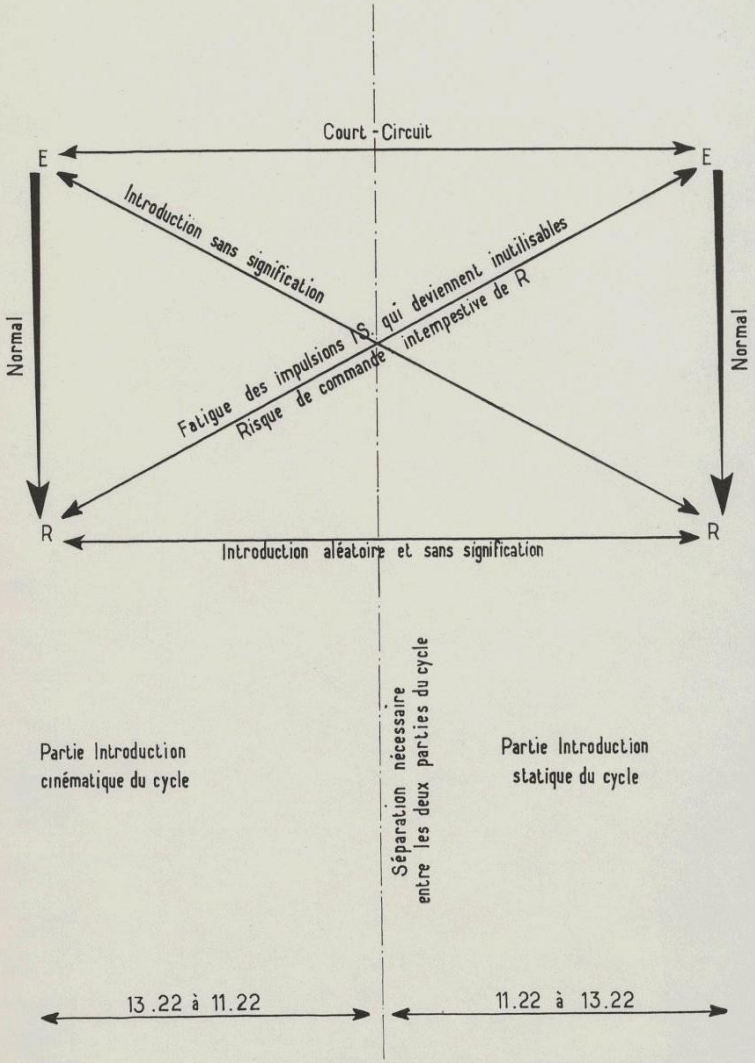


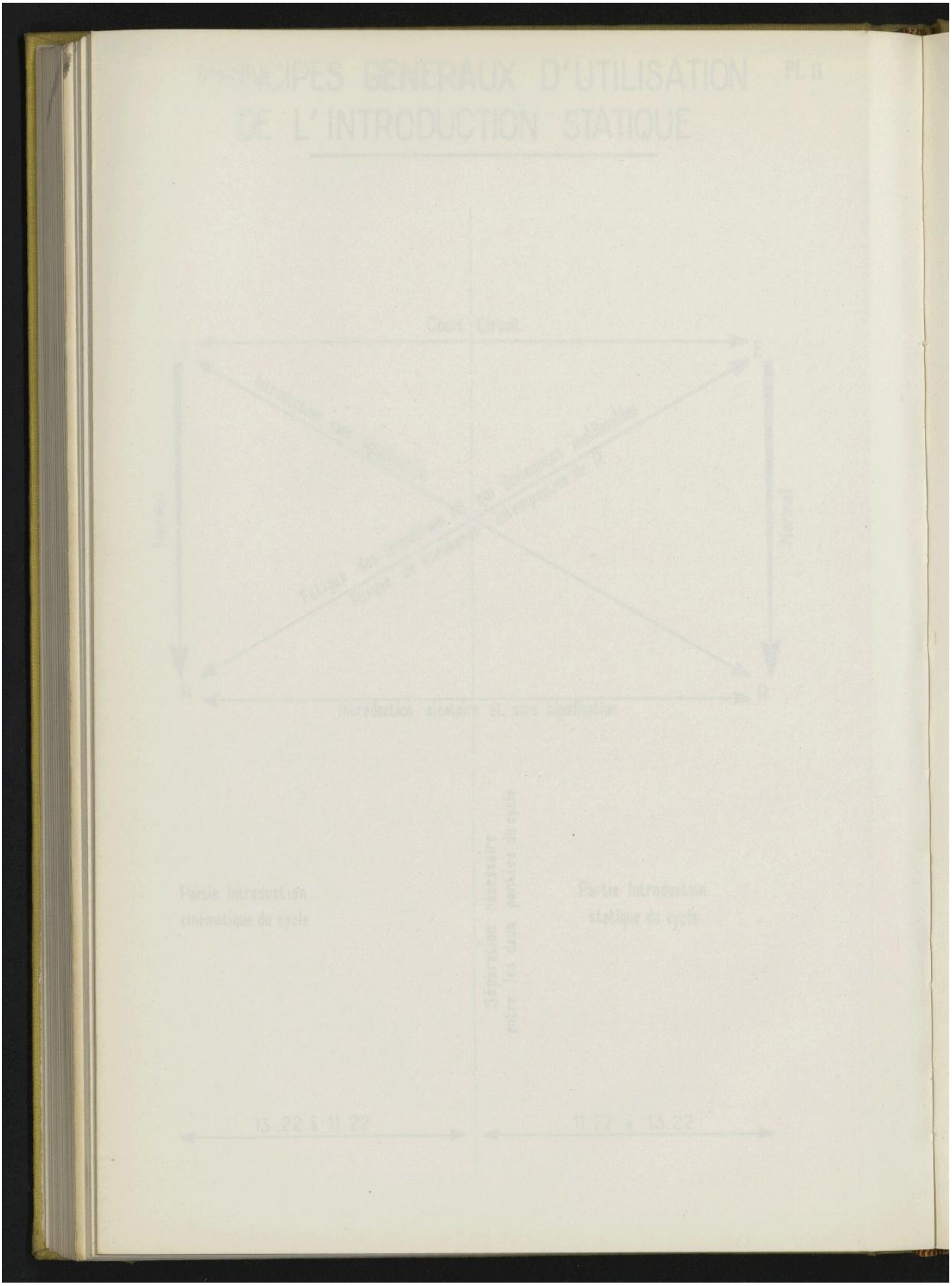




PRINCIPES GENERAUX D'UTILISATION DE L'INTRODUCTION STATIQUE

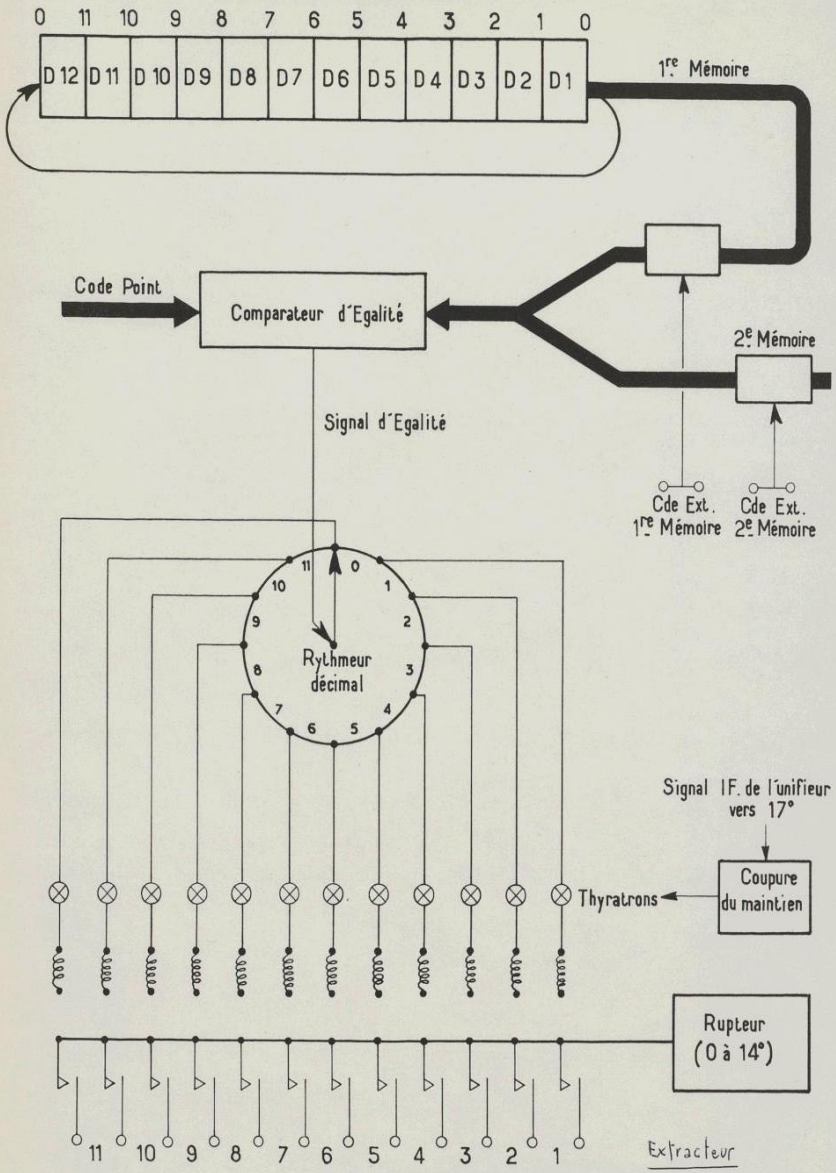
Pl. 11

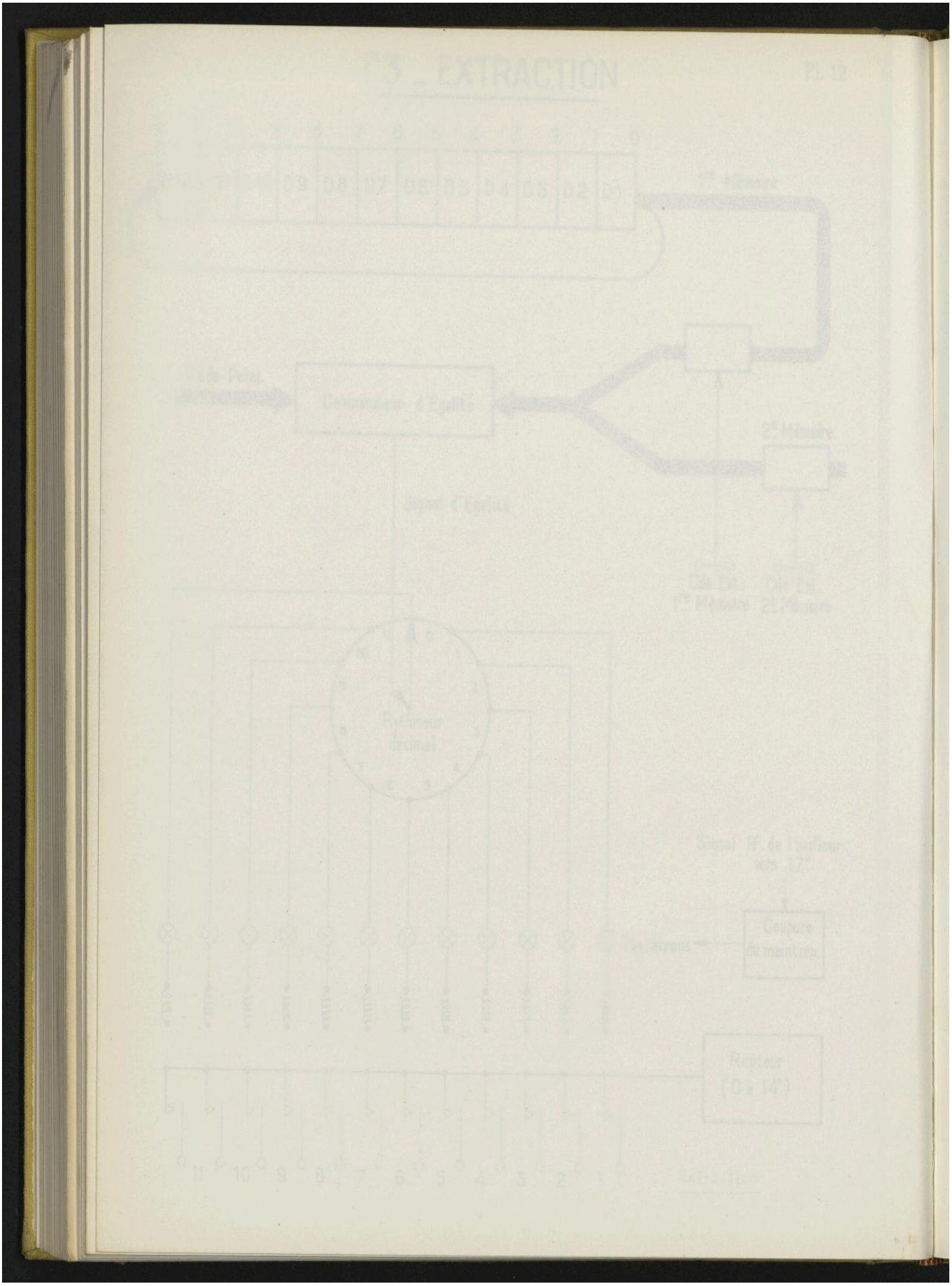




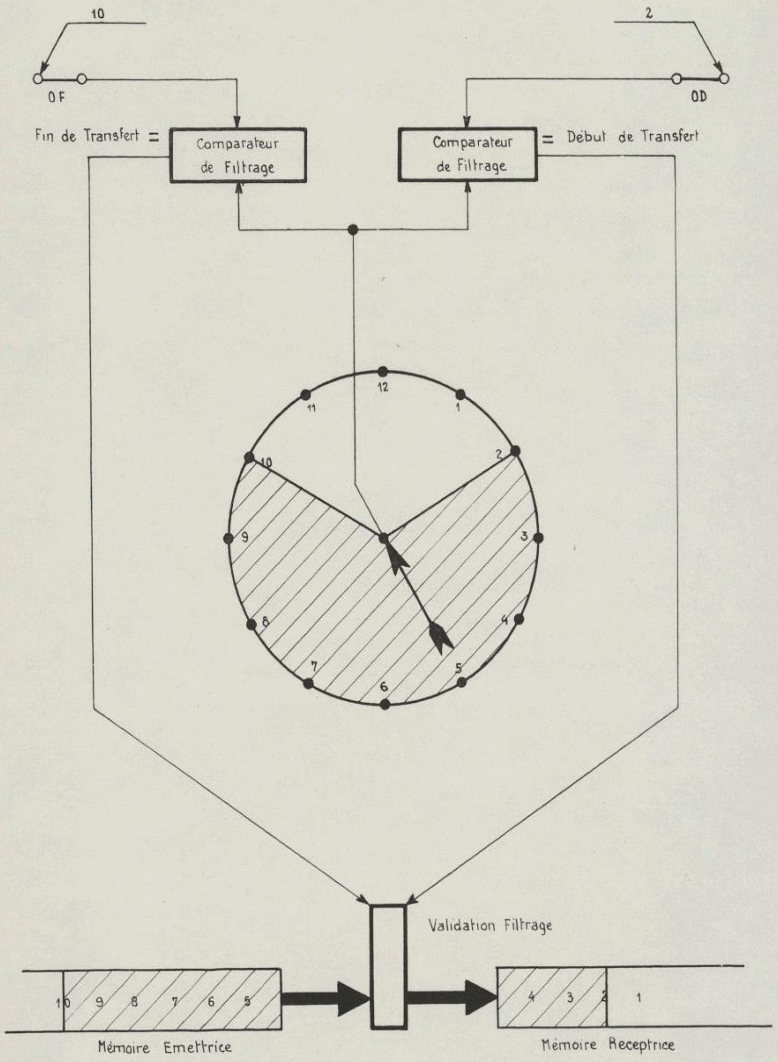
Γ3 _ EXTRACTION

Pl. 12

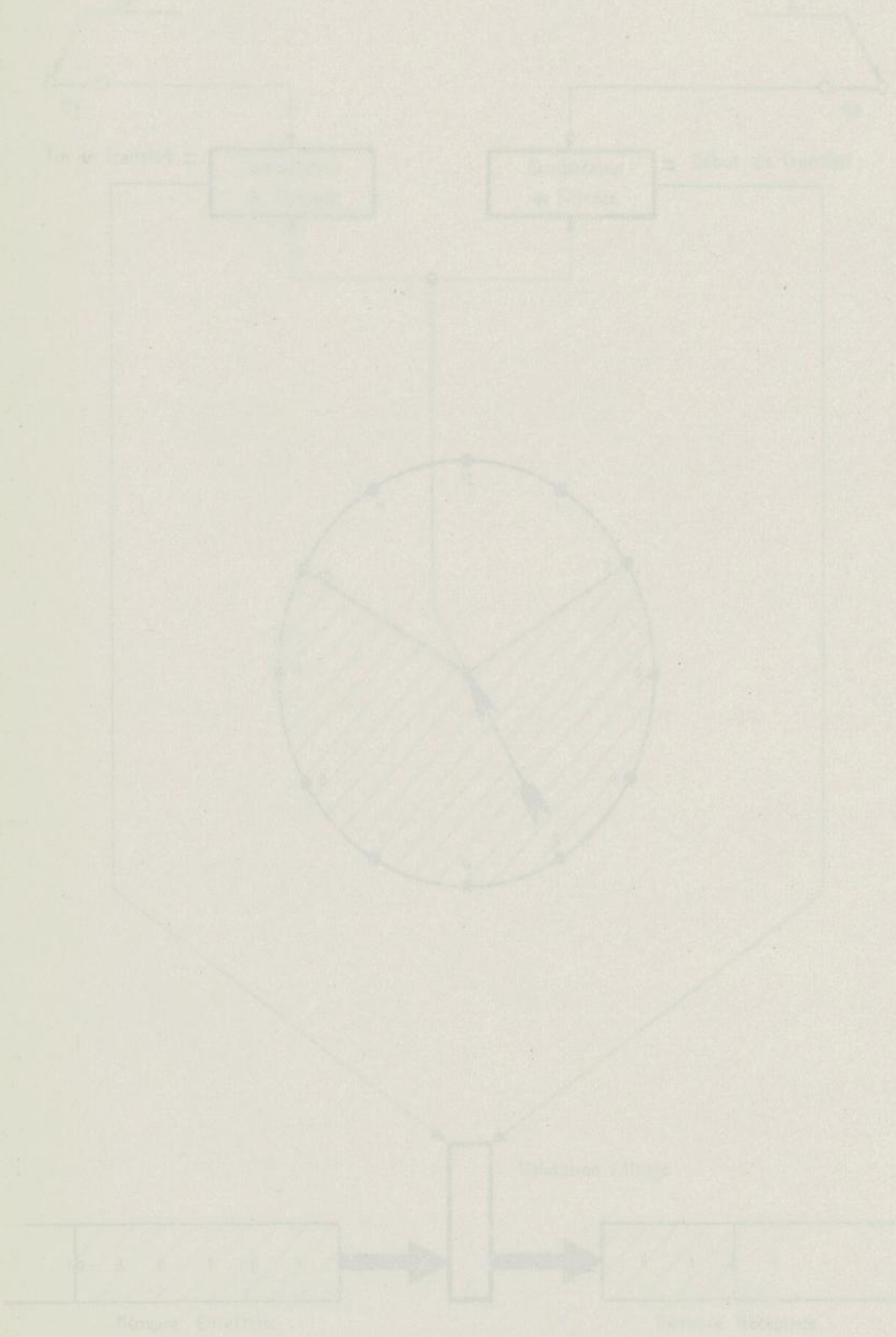




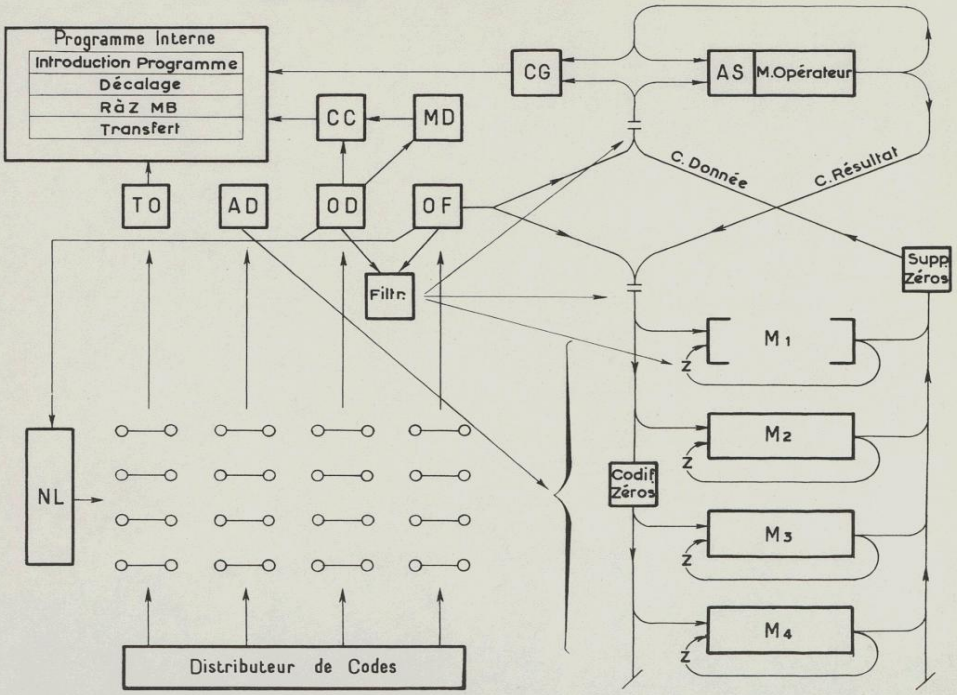
Γ3 _ FILTRAGE

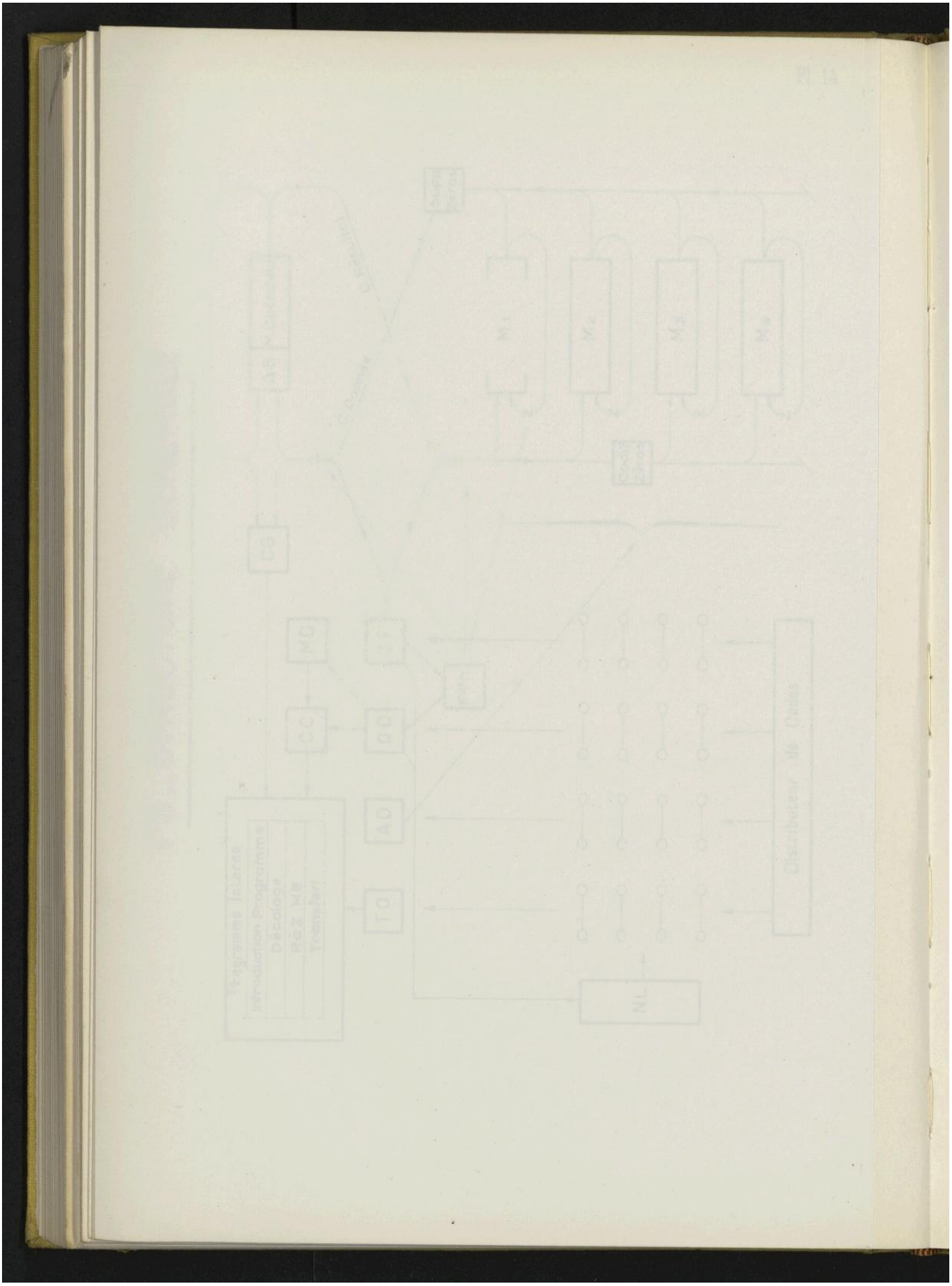


Γ3 - FILTRAGE

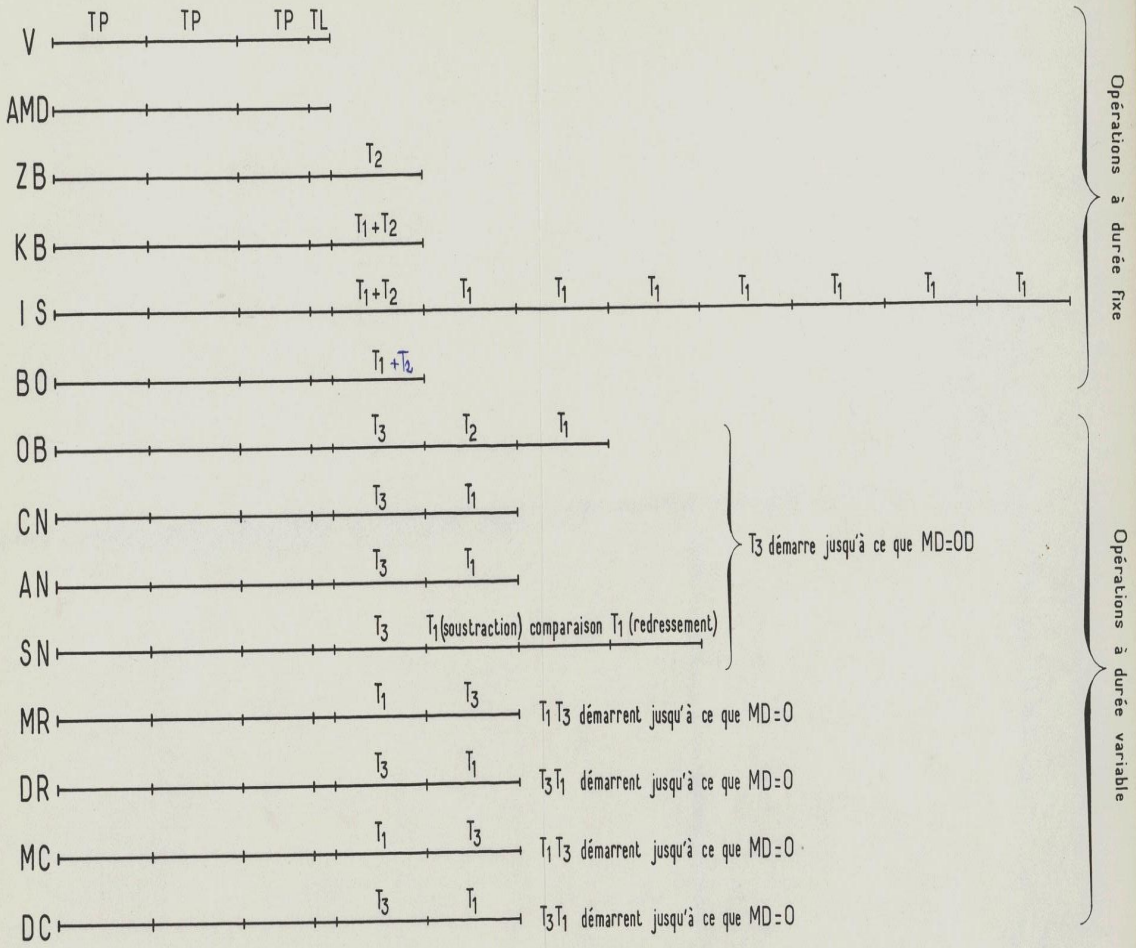


Γ3_STRUCTURE GENERALE





OPERATIONS Γ3_PROGRAMMES INTERNES

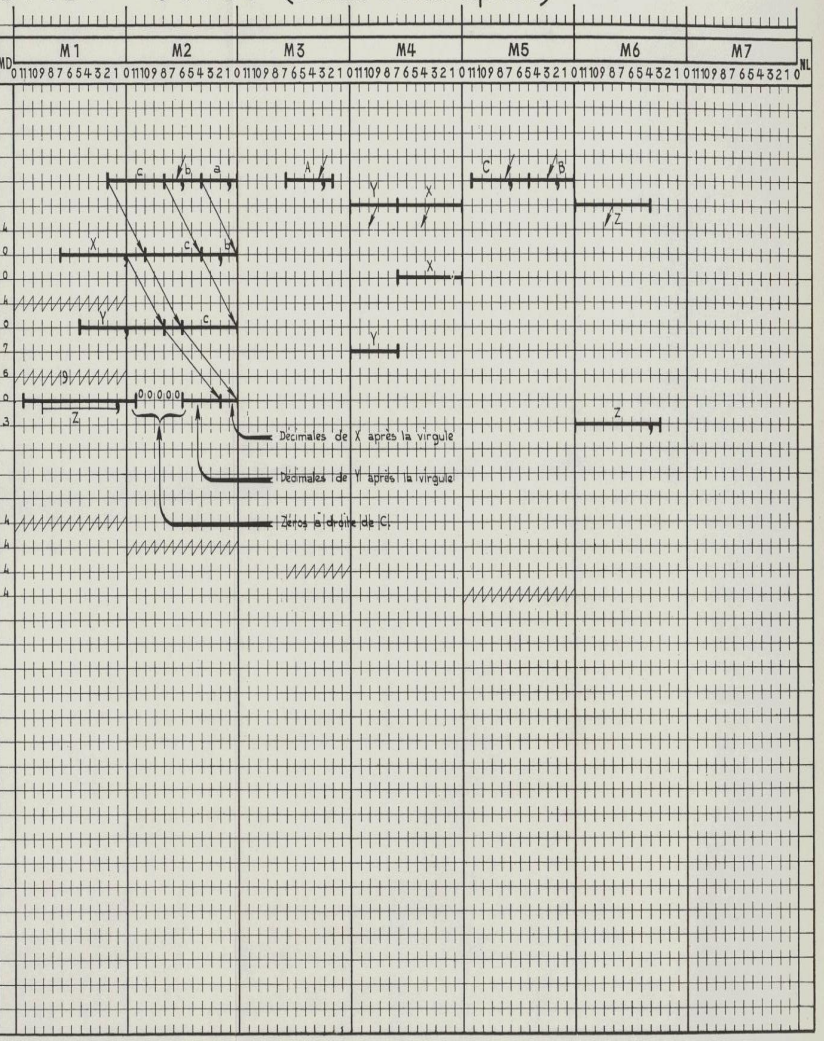


Durée d'un train 0ms,172 soit en BS 155 trains théoriques par point machine.
en PRD 161 trains théoriques par point machine.
(Ajouter 30 trains au nombre théorique trouvé lors de l'étude d'un problème.)

MULTIPLICATIONS SUCCESSIVES

A x a = X B x b = Y C x c = Z (Z arrondi à l'unité supèrieure)

Sélection Condition	Instruction N°	S	C	NS	Sélection lignes	NL	Codes TO	AD	OD	OF	Opération	MD
											Introd	
											Extract	
						0					VPM	4
						1	4	3	2	7	MC	0
						2	8	4		7	OB	0
						3	6		4		BO	4
						4	4	5		5	MC	0
						5	8	4	7		OB	7
						6	6		6	9	BO	6
						7	4	5	5	4	MC	0
						8	8	6	3		OB	3
											"	
											"	
											"	
						28	6		4		BO	4
						29	3	2			ZB	4
						30	3	3		7	ZB	4
						31	3	5			ZB	4



CONNECTEURS _ GAMMA _ M _

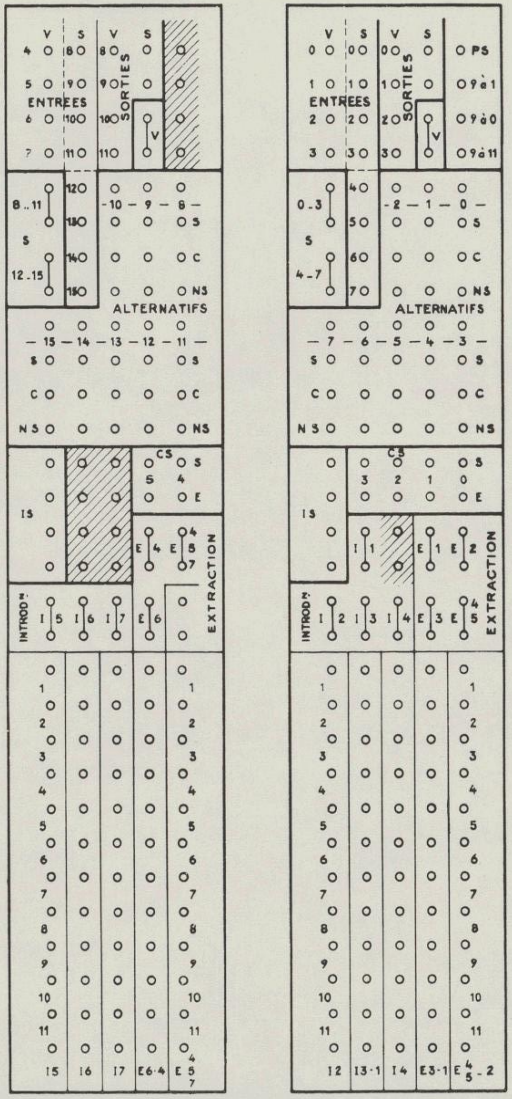
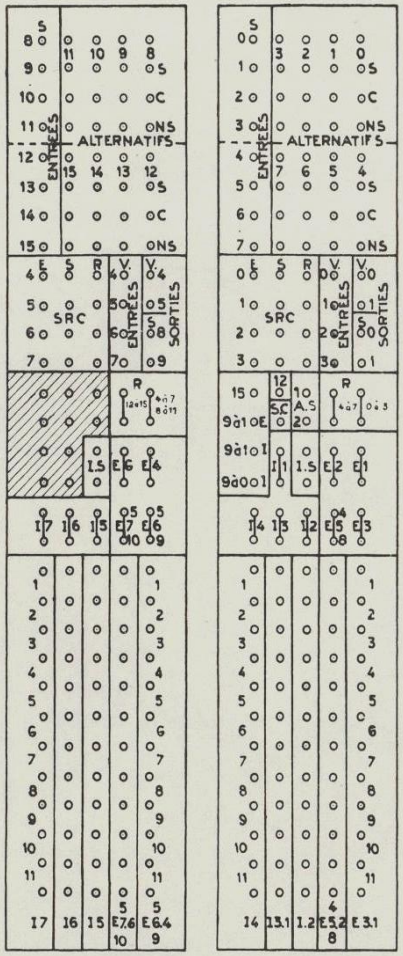


TABLEAU DE CODE GAMMA _ M _

OF ↘	AD →				0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	← AD	
OD ↓	0	4	8	12	Jamais	>	=	≥	$\overline{MS}$				9,14	15,14	1,14	0,14	11,14	12,14	13,14	14,14	0	V
	1	5	9	13	Toujours	≤	≠	<	$\overline{MS}$				8,3	9,3	0,3	11,3	12,3	13,3	14,3	15,3		
	2	6	10	14	V_0	V_1	V_2	V_3	V_4	V_5	V_6	V_7		2,14	3,14	4,14	5,14	6,14	7,14	8,14		
	3	7	11	15	$\overline{V}_0$	$\overline{V}_1$	$\overline{V}_2$	$\overline{V}_3$	$\overline{V}_4$	$\overline{V}_5$	$\overline{V}_6$	$\overline{V}_7$	Pl. 3	1,3	2,3	3,3	4,3	5,3	6,3	7,3		
0	0	1	2	3	CONDITION DE TRANSFERT EN NL																	
1	4	5	6	7																	1	
2	8	9	10	11																	2	AMD
3	12	13	14	15					R à Z Filtrée				R à Z Filtrée				Maintien MD				3	ZB
4	16	17	18	19						OF → M_1				OF → MB							4	KB
5	20	21	22	23					M_1	Intr.f → M_1 OD = MD			MB	Intr.f → MB							5	IS
6	24	25	26	27	OF → M_1				Maintien f. M_1				MB → M_1				R à Z totale M_1 OD → MD				6	BO
7	28	29	30	31													M_2 → MD				7	BD
8	32	33	34	35					R à Z f. M_1				R à Z f. MB M_1 → MB				Opérations avec cadrage préalable → MD = OD				8	OB
9	36	37	38	39	Suppres.n cadrage préalable	M_1 ≧ OF			M_1 t. ≧ M_1 f.				M_1 ≧ MB								9	CN
10	40	41	42	43		M_1 + OF			M_1 x 2				M_1 + MB								10	AN
11	44	45	46	47	Maintien MD	M_1 − OF			R à Z f. M_1				M_1 − MB								11	SN
12	48	49	50	51	M_1 x OF = M_1								M_1 x MB = M_1				Nombre de décalages de M_1 = OD				12	MR
13	52	53	54	55	M_1 / OF = M_1								M_1 / MB = M_1								13	DR
14	56	57	58	59	M_2 x OF = M_1 M_2										M_2 x MB = M_1 M_2		Nombre de décalages de M_1 M_2 = MD				14	MC
15	60	61	62	63	M_1 M_2 / OF = M_2										M_1 M_2 / MB = M_2						15	DC
↑ OD	NL				0				1				2		3 à 15		← AD				↑ TO	

_GAMMA_3_B_



_GAMMA_3_B_

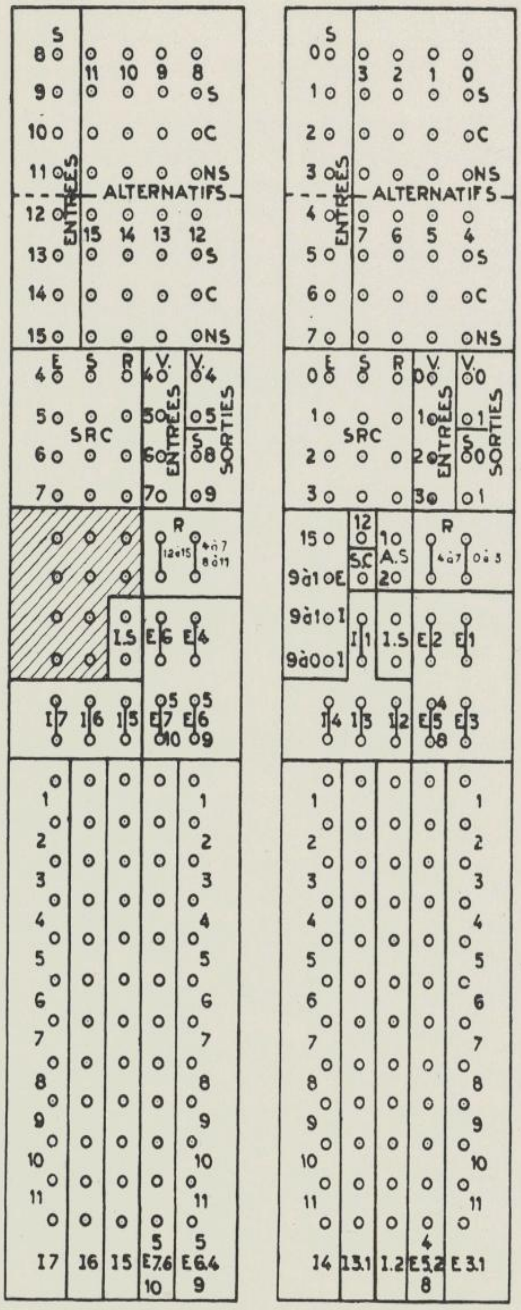
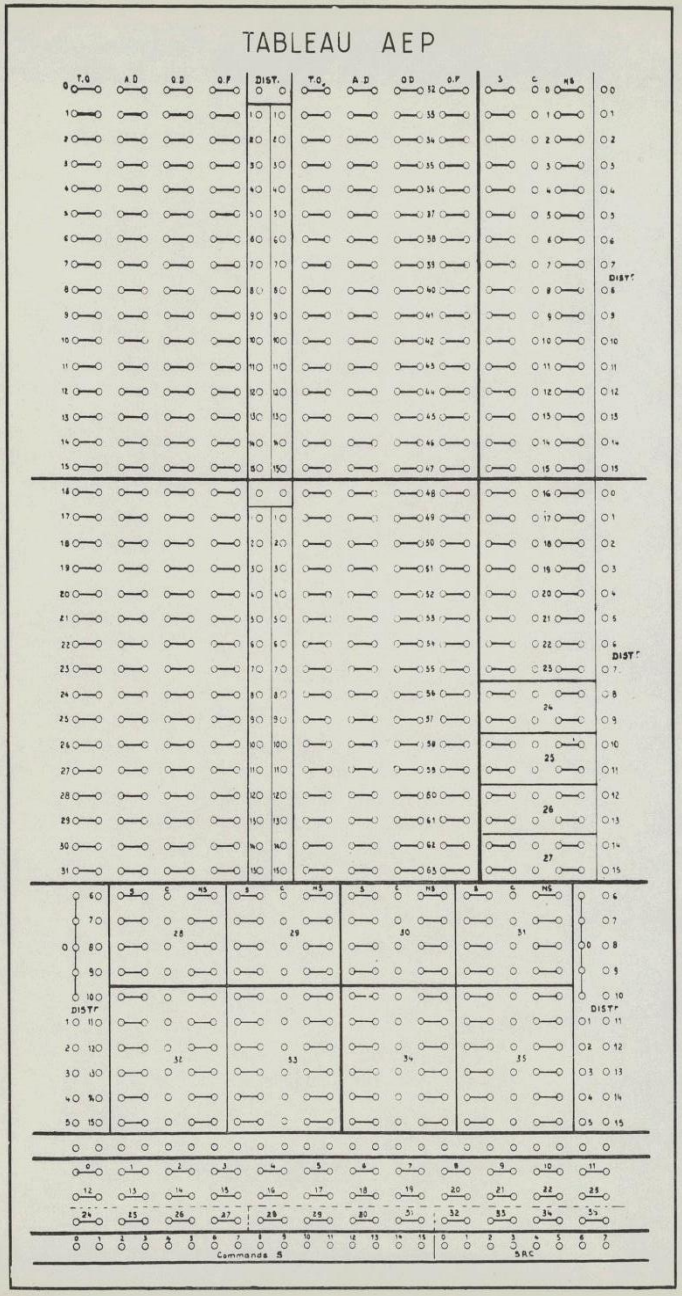
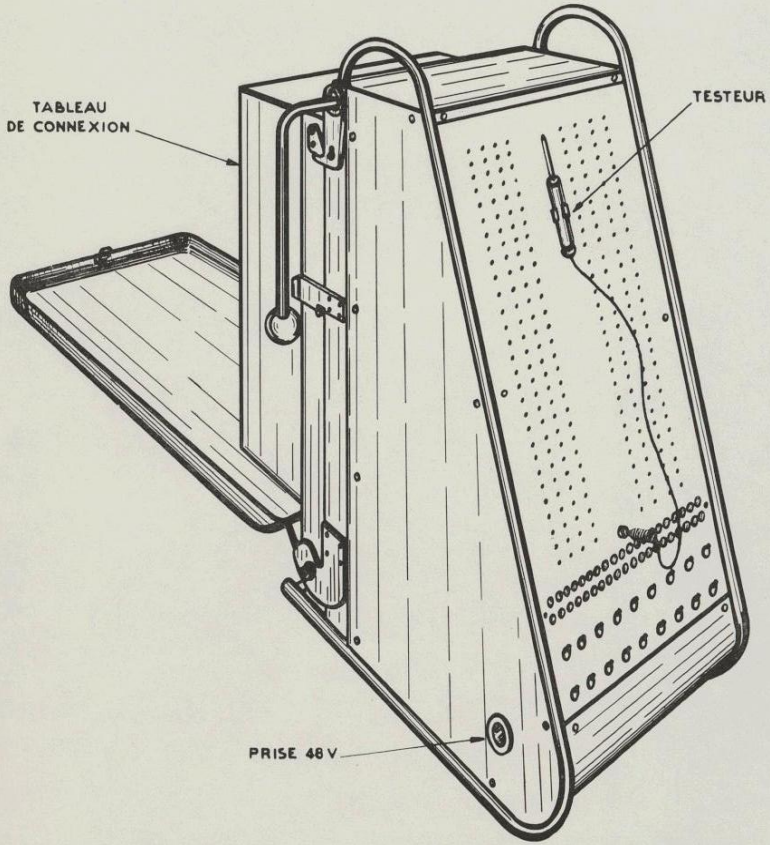
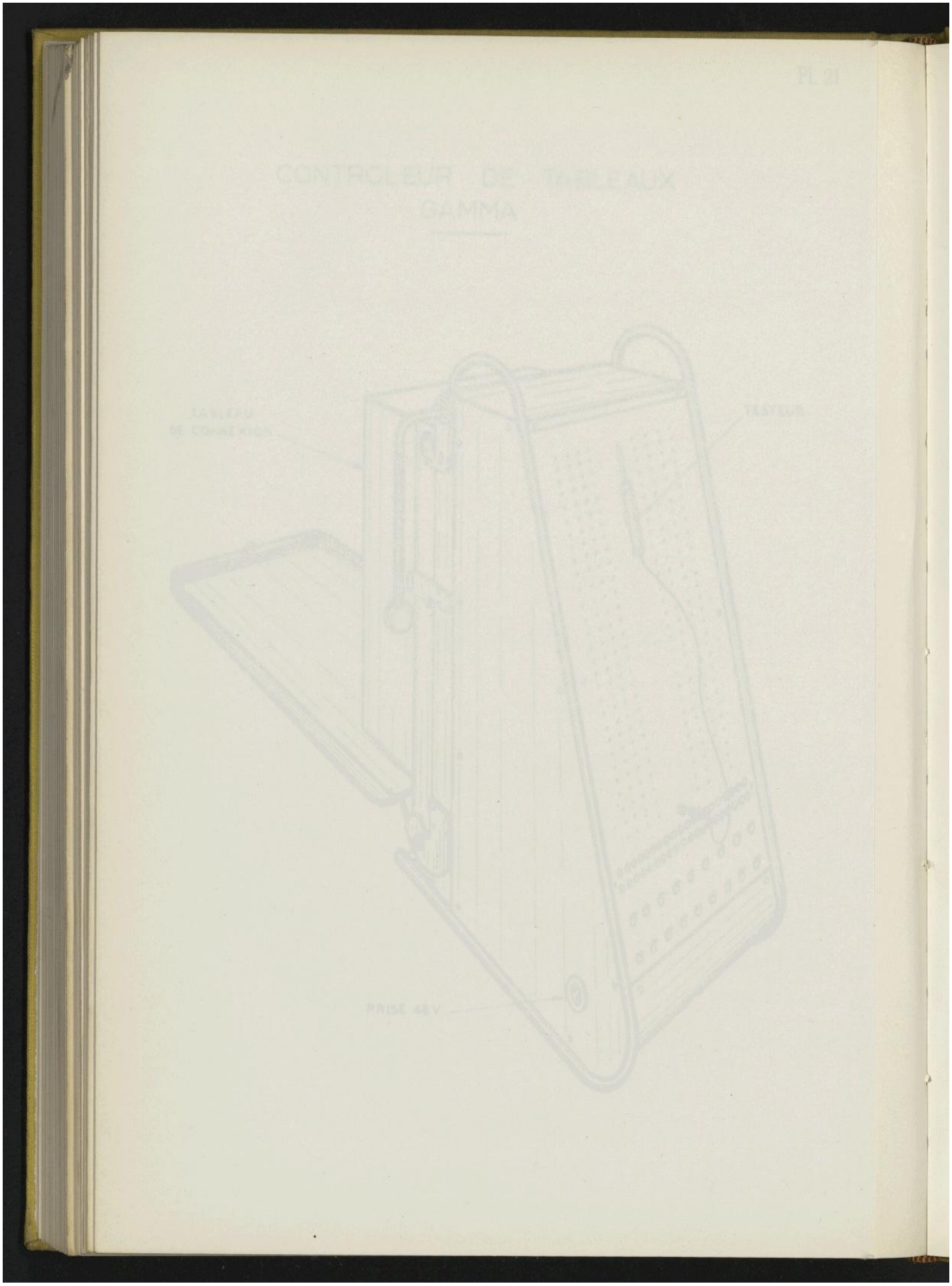


TABLEAU AEP

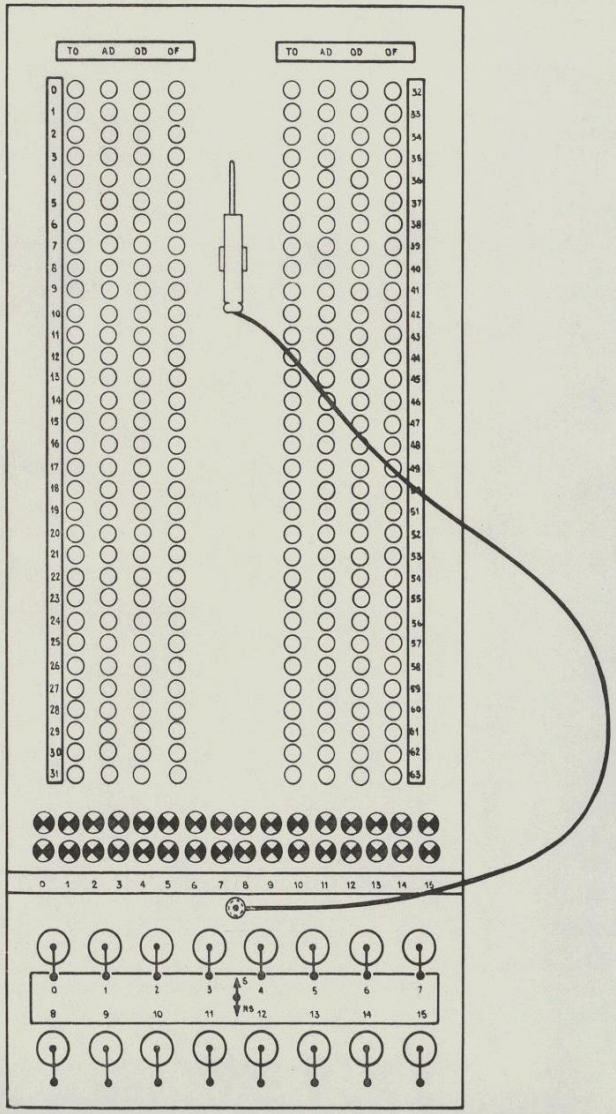


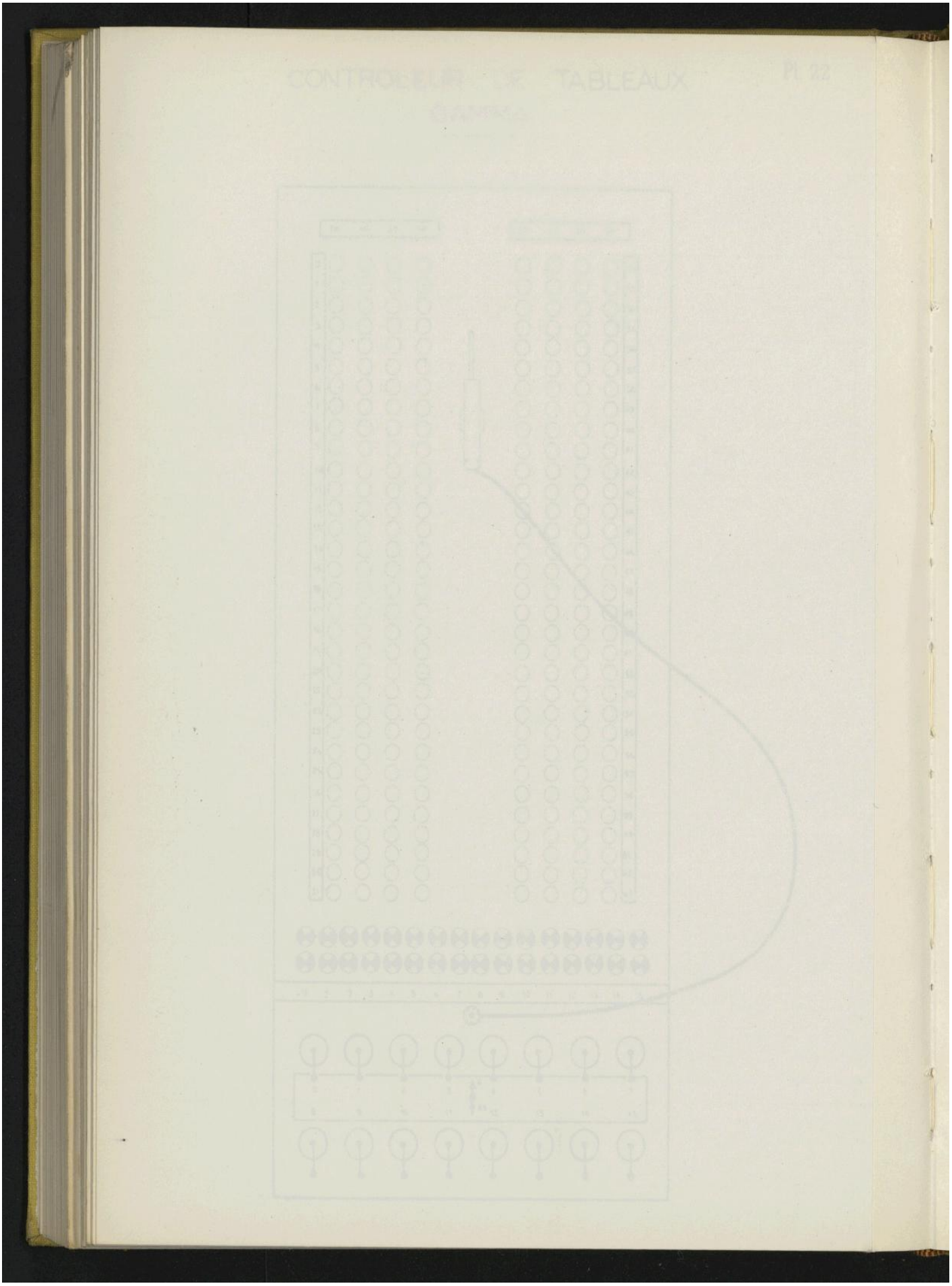
CONTROLEUR DE TABLEAUX GAMMA





CONTROLEUR DE TABLEAUX GAMMA





REMISES

Cinématique

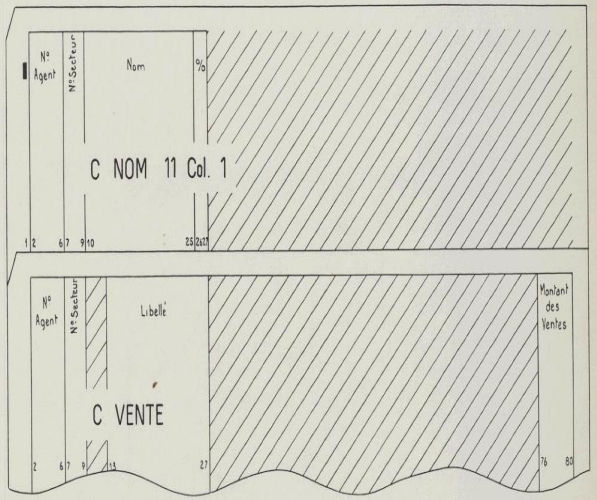
Sélection Condition	Instruction N°	S	C	NS	Sélection lignes	NL	Codes TO	AD	OD	OF	Opération	MD	M1	M2	M3	M4	M5	M6	M7	NL
1re Solution						0		10		5	VPM	0	T							
						1	9	2		4	CN	0								
						2	12	2	5	8	MR	0	Brut							
						3	8	4	3	11	OB	3				Brut				
						4	6				BO	0								
T ≥ 1000 V_0						5			3		$\overline{VR}$	0								
CN 2500						6		3	2	9	$\overline{VC}$ <	0								
T ≥ 10000 V_1						7		1	2	7	$\overline{VR}$	0								
						8	6	2	1	4	BO	1	250							
						9	10	2	8	10	AN	8	+75							
						10	10	4		3	AN	0	+125							
						11	12	4	3	11	MR	0	Remise							
						12	8	3			OB	0			Remise					
						13	6				BO	0								
						14	3	2	5	8	ZB									
2me Solution											Intr		T	Pu 25						
						0		10		5	VPM	0								
						1	9	2		4	CN	0	T							
						2	12	2	5	8	MR	0	Brut							
						3	8	3	3	11	OB	3			Brut					
						4	6		A	B	BO	A	1 ou 5							
T ≥ 10000	0	1	A	0		5		3	1	13	VC <	A								
	0	1	B			6	4			8	KB	A	8							
T ≥ 1000	1	5		0		7	12	2	2	4	MR	0	% R							
						8	12	3	3	11	MR	0	Remise							
						9	8	4		11	OB	0				Remise				
						10	6				BO	0								
						11	3	2			ZB	0								

BONIFICATIONS

Nom et Libellé	N° Secteur	N° d'Agent	Ventes et Total	Ristournes	Primes	%
DURANTIER JEAN	100	21250				
BAS. COTON			11480			
CHEM. NYLON			12000			
GANTS			11000			
			34480	1120		
GERMINAL ANDRE		21328				
BAS. NYLON			15000			
IMPERMEABLE			9000			
			24000			
RAMONAUD GASTON		21450				
CHEM. NYLON			25000			
ANORAK			5000			
			30000	200		
BIGOTTEAU LEON		21552				
GANTS			8000			
IMPERMEABLES			27000			
ANORAK			7000			
			42000	3000	765	50

Nom et Libellé	N° Secteur	N° d'Agent	Ventes et Total	Ristournes	Primes	%
GARANCIA LOUIS	125	30250				

Feuillet Plein



BONIFICATIONS

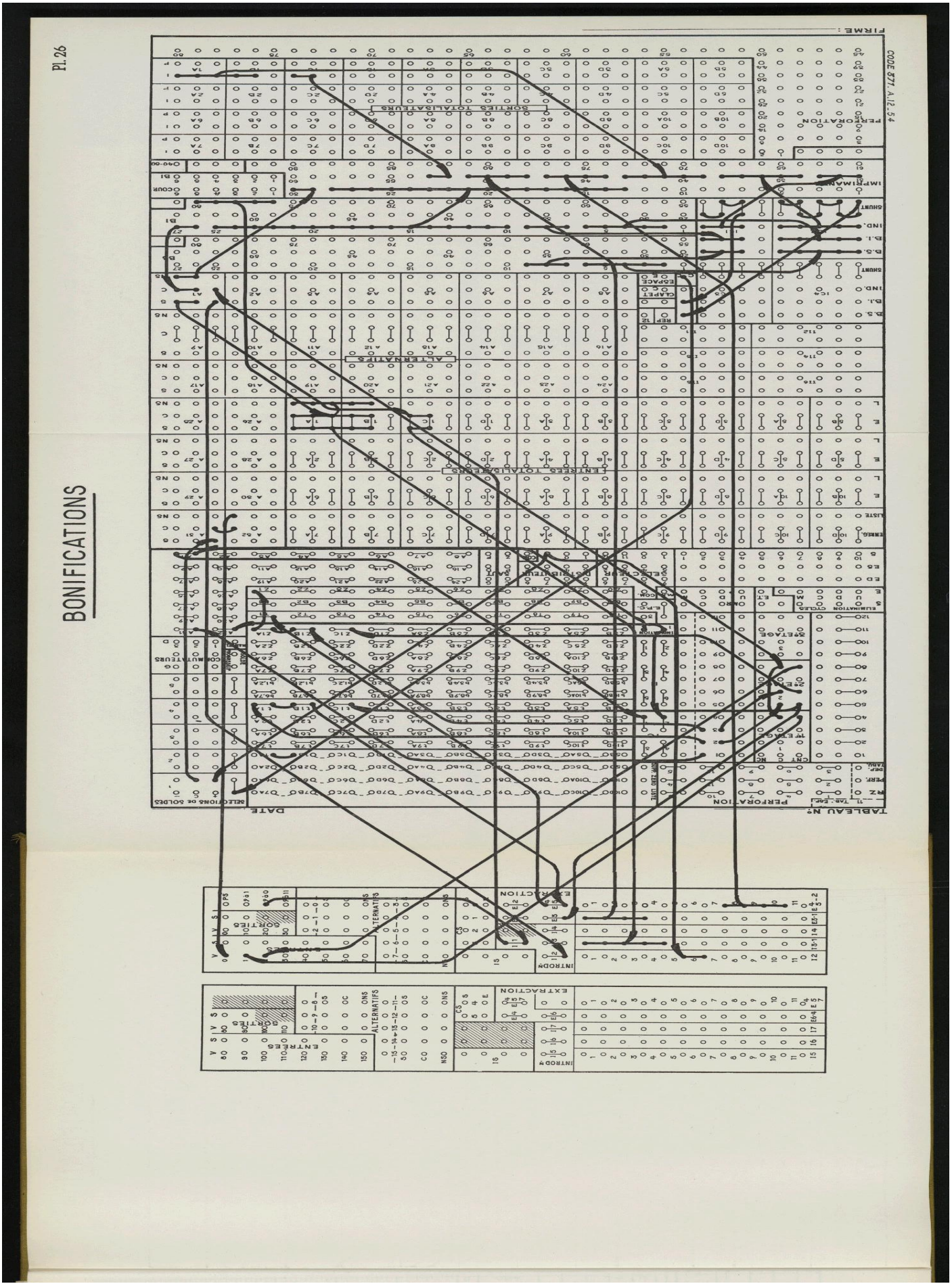
Cinématique

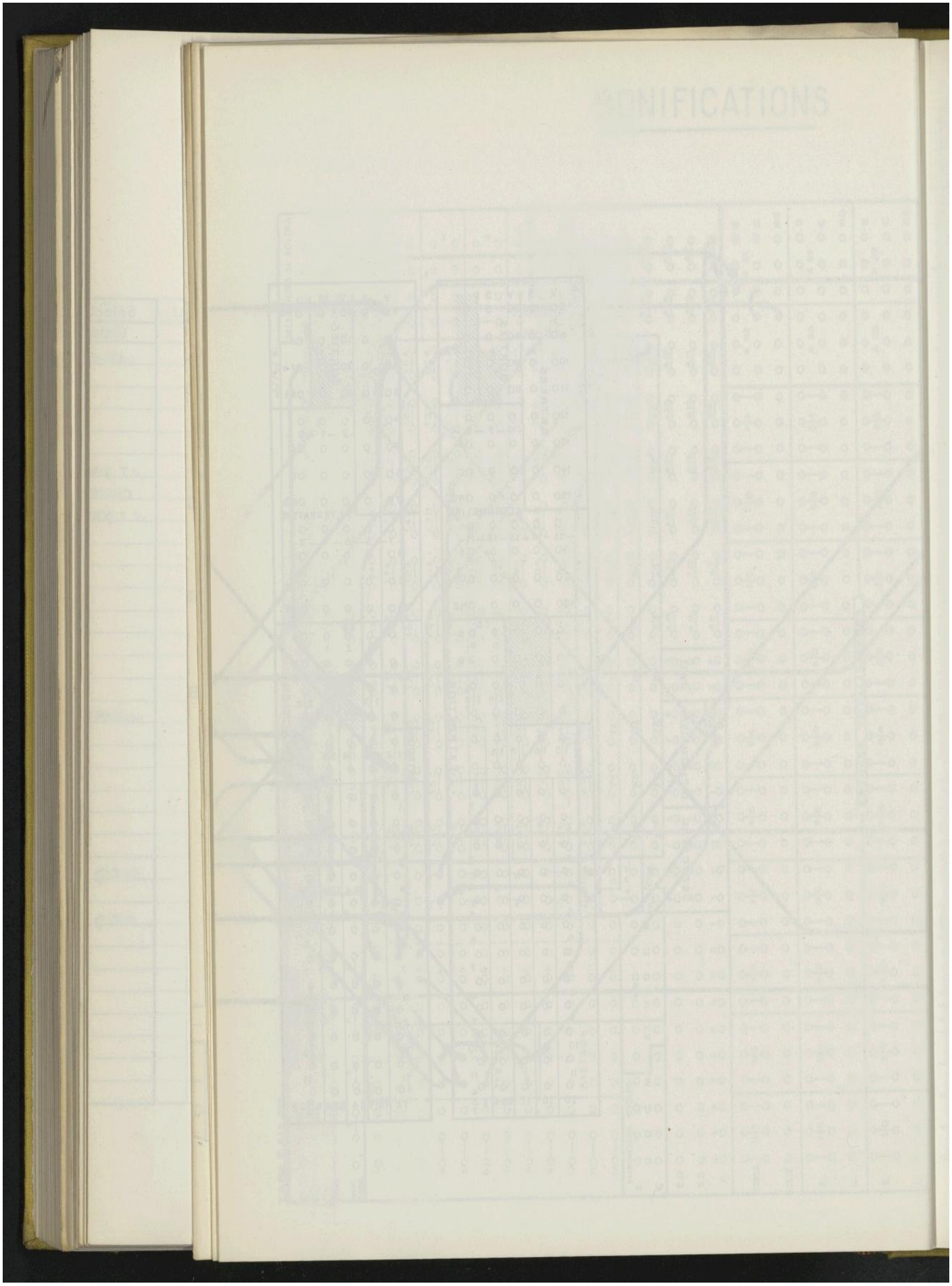
Cdes S N°	S	C	NS	Sélection lignes	NL	Codes TO	AD	OD	OF	Opération	MD	M1	M2	M3	M4	M5	M6
VPM	11	14			0		12		5	VPM	0						
Cartes			V_0		1			11	7	$\overline{VR}$	0						
					2	10	4		5	AN	0					Carte 3	
					3	8	4		5	OB	0						
					4	6				BO	0					carte	
1er Etage			V_1		5		1		3	$\overline{VR}$	0						
					6	6	4	5	7	BO	5						
					7	10		5	1	AN	5						
					8	8	4	5	7	OB	5						
					9	6	4		5	BO	0						
					10	3	4		5	ZB	0						
					11	9		4	3	CN	0						
					12	3	4	7	11	ZB	0						
					13		3	4		VC >	0						
					14	4	4	7	11	KB	0				11	1er Etage	
					15			6	1	VS	0						
					16		1	4	12	VC >	0						
					17	4	4	9	2	KB	0				2		
					18			6	1	VS	0						
					19	11		4	3	SN	0	Excédent					
					20	13		4	4	DR	0						
					21	8	4	7	11	OB	7				R		
					22	10	2		5	AN	0						
					23	8	2		5	OB	0						
					24	6				BO	0						
2me Etage			V_2		25		2		3	$\overline{VR}$	0						
					26	6	2		5	BO	0						
					27	13	4	5	7	DR	0						
					28	6	1		5	BO	0						
					29	9		3	1	CN 1000	0						
					30		1	9	8	VC >	0						
					31	9		2	5	CN 500	0						
					32		1	8	12	VC >	0						
					33	6				BO	0						
					34			10	9	VS	0						
					35	6		2	5	BO	2					2me Etage	
					36	2				AMD	0						
					37			10	9	VS	0						
					38	11		2	5	SN	0	RM 500					
					39	12	2	5	7	MR	0	(RM 500) A					
					40	12		2		MR	0						
					41	10		2	5	AN	0	Prime					
					42	8	3		5	OB	0			Prime			
					43	3	4	5	7	ZB	0						
					44	3	2			ZB	0						
					45	6				BO	0						

Prime Remise

EXTRACTION

Pl. 25





Problème: $B = \frac{UB \cdot 10^3}{V\sqrt{3}}$

B tjs compris entre 1 et 1,99

Sélection lignes	NL	Codes TO	AD	OD	OF	Opération	MD	M1	M2	M3	M4
	Carte	N° X					0			V	VB
		4	2	7	1	KB	0				
	9	4	2	6	7	KB	0				
		4	2	5	3	KB	0				
		4	2	4	2	KB	0		$\sqrt{3}$		
		6	3		4	BO	0	V			
	8										
		12	2	4	8	MR	0	$V \cdot \sqrt{3}$			
		6	1	2		BO	2	$V\sqrt{3}$			
	7	8	2	4	10	OB	4		$V \cdot \sqrt{3}$		
		6	4	3	6	BO	3	UB			
		9	2	6	9	CN	6	$UB \cdot 10^3$	Multiplication par 10^3		
	6										
		13	2	4	10	DB	0	B			
		6	1		3	BO	0	B	décimales		
	5	8	3	9		OB	9			B	
		6				BO	0				
		3	2			ZB	0				
						Suite du Programme					

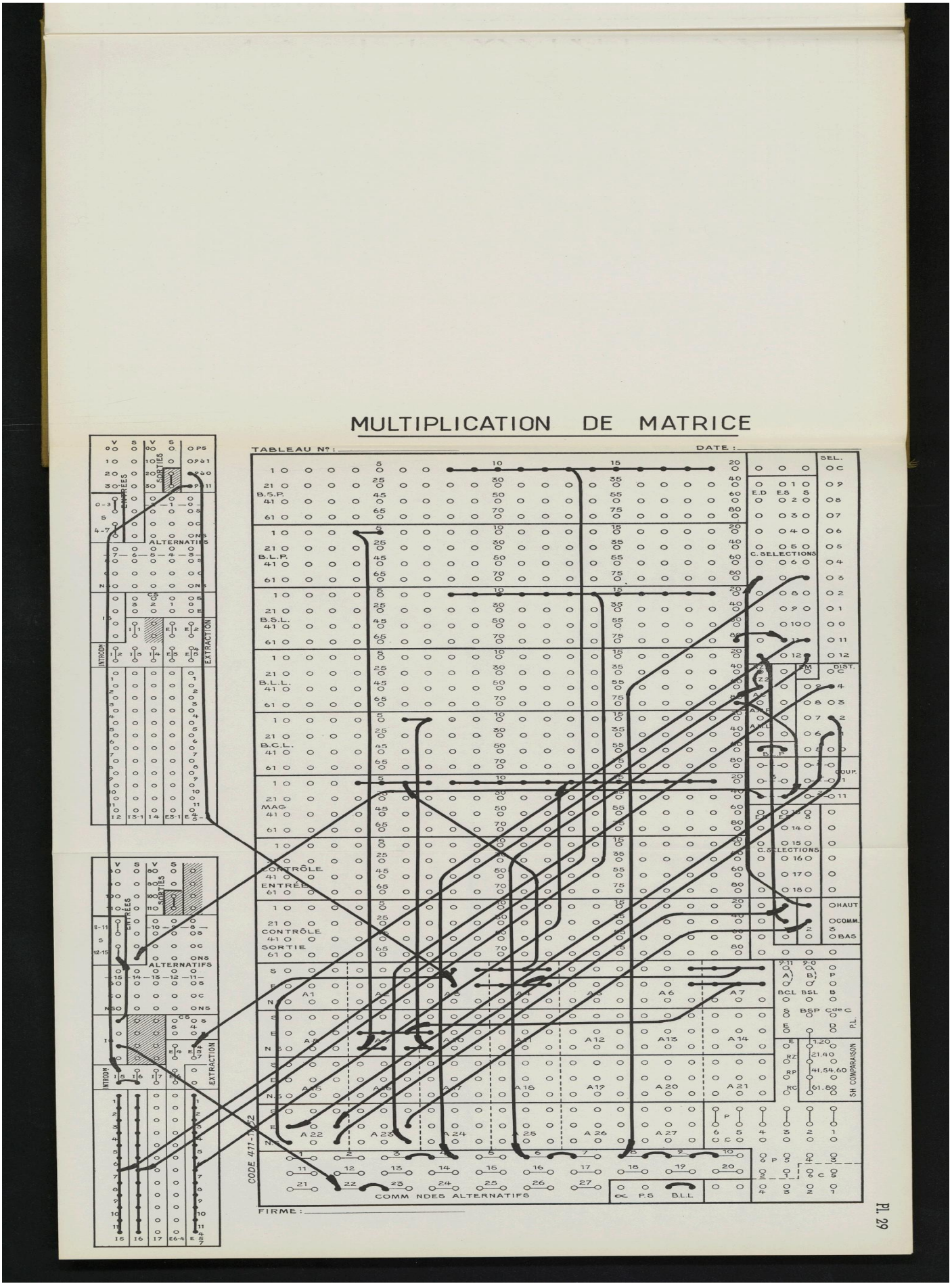
MULTIPLICATION DE MATRICE (A x B + Σ)

	Cdes S				Sélection lignes	NL	Codes				Opération	MD
	Nº	S	C	NS			TO	AD	OD	OF		
						0	6	5		2	BO	0
						1	9		3	1	CN	0
						2	10	6		2	AN	0
						3	11		1	5	SN	0
						4		4	13	4	VMS	0
						5	8	6		2	OB	0
						6	6	5	2	11	BO	2
						7	8	2			OB	0
						8	6	5	11		BO	11
						9	14	6	2		MC	0
						10	3	3			ZB	0
						11	2		9		AMD	9
						12	15		10	1	DCC	9-δ
						13	6	1	2		BO	2
						14	8	6	2		OB	2
						15	6	6		2	BO	0
						16		1	4	9	V ≤	0
						17	6	5		2	BO	0
						18	10	2		1	AN	0
						19	11			8	SN	0
						20	8	6		2	OB	0
						21	6	6	10	11	BO	10
						22	9				CN	10
						23		2	6	5	VC ≠	10
						24	3	6			ZB	10
						25	6	3	11		BO	11
						26	9				CN	11
						27	4	3	11	2	KB	11
						28		2	12	13	VC ≠	11
						29	6	7			BO	0
						30	8	5			OB	0
						31	6	5		2	BO	0
						32	9	6		2	CN	0
						33	11	6		2	SN	0
						34	12	1			MR	0
						35	8	2		1	OB	0
						36	11			8	SN	0
						37		4	11	1	VMS +	0
						38	6	5	2		BO	2
						39		3	10	9	VC <	2
						40	6	6	2		BO	2
						41		2	12		VC =	2
						42	7				BD	
						43	14				MC	0
						44	6	1	2		BO	2
						45		3	12		VC ≥	2
						46	10	6	2		AN	2
						47			12	5	V ↑ja	2
						48	10	5	2		AN	2
						49	12		2		MC	0
						50			2	13	V ↑ja	0
						51	6	6			BO	0
						52	8	7			OB	0
						53	3	5			ZB	0
						54	3	6			ZB	0
11 col. 80 BLL	5'16	3	A	9		55		12	15	12	VPM 11,14	0
						63	A	7				

M1 M2 M3 M4 M5 M6 M7

EXTRACTION

Pl. 28

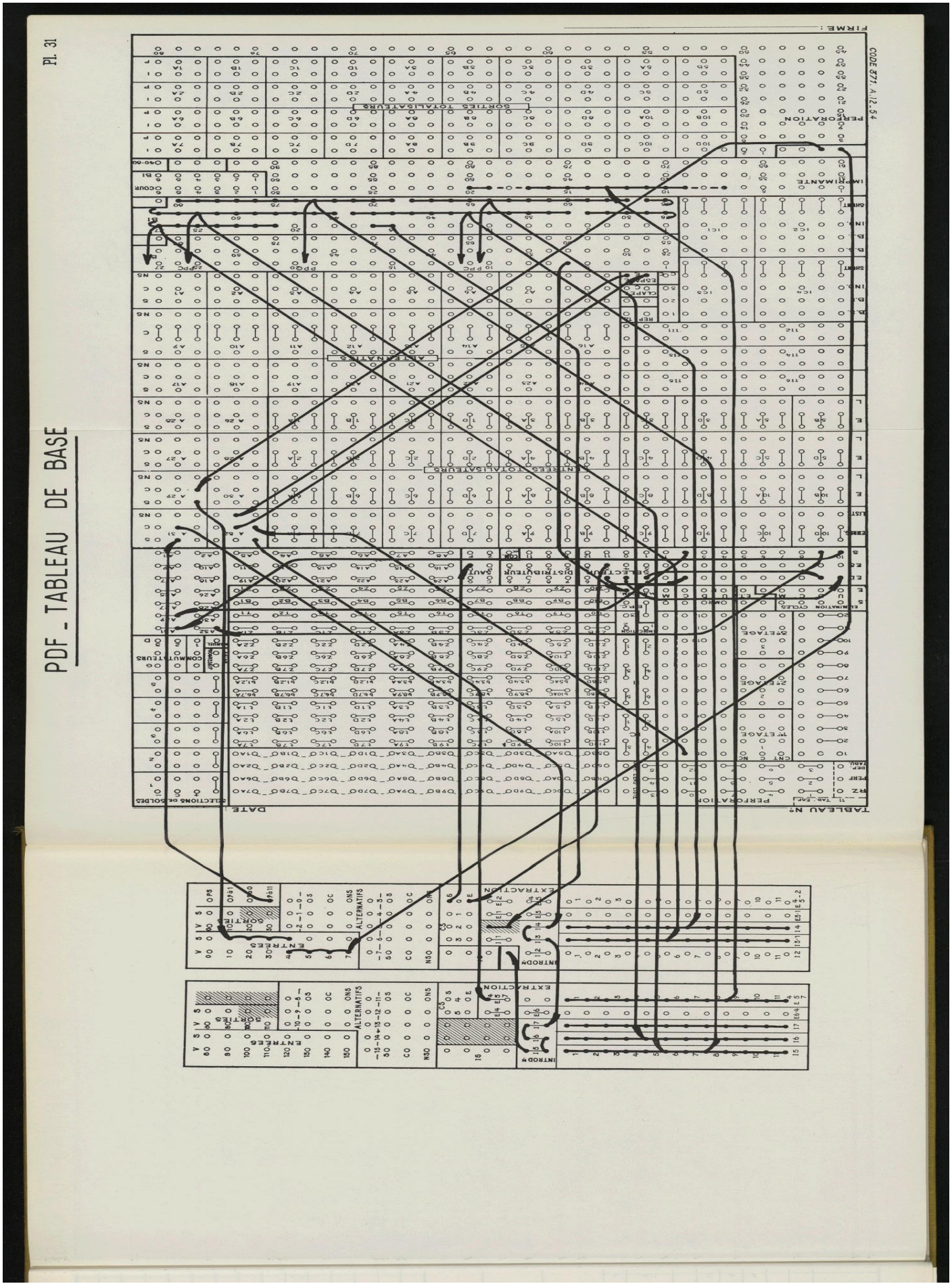


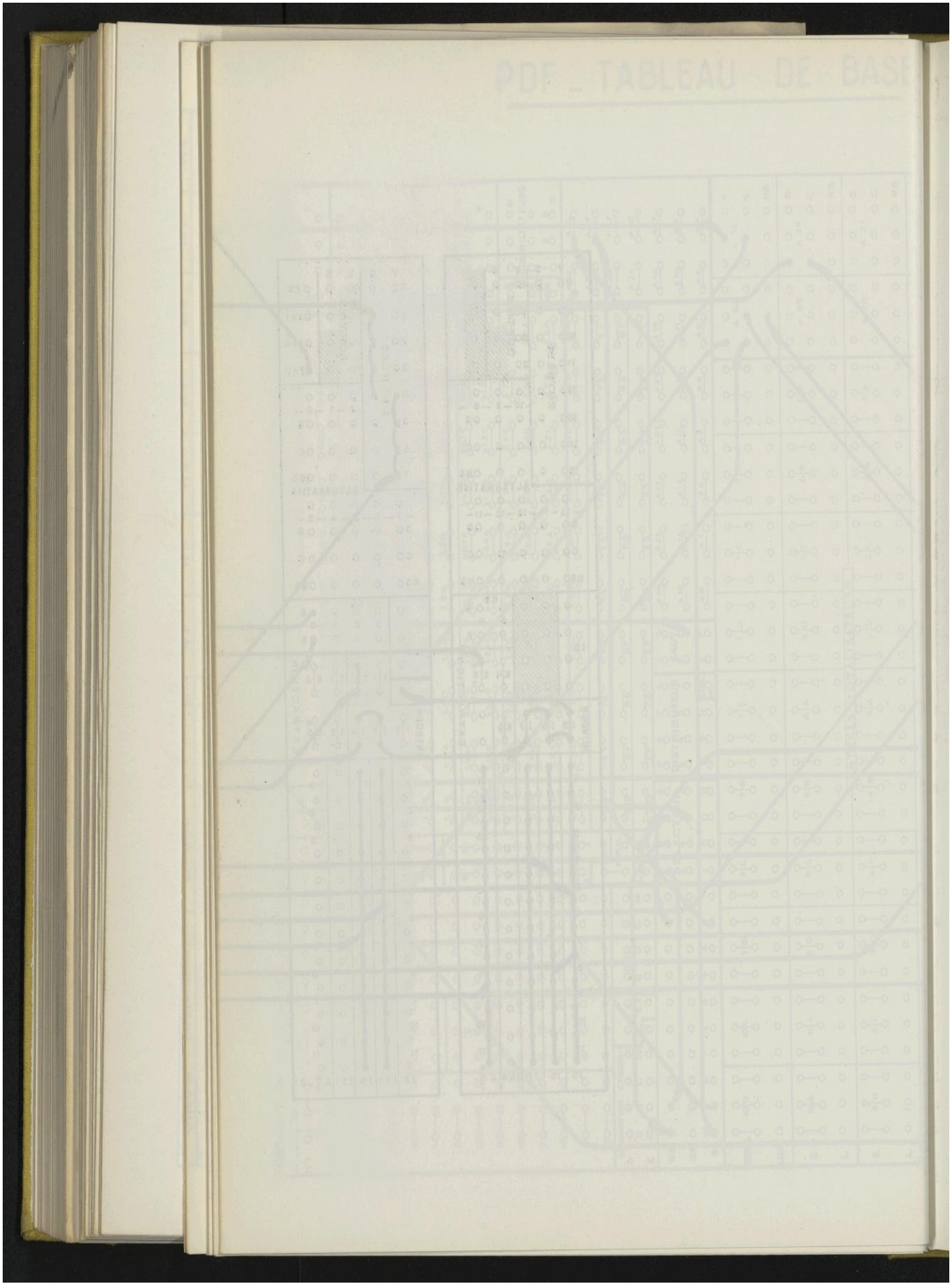
RACINE CARREE

$x = \sqrt{A}$ $\quad x_0 = (A) + 0{,}2$ $\quad x_1 = \frac{1}{2}\left(x_0 + \frac{A}{x_0}\right)$

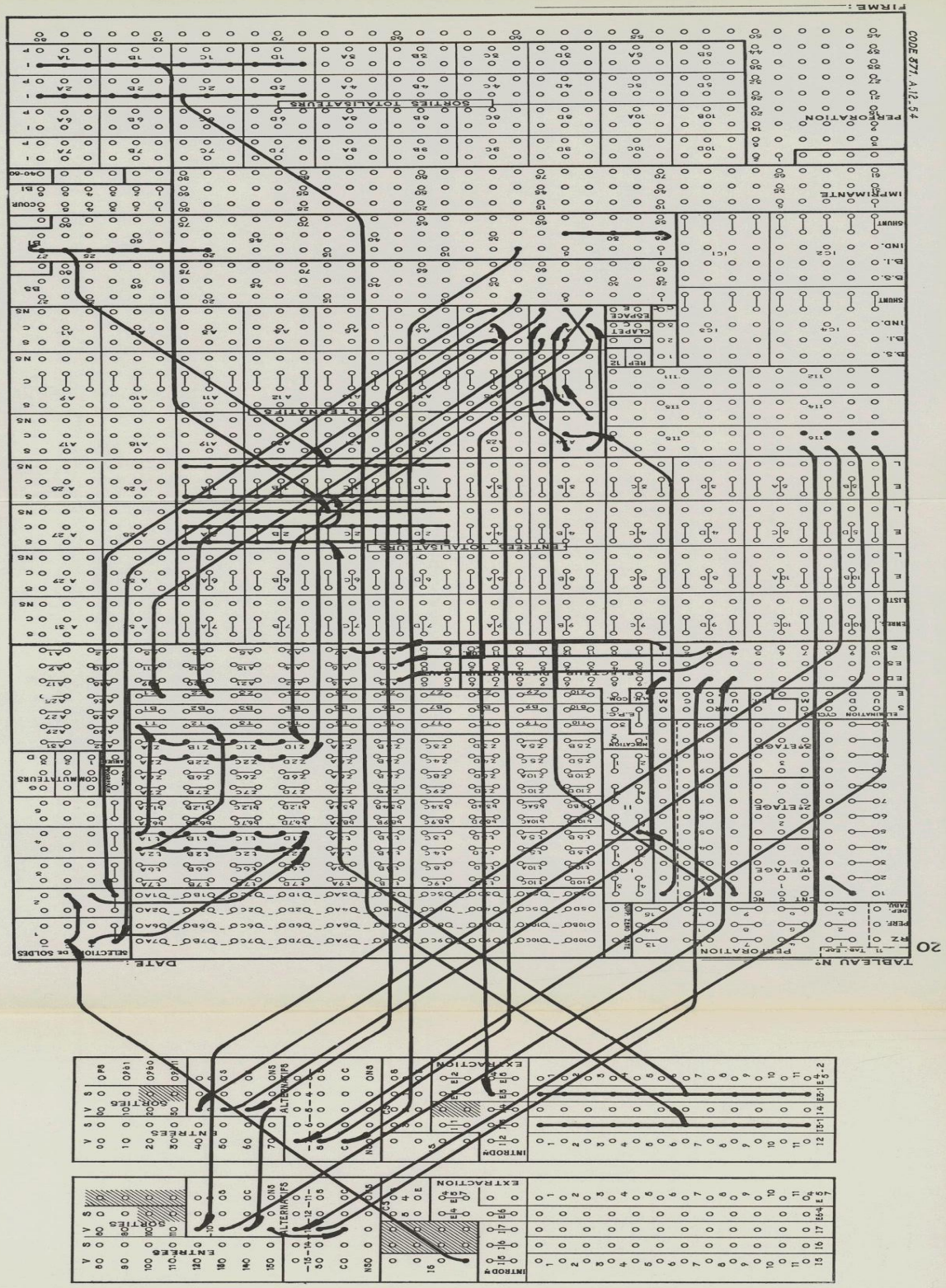
Remarques	PM	AS	TO	AD	OD	OF	Opération	MD	M1	M2	Mx	1	2	3	4	5	6	7	108	109	110	111	112	113	114	115
																A										
C.P.A	9		6	5		2	BO	0	e			e														
			10			1	AN	0				e+1							Calcul	de						
			10		1	5	AN	0				e+51	= a						l'exposant							
			13		2	2	DR	0	E			e+51 / 2														
	8		8	6	0	2	OB	0									E									
			9		2	1	CN	0											Etude si a pair ou impair							
			6	5	2	11	BO	2	A			A														
	7			3	15	4	VC ≥	2																		
			2		3		AMD	3											pour cadrage de A si a pair						
			8	6	2		OB	2									A									
			8	5	1		OB	1											pour la précision							
	6		10		9	2	AN	1				x_0							$x_0 = A + 0{,}2$ si a pair, $x_0 = A + 0{,}2$ si a impair							
			9		10	1	CN	1											examen du décalage							
				3	15	12	VC ≥	1																		
			8	6	2		OB	2									x_0		$x_0 \div A$ si (A) + 0,2 > 1							
	5		6	5	1		BO	1				A														
			2				AMD	0																		
			15	6	2		DC	0					A / x_0													
	4		6	2	2		BO	2				A														
			10	6	2		AN	2				x_0														
			12		2		MR	0				$x_0 +$	A / x_0													
			12		10	5	MR	0				x_1							M.en par 0,5 pour D.en par 2							
	3		6	1	1		BO	1											cadrage (limitation)							
			8	6	2	11	OB	2									x_1									
	2		6	5	1		BO	1				A														
			2				AMD	0											itérations							
			15	6	2		DC	0					A / x_1													
	1		6	2	2		BO	2				A														
			10	6	2		AN	2				$x_1 +$	A / x_1													
			12		2		MR	0																		
			12		10	5	MR	0				x_2														
	0		6	1	1		BO	1																		
			8	6	2		OB	2									x_2									
	11		6	5	1		BO	1				A														
			2				AMD	0																		
			15	6	2		DC	0					A / x_2													
	12		6	2	2		BO	2				A														
			10	6	2		AN	2				$x_2 +$	A / x_2													
			12		2		MR	0																		
			12		10	5	MR	0				x_3														
C.P.B	9		6	1	1		BO	1																		
			8	6	2		OB	2									x_3									
	8		6	5	1		BO	1				A														
			2				AMD	0																		
			15	6	2		DC	0					A / x_3													
	7		6	2	2		BO	2				A														
			10	6	2		AN	2				$x_3 +$	A / x_3													
			12		2		MR	0																		
			12		10	5	MR	0				x_4														
	6		6	1	1		BO	0																		
			8	6	2		OB	2									x_4									
	5		6	5	1		BO	1				A														
			2				AMD	2																		
			15	6	2		DC	0					A / x_4													
	4		6	2	2		BO	2				A														
			10	6	2		AN	2				$x_4 +$	A / x_4													
			12		2		MR	0																		
			12		10	5	MR	0				x_5														
	3		6	1	1		BO	1																		
			8	6	2		OB	2									x_5									
	2		6	5	1		BO	1				A														
			2				AMD	0																		
			15	6	2		DC	0					A / x_5													
	1		6	2	2		BO	2				A														
			10	6	2		AN	2				A +	A / x_5													
			12		2		MR	0																		
			12		10	5	MR	0				x_6														
	0		6	1	1		BO	1																		
			8	6	2		OB	2									x_6									
	11		6	6			BO	0				x_6														
			3	2			ZB	0																		

Pl. 30

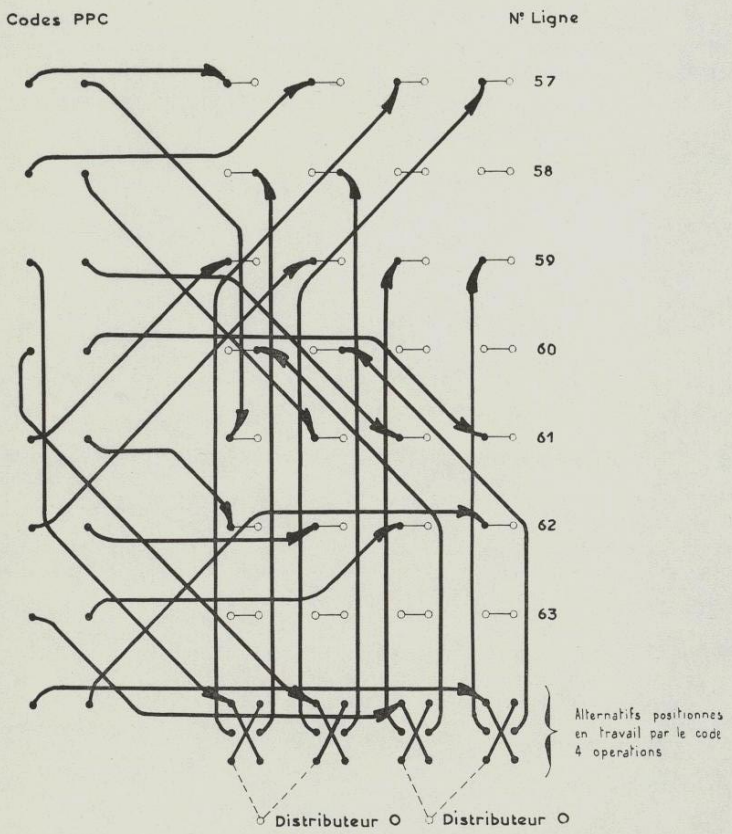




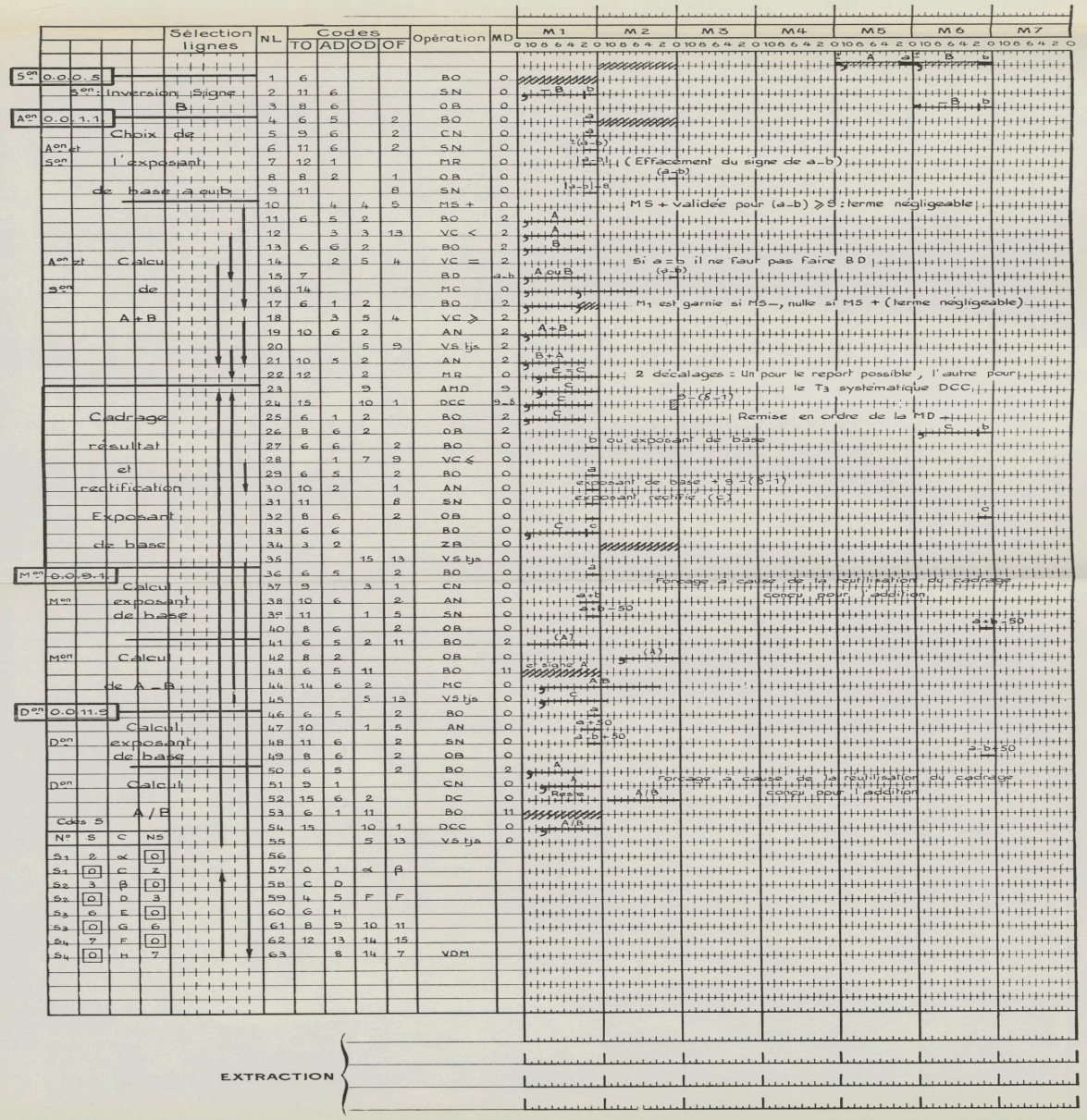
PDF _ ENREGISTREMENT DANS LES TOTALISATEURS



CONNEXION DES LIGNES PPC POUR LE TABLEAU Γ DE POINT DECIMAL FLOTTANT



POINT DECIMAL FLOTTANT A + B , A − B , AB , $\frac{A}{B}$



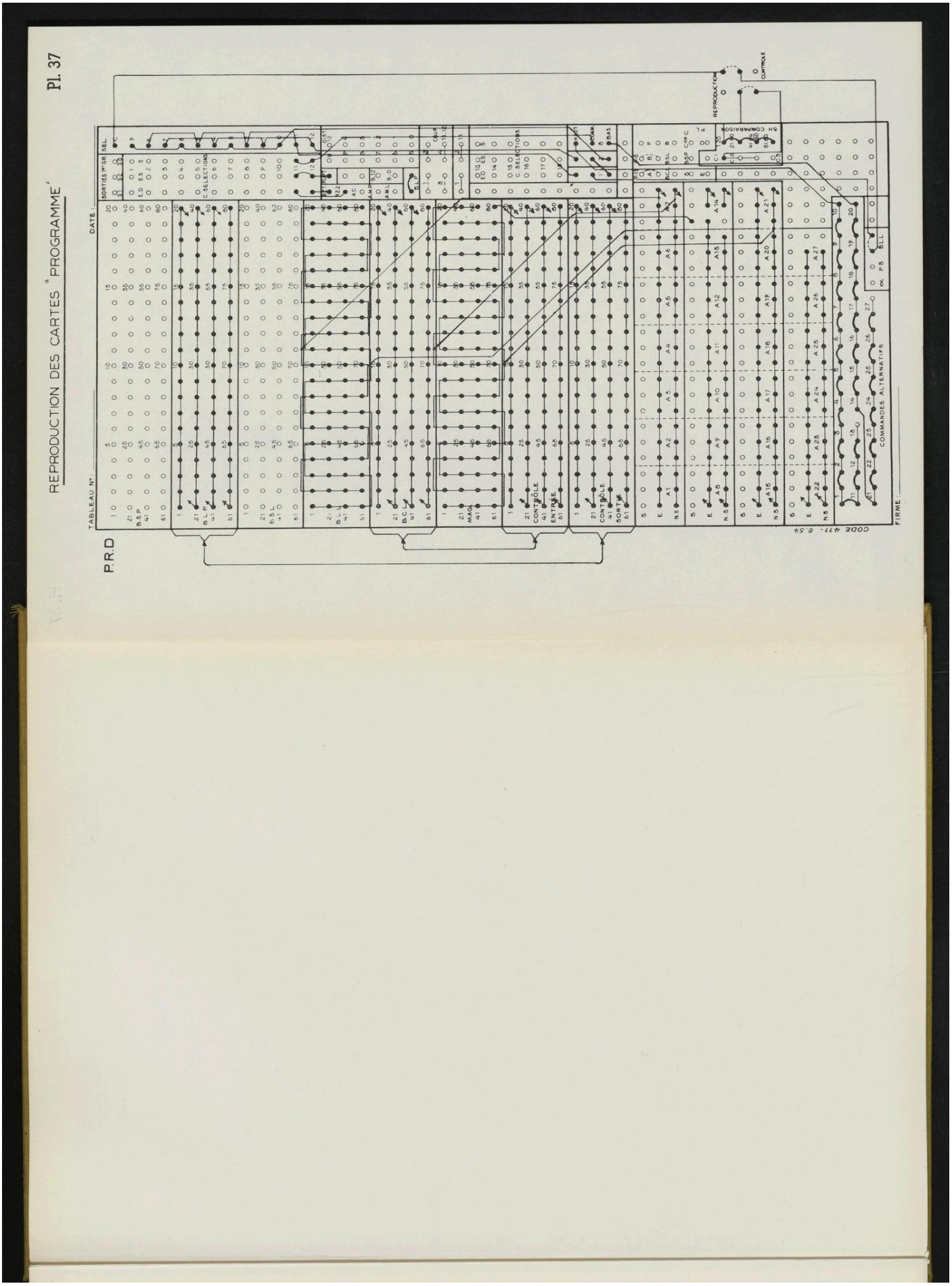
$$a x^2 + b x + c = 0 \qquad x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Remarques	PM	AS	TO	AD	OD	OF	Opération	MD	1	2	3	4	5	6	7	108	109	110	111	112	113	114	115
															x_1		x_2						
CD1											c	b	a										
CD2	9		6	5			BO		a														
			8	6			OB							a									
Ext. x_1			0	0	1	1	Add.		2a					2a									
	7		8	8			OB									2a							
			8	5			OB						2a										
			0	0	1	1	Add		4a					4a									
	5		6	3			BO		c														
			8	5			OB						c										
			0	0	9	1	Mult.		4ac					4ac									
	3		8	3			OB				4ac												
			6	4			BO		b														
			8	5			OB						b										
			8	6			OB							b									
			0	0	9	1	Mult.		b^2					b^2									
	1		8	5			OB						b^2										
			6	3			BO		4ac														
			8	6			OB							4ac									
			0	0	0	1	Soust.		Δ					Δ									
	11		8	5			OB						Δ										
			6	9			BO		x_2														
			8	7			OB								x_2								
CP 3-4		Carles		racine					$\sqrt{\Delta}$					$\sqrt{\Delta}$									
Ext. x_2																							
CP5	9		8	3			OB				$\sqrt{\Delta}$												
			6	0			BO																
			11	4			SN		$-b$														
			8	4			OB					$-b$											
			8	5			OB						$-b$										
			0	0	0	1	Soust.		$-b-\sqrt{}$					$-b-\sqrt{}$									
	7		8	5			OB						$-b-\sqrt{}$										
			6	8			BO		2a														
			8	6			OB							2a									
			0	0	11	9	Div.		x_1					x_1									
	5		8	7			OB								x_1								
			6	4			BO		$-b$														
			8	5			OB						$-b$										
			6	3			BO		$\sqrt{\Delta}$														
			8	6			OB							$\sqrt{\Delta}$									
			0	0	1	1	Add		$-b+\sqrt{}$					$-b+\sqrt{}$									
	3		8	5			OB						$-b+\sqrt{}$										
			6	8			BO		2a														
			8	6			OB							2a									
			0	0	11	9	Div.		x_2					x_2									
	1		8	9			OB										x_2						
			3	3			ZB																
			3	4			ZB																
			3	5			ZB																

Pl. 35

INTRODUCTION DE DONNEES

Remarques	PM	AS	TO	AD	OD	OF	Opération	Arm.	1	2	3	4	5	6	7	108	109	110	111	112	113	114	115	209	209	210	211	212	213	214	215	308	309	310	311	312	313	314	315
RAZ																																							
C.D. 01											A	B	C	D																									
C.P. 02	9	1					Cmd Arm.	1																															
	8		6	3			BO		A																														
			8	8			OB									A																							
	7	2					Cmd Arm.	2																															
	6		6	4			BO		B															B															
			8	8			OB																																
	5	3					Cmd Arm.	3																															
	4		6	5			BO		C																														
			8	8			OB																									C							
			6	6			BO		D																														
			8	9			OB																										D						
	3	1					Cmd Arm.	1																															
	2		3	3			ZB																																
			3	4			ZB																																
			3	5			ZB																																
			3	6			ZB																																
C.D. 03											a_1	a_2	a_3	a_4	a_5																								
C.P. 04	9		6	3			BO		a_1																														
			8	9			OB										a_1																						
			6	4			BO		a_2																														
			8	10			OB											a_2																					
			6	5			BO		a_3																														
			8	11			OB												a_3																				
	8		6	6			BO		a_4																														
			8	12			OB														a_4																		
			6	7			BO		a_5																														
			8	13			OB															a_5																	
	7		3	3			ZB																																
			3	4			ZB																																
			3	5			ZB																																
			3	6			ZB																																
			3	7			ZB																																
	6	2					Cmd Arm.	2																															
CD. 05											b_1	b_2	b_3																										
CP. 06	9		6	3			BO		b_1																														
			8	9			OB																		b_1														
			6	4			BO		b_2																														
			8	10			OB																			b_2													
			6	5			BO		b_3																														
			8	11			OB																				b_3												
	8		3	3			ZB																																
			3	4			ZB																																
			3	5			ZB																																
	7	3					ZB																																
C.D. 07											c_1	c_2	c_3	d_1	d_2																								
C.P. 08	9		6	3			BO		c_1																														
			8	10			OB																										c_1						
			6	4			BO		c_2																														
			8	11			OB																											c_2					
			6	5			BO		c_3																														
			8	12			OB																												c_3				
	8		6	6			BO		d_1																														
			8	13			OB																													d_1			
			6	7			BO		d_2																														
			8	14			OB																														d_2		
	7		3	3			ZB																																
			3	4			ZB																																
			3	5			ZB																																
			3	6			ZB																																
			3	7			ZB																																



SOUS - PROGRAMME P. D. F _ TABLEAU Γ

	Cdes S				Sélection lignes	NL	Codes				Opération	MD	M1	M2	M3	M4	M5	M6	M7
	Nº	S	C	NS			TO	AD	OD	OF			0 10 8 6 4 2	0 10 8 6 4 2	0 10 8 6 4 2	0 10 8 6 4 2	0 10 8 6 4 2	0 10 8 6 4 2	0 10 8 6 4 2 0
4 op.	8	2	A	0		0	0	1	A	B									
		0	C	2		1	C	D											
	9	3	B	0		2	4	5	E	F									
		0	D	3		3	G	H			P.P.C.								
	10	6	E	0		4	8	9	I	J									
		0	G	6		5	K	L											
	11	7	F	0		6	M	N											
		0	H	7		7	P	Q	14	15									
8 op.	12	10	K	0		8		8		3	V.P.M.P'3								
0.0.2.5.		0	I	10	AB	9	6		11	2	B O	11	2						
0.0.2.9	13	11	L	0	AB-C	10	4	2	11	1	K B	11	9						
0.0.2.13		0	J	11	AB+C	11	6	1	11		B O Ax^2+Bx+C	11	9						
	14	12	M	0		12	9	2	11		C N	11		AB >	AB-C <	AB+C =			
		0	P	12		13	6	5	2	11	B O	2	A						
	15	13	N	0		14	8	2	1		O B	1		A					
		0	Q	13		15	6	5	11		B O	11							
						16	14	6	2		M C	0	AB						
						17	15		10	1	D C C	11 ou 10			11 ou 0	11 si 2 décalages 0 si 1 décalage			
						18	6	1	2		B O	2	AB						
						19	8	6	2		O B	2						AB	
						20	6	6		2	B O	0	b						
						21	10	2		1	A N	0	b-b ou 0		(car 1 11 en binaire = -1)				
						22	10	5		2	A N	0	b-a						
						23	11		1	5	S N	0	b-a = 50 = ab						
						24	8	6		2	O B	0							ab
						25	6	6			B O	0	AB	ab					
						26	1	1			V C > Ligne	64							
						27		2	7	12	V C =	0							
0.0.7.1					B+C	28	6				B O	0							
						29	11	7			S N	0	-c						
						30	8	7			O B	0							-c
0.0.7.13					B+C	31	6	6		2	B O	0	b						
						32	9	7		2	C N	0							
						33	11	7		2	S N	0	b-c						
						34	12	1			M R	0	(b-c) = a						
						35	8	2			O B	0			d				
						36	11			8	S N	0	(b-c) = 9						
						37		4	11	1	V M S +	0							
						38	6	6	2		B O	2	B						
						39		3	10	9	V C <	2							
						40	6	7	2		B O	2	C						
						41		2	12		V C =	2							
						42	7				B D	d	Égalité B si B<C						
						43	14				M C	0							
						44	6	1	2		B O	2							
						45		3	12		V C ≥	2							
						46	10	7	2		A N	2	+C si B<C						
						47			12	5	V tjrs	2							
						48	10	6	2		A N	2							
						49	12		2		M R	0	B si C≤B						
						50	2		9		A M D	9			9 - (6-1)				
						51	15		10	1	D C C	5							
						52	6	1	2		B O	2							
						53	8	7	2		O B	2							B+C
						54	6	7		2	B O	0	b						
						55		1	14	5	V C ≤	0							
						56	6	6		2	B O	0	c si b<c	b<c					
						57	10	2		1	A N	0	b ou c+5						
						58	11			8	S N	0	b						
						59	8	7		2	O B	0							B+C
						60	6	7			B O	0	B+C						
						61	9		10	1	C N	0							
						62	1			1	V tjrs	0							
0.0.15.13.					B/C	63	1		3	9	V tjrs	0							

L.64 L.78

Pl. 38

SOUS - PROGRAMME P.D.F. _ TABLEAU A.E.P

L.11 L.50 8 VPM

Cdes S N°	S	C	NS	Sélection lignes	NL	Codes TO	AD	OD	OF	Opération	MD	M1	M2	M3	M4	M5	M6	M7
					0	3	2			ZB	0							
					1	1	3	1		V≥	0							
					2	3	7			ZB	0							
					3	6				BO	0							
A x²+Bx+C4 col 6 sur CD avec 0.0.2.13 en ligne 11	4	0	X	1	4			2	X	Vtjrs/Vms	0							
					5	8	6			OB	0							
					6	6	4			BO	0							
					7	3	4			ZB	0							
					8	8	7			OB	0							
					9	9	4			CN	0							
					10		2	2	13	V≠	0							
					11	6	6			BO	0							
					12	3	6			ZB	0							
					13			2	1	Vtjrs	0							
A/B →					14	6	6		2	BO	0							
					15	10		1	5	AN	0							
					16	11	7		2	SN	0							
					17	8	7		2	OB	0							
					18	6	6	2		BO	2							
					19	9	1			CN	0							
					20	15	7	2		DC	0							
					21	6	1	11		BO	11							
					22	15		10	1	DCC	d							
					23			12	9	Vtjrs	d							
					24	9		1	2	CN	0							
					25	6	6			BO	0							
					26	1	2	7	9	VC≠	0							
					27	11	2			SN	0							
					28	4	3	1	1	KB	0							
					29	1		8	1	Vtjrs	0							
					30	10	2			AN	0							
					31	4	3	1		KB	0							
					32	8	6			OB	0							
sin. cos. } →					33	6	3	9	11	BO	9							
					34	9	3	2	4	CN	2							
					35		2	2		VC =	2							
					36													
SINUS					37	10		2	1	AN	2							
					38	8	3	9	11	OB	9							
					39	6				BO	0							
					40	14	4			MC	0							
					41	15	3	9	11	DC	0							
					42	6	3	1	2	BO	1							
					43	9		1	1	CN	1							
					44	1	1	6		VC >	1							
					45	6	7			BO	0							
					46	1	2	12	8	VC =	0							
S1 collé par carte V (5 col. 6) →	1	6	A	4	47	11	2			SN	0							
	"	6	B	3	48	A	B	C	D									
	"	0	C	1	49	E	F	G	C									
	"	0	D	2	50	H	2	I										
	"	15	E	1	51	J	K	N	L									
	"	7	F	0	52	8	M	I										
	"	2	G	13	53	O		P	5									
	"	6	H	10	54	12	2	2		MR	0							
	"	2	I	0	55	2		2		AMD	2							
	"	10	J	4	56	11	7	2		SN	2							
	"	7	K	3	57	9		4	1	CN	2							
	"	0	L	3	58	10	7	2		AN	2							
	"	2	M	7	59	8	7	2		OB	2							
	"	2	N	1	60	1	1	12		VC >	2							
	"	6	O	1	61	6	7			BO	0							
	"	1	P	8	62	3	2			ZB	0							
	"	0	Q	10														
					63		Q	2	1	VPM 0,3	0							

Pl. 39

CARTES SIN. x COS. x

REDUCTION DE x EN RADIANS : $-2\pi < x < +2\pi$

REDUCTION AU 1er QUADRANT

Cdes S				Sélection lignes	NL	Codes				Opération	MD	M1	M2	M3	M4	M5	M6	M7
Nº	S	C	NS			TO	AD	OD	OF			0 10 8 6 4 2 0	10 8 6 4 2 0	10 8 6 4 2 0	10 8 6 4 2 0	10 8 6 4 2 0	10 8 6 4 2 0	10 8 6 4 2 0
						3	7											
					9	4	7	10	6									
						4	7	9	2									
						4	7	8	8									
						4	7	7	3									
					8	4	7	6	1									
						4	7	5	8									
						4	7	4	5									
						4	7	3	3									
					7	4	7	1	5									
						4	7		1									
						6	7											
						8	3											
					6	4	3	1	7									
								15	13	A/B								
					5													
						6	7		2									
					4	11		1	5									
							4	2	1	VMS+								
						3	3											
						8	2		1									
					3	6	7	2	11									
						8	2	3										
						6												
						9	3											
					2	7												
						15		10	1									
						8	2											
						6	6	11										
					1	8	3	11										
						6		9										
						14	3											
						2		9										
					0	15		10	1									
						2		2										
						8	7	2										
						6												
					11		2	1	12	VC =								
						4	1	1	5									
						10	2		1									
						8	7		2									
					12													
								7	1	A-B								
						8	6											
					9	3	4											
						4	4	10	1									
						4	4	9	5									
						4	4	8	7									
					8	4	4	6	7									
						4	4	5	9									
						4	4	4	6									
						4	4	3	3									
					7	4	4	2	2									
						6	6	11										
						8	3	11										
						6	6		2									
					6	11		1	4									
						11			2									
						8	2		1									
							4	1	1	VMS+								
					5	3	6											
						6	6	2	11									
						8	2	4										
						3	1	1										
					4	7												
						15		10	1									
						2		1										

Pl. 40

CARTES SIN. x COS. x

	Codes S				Sélection lignes	NL	Codes				Opération	MD	M1	M2	M3	M4	M5	M6	M7
	Nº	S	C	NS			TO	AD	OD	OF			0 10 8 6 4 2 0	10 8 6 4 2 0	10 8 6 4 2 0	10 8 6 4 2 0	10 8 6 4 2 0	10 8 6 4 2 0	10 8 6 4 2 0
REDUCTION AU 1er QUADRANT							15	4											
						3	8	4											
							6	2		1									
							8	3		11									
							6	4	1										
						2	8	2											
							8	6											
							8	4											
							3	7											
						1	4	7	9	1									
							4	3	9	1									
							4	3	2	4									
							4	3	3	1									
						0	1		8	5	VS tjrs								
1 CV ITERATIONS																			
NORMALISATION SIGNES AVEC ZEROS A DROITE							6	6											
						7	2		9										
							15		10	1									
							2												
							4	1		2									
						6	4	1	1	4									
							10	2		1									
							8	6											
							6	7											
						5	2		9										
							15		10	1									
							2												
							4	1		2									
						4	4	1	1	4									
							10	2		1									
							8	7											
							4	2		5									
						3	6		1										
							14	3		1									
							6	2	11										
							9		11										
						2	6	6											
							8	2											
							6	7											
								2	2		VC =								
						1	8	6											
							6	2											
							8	7											
							6	3		1									
						0	8	2											
							4	1	8	13									
							7												
							8	4											
						11	6	4	9	10									
							6	1	11										
							8	7	11										
							6	4	8	9									
						12	12	3	11										
							6	1	11										
							8	6	11										

Pl. 41