

M0019/87A

UNIVERSITE DES SCIENCES ET DE LA TECHNOLOGIE

HOUARI BOUMEDIENE ( U.S.T.H.B ) ALGER

INSTITUT DE PHYSIQUE

T. H E S E

المدرسة الوطنية المتعددة الفتيات  
BIBLIOTHEQUE — المكتبة  
Ecole Nationale Polytechnique

Présenté à l'U.S.T.H.B pour l'obtention du grade de

M A G I S T E R

Spécialité : Electronique des systèmes

par

M<sup>lle</sup> Nadia SIDAHMED

REALISATION D'UN MICROSYSTEME ET DEVELOPPEMENT DU LOGICIEL  
POUR LA COMMANDE ET L'EXPLOITATION D'UN  
CALCULATEUR HYBRIDE

Soutenue publiquement le 17 Juin 1987  
devant le jury composé de

LARIBI	Sid Ahmed	PROFESSEUR	U.S.T.H.B	PRESIDENT
SANSAL	Boualem	PROFESSEUR	U.S.T.H.B	RAPPORTEUR
OUIGUINI	Rachid	Chargé de RECHERCHE	H.C.R	EXAMINATEUR
TOUMI	Redouane	M. de CONFERENCE	U.S.T.H.B	"
KACIMI	Messacud	Chargé de COURS	U.S.T.H.B	"

UNIVERSITE DES SCIENCES ET DE LA TECHNOLOGIE

HOUARI BOUMEDIENE ( U.S.T.H.B ) ALGER

INSTITUT DE PHYSIQUE

T H E S E

المدرسة الوطنية المتعددة التقنيات  
المكتبة — BIBLIOTHEQUE  
Ecole Nationale Polytechnique

Présenté à l'U.S.T.H.B pour l'obtention du grade de

M A G I S T E R

Spécialité : Electronique des systèmes

par

M<sup>lle</sup> Nadia SIDAHMED

REALISATION D'UN MICROSYSTEME ET DEVELOPPEMENT DU LOGICIEL  
POUR LA COMMANDE ET L'EXPLOITATION D'UN  
CALCULATEUR HYBRIDE

Soutenue publiquement le 17 Juin 1987  
devant le jury composé de

LARIBI	Sid Ahmed	PROFESSEUR	U.S.T.H.B	PRESIDENT
SANSAL	Boualem	PROFESSEUR	U.S.T.H.B	RAPPORTEUR
OUIGUINI	Rachid	Chargé de RECHERCHE	H.C.R	EXAMINATEUR
TOUMI	Redouane	M. de CONFERENCE	U.S.T.H.B	"
KACIMI	Messaoud	Chargé de COURS	U.S.T.H.B	"

## REMERCIEMENTS

المدرسة الوطنية المتعددة التقنيات  
BIBLIOTHEQUE — المكتبة  
Ecole Nationale Polytechnique

Je presente l'expression de ma profonde reconnaissance a Monsieur le professeur SA. LARIBI pour m'avoir fait l'honneur d'accepter la presidence du jury de cette these.

Le present travail a ete concretise au laboratoire ROBOTIQUE du CDTA, que Monsieur A. BESSALA directeur du CDTA trouve ici l'expression de ma profonde gratitude.

J'exprime toute ma reconnaissance a Monsieur le professeur B. SANSAL pour m'avoir accueillie au sein du laboratoire electronique des systemes et pour m'avoir dirigees et encouragees tout au long de ce travail.

Mes plus vifs remerciements vont a Monsieur R. OUIQUINI responsable du laboratoire ROBOTIQUE du CDTA pour m'avoir aidee et encouragee tout au long de ce travail et pour tous les moyens qu'il a mis a ma disposition pour concretiser ce travail.

Cette etude a pu etre menee a bien grace aux nombreux conseils et a l'aide amicale de Monsieur le professeur R. TOUMI auquel j'adresse mes sincereres remerciements.

Je remercie tres chaleureusement Monsieur M. KACIMI pour m'avoir fait l'honneur d'accepter de participer au jury de cette these.

Je remercie vivement Mademoiselle A. PELIPA et Monsieur F. GUETTACHE pour l'aide immense et precieuse qu'ils m'ont pas manque de m'apporter.

Je remercie particulierement Mr M. BOUARAB pour son aide amicale et sa participation au projet.

Enfin , mes remerciements vont au personnel du laboratoire Robotique pour tout le soutien qui m'a été apporté.

**INTRODUCTION**

**CHAPITRE I : CALCULATEURS ANALOGIQUES ET CALCULATEURS  
NUMERIQUES.**

**I.1 - CALCULATEURS ANALOGIQUES.**

**I.1.1 INTRODUCTION**

**I.1.2 PRINCIPE DES CALCULATEURS ANALOGIQUES**

**I.1.3 STRUCTURE DES CALCULATEURS ANALOGIQUES**

**I.1.4 MODES ELEMENTAIRES DE CALCUL**

**I.1.5 DEROULEMENT DU CALCUL ANALOGIQUE**

**I.1.6 APPLICATIONS DES CALCULATEURS ANALOGIQUES**

**I.1.7 ETAPES DE RESOLUTION DU CALCUL**

**I.1.8 CONCLUSION**

**I.2 - CALCULATEURS NUMERIQUES**

**I.2.1 INTRODUCTION**

**I.2.2 LANGAGES DE SIMULATION**

**I.2.3 STRUCTURE GENERALE D'UN PROGRAMME DE  
SIMULATION**

**I.2.4 TECHNIQUES DE PROGRAMMATION**

**I.2.5 SIMULATION INTERACTIVE**

**I.2.6 ARCHITECTURES PARALLELES**

**I.2.7 CONCLUSION**

**CHAPITRE II : CALCULATEURS HYBRIDES**

**II.1 - INTRODUCTION**

**II.2 - BREF HISTORIQUE**

**II.3 - CALCUL HYBRIDE ET CALCULATEUR HYBRIDE**

**II.4 - FORMES SPECIALES DE CALCULATEURS HYBRIDES**

II.5 - ORGANISATION GENERALE

II.6 - LOGICIEL D'UN CALCULATEUR HYBRIDE

II.7 - DOMAINES D'APPLICATIONS

II.8 - CONCLUSION

CHAPITRE III : PRESENTATION DU CALCULATEUR HYBRIDE

III.1 - INTRODUCTION

III.2 - STRUCTURE MATERIELLE

III.2.1 DESCRIPTION DU CHAMP ANALOGIQUE

III.2.2 INTERFACE HYBRIDE

III.2.3 CHAMP NUMERIQUE

III.3 - STRUCTURE LOGICIELLE

III.3.1 INTRODUCTION

III.3.2 STRUCTURE ET FONCTIONNEMENT DU MONITEUR

III.4 - INTERACTION HOMME MACHINE ET DEFINITION  
DES TACHES

III.4.1 INTERACTION HOMME MACHINE

III.4.2 DEFINITION DES TACHES

III.4.3 FONCTIONNEMENT DYNAMIQUE DU SYSTEME

CHAPITRE IV : APPLICATIONS ET RESULTATS

CONCLUSION

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

---

L'étude et l'analyse de la dynamique des systèmes sont des domaines importants dans l'industrie et dans diverses disciplines scientifiques, économiques ...

Ces études s'avèrent la plupart du temps délicates sur le système lui même pour des raisons de sécurité, de coût et de temps.

La simulation présente de grands avantages car elle permet l'étude de ces systèmes de façon plus rapide et plus économique.

De plus, elle permet une analyse de l'influence des divers paramètres de façon aisée.

On distingue plusieurs types de calculateurs électroniques permettant la simulation du comportement dynamique des systèmes :

- le calculateur analogique : c'est le premier calculateur électronique utilisé pour la simulation.

Du fait de sa structure, il est bien adapté à la simulation de processus physiques continus dont le comportement est régi par un système d'équations différentielles linéaires ou non linéaires. C'est une machine à traitement parallèle. La vitesse de calcul des opérateurs analogiques permet la simulation en temps réel de phénomènes rapides.

L'un de ces inconvénients est le manque de précision.

- Le calculateur numérique conventionnel : la diminution du temps de cycle CPU ainsi que la diminution du cycle

d'accès en mémoire centrale ont permis d'aboutir à une simulation numérique moins coûteuse en temps. Le développement de langages de haut niveau a en outre permis de rendre la programmation plus aisée en simulation numérique.

- Les calculateurs numériques spécialisés : pour répondre aux exigences en calculs dans certains domaines (météorologie, sismologie ...) il s'est avéré nécessaire d'augmenter les performances des calculateurs numériques conventionnels en élaborant de nouvelles architectures.

Ceci peut être obtenu par exemple en augmentant le nombre de processeurs de calcul.

- Le calculateur hybride : ce calculateur est issu du couplage d'un calculateur numérique et d'un calculateur analogique à l'aide d'une interface hybride.

Ce type de calculateurs a été conçu vers la fin des années 50 pour des applications spécifiques à l'aérospatial, applications pour lesquelles les calculateurs analogiques et les calculateurs numériques conventionnels présentaient des limitations.

Les applications traitées par ce type de calculateurs se sont très vite étendues à d'autres domaines tels que nucléaire, biomedical etc ...

Ils sont utilisés pour l'identification de processus, la modélisation, l'optimisation ... (13).

Le développement des microprocesseurs à faible coût a favorisé l'émergence de calculateurs hybrides dans lesquels le calculateur numérique est remplacé par un

microcalculateur : on obtient alors des calculateurs hybrides a microprocesseur.

Dans ce type de configuration, le role du microcalculateur est lie aux performances du microprocesseur choisi. Ce role serait soit d'assurer la gestion du calculateur hybride ou alors tout en assurant cette gestion, il peut intervenir dans le calcul.

L'amelioration des performances des circuits analogiques ainsi que le developpement des architectures a multiprocesseurs ont ete mis a profit pour l'elaboration de calculateurs hybrides a multimicroprocesseurs. Ces calculateurs sont utilises en particulier en tant que simulateurs de vol (34).

L'objet de notre travail porte sur le developpement d'un calculateur hybride a microprocesseur en vue de son utilisation pour la simulation de processus physiques continus.

Nous disposons de la partie analogique d'un calculateur hybride. Nous nous sommes interessees a la conception et la realisation de la partie numerique de ce calculateur. Celle-ci est concue a base d'un microprocesseur 8 bits.

Nous avons egalement developpe le logiciel permettant la commande et l'exploitation du calculateur hybride.

Le plan de notre travail est structure de la facon suivante :

- Dans un premier chapitre nous decrivons les calculateurs analogiques ainsi que les calculateurs numeriques qui sont parties integrantes des calculateurs



hybrides. Nous verrons les modes de programmation de chacun de ces calculateurs.

- Le second chapitre est consacré à la description des calculateurs hybrides et à la programmation de ces derniers.

- Le chapitre trois est destiné :

+ D'une part à la description des différentes parties de notre calculateur.

- La partie analogique déjà existante

- Le microsysteme et l'interface analogique que nous avons réalisés.

+ D'autre part, on décrira le logiciel que nous avons développé pour la gestion et l'exploitation du calculateur hybride. Nous avons traité deux exemples d'applications permettant de mettre en oeuvre l'ensemble des programmes constituant ce logiciel.

- Les résultats de ces deux applications font l'objet du chapitre quatre. L'analyse de ces résultats nous a permis de confirmer le bon fonctionnement du calculateur.

- C H A P I T R E I -

I.1 CALCULATEURS ANALOGIQUES

---

I.1.1 Introduction

---

L'etude de la dynamique des systemes physiques a toujours ete un facteur tres important dans le developpement industriel.

Cependant, pour des raisons de securite et de cout, cette etude est rarement menee sur le systeme lui meme.

C'est ainsi que la notion de simulation est apparue. Celle-ci est basee sur le principe des analogies. Il existe differents types d'analogies (22) :

- L'analogie electrique
- L'analogie mecanique
- L'analogie hydraulique
- L'analogie pneumatique

On dit que deux phenomenes totalement differents sont analogues lorsqu'ils sont caracterises par le meme ensemble d'equations (5) (12).

De ce fait la connaissance de l'evolution d'un phenomene entraine la connaissance de l'evolution des phenomenes analogues au premier.

Le choix d'une des analogies est dicte par des considerations pratiques.

L'experience menee sur le modele analogue doit permettre une etude rapide, commode, economique et precise du systeme reel.

Parmi les analogies citees precedemment, l'analogie electrique a toujours ete la plus utilisee.

Ceci vient du fait que le domaine electrique est tres accessible a l'experience (14).

### I.1.2 Principes des calculateurs analogiques a courant ----- continu : -----

On fait correspondre au systeme a etudier, un systeme electrique caracterise par des courants et des tensions continus (12).

Ceux ci sont regis par les memes equations que celles qui definissent les relations entre les grandeurs du systeme reel.

Le calculateur analogique est forme d'un ensemble de blocs operateurs elementaires effectuant chacun une ou plusieurs operations mathematiques sur des tensions continues.

Chaque operation est effectuee par rapport a une seule variable independante : le temps.

Les operations effectuees par les operateurs d'un calculateur analogique sont variees : addition, integration, multiplication, generation de fonction a une seule variable ...

Ces operateurs sont selectionnes et interconnectes de facon que les equations ainsi cablees soient les memes que celles decrivant le comportement dynamique du systeme a etudier (11) (18).

Les operateurs sont assembles au niveau du panneau de cablage ce qui permet d'obtenir le systeme analogue. Le calcul analogique est un calcul parallele car les differentes operations sont executees simultanement.

C'est la caracteristique la plus importante de ce calculateur. Le calculateur analogique est particulierement bien adapte a la resolution des equations differentielles (11) (12) (18) ainsi que des systemes differentiels d'ordre eleve.

Les relations entre les grandeurs caracterisant un grand nombre des systemes physiques se traduisent par des equations differentielles.

Pour cette raison, l'utilisation de ce calculateur est preponderante dans les etudes de la dynamique des systemes physiques sans etre limitee a cette classe de problemes. C'est le calculateur analogique qui a permis le developpement de la simulation (1).

Il a donne un sens pratique a l'elaboration des modeles d'equations differentielles.

### I.1.3 Description des calculateurs analogiques

#### I.1.3 a Element de base

L'amplificateur est l'organe essentiel du calculateur analogique.

C'est l'element de base d'un grand nombre d'operateurs. Associe a un ensemble de resistances ou de

condensateurs, il permet les operations telles que :  
sommation, integration. On peut classer les differents  
operateurs en deux groupes :

### I.1.3 b Operateurs lineaires

---

Ces operateurs permettent de faire :

- La somme ou l'integrale de grandeurs evoluant en fonction du temps.
- Le produit par une constante numerique d'une grandeur evoluant en fonction du temps.

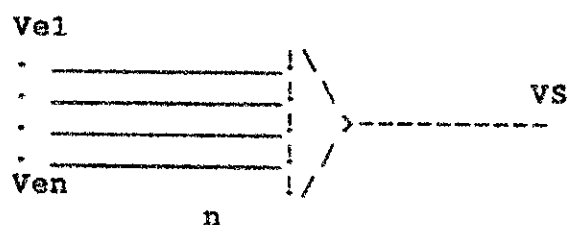
Ces operateurs sont suffisants pour resoudre des equations differentielles lineaires a coefficients constants.

Symbole de ces operateurs

---

#### 1 - Sommateur

---



$$VS(t) = - \sum_{i=1}^n K_i V_{ei}(t)$$

$K_i = 1$  ou  $10$

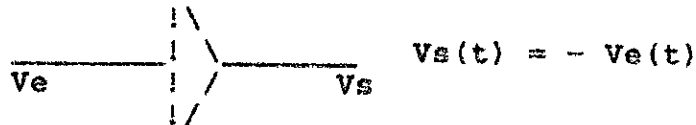
Les valeurs de  $K_i$  dependent des elements associes a l'amplificateur de base.

2 - L'inverseur

L'inverseur est un cas particulier du sommateur.

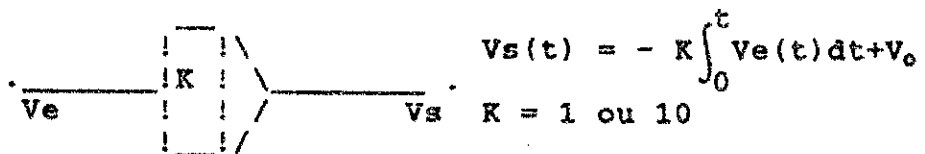
Le nombre de tensions d'entrees est reduit a un.

Symbole



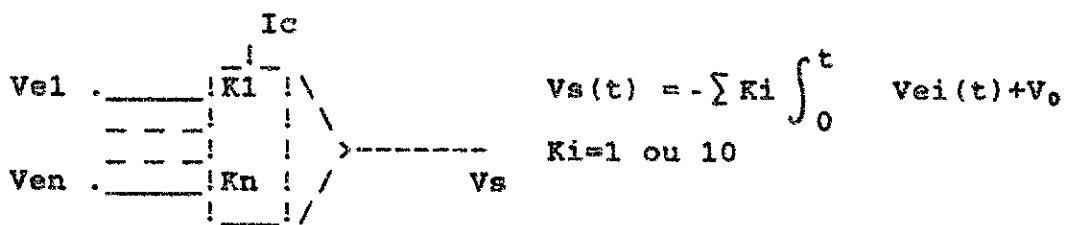
3 - L'integrateur

a) Symbole de l'integrateur a une entree.



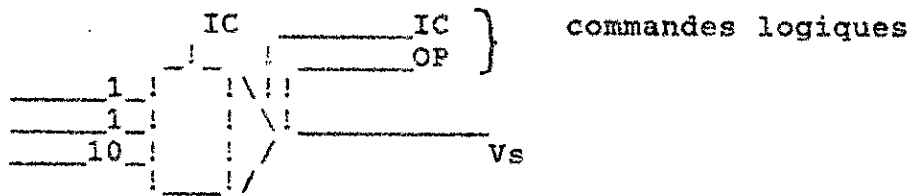
Ic est une tension analogique

b) Symbole de l'integrateur sommateur



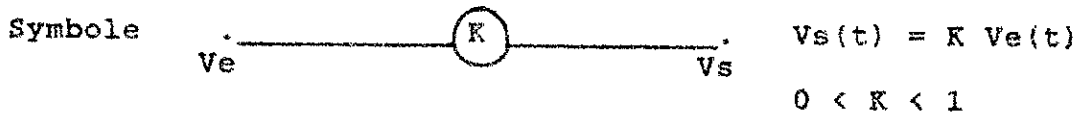
L'integrateur est l'element de calcul le plus important. Nous verrons le role qu'il joue dans le calcul plus loin.

En plus des entrees analogiques, il possede deux entrees logiques IC et OP.



4 - Le potentiometre

---

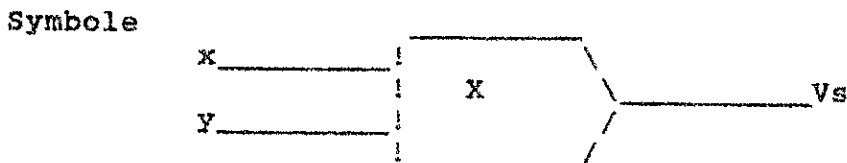


I.1.3 c Operateurs non lineaires

---

Ces operateurs permettent de faire :

- Le produit et la division de deux grandeurs
  - La generation d'une fonction du temps
  - Les multiplieurs
- 



On peut avoir les deux cas suivants

$$V_s = axy$$

$$V_s = xy$$

- Les generateurs de fonctions
- 



Nous distinguons deux types de generateurs de fonctions :

1 - Les generateurs de fonctions universels :

Ces generateurs generent une fonction  $y = f(x)$  quelconque

2 - Les generateurs de fonctions specialises :

Ils generent les fonctions suivantes : sinus, cosinus, carres ... etc

On trouve aussi :

3 - Des generateurs de bruit blanc :

Utiles dans le cas d'analyse des signaux.

- Des generateurs permettant la conversion de coordonnees cartesiennes  $(x,y)$  en coordonnees polaires  $(R,\theta)$
- Des generateurs de retard variable.

Il n'est pas tres aise de realiser des retards variables de maniere purement analogique.

I.1.3 d Operateurs logiques ( ou logique parallele)

Les calculateurs analogiques ont beneficie de l'evolution de la technologie des composants de base : (miniaturisation, augmentation de la fiabilite et de la bande passante).

Cependant, c'est l'apparition d'une logique parallele qui a permis une plus grande souplesse dans la resolution de certains problemes.



En effet, une limitation des calculateurs analogiques decrits precedemment est apparue dans le cas de problemes faisant intervenir des calculs iteratifs rapides.

Ex : problemes d'optimisation des parametres ... (10)

L'automatisation de ces calculs fut obtenue en ajoutant des elements logiques qui assurent la commande du calcul (22). Les calculateurs ainsi obtenus sont appeles "iterative differentiel analyser" ou calculateur analogique iteratif (IDA) (10) .

Ex : Le calculateur EAI 180 (22)

Les operateurs logiques sont :

- 1 - Les elements sans memoire : portes logiques, inverseurs.
- 2 - Les elements a memoire : bascules, registres a decalages, compteurs.

#### I.1.4 Modes de calcul du calculateur analogique

##### I.1.4 a Etats ou modes possibles des operateurs de calcul

La resolution d'un systeme d'equations sur calculateur analogique necessite le passage par des phases differentes. Ces phases correspondent aux etats possibles des operateurs de calcul.

On distingue quatre etats ou modes possibles :

- trois etats sont directement lies a la commande des entrees logiques IC et OP des integrateurs. On les appelle mode de fonctionnement (22).
- Un etat correspondant a l'etalonnage des potentiometres .

## A - Modes de fonctionnement

---

### 1) Mode conditions initiales (ou mode IC)

---

Ce mode permet l'introduction des conditions initiales du système d'équations à résoudre.

Ces conditions initiales sont introduites à travers le chargement des capacités des intégrateurs comme le montre la figure I.1.

La capacité est chargée avec une constante de temps  $RC$  à la valeur imposée au niveau de la borne d'entrée CI.

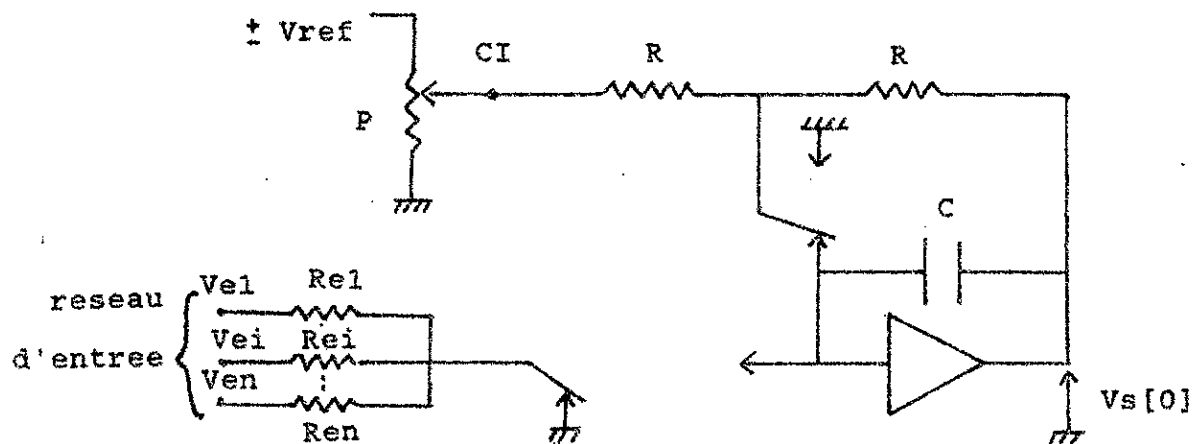


fig I.1

### 2) Mode opératoire ou calcul (mode OP)

---

Dans ce cas, les tensions représentant les variables du système d'équations sont appliquées aux entrées des intégrateurs.

Le système électrique analogique évolue suivant sa dynamique propre, fournissant la solution du système à résoudre.

La commande des integrateurs est representee fig I.2

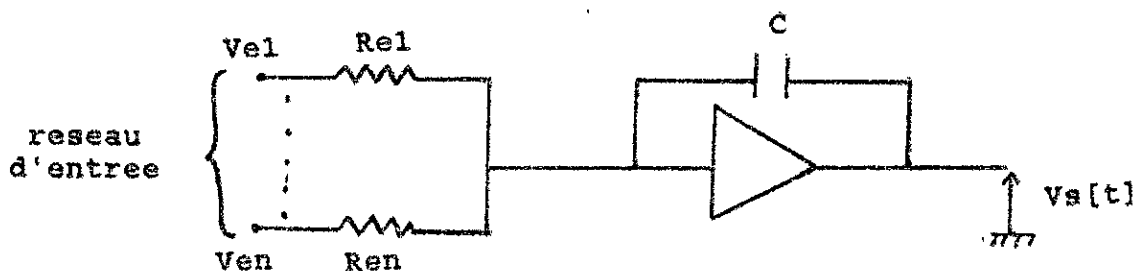


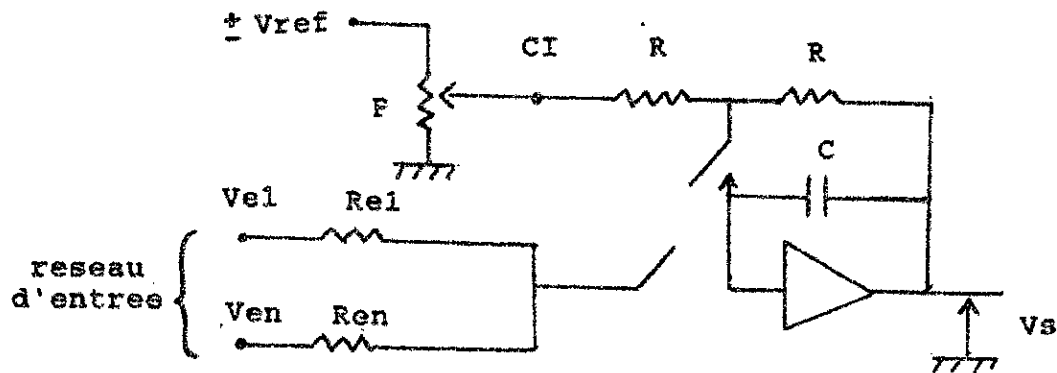
fig I.2

3) Mode memoire ou gel (mode HD)

Il est possible de suspendre le deroulement d'un calcul en le figeant dans l'etat ou il se trouve.

Ce mode permet la lecture de toutes les valeurs prises par les variables a cet instant.

La commande des integrateurs est representee fig I-3.



figI-3

Ces trois modes sont rendus possibles par l'intermediaire de signaux logiques fournis par les operateurs logiques.

Nous pouvons resumer ces trois modes par le tableau suivant :

Entree IC	Entree OP	Mode
0	1	OP
1	0	IC
0	0	non utilise
1	1	HD

B - Mode etalonnage des potentiometres (mode PS)  
-----

C'est aussi le mode arret.

Avant de lancer un calcul, il est necessaire d'introduire les differentes valeurs des parametres du systeme d'equations a resoudre.

Ceci est realise par reglage des potentiometres.

Dans ce mode, les entrees de tous les amplificateurs sont a la masse pour eviter qu'ils se saturent.

Seules les entrees des potentiometres sont connectees a la tension de reference.

La lecture des tensions de sortie des potentiometres permet alors de fixer les valeurs des coefficients.

En plus du reglage de ces coefficients, un certain nombre d'autres reglages peuvent s'averer necessaires.

- Affichage des generateurs de fonctions.
- Selection des constantes d'integration
- Initialisation de la logique parallele :  
bistable, registre, horloge ...

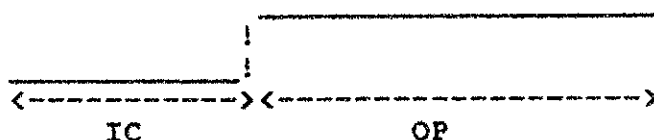
### I.1.5 Modes de déroulement du calcul analogique

---

Suivant la façon dont se succèdent les modes de fonctionnement on distingue trois modes de calcul possibles :

#### I.1.5 a Mode normal

C'est la succession des modes IC et OP selon le chronogramme suivant :



Dans ce mode on distingue deux phases.

#### - Phase IC :

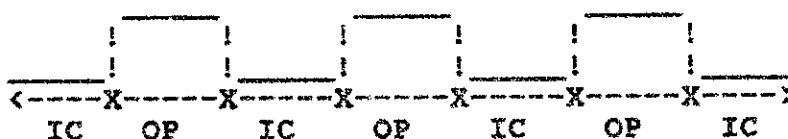
---

Celle-ci correspond à l'introduction des conditions initiales. Ceci est effectuée après avoir ajusté les valeurs de tous les paramètres intervenant dans le calcul. Lorsque les conditions initiales sont chargées (durée de IC) on passe à la phase calcul.

Durant cette phase, les opérateurs intervenant dans le calcul sont connectés entre eux, et les tensions évoluent de la même façon que les grandeurs physiques du système à étudier.

#### I.1.5 b Mode répétitif

Chronogramme



Certains problemes necessitent la repetition automatique des calculs.

Ceci est realise par la succession rapide des modes IC et OP.

La logique parallele est programme de sorte a commander la repetition de ces phases.

Pendant la duree IC, les conditions initiales sont rechargees et pendant la duree OP, les operateurs remplissent leur fonction de calcul avec des conditions initiales modifiees ou non.

Les resultats peuvent etre suivis sur oscilloscope.

L'interet de ce mode de calcul est de permettre l'observation immediate de l'effet de la variation d'un parametre sur la solution.

Les calculateurs analogiques les plus perfectionnes sont dotes d'une programmation automatique de modification d'un parametre.

#### I.1.5 c Mode iteratif

Chronogramme



La difference entre ce mode de calcul et le precedent vient du fait que la phase OP utilise les resultats du calcul precedent. Ceux-ci sont stockes durant la phase HD qui ne figure pas dans le mode repetitif.

Le mode iteratif permet la convergence du calcul vers la solution desiree.

#### I.1.6 Applications des calculateurs analogiques

La classe de problemes que l'on peut cabler sur calculateur analogique est celle relevant d'une formulation qui se presente sous forme d'un systeme d'equations comprenant :

- Des equations differentielles ordinaires (lineaires ou non).
- Des relations algebriques
- Eventuellement des relations logiques (21)

Ce sont des systemes d'equations a une seule variable independante.

Cependant, les calculateurs analogiques ne se limitent pas uniquement a ce type de problemes.

En effet, ils permettent la resolution de systemes d'equations aux derivees partielles.

Ces equations presentent des difficultes du fait que l'on ne dispose sur le calculateur analogique que d'une seule variable independante d'integration : le temps machine.

Certaines methodes permettent de pallier a cette difficulte :

On peut en citer quelques unes :

- Methode des differences (5)
- Methode des caracteristiques
- Methode des fonctions propres ...

Les calculateurs analogiques ne sont pas très adaptés à la résolution des équations algébriques. En effet, on constate dans les boucles algébriques des instabilités entraînant une divergence de la solution (5) (10) (21). On essaie de faire disparaître ces phénomènes en disposant des constantes de temps dans les boucles de calcul.

#### I.1.7 Etapes de résolution sur calculateur analogique

(ou programmation du calculateur analogique)

Le traitement sur calculateur analogique nécessite le passage par un certain nombre d'étapes constituant la programmation du calculateur.

Avant d'aborder cette programmation deux étapes sont nécessaires :

1) Définition du modèle mathématique du phénomène à étudier en alliant connaissance et expérience. Ce modèle est constitué d'un système d'équations accompagné d'une liste de coefficients, de paramètres ainsi que de résultats de mesure.

2) Transformations de ce modèle sous forme assimilable par la machine c'est à dire permettant de le mettre sous forme canonique.

Les transformations effectuées ne doivent évidemment pas affecter le modèle.



## Programmation du calculateur

---

A ce niveau, nous disposons d'un système d'équations canoniques :

$$\frac{dx_i}{dt} = f_i(x_1, x_2, \dots, x_n) \\ i = 1, 2, \dots, n$$

### 1) Ecriture des équations machines

---

On établit une correspondance entre les unités physiques et les unités machines (volts et secondes).

Ceci nous permet de transformer le système d'équations initial en équations machines.

Pour cela, nous devons effectuer les mises à l'échelle suivantes :

#### a) Mise à l'échelle en amplitude

---

Les variables du problème sont représentées sur la machine par les tensions de sortie des amplificateurs.

Pour éviter la saturation de ces derniers, la connaissance des valeurs maximales de chacune des variables  $X_i$  est nécessaire

#### b) Mise à l'échelle en temps (ou choix du temps machine)

---

La variable indépendante des équations canoniques peut représenter le temps, l'espace ou toute autre grandeur. On lui fait correspondre le temps machine par la relation  $\tau = \beta t$

$\beta$  est appelé facteur d'échelle

- Si l'on veut effectuer une étude du phénomène en temps réel on prend  $\beta = 1$
- Si le phénomène à étudier est trop rapide, on a la possibilité de le ralentir en prenant  $\beta > 1$
- Dans le cas de processus trop lents, il est nécessaire d'accélérer le phénomène pour effectuer son étude, ceci est possible en prenant  $\beta < 1$ .

On aboutit ainsi aux équations machine

## 2 ) Etablissement du schéma analogique

On dessine le schéma de principe représentant ces équations en utilisant les symboles des blocs opérateurs:

- Les variables seront élaborées par des amplificateurs
- Les constantes par des potentiomètres

3 ) En s'aidant de ce schéma, on réalise le câblage sur le panneau de la machine

L'avantage de cette programmation vient du fait que le manipulateur ne perd pas de vue le sens physique du phénomène qu'il étudie.

## 4 ) Calcul du contrôle statique

On procède à un certain nombre de calculs permettant le contrôle statique.

Pour cela on choisit au préalable un lot de conditions initiales arbitraires et un nombre de valeurs de variables d'entrée permettant de fournir des sorties non saturées.

On calcule a l'aide des equations machines les grandeurs de sortie de tous les operateurs concernes en les supposant parfaits.

5 ) Essais statiques  
-----

On procede aux verifications statiques en utilisant les resultats precedants.

Pour eviter des problemes d'instabilite, les boucles algebriques seront ouvertes.

6 ) Controle dynamique  
-----

Un controle dynamique permet la verification des constantes d'integration.

7 ) Mise en conditions initiales  
-----

On ajuste toutes les valeurs des conditions initiales a leur valeur reelle.

8 ) Passage en calcul  
-----

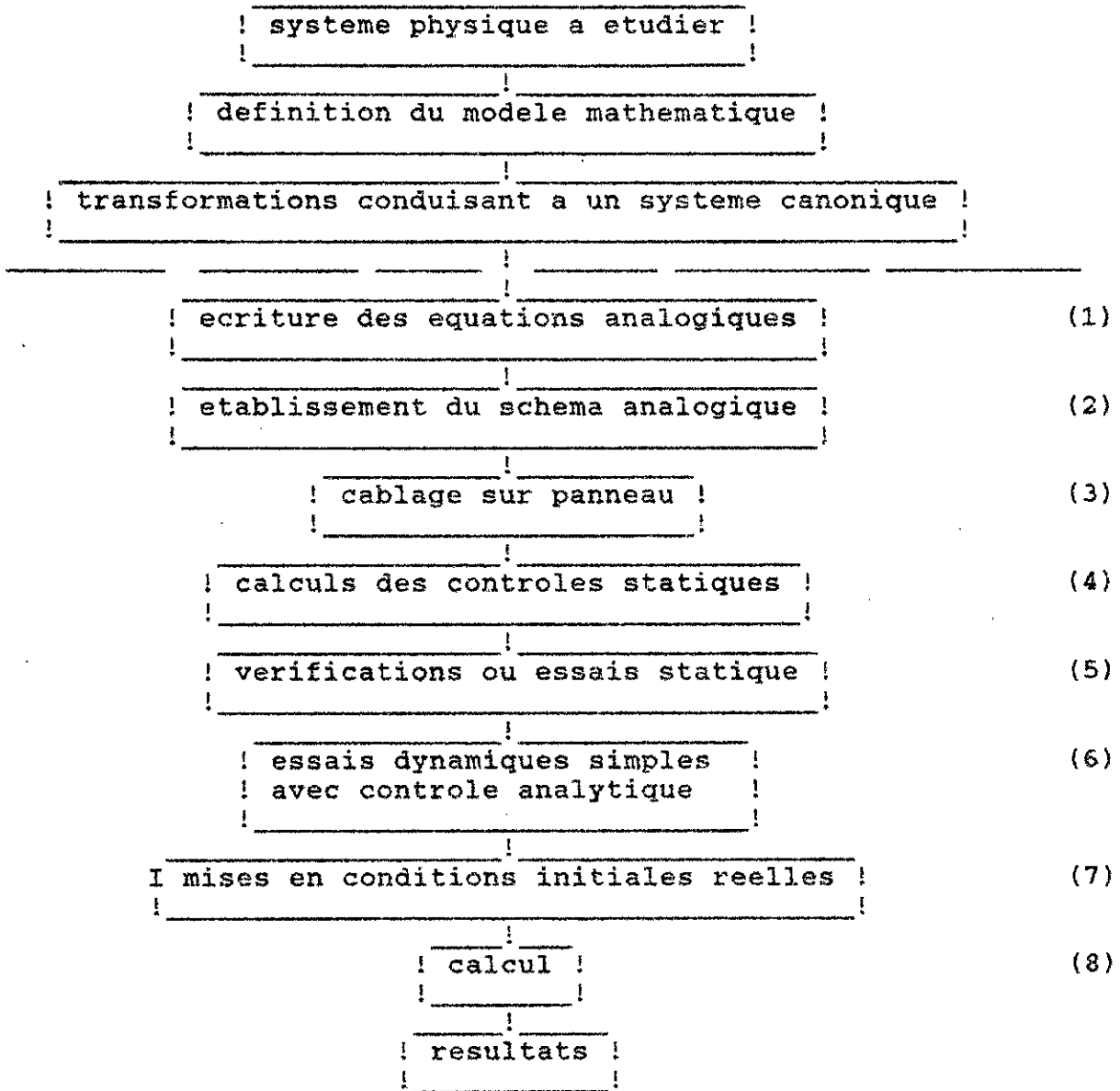
9 ) Phase d'enregistrement des resultats  
-----

Un organigramme resumant toutes ces etapes est represente page (18.b)

Il est important de noter que le temps de preparation augmente avec les dimensions du probleme.

En effet, le nombre d'operateurs intervenant dans le cablage augmente ce qui entraine un plus grand nombre de verifications.

Cependant, les dimensions du probleme n'affectent en rien le temps d'execution.



Organigramme des etapes de resolution sur calculateur analogique

---

En effet, le calculateur étant une machine à traitement parallèle, tous les blocs impliqués dans le calcul opèrent de façon simultanée.

Nous voyons donc que la programmation du calculateur analogique ne nécessite aucun langage particulier.

Cependant, il existe des programmes numériques qui fournissent une aide à la programmation des calculateurs analogiques. On peut citer le code APACHE qui est l'un des plus connus. Il a été conçu sur un calculateur analogique (SAI 231 R PACE) et un calculateur numérique IBM 360 (11).

Celui-ci existe en plusieurs versions : (21)

Il permet dans sa version la plus simple :

- D'établir les équations machines.
- De fournir les valeurs d'étalonnage des potentiomètres
- Le test statique

Ce qui permet de libérer l'analogiste de calculs longs et fastidieux.

Le calculateur analogique est soumis à des limitations qui sont liées à sa précision médiocre en statique  $\approx 10^{-4}$  de la pleine échelle.

Cette précision se dégrade en dynamique, surtout si les valeurs des facteurs d'échelle qui sont imposées par la plage de linéarité des amplificateurs sont mal choisies (20).

### I.1.8 ) CONCLUSION

-----

C'est le calculateur analogique qui a donné un sens pratique à l'élaboration des modèles d'équations différentielles (1).

Il a permis le développement de la simulation.

Les premiers calculateurs analogiques furent conçus vers les années 40 pour des applications spécifiques à l'aérospatiale, simulateurs de vol en particulier.

Ce n'est que vers les années 50 qu'ils ont atteint leur statut d'outils universels [2].

Les techniques de calcul analogiques ont été très employées car elles seules permettaient le calcul de grandeurs mettant en jeu de nombreuses opérations (22).

Cependant, l'application majeure de ces calculateurs est restée la simulation pour laquelle ils ont joué un très grand rôle.

Le développement des grands calculateurs numériques a créé un potentiel pour la résolution de problèmes mathématiques de complexité illimitée. Ceci a eu pour effet de réduire le champ d'application des calculateurs analogiques.

De plus, vers les années 60, avec la création des langages de simulation le rôle de ces calculateurs a diminué (21).

La conception de ces machines a donc été rationalisée et leur automatisation a été poussée.

Le calculateur analogique est encore utilise dans certaines applications recentes (19).

Plus particulierement, les automaticiens le considerent toujours comme un outil bien adapte a leurs problemes(5) qui sont resolu et plus facilement mis en oeuvre que par l'emploi d'un calculateur numerique [22].

Le poste de controle et de commande des calculateurs analogiques a ete remplace par un microprocesseur ce qui fut un premier pas vers les calculateurs hybrides.

## Calculateurs Numeriques

---

### I.2.1 Introduction

---

Le developpement des calculateurs numeriques s'est effectue de facon extraordinaire grace a l'evolution de la technologie des semiconducteurs .

De ce fait , le domaine d'application couvert par ces calculateurs est devenu de plus en plus vaste .

Cette evolution a permis d'aborder avec les calculateurs numeriques un type de problemes qui etaient alors reserves aux calculateurs analogiques : la simulation des processus physiques continus (5) .

### I.2.2 Langages de simulation

---

Durant les annees 50 , la puissance des calculateurs analogiques ainsi que leur facilite de programmation en faisaient les outils privilegies pour la simulation par rapport aux calculateurs numeriques existants .

L'evolution de la technologie a permis d'augmenter les performances des calculateurs numeriques de facon considerable (6) . En effet , on note une augmentation de la vitesse d'execution des operations au niveau du processeur ainsi qu'un acces plus rapide aux memoires .

Entre les annees 50 et 60 , de tres grands progres ont ete realises dans l'aspect theorique de la resolution numerique des equations differentielles .



C'est ainsi que , vers les années 60 , se sont développés les premiers langages de simulation numérique pour systèmes continus (2) .

Ces premiers langages étaient écrits en Assembleur . L'apparition des langages évolués tels que FORTRAN , ALGOL a permis de simplifier l'écriture de ces langages .

### I.2.3 Structure générale d'un programme de simulation

---

Un tel programme est constitué de 3 fonctions de base :

1) Initialisation : Trois classes de quantités sont généralement initialisées :

- a ) Les valeurs initiales des variables d'état
- b ) Les valeurs des paramètres du problème
- c ) Les constantes telles que TMAX qui est la durée de la simulation .

2) Execution : c'est la partie dynamique du problème. C'est la phase principale du programme .

Les équations du problème sont utilisées pour générer la solution .

La commande d'exécution fait appel à l'algorithme d'intégration choisi .

3) Phase finale : Elle fait appel aux fonctions suivantes :

- a ) Sortie complète des solutions
- b ) Les résultats sont testés pour savoir si on doit lancer une autre exécution ou si on doit arrêter la simulation .

L'avantage offert par les langages de simulation numerique est l'affranchissement des problemes d'echelles sur les variables .

#### I.2.4 Techniques de programmation

-----

Il existe plusieurs techniques de programmation des langages de simulation . On peut citer :

- Les langages " adaptes aux equations"
- Les langages a structure de blocs

a ) Langages adaptes aux equations : Ils permettent l'introduction du probleme sous forme d'equation d'etat .

b ) Langages a structure de blocs : Ce sont des techniques de simulation a hautes performances . Chaque instruction de ce type de langage joue le meme role qu'un operateur elementaire appartenant a un calculateur analogique . Ce type de langage permet d'augmenter la vitesse d'execution dans le cas d'un systeme de simulation sur minicalculateur (3) .

L'un des premiers langages utilisant la structure de blocs est DARE/ELEVEN .

Celui ci combine des segments de programmes ecrits en langage d'equations (representation en virgule flottante) avec des segments de programmes ecrits en langage de blocs .

Ces blocs representent les " composants " rapides du systeme a simuler exactement de la meme facon qu'un calculateur hybride combine des calculs analogiques .

### I.2.5 Simulation interactive

---

La simulation interactive des systemes permet a l'experimentateur d'introduire et d'editer ses programmes et donnees a partir d'un clavier .

Il a alors la possibilite d'observer les solutions sous forme de graphes ou autres des que celles ci sont calculees .

Il peut modifier les parametres et le modele en ligne sur la base des resultats les plus recents .

De plus , il peut etudier immediatement l'effet de ces modifications en lançant une autre execution .

Exemple de langage interactif :

DARE/ELEVEN adapte au minicalculateurs PDP 11

DARE PI adapte aux supermini de 32 bits serie VAX

### I.2.6 Architectures paralleles

---

La technologie et les techniques de programmation ont evolue de facon considerable permettant ainsi d'elargir le champ d'application des calculateurs numeriques conventionnels ( VAN NEUMANN ) .

Cependant , les performances , necessaires pour la resolution de certains problemes tels que : Meteorologie, Sismologie ... , n'ont pu etre atteintes avec ces calculateurs .

En effet , ces problemes sont caracterises par un algorithme complexe , un volume de donnees considerable tout en etant soumis a des contraintes de temps .

Leur resolution necessite des moyens permettant un traitement simultane d'operations .

Les recherches ont donc ete orientees vers une nouvelle organisation logique des calculateurs numeriques pour aboutir a une architecture caracterisee par plusieurs processeurs .

On parle alors d'organisation a multiprocesseurs . La puissance de calcul de ce type de calculateurs est superieure a celle des calculateurs numeriques conventionnels .

Il existe une tres grande variete d'architectures a multiprocesseurs . Le choix de l'une d'elle est dicte par des considerations pratiques .

Un exemple simple d'architecture d'un systeme a multiprocesseurs pour la simulation rapide de systemes continus de grande taille est donne fig I.4.[2] [3]

Le PDP-11/40 est le processeur de controle ( ou principal ) . Il controle de 1 a 16 processeurs arithmetiques tres rapides ( processeurs secondaires )

A chaque processeur secondaire est associe une memoire dans laquelle sont stockes les programmes ainsi qu'une routine d'integration .

La fonction de chaque processeur est d'integrer un sous ensemble du systeme d'equations .

A chaque pas d'integration , le processeur principal est interrompu par les processeurs secondaires de sorte a permettre le transfert des variables necessaires a d'autres processeurs pour continuer le calcul.

-----  
 Processeur de commande pour les E / S, l'unité disque et  
 pour les communications inter processeurs  
 -----

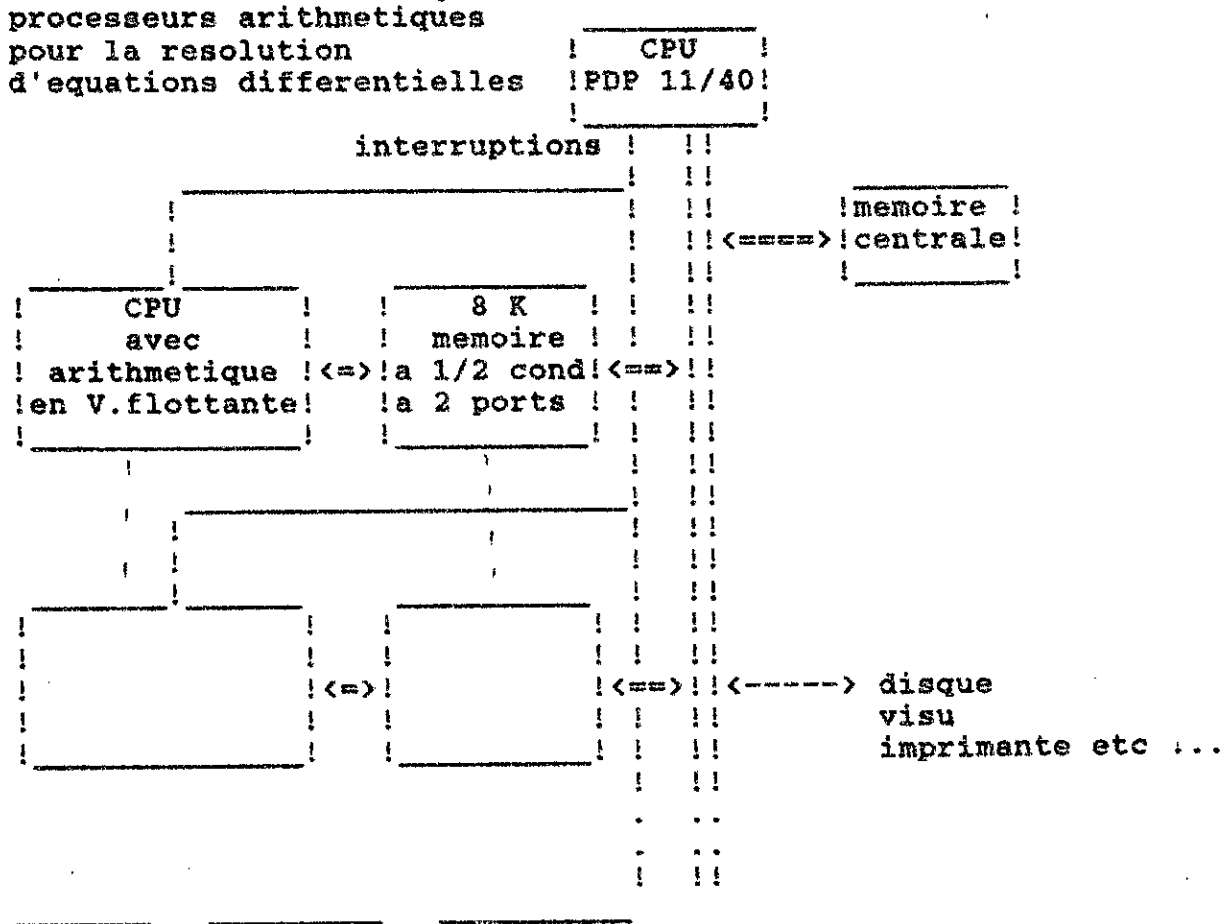


fig 1.4 Architecture d'un systeme a multiprocesseurs  
 -----

### I.2.7 Conclusion

La simulation numerique offre beaucoup de facilites de mise en oeuvre grace aux techniques de programmation que nous avons citees.

Elles permettent a l'utilisateur d'eviter les contraintes liees a la programmation .

Les architectures a multiprocesseurs presentent un interet particulier car il est souvent possible de diviser le calcul en parties independantes qui peuvent etre effectuees simultanement [7]. Cependant, un fort couplage entre les systemes d'equations qui sont affectes aux differents processeurs constituant l'architecture , fait perdre de son interet a cette derniere . En effet , dans ce cas un temps important est consacre aux echanges interprocesseurs .

De plus , la mise en oeuvre sur un systeme a multiprocesseurs peut presenter des difficultes quant au partitionnement des segments de programmes entre les differents processeurs .

- C H A P I T R E II -

CALCULATEURS HYBRIDES

---

I.1 ) INTRODUCTION

---

Necessite d'une conception hybride des calculateurs

---

Les outils de calcul ont ete developpes en fonction de la nature des problemes que les scientifiques etaient amenes a resoudre tout en etant conditionnes par l'essor de la technologie.

Les besoins en simulation ont mene au developpement des calculateurs analogiques et ceux de resoudre des systemes algebriques de plus en plus complexes ont mene au developpement des calculateurs numeriques.

Le couplage entre un calculateur analogique et un calculateur numerique a ete recherche par les specialistes du calcul analogique qui souhaitaient augmenter la flexibilite d'une simulation en utilisant la memoire et la logique d'un calculateur numerique (15). De plus, l'apport des calculateurs dans le developpement industriel est essentiel.

Les systemes industriels ont un caractere hybride : Ils sont constitues d'un systeme dynamique evoluant continuellement associe a divers automatismes et organes logiques de commande et de controle.

Leur simulation etait au depart impossible a l'aide des calculateurs analogiques ou des calculateurs numeriques existants.

Ceci a conduit tout naturellement a l'elaboration de calculateurs hybrides.

Ceux-ci sont obtenus en associant les deux moyens de calculs deja existants a l'aide de circuits d'interface appropries.

Cette organisation est tres interessante car elle permet de combiner les qualites des deux calculateurs a savoir :

- Rapidite d'integration de l'analogique et calcul parallele.
- Grande capacite memoire et grande precision du numerique.

## II.2 ) Bref historique

C'est en 1958 que furent realises les premiers calculateurs hybrides, issus de la liaison d'un calculateur analogique et d'un calculateur numerique dans les laboratoires d'etudes spatiales a LOS ANGELES et en CALIFORNIE (8) (12) (13).

Ces calculateurs furent developpes dans le but de simuler le mouvement d'un missile a longue portee.

Les calculs de coordonnees de la trajectoire demandant une grande precision furent confies au calculateur numerique alors que le mouvement autour du centre de gravite, necessitant une grande vitesse de calculs, fut simule par le calculateur analogique.



Parmi les premières combinaisons qui employaient les plus grands et plus rapides calculateurs numériques, on peut citer :

UNIVAC 1103 A ou IBM 704 qui étaient associés à 300 ou 400 amplificateurs d'ANALOG FACE (5).

C'est au début des années 60 que furent commercialisés les premiers calculateurs hybrides universels.

On peut citer : HYDAC 2000 développé par EAI. Ce fut le premier ensemble hybride de série (13).

Il était composé d'un calculateur analogique 231RVEAI d'une console logique DOS 350 EAI et d'un calculateur numérique PDF 24 (10) (13).

Cependant la plupart des calculateurs hybrides étaient largement influencés par les exigences des applications spécifiques à l'industrie aérospatiale (9).

### II.3 ) Calcul hybride et calculateurs hybrides

---

II.3.1 ) Calcul hybride : La notion de calcul hybride ne se limite pas uniquement à celle associant des techniques de calcul numérique et des techniques de calcul analogique pour la résolution d'un problème (5).

En effet, on trouve dans la littérature des exemples de ce type de calcul.

Nous allons en citer deux exemples :

#### II.3.1.a ) Machine analogique - Programme numérique

---

Dans ce cas nous avons à faire à une simulation purement analogique : une fois que le modèle est câblé sur le

panneau, c'est le calculateur analogique qui va effectuer le calcul.

Cependant, la preparation du probleme a etudier se fait a l'aide d'un ordinateur qui joue donc un role d'interface entre l'utilisateur et la simulation analogique.

En effet, il existe des logiciels qui, a partir des equations definissant le modele physique reel, permettent d'obtenir :

a ) les parametres du modele analogique tels que:

- Les valeurs des coefficients des potentiometres.
- Le choix des gains des amplificateurs.
- Le choix de constantes de temps des integrateurs.

b ) Le choix des composants et la creation d'operateurs intermediaires si besoin est.

Dans certains cas, il peut exister une liaison entre l'ordinateur et la machine analogique.

Le programme numerique permettra les operations suivantes:

- Verifier le cablage.
- Regler l'execution.

L'un des premiers langages de simulation analogique est le langage APACHE developpe a EURATOM a ISPRA. Ce langage fut ecrit en assembleur sur IBM 7090 (5) (8) (12).

#### II.3.1.b ) Simulation analogique sur ordinateur

Ceci est possible a l'aide de langages numeriques conformes aux normes CSSL.

Ces langages permettent a l'utilisateur d'imaginer qu'il utilise une machine analogique pour etudier son probleme. Leur facilite de programmation croissante permet leur utilisation par des non specialistes, ce qui leur permet d'aborder des problemes de plus en plus complexes. Ces langages donnent a l'utilisateur le moyen de :

- Choisir un certain nombre d'operateurs.
- De les "Interconnecter".
- D'afficher des valeurs de coefficient
- De lancer la simulation selon les modes classiques :
  - Conditions initiales.
  - calcul
  - memoire

On peut citer quelques uns de ces langages :

- CSMP developpe par IBM.
- DAREP, DARE/ELEVEN qui se presentent sous forme de fichiers modulaires (3).

Cependant, la notion de calcul hybride a eu pour origine la comparaison et la necessite de reunir le calcul analogique et le calcul numerique du fait de leur complementarite.

Ceci fut realise moyennant un certain nombre d'etapes.

Si l'on va du fait que l'on veut ameliorer le calcul analogique on distingue quatre etapes (21).

#### II.3.2.a ) L'analyseur differentiel digital (1950-1960)

-----

Du fait du manque de precision des operations elementaires lineaires des calculateurs analogiques, il s'est avere necessaire d'utiliser des techniques de base

purement numeriques pour realiser des operateurs ayant les memes fonctions que les operateurs du calculateur analogique.

Ces operateurs etaient mis en oeuvre de la meme facon que les operateurs d'un calculateur analogique (en realisant un schema reliant ces differents operateurs ).

Celui-ci fut tres vite abandonne du fait du progres des autres techniques (numeriques et analogiques).

#### II.3.2.b ) Calculateur analogique associe a des elements logiques

L'interet de ces calculateurs est de disposer d'un mode repetitif permettant le passage du mode conditions initiales au mode calcul a une cadence elevee.

On peut disposer de moyens supplementaires permettant de corriger automatiquement un certain nombre de parametres en fonction des resultats obtenus dans les solutions precedentes.

On obtient ainsi les analyseurs differentiels iteratifs (IDA).

#### II.3.2.c ) Adjonction d'elements memoire importants aux IDA

Les possibilites de memorisation des IDA sont tres reduites et les calculs proprement dits ne peuvent etre qu'analogiques. Une autre etape est donc celle dans laquelle des possibilites de memorisation importantes sont ajoutees aux IDA.

Ceci etait necessaire dans le cas ou il fallait conserver certaines donnees sous forme de tables (ex : generateurs

de fonctions a une ou plusieurs variables).

Ceci a permis d'elargir le champs d'application des IDA

II.3.2.d ) Calculateur analogique et logique parallele  
associes a un calculateur numerique

C'est la derniere etape qui a permit d'aboutir a un systeme hybride vrai ou Analog/Hybrid computer.

Dans ce systeme, le calculateur analogique et le calculateur numerique sont couples a l'aide d'elements d'interface hybrides.

Ces differentes etapes sont resumees sur la fig II.1.

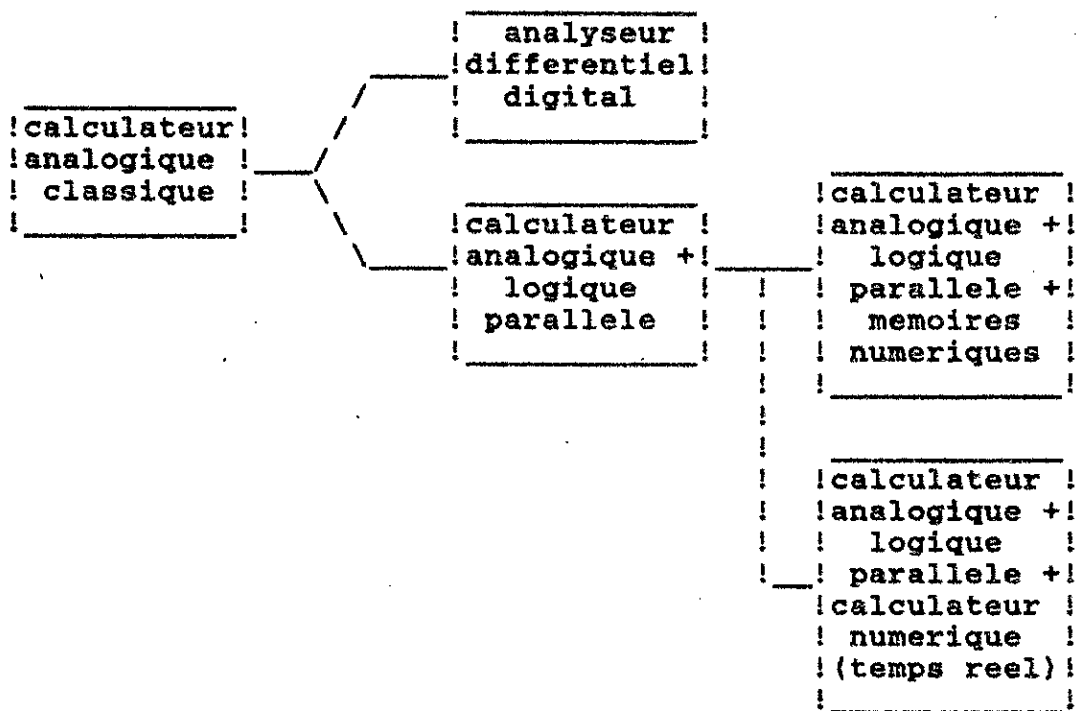


figure II.1 ETAPES DU CALCUL HYBRIDE

## II.4 ) Formes speciales de calculateurs hybrides (8), (22)

---

Le terme "calculateur hybride" a ete employe pour indiquer la combinaison de calculs continus et discrets (8), (22).

De ce fait, on va citer deux formes speciales de calculateurs "hybrides" qui ne resultent pas de l'association d'un calculateur analogique et d'un calculateur numerique.

### II.4.a ) Calculateur hybride parallele

---

On peut citer le calculateur HYDAC 2000 developpe par EAI. Il est constitue de l'association d'un calculateur analogique avec un systeme universel de blocs logiques programmables, un multiplexeur, un CA/D et des CD/A, une unite memoire pour le stockage des fonctions analogiques echantillonnees, ainsi que des nombreux additionneurs et soustracteurs.

Les applications de ces calculateurs englobent une gamme intermediaire de problemes hybrides :

- Simulation d'un retard.
- Generateur de fonction a une ou plusieurs variables.
- systemes asservis de controle.
- Automatisation du calculateur analogique dans le cas d'etudes d'optimisation ou de recherche de parametres.
- Simulation d'un systeme asservi de donnees numeriques echantillonnees.

#### II.4.b ) Calculateur hybride serie

Un simulateur de vol a ete developpe a MIT pour l'entrainement du personnel navigant.

C'est le Pulsed-Analog-Computer.

Ce systeme est constitue d'un calculateur numerique qui controle un petit nombre de composants analogiques fonctionnels tels que : multiplieurs, integrateurs ....

Ces composants analogiques sont controles par le programme numerique. Ils realisent des operations equivalentes a des sous-programmes pour le calculateur numerique.

#### Definition du calcul hybride

Le calcul hybride peut etre defini comme une utilisation de calculateurs analogiques et numeriques relies entre eux par un sous-ensemble de liaison (ou interface) et mis en oeuvre selon des techniques de programmation propres a ce mode de calcul.

Selon les fonctions devolues au calculateur numerique on distingue :

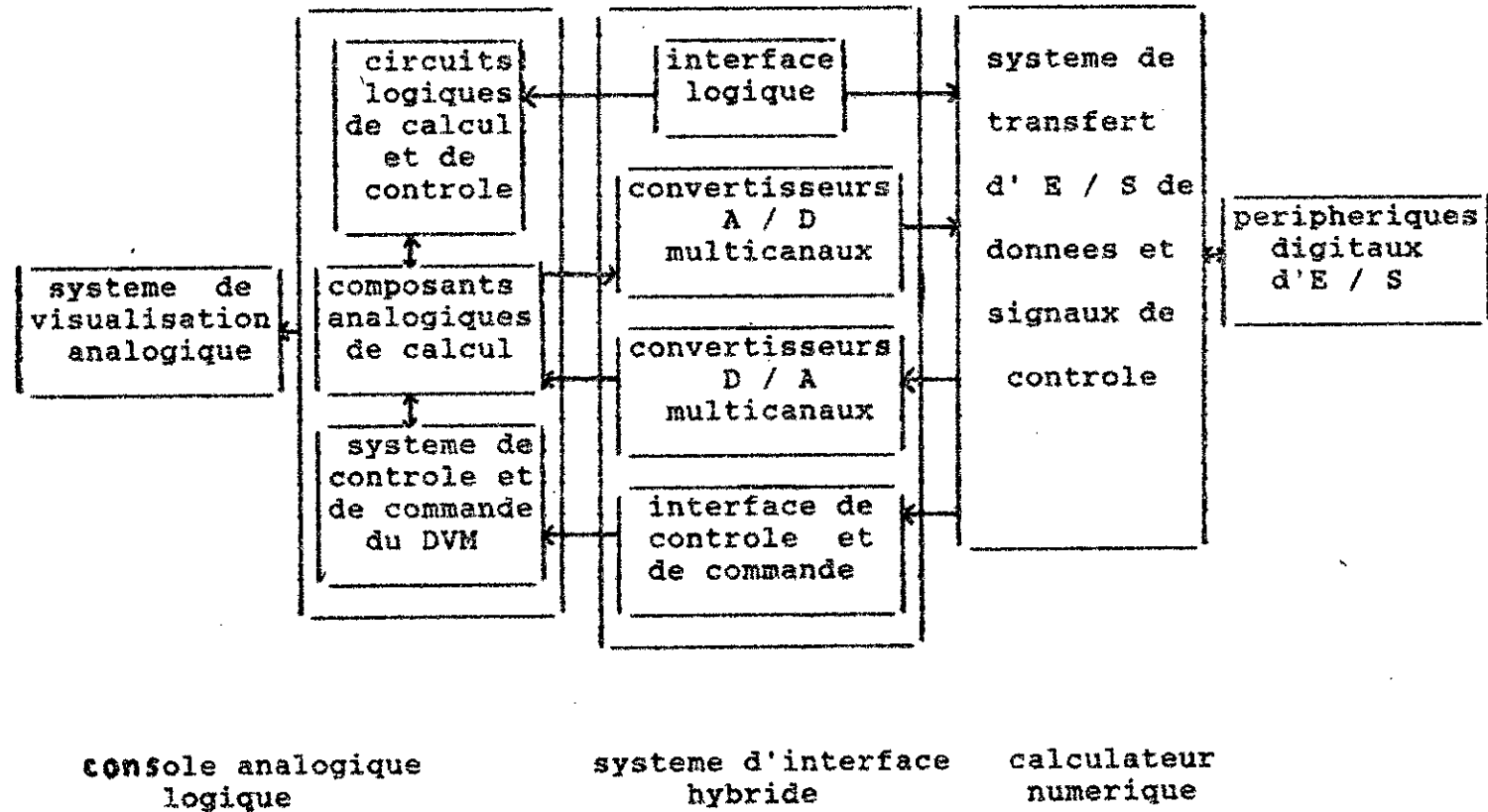
- Le calcul hybride serie (5) (13)

Le calculateur numerique est utilise en serie avec le calculateur analogique.

- Le calcul hybride parallele

Dans ce cas le calculateur numerique est utilise en parallele avec le calculateur analogique.

fig (II.2) Organisation generale d'un calculateur hybride (2)





On voit apparaitre dans cette organisation trois blocs distincts :

- a ) Le calculateur numerique.
- b ) Le calculateur analogique ou console analogique/logique.
- c ) Le systeme d'interface hybride assurant le lien entre les deux blocs precedents.

Le calculateur analogique peut etre considere comme un peripherique du calculateur numerique.

Les donnees traitees par les deux calculateurs sont de nature differente (continues et discrettes). Des elements d'interface sont donc necessaires pour permettre le traitement d'une meme donnee par les deux calculateurs. Ces elements d'interface assurent la liaison entre les deux calculateurs.

#### II.5.1 Systeme d'interface hybride

---

Celui-ci est compose de trois parties :

##### II.5.1.a ) Une interface de commande et de controle

---

Elle permet le transfert d'informations logiques dans les deux sens.

Ces informations permettent la commande des modes analogiques, logiques et des vitesses d'integration.

Elle permet aussi :

- Le reglage de tous les parametres d'une simulation.
- De lire toutes les valeurs des operateurs.

II.5.1.b ) Une interface logique : Celle-ci assure la liaison entre la partie logique du calculateur analogique et le calculateur numerique.

Une information logique peut interrompre un programme numerique en cours.

II.5.1.c ) Une interface de donnees : elle est constituee d'un certain nombre de convertisseurs A/D et D/A permettant le transfert d'une information du domaine analogique ou continu au domaine numerique ou discret.

Le transfert de l'information est represente en fig II.3.

II.5.2 ) Description de la partie analogique

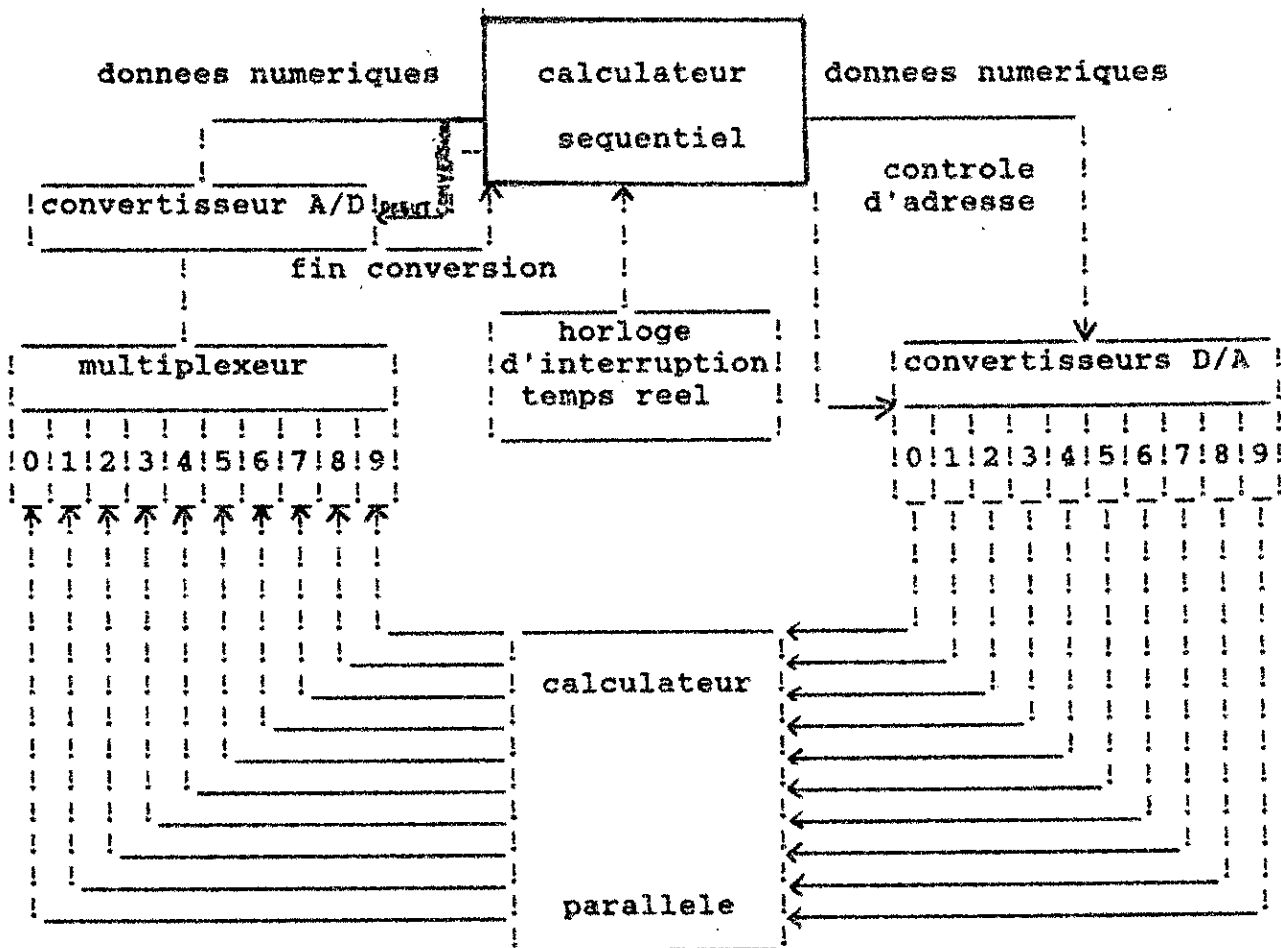
Les operateurs analogiques et logiques rencontres dans un systeme hybride sont identiques a ceux d'un calculateur analogique mis a part les operateurs a commande programmable, on distingue :

- Les potentiometres programmables.
- Les amplificateurs programmables .....

Ces operateurs sont des operateurs hybrides car ils effectuent le produit d'une tension analogique par un mot binaire, le resultat etant donne sous forme d'une tension analogique.

Leur programmation s'effectue a partir du calculateur numerique.

fig II 3 Transfert des donnees du calculateur numerique  
vers (du) calculateur analogique



C'est le programme sequentiel qui controle le cycle de conversion.

Le cycle est initialise par une horloge d'interruption.

On dispose d'un grand nombre de convertisseurs D / A.

Les sorties analogiques sont multiplexees avant d'etre converties par le CA / D.

L'introduction du coefficient d'un potentiometre ou du gain d'un amplificateur s'effectue a partir du clavier d'une console digitale.

### II.5.3 Partage des taches entre les elements constitutifs d'un calculateur hybride

Nous venons de voir la composition d'un calculateur hybride, nous pouvons donner une idee du role que joue chacun des elements dans le calcul hybride.

#### II.5.3 a Calculateur analogique

La tache generalement confiee au calculateur analogique d'un ensemble hybride est la resolution des equations differentielles du fait de sa grande vitesse de resolution de ce type d'equations.

#### II.5.3 b Logique parallele

Les fonctions de la logique parallele sont :

- Automatisation du calculateur analogique
- Commandes des moyens de sortie analogiques tels que enregistreurs, oscilloscope.
- Simulation d'operations logiques (pour des operations simples, la logique parallele est beaucoup plus rapide que le calculateur numerique).
- Synchronisation de l'ensemble hybride.

Le calculateur analogique pourra résoudre a grande vitesse un système d'équations différentielles alors que la logique parallèle commandera le déroulement des cycles de calcul, la modification des conditions initiales et les organes d'enregistrement et de visualisation.

### II.5.3 c Calculateur numérique

-----

Dans un ensemble hybride, les tâches principales d'un calculateur numérique sont :

- Le réglage et la lecture des composants analogiques
- L'automatisation des opérations sur le calculateur analogique.
- Le calculateur numérique peut avoir en mémoire un programme très élaboré permettant la comparaison d'un résultat de calcul à des données physiques en vue de déterminer de nouvelles valeurs de coefficients ou de choisir un modèle mathématique différent.
- Certaines opérations du calcul analogique (par exemples les intégrations lentes)
- La génération de fonctions à une ou plusieurs variables.
- Génération de retards
- Résolution d'équations algébriques
- Simulation d'un calculateur numérique en ligne. . .

## II.5.4 Programmes et langages

---

le logiciel hybride facilite la communication entre l'utilisateur et le calculateur hybride.

Il a beaucoup de points communs avec le logiciel numerique classique. Il ne se differencie que par les programmes d'un certain type :

- Programmes relatifs au sous ensemble de liaison
- Programmes utilitaires
- Programmes de mise au point et de verification.

Les langages de programmation, les programmes bibliotheque et le systeme moniteur ne sont pas specifiques de l'ensemble hybride (13).

Dans de nombreux problemes peuvent s'exercer des contraintes de temps provenant du besoin de communiquer avec un systeme externe au calculateur numerique.

De ce fait, le programme hybride doit donc etre capable d'accepter, de traiter et de repondre a des informations provenant de systemes externes.

Pour cela, il doit etre constitue d'un moniteur temps reel.

## II.6 Logiciel d'un calculateur hybride

---

### II.6.1 Logiciel minimal

---

Une premiere etape dans le developpement du logiciel est celle permettant l'automatisation du calcul analogique.

Ceci a mene a un logiciel minimal forme d'un ensemble de sous programmes appelables en FORTRAN.

On peut citer FORTRAN LINKAGE (developpe par EAI) ainsi que HCR (hybrid communication routine) developpe par Applied Dynamics.

Ces logiciels permettaient les operations suivantes :

II.6.1 a Pour l'interface de controle  
-----

- Selection d'un operateur.
- Lecture de sa valeur de sortie
- Selection du mode operatoire analogique
- Choix de la vitesse de calcul
- Reglage d'un attenuateur

II.6.1 b Pour l'interface logique  
-----

- Selection d'un mode operatoire
- Choix d'une horloge de synchronisation
- Affichage ou lecture d'un mot logique
- Affichage d'une ligne de controle
- Lecture d'une ligne d'etat

II.6.1 c Pour l'interface de donnees  
-----

- Lecture d'un ou plusieurs convertisseurs A/D
- Selection d'un convertisseur D/A

La faiblesse de ce type de logiciel est son caractere ponctuel qui nuit a la vitesse d'execution dans le cas ou l'on effectue une recherche de parametres.

## II.6.2 Logiciels hybrides conformes aux normes CSSL

---

Par la suite, les concepteurs de langages CSSL ont pense a developper des langages de simulation s'adaptant aux calculateurs hybrides, sans toutefois aboutir a des langages universels (2).

Les calculateurs ainsi obtenus permettent le calcul parallele tout en beneficiant des facilites de programmation des calculateurs numeriques.

Le calcul hybride a atteint son dernier stade en trois etapes :

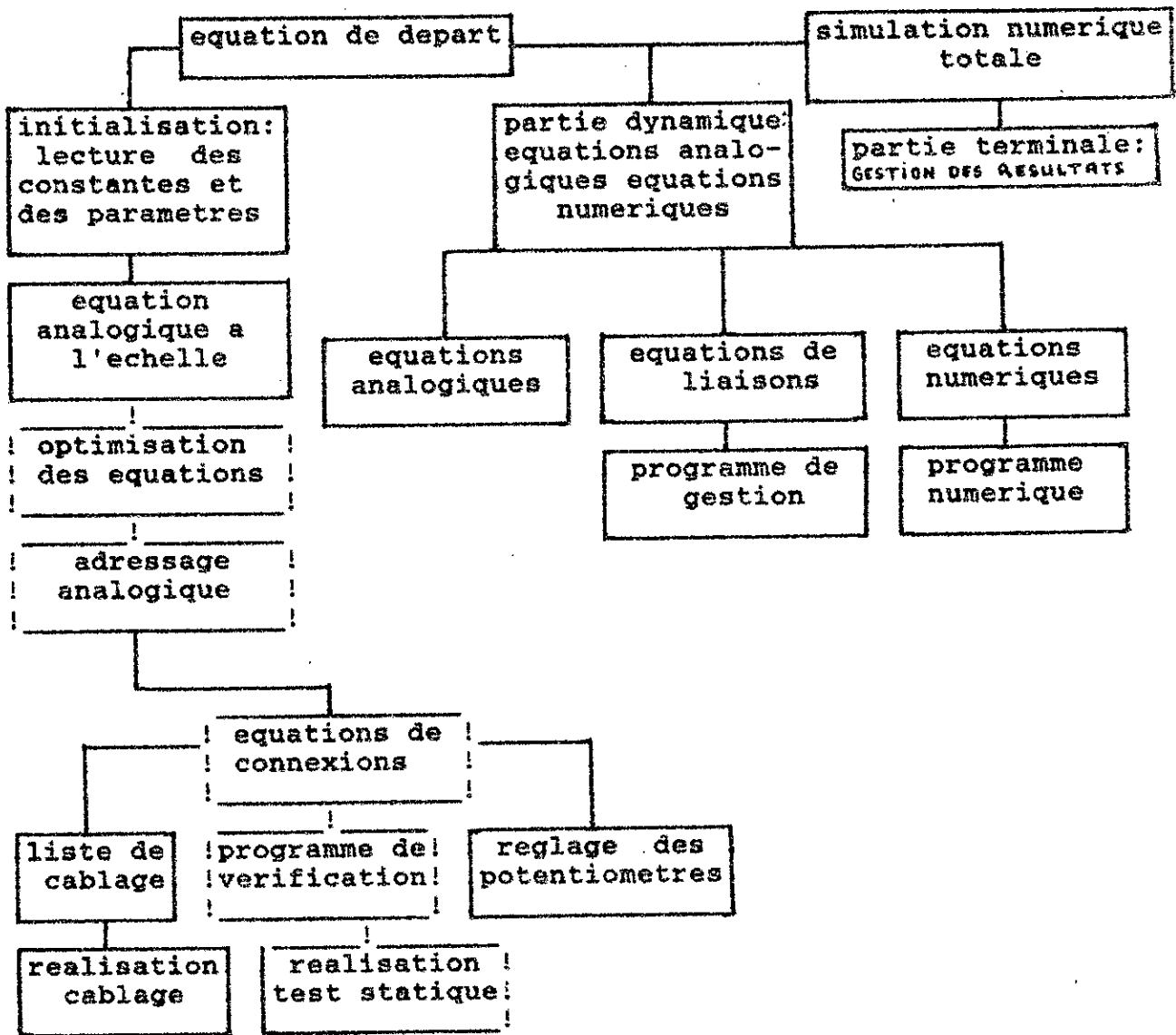
Premiere etape (1958 - 1962) : couplage d'un calculateur analogique et d'un calculateur numerique.

Deuxieme etape (1962 - 1966) : Realisation d'ensembles hybrides avec un logiciel limite.

Troisieme etape 1966 ---> : ensemble de calcul hybride avec un logiciel evolue.

Un organigramme resumant les etapes de programmation sur calculateur hybride est donne fig II.4.





P R E P A R A T I O N

E X E C U T I O N

fig (II.4) organigramme de resolution d'un probleme hybride

## II.7 Domaines d'applications du calculateur hybride

---

L'operation de base du calculateur hybride est, comme pour le calculateur analogique, l'integration continue par rapport au temps machine.

De ce fait, le calculateur hybride est tres bien adapte a :

- La resolution des equations differentielles ordinaires notamment dans le cas ou le systeme presente des non linearites dont l'approche analytique de la solution est complexe voire impossible.

- La resolution des equations aux derivees partielles (9), (25).

- Il permet aussi la resolution d'equations algebriques bien qu'il ne soit pas tres adapte a ce type de problemes (5).

Les applications les plus courantes sont relatives a :

- La dynamique des systemes (dans l'industrie la plupart des grands systemes sont controles par des ordinateurs)

- A la regulation de ces systemes

- A l'optimisation de leur fonctionnement (le processus est simule par le calculateur analogique. Les calculs d'optimisation sont effectues par le calculateur numerique).

- A l'identification des processus (recherche des parametres pour obtenir la forme generale d'un modele mathematique pouvant rendre compte de la dynamique du processus).

Toutes ces applications sont rencontrées en grande partie dans le domaine industriel et en particulier dans le domaine spatial, automobile, énergie nucléaire et chimie. D'autres domaines sont ceux de la médecine et de la biologie animale ou végétale.

- C O N C L U S I O N -  
=====

Pendant de nombreuses années, les calculateurs hybrides ont joué un rôle très important dans l'industrie aérospatiale.

Vers les années 60, le développement de ces calculateurs fut consolidé par un grand potentiel d'utilisateurs dans ce secteur.

Les applications dans d'autres industries et dans des disciplines scientifiques furent aussi remarquables.

Avec l'évolution de la technologie et la configuration à multiprocesseurs, les calculateurs numériques ont eu tendance à remplacer les calculateurs hybrides dans les tâches de simulation de petits et grands systèmes physiques.

Cependant, les langages de simulation numérique adaptés à la configuration à multiprocesseurs sont très difficiles à mettre en œuvre dans certains cas.

En effet, les plus grandes difficultés sont liées au partitionnement et à l'échange entre processeurs.

De plus, il semble que la façon la plus simple d'écrire le programme pour un système est lorsque le jeu d'équations différentielles le définissant peut se décomposer en sous-système pouvant chacun être résolu par son propre processeur.

Il arrive souvent que la mise au point et l'organisation d'une simulation hybride presentent nettement moins de difficulte que dans le cas d'une simulation purement numerique et ceci dans le cas de systemes fortement couples.

Compte tenu des facilites de mise en oeuvre, du cout et de la possibilite de s'adapter a des applications particulieres, la simulation hybride se revelera plus adequate.

De plus, les calculateurs hybrides existants seront toujours utilises des qu'ils accomplissent leur tache de facon satisfaisante [26] [32] [27].

On trouve encore dans un grand nombre d'universites des installations modestes de calculateurs hybrides (5).

Ces installations permettent la comprehension de la dynamique des systemes par les etudiants qui ne peuvent disposer d'installations experimentales reelles.

- CHAPITRE III -  
-----

III.1 Introduction  
-----

Nous disposons de la partie analogique d'un calculateur hybride ainsi que de l'interface logique.

Nous avons eu a realiser toute la partie microsysteme destinee a gerer le calculateur hybride ainsi obtenu.

Ce microsysteme est constitue d'une carte micro-calculateur batie autour d'un microprocesseur 8 bits (MC 6800 de MOTOROLA).

Nous avons egalement realise une partie de l'interface hybride : l'interface analogique qui est destinee a la conversion des tensions analogiques et a leur visualisation sur l'ecran.

Nous avons d'autre part developpe un logiciel permettant l'exploitation du calculateur hybride. La premiere partie de ce chapitre est destinee a la description materielle du calculateur hybride.

La seconde partie sera consacree au logiciel que nous avons developpe.

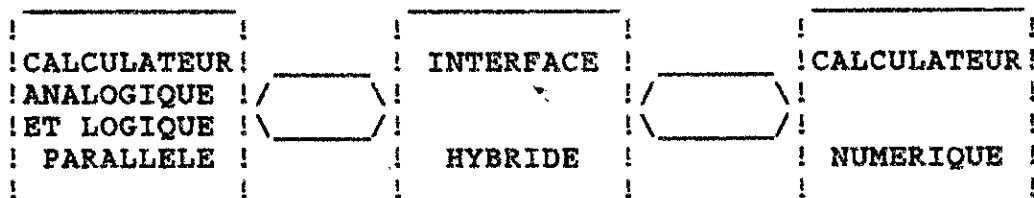
### III.2 Description du calculateur hybride

---

Dans le but de comprendre le fonctionnement du calculateur hybride nous allons decrire tous les elements qui le constituent :

D'une facon globale, la configuration du calculateur hybride peut se decomposer en une partie numerique et une partie analogique (fig III.1).

Les liaisons entre les deux parties sont assurees par une interface hybride.



(fig III.1)

#### Synoptique du calculateur hybride

La partie analogique est constituee :

---

- D'un champ analogique : dans lequel se trouvent les elements operationnels de base effectuant les operations fondamentales du calcul analogique.
- D'un champ logique : Compose d'operateurs logiques permettant entre autres de commander les modes des operateurs analogiques.

#### III.2.1 Description du champ analogique

---

La structure de ce champ est modulaire, chacune des cartes de ce champ comporte un seul type d'operateurs.

Les entrees et les sorties de ces operateurs sont reliees au niveau du panneau de cablage permettant ainsi de les interconnecter en vue d'obtenir le modele de simulation.

On distingue :

-----  
III.2.1 a Les operateurs purement analogiques :  
-----

Ceux-ci effectuent un traitement continu sur des tensions continues.

Les operateurs dont nous disposons sont les suivants :

- Sommateur
- Racine carree
- Track / Store
- Multiplieur / Diviseur
- Sinus / Cosinus
- Logarithme / Exponentiel
- Integrateur

III.2.1 b Les operateurs logiques (ou logique parallele)  
-----

Ces operateurs permettent le controle des operateurs analogiques de sorte a decharger le calculateur numerique au cours du calcul.

Il se compose d'elements logiques usuels :

- Portes logiques
- Elements memoire
- Compteurs ...



### III.2.1 c Les operateurs analogiques a commande numerique

---

La premiere etape permettant le passage d'un calculateur analogique a un calculateur hybride est celle liee a la presence d'operateurs appeles :  
operateurs hybrides

Ces operateurs effectuent la multiplication d'une tension analogique par une tension binaire, le resultat etant delivre sous forme de tension analogique.

La particularite de ces operateurs est leur capacite a etre commandes a partir du calculateur numerique.

On distingue :

- Les potentiometres numeriques (PTN)
- Les amplificateurs programmable (APR)
- Les generateurs de fonctions (GDF)

Ces operateurs peuvent etre controles et adresses a partir du microsysteme par programme.

Les coefficients des PTN et les gains des APR sont introduits a partir d'un clavier.

Les PTN et les APR sont realises autour de convertisseurs MDAC 7521 d'Analog Devices.

Ce sont des convertisseurs D/A 11 bits + signe

Une description de ces operateurs est donnee en

Annexe I.

### III.2.2 Description des interfaces

On distingue deux types d'interfaces :

- L'interface logique
- L'interface analogique

#### III.2.2 a Interface logique

On dispose de deux cartes d'interface logique.

a) La carte permettant la selection et le chargement des potentiometres numeriques, des amplificateurs programmables et eventuellement des generateurs de fonctions.

Le synoptique de cette carte est donne (fig III.2).

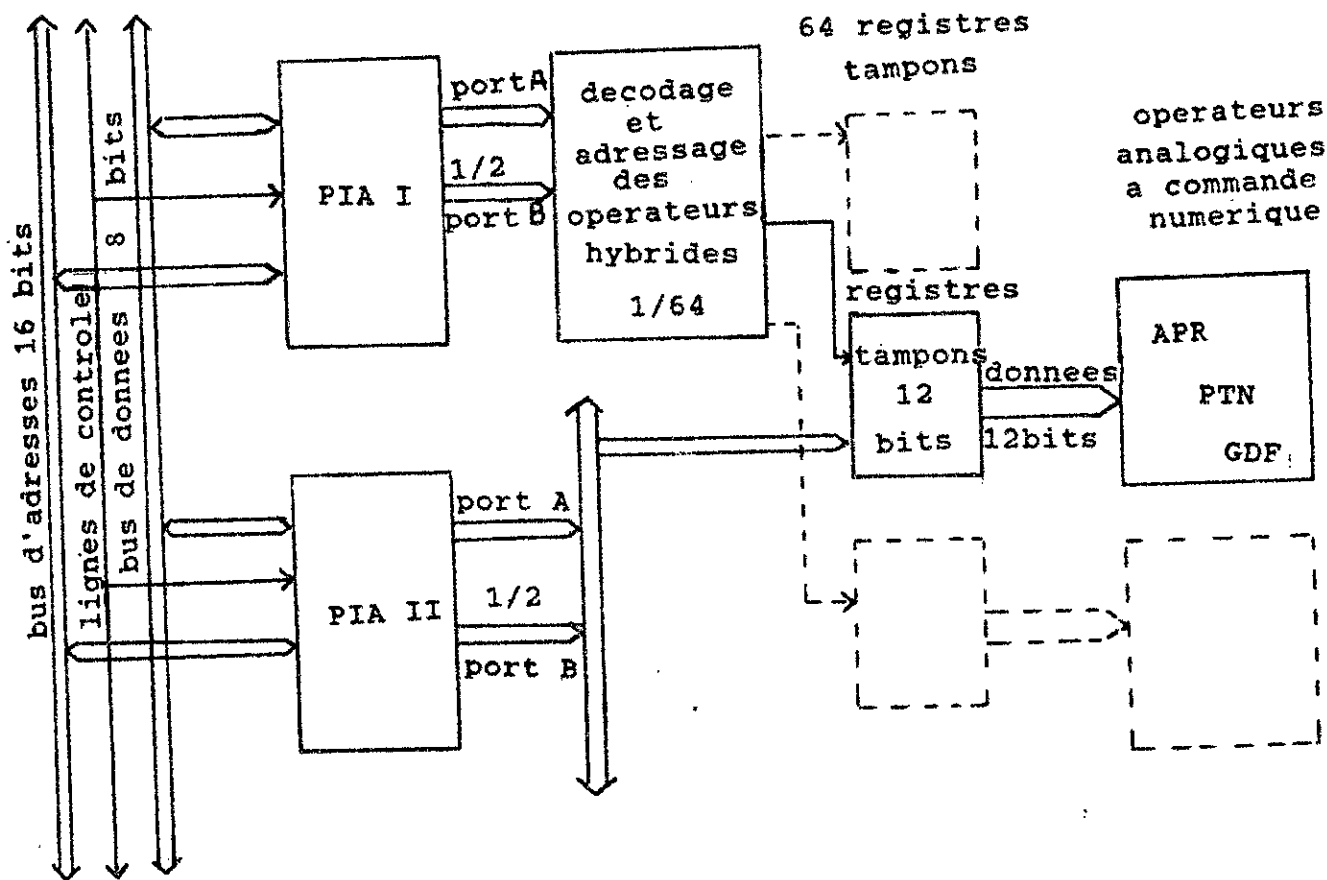


fig III.2

SYNOPTIQUE DE L'INTERFACE LOGIQUE

Pour la communication avec le microsysteme, nous avons des interfaces d'E/S parallele du type PIA.

Le PIA I permet la selection d'un ou plusieurs operateurs.

Le PIA II permet le chargement d'une donnee binaire 12 bits au niveau de l'operateur selectionne.

Cette donnee est introduite par l'utilisateur a partir d'un clavier sous forme d'un nombre decimal ASCII.

Dans le cas des potentiometres numeriques, cette donnee represente le coefficient du PTN. Cette valeur est comprise entre -1 et +1.

Nous avons developpe un programme permettant la conversion de cette donnee decimale ASCII en une configuration binaire 12 bits qui representera l'entree binaire N du PTN.

Un autre programme permet la conversion d'une donnee representant le gain d'un amplificateur en une configuration binaire 12 bits. Dans notre cas nous avons un gain variant entre +1 et 100.

Pour pouvoir charger ces donnees binaires au niveau des entrees des PTN et des APR, la carte d'interface logique dispose de registres tampons (12 bits) en sortie.

b) La selection des integrateurs et le chargement des constantes de temps sont effectues par une carte d'interface logique identique a la precedente.

Cependant, cette carte n'utilise qu'un seul PIA.

Le port A et la moitie du port B sont destines a la selection d'un ou plusieurs integrateurs parmi 24.

Les quatre dernières lignes du port B sont utilisées pour le chargement de la constante de temps.

Ces lignes commandent un interrupteur électronique à 4 voies (A,B,C,D).

On a la possibilité de choisir parmi quatre constantes de temps 1s ; 0,1s ; 1ms ; 0,1ms (Voir ANNEXE II).

### III.2.2.d ) Interface analogique

L'interface analogique est constituée d'une carte DVM que nous avons réalisée.

Celle ci permet :

- La sélection d'un opérateur parmi 16 (extension possible).
- De convertir une tension analogique sous forme d'un mot binaire de 12 bits.

Nous avons développé et testé un programme permettant de convertir cette grandeur binaire (12 bits) en decimal ASCII pour affichage sur l'écran de visualisation.

#### Description et fonctionnement de cette interface

Le synoptique de la carte DVM est donné (fig III.3).

(\*) Il n'a pas été choisi de mémoriser la sortie de chaque opérateur analogique ce qui aurait conduit à placer un CA/N pour chaque sortie.

Les sorties des opérateurs analogiques sont ramenées au niveau de la carte DVM par l'intermédiaire de câbles blindés pour éviter les effets d'inductions parasites.

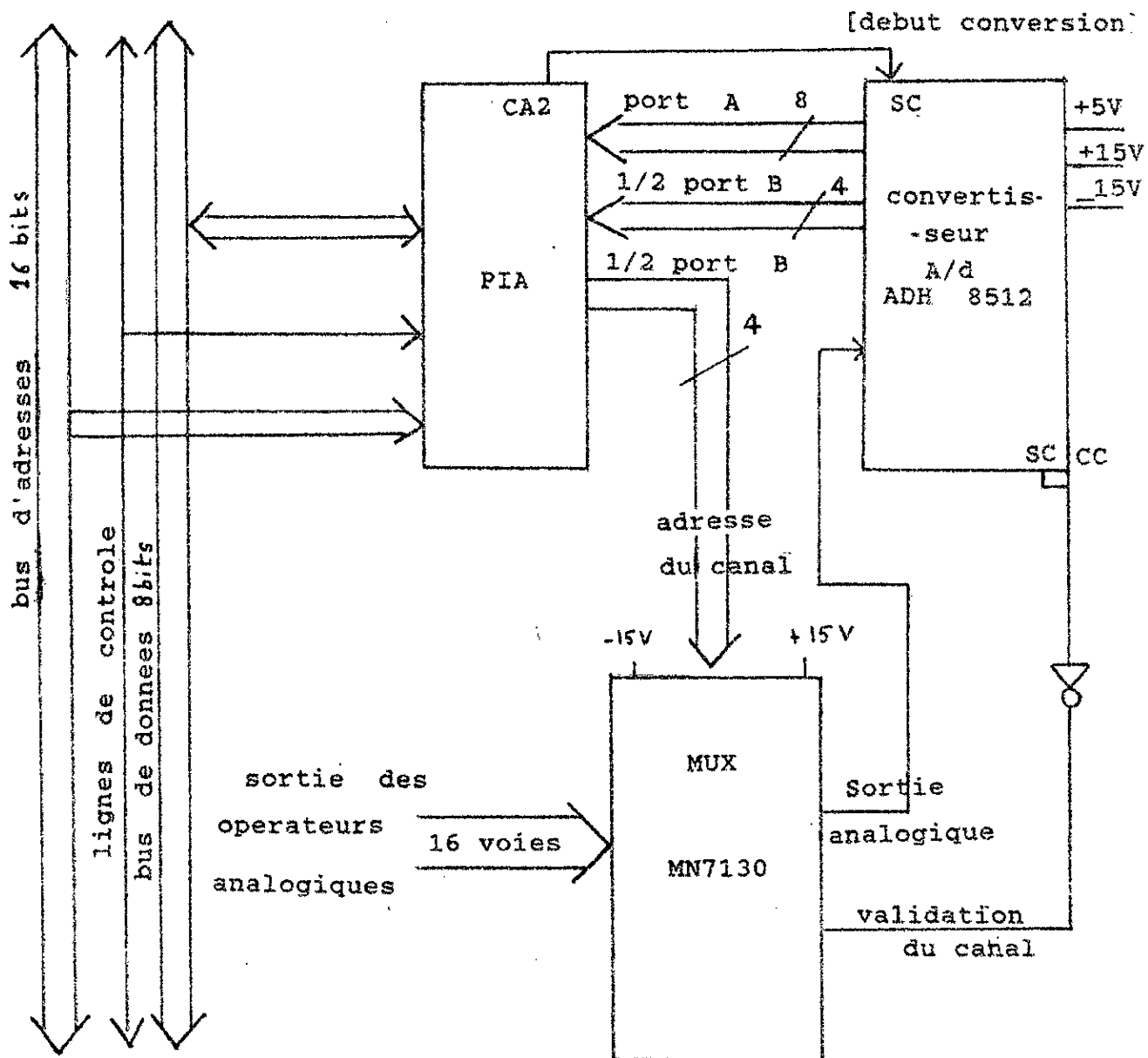


fig (III.3) Synoptique de la carte DVM

Ces tensions sont ramenees a l'entree d'un multiplexeur permettant la selection de l'un des operateurs.

Dans notre cas on s'est limite au multiplexage de 16 voies mais une extension est possible en ajoutant un second niveau de multiplexage.

Les elements qui composent cette carte sont :

- 1 PIA permettant la communication avec le microsysteme.
- 1 convertisseur A/D 12 bits : ADH 8512 (disponible au laboratoire ).
- 1 multiplexeur 1/16 : MN7130.

Le PIA permet la selection d'un canal du multiplexeur a travers quatre lignes du port B.

La ligne CA2 du PIA est programmee en sortie en mode impulsionnel. Elle permet de lancer le cycle de conversion A/D.

Le temps de conversion de l'ADH 8512 est de 1,8  $\mu$ s.

La conversion se fait en choisissant l'une des configurations suivantes :

- binaire pur.
- binaire decale.
- complement a deux.

Dans notre cas, la configuration binaire decalee est la plus adequate pour l'ecriture du programme de conversion du binaire en decimal ASCII.

La precision de ce convertisseur est de +0,012% de la pleine echelle. Celle des operateurs analogiques est reportee en ANNEXE III.

Les tensions converties par le CA/D varient entre -10V et +10V. Celui-ci peut être programme suivant trois modes :

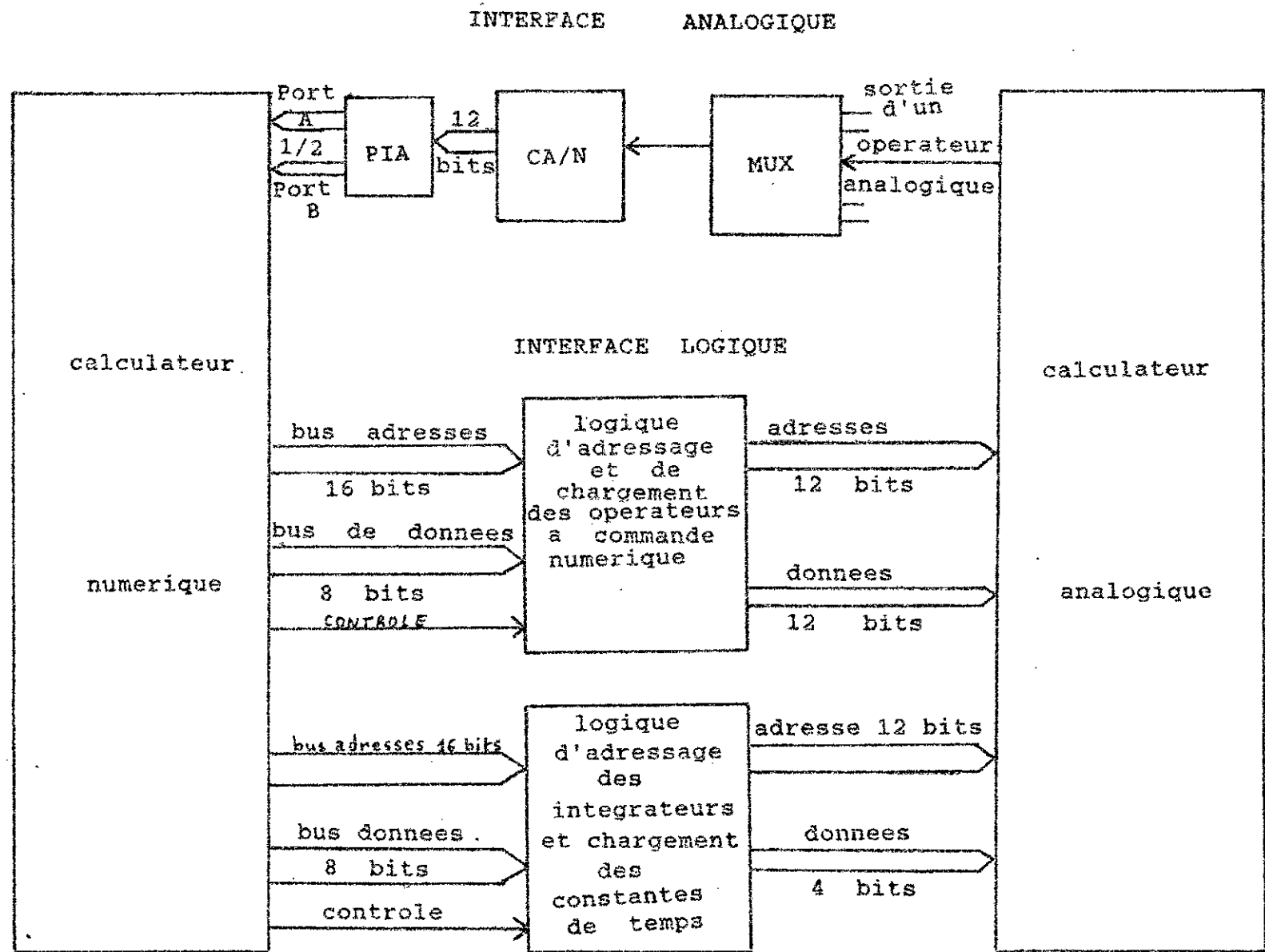
- 1) Ignorer une nouvelle commande de conversion jusqu'à ce que la conversion précédente soit achevée.
- 2) Entamer un nouveau cycle de conversion en interrompant le précédent.
- 3) Permettre des conversions successives (mode répétitif).

Il est spécialement adapté au multiplexage et à l'interfacage avec les microprocesseurs.

Nous avons dans notre cas choisi le premier mode de programmation.

Ce convertisseur est équipé de registres tampons en sortie qui permettent de mémoriser la configuration binaire obtenue tant qu'un nouveau cycle n'a pas démarré. Les cartes interfaces ayant été décrites nous pouvons résumer la liaison entre le calculateur analogique et le calculateur numérique par la fig III.4.

FIG III.4 - Synoptique de l'interface hybride.





### III.2.3 Description du champ numerique

---

#### III.2.3 a Fonctions realisees par le champ numerique

---

le champ analogique que nous avons decrit doit etre couple a un microsysteme dont le role est d'assurer la gestion et l'exploitation du calcul analogique et eventuellement du calcul numerique.

Pour cela, nous avons realise une carte microcalculateur permettant les fonctions suivantes :

- Controle de modes analogiques (ou logiques).
- Choix des constantes de temps des integrateurs.
- Chargement et affichage des differents parametres :
  - \* Coefficients des PTN
  - \* Gain des APR
  - \* Valeur des GDF
- Commutation des circuits d'interface et multiplexeur.
- Adressage du DVM.

#### III.2.3.b Composition de la carte microcalculateur

---

Cette carte est composee :

- D'un bus de donnees paralleles , bidirectionnel de huit bits .
- D'un bus d'adresse unidirectionnel 16 bits et de lignes de commande .

Les elements constituant cette carte sont :

- Un microprocesseur 8 bits (le MC 6800 de Motorola) ainsi que les buffers necessaires a son fonctionnement (8T26,8T95) (le MC 6800 a ete utilise pour des raisons de disponibilite).

- Une memoire centrale composee des elements suivants:

\* 3 EPROM 2516/TI dont la capacite de chacune est de 4 K/octets . Nous avons implante dans ces memoires , le moniteur que nous avons concu ainsi que tous les programmes que nous avons developpe pour l'exploitation du calculateur hybride.

\* 1 RAM 2716 d'une capacite de 2 Koctets . Elle permet le stockage des informations temporaires.

- Les circuits de decodage permettant l'adressage de tous les composants implantes sur la carte microcalculateur.

- Une interface serie ou ACIA ( MC 6850 ) permettant le dialogue entre l'utilisateur et l'unite centrale par l'intermediaire d'une console de visualisation .

- Une interface parallele ou PIA ( MC 6820 ) dont la ligne CA2 commande le demarrage du cycle de conversion de la carte DVM . Les lignes PA0 et PA1 du port A commandent le calcul en mode normal .

- Nous avons prevu un PICU dans le cas d'une extension future du systeme . Ce PICU peut permettre la gestion de huit niveaux d'interruptions hierarchisees .

- Une horloge programmable realisee a partir de deux timers MC 6840 montes en cascade selon la (fig III 5).

Les sorties O'3 et O'2 commandent les entrees IC et OP des integrateurs . Cela permet de mettre les integrateurs dans l'un des trois modes suivants :

- Mode conditions initiales ou IC
- Mode calcul ou OP
- Mode memoire ou H

La table des sequences des modes est la suivante .

	Entree IC	Entree OP
mode IC	1	0
mode OP	0	1
mode H	0	0

Les timers sont programmes de sorte a piloter deux modes de fonctionnement du calculateur hybride :

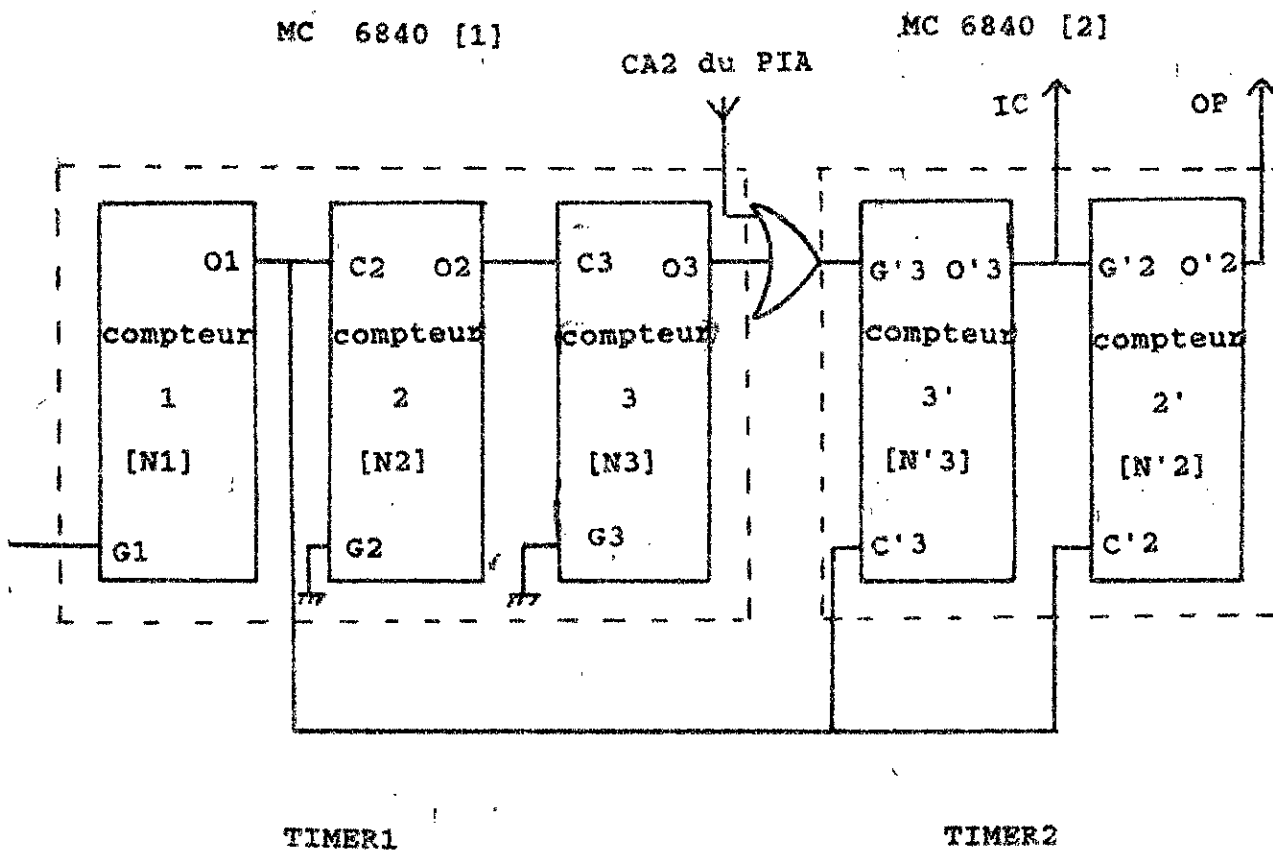
- Mode repetitif
- Mode iteratif

Un chronogramme de ces deux modes de fonctionnement ainsi que celui du mode normal est donne en figure III.6

fig III.5 HORLOGE PROGRAMMABLE

schema de cablage des

timers



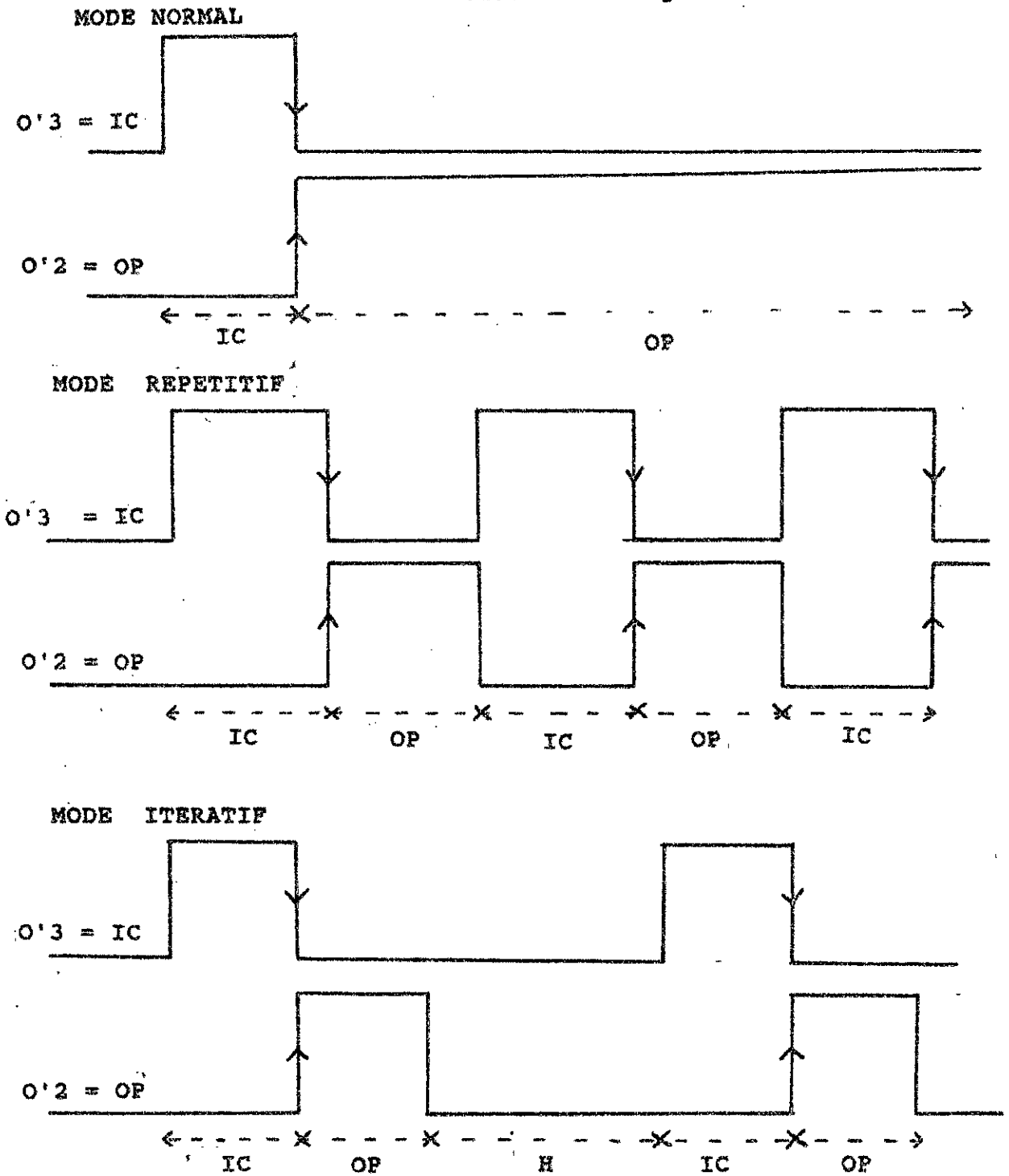
Chaque timer est constitue de trois compteurs

Ni est un nombre binaire programme sur 8 bits ou 16 bits.

Chaque compteur est programme soit en mode continu

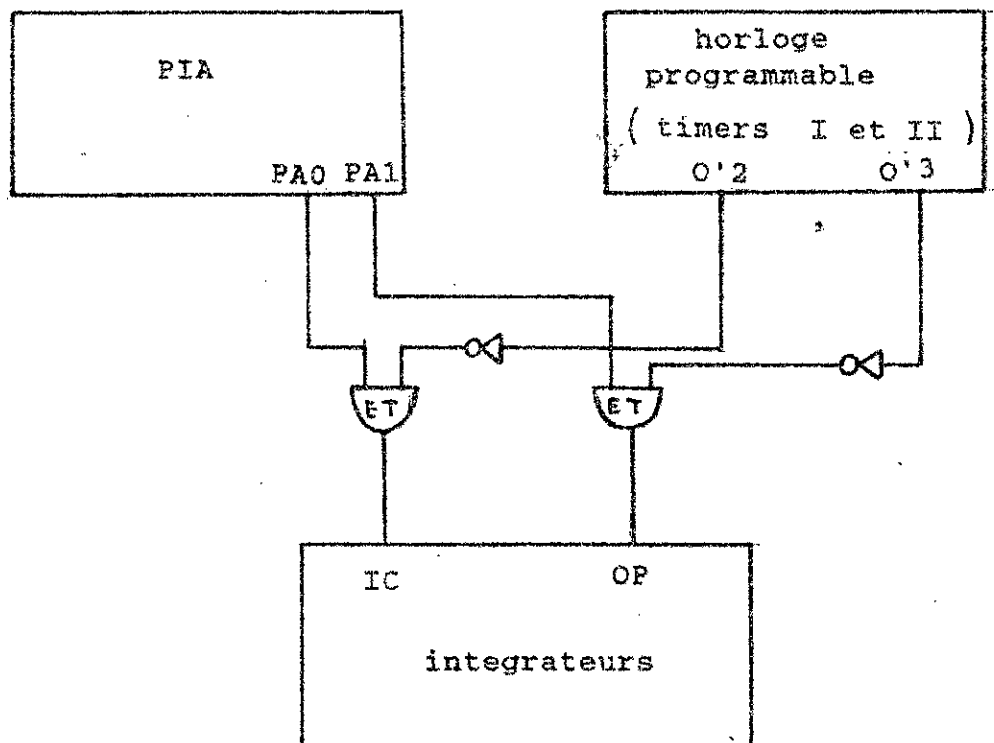
(multivibrateur) ou en mode impulsif (monostable)

fig (III.6) Chronogramme des modes de fonctionnement du  
calculateur hybride



Les durees IC et OP peuvent etre modifiees par programme.  
Dans notre cas ces durees peuvent varier de la  
microseconde a quelques dizaines de secondes .

Le schema de commande des entrees IC et OP des integrateurs est le suivant :



Dans le cas du mode normal, les timers sont inhibes. Les sorties O'2 et O'3 sont a zero. Les lignes PA0 et PA1 du PIA sont programmees de sorte a obtenir les etats logiques decrits dans le chronogramme 5 .

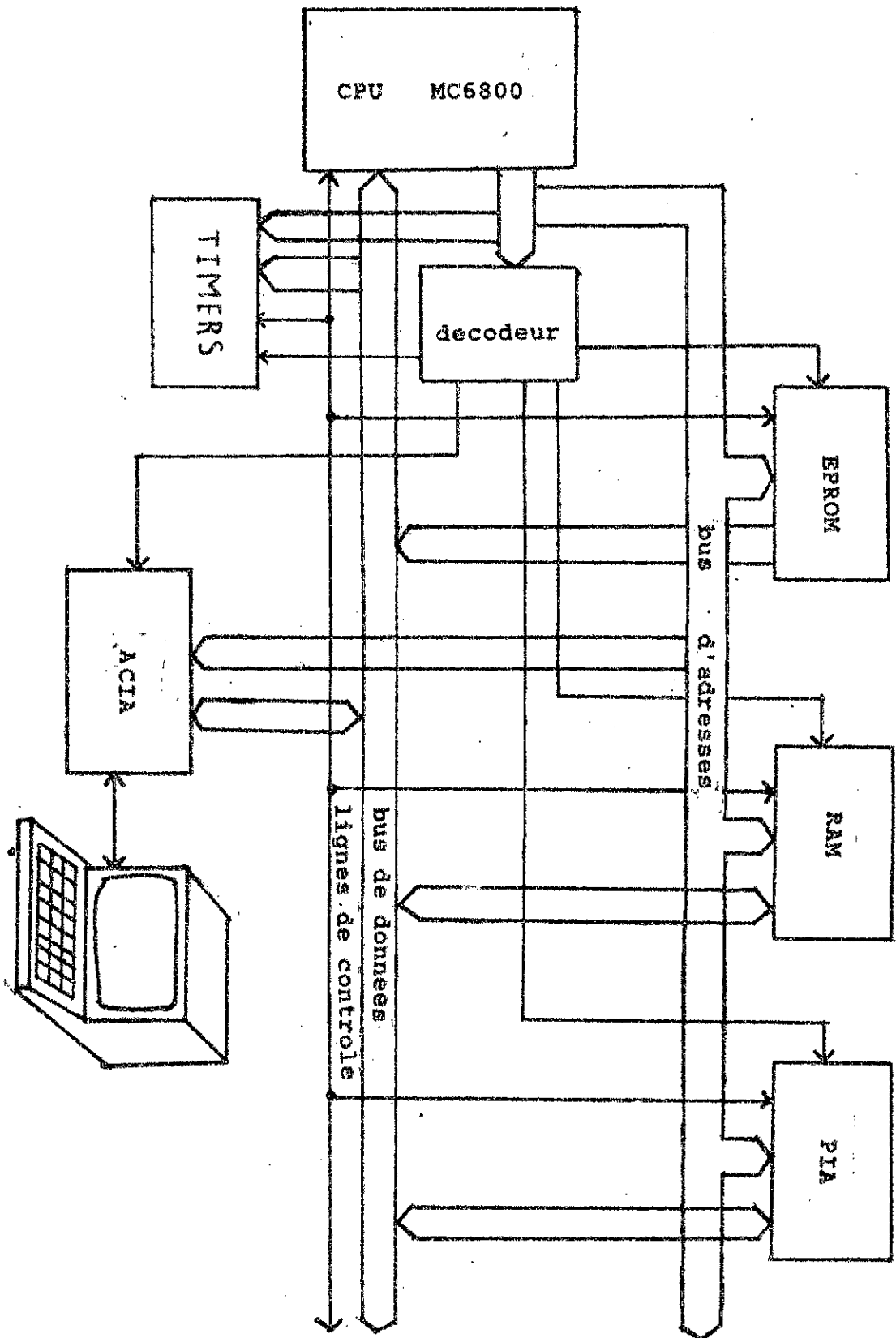
Dans le cas des modes repetitifs et iteratifs , les timers sont programmes de sorte a commander les entrees IC et OP suivant le chronogramme 5 .

Les lignes PA0 et PA1 du PIA sont a 1 .

Le synoptique de la carte microcalculateur est donne fig  
III.7 .

La configuration du calculateur hybride est donnee fig  
III.8 .

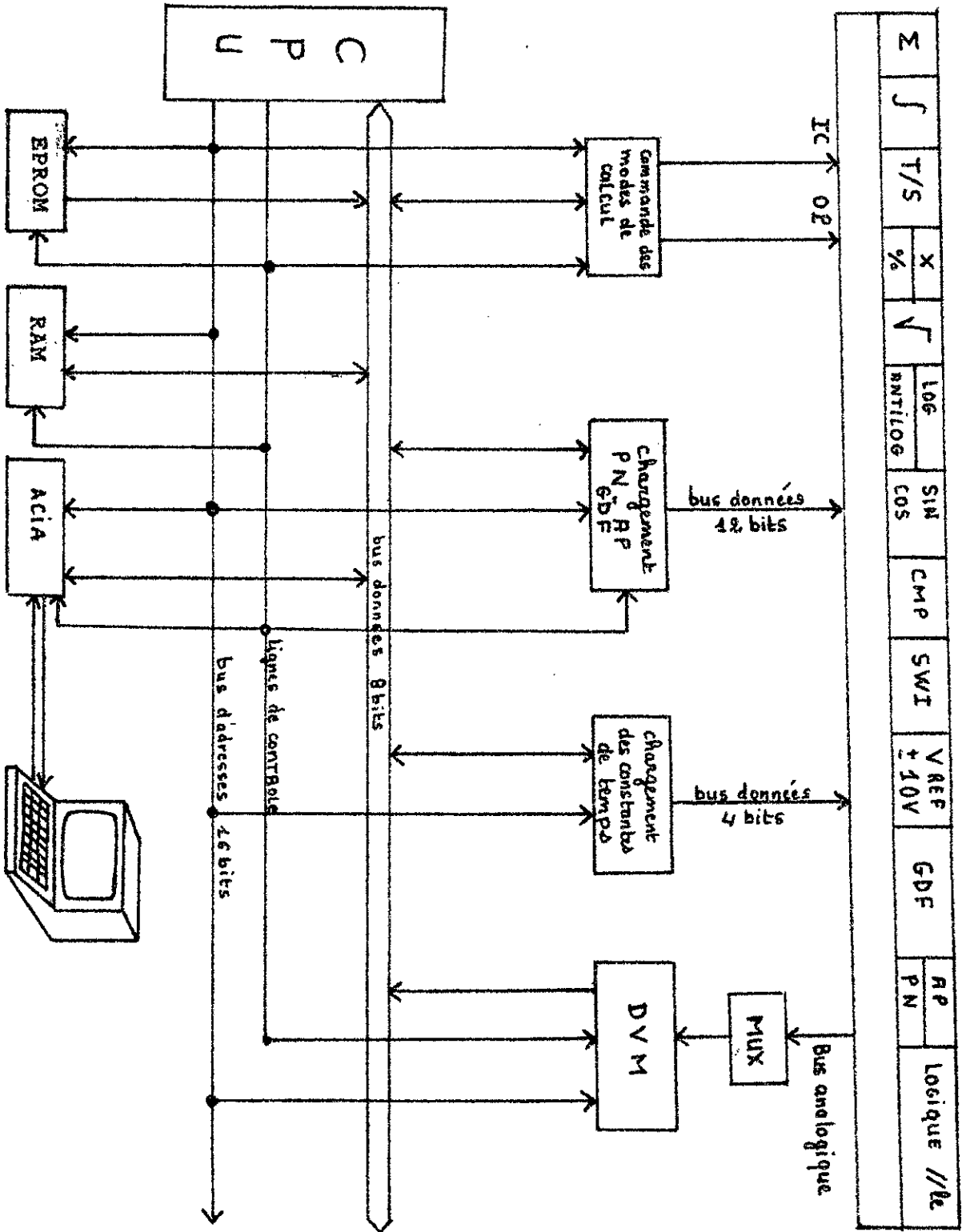
(fig III.7) Synoptique de la carte unite centrale





Configuration du calculateur hybride

fig III.8



### III.3 Description du logiciel developpe

---

#### III.3.1 Introduction :

---

Le logiciel que nous avons developpe se compose de deux parties distinctes :

- Une partie destinee a la gestion des differentes ressources dont a besoin le systeme (ces ressources peuvent etre aussi bien materielles que logicielles). Cette partie est formee d'un ensemble de programmes realisant les fonctions de base d'un systeme d'exploitation multitaches temps reel.

Nous appellerons cet ensemble de programmes : moniteur ou noyau.

- La seconde partie est formee de programmes correspondant aux specifications de l'application a laquelle est destine ce systeme.

Nous avons dans notre cas, effectue un decoupage de l'application en un certain nombre de taches. Ces taches sont celles necessaires a la resolution d'un systeme d'equations differentielles.

La premiere partie de ce chapitre est consacree a la description du moniteur developpe.

Dans la deuxieme partie, on decrira les fonctions realisees par les differentes taches en faisant apparaitre l'aspect interactif entre l'utilisateur et la machine ainsi obtenue.

### III.3.2 Structure et fonctionnement du moniteur

---

#### III.3.2 a Structure du moniteur

---

Le moniteur que nous avons developpe est un ensemble de programmes destines a gerer entre autre un certain nombre de taches qui cooperent dans le but d'automatiser la gestion du calculateur analogique.

La structure logicielle de ce moniteur se presente sous forme de couches successives.

A chacune des couches correspond un certain nombre de programmes realisant des fonctions distinctes d'une couche a l'autre.

La couche la plus profonde realise les operations de base du moniteur. Elle est la plus proche du materiel.

Lorsqu'on passe d'une couche a l'autre en s'eloignant du materiel, les operations correspondantes deviennent de plus en plus complexes. Elles utilisent les operations des couches inferieures. On distingue les couches suivantes :

- Initialisation du systeme.

C'est un programme permettant

- \* d'une part l'initialisation du materiel

- \* d'autre part l'initialisation du logiciel

- Les primitives de synchronisation P et V

- La gestion des interruptions (materielles ou logicielles)

- L'ordonnancement des taches ( Scheduling)

D'autres couches peuvent être développées à partir de celles citées précédemment.

### Objets manipulés par le moniteur

Les opérations du moniteur que nous avons conçu sont réalisées par la manipulation de quatre types d'objets :

- Les tâches
- Les événements
- Les sémaphores
- Les files

#### A / Les tâches

Les tâches manipulées par tout moniteur temps réel sont de deux types :

on distingue :

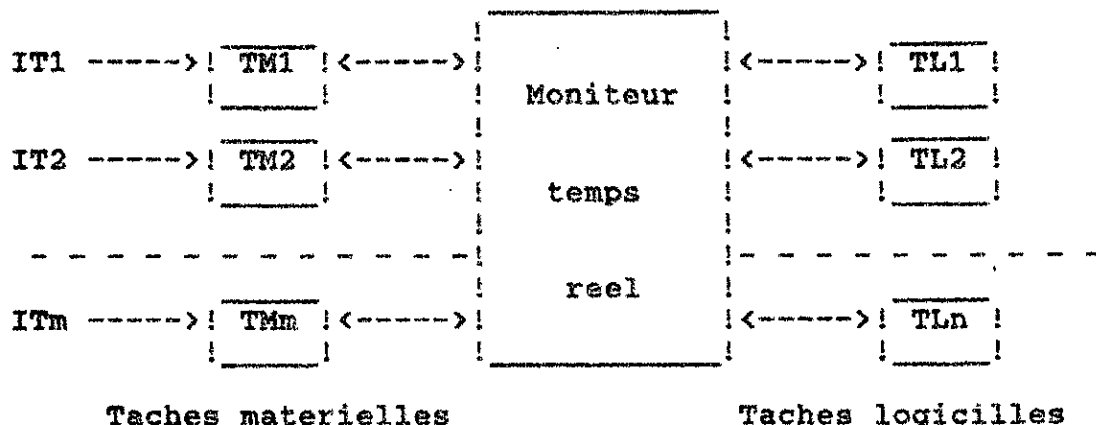
##### \* Les tâches matérielles (ou immédiates)

Elles sont liées à l'extérieur et ont des contraintes temps réels. Ces tâches sont associées à des interruptions matérielles et elles sont gérées par le système d'interruption de la machine (priorité, masquage, validations, branchement)

##### \* Les tâches logicielles (ou différées)

Ces tâches ne sont pas directement liées à l'extérieur. On leur associe une priorité logicielle gérée par le moniteur temps réel.

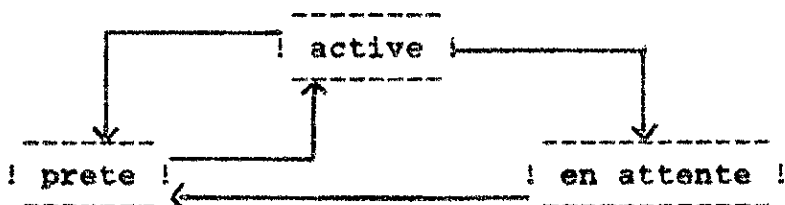
La structure d'un tel logiciel peut se resumer de la facon suivante :



+ Les etats d'une tache

Une tache de notre application peut se trouver dans trois etats : tache active, tache en attente, tache prete.

Les transitions possibles entre ces etats pour une tache du systeme peuvent etre representees de la facon suivante :



Active : Dans cet etat, l'unite de traitement lui est affectee. Il faut pour cela qu'elle satisfasse certains criteres : plus forte priorite, disponibilite des ressources qu'elle utilise.

En attente : Elle passe dans cet etat lorsqu'etant en execution elle demande une ressource deja occupee, ou lorsque pour se poursuivre elle necessite l'apparition d'un evenement.

Après disparition de la condition de blocage, cette tâche passera dans l'état prête.

Prête : Une tâche est prête lorsqu'elle est candidate à être exécutée par l'unité de traitement. Toutes les ressources autres que le processeur sont disponibles.

+ Gestion des tâches : Le passage d'une tâche d'un état à l'autre est réalisé à l'aide du module ordonnanceur qui fait appel aux opérations suivantes :

- Demarrer (une tâche) : qui consiste à allouer le processeur à la tâche lorsque celle-ci est prête.
- Arrêter (une tâche) : qui consiste à suspendre l'exécution d'une tâche.

Dans ce cas, deux possibilités peuvent se présenter :

1 ) La tâche passe de l'état active à l'état prête lorsqu'elle reçoit une interruption d'une tâche plus prioritaire.

2 ) La tâche passe de l'état active à l'état en attente lorsqu'elle sollicite une ressource dont elle ne peut disposer. Cette ressource en effet, peut ne pas être libre. De plus cette tâche peut être en attente d'un événement.

- Continuer : Pour qu'une tâche suspendue puisse reprendre son exécution, il faut qu'elle reçoive un signal de reactivation. Ceci est réalisé à l'aide de l'opération "continuer".
- Se terminer : Une tâche en exécution se terminera par une requête Fin de tâche adressée au moniteur.

De ce fait, la tache courante est retiree au processeur.

Elle mise dans l'etat prete.

Toute reactivation de cette tache necessitera l'operation "Demarrer".

B / L'evenement :

Un evenement agit sur un objet contenant une marque representant le fait qu'une condition attendue par une tache est realisee ou non.

Cet objet est materialise par un semaphore.

L'evenement agit sur un semaphore a l'aide de deux primitives P et V.

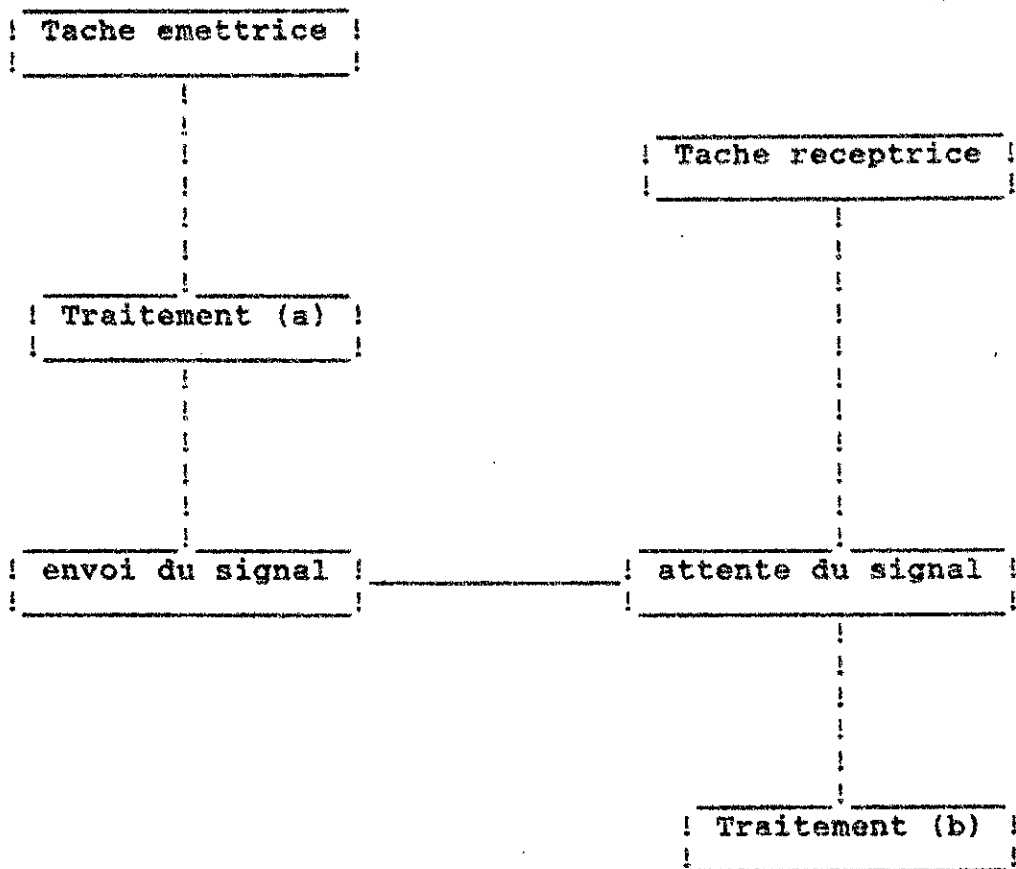
L'evenement est en fait l'outil de signalisation entre les taches.

Une tache signalant un evenement est une tache emettrice.

Si elle attend un evenement, c'est une tache receptrice.

La manipulation d'un evenement est representee fig III.9

Dans le cas ou l'evenement est externe, il correspond alors a une interruption cablee.



(fig III.9) Manipulation d'un evenement



C / Les semaphores :

Ce sont des structures de données de deux bytes destinées à régir les relations entre tâches. Ils permettent de résoudre les problèmes d'exclusion mutuelle ainsi que les problèmes de synchronisation et de communication entre tâches.

Exclusion mutuelle :

Dans le cas où les tâches évoluent simultanément, il arrive souvent qu'une ressource commune soit sollicitée par plusieurs tâches. On arrive donc à un conflit de partage de ressource entre tâches. La notion d'exclusion mutuelle implique un certain nombre de propriétés.

- A tout instant, une tâche au plus peut utiliser la ressource.
- Si plusieurs tâches sont bloquées en attente d'une ressource libre, l'une de ces tâches doit pouvoir accéder à cette ressource.

Pour résoudre ce genre de problèmes on définit un semaphore associé à cette ressource.

L'un des bytes contient une variable indiquant si la ressource est libre ou non.

Les opérations qu'effectue une tâche sur un semaphore sont les suivantes :

- 1 ) Elle doit pouvoir consulter cette variable pour savoir si elle a accès ou non à cette ressource.
- 2 ) Elle doit pouvoir signaler la libération de cette ressource lorsqu'elle n'en a plus besoin.

Les operations que l'on vient de citer se presentent sous forme de procedures qui sont appelees par les taches considerees.

Ces procedures sont appelees primitives.

Une tache sollicitant une ressource fait appel a la primitive P.

Une tache liberant une ressource, fait appel a la primitive V.

Le detail de ces primitives sera donne plus loin.

#### Synchronisation et communication

En general, dans un systeme, aucune tache n'est isolee des autres.

En effet, a certains moments de son execution toute tache communique avec d'autres taches en echangeant des signaux de synchronisation et eventuellement des donnees.

Deux aspects fondamentaux sont a considerer :

- La synchronisation entre taches : une tache active changeant d'etat permet a une autre tache de changer elle meme d'etat.

- La communication des donnees d'une tache a une autre.

Si l'on place du point de vue systeme, il est tres commode de considerer comme des ressources les signaux de synchronisation echanges par les taches (evenements).

A ce moment la, la notion de blocage se confond avec l'absence d'une ressource necessaire a l'execution de l'instruction suivante. De ce fait on associera a ces evenements (signaux de synchronisation) un semaphore que

M0019/87A

*1 seul*

UNIVERSITE DES SCIENCES ET DE LA TECHNOLOGIE

HOUARI BOUMEDIENE ( U.S.T.H.B ) ALGER

*ea*

INSTITUT DE PHYSIQUE

T H E S E

المدرسة الوطنية المتعددة التقنيات  
BIBLIOTHEQUE — المكتبة  
Ecole Nationale Polytechnique

Présenté à l'U.S.T.H.B pour l'obtention du grade de

M A G I S T E R

Spécialité : Electronique des systèmes

par

M<sup>lle</sup> Nadia SIDAHMED

REALISATION D'UN MICROSYSTEME ET DEVELOPPEMENT DU LOGICIEL  
POUR LA COMMANDE ET L'EXPLOITATION D'UN  
CALCULATEUR HYBRIDE

Soutenue publiquement le 17 Juin 1987  
devant le jury composé de

LARIBI	Sid Ahmed	PROFESSEUR	U.S.T.H.B	PRESIDENT
SANSAL	Boualem	PROFESSEUR	U.S.T.H.B	RAPPORTEUR
OUGUINI	Rachid	Chargé de RECHERCHE	H.C.R	EXAMINATEUR
TOUMI	Redouane	M. de CONFERENCE	U.S.T.H.B	"
KACIMI	Messacud	Chargé de COURS	U.S.T.H.B	"

UNIVERSITE DES SCIENCES ET DE LA TECHNOLOGIE

HOUARI BOUMEDIENE ( U.S.T.H.B ) ALGER

INSTITUT DE PHYSIQUE

T H E S E

المدرسة الوطنية المتعددة التقنيات  
المكتبة — BIBLIOTHEQUE  
Ecole Nationale Polytechnique

Présenté à l'U.S.T.H.B pour l'obtention du grade de

M A G I S T E R

Spécialité : Electronique des systèmes

par

M<sup>lle</sup> Nadia SIDAHMED

RÉALISATION D'UN MICROSYSTEME ET DEVELOPPEMENT DU LOGICIEL  
POUR LA COMMANDE ET L'EXPLOITATION D'UN  
CALCULATEUR HYBRIDE

Soutenue publiquement le 17 Juin 1987

devant le jury composé de

LARIBI	Sid Ahmed	PROFESSEUR	U.S.T.H.B	PRESIDENT
SANSAL	Boualem	PROFESSEUR	U.S.T.H.B	RAPPORTEUR
OUIGUINI	Rachid	Chargé de RECHERCHE	H.C.R	EXAMINATEUR
TOUMI	Redouane	M. de CONFERENCE	U.S.T.H.B	"
KACIMI	Messaoud	Chargé de COURS	U.S.T.H.B	"

## REMERCIEMENTS

المدرسة الوطنية المتعددة التقنيات  
BIBLIOTHEQUE — المكتبة  
Ecole Nationale Polytechnique

Je presente l'expression de ma profonde reconnaissance a Monsieur le professeur SA. LARIBI pour m'avoir fait l'honneur d'accapter la presidence du jury de cette these.

Le present travail a ete concretise au laboratoire ROBOTIQUE du CDTA, que Monsieur A. BESSALA directeur du CDTA trouve ici l'expression de ma profonde gratitude.

J'exprime toute ma reconnaissance a Monsieur le professeur B. SANSAL pour m'avoir accueillie au sein du laboratoire electronique des systemes et pour m'avoir dirigees et encouragees tout au long de ce travail.

Mes plus vifs remerciements vont a Monsieur R. OUIQUINI responsable du laboratoire ROBOTIQUE du CDTA pour m'avoir aidee et encouragee tout au long de ce travail et pour tous les moyens qu'il a mis a ma disposition pour concretiser ce travail.

Cette etude a pu etre menee a bien grace aux nombreux conseils et a l'aide amicale de Monsieur le professeur R. TOUMI auquel j'adresse mes sinceres remerciements.

Je remercie tres chaleureusement Monsieur M. KACIMI pour m'avoir fait l'honneur d'accepter de participer au jury de cette these.

Je remercie vivement Mademoiselle A. MELIFA et Monsieur F. GUETTACHE pour l'aide immense et precieuse qu'ils n'ont pas manque de m'apporter.

Je remercie particulierement Mr M. BOUARAB pour son aide amicale et sa participation au projet.

Enfin , mes remerciements vont au personnel du laboratoire Robotique pour tout le soutien qui m'a été apporté.

INTRODUCTION

CHAPITRE I : CALCULATEURS ANALOGIQUES ET CALCULATEURS  
NUMERIQUES.

I.1 - CALCULATEURS ANALOGIQUES.

I.1.1 INTRODUCTION

I.1.2 PRINCIPE DES CALCULATEURS ANALOGIQUES

I.1.3 STRUCTURE DES CALCULATEURS ANALOGIQUES

I.1.4 MODES ELEMENTAIRES DE CALCUL

I.1.5 DEROULEMENT DU CALCUL ANALOGIQUE

I.1.6 APPLICATIONS DES CALCULATEURS ANALOGIQUES

I.1.7 ETAPES DE RESOLUTION DU CALCUL

I.1.8 CONCLUSION

I.2 - CALCULATEURS NUMERIQUES

I.2.1 INTRODUCTION

I.2.2 LANGAGES DE SIMULATION

I.2.3 STRUCTURE GENERALE D'UN PROGRAMME DE  
SIMULATION

I.2.4 TECHNIQUES DE PROGRAMMATION

I.2.5 SIMULATION INTERACTIVE

I.2.6 ARCHITECTURES PARALLELES

I.2.7 CONCLUSION

CHAPITRE II : CALCULATEURS HYBRIDES

II.1 - INTRODUCTION

II.2 - BREF HISTORIQUE

II.3 - CALCUL HYBRIDE ET CALCULATEUR HYBRIDE

II.4 - FORMES SPECIALES DE CALCULATEURS HYBRIDES

- II.5 - ORGANISATION GENERALE
  - II.6 - LOGICIEL D'UN CALCULATEUR HYBRIDE
  - II.7 - DOMAINES D'APPLICATIONS
  - II.8 - CONCLUSION
- CHAPITRE III : PRESENTATION DU CALCULATEUR HYBRIDE
- III.1 - INTRODUCTION
  - III.2 - STRUCTURE MATERIELLE
    - III.2.1 DESCRIPTION DU CHAMP ANALOGIQUE
    - III.2.2 INTERFACE HYBRIDE
    - III.2.3 CHAMP NUMERIQUE
  - III.3 - STRUCTURE LOGICIELLE
    - III.3.1 INTRODUCTION
    - III.3.2 STRUCTURE ET FONCTIONNEMENT DU MONITEUR
  - III.4 - INTERACTION HOMME MACHINE ET DEFINITION DES TACHES
    - III.4.1 INTERACTION HOMME MACHINE
    - III.4.2 DEFINITION DES TACHES
    - III.4.3 FONCTIONNEMENT DYNAMIQUE DU SYSTEME
- CHAPITRE IV : APPLICATIONS ET RESULTATS
- CONCLUSION

I N T R O D U C T I O

L'etude et l'analyse de la dynamique des systemes sont des domaines importants dans l'industrie et dans diverses disciplines scientifiques, economiques ...

Ces etudes s'averent la plupart du temps delicates sur le systeme lui meme pour des raisons de securite, de cout et de temps.

la simulation presente de grands avantages car elle permet l'etude de ces systemes de facon plus rapide et plus economique.

De plus, elle permet une analyse de l'influence des divers parametres de facon aisee.

On distingue plusieurs types de calculateurs electroniques permettant la simulation du comportement dynamique des systemes :

- le calculateur analogique : c'est le premier calculateur electronique utilise pour la simulation.

Du fait de sa structure, il est bien adapte a la simulation de processus physiques continus dont le comportement est regi par un systeme d'equations differentielles lineaires ou non lineaires. C'est une machine a traitement parallele. La vitesse de calcul des operateurs analogiques permet la simulation en temps reel de phenomenes rapides.

L'un de ces inconvenient est le manque de precision.

- Le calculateur numerique conventionnel : la diminution du temps de cycle CPU ainsi que la diminution du cycle



d'accès en mémoire centrale ont permis d'aboutir à une simulation numérique moins coûteuse en temps. Le développement de langages de haut niveau a en outre permis de rendre la programmation plus aisée en simulation numérique.

- Les calculateurs numériques spécialisés : pour répondre aux exigences en calculs dans certains domaines (météorologie, sismologie ...) il s'est avéré nécessaire d'augmenter les performances des calculateurs numériques conventionnels en élaborant de nouvelles architectures.

Ceci peut être obtenu par exemple en augmentant le nombre de processeurs de calcul.

- Le calculateur hybride : ce calculateur est issu du couplage d'un calculateur numérique et d'un calculateur analogique à l'aide d'une interface hybride.

Ce type de calculateurs a été conçu vers la fin des années 50 pour des applications spécifiques à l'aérospatial, applications pour lesquelles les calculateurs analogiques et les calculateurs numériques conventionnels présentaient des limitations.

Les applications traitées par ce type de calculateurs se sont très vite étendues à d'autres domaines tels que nucléaire, biomedical etc ...

Ils sont utilisés pour l'identification de processus, la modélisation, l'optimisation ... (13).

Le développement des microprocesseurs à faible coût a favorisé l'émergence de calculateurs hybrides dans lesquels le calculateur numérique est remplacé par un

microcalculateur : on obtient alors des calculateurs hybrides a microprocesseur.

Dans ce type de configuration, le role du microcalculateur est lie aux performances du microprocesseur choisi. Ce role serait soit d'assurer la gestion du calculateur hybride ou alors tout en assurant cette gestion, il peut intervenir dans le calcul.

L'amelioration des performances des circuits analogiques ainsi que le developpement des architectures a multiprocesseurs ont ete mis a profit pour l'elaboration de calculateurs hybrides a multimicroprocesseurs. Ces calculateurs sont utilises en particulier en tant que simulateurs de vol (34).

L'objet de notre travail porte sur le developpement d'un calculateur hybride a microprocesseur en vue de son utilisation pour la simulation de processus physiques continus.

Nous disposons de la partie analogique d'un calculateur hybride. Nous nous sommes interessees a la conception et la realisation de la partie numerique de ce calculateur. Celle-ci est concue a base d'un microprocesseur 8 bits.

Nous avons egalement developpe le logiciel permettant la commande et l'exploitation du calculateur hybride.

Le plan de notre travail est structure de la facon suivante :

- Dans un premier chapitre nous decrivons les calculateurs analogiques ainsi que les calculateurs numeriques qui sont parties integrantes des calculateurs

hybrides. Nous verrons les modes de programmation de chacun de ces calculateurs.

- Le second chapitre est consacré à la description des calculateurs hybrides et à la programmation de ces derniers.

- Le chapitre trois est destiné :

+ D'une part à la description des différentes parties de notre calculateur.

- La partie analogique déjà existante

- Le microsysteme et l'interface analogique que nous avons réalisés.

+ D'autre part, on décrira le logiciel que nous avons développé pour la gestion et l'exploitation du calculateur hybride. Nous avons traité deux exemples d'applications permettant de mettre en oeuvre l'ensemble des programmes constituant ce logiciel.

- Les résultats de ces deux applications font l'objet du chapitre quatre. L'analyse de ces résultats nous a permis de confirmer le bon fonctionnement du calculateur.

- C H A P I T R E I -

I.1 CALCULATEURS ANALOGIQUES

---

I.1.1 Introduction

---

L'etude de la dynamique des systemes physiques a toujours ete un facteur tres important dans le developpement industriel.

Cependant, pour des raisons de securite et de cout, cette etude est rarement menee sur le systeme lui meme.

C'est ainsi que la notion de simulation est apparue. Celle-ci est basee sur le principe des analogies. Il existe differents types d'analogies (22) :

- L'analogie electrique
- L'analogie mecanique
- L'analogie hydraulique
- L'analogie pneumatique

On dit que deux phenomenes totalement differents sont analogues lorsqu'ils sont caracterises par le meme ensemble d'equations (5) (12).

De ce fait la connaissance de l'evolution d'un phenomene entraine la connaissance de l'evolution des phenomenes analogues au premier.

Le choix d'une des analogies est dicte par des considerations pratiques.

L'experience menee sur le modele analogue doit permettre une etude rapide, commode, economique et precise du systeme reel.

Parmi les analogies citees precedemment, l'analogie electrique a toujours ete la plus utilisee.

Ceci vient du fait que le domaine electrique est tres accessible a l'experience (14).

### I.1.2 Principes des calculateurs analogiques a courant ----- continu : -----

On fait correspondre au systeme a etudier, un systeme electrique caracterise par des courants et des tensions continus (12).

Ceux ci sont regis par les memes equations que celles qui definissent les relations entre les grandeurs du systeme reel.

Le calculateur analogique est forme d'un ensemble de blocs operateurs elementaires effectuant chacun une ou plusieurs operations mathematiques sur des tensions continues.

Chaque operation est effectuee par rapport a une seule variable independante : le temps.

Les operations effectuees par les operateurs d'un calculateur analogique sont variees : addition, integration, multiplication, generation de fonction a une seule variable ...

Ces operateurs sont selectionnes et interconnectes de facon que les equations ainsi cablees soient les memes que celles decrivant le comportement dynamique du systeme a etudier (11) (18).

Les operateurs sont assembles au niveau du panneau de cablage ce qui permet d'obtenir le systeme analogue. Le calcul analogue est un calcul parallele car les differentes operations sont executees simultanement.

C'est la caracteristique la plus importante de ce calculateur. Le calculateur analogue est particulierement bien adapte a la resolution des equations differentielles (11) (12) (18) ainsi que des systemes differentiels d'ordre eleve.

Les relations entre les grandeurs caracterisant un grand nombre des systemes physiques se traduisent par des equations differentielles.

Pour cette raison, l'utilisation de ce calculateur est preponderante dans les etudes de la dynamique des systemes physiques sans etre limitee a cette classe de problemes. C'est le calculateur analogue qui a permis le developpement de la simulation (1).

Il a donne un sens pratique a l'elaboration des modeles d'equations differentielles.

### I.1.3 Description des calculateurs analogiques

#### I.1.3 a Element de base

L'amplificateur est l'organe essentiel du calculateur analogue.

C'est l'element de base d'un grand nombre d'operateurs. Associe a un ensemble de resistances ou de

condensateurs, il permet les operations telles que :  
somme, integration. On peut classer les differents  
operateurs en deux groupes :

### I.1.3 b Operateurs lineaires

---

Ces operateurs permettent de faire :

- La somme ou l'integrale de grandeurs evoluant en fonction du temps.
- Le produit par une constante numerique d'une grandeur evoluant en fonction du temps.

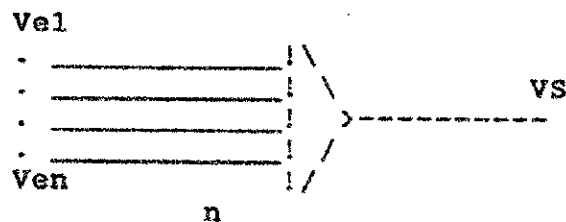
Ces operateurs sont suffisants pour resoudre des equations differentielles lineaires a coefficients constants.

Symbole de ces operateurs

---

#### 1 - Sommateur

---



$$VS(t) = - \sum_{i=1}^n Ki Vei(t)$$

$Ki = 1$  ou  $10$

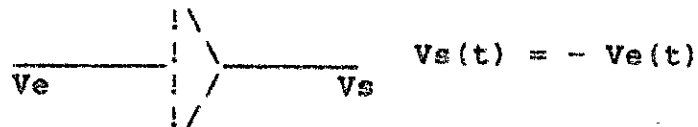
Les valeurs de  $Ki$  dependent des elements associes a l'amplificateur de base.

## 2 - L'inverseur

L'inverseur est un cas particulier du sommateur.

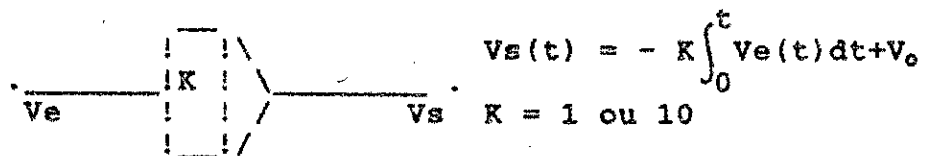
Le nombre de tensions d'entrees est reduit a un.

Symbole



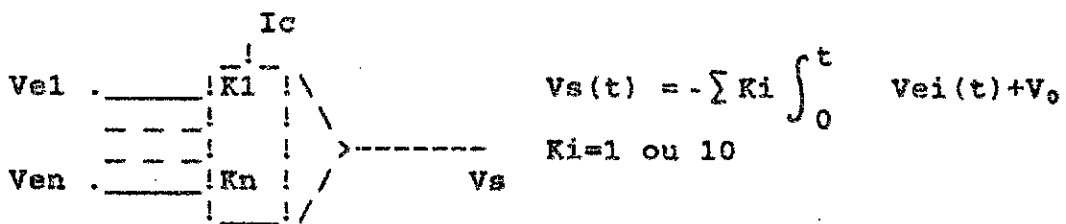
## 3 - L'integrateur

a) Symbole de l'integrateur a une entree.



Ic est une tension analogique

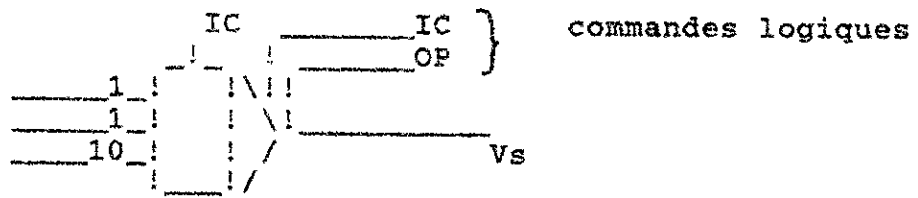
b) Symbole de l'integrateur sommateur



L'integrateur est l'element de calcul le plus important. Nous verrons le role qu'il joue dans le calcul plus loin.

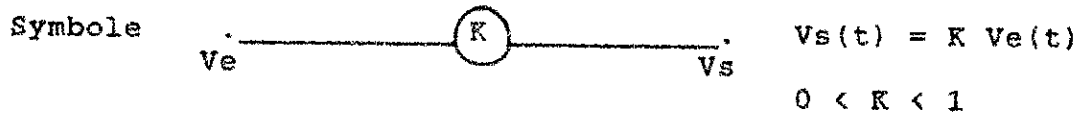
En plus des entrees analogiques, il possede deux entrees logiques IC et OP.





4 - Le potentiometre

---

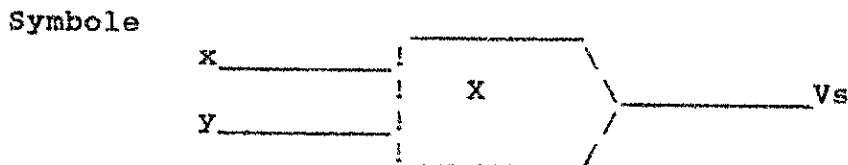


I.1.3 c Operateurs non lineaires

---

Ces operateurs permettent de faire :

- Le produit et la division de deux grandeurs
  - La generation d'une fonction du temps
  - Les multiplieurs
- 

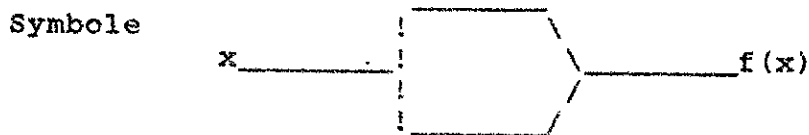


On peut avoir les deux cas suivants

$$V_s = axy$$

$$V_s = xy$$

- Les generateurs de fonctions
- 



Nous distinguons deux types de generateurs de fonctions :

1 - Les generateurs de fonctions universels :

Ces generateurs generent une fonction  $y = f(x)$  quelconque

2 - Les generateurs de fonctions specialises :

Ils generent les fonctions suivantes : sinus, cosinus, carres ... etc

On trouve aussi :

3 - Des generateurs de bruit blanc :

Utiles dans le cas d'analyse des signaux.

- Des generateurs permettant la conversion de coordonnees cartesiennes  $(x,y)$  en coordonnees polaires  $(R,\theta)$
- Des generateurs de retard variable.

Il n'est pas tres aise de realiser des retards variables de maniere purement analogique.

I.1.3 d Operateurs logiques ( ou logique parallele)

Les calculateurs analogiques ont beneficie de l'evolution de la technologie des composants de base : (miniaturisation, augmentation de la fiabilite et de la bande passante).

Cependant, c'est l'apparition d'une logique parallele qui a permis une plus grande souplesse dans la resolution de certains problemes.

En effet, une limitation des calculateurs analogiques decrits precedemment est apparue dans le cas de problemes faisant intervenir des calculs iteratifs rapides.

Ex : problemes d'optimisation des parametres ... (10)

L'automatisation de ces calculs fut obtenue en ajoutant des elements logiques qui assurent la commande du calcul (22). Les calculateurs ainsi obtenus sont appeles "iterative differentiel analyser" ou calculateur analogique iteratif (IDA) (10) .

Ex : Le calculateur EAI 180 (22)

Les operateurs logiques sont :

- 1 - Les elements sans memoire : portes logiques, inverseurs.
- 2 - Les elements a memoire : bascules, registres a decalages, compteurs.

#### I.1.4 Modes de calcul du calculateur analogique

---

##### I.1.4 a Etats ou modes possibles des operateurs de calcul

---

La resolution d'un systeme d'equations sur calculateur analogique necessite le passage par des phases differentes. Ces phases correspondent aux etats possibles des operateurs de calcul.

On distingue quatre etats ou modes possibles :

- trois etats sont directement lies a la commande des entrees logiques IC et OP des integrateurs. On les appelle mode de fonctionnement (22). - Un etat correspondant a l'etalonnage des potentiometres .

## A - Modes de fonctionnement

### 1) Mode conditions initiales (ou mode IC)

Ce mode permet l'introduction des conditions initiales du système d'équations à résoudre.

Ces conditions initiales sont introduites à travers le chargement des capacités des intégrateurs comme le montre la figure I.1.

La capacité est chargée avec une constante de temps  $RC$  à la valeur imposée au niveau de la borne d'entrée CI.

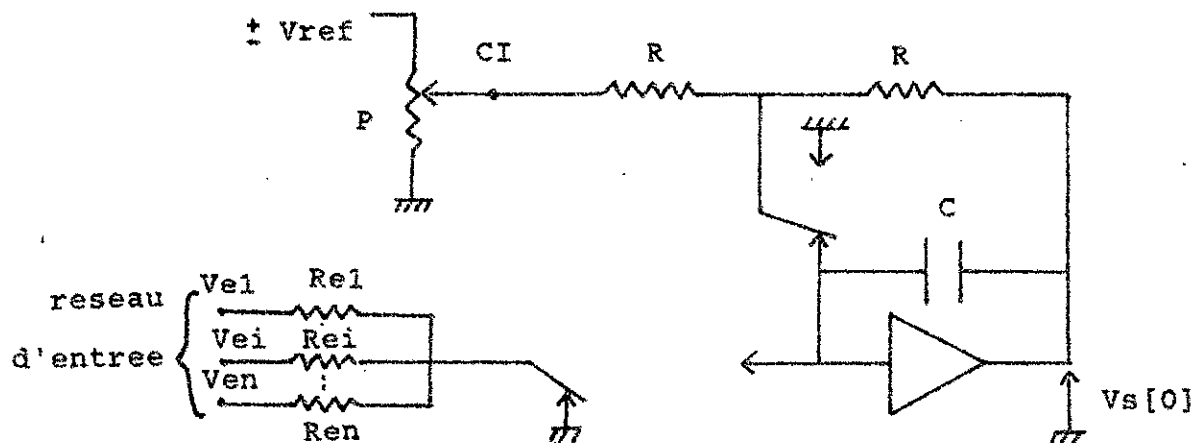


fig I.1

### 2) Mode opératoire ou calcul (mode OP)

Dans ce cas, les tensions représentant les variables du système d'équations sont appliquées aux entrées des intégrateurs.

Le système électrique analogique évolue suivant sa dynamique propre, fournissant la solution du système à résoudre.

La commande des integrateurs est representee fig I.2

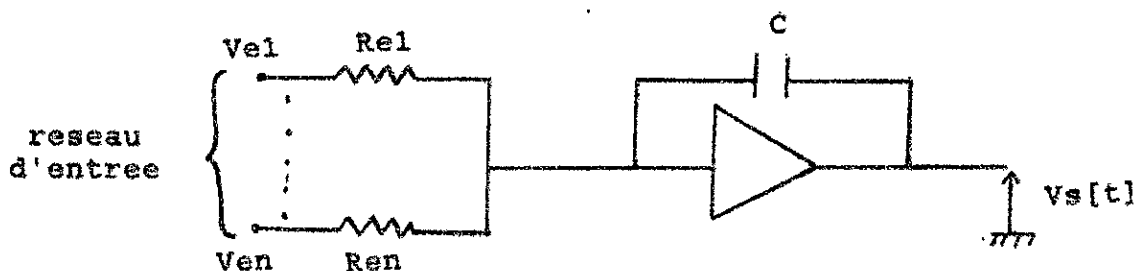


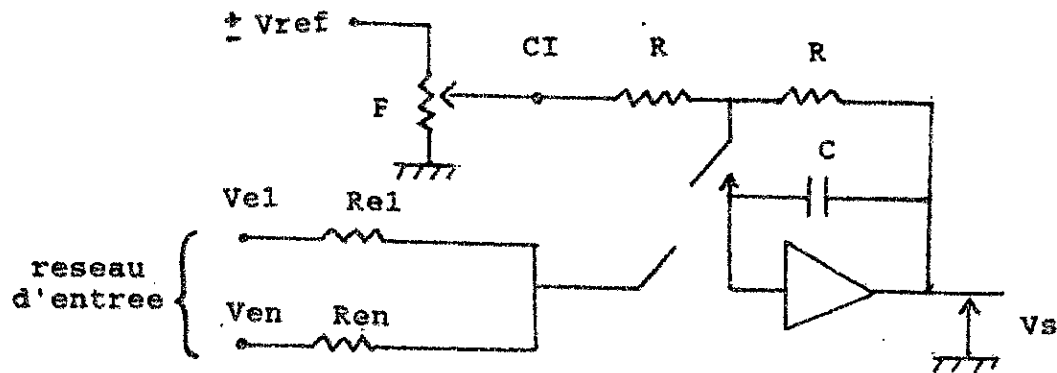
fig I.2

3) Mode memoire ou gel (mode HD)

Il est possible de suspendre le deroulement d'un calcul en le figeant dans l'etat ou il se trouve.

Ce mode permet la lecture de toutes les valeurs prises par les variables a cet instant.

La commande des integrateurs est representee fig I-3.



figI-3

Ces trois modes sont rendus possibles par l'intermediaire de signaux logiques fournis par les operateurs logiques.

Nous pouvons resumer ces trois modes par le tableau suivant :

Entree IC	Entree OP	Mode
0	1	OP
1	0	IC
0	0	non utilise
1	1	HD

### B - Mode etalonnage des potentiometres (mode PS)

---

C'est aussi le mode arret.

Avant de lancer un calcul, il est necessaire d'introduire les differentes valeurs des parametres du systeme d'equations a resoudre.

Ceci est realise par reglage des potentiometres.

Dans ce mode, les entrees de tous les amplificateurs sont a la masse pour eviter qu'ils se saturent.

Seules les entrees des potentiometres sont connectees a la tension de reference.

La lecture des tensions de sortie des potentiometres permet alors de fixer les valeurs des coefficients.

En plus du reglage de ces coefficients, un certain nombre d'autres reglages peuvent s'averer necessaires.

- Affichage des generateurs de fonctions.
- Selection des constantes d'integration
- Initialisation de la logique parallele :  
bistable, registre, horloge ...

### I.1.5 Modes de déroulement du calcul analogique

Suivant la façon dont se succèdent les modes de fonctionnement on distingue trois modes de calcul possibles :

#### I.1.5 a Mode normal

C'est la succession des modes IC et OP selon le chronogramme suivant :



Dans ce mode on distingue deux phases.

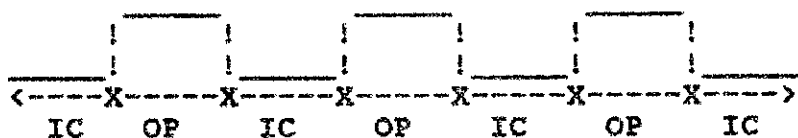
#### - Phase IC :

Celle-ci correspond à l'introduction des conditions initiales. Ceci est effectuée après avoir ajusté les valeurs de tous les paramètres intervenant dans le calcul. Lorsque les conditions initiales sont chargées (durée de IC) on passe à la phase calcul.

Durant cette phase, les opérateurs intervenant dans le calcul sont connectés entre eux, et les tensions évoluent de la même façon que les grandeurs physiques du système à étudier.

#### I.1.5 b Mode répétitif

Chronogramme



Certains problemes necessitent la repetition automatique des calculs.

Ceci est realise par la succession rapide des modes IC et OP.

La logique parallele est programme de sorte a commander la repetition de ces phases.

Pendant la duree IC, les conditions initiales sont rechargees et pendant la duree OP, les operateurs remplissent leur fonction de calcul avec des conditions initiales modifiees ou non.

Les resultats peuvent etre suivis sur oscilloscope.

L'interet de ce mode de calcul est de permettre l'observation immediate de l'effet de la variation d'un parametre sur la solution.

Les calculateurs analogiques les plus perfectionnes sont dotes d'une programmation automatique de modification d'un parametre.

#### I.1.5 c Mode iteratif

Chronogramme



La difference entre ce mode de calcul et le precedent vient du fait que la phase OP utilise les resultats du calcul precedent. Ceux-ci sont stockes durant la phase HD qui ne figure pas dans le mode repetitif.



Le mode iteratif permet la convergence du calcul vers la solution desiree.

#### I.1.6 Applications des calculateurs analogiques

---

La classe de problemes que l'on peut cabler sur calculateur analogique est celle relevant d'une formulation qui se presente sous forme d'un systeme d'equations comprenant :

- Des equations differentielles ordinaires (lineaires ou non).
- Des relations algebriques
- Eventuellement des relations logiques (21)

Ce sont des systemes d'equations a une seule variable independante.

Cependant, les calculateurs analogiques ne se limitent pas uniquement a ce type de problemes.

En effet, ils permettent la resolution de systemes d'equations aux derivees partielles.

Ces equations presentent des difficultes du fait que l'on ne dispose sur le calculateur analogique que d'une seule variable independante d'integration : le temps machine.

Certaines methodes permettent de pallier a cette difficulte :

On peut en citer quelques unes :

- Methode des differences (5)
- Methode des caracteristiques
- Methode des fonctions propres ...

Les calculateurs analogiques ne sont pas tres adaptes a la resolution des equations algebriques. En effet, on constate dans les boucles algebriques des instabilites entrainant une divergence de la solution (5) (10) (21). On essaie de faire disparaitre ces phenomenes en disposant des constantes de temps dans les boucles de calcul.

#### I.1.7 Etapes de resolution sur calculateur analogique

(ou programmation du calculateur analogique)

Le traitement sur calculateur analogique necessite le passage par un certain nombre d'etapes constituant la programmation du calculateur.

Avant d'aborder cette programmation deux etapes sont necessaires :

1) Definition du modele mathematique du phenomene a etudier en alliant connaissance et experience. Ce modele est constitue d'un systeme d'equations accompagne d'une liste de coefficients, de parametres ainsi que de resultats de mesure.

2) Transformations de ce modele sous forme assimilable par la machine c'est a dire permettant de le mettre sous forme canonique.

Les transformations effectuees ne doivent evidemment pas affecter le modele.

## Programmation du calculateur

---

A ce niveau, nous disposons d'un système d'équations canoniques :

$$\frac{dx_i}{dt} = f_i(x_1, x_2, \dots, x_n)$$

$i = 1, 2, \dots, n$

### 1) Ecriture des équations machines

---

On établit une correspondance entre les unités physiques et les unités machines (volts et secondes).

Ceci nous permet de transformer le système d'équations initial en équations machines.

Pour cela, nous devons effectuer les mises à l'échelle suivantes :

#### a) Mise à l'échelle en amplitude

---

Les variables du problème sont représentées sur la machine par les tensions de sortie des amplificateurs.

Pour éviter la saturation de ces derniers, la connaissance des valeurs maximales de chacune des variables  $X_i$  est nécessaire

#### b) Mise à l'échelle en temps (ou choix du temps machine)

---

La variable indépendante des équations canoniques peut représenter le temps, l'espace ou toute autre grandeur. On lui fait correspondre le temps machine par la relation  $\tau = \beta t$

$\beta$  est appelé facteur d'échelle

On calcule a l'aide des equations machines les grandeurs de sortie de tous les operateurs concernes en les supposant parfaits.

5 ) Essais statiques  
-----

On procede aux verifications statiques en utilisant les resultats precedants.

Pour eviter des problemes d'instabilite, les boucles algebriques seront ouvertes.

6 ) Controle dynamique  
-----

Un controle dynamique permet la verification des constantes d'integration.

7 ) Mise en conditions initiales  
-----

On ajuste toutes les valeurs des conditions initiales a leur valeur reelle.

8 ) Passage en calcul  
-----

9 ) Phase d'enregistrement des resultats  
-----

Un organigramme resumant toutes ces etapes est represente page (18.b)

Il est important de noter que le temps de preparation augmente avec les dimensions du probleme.

En effet, le nombre d'operateurs intervenant dans le cablage augmente ce qui entraine un plus grand nombre de verifications.

Cependant, les dimensions du probleme n'affectent en rien le temps d'execution.

- Si l'on veut effectuer une étude du phénomène en temps réel on prend  $\beta = 1$

- Si le phénomène à étudier est trop rapide, on a la possibilité de le ralentir en prenant  $\beta > 1$

- Dans le cas de processus trop lents, il est nécessaire d'accélérer le phénomène pour effectuer son étude, ceci est possible en prenant  $\beta < 1$ .

On aboutit ainsi aux équations machine

## 2 ) Etablissement du schéma analogique

On dessine le schéma de principe représentant ces équations en utilisant les symboles des blocs opérateurs:

- Les variables seront élaborées par des amplificateurs
- Les constantes par des potentiomètres

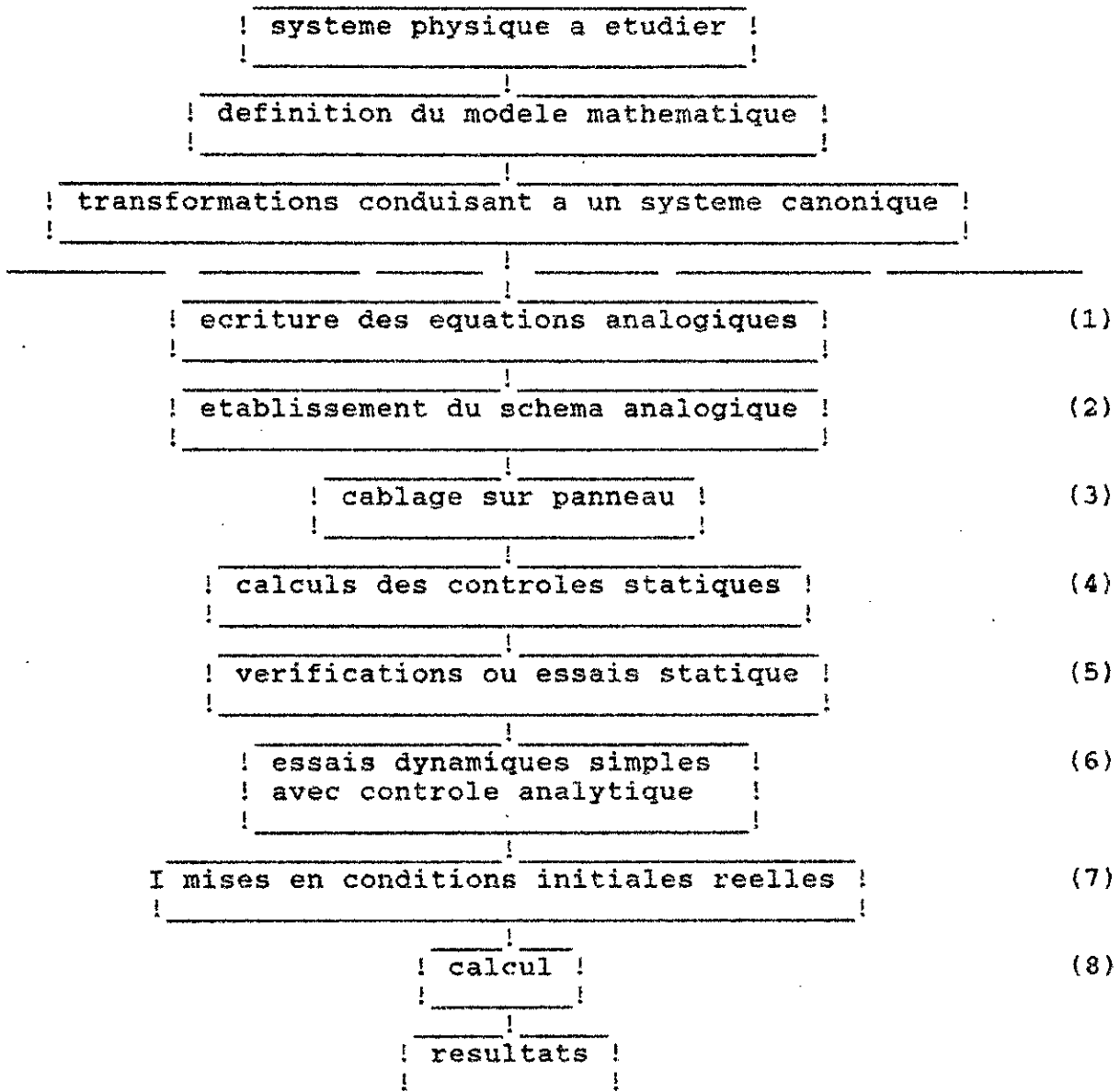
3 ) En s'aidant de ce schéma, on réalise le câblage sur le panneau de la machine

L'avantage de cette programmation vient du fait que le manipulateur ne perd pas de vue le sens physique du phénomène qu'il étudie.

## 4 ) Calcul du contrôle statique

On procède à un certain nombre de calculs permettant le contrôle statique.

Pour cela on choisit au préalable un lot de conditions initiales arbitraires et un nombre de valeurs de variables d'entrée permettant de fournir des sorties non saturées.



Organigramme des etapes de resolution sur calculateur analogique

---

En effet, le calculateur étant une machine à traitement parallèle, tous les blocs impliqués dans le calcul opèrent de façon simultanée.

Nous voyons donc que la programmation du calculateur analogique ne nécessite aucun langage particulier.

Cependant, il existe des programmes numériques qui fournissent une aide à la programmation des calculateurs analogiques. On peut citer le code APACHE qui est l'un des plus connus. Il a été conçu sur un calculateur analogique (SAI 231 R PACE) et un calculateur numérique IBM 360 (11).

Celui-ci existe en plusieurs versions : (21)

Il permet dans sa version la plus simple :

- D'établir les équations machines.
- De fournir les valeurs d'étalonnage des potentiomètres
- Le test statique

Ce qui permet de libérer l'analogiste de calculs longs et fastidieux.

Le calculateur analogique est soumis à des limitations qui sont liées à sa précision médiocre en statique  $\approx 10^{-4}$  de la pleine échelle.

Cette précision se dégrade en dynamique, surtout si les valeurs des facteurs d'échelle qui sont imposées par la plage de linéarité des amplificateurs sont mal choisies (20).

### I.1.8 ) CONCLUSION

C'est le calculateur analogique qui a donné un sens pratique à l'élaboration des modèles d'équations différentielles (1).

Il a permis le développement de la simulation.

Les premiers calculateurs analogiques furent conçus vers les années 40 pour des applications spécifiques à l'aérospatiale, simulateurs de vol en particulier.

Ce n'est que vers les années 50 qu'ils ont atteint leur statut d'outils universels [2].

Les techniques de calcul analogiques ont été très employées car elles seules permettaient le calcul de grandeurs mettant en jeu de nombreuses opérations (22).

Cependant, l'application majeure de ces calculateurs est restée la simulation pour laquelle ils ont joué un très grand rôle.

Le développement des grands calculateurs numériques a créé un potentiel pour la résolution de problèmes mathématiques de complexité illimitée. Ceci a eu pour effet de réduire le champ d'application des calculateurs analogiques.

De plus, vers les années 60, avec la création des langages de simulation le rôle de ces calculateurs a diminué (21).

La conception de ces machines a donc été rationalisée et leur automatisation a été poussée.



Le calculateur analogique est encore utilise dans certaines applications recentes (19).

Plus particulierement, les automaticiens le considerent toujours comme un outil bien adapte a leurs problemes(5) qui sont resolu et plus facilement mis en oeuvre que par l'emploi d'un calculateur numerique [22].

Le poste de controle et de commande des calculateurs analogiques a ete remplace par un microprocesseur ce qui fut un premier pas vers les calculateurs hybrides.

## Calculateurs Numeriques

---

### I.2.1 Introduction

---

Le developpement des calculateurs numeriques s'est effectue de facon extraordinaire grace a l'evolution de la technologie des semiconducteurs .

De ce fait , le domaine d'application couvert par ces calculateurs est devenu de plus en plus vaste .

Cette evolution a permis d'aborder avec les calculateurs numeriques un type de problemes qui etaient alors reserves aux calculateurs analogiques : la simulation des processus physiques continus (5) .

### I.2.2 Langages de simulation

---

Durant les annees 50 , la puissance des calculateurs analogiques ainsi que leur facilite de programmation en faisaient les outils privilegies pour la simulation par rapport aux calculateurs numeriques existants .

L'evolution de la technologie a permis d'augmenter les performances des calculateurs numeriques de facon considerable (6) . En effet , on note une augmentation de la vitesse d'execution des operations au niveau du processeur ainsi qu'un acces plus rapide aux memoires .

Entre les annees 50 et 60 , de tres grands progres ont ete realises dans l'aspect theorique de la resolution numerique des equations differentielles .

C'est ainsi que , vers les années 60 , se sont développés les premiers langages de simulation numérique pour systèmes continus (2) .

Ces premiers langages étaient écrits en Assembleur . L'apparition des langages évolués tels que FORTRAN , ALGOL a permis de simplifier l'écriture de ces langages .

### 1.2.3 Structure générale d'un programme de simulation

---

Un tel programme est constitué de 3 fonctions de base :

1) Initialisation : Trois classes de quantités sont généralement initialisées :

- a ) Les valeurs initiales des variables d'état
- b ) Les valeurs des paramètres du problème
- c ) Les constantes telles que TMAX qui est la durée de la simulation .

2) Execution : c'est la partie dynamique du problème. C'est la phase principale du programme .

Les équations du problème sont utilisées pour générer la solution .

La commande d'exécution fait appel à l'algorithme d'intégration choisi .

3) Phase finale : Elle fait appel aux fonctions suivantes :

- a ) Sortie complète des solutions
- b ) Les résultats sont testés pour savoir si on doit lancer une autre exécution ou si on doit arrêter la simulation .

L'avantage offert par les langages de simulation numerique est l'affranchissement des problemes d'echelles sur les variables .

#### I.2.4 Techniques de programmation

---

Il existe plusieurs techniques de programmation des langages de simulation . On peut citer :

- Les langages " adaptes aux equations"
- Les langages a structure de blocs

a ) Langages adaptes aux equations : Ils permettent l'introduction du probleme sous forme d'equation d'etat .

b ) Langages a structure de blocs : Ce sont des techniques de simulation a hautes performances . Chaque instruction de ce type de langage joue le meme role qu'un operateur elementaire appartenant a un calculateur analogique . Ce type de langage permet d'augmenter la vitesse d'execution dans le cas d'un systeme de simulation sur minicalculateur (3) .

L'un des premiers langages utilisant la structure de blocs est DARE/ELEVEN .

Celui ci combine des segments de programmes ecrits en langage d'equations (representation en virgule flottante) avec des segments de programmes ecrits en langage de blocs .

Ces blocs representent les " composants " rapides du systeme a simuler exactement de la meme facon qu'un calculateur hybride combine des calculs analogiques .

### I.2.5 Simulation interactive

---

La simulation interactive des systemes permet a l'experimentateur d'introduire et d'editer ses programmes et donnees a partir d'un clavier .

Il a alors la possibilite d'observer les solutions sous forme de graphes ou autres des que celles ci sont calculees .

Il peut modifier les parametres et le modele en ligne sur la base des resultats les plus recents .

De plus , il peut etudier immediatement l'effet de ces modifications en lançant une autre execution .

Exemple de langage interactif :

DARE/ELEVEN adapte au minicalculateurs PDP 11

DARE PI adapte aux supermini de 32 bits serie VAX

### I.2.6 Architectures paralleles

---

La technologie et les techniques de programmation ont evolue de facon considerable permettant ainsi d'elargir le champ d'application des calculateurs numeriques conventionnels ( VAN NEUMANN ) .

Cependant , les performances , necessaires pour la resolution de certains problemes tels que : Meteorologie, Sismologie ... , n'ont pu etre atteintes avec ces calculateurs .

En effet , ces problemes sont caracterises par un algorithme complexe , un volume de donnees considerable tout en etant soumis a des contraintes de temps .

Leur resolution necessite des moyens permettant un traitement simultane d'operations .

Les recherches ont donc ete orientees vers une nouvelle organisation logique des calculateurs numeriques pour aboutir a une architecture caracterisee par plusieurs processeurs .

On parle alors d'organisation a multiprocesseurs . La puissance de calcul de ce type de calculateurs est superieure a celle des calculateurs numeriques conventionnels .

Il existe une tres grande variete d'architectures a multiprocesseurs . Le choix de l'une d'elle est dicte par des considerations pratiques .

Un exemple simple d'architecture d'un systeme a multiprocesseurs pour la simulation rapide de systemes continus de grande taille est donne fig I.4.[2] [3]

Le PDP-11/40 est le processeur de controle ( ou principal ) . Il controle de 1 a 16 processeurs arithmetiques tres rapides ( processeurs secondaires )

A chaque processeur secondaire est associe une memoire dans laquelle sont stockes les programmes ainsi qu'une routine d'integration .

La fonction de chaque processeur est d'integrer un sous ensemble du systeme d'equations .

A chaque pas d'integration , le processeur principal est interrompu par les processeurs secondaires de sorte a permettre le transfert des variables necessaires a d'autres processeurs pour continuer le calcul.

-----  
 Processeur de commande pour les E / S, l'unité disque et  
 pour les communications inter processeurs  
 -----

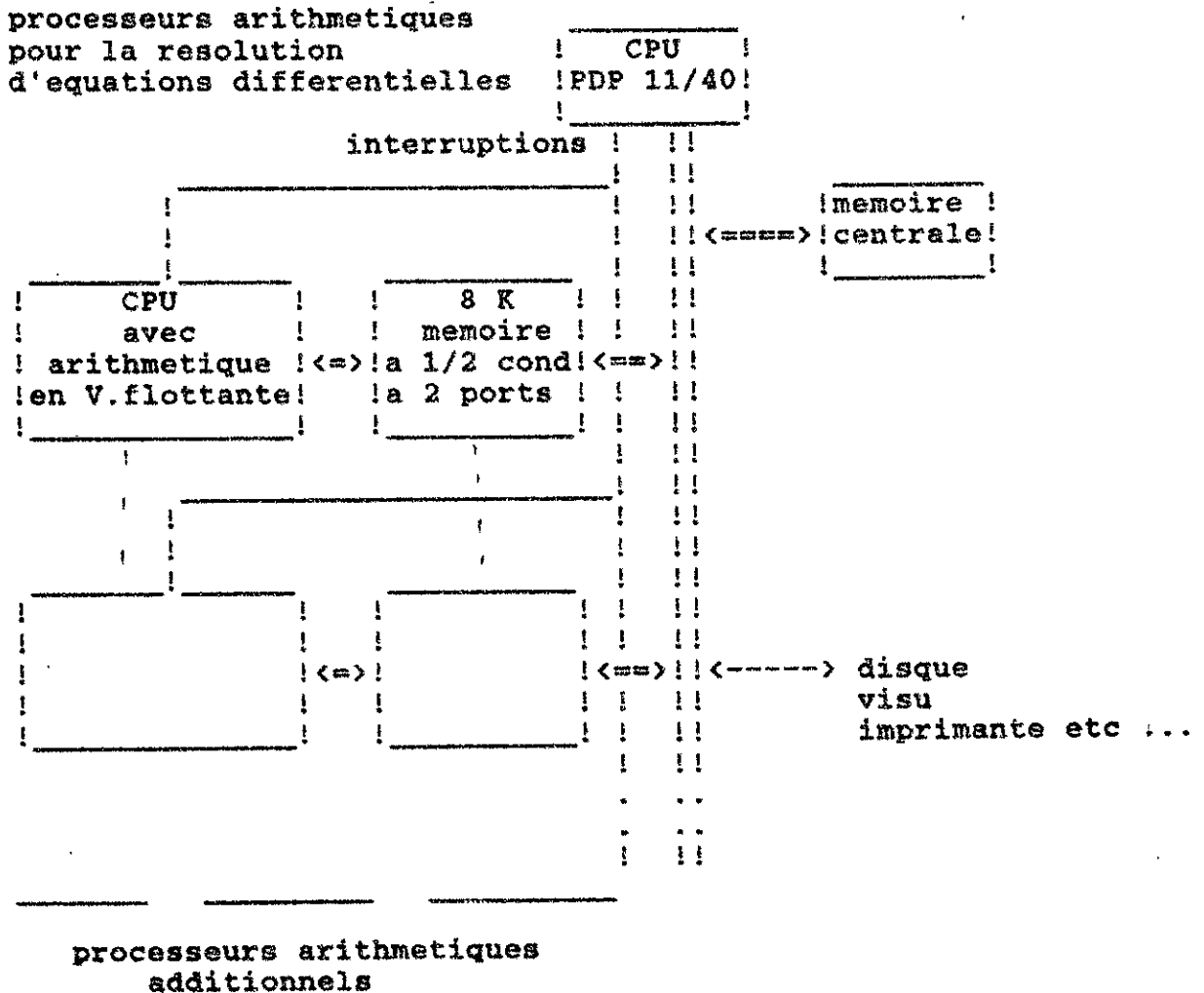


fig 1.4 Architecture d'un systeme a multiprocesseurs  
 -----

### I.2.7 Conclusion

La simulation numerique offre beaucoup de facilites de mise en oeuvre grace aux techniques de programmation que nous avons citees.

Elles permettent a l'utilisateur d'eviter les contraintes liees a la programmation .

Les architectures a multiprocesseurs presentent un interet particulier car il est souvent possible de diviser le calcul en parties independantes qui peuvent etre effectuees simultanement [7]. Cependant, un fort couplage entre les systemes d'equations qui sont affectes aux differents processeurs constituant l'architecture , fait perdre de son interet a cette derniere . En effet , dans ce cas un temps important est consacre aux echanges interprocesseurs .

De plus , la mise en oeuvre sur un systeme a multiprocesseurs peut presenter des difficultes quant au partitionnement des segments de programmes entre les differents processeurs .



- C H A P I T R E II -

CALCULATEURS HYBRIDES

---

I.1 ) INTRODUCTION

---

Necessite d'une conception hybride des calculateurs

---

Les outils de calcul ont ete developpes en fonction de la nature des problemes que les scientifiques etaient amenes a resoudre tout en etant conditionnes par l'essor de la technologie.

Les besoins en simulation ont mene au developpement des calculateurs analogiques et ceux de resoudre des systemes algebriques de plus en plus complexes ont mene au developpement des calculateurs numeriques.

Le couplage entre un calculateur analogique et un calculateur numerique a ete recherche par les specialistes du calcul analogique qui souhaitaient augmenter la flexibilite d'une simulation en utilisant la memoire et la logique d'un calculateur numerique (15). De plus, l'apport des calculateurs dans le developpement industriel est essentiel.

Les systemes industriels ont un caractere hybride : Ils sont constitues d'un systeme dynamique evoluant continuellement associe a divers automatismes et organes logiques de commande et de controle.

Leur simulation etait au depart impossible a l'aide des calculateurs analogiques ou des calculateurs numeriques existants.

Ceci a conduit tout naturellement a l'elaboration de calculateurs hybrides.

Ceux-ci sont obtenus en associant les deux moyens de calculs deja existants a l'aide de circuits d'interface appropries.

Cette organisation est tres interessante car elle permet de combiner les qualites des deux calculateurs a savoir :

- Rapidite d'integration de l'analogique et calcul parallele.
- Grande capacite memoire et grande precision du numerique.

## II.2 ) Bref historique

C'est en 1958 que furent realises les premiers calculateurs hybrides, issus de la liaison d'un calculateur analogique et d'un calculateur numerique dans les laboratoires d'etudes spatiales a LOS ANGELES et en CALIFORNIE (8) (12) (13).

Ces calculateurs furent developpes dans le but de simuler le mouvement d'un missile a longue portee.

Les calculs de coordonnees de la trajectoire demandant une grande precision furent confies au calculateur numerique alors que le mouvement autour du centre de gravite, necessitant une grande vitesse de calculs, fut simule par le calculateur analogique.

Parmi les premières combinaisons qui employaient les plus grands et plus rapides calculateurs numériques, on peut citer :

UNIVAC 1103 A ou IBM 704 qui étaient associés à 300 ou 400 amplificateurs d'ANALOG FACE (5).

C'est au début des années 60 que furent commercialisés les premiers calculateurs hybrides universels.

On peut citer : HYDAC 2000 développé par EAI. Ce fut le premier ensemble hybride de série (13).

Il était composé d'un calculateur analogique 231RVEAI d'une console logique DOS 350 EAI et d'un calculateur numérique PDP 24 (10) (13).

Cependant la plupart des calculateurs hybrides étaient largement influencés par les exigences des applications spécifiques à l'industrie aérospatiale (9).

### II.3 ) Calcul hybride et calculateurs hybrides

---

II.3.1 ) Calcul hybride : La notion de calcul hybride ne se limite pas uniquement à celle associant des techniques de calcul numérique et des techniques de calcul analogique pour la résolution d'un problème (5).

En effet, on trouve dans la littérature des exemples de ce type de calcul.

Nous allons en citer deux exemples :

#### II.3.1.a ) Machine analogique - Programme numérique

---

Dans ce cas nous avons à faire à une simulation purement analogique : une fois que le modèle est câblé sur le

panneau, c'est le calculateur analogique qui va effectuer le calcul.

Cependant, la preparation du probleme a etudier se fait a l'aide d'un ordinateur qui joue donc un role d'interface entre l'utilisateur et la simulation analogique.

En effet, il existe des logiciels qui, a partir des equations definissant le modele physique reel, permettent d'obtenir :

a ) les parametres du modele analogique tels que:

- Les valeurs des coefficients des potentiometres.
- Le choix des gains des amplificateurs.
- Le choix de constantes de temps des integrateurs.

b ) Le choix des composants et la creation d'operateurs intermediaires si besoin est.

Dans certains cas, il peut exister une liaison entre l'ordinateur et la machine analogique.

Le programme numerique permettra les operations suivantes:

- Verifier le cablage.
- Regler l'execution.

L'un des premiers langages de simulation analogique est le langage APACHE developpe a EURATOM a ISPRA. Ce langage fut ecrit en assembleur sur IBM 7090 (5) (8) (12).

#### II.3.1.b ) Simulation analogique sur ordinateur

Ceci est possible a l'aide de langages numeriques conformes aux normes CSSL.

Ces langages permettent a l'utilisateur d'imaginer qu'il utilise une machine analogique pour etudier son probleme. Leur facilite de programmation croissante permet leur utilisation par des non specialistes, ce qui leur permet d'aborder des problemes de plus en plus complexes. Ces langages donnent a l'utilisateur le moyen de :

- Choisir un certain nombre d'operateurs.
- De les "Interconnecter".
- D'afficher des valeurs de coefficient
- De lancer la simulation selon les modes classiques :
  - Conditions initiales.
  - calcul
  - memoire

On peut citer quelques uns de ces langages :

- CSMP developpe par IBM.
- DAREP, DARE/ELEVEN qui se presentent sous forme de fichiers modulaires (3).

Cependant, la notion de calcul hybride a eu pour origine la comparaison et la necessite de reunir le calcul analogique et le calcul numerique du fait de leur complementarite.

Ceci fut realise moyennant un certain nombre d'etapes.

Si l'on va du fait que l'on veut ameliorer le calcul analogique on distingue quatre etapes (21).

#### II.3.2.a ) L'analyseur differentiel digital (1950-1960)

---

Du fait du manque de precision des operations elementaires lineaires des calculateurs analogiques, il s'est avere necessaire d'utiliser des techniques de base

purement numeriques pour realiser des operateurs ayant les memes fonctions que les operateurs du calculateur analogique.

Ces operateurs etaient mis en oeuvre de la meme facon que les operateurs d'un calculateur analogique (en realisant un schema reliant ces differents operateurs ).

Celui-ci fut tres vite abandonne du fait du progres des autres techniques (numeriques et analogiques).

II.3.2.b ) Calculateur analogique associe a des elements logiques

L'interet de ces calculateurs est de disposer d'un mode repetitif permettant le passage du mode conditions initiales au mode calcul a une cadence elevee.

On peut disposer de moyens supplementaires permettant de corriger automatiquement un certain nombre de parametres en fonction des resultats obtenus dans les solutions precedentes.

On obtient ainsi les analyseurs differentiels iteratifs (IDA).

II.3.2.c ) Adjonction d'elements memoire importants aux IDA

Les possibilites de memorisation des IDA sont tres reduites et les calculs proprement dits ne peuvent etre qu'analogiques. Une autre etape est donc celle dans laquelle des possibilites de memorisation importantes sont ajoutees aux IDA.

Ceci etait necessaire dans le cas ou il fallait conserver certaines donnees sous forme de tables (ex : generateurs

de fonctions a une ou plusieurs variables).

Ceci a permis d'elargir le champs d'application des IDA

II.3.2.d ) Calculateur analogique et logique parallele  
associes a un calculateur numerique

C'est la derniere etape qui a permit d'aboutir a un  
systeme hybride vrai ou Analog/Hybrid computer.

Dans ce systeme, le calculateur analogique et le  
calculateur numerique sont couples a l'aide d'elements  
d'interface hybrides.

Ces differentes etapes sont resumees sur la fig II.1.

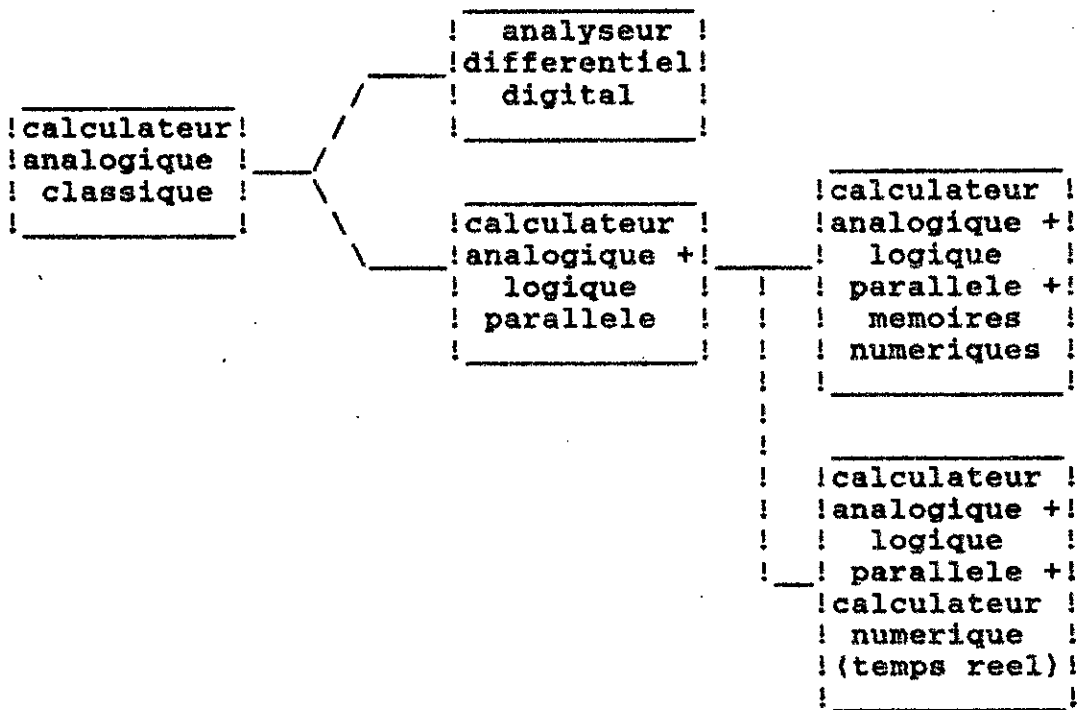


figure II.1 ETAPES DU CALCUL HYBRIDE

## II.4 ) Formes speciales de calculateurs hybrides (8), (22)

---

Le terme "calculateur hybride" a ete employe pour indiquer la combinaison de calculs continus et discrets (8), (22).

De ce fait, on va citer deux formes speciales de calculateurs "hybrides" qui ne resultent pas de l'association d'un calculateur analogique et d'un calculateur numerique.

### II.4.a ) Calculateur hybride parallele

---

On peut citer le calculateur HYDAC 2000 developpe par EAI. Il est constitue de l'association d'un calculateur analogique avec un systeme universel de blocs logiques programmables, un multiplexeur, un CA/D et des CD/A, une unite memoire pour le stockage des fonctions analogiques echantillonnees, ainsi que des nombreux additionneurs et soustracteurs.

Les applications de ces calculateurs englobent une gamme intermediaire de problemes hybrides :

- Simulation d'un retard.
- Generateur de fonction a une ou plusieurs variables.
- systemes asservis de controle.
- Automatisation du calculateur analogique dans le cas d'etudes d'optimisation ou de recherche de parametres.
- Simulation d'un systeme asservi de donnees numeriques echantillonnees.



#### II.4.b ) Calculateur hybride serie

Un simulateur de vol a ete developpe a MIT pour l'entrainement du personnel navigant.

C'est le Pulsed-Analog-Computer.

Ce systeme est constitue d'un calculateur numerique qui controle un petit nombre de composants analogiques fonctionnels tels que : multiplieurs, integrateurs ....

Ces composants analogiques sont controles par le programme numerique. Ils realisent des operations equivalentes a des sous-programmes pour le calculateur numerique

#### Definition du calcul hybride

Le calcul hybride peut etre defini comme une utilisation de calculateurs analogiques et numeriques relies entre eux par un sous-ensemble de liaison (ou interface) et mis en oeuvre selon des techniques de programmation propres a ce mode de calcul.

Selon les fonctions devolues au calculateur numerique on distingue :

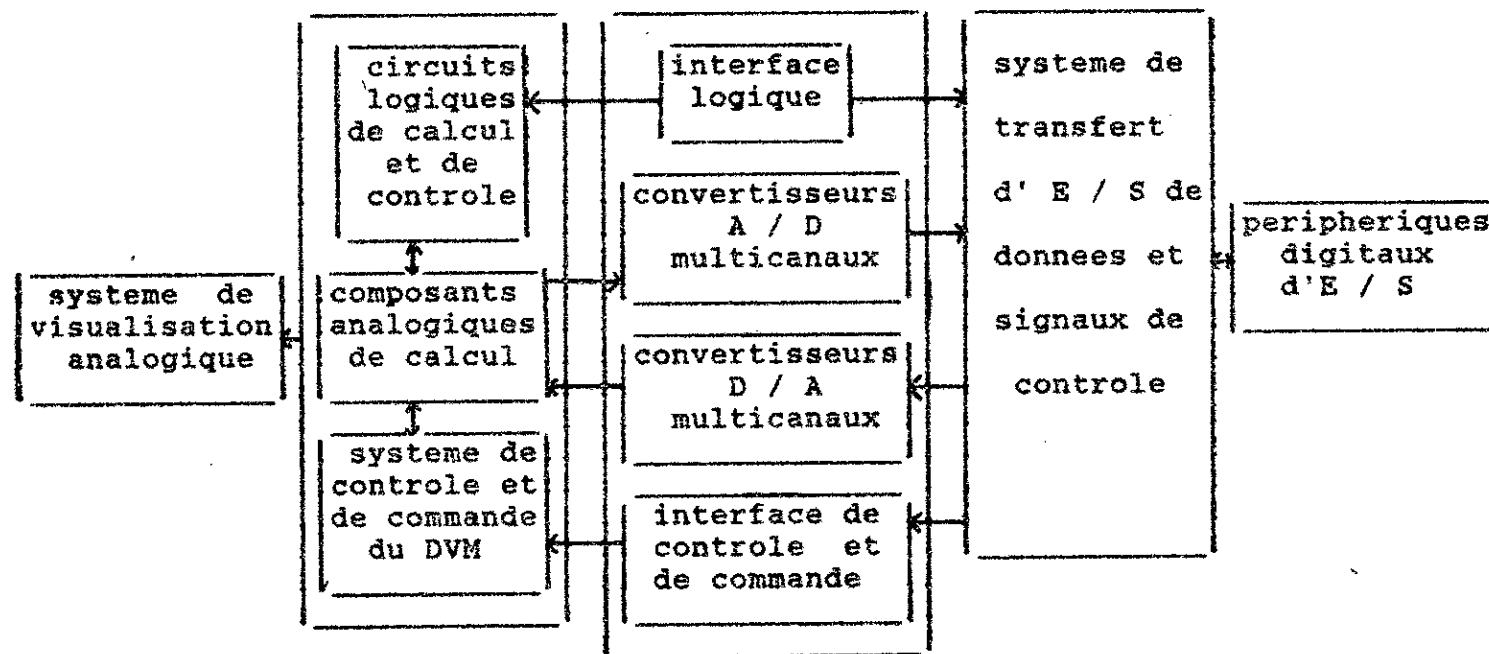
- Le calcul hybride serie (5) (13)

Le calculateur numerique est utilise en serie avec le calculateur analogique.

- Le calcul hybride parallele

Dans ce cas le calculateur numerique est utilise en parallele avec le calculateur analogique.

fig (II.2) Organisation generale d'un calculateur hybride (2)



console analogique  
logique

systeme d'interface  
hybride

calculateur  
numerique

On voit apparaitre dans cette organisation trois blocs distincts :

- a ) Le calculateur numerique.
- b ) Le calculateur analogique ou console analogique/logique.
- c ) Le systeme d'interface hybride assurant le lien entre les deux blocs precedents.

Le calculateur analogique peut etre considere comme un peripherique du calculateur numerique.

Les donnees traitees par les deux calculateurs sont de nature differente (continues et discrettes). Des elements d'interface sont donc necessaires pour permettre le traitement d'une meme donnee par les deux calculateurs. Ces elements d'interface assurent la liaison entre les deux calculateurs.

#### II.5.1 Systeme d'interface hybride

Celui-ci est compose de trois parties :

##### II.5.1.a ) Une interface de commande et de controle

Elle permet le transfert d'informations logiques dans les deux sens.

Ces informations permettent la commande des modes analogiques, logiques et des vitesse d'integration.

Elle permet aussi :

- Le réglage de tous les parametres d'une simulation.
- De lire toutes les valeurs des operateurs.

II.5.1.b ) Une interface logique : Celle-ci assure la liaison entre la partie logique du calculateur analogique et le calculateur numerique.

Une information logique peut interrompre un programme numerique en cours.

II.5.1.c ) Une interface de donnees : elle est constituee d'un certain nombre de convertisseurs A/D et D/A permettant le transfert d'une information du domaine analogique ou continu au domaine numerique ou discret.

Le transfert de l'information est represente en fig II.3.

II.5.2 ) Description de la partie analogique

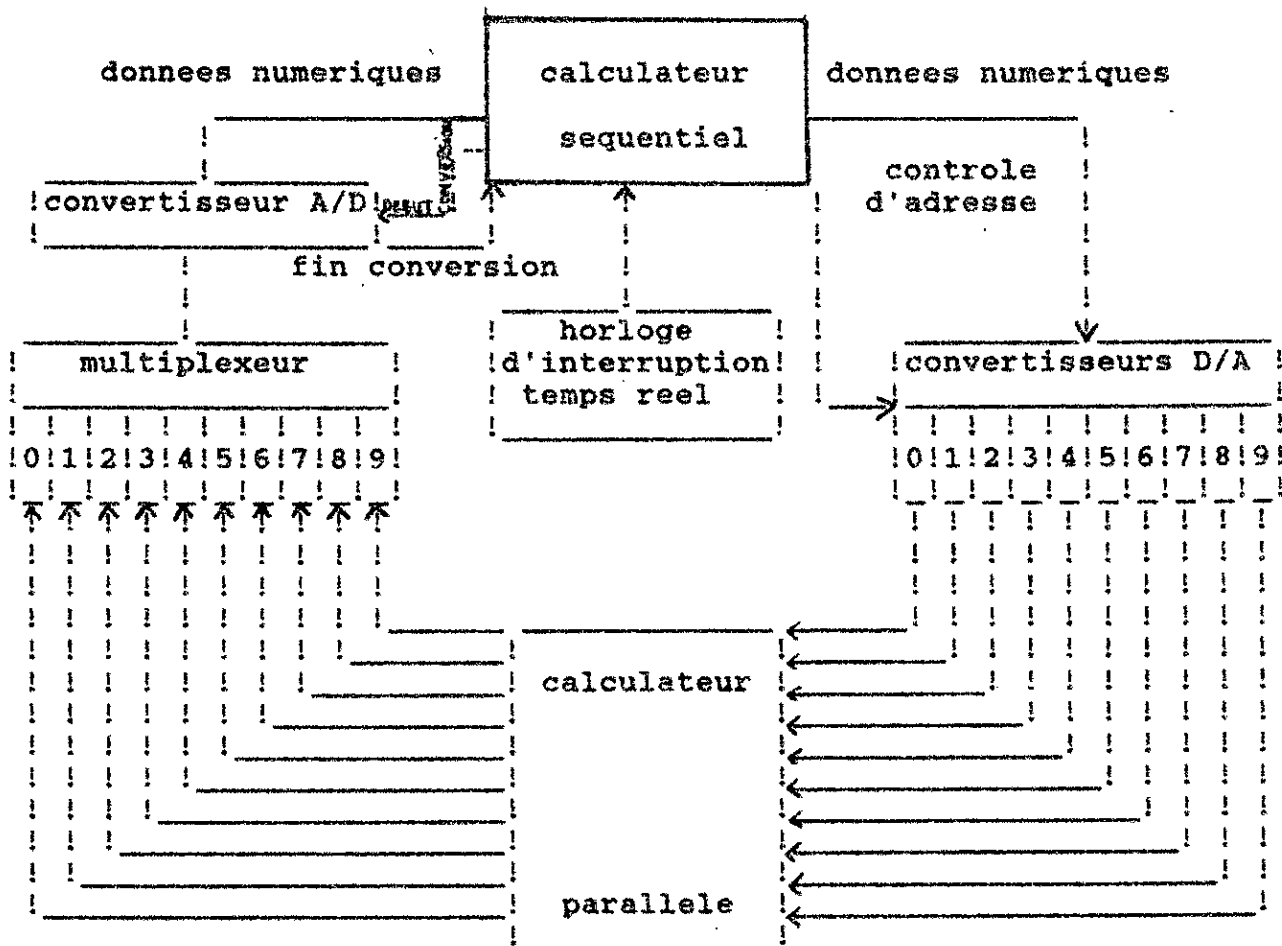
Les operateurs analogiques et logiques rencontres dans un systeme hybride sont identiques a ceux d'un calculateur analogique mis a part les operateurs a commande programmable, on distingue :

- Les potentiometres programmables.
- Les amplificateurs programmables .....

Ces operateurs sont des operateurs hybrides car ils effectuent le produit d'une tension analogique par un mot binaire, le resultat etant donne sous forme d'une tension analogique.

Leur programmation s'effectue a partir du calculateur numerique.

fig 3 Transfert des donnees du calculateur numerique  
vers (du) calculateur analogique



C'est le programme sequentiel qui controle le cycle de conversion.

Le cycle est initialise par une horloge d'interruption.

On dispose d'un grand nombre de convertisseurs D / A.

Les sorties analogiques sont multiplexees avant d'etre converties par le CA / D.

L'introduction du coefficient d'un potentiometre ou du gain d'un amplificateur s'effectue a partir du clavier d'une console digitale.

### II.5.3 Partage des taches entre les elements constitutifs d'un calculateur hybride

Nous venons de voir la composition d'un calculateur hybride, nous pouvons donner une idee du role que joue chacun des elements dans le calcul hybride.

#### II.5.3 a Calculateur analogique

La tache generalement confiee au calculateur analogique d'un ensemble hybride est la resolution des equations differentielles du fait de sa grande vitesse de resolution de ce type d'equations.

#### II.5.3 b Logique parallele

Les fonctions de la logique parallele sont :

- Automatisation du calculateur analogique
- Commandes des moyens de sortie analogiques tels que enregistreurs, oscilloscope.
- Simulation d'operations logiques (pour des operations simples, la logique parallele est beaucoup plus rapide que le calculateur numerique).
- Synchronisation de l'ensemble hybride.

Le calculateur analogique pourra résoudre a grande vitesse un système d'équations différentielles alors que la logique parallèle commandera le déroulement des cycles de calcul, la modification des conditions initiales et les organes d'enregistrement et de visualisation.

### II.5.3 c Calculateur numérique

-----

Dans un ensemble hybride, les tâches principales d'un calculateur numérique sont :

- Le réglage et la lecture des composants analogiques
- L'automatisation des opérations sur le calculateur analogique.
- Le calculateur numérique peut avoir en mémoire un programme très élaboré permettant la comparaison d'un résultat de calcul à des données physiques en vue de déterminer de nouvelles valeurs de coefficients ou de choisir un modèle mathématique différent.
- Certaines opérations du calcul analogique (par exemples les intégrations lentes)
- La génération de fonctions à une ou plusieurs variables.
- Génération de retards
- Résolution d'équations algébriques
- Simulation d'un calculateur numérique en ligne. . .

## II.5.4 Programmes et langages

---

le logiciel hybride facilite la communication entre l'utilisateur et le calculateur hybride.

Il a beaucoup de points communs avec le logiciel numerique classique. Il ne se differencie que par les programmes d'un certain type :

- Programmes relatifs au sous ensemble de liaison
- Programmes utilitaires
- Programmes de mise au point et de verification.

Les langages de programmation, les programmes bibliotheque et le systeme moniteur ne sont pas specifiques de l'ensemble hybride (13).

Dans de nombreux problemes peuvent s'exercer des contraintes de temps provenant du besoin de communiquer avec un systeme externe au calculateur numerique.

De ce fait, le programme hybride doit donc etre capable d'accepter, de traiter et de repondre a des informations provenant de systemes externes.

Pour cela, il doit etre constitue d'un moniteur temps reel.

## II.6 Logiciel d'un calculateur hybride

---

### II.6.1 Logiciel minimal

---

Une premiere etape dans le developpement du logiciel est celle permettant l'automatisation du calcul analogique.



Ceci a mene a un logiciel minimal forme d'un ensemble de sous programmes appelables en FORTRAN.

On peut citer FORTRAN LINKAGE (developpe par EAI) ainsi que HCR (hybrid communication routine) developpe par Applied Dynamics.

Ces logiciels permettaient les operations suivantes :

II.6.1 a Pour l'interface de controle  
-----

- Selection d'un operateur.
- Lecture de sa valeur de sortie
- Selection du mode operatoire analogique
- Choix de la vitesse de calcul
- Reglage d'un attenuateur

II.6.1 b Pour l'interface logique  
-----

- Selection d'un mode operatoire
- Choix d'une horloge de synchronisation
- Affichage ou lecture d'un mot logique
- Affichage d'une ligne de controle
- Lecture d'une ligne d'etat

II.6.1 c Pour l'interface de donnees  
-----

- Lecture d'un ou plusieurs convertisseurs A/D
- Selection d'un convertisseur D/A

La faiblesse de ce type de logiciel est son caractere ponctuel qui nuit a la vitesse d'execution dans le cas ou l'on effectue une recherche de parametres.

## II.6.2 Logiciels hybrides conformes aux normes CSSL

---

Par la suite, les concepteurs de langages CSSL ont pense a developper des langages de simulation s'adaptant aux calculateurs hybrides, sans toutefois aboutir a des langages universels (2).

Les calculateurs ainsi obtenus permettent le calcul parallele tout en beneficiant des facilites de programmation des calculateurs numeriques.

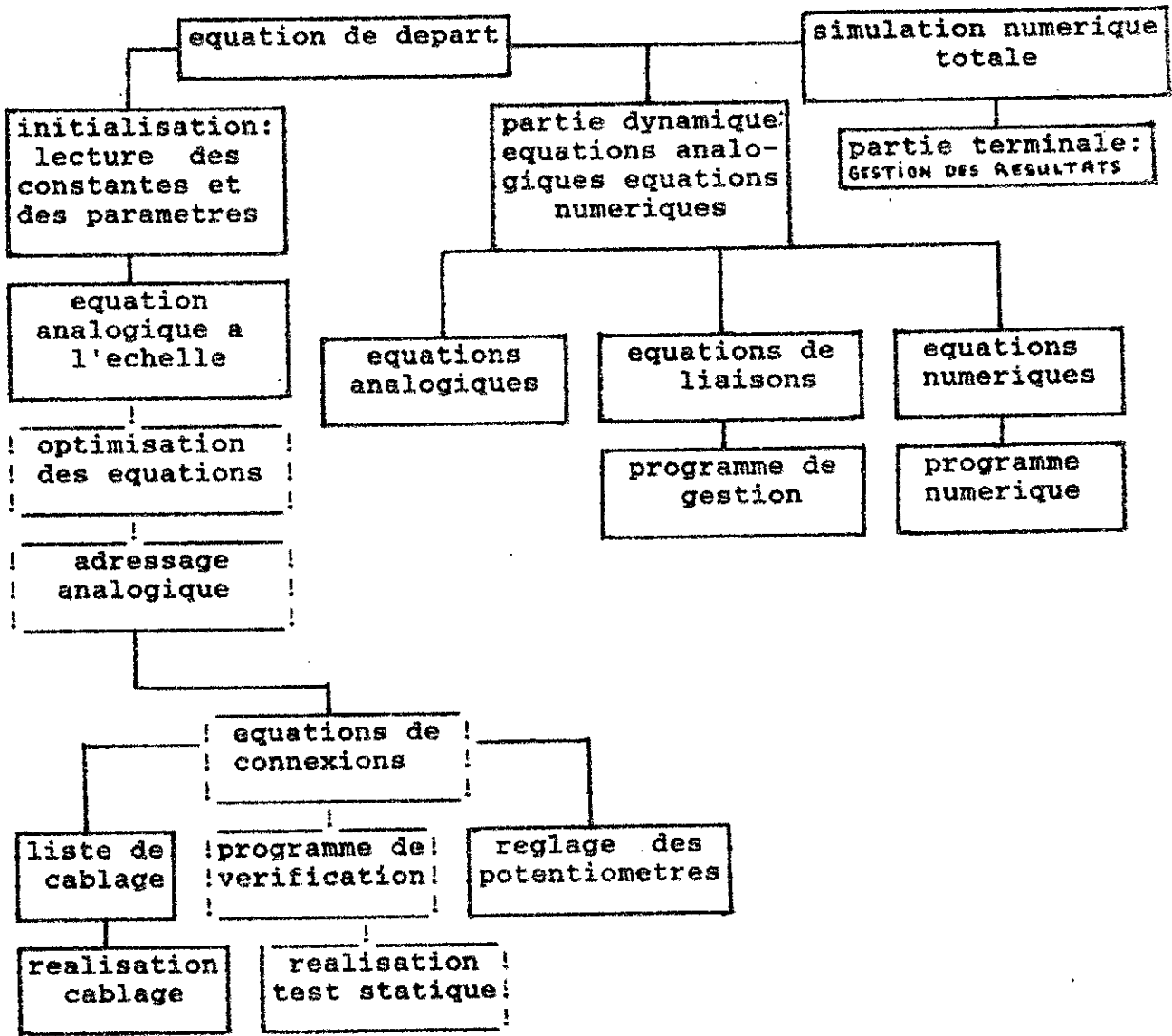
Le calcul hybride a atteint son dernier stade en trois etapes :

Premiere etape (1958 - 1962) : couplage d'un calculateur analogique et d'un calculateur numerique.

Deuxieme etape (1962 - 1966) : Realisation d'ensembles hybrides avec un logiciel limite.

Troisieme etape 1966 ---> : ensemble de calcul hybride avec un logiciel evolue.

Un organigramme resumant les etapes de programmation sur calculateur hybride est donne fig II.4.



P R E P A R A T I O N

E X E C U T I O N

fig (II.4) organigramme de resolution d'un probleme hybride

## II.7 Domaines d'applications du calculateur hybride

---

L'operation de base du calculateur hybride est, comme pour le calculateur analogique, l'integration continue par rapport au temps machine.

De ce fait, le calculateur hybride est tres bien adapte a :

- La resolution des equations differentielles ordinaires notamment dans le cas ou le systeme presente des non linearites dont l'approche analytique de la solution est complexe voire impossible.

- La resolution des equations aux derivees partielles (9), (25).

- Il permet aussi la resolution d'equations algebriques bien qu'il ne soit pas tres adapte a ce type de problemes (5).

Les applications les plus courantes sont relatives a :

- La dynamique des systemes (dans l'industrie la plupart des grands systemes sont controles par des ordinateurs)

- A la regulation de ces systemes

- A l'optimisation de leur fonctionnement (le processus est simule par le calculateur analogique. Les calculs d'optimisation sont effectues par le calculateur numerique).

- A l'identification des processus (recherche des parametres pour obtenir la forme generale d'un modele mathematique pouvant rendre compte de la dynamique du processus).

Toutes ces applications sont rencontrées en grande partie dans le domaine industriel et en particulier dans le domaine spatial, automobile, énergie nucléaire et chimie. D'autres domaines sont ceux de la médecine et de la biologie animale ou végétale.

- C O N C L U S I O N -

=====

Pendant de nombreuses années, les calculateurs hybrides ont joué un rôle très important dans l'industrie aérospatiale.

Vers les années 60, le développement de ces calculateurs fut consolidé par un grand potentiel d'utilisateurs dans ce secteur.

Les applications dans d'autres industries et dans des disciplines scientifiques furent aussi remarquables.

Avec l'évolution de la technologie et la configuration à multiprocesseurs, les calculateurs numériques ont eu tendance à remplacer les calculateurs hybrides dans les tâches de simulation de petits et grands systèmes physiques.

Cependant, les langages de simulation numérique adaptés à la configuration à multiprocesseurs sont très difficiles à mettre en œuvre dans certains cas.

En effet, les plus grandes difficultés sont liées au partitionnement et à l'échange entre processeurs.

De plus, il semble que la façon la plus simple d'écrire le programme pour un système est lorsque le jeu d'équations différentielles le définissant peut se décomposer en sous-système pouvant chacun être résolu par son propre processeur.

Il arrive souvent que la mise au point et l'organisation d'une simulation hybride presentent nettement moins de difficulte que dans le cas d'une simulation purement numerique et ceci dans le cas de systemes fortement couples.

Compte tenu des facilites de mise en oeuvre, du cout et de la possibilite de s'adapter a des applications particulieres, la simulation hybride se revelera plus adequate.

De plus, les calculateurs hybrides existants seront toujours utilises des qu'ils accomplissent leur tache de facon satisfaisante [26] [32] [27].

On trouve encore dans un grand nombre d'universites des installations modestes de calculateurs hybrides (5).

Ces installations permettent la comprehension de la dynamique des systemes par les etudiants qui ne peuvent disposer d'installations experimentales reelles.

- CHAPITRE III -  
-----

III.1 Introduction  
-----

Nous disposons de la partie analogique d'un calculateur hybride ainsi que de l'interface logique.

Nous avons eu a realiser toute la partie microsysteme destinee a gerer le calculateur hybride ainsi obtenu.

Ce microsysteme est constitue d'une carte micro-calculateur batie autour d'un microprocesseur 8 bits (MC 6800 de MOTOROLA).

Nous avons egalement realise une partie de l'interface hybride : l'interface analogique qui est destinee a la conversion des tensions analogiques et a leur visualisation sur l'ecran.

Nous avons d'autre part developpe un logiciel permettant l'exploitation du calculateur hybride. La premiere partie de ce chapitre est destinee a la description materielle du calculateur hybride.

La seconde partie sera consacree au logiciel que nous avons developpe.



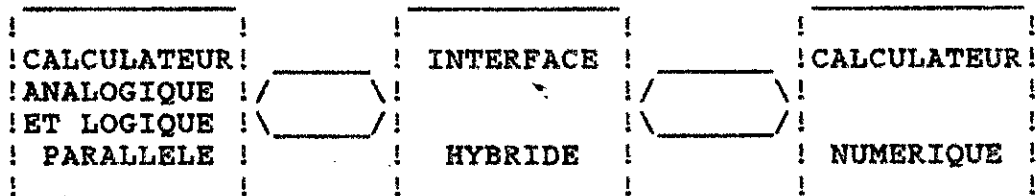
### III.2 Description du calculateur hybride

---

Dans le but de comprendre le fonctionnement du calculateur hybride nous allons decrire tous les elements qui le constituent :

D'une facon globale, la configuration du calculateur hybride peut se decomposer en une partie numerique et une partie analogique (fig III.1).

Les liaisons entre les deux parties sont assurees par une interface hybride.



(fig III.1)

#### Synoptique du calculateur hybride

La partie analogique est constituee :

---

- D'un champ analogique : dans lequel se trouvent les elements operationnels de base effectuant les operations fondamentales du calcul analogique.
- D'un champ logique : Compose d'operateurs logiques permettant entre autres de commander les modes des operateurs analogiques.

#### III.2.1 Description du champ analogique

---

La structure de ce champ est modulaire, chacune des cartes de ce champ comporte un seul type d'operateurs.

Les entrees et les sorties de ces operateurs sont reliees au niveau du panneau de cablage permettant ainsi de les interconnecter en vue d'obtenir le modele de simulation.

On distingue :

-----  
III.2.1 a Les operateurs purement analogiques :  
-----

Ceux-ci effectuent un traitement continu sur des tensions continues.

Les operateurs dont nous disposons sont les suivants :

- Sommateur
- Racine carree
- Track / Store
- Multiplieur / Diviseur
- Sinus / Cosinus
- Logarithme / Exponentiel
- Integrateur

III.2.1 b Les operateurs logiques (ou logique parallele)  
-----

Ces operateurs permettent le controle des operateurs analogiques de sorte a decharger le calculateur numerique au cours du calcul.

Il se compose d'elements logiques usuels :

- Portes logiques
- Elements memoire
- Compteurs ...

### III.2.1 c Les operateurs analogiques a commande numerique

---

La premiere etape permettant le passage d'un calculateur analogique a un calculateur hybride est celle liee a la presence d'operateurs appeles :  
operateurs hybrides

Ces operateurs effectuent la multiplication d'une tension analogique par une tension binaire, le resultat etant delivre sous forme de tension analogique.

La particularite de ces operateurs est leur capacite a etre commandes a partir du calculateur numerique.

On distingue :

- Les potentiometres numeriques (PTN)
- Les amplificateurs programmable (APR)
- Les generateurs de fonctions (GDF)

Ces operateurs peuvent etre controles et adresses a partir du microsysteme par programme.

Les coefficients des PTN et les gains des APR sont introduits a partir d'un clavier.

Les PTN et les APR sont realises autour de convertisseurs MDAC 7521 d'Analog Devices.

Ce sont des convertisseurs D/A 11 bits + signe

Une description de ces operateurs est donnee en

Annexe I.

### III.2.2 Description des interfaces

On distingue deux types d'interfaces :

- L'interface logique
- L'interface analogique

#### III.2.2 a Interface logique

On dispose de deux cartes d'interface logique.

a) La carte permettant la selection et le chargement des potentiometres numeriques, des amplificateurs programmables et eventuellement des generateurs de fonctions.

Le synoptique de cette carte est donne (fig III.2).

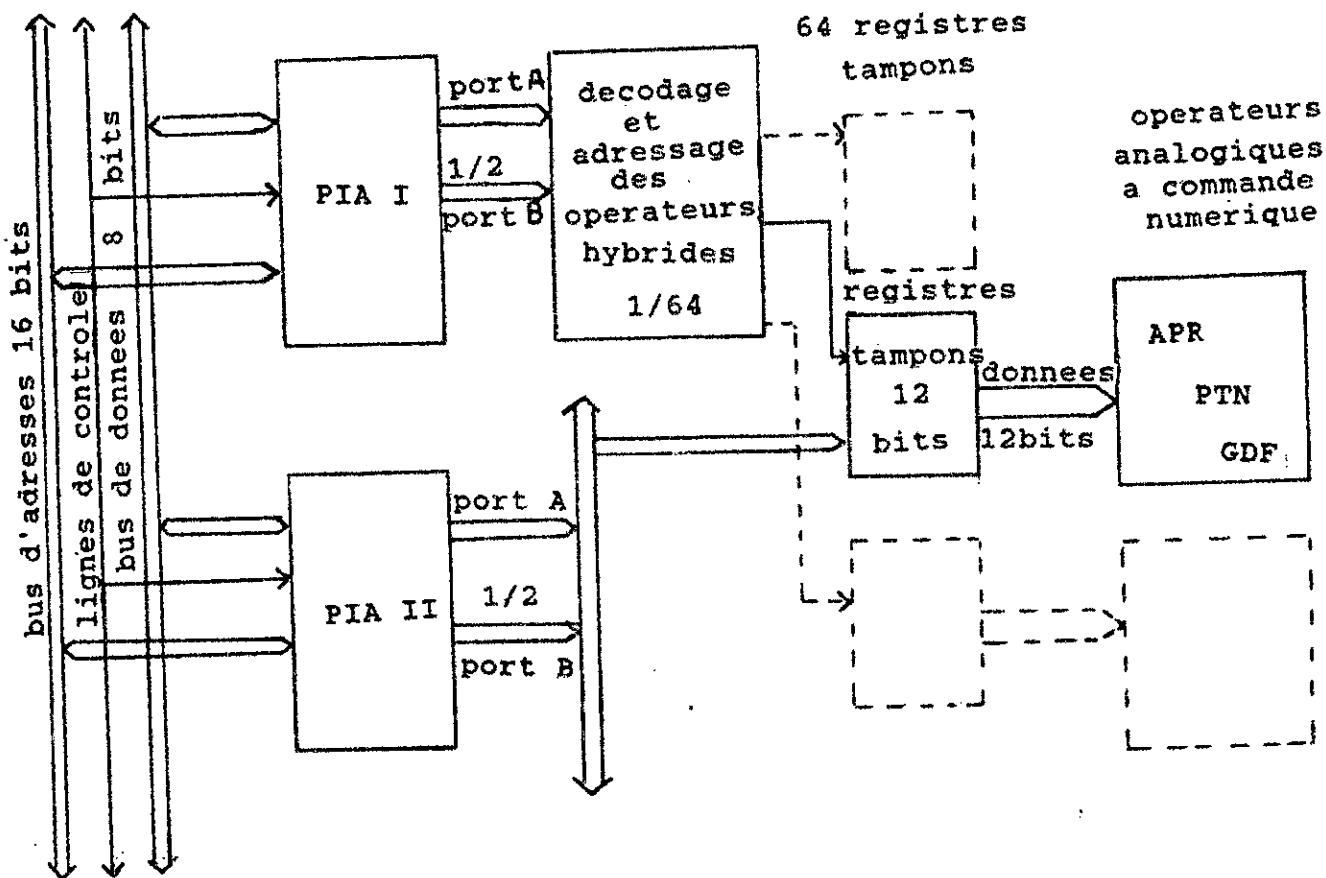


fig III.2

SYNOPTIQUE DE L'INTERFACE LOGIQUE

Pour la communication avec le microsysteme, nous avons des interfaces d'E/S parallele du type PIA.

Le PIA I permet la selection d'un ou plusieurs operateurs.

Le PIA II permet le chargement d'une donnee binaire 12 bits au niveau de l'operateur selectionne.

Cette donnee est introduite par l'utilisateur a partir d'un clavier sous forme d'un nombre decimal ASCII.

Dans le cas des potentiometres numeriques, cette donnee represente le coefficient du PTN. Cette valeur est comprise entre -1 et +1.

Nous avons developpe un programme permettant la conversion de cette donnee decimale ASCII en une configuration binaire 12 bits qui representera l'entree binaire N du PTN.

Un autre programme permet la conversion d'une donnee representant le gain d'un amplificateur en une configuration binaire 12 bits. Dans notre cas nous avons un gain variant entre +1 et 100.

Pour pouvoir charger ces donnees binaires au niveau des entrees des PTN et des APR, la carte d'interface logique dispose de registres tampons (12 bits) en sortie.

b) La selection des integrateurs et le chargement des constantes de temps sont effectues par une carte d'interface logique identique a la precedente.

Cependant, cette carte n'utilise qu'un seul PIA.

Le port A et la moitie du port B sont destines a la selection d'un ou plusieurs integrateurs parmi 24.

Les quatre dernières lignes du port B sont utilisées pour le chargement de la constante de temps.

Ces lignes commandent un interrupteur électronique à 4 voies (A,B,C,D).

On a la possibilité de choisir parmi quatre constantes de temps 1s ; 0,1s ; 1ms ; 0,1ms (Voir ANNEXE II).

#### III.2.2.d ) Interface analogique

L'interface analogique est constituée d'une carte DVM que nous avons réalisée.

Celle ci permet :

- La sélection d'un opérateur parmi 16 (extension possible).
- De convertir une tension analogique sous forme d'un mot binaire de 12 bits.

Nous avons développé et testé un programme permettant de convertir cette grandeur binaire (12 bits) en decimal ASCII pour affichage sur l'écran de visualisation.

#### Description et fonctionnement de cette interface

Le synoptique de la carte DVM est donné (fig III.3).

(\*) Il n'a pas été choisi de mémoriser la sortie de chaque opérateur analogique ce qui aurait conduit à placer un CA/N pour chaque sortie.

Les sorties des opérateurs analogiques sont ramenées au niveau de la carte DVM par l'intermédiaire de câbles blindés pour éviter les effets d'inductions parasites.

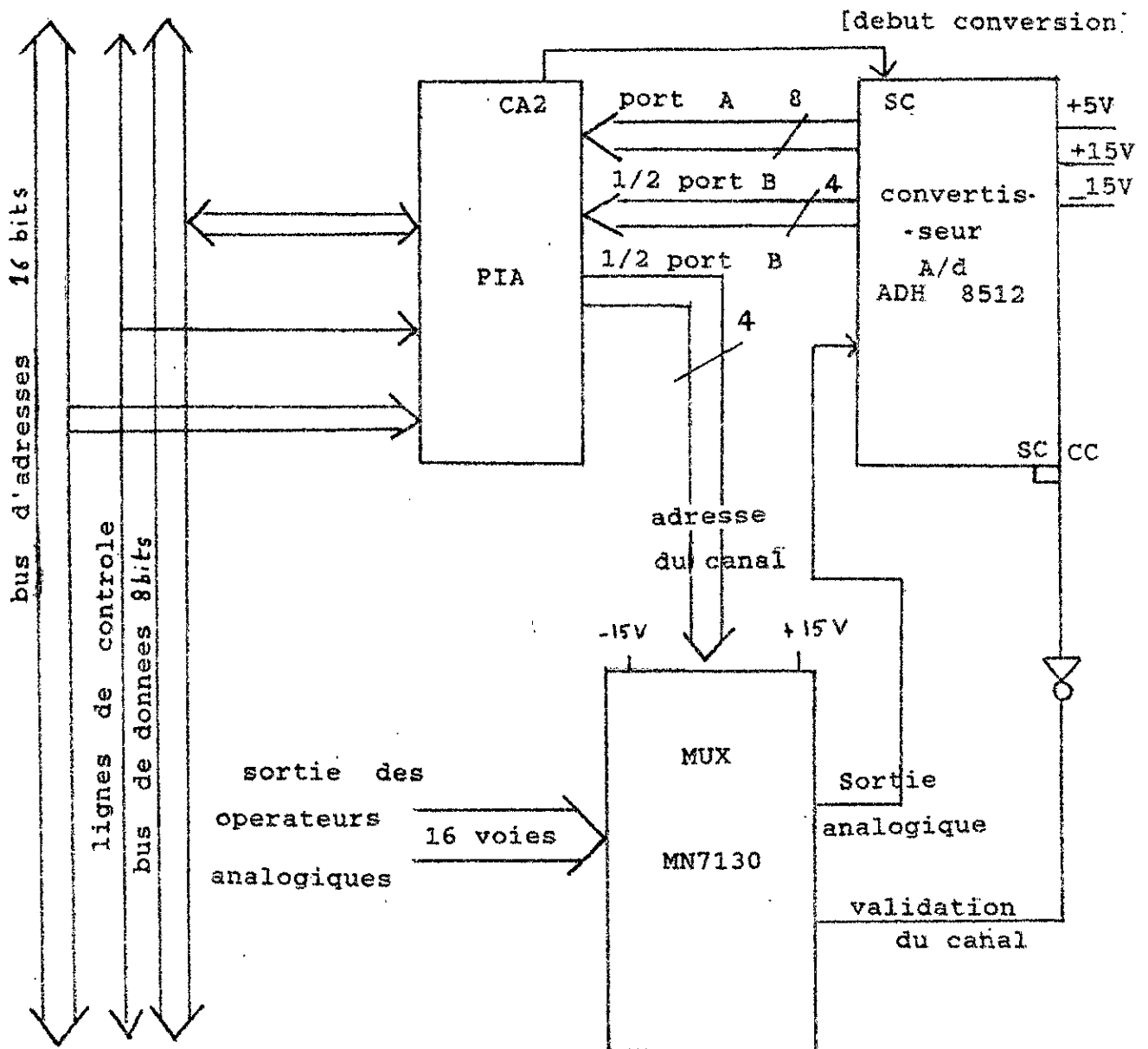


fig (III.3)

Synoptique de la carte DVM

Ces tensions sont ramenees a l'entree d'un multiplexeur permettant la selection de l'un des operateurs.

Dans notre cas on s'est limite au multiplexage de 16 voies mais une extension est possible en ajoutant un second niveau de multiplexage.

Les elements qui composent cette carte sont :

- 1 PIA permettant la communication avec le microsysteme.
- 1 convertisseur A/D 12 bits : ADH 8512 (disponible au laboratoire ).
- 1 multiplexeur 1/16 : MN7130.

Le PIA permet la selection d'un canal du multiplexeur a travers quatre lignes du port B.

La ligne CA2 du PIA est programmees en sortie en mode impulsionnel. Elle permet de lancer le cycle de conversion A/D.

Le temps de conversion de l'ADH 8512 est de  $1,8 \mu s$ .

La conversion se fait en choisissant l'une des configurations suivantes :

- binaire pur.
- binaire decale.
- complement a deux.

Dans notre cas, la configuration binaire decalee est la plus adequate pour l'écriture du programme de conversion du binaire en decimal ASCII.

La precision de ce convertisseur est de  $+0,012\%$  de la pleine echelle. Celle des operateurs analogiques est reportee en ANNEXE III.



Les tensions converties par le CA/D varient entre -10V et +10V. Celui-ci peut être programme suivant trois modes :

- 1) Ignorer une nouvelle commande de conversion jusqu'à ce que la conversion précédente soit achevée.
- 2) Entamer un nouveau cycle de conversion en interrompant le précédent.
- 3) Permettre des conversions successives (mode répétitif).

Il est spécialement adapté au multiplexage et à l'interfacage avec les microprocesseurs.

Nous avons dans notre cas choisi le premier mode de programmation.

Ce convertisseur est équipé de registres tampons en sortie qui permettent de mémoriser la configuration binaire obtenue tant qu'un nouveau cycle n'a pas démarré. Les cartes interfaces ayant été décrites nous pouvons résumer la liaison entre le calculateur analogique et le calculateur numérique par la fig III.4.

INTERFACE ANALOGIQUE

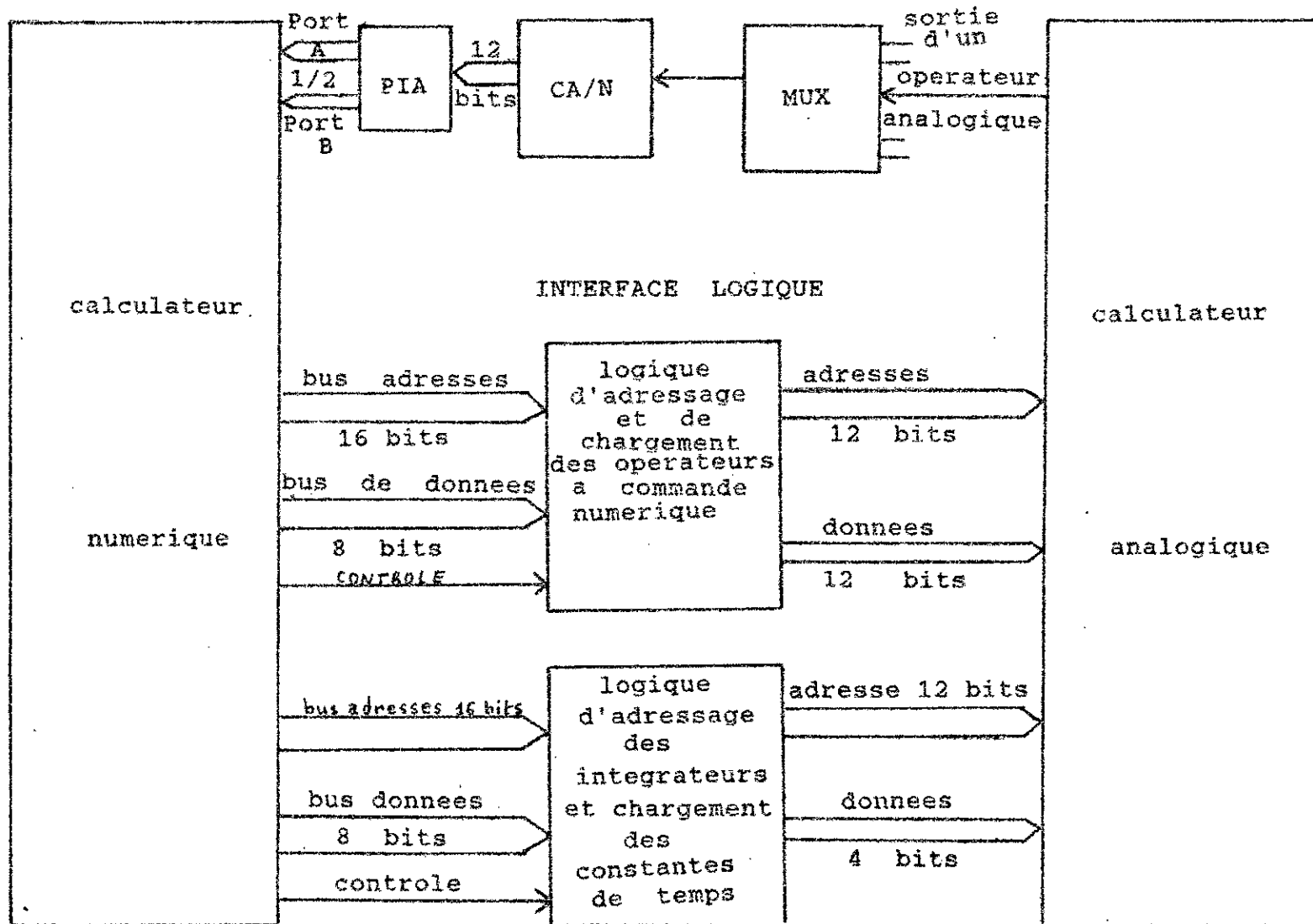


Fig III.4 - Synoptique de l'interface hybride.

### III.2.3 Description du champ numerique

---

#### III.2.3 a Fonctions realisees par le champ numerique

---

le champ analogique que nous avons decrit doit etre couple a un microsysteme dont le role est d'assurer la gestion et l'exploitation du calcul analogique et eventuellement du calcul numerique.

Pour cela, nous avons realise une carte microcalculateur permettant les fonctions suivantes :

- Controle de modes analogiques (ou logiques).
- Choix des constantes de temps des integrateurs.
- Chargement et affichage des differents parametres :
  - \* Coefficients des PTN
  - \* Gain des APR
  - \* Valeur des GDF
- Commutation des circuits d'interface et multiplexeur.
- Adressage du DVM.

#### III.2.3.b Composition de la carte microcalculateur

---

Cette carte est composee :

- D'un bus de donnees paralleles , bidirectionnel de huit bits .
- D'un bus d'adresse unidirectionnel 16 bits et de lignes de commande .

Les elements constituant cette carte sont :

- Un microprocesseur 8 bits (le MC 6800 de Motorola) ainsi que les buffers necessaires a son fonctionnement (8T26,8T95) (le MC 6800 a ete utilise pour des raisons de disponibilite).

- Une memoire centrale composee des elements suivants:

\* 3 EPROM 2516/TI dont la capacite de chacune est de 4 K/octets . Nous avons implante dans ces memoires , le moniteur que nous avons concu ainsi que tous les programmes que nous avons developpe pour l'exploitation du calculateur hybride.

\* 1 RAM 2716 d'une capacite de 2 Koctets . Elle permet le stockage des informations temporaires.

- Les circuits de decodage permettant l'adressage de tous les composants implantes sur la carte microcalculateur.

- Une interface serie ou ACIA ( MC 6850 ) permettant le dialogue entre l'utilisateur et l'unite centrale par l'intermediaire d'une console de visualisation .

- Une interface parallele ou PIA ( MC 6820 ) dont la ligne CA2 commande le demarrage du cycle de conversion de la carte DVM . Les lignes PA0 et PA1 du port A commandent le calcul en mode normal .

- Nous avons prevu un PICU dans le cas d'une extension future du systeme : Ce PICU peut permettre la gestion de huit niveaux d'interruptions hierarchisees .

- Une horloge programmable realisee a partir de deux timers MC 6840 montes en cascade selon la (fig III 5).

Les sorties O'3 et O'2 commandent les entrees IC et OP des integrateurs . Cela permet de mettre les integrateurs dans l'un des trois modes suivants :

- Mode conditions initiales ou IC
- Mode calcul ou OP
- Mode memoire ou H

La table des sequences des modes est la suivante .

	Entree IC	Entree OP
mode IC	1	0
mode OP	0	1
mode H	0	0

Les timers sont programmes de sorte a piloter deux modes de fonctionnement du calculateur hybride :

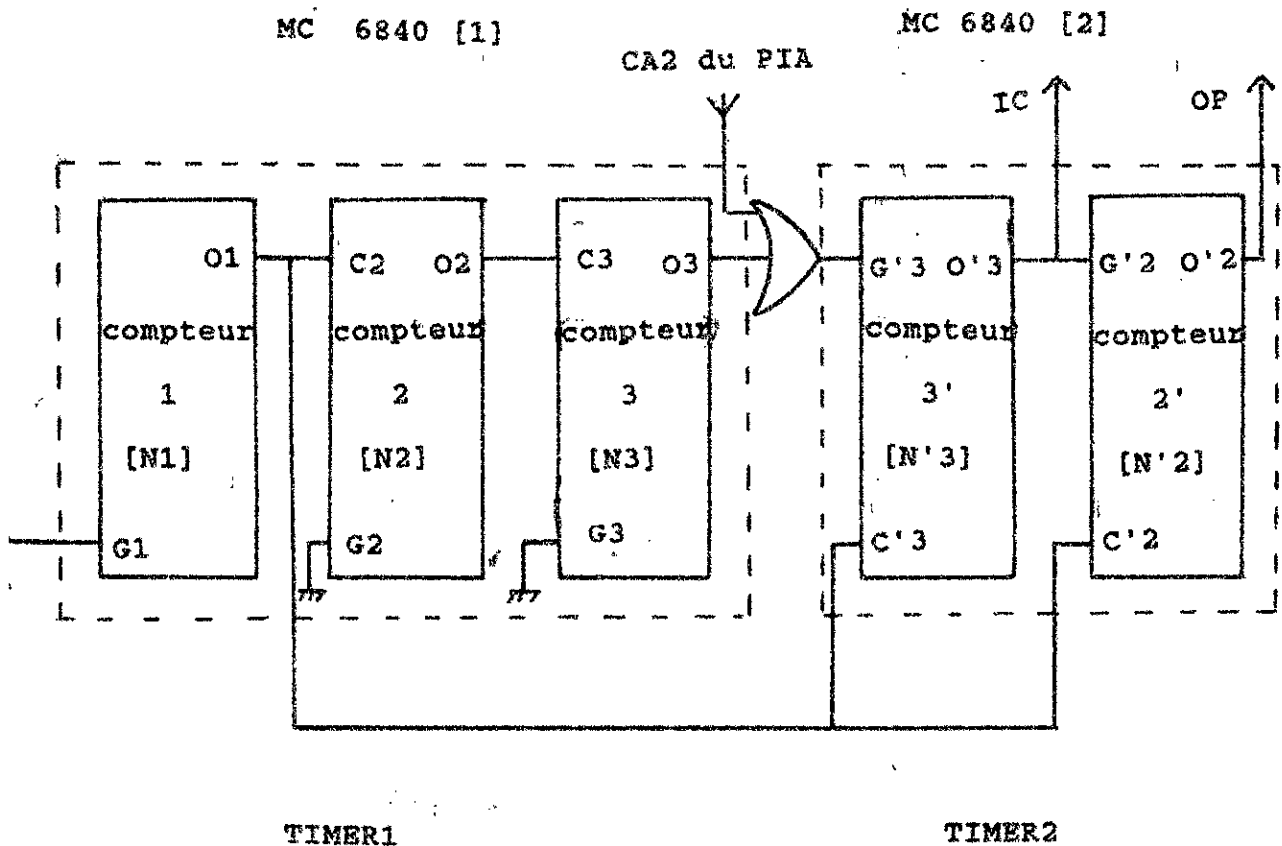
- Mode repetitif
- Mode iteratif

Un chronogramme de ces deux modes de fonctionnement ainsi que celui du mode normal est donne en figure III.6

fig III.5 HORLOGE PROGRAMMABLE

schema de cablage des

timers

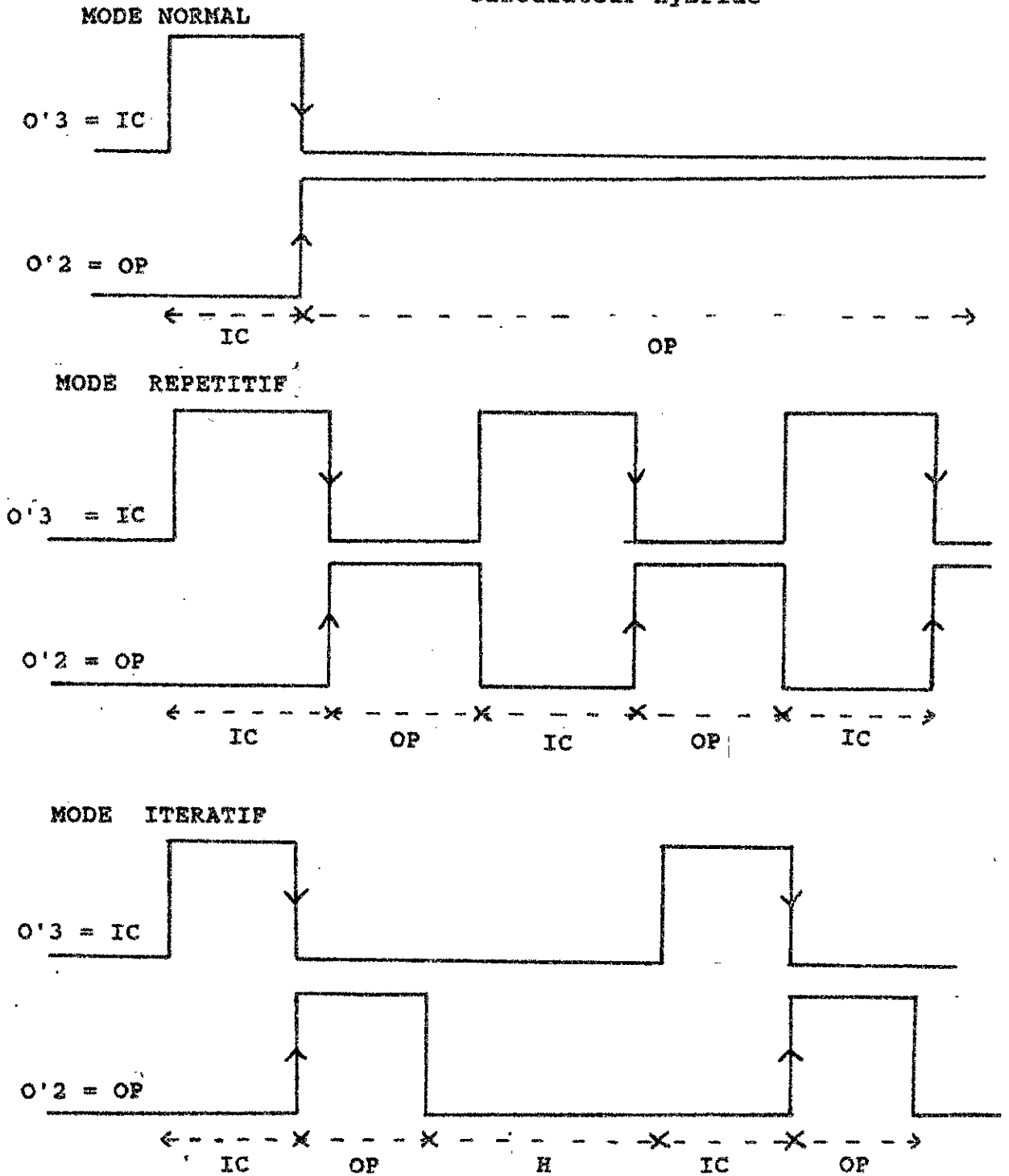


Chaque timer est constitue de trois compteurs

Ni est un nombre binaire programme sur 8 bits ou 16 bits.

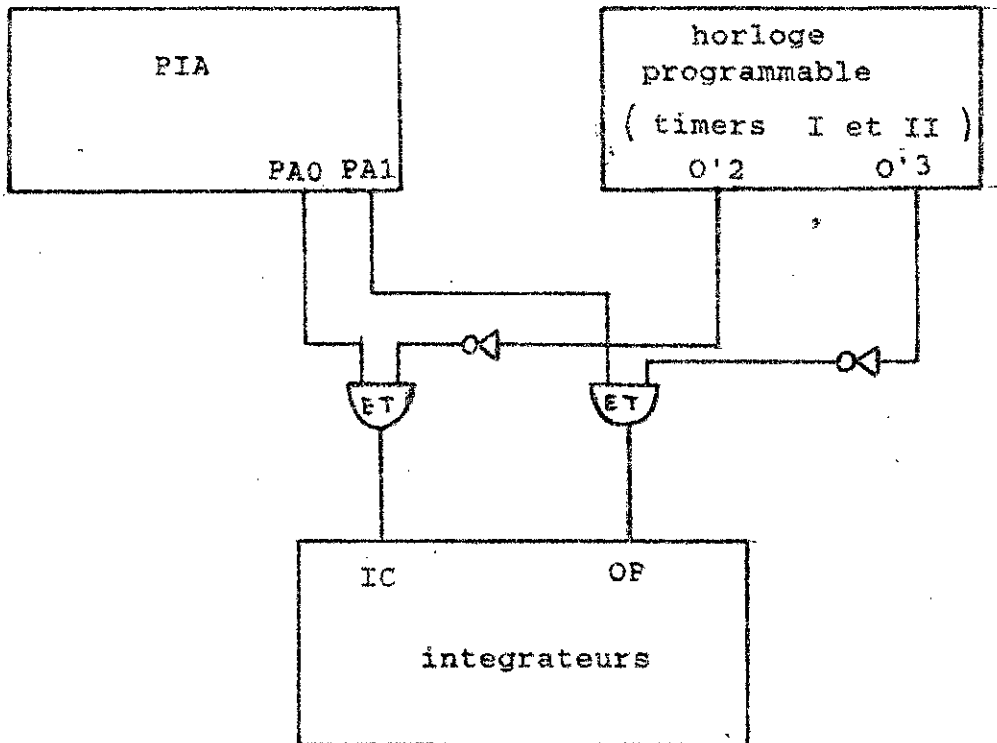
Chaque compteur est programme soit en mode continu (multivibrateur) ou en mode impulsienel (monostable)

fig (III.6) Chronogramme des modes de fonctionnement du  
calculateur hybride



Les durees IC et OP peuvent etre modifiees par programme.  
Dans notre cas ces durees peuvent varier de la  
microseconde a quelques dizaines de secondes .

Le schema de commande des entrees IC et OP des integrateurs est le suivant :



Dans le cas du mode normal, les timers sont inhibes. Les sorties O'2 et O'3 sont a zero. Les lignes PA0 et PA1 du PIA sont programmees de sorte a obtenir les etats logiques decrits dans le chronogramme 5 .

Dans le cas des modes repetitifs et iteratifs , les timers sont programmes de sorte a commander les entrees IC et OP suivant le chronogramme 5 .

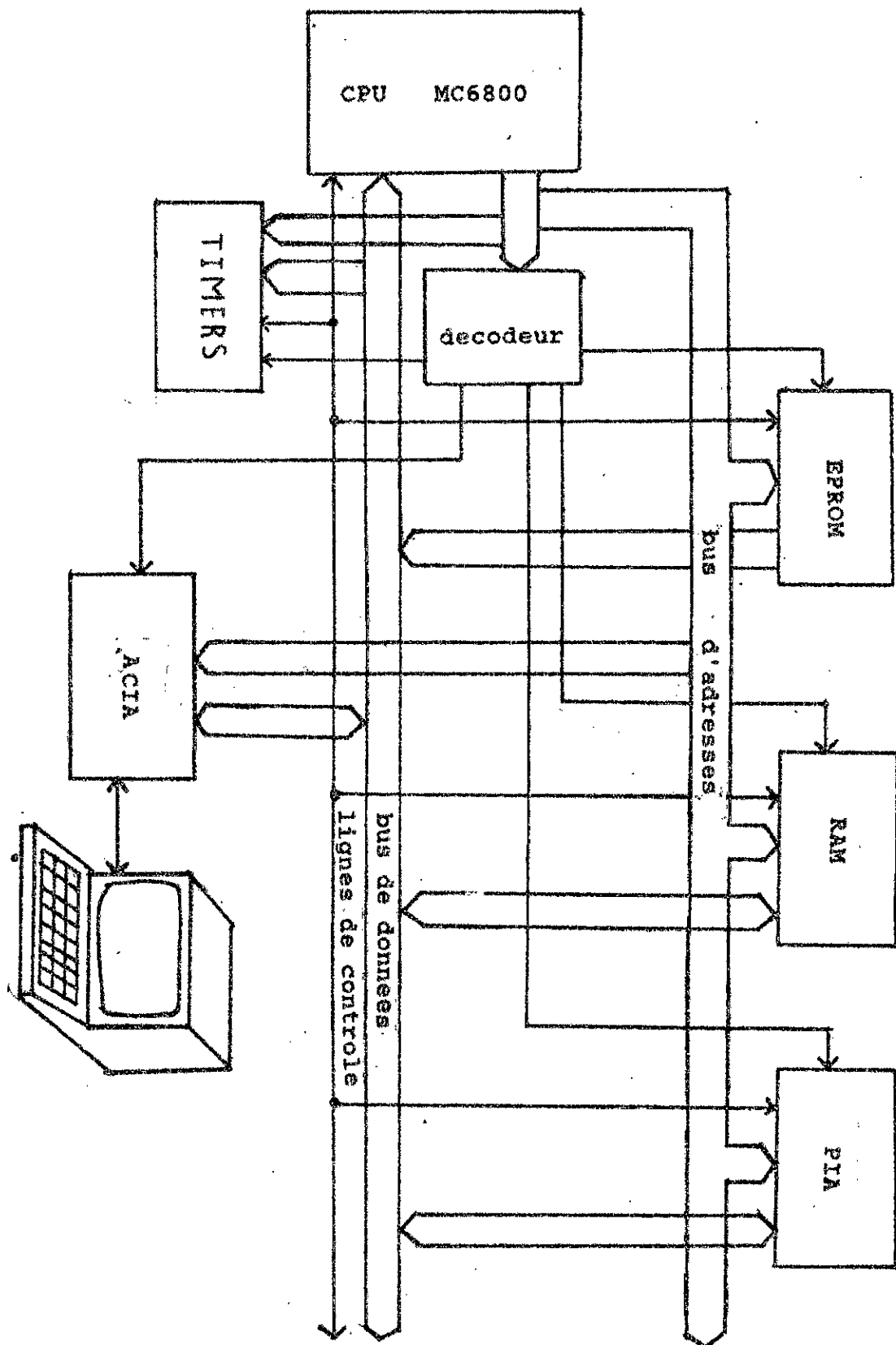
Les lignes PA0 et PA1 du PIA sont a 1 .



Le synoptique de la carte microcalculateur est donne fig  
III.7 .

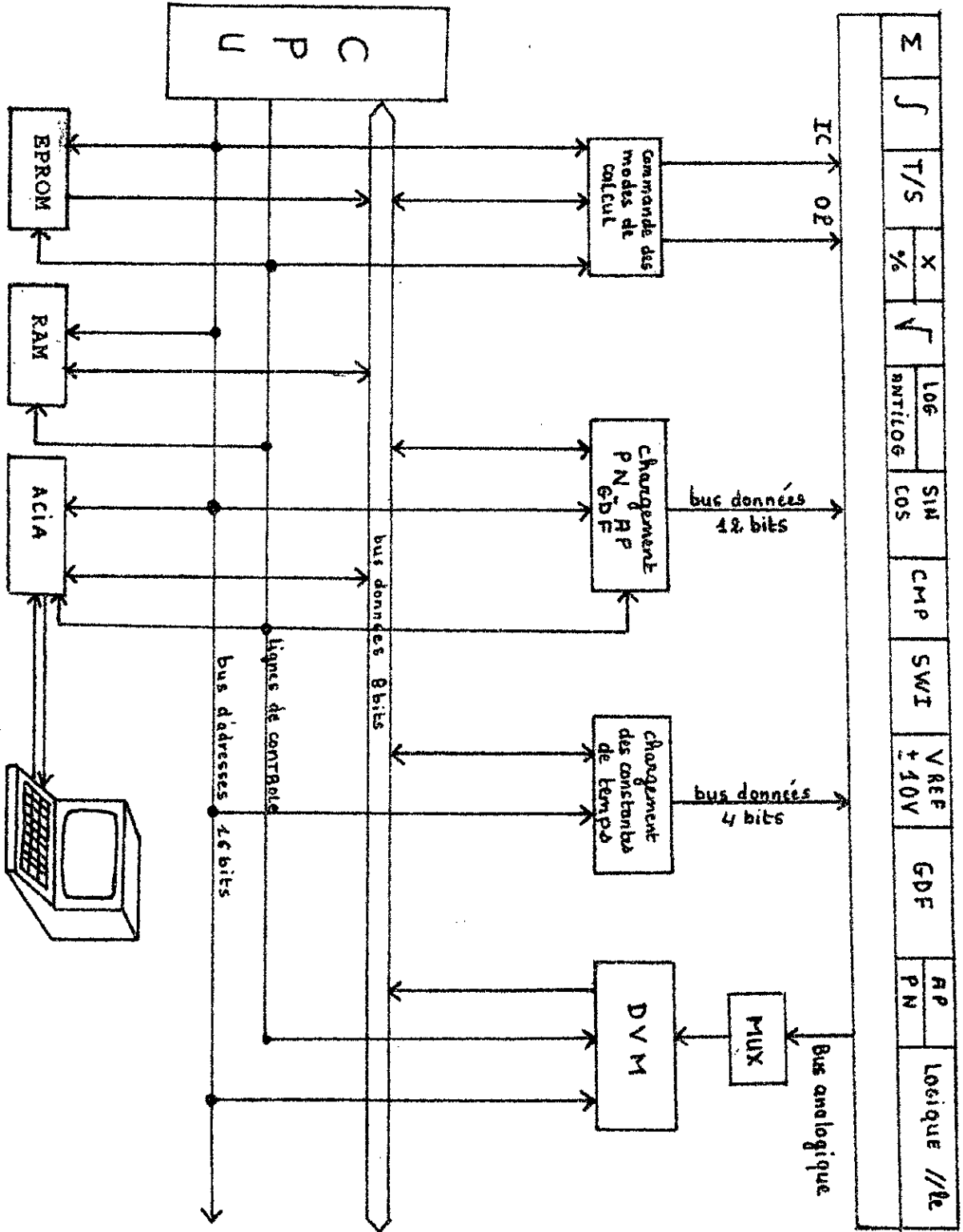
La configuration du calculateur hybride est donnee fig  
III.8 .

(fig III.7) Synoptique de la carte unite centrale



Configuration du calculateur hybride

fig III.8



### III.3 Description du logiciel developpe

---

#### III.3.1 Introduction :

---

Le logiciel que nous avons developpe se compose de deux parties distinctes :

- Une partie destinee a la gestion des differentes ressources dont a besoin le systeme (ces ressources peuvent etre aussi bien materielles que logicielles). Cette partie est formee d'un ensemble de programmes realisant les fonctions de base d'un systeme d'exploitation multitaches temps reel.

Nous appellerons cet ensemble de programmes : moniteur ou noyau.

- La seconde partie est formee de programmes correspondant aux specifications de l'application a laquelle est destine ce systeme.

Nous avons dans notre cas, effectue un decoupage de l'application en un certain nombre de taches. Ces taches sont celles necessaires a la resolution d'un systeme d'equations differentielles.

La premiere partie de ce chapitre est consacree a la description du moniteur developpe.

Dans la deuxieme partie, on decrira les fonctions realisees par les differentes taches en faisant apparaitre l'aspect interactif entre l'utilisateur et la machine ainsi obtenue.

### III.3.2 Structure et fonctionnement du moniteur

---

#### III.3.2 a Structure du moniteur

---

Le moniteur que nous avons developpe est un ensemble de programmes destines a gerer entre autre un certain nombre de taches qui cooperent dans le but d'automatiser la gestion du calculateur analogique.

La structure logicielle de ce moniteur se presente sous forme de couches successives.

A chacune des couches correspond un certain nombre de programmes realisant des fonctions distinctes d'une couche a l'autre.

La couche la plus profonde realise les operations de base du moniteur. Elle est la plus proche du materiel.

Lorsqu'on passe d'une couche a l'autre en s'eloignant du materiel, les operations correspondantes deviennent de plus en plus complexes.

Elles utilisent les operations des couches inferieures.

On distingue les couches suivantes :

- Initialisation du systeme.

C'est un programme permettant

- \* d'une part l'initialisation du materiel

- \* d'autre part l'initialisation du logiciel

- Les primitives de synchronisation P et V

- La gestion des interruptions (materielles ou logicielles)

- L'ordonnancement des taches ( Scheduling)

D'autres couches peuvent être développées à partir de celles citées précédemment.

#### Objets manipulés par le moniteur

Les opérations du moniteur que nous avons conçu sont réalisées par la manipulation de quatre types d'objets :

- Les tâches
- Les événements
- Les sémaphores
- Les files

#### A / Les tâches

Les tâches manipulées par tout moniteur temps réel sont de deux types :

on distingue :

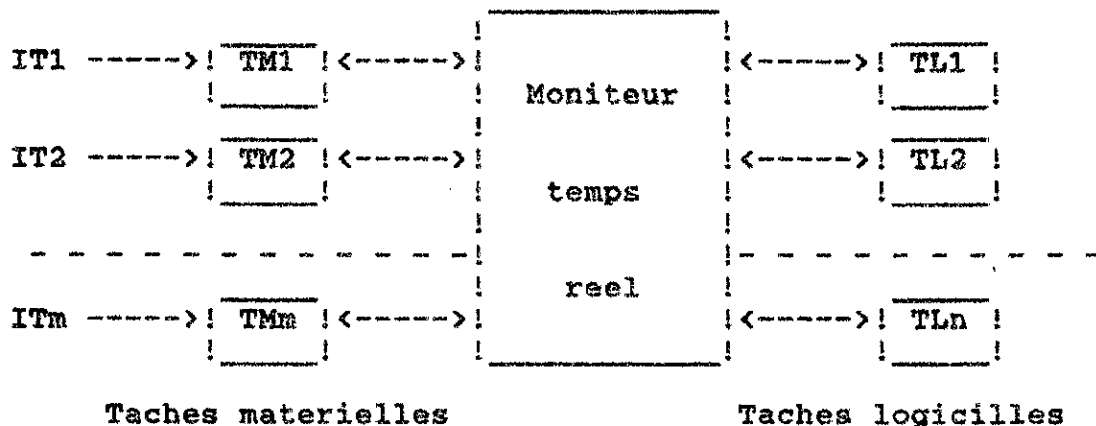
##### \* Les tâches matérielles (ou immédiates)

Elles sont liées à l'extérieur et ont des contraintes temps réels. Ces tâches sont associées à des interruptions matérielles et elles sont gérées par le système d'interruption de la machine (priorité, masquage, validations, branchement)

##### \* Les tâches logicielles (ou différées)

Ces tâches ne sont pas directement liées à l'extérieur. On leur associe une priorité logicielle gérée par le moniteur temps réel.

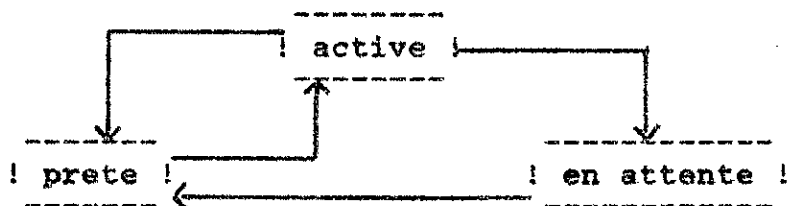
La structure d'un tel logiciel peut se resumer de la facon suivante :



+ Les etats d'une tache

Une tache de notre application peut se trouver dans trois etats : tache active, tache en attente, tache prete.

Les transitions possibles entre ces etats pour une tache du systeme peuvent etre representees de la facon suivante :



Active : Dans cet etat, l'unite de traitement lui est affectee. Il faut pour cela qu'elle satisfasse certains criteres : plus forte priorite, disponibilite des ressources qu'elle utilise.

En attente : Elle passe dans cet etat lorsqu'etant en execution elle demande une ressource deja occupee, ou lorsque pour se poursuivre elle necessite l'apparition d'un evenement.

Après disparition de la condition de blocage, cette tâche passera dans l'état prête.

Prête : Une tâche est prête lorsqu'elle est candidate à être exécutée par l'unité de traitement. Toutes les ressources autres que le processeur sont disponibles.

+ Gestion des tâches : Le passage d'une tâche d'un état à l'autre est réalisé à l'aide du module ordonnanceur qui fait appel aux opérations suivantes :

- Demarrer (une tâche) : qui consiste à allouer le processeur à la tâche lorsque celle-ci est prête.
- Arrêter (une tâche) : qui consiste à suspendre l'exécution d'une tâche.

Dans ce cas, deux possibilités peuvent se présenter :

1 ) La tâche passe de l'état active à l'état prête lorsqu'elle reçoit une interruption d'une tâche plus prioritaire.

2 ) La tâche passe de l'état active à l'état en attente lorsqu'elle sollicite une ressource dont elle ne peut disposer. Cette ressource en effet, peut ne pas être libre. De plus cette tâche peut être en attente d'un événement.

- Continuer : Pour qu'une tâche suspendue puisse reprendre son exécution, il faut qu'elle reçoive un signal de reactivation. Ceci est réalisé à l'aide de l'opération "continuer".
- Se terminer : Une tâche en exécution se terminera par une requête Fin de tâche adressée au moniteur.



De ce fait, la tache courante est retiree au processeur.  
Elle mise dans l'etat prete.  
Toute reactivation de cette tache necessitera l'operation  
"Demarrer".

B / L'evenement :

Un evenement agit sur un objet contenant une  
marque representant le fait qu'une condition attendue par  
une tache est realisee ou non.

Cet objet est materialise par un semaphore.

L'evenement agit sur un semaphore a l'aide de deux  
primitives P et V.

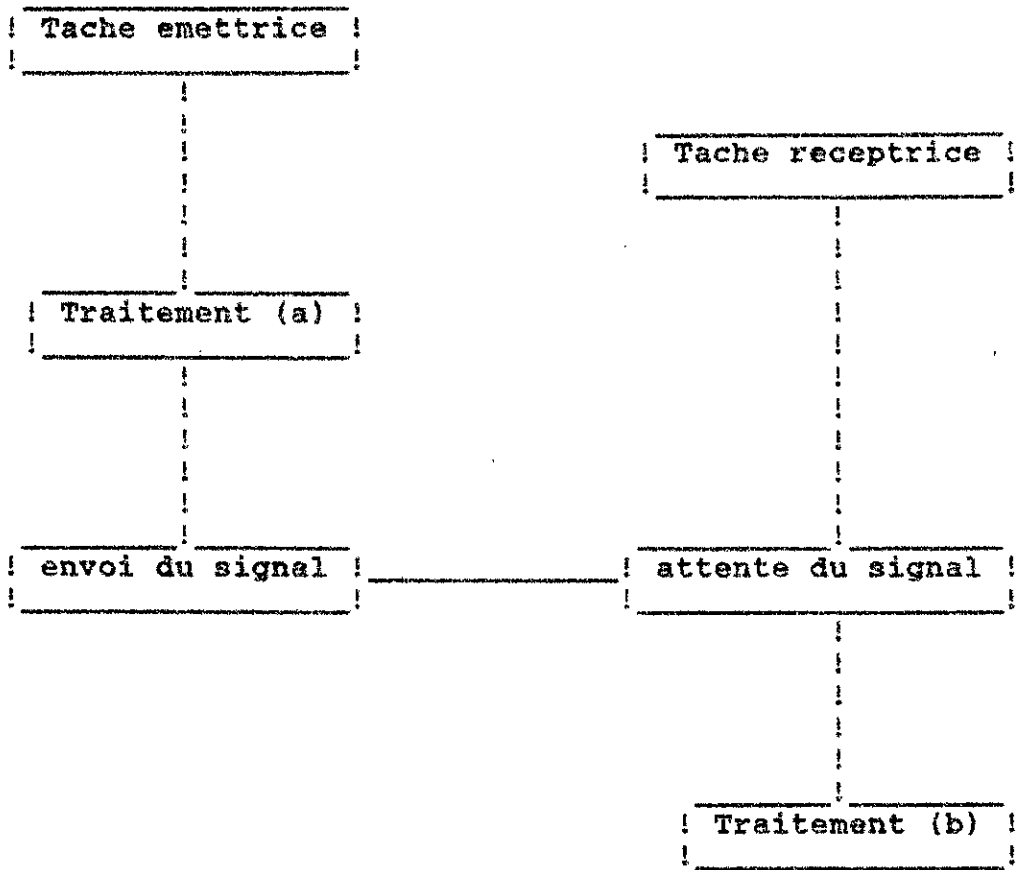
L'evenement est en fait l'outil de signalisation entre  
les taches.

Une tache signalant un evenement est une tache  
emettrice.

Si elle attend un evenement, c'est une tache receptrice.

La manipulation d'un evenement est representee fig III.9

Dans le cas ou l'evenement est externe, il correspond  
alors a une interruption cablee.



(fig III.9) Manipulation d'un evenement

C / Les semaphores :

Ce sont des structures de données de deux bytes destinées à régir les relations entre tâches. Ils permettent de résoudre les problèmes d'exclusion mutuelle ainsi que les problèmes de synchronisation et de communication entre tâches.

Exclusion mutuelle :

Dans le cas où les tâches évoluent simultanément, il arrive souvent qu'une ressource commune soit sollicitée par plusieurs tâches. On arrive donc à un conflit de partage de ressource entre tâches. La notion d'exclusion mutuelle implique un certain nombre de propriétés.

- A tout instant, une tâche au plus peut utiliser la ressource.
- Si plusieurs tâches sont bloquées en attente d'une ressource libre, l'une de ces tâches doit pouvoir accéder à cette ressource.

Pour résoudre ce genre de problèmes on définit un semaphore associé à cette ressource.

L'un des bytes contient une variable indiquant si la ressource est libre ou non.

Les opérations qu'effectue une tâche sur un semaphore sont les suivantes :

- 1 ) Elle doit pouvoir consulter cette variable pour savoir si elle a accès ou non à cette ressource.
- 2 ) Elle doit pouvoir signaler la libération de cette ressource lorsqu'elle n'en a plus besoin.

Les operations que l'on vient de citer se presentent sous forme de procedures qui sont appelees par les taches considerees.

Ces procedures sont appelees primitives.

Une tache sollicitant une ressource fait appel a la primitive P.

Une tache liberant une ressource fait appel a la primitive V.

Le detail de ces primitives sera donne plus loin.

#### Synchronisation et communication

En general, dans un systeme, aucune tache n'est isolee des autres.

En effet, a certains moments de son execution toute tache communique avec d'autres taches en echangeant des signaux de synchronisation et eventuellement des donnees.

Deux aspects fondamentaux sont a considerer :

- La synchronisation entre taches : une tache active changeant d'etat permet a une autre tache de changer elle meme d'etat.

- La communication des donnees d'une tache a une autre.

Si l'on place du point de vue systeme, il est tres commode de considerer comme des ressources les signaux de synchronisation echanges par les taches (evenements).

A ce moment la, la notion de blocage se confond avec l'absence d'une ressource necessaire a l'execution de l'instruction suivante. De ce fait on associera a ces evenements (signaux de synchronisation) un semaphore que

D / Files :

-----  
On distingue deux types de files manipulees par le moniteur.

1 ) Les files des taches pretes

-----  
Le moniteur developpe dans notre application gere un ensemble de taches suivant un ordre de priorites. Nous avons donc defini differents niveaux de priorite (11) L'ensemble de ces niveaux de priorite se trouvent dans la " liste des taches pretes ".

Il peut arriver que dans une application deux ou plusieurs taches aient le meme niveau de priorite.

De ce fait a chacun de ces niveaux, on associe a l'initialisation une file de taches ayant ce meme rang de priorite.

La strategie de gestion de ces files et du types FIFO.

2 ) Les files de taches en attente a un semaphore

-----  
Au cours de l'execution des taches, il peut arriver que plusieurs d'entre elles demandent une meme ressource ou attendent un meme signal de synchronisation pour continuer leur execution (ou pour demarrer leur execution).

Tant que cette ressource n'est pas disponible ou que le signal de synchronisation n'est pas signale les taches sont mises en file d'attente au semaphore correspondant.

Ce sont les primitives P et V qui gerent les files d'attente au semaphore.

La gestion de ces files est a strategie FIFO.

l'on appellera semaphore de synchronisation et de communication.

De la meme facon que pour les semaphores associes a une ressource, ce sont les primitives P et V qui agiront sur ces semaphores pour permettre une bonne synchronisation entre taches et eventuellement l'echange de donnees entre elles.

Dans le cas de notre application nous avons defini un certain nombre de semaphores de synchronisation.

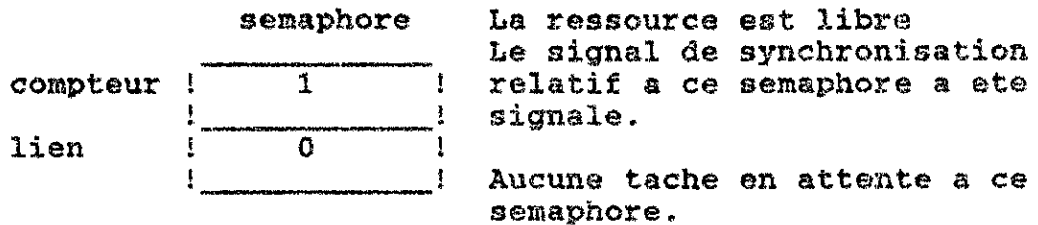
Situations relatives a la disponibilite d'une ressource  
ou au mecanisme de synchronisation

Les semaphores sont formes de deux octets.

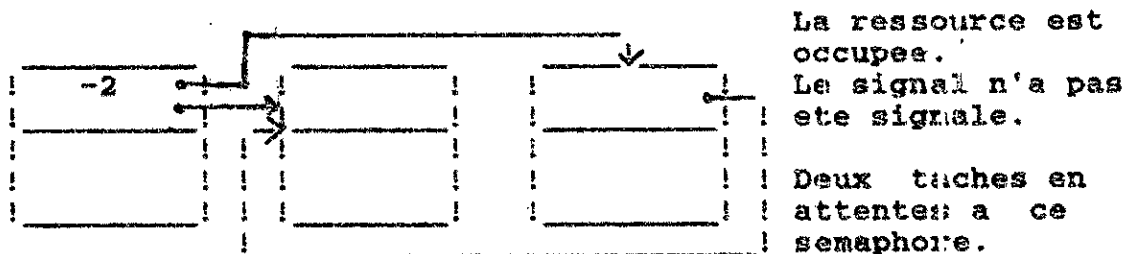
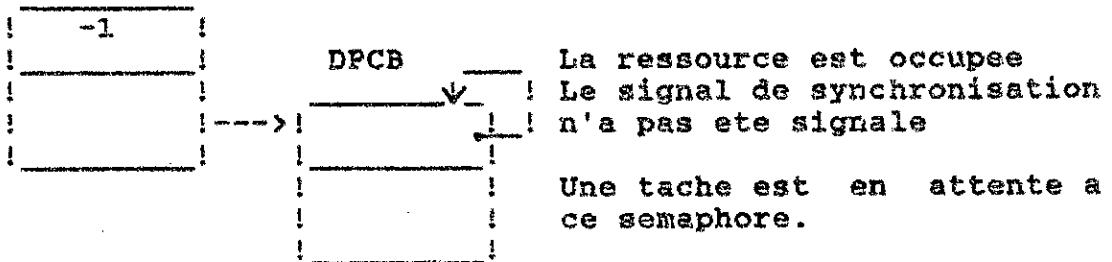
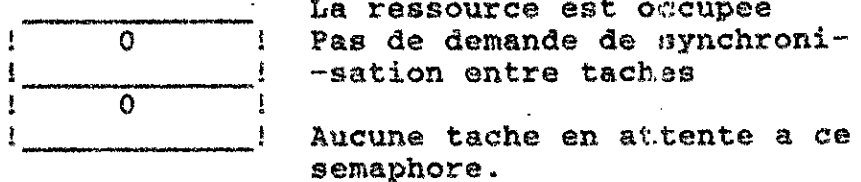
- Le premier etant le compteur. La valeur que peut prendre celui-ci est negative, nulle ou egale a 1.
- Le second octet etant le lien. Il contient l'adresse du DPCB que l'on definira plus loin).

On peut resumer les differentes situations de la facon suivante :

a )



b )



## Structure de donnees

Avant de presenter le fonctionnement du moniteur, il est necessaire de definir les structures de donnees qui sont manipulees par celui-ci.

Ces structures de donnees sont des elements logiciels constitues de blocs de cellules memoire contenant des donnees specifiques aux objets manipules par le moniteur (tache, ressource, evenement, file)

Les structures de donnees mises en jeu dans notre application sont rassemblees en tables dans un module appele configuration du systeme.

Celle-ci est formee de deux tables : ITT et IST

### a ) Table ITT

A chaque tache on associe un descripteur statique appele SPCB contenant des informations non volatiles relatives a une tache telles que : priorite, longueur de pile et pointeur d'extension. Ce dernier est l'adresse d'un descripteur dynamique appele DPCB. Il renferme des informations qui evoluent suivant l'etat dans lequel se trouve la tache.

Les SPCB sont regroupees dans une table appelee ITT (Initial Task Table).

Les DPCB ne sont crees qu'a l'initialisation du systeme et ce a partir de la table ITT.



### Structure d'un SPCB

-----

C'est une structure formee de 7 Bytes. Chaque SPCB se presente sous la forme suivante :

nom de la tache	2 octets
priorite	1 octet
longueur de la pile	2 octets
adresse du DPCB	2 octets

### 2 ) Table IST

-----

La synchronisation des taches se fait a l'aide d'un mecanisme regi par les differents temoins de synchronisation : les semaphores.

L'ensemble des semaphores d'une application sont regroupes dans une table appelee IST (Initial Semaphore Table).

### Structure des DPCB

-----

Dans le cas de notre application, les informations necessaires a la gestion dynamique des taches sont les suivantes :

- le lien permettant le chainage des taches en attente dans la liste des taches pretes ou en attente a un semaphore.

celui-ci est contenu dans un octet.

- L'adresse du pointeur de pile (2 octets) correspond a l'adresse de la pile reservee a la tache correspondante.

- La priorite de la tache (1 octet)

lien
SP
priorite

Les piles sont des structures de donnees necessaires a la gestion dynamique des taches.

Ce sont des zones memoires de longueur donnee permettant la sauvegarde du contexte d'une tache chaque fois que celle-ci est interrompue.

La longueur de la pile est donnee dans le SPCB.

Le pointeur de pile est contenu dans le DPCB. Il correspond a l'adresse de la pile permettant de recuperer le contexte de la tache.

Ces zones memoires sont allouees a chacune des taches a l'initialisation du systeme.

### III.3.2 b Fonctionnement du moniteur

#### Initialisation du systeme

Le programme d'initialisation du systeme se presente sous forme de deux modeles distincts :

- Module d'initialisation du materiel
- Module d'initialisation du logiciel

#### Initialisation du materiel

Du point de vue materiel, les elements a initialiser sont :

- L' ACIA permettant le dialogue homme-machine
- Les PIA d'adressage des operateurs et de chargement des donnees telles que : coefficients des potentiometres,

gains des amplificateurs ...

- Le PIA de chargement des constantes de temps des integrateurs

- Les timers commandant les entrees IC et OP des integrateurs. A l'initialisation les timers doivent etre inhibes.

- Le PIA de la carte DVM

- "le PICU"

Initialisation du logiciel

---

Le module d'initialisation du logiciel est un programme permettant :

- de creer les DPCB des taches a partir des SPCB

- De creer la liste des taches pretes.

Celle-ci est formee de onze niveau de priorite.

Chaque niveau est constitue de deux octets.

Le premier (HEAD) contient l'adresse sur 8 bits du DPCB de la premiere tache prete de la file ayant ce niveau de priorite.

Le second (TAIL) contient l'adresse du DPCB de la derniere tache de la file.

A l'initialisation, toute les taches du systeme sont pretes.

- Roles des primitives P et V du Scheduler

---

Les primitives P et V que nous avons choisies ont ete definies par DIJKSTRA (68). Ce sont les mecanismes de base permettant de resoudre les problemes de synchronisation et de communication entre taches ainsi

que d'exclusion mutuelle. Le Scheduler est le module ordonnanceur. Il decide de l'enchainement des taches chaque fois qu'une tache est interrompue ou qu'elle a termine son execution (voir organigramme) (fig III.12).

Il alloue le processeur aux taches suivant le principe des priorites. Pour cela il scrute la liste des taches pretes. Il met en execution la premiere tache prete qu'il trouve dans cette liste.

#### Primitive P

Cette primitive est appelee par la tache qui sollicite une ressource ou qui attend un signal de synchronisation d'une autre tache pour son activation ou pour continuer son execution. Par un appel de la primitive P, la tache sera mise en attente au semaphore correspondant a la ressource ou a l'evenement attendu.

Le role de la primitive P est donne dans un organigramme (fig III 10)

#### Primitive V

Cette primitive est appelee par une tache liberant une ressource ou par une tache decidant d'activer une autre tache (synchronisation).

Le role de cette primitive est donne dans un organigramme (fig III 11).

Les programmes correspondants sont écrits en assembleur.

Les primitives P et V ne peuvent être interrompues en cours d'exécution.

Pour cela, dès la première instruction, on inhibe les interruptions.

On habilite les interruptions dès que l'une des tâches prend possession du processeur.

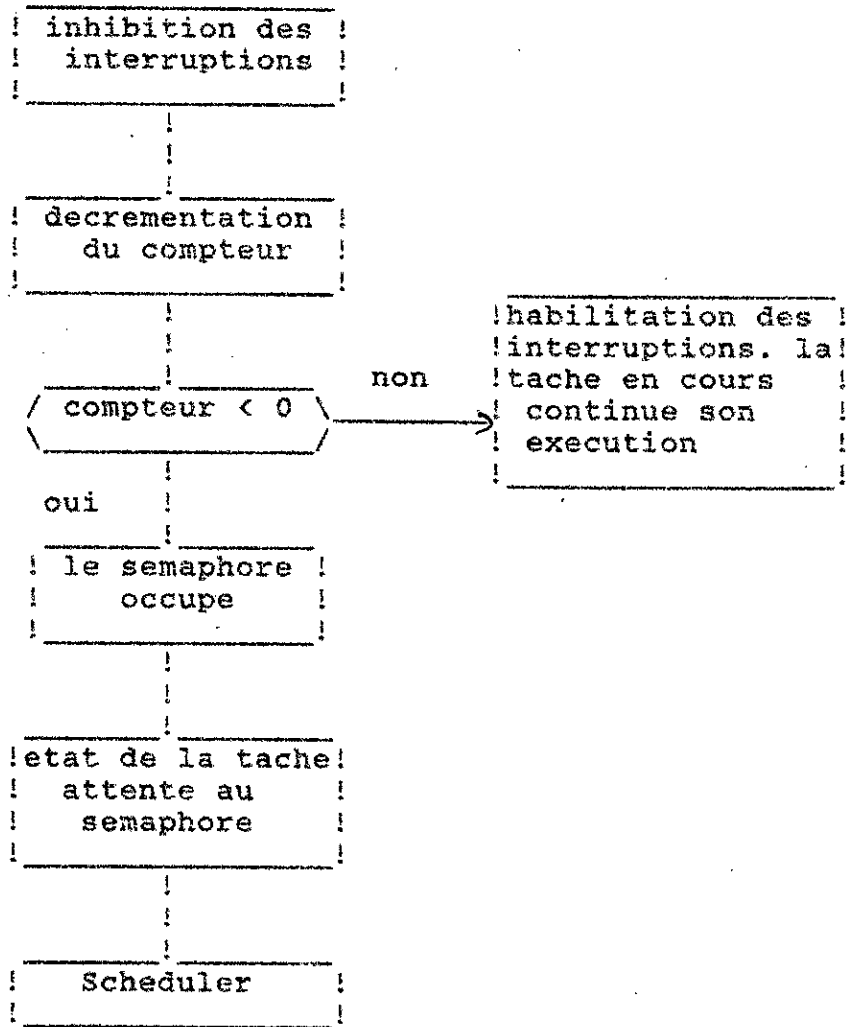


fig III.10 ORGANIGRAMME DE LA PRIMITIVE P

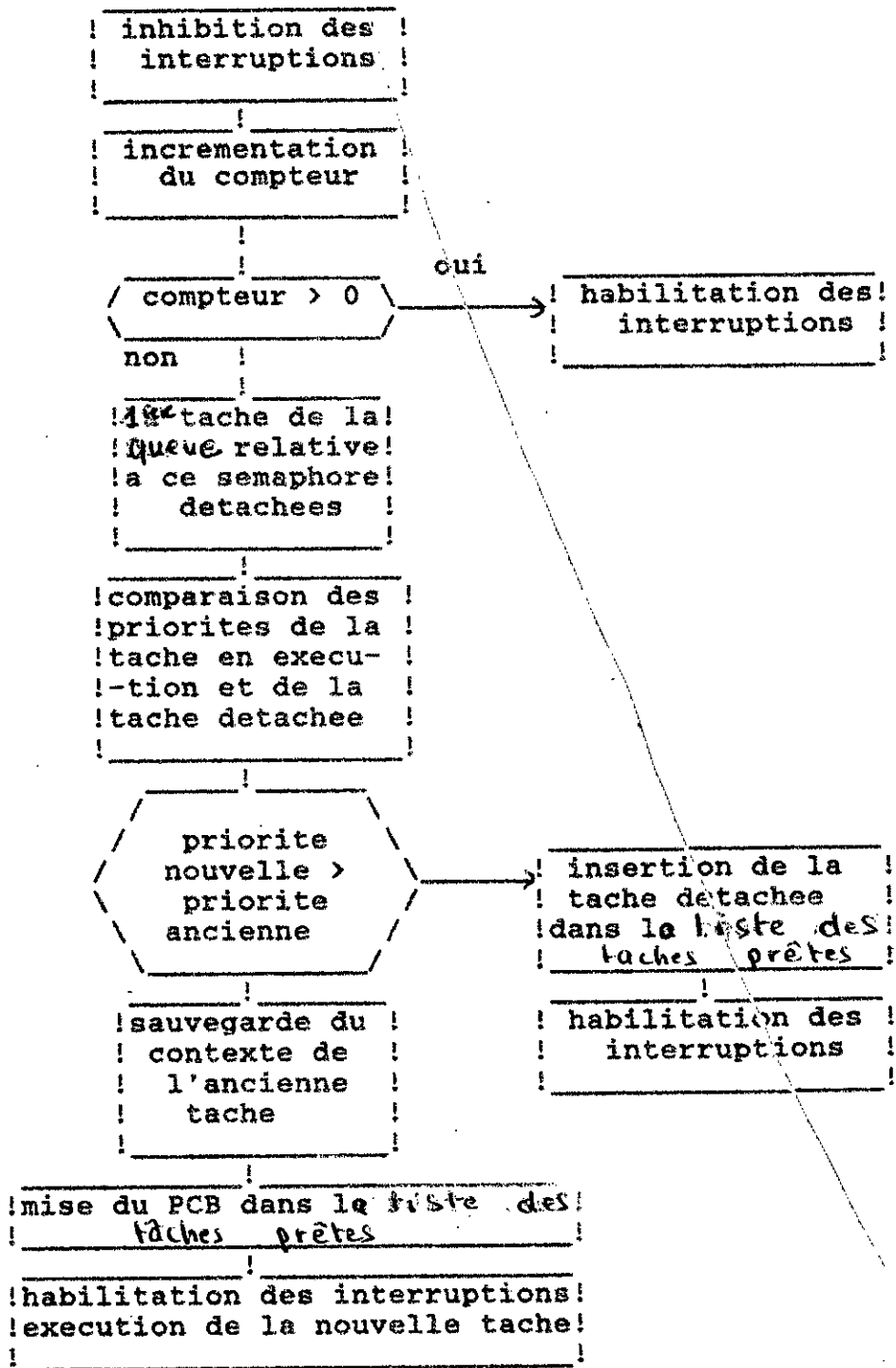


fig III.11 ORGANIGRAMME DE LA PRIMITIVE V

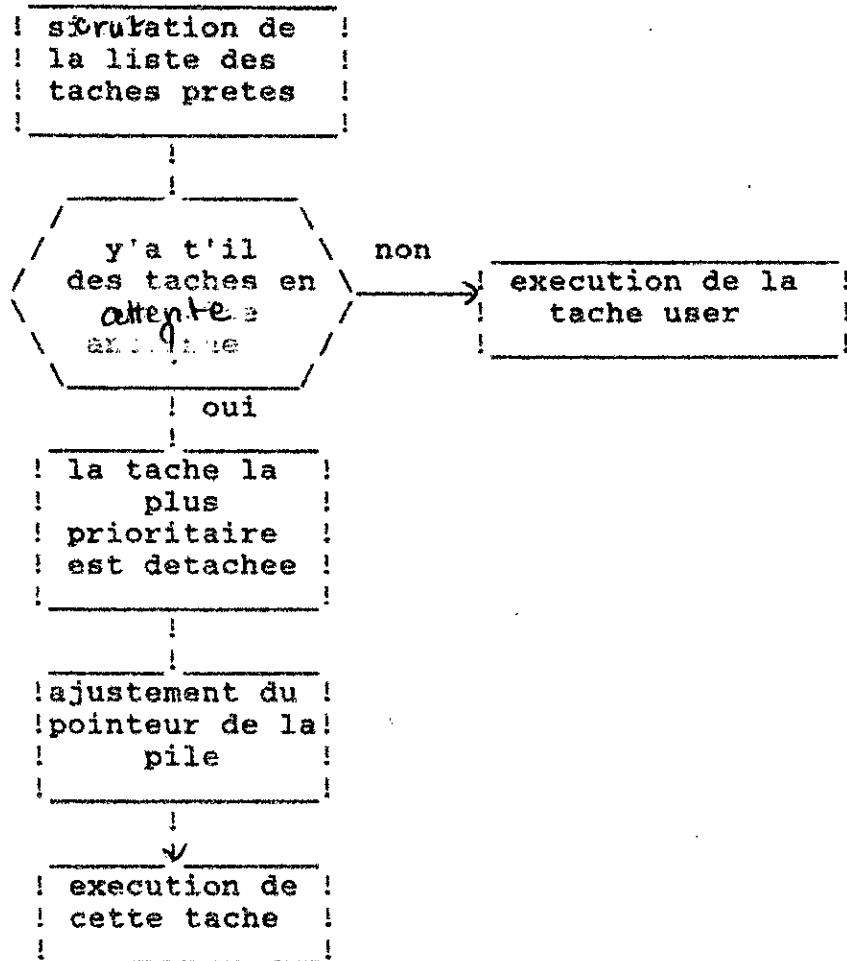


fig III.12 ORGANIGRAMME DU SCHEDULER



### III.4 Interaction homme-machine et definition des taches

---

#### III.4.1 Interaction homme machine

---

L'utilisateur dispose d'une console de visualisation lui permettant le dialogue avec la machine.

Le mode de dialogue que nous avons choisi est le mode par MENUS.

Celui-ci consiste a afficher differents menus permettant a l'utilisateur de choisir.

- L'une des operations suivantes : Visualisation, chargement, calcul.
- Un operateur tel que : potentiometre numerique, amplificateur programmable, Integrateur ou GDF.
- L'un des modes de calcul suivants : normal, repetitif iteratif.

Les tableaux correspondants a ces menus sont representes en fig III.13

Menu I
V = Visualisation
C = Chargement
E = Calcul
!Choisir entre V, C ou E !

S/Menu I	
P	= potentiometre
A	= Amplificateur
I	= Integrateur
E	= Sortie

fig III.13

S/Menu II	
N	= Mode normal
R	= mode repetitif
I	= mode iteratif

#### III.4.2 Definition des taches

---

Dans le decoupage en taches que nous avons effectue, nous nous sommes limites a l'ensemble des fonctions necessaires a la simulation d'un systeme d'equations differentielles. Ce decoupage a fait apparaitre un ensemble de sept taches. Le moniteur est concu pour gerer un ensemble des taches auxquelles on a assigne une priorite.

Ces taches sont :

\*) Tache USER : C'est la tache qui permet l'affichage  
-----  
des differents menus.

En outre, cette tache permet la synchronisation entre  
toutes les taches.

C'est la tache la moins prioritaire - niveau 10 -

\*) Tache 2 :  
-----

Cette tache permet la lecture des tensions de sorties des  
potentiometres ou des amplificateurs, ce qui est  
necessaire pour le reglage des coefficients et des gains.

Le semaphore associe a cette tache est SEMAFF.

C'est la tache la plus prioritaire -niveau 0-

\*) Tache 3 :  
-----

Elle a pour role l'affichage des constantes de temps des  
integrateurs.

Semaphore associe : SEMVSC -priorite 1-

\*) tache 4 :  
-----

Celle-ci permet le chargement des coefficients des  
potentiometres a partir du clavier.

Semaphore associe : SEMCPOT -priorite 2-

\*) tache 5 :  
-----

Elle permet le chargement des gains des amplificateurs a  
partir du clavier.

Semaphore associe : SEMCAMP -priorite 9-

\*) Tache 6 :

-----

C'est la tache permettant le choix d'une constante de temps ainsi que le chargement de celle-ci au niveau de l'un des integrateurs.

Semaphore associe : SEMCINT -priorite 4-

\*) Tache 7 :

-----

Cette tache permet de choisir la duree des cycles IC, OP et H a partir du clavier, de charger les registres de l'horloge programmable et de lancer l'execution du calcul.

Semaphore associe : SEMCAL -priorite 5-

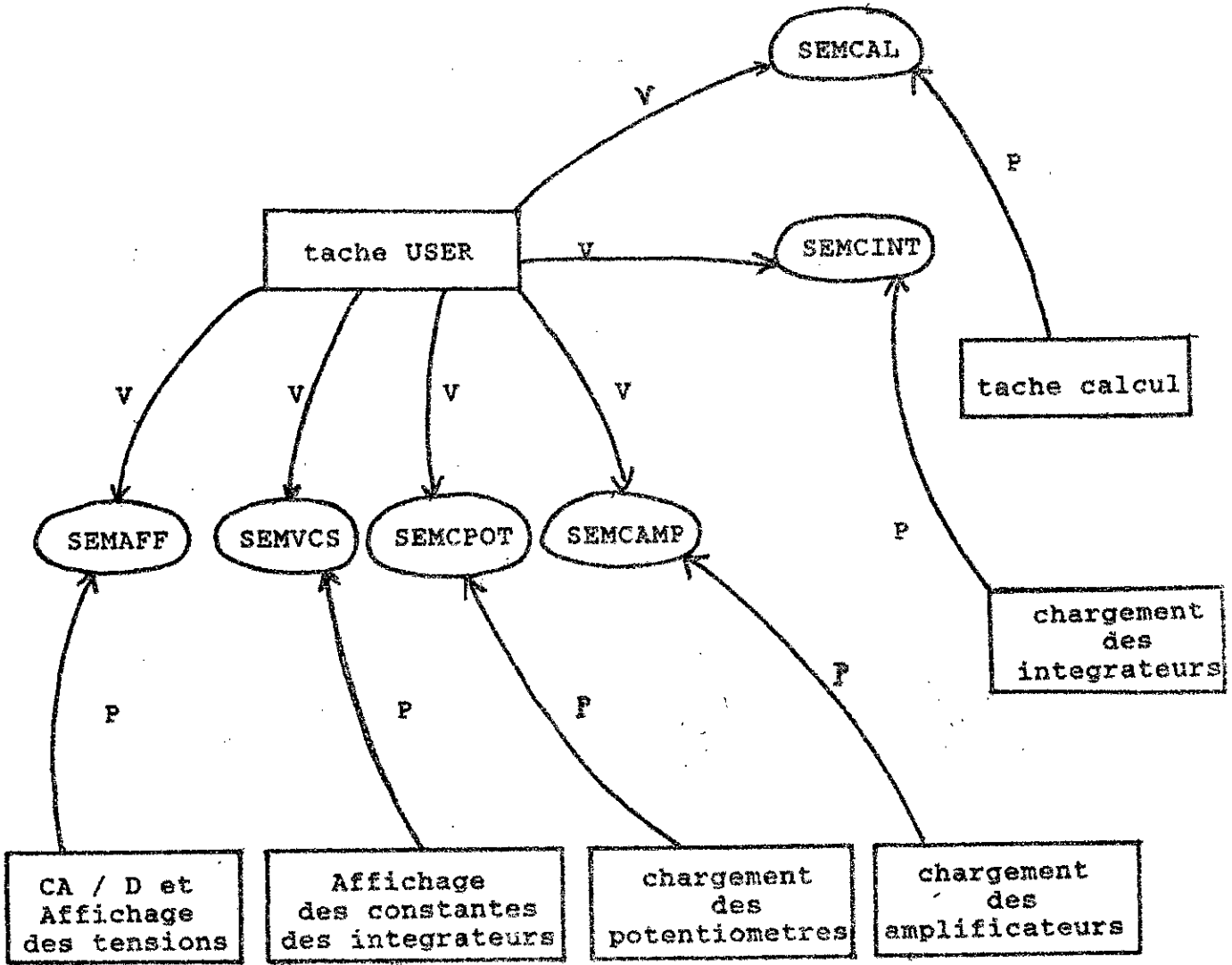
Le fonctionnement dynamique du systeme est represente (fig III.14).

#### III.4.3 Fonctionnement dynamique du systeme.

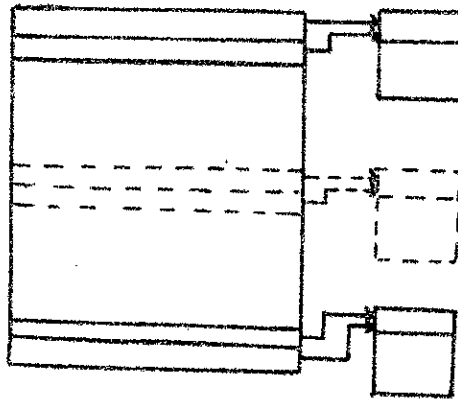
-----

L'enchainement des differentes taches ne s'effectue pas de facon aleatoire. Il depend de l'evolution du systeme qui n'est pas figee. Cet enchainement est assure a l'aide des mecanismes de synchronisation definis dans le moniteur.

A l'initialisation du systeme, toutes les taches sont pretes a l'execution (leur DPCB sont mis dans la liste des taches pretes )



Fonctionnement dynamique du systeme (fig III.14)



Les semaphores relatifs a chacune des taches, sont initialises a zero.

Le scheduler commence par mettre en execution la tache de plus grande priorite (tache 2)

Celle-ci demarre par les instructions suivantes :

LDX SEMAFF      Demande d'activation

JSR PRIMP      Attente d'acquittement de la tache USER

A ce niveau, le compteur relatif au semaphore SEMAFF est decremente. Il devient egal a -1.

Ceci a pour effet de bloquer l'execution de la tache 2.

Celle-ci est detachee de la liste des taches pretes (son DPCB) et elle est mise en attente au semaphore SEMAFF.

Elle ne pourra etre activee que lorsque l'une des taches agira sur le semaphore SEMAFF par l'intermediaire de la primitive V. Ceci est effectue par la tache USER.

Il en est de meme pour toutes les autres taches de sorte que la tache USER sera la premiere a etre executee.

C'est la seule tache qui n'attend pas de signal d'une autre tache pour etre activee.

Les dernieres instructions des taches 2 a 7 sont :

LDX SEMXX

JSR PRIMP

Ceci a pour effet de remettre chacune des taches en attente a leur semaphore correspondant des qu'elles ont termine leur execution.

Lorsque l'utilisateur choisit l'affichage des tensions de sortie des potentiometres, la tache USER affiche le tableau suivant :

TENSION DE SORTIE DES POTENTIOMETRES	
POT1 =	POT5 =
K1 =	K5 =
POT2 =	POT6 =
K2 =	K6 =
POT3 =	POT7 =
K3 =	K7 =
POT4 =	POT8 =
K4 =	K8 =

A ce niveau, la tache USER effectue un appel a la primitive V qui a pour role d'incrementer le compteur du semaphore SEMAFF.

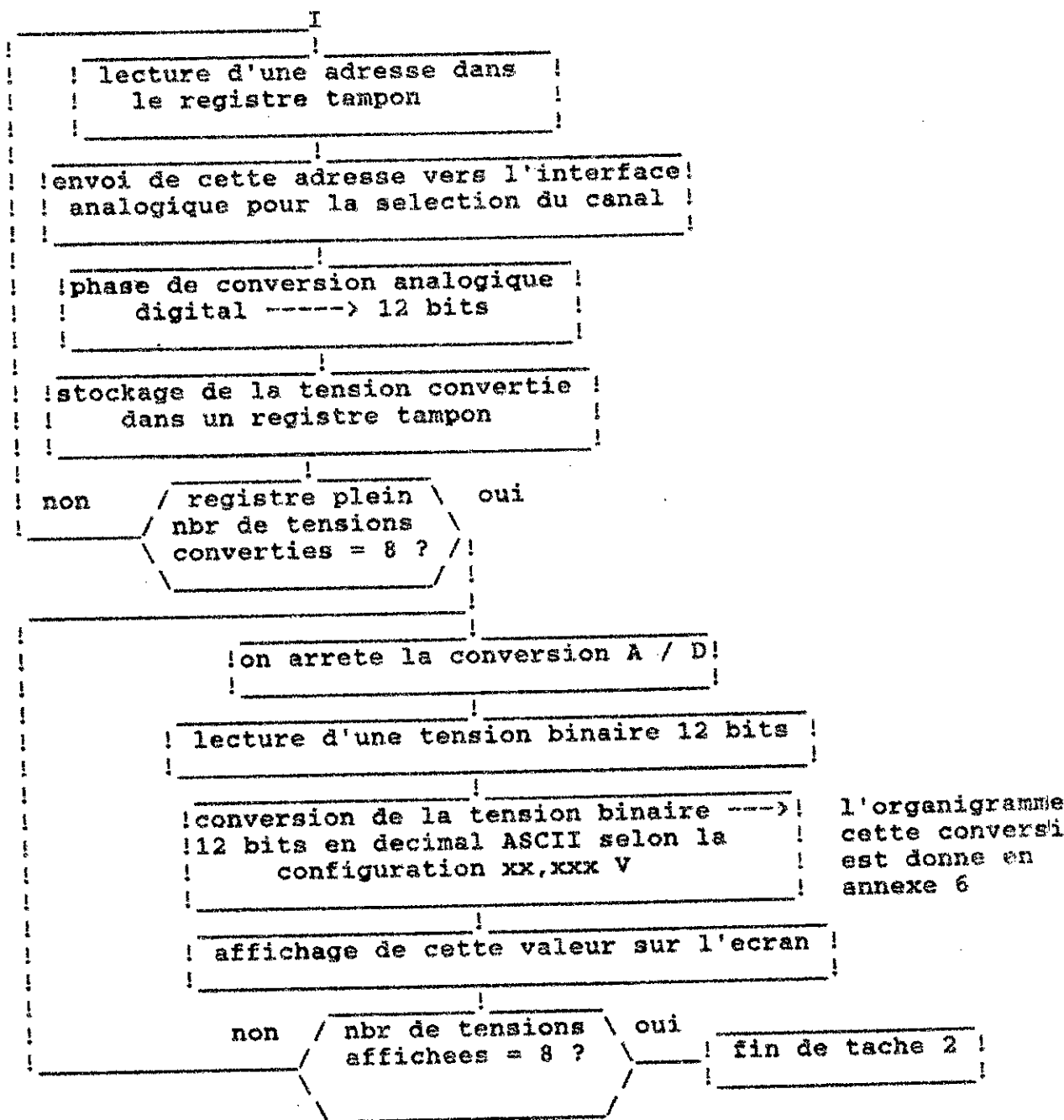
Ceci a pour effet d'activer la tache d'affichage des tensions de sortie des potentiometres. Celle-ci ayant une priorite superieure a celle de la tache USER, cette derniere sera suspendue.

L'organigramme de la tache 2 est donne (fig III.15).

Les adresses des potentiometres sont traduites en un mot binaire de 12 bits. Elles sont memorisees dans un registre tampon.

fig III.15 Organigramme de la tache 2

Affichage des tensions de sortie des potentiometres



La tache 2 se remet en attente au semaphore SEMAFF apres un appel a la primitive P.



Lorsque les valeurs des tensions des potentiometres sont affichees, la tache USER reprend son execution. Toutes les autres taches sont en attente a leur semaphore correspondant.

Lorsque l'utilisateur choisit l'affichage des tensions de sortie des amplificateurs (pour verifier que les gains sont bien ajustes), la tache USER affiche le tableau suivant :

TENSION DE SORTIE DES AMPLIFICATEURS			
AMPLI1 =		AMPLI5 =	
G1 =		G5 =	
AMPLI2 =		AMPLI6 =	
G2 =		G6 =	
AMPLI3 =		AMPLI7 =	
G3 =		G7 =	
AMPLI4 =		AMPLI8 =	
G4 =		G8 =	

La tache USER active alors la tache 2 ce qui permet de convertir et d'afficher les tensions de sortie de tous les amplificateurs.

Une fois le tableau affiche, la tache 2 se remet en attente a son semaphore et la tache USER reprend son execution.

Si l'on se place dans le cas ou l'on veut charger le coefficient de l'un des potentiometres, la tache USER active la tache 4.

Celle-ci démarre en affichant le message suivant :

Chargement du coefficient d'un potentiometre.

NO du potentiometre a charger : \_\_\_\_\_

A ce niveau, la tache 4 est en attente de caractere.

L'utilisateur doit introduire le numero du potentiometre qu'il veut charger. Apres introduction de ce numero on obtient le message suivant :

Valeur du coefficient a charger : \_\_\_\_\_

L'utilisateur doit alors introduire la valeur du coefficient choisi. Cette valeur est comprise entre -1 et +1. Lorsque cette valeur est introduite un retour chariot a pour effet de charger cette donnee au niveau du potentiometre. La tache 4 se remet alors en attente a son semaphore.

L'organigramme de cette tache est donne fig III.16.

- Chargement du gain d'un amplificateur

La tache USER active la tache 5.

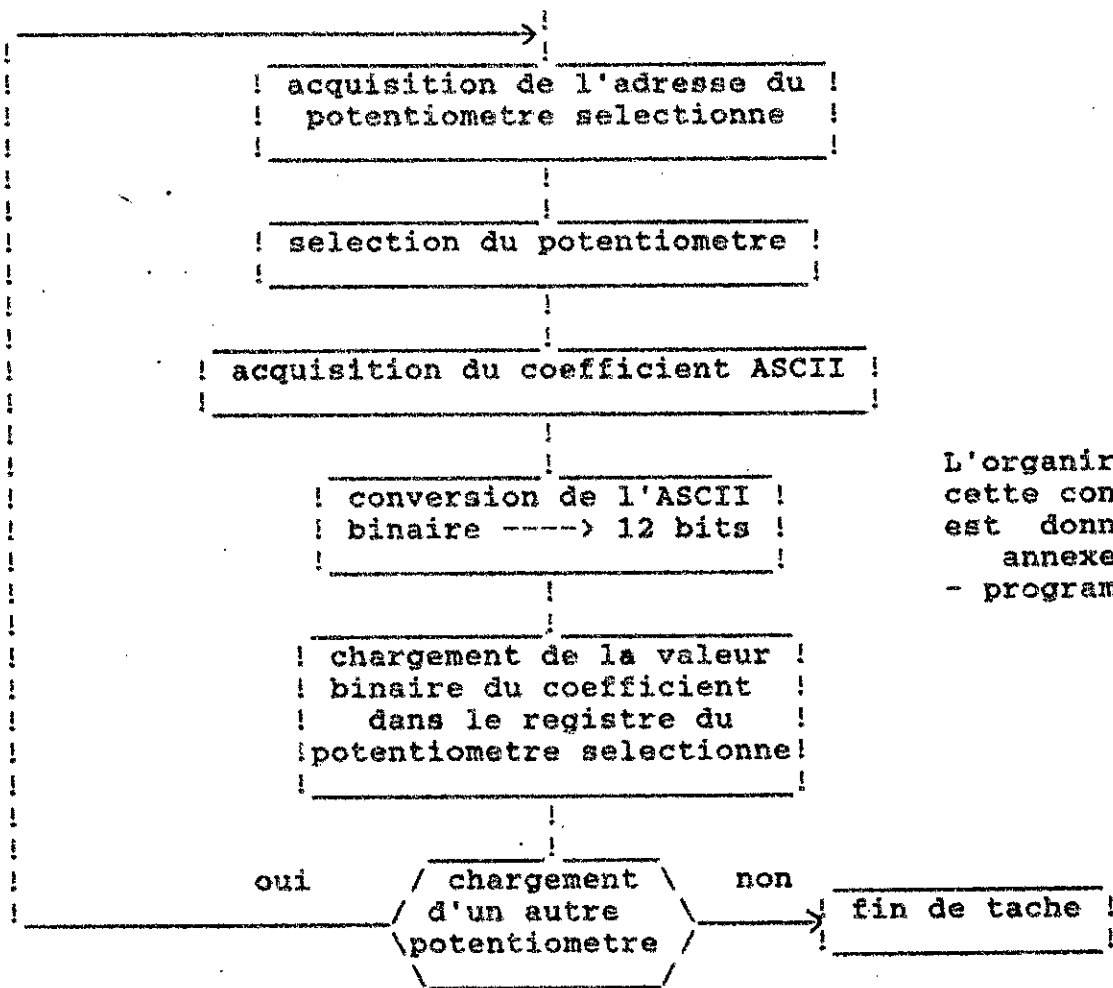
Celle-ci permet l'affichage du tableau suivant :

CHARGEMENT DU GAIN D'UN AMPLIFICATEUR	
NUMERO DE L'AMPLIFICATEUR =	_____
VALEUR DU GAIN A CHARGER =	_____

La valeur du gain doit etre comprise entre 1 et 100.

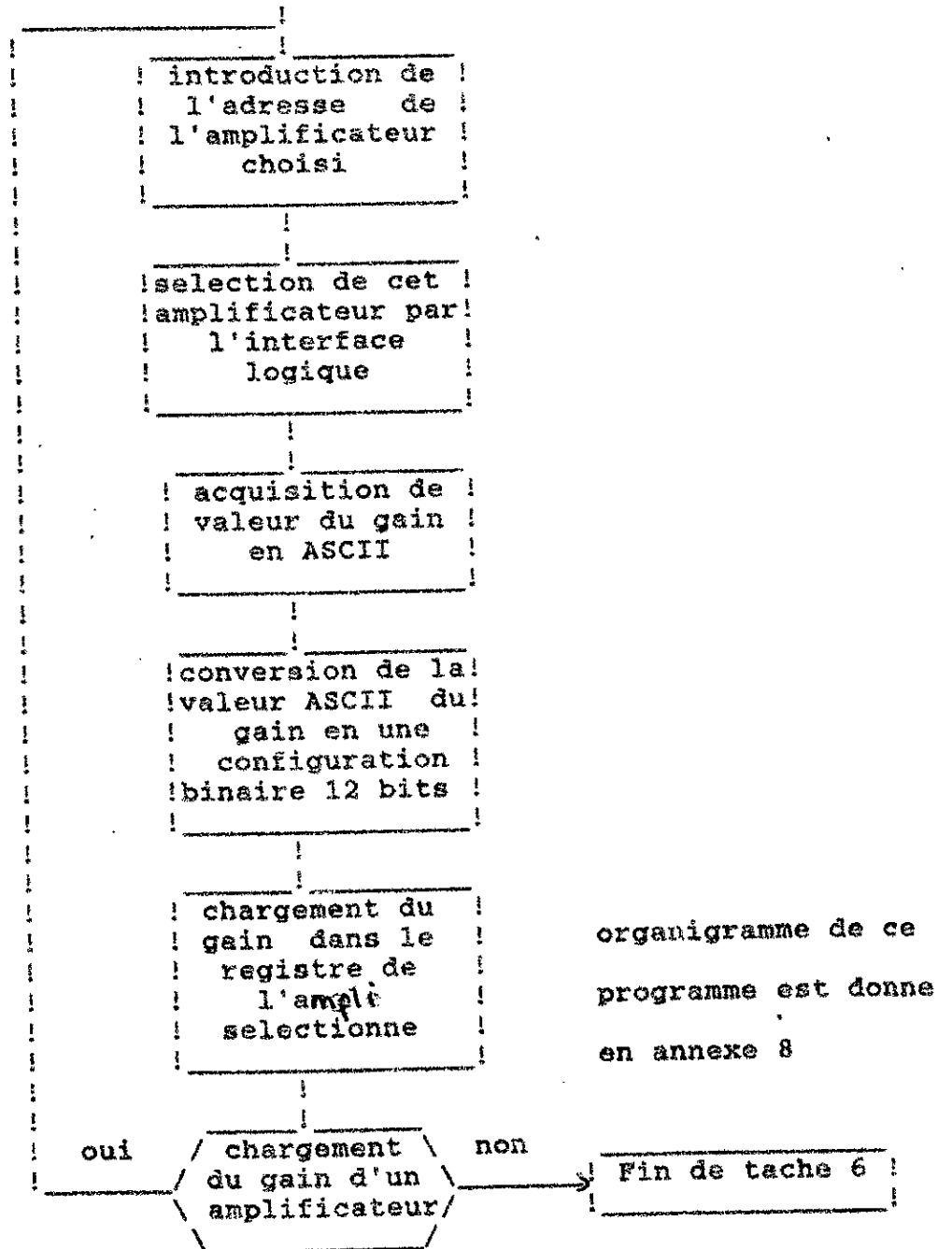
On donne l'organigramme de la tache 5 en fig III.17.

fig III.16 Organigramme de la tache 4  
-----  
chargement d'un coefficient  
-----



L'organigramme de cette conversion est donne en annexe 7 - programme K

fig III.17 CHARGEMENT DU GAIN D'UN AMPLIFICATEUR



Dans le cas du chargement de la constante de temps d'un integrateur , la tache USER active la tache 6 .

Celle ci commence par l'affichage du tableau suivant

```
!                                     !  
!           Chargement de la constante !  
!           de temps d'un integrateur  !  
!                                     !  
!   Numero de l'integrateur =         !  
!   Constante d'integration choisie = !  
!                                     !
```

L'organigramme de cette tache est donne fig III.18

L'utilisateur a la possibilite de verifier les valeurs des constantes de temps de chaque integrateur .

Ceci est possible lorsque la tache USER active la tache 7 .

On obtient le tableau suivant :

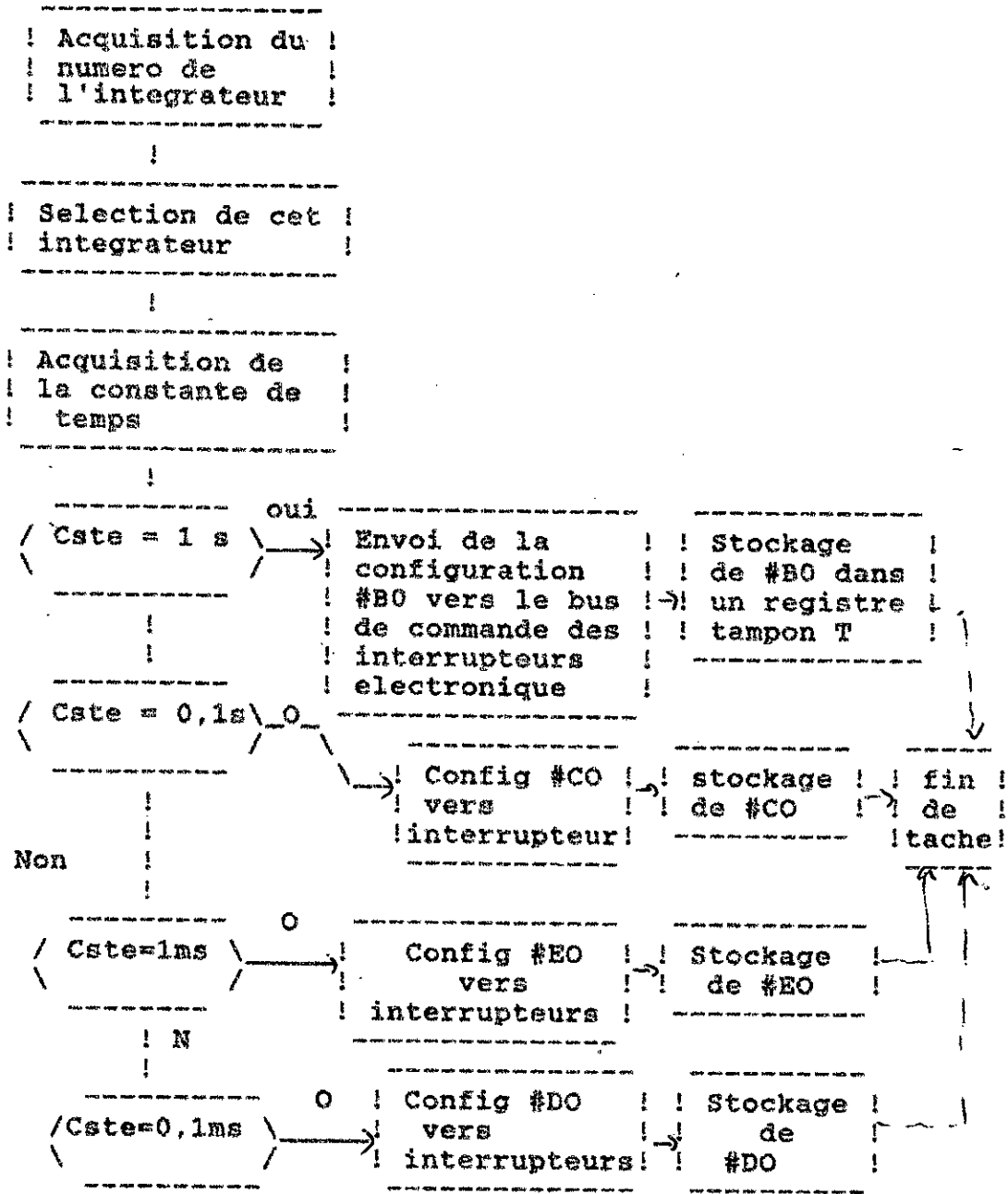
```
-----  
!           Constantes de temps       !  
!           des integrateurs          !  
!                                     !  
!   Cste 1 = . .           Cste 3 = . . !  
!   Cste 2 = . .           Cste 4 = . . !  
!                                     !  
-----
```

Nous nous sommes limite a quatre integrateurs necessaires a la resolution d'un systeme de quatre equations differentielles du premier ordre .

L'organigramme de la tache 3 est donne fig III.19

Les adresses sur 12 bits des integrateurs sont contenues dans un registre tampon I

Les configurations permettant la commande des interrupteurs electroniques pour le choix d'une constante de temps sont stockees dans un registre tampon II .



(fig III.18) CHARGEMENT DE LA CONSTANCE DE TEMPS D'UN INTEGRATEUR

lecture des constantes  
de temps des  
integrateurs

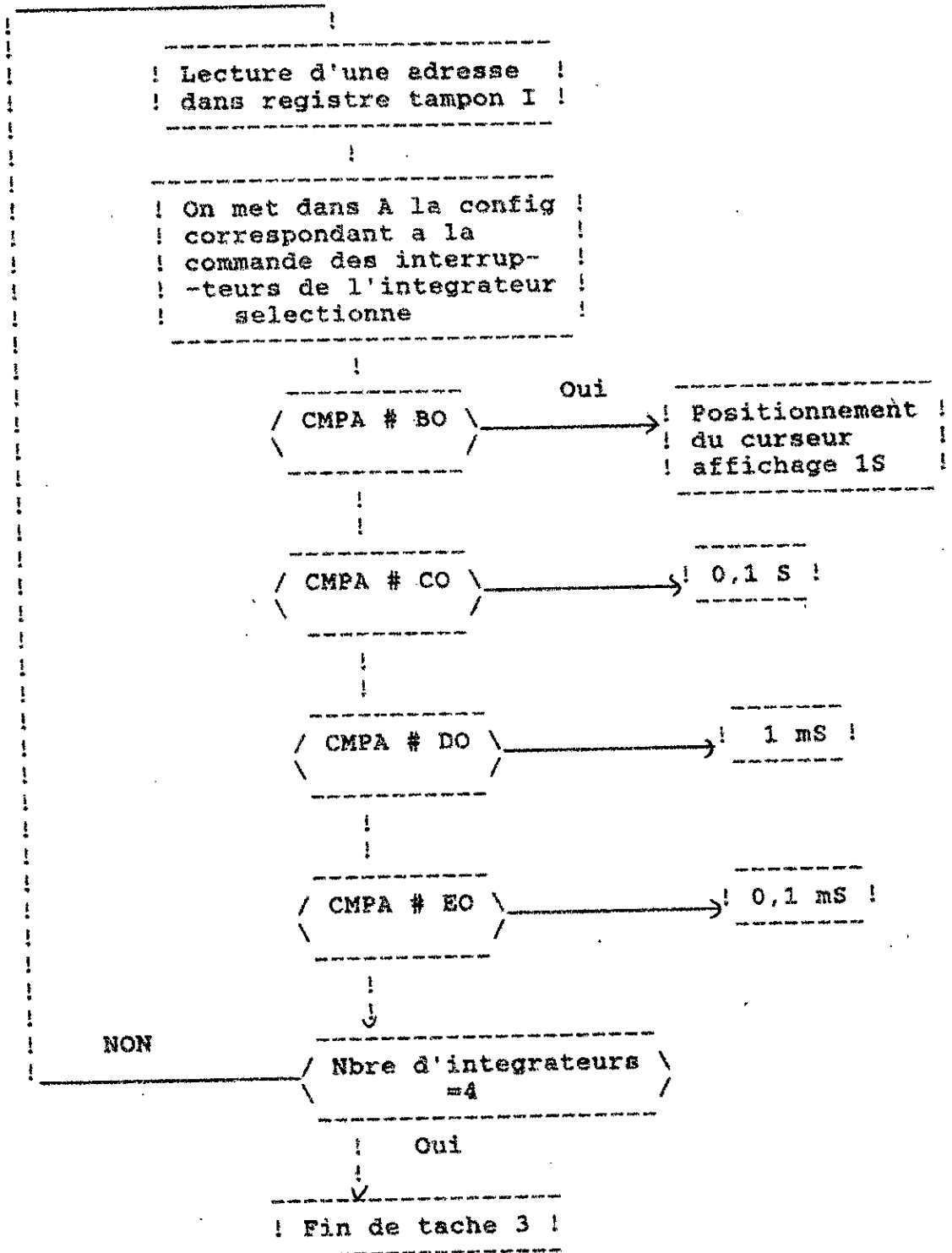


fig III.19

Lorsque l'utilisateur a introduit toutes les conditions initiales ainsi que les parametres et constantes du systeme d'equation a simuler , il peut lancer l'execution du calcul .

La tache USER permet apres affichage du s/menu II d'activer la tache calcul (tache 7 ) .

Celle ci permet le choix des cycles de calcul .

Elle permet en outre la commande des entrees logiques IC et OP des integrateurs .

Cette commande est effectuee par le chargement des registres des timers dans le cas des modes Repetitif et Iteratif .

Dans le cas du mode Normal , les entrees IC et OP sont commandees a partir des lignes PA0 et PA1 d'une interface parallele ou PIA .

L'organigramme de la tache 7 est donne fig III.20

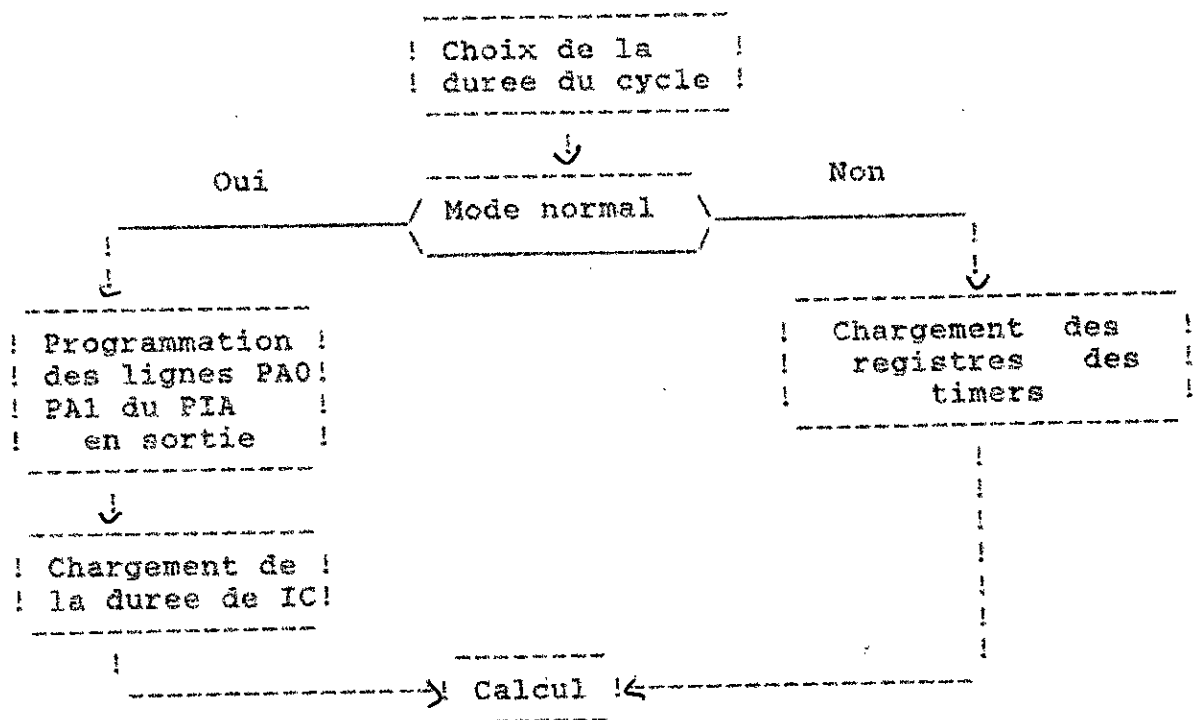


Fig III.20 Organigramme de la tache calcul



## CHAPITRE IV

---

### Introduction

Afin de tester les performances du calculateur realise nous avons retenu deux applications :

- 1 - Resolution d'une equation differentielle du second ordre.
- 2 - Resolution d'un systeme de quatre equations differentielles du premier ordre.

La procedure permettant la resolution de ces equations consiste a convertir les equations physiques en equations machines. Cette conversion necessite la definition de deux facteurs d'echelles :

- Facteur d'amplitude et facteur de temps.

Le choix du facteur d'echelle en amplitudes depend des valeurs limites de fonctionnement des amplificateurs et le facteur d'echelle temps est conditionne par le temps de reponse des enregistreurs. Compte tenu du fait que la tension de sortie maximale des amplificateurs est egale a 10 V, il s'ensuit que toute grandeur physique peut etre convertie en unite machine y telle que

$$\frac{y}{y_{\max}} = \frac{V}{10} \quad (1)$$

V etant la tension en volts correspondant a la variable physique y du probleme.

Le facteur d'échelle en temps  $\beta$  est tel que :

$$\tau = \beta t \quad (2)$$

$\tau$  = temps machine

$t$  = temps reel

Dans notre cas, nous avons choisi  $\beta = 1$

Les equations (1) et (2) peuvent etre utilisees pour realiser la conversion equations physiques ---> equations machines et inversement.

1) Resolution d'une equation differentielle du second ordre a coefficients constants sans second membre

L'equation que nous allons resoudre se presente sous forme generale suivante :

$$\frac{d^2 y}{dt^2} + 2 \zeta \omega_0 \frac{dy}{dt} + \omega_0^2 y = 0 \quad (1)$$

Cette equation regit un mouvement oscillatoire amorti.

$\omega_0$  etant la pulsation propre du systeme physique.

$\zeta$ , le coefficient d'amortissement des oscillations

On va etudier deux cas possibles en modifiant le facteur d'amortissement .

a ) L'equation etudiee est :

$$\frac{d^2 y}{dt^2} + 6 \frac{dy}{dt} + 36 y = 0 \quad (2) \quad \frac{dy}{dt}(0) = 0, y(0) = 150$$

Dans laquelle  $\omega_0 = 6 \text{ rd/s}$

$$\zeta = 0,5$$

- Mise a l'echelle en amplitude :

Les oscillations etant amorties, il en resulte que les amplitudes maximales  $y_{max}$ ,  $\frac{dy}{dt}$  max et  $\frac{d^2y}{dt^2}$  max

correspondent au temps  $t = 0$

Il s'ensuit que celles-ci peuvent etre obtenues en resolvant l'equation (2) en negligiant le terme d'amortissement. On obtient l'equation (3).

$\frac{d^2y}{dt^2} + 36y = 0$  (3) avec  $\frac{dy}{dt}(0)$  et  $y(0) = 150$

La solution de l'equation (3) est donne par :

$y(t) = 150 \cos 6t$  (3)

Celle-ci nous permet d'evaluer les valeurs maximales suivantes :

$y_{max} = 150$

$y'_{max} = 150 \times 6$

$y''_{max} = 150 \times 6 \times 6$

On choisit comme valeurs maximales  $y_{max} = 200$  ce qui donne  $y'_{max} = 200 \times 6$  et  $y''_{max} = 200 \times 6 \times 6$

On pose :

$\gamma = \frac{y}{y_{max}}$ ,  $\gamma' = \frac{y'}{y'_{max}}$ ,  $\gamma'' = \frac{y''}{y''_{max}}$

$\gamma$ ,  $\gamma'$ ,  $\gamma''$  sont les variables machines.

L'equation (2) s'ecrit alors

$\frac{y''}{y''_{max}} y''_{max} + 6 \frac{y'}{y'_{max}} y'_{max} + 36 \frac{y}{y_{max}} y_{max} = 0$

Ou encore en tenant compte de ce qui precede on obtient l'equation machine suivante :

$$\boxed{y'' + y' + y = 0} \quad (4)$$

l'identification de cette equation avec l'equation (1) donne  $w_0 = 1$  rd/s et  $\xi = 0,5$

On obtient donc une equation correspondant a des oscillations amorties de frequence propre  $w_0 = 1$  rd/s et dont l'amortissement  $\xi = 0,5$  [identique a celui de l'equation (2)].

Simulation de l'equation (4) sur le calculateur hybride

---

Le schema synoptique correspondant a cette equation est donne fig IV.1

Le coefficient du potentiometre P1 est  $K1 = -0,75$

La condition initiale  $y(0) = 150$  correspond a la tension de sortie du potentiometre.

Cette derniere <sup>est</sup> egale a  $10 \times 0,75 = 7,5$  V

Les resultats de la simulation sont donnees (fig IV.2)

Verification de la valeur de  $\xi$  a partir du graphe

---

Soit la relation

$$w'_0 = w_0 \sqrt{1 - \xi^2} \quad (d)$$

$$w_0 = \text{pulsation propre} = 1 \text{ rd/s}$$

$$w'_0 = \frac{2\pi}{T}$$

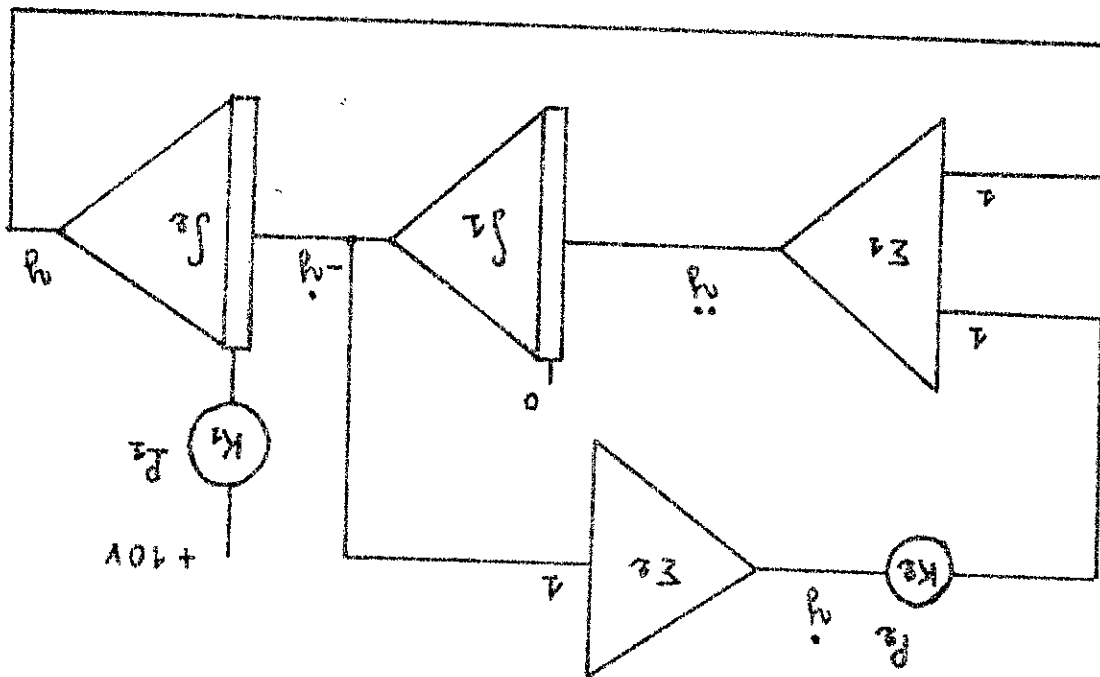
La mesure de T sur le graphe donne  $T = 7,25$  s

$$\text{donc } w'_0 = \frac{6,28}{7,25} = 0,866 \text{ rd/s}$$

- fig - IV-3 -

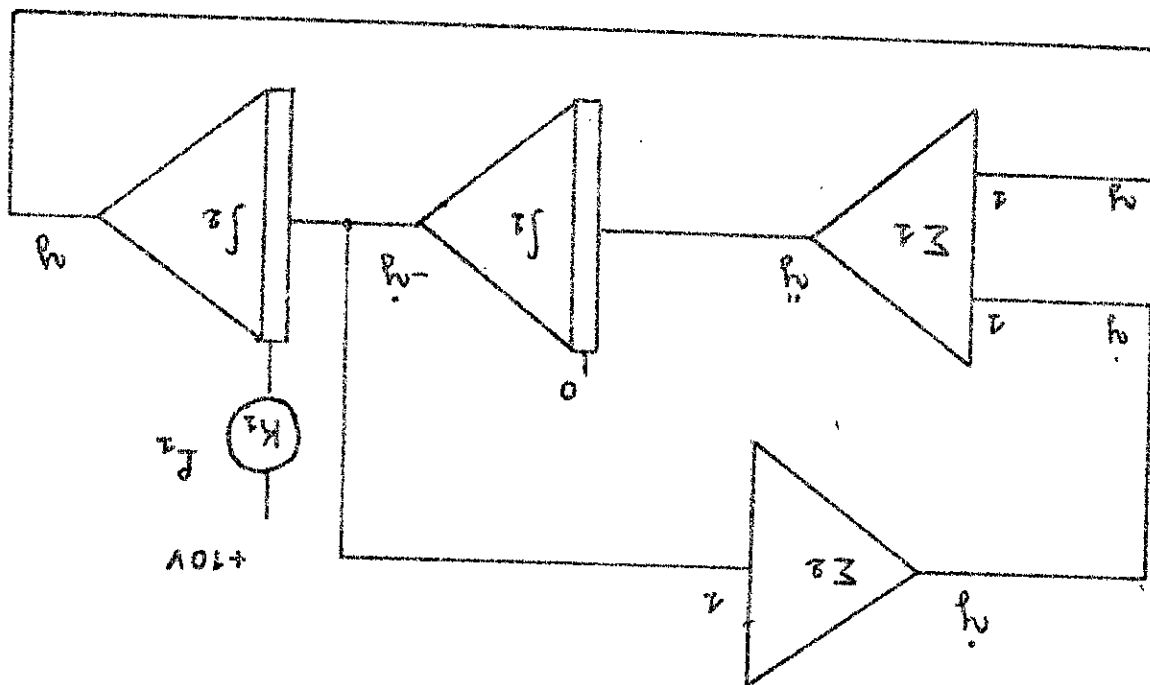
$K_2 = 0,8$

$K_1 = -0,75$

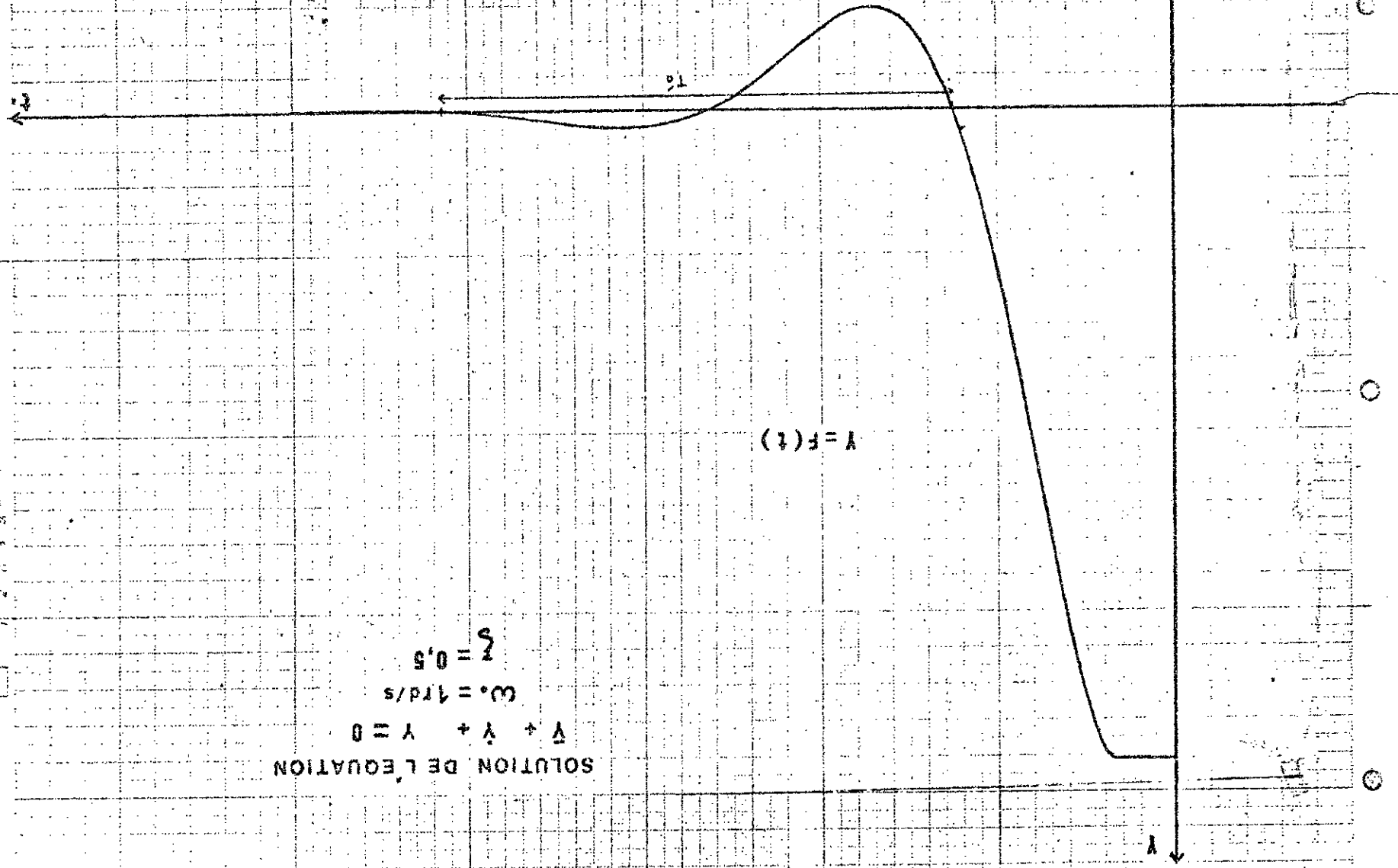


- fig - IV-1 -

$K_2 = -0,75$



*3-11-67*



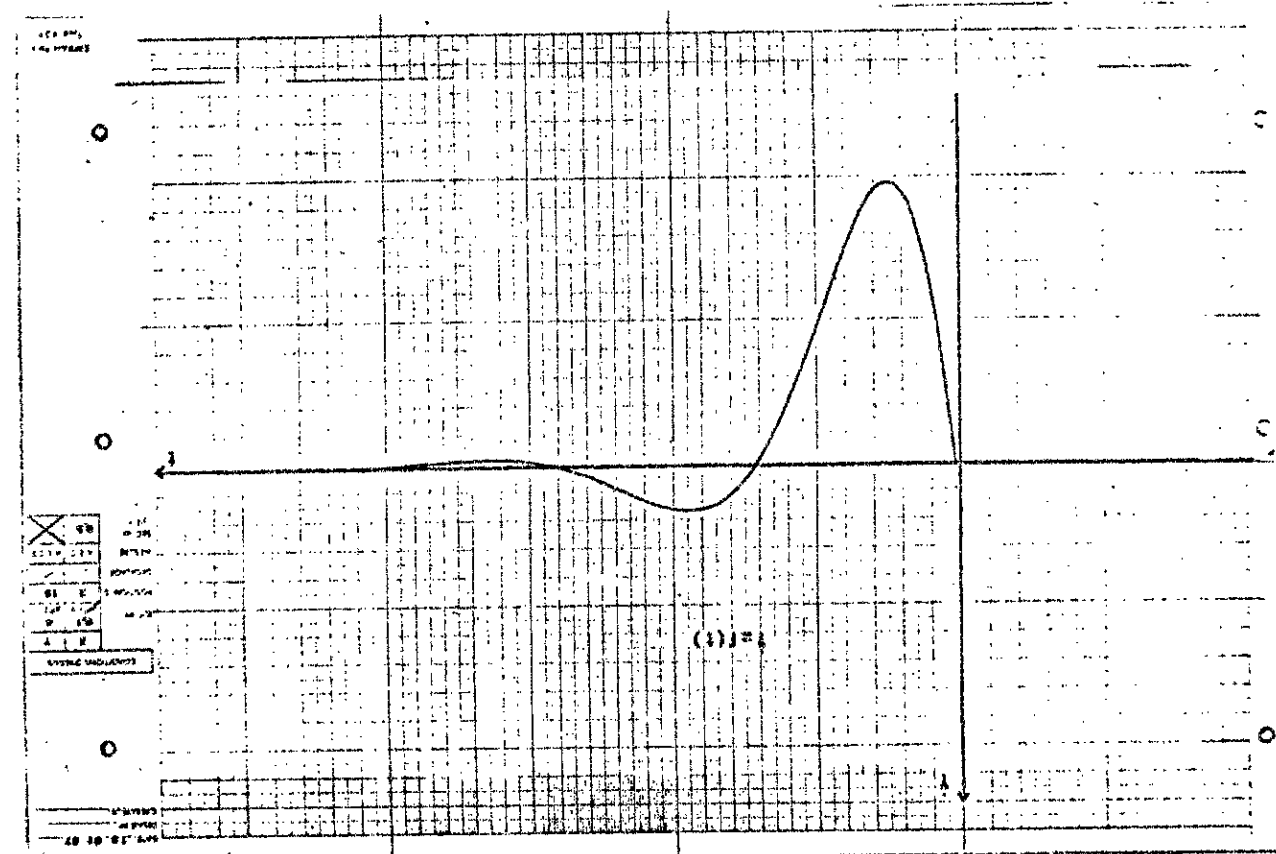
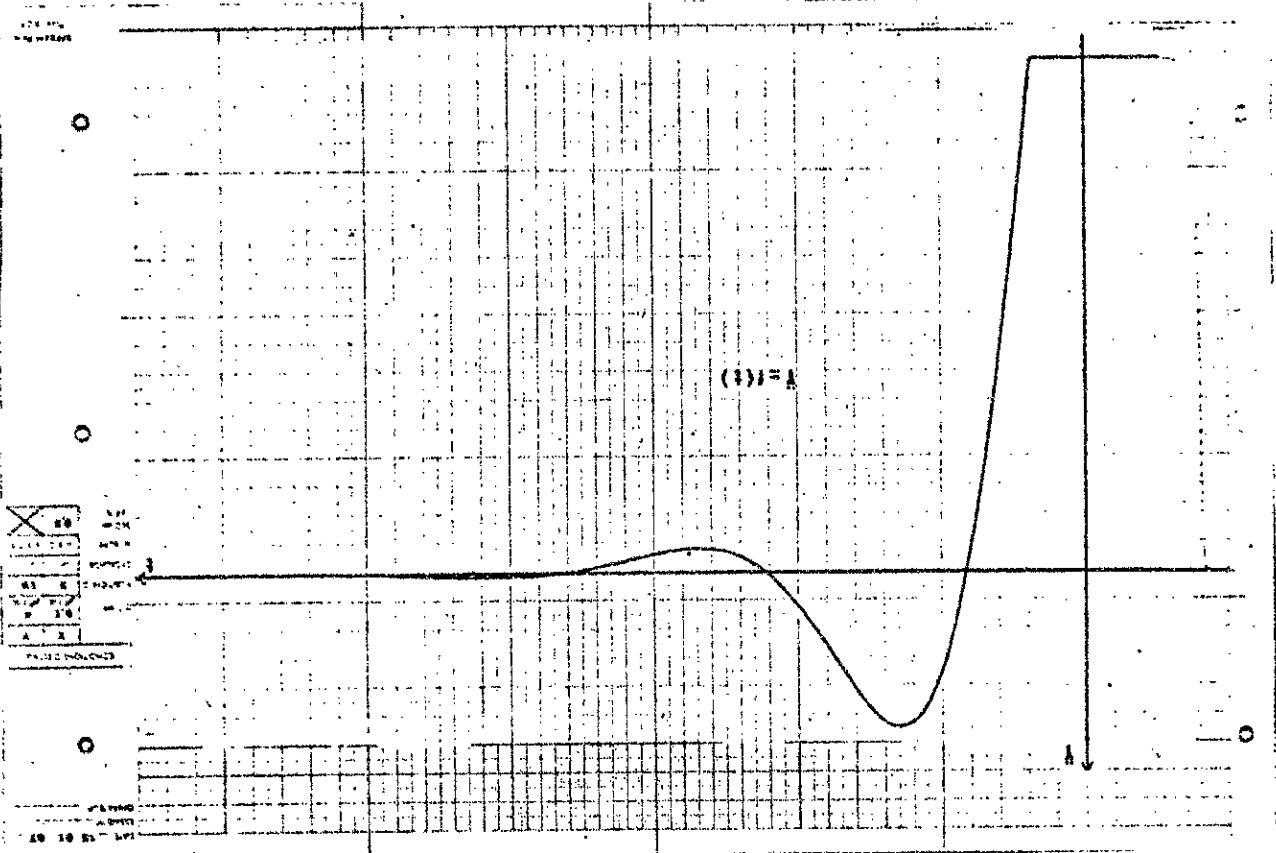
SOLUTION DE L'EQUATION  
 $y + y' + y = 0$   
 $\omega = 1 \text{ rad/s}$   
 $\xi = 0,5$

$y = f(t)$

0,5	SCALE
2	POSITION
0,1	AMPLITUDE
X	UNIT

- 95 0 -

fig. IV.2a



D'après (4) on peut écrire

$$\xi^2 = 1 - \frac{w_0'^2}{w_0^2}$$

$$\text{AN : } \xi^2 = 1 - 0,866^2 = 1 - 0,75 = 0,25$$

on obtient bien  $\xi = 0,5$

b) Soit l'équation

$$\frac{d^2 y}{dt^2} + 1,2 \frac{dy}{dt} + 36 y = 0 \quad \text{avec} \quad \frac{dy}{dt}(0) = 0 \quad \text{et} \quad y(0) = 150$$

pour laquelle  $\xi = 0,1$

$$w_0 = 6 \text{ rd/s}$$

Après transformation de l'équation (5) en équation machine on obtient

$$\boxed{y'' + 0,2 y' + y = 0} \quad (6)$$

Pour laquelle  $\xi = 0,1$

$$\text{et } w_0 = 1 \text{ rd/s}$$

Le schéma synoptique de l'équation (6) est donné (fig IV.3)

Les résultats de la simulation sont donnés (fig IV.4)

D'après le graphe, la pseudo période est  $T = 6\text{s}$  donc  $w_0 \approx 1 \text{ rd/s}$

Soit  $t_s$  l'instant pour lequel  $y = 5\%$  de  $y_{\max}$ . Pour des facteurs d'amortissement faibles, on a la relation suivante :

$$t_s = \frac{3}{\xi w_0} \quad \implies \quad \xi = \frac{3}{t_s w_0}$$



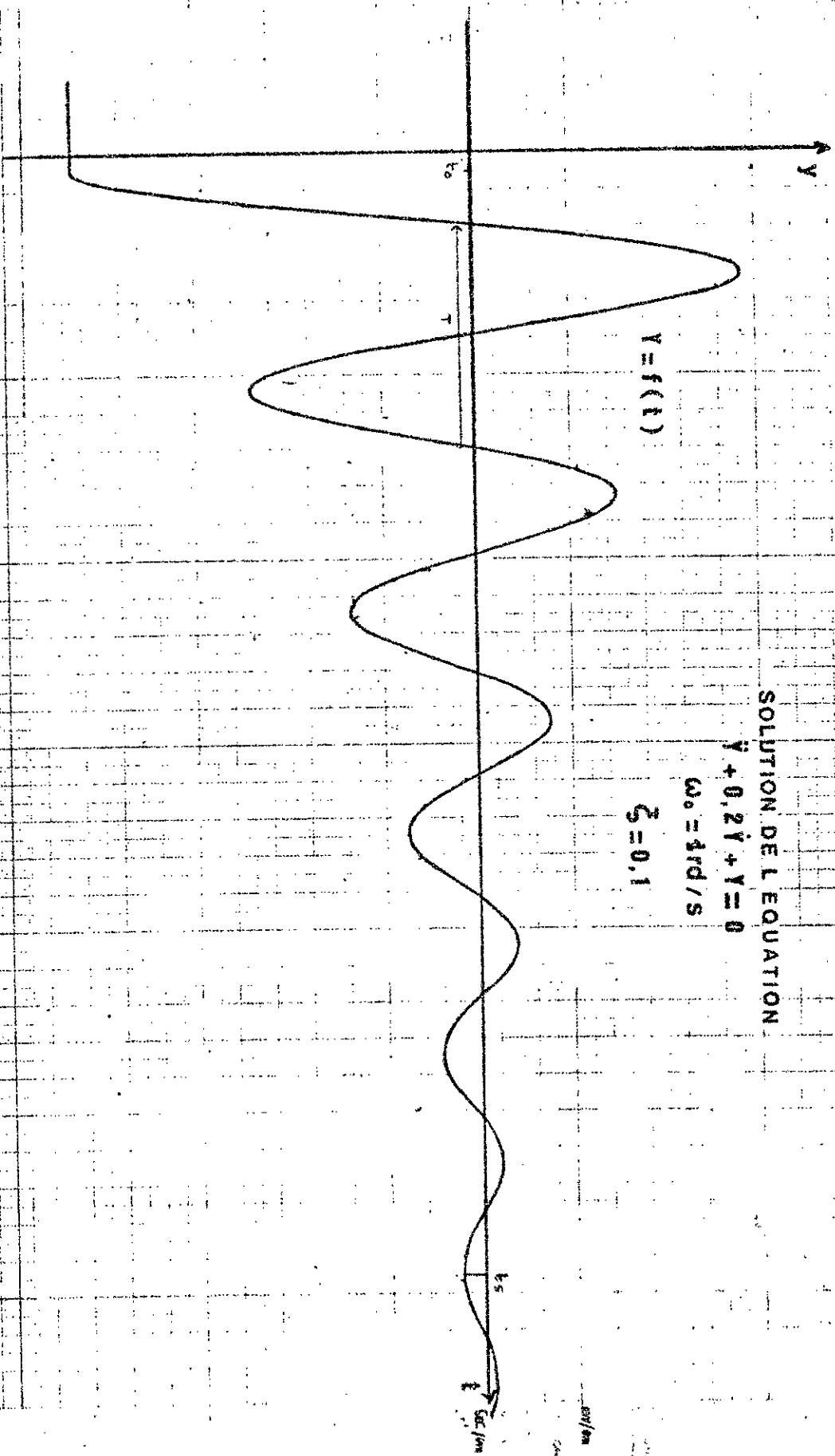
SOLUTION DE L'EQUATION

$\ddot{Y} + 0,2\dot{Y} + Y = 0$

$\omega_0 = 1 \text{rd/s}$

$\zeta = 0,1$

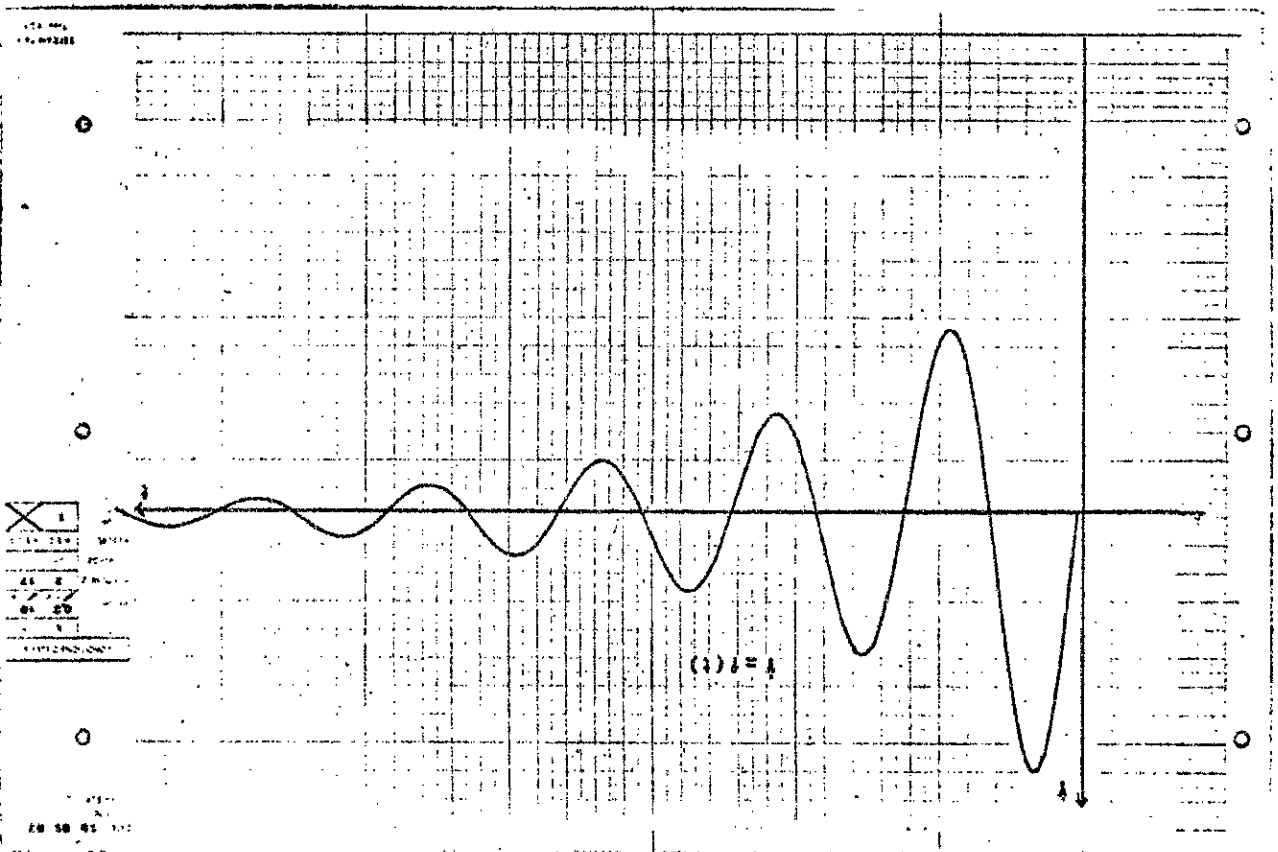
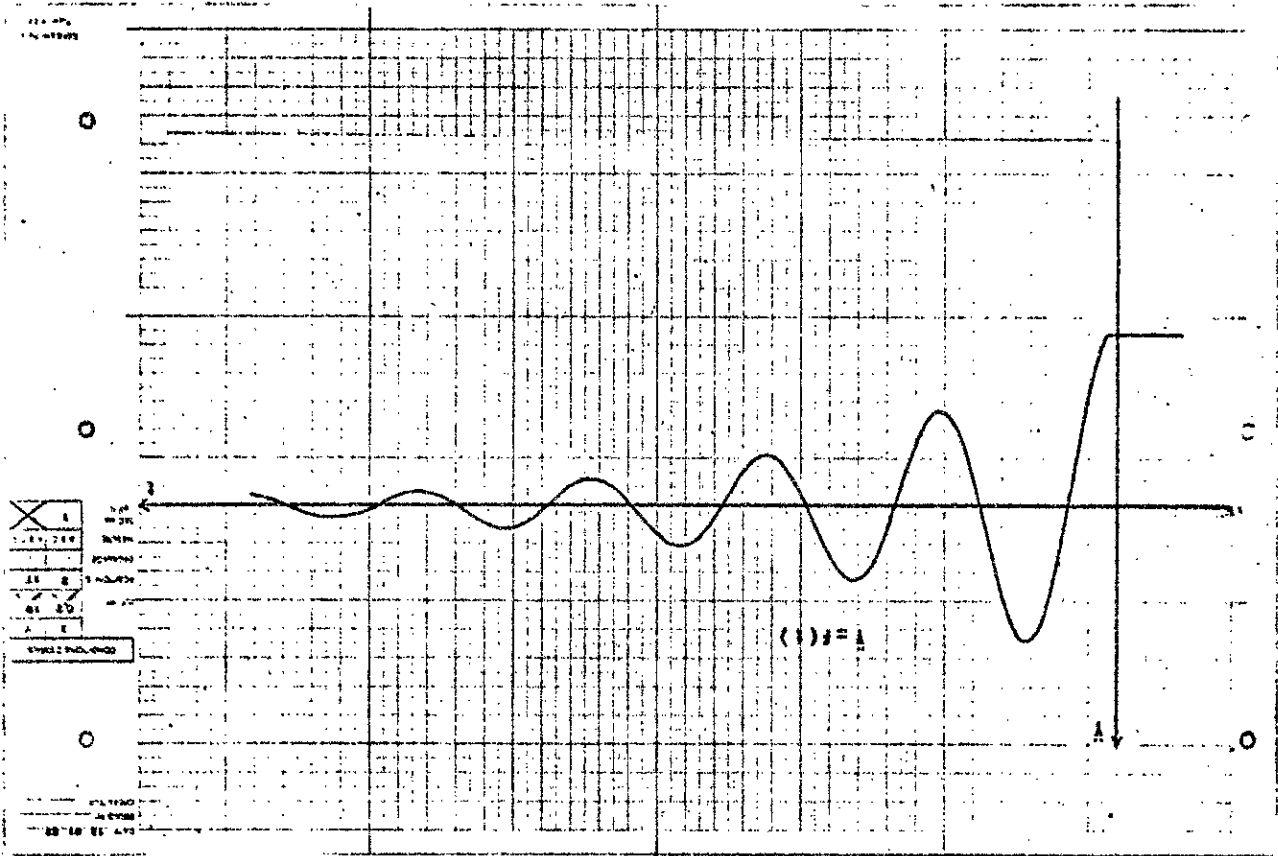
$Y = f(t)$



*Fig. IV.4*

SCALE	X	Y
mm/cm	0.2	10
mm/cm	2	57
mm/cm	1	

- fig. IV. 4a -



La mesure de  $t_s$  sur le graphe donne  $t_s = 30$  s donc on en déduit  $f$  :

$$f = \frac{3}{30 \times 1} = 0,1$$

2 ) Le deuxième exemple que nous avons retenu est le modèle décrivant le comportement dynamique d'un système formé de deux pendules identiques couplés par un ressort. Le modèle mathématique correspondant se compose de deux équations différentielles du second ordre :

$$I \quad \begin{cases} J \ddot{\theta}_1(t) + (ka + b) \theta_1(t) = ka \theta_2(t) \\ J \ddot{\theta}_2(t) + (ka + b) \theta_2(t) = ka \theta_1(t) \end{cases}$$

$\theta_1, \theta_2$  sont les positions angulaires des deux pendules

$$J = Ml$$

$M$  = masse d'un pendule

$l$  = longueur d'un pendule

$$b = Mgl$$

$a$  = distance du point de fixation du ressort à l'axe de rotation

$k$  = constante de raideur du ressort

La transformation canonique du système (I) donne :

$$x_1 = \theta_1, x_2 = \dot{\theta}_1, x_3 = \theta_2, x_4 = \dot{\theta}_2$$

d'où

$$II \quad \begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -C_2 x_1 + C_1 x_3 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = -C_2 x_3 + C_1 x_1 \end{cases} \quad \begin{aligned} C_1 &= \frac{ka}{J} \\ C_2 &= \frac{ka + b}{J} \end{aligned}$$

La résolution analytique de ce système d'équations pour différentes valeurs des conditions initiales peut nous

renseigner sur les valeurs maximales des variables.

La solution generale du systeme (I) est donnee par

$$\theta_1(t) = A \cos(\omega_1 t + \varphi) + B \cos(\omega_2 t + \psi)$$

$$\theta_2(t) = -A \cos(\omega_1 t + \varphi) + B \cos(\omega_2 t + \psi)$$

$$\omega_1 = \sqrt{\frac{b}{J}} \quad \text{et} \quad \omega_2 = \sqrt{\frac{b + 2ka}{J}}$$

Nous avons retenu les parametres suivants :

$$l = 0,65 \text{ m}, \quad a = 0,2 \text{ m}, \quad m = 0,47 \text{ Kg}, \quad k = 4\text{N/m}, \quad g = 9,8\text{m/s}^2$$

$$\text{d'ou : } \omega_1 = 4,085 \text{ rd/s}, \quad \omega_2 = 3,88 \text{ rd/s} \quad C_1 = 0,805 \text{ et}$$

$$C_2 = 15,88$$

1 ier cas Les deux pendules sont laches sans vitesse initiale avec la meme elongation.

$$\theta_1(0) = \theta_2(0) = -0,314 \text{ rd} \leftarrow$$

$$\dot{\theta}_1(0) = \dot{\theta}_2(0) = 0$$

la solution correspondante s'ecrit :

$$\theta_1(t) = \theta_0 \cos \omega_2 t$$

$$\theta_2(t) = \theta_0 \cos \omega_2 t$$

Les deux pendules ont donc des mouvements identiques.

Ils battent en phase.

Appelons  $X_1, X_2, X_3, X_4, X'_1, X'_2, X'_3, X'_4$  les variables machine.

La solution analytique permet la determination des valeurs maximales des variables du probleme.

$$X_1 \text{ max} = \theta_0$$

$$X_2 \text{ max} = X'_1 \text{ max} = \theta_0 \cdot w_2$$

$$X'_2 \text{ max} = X''_1 \text{ max} = \theta_0 \cdot w_2^2$$

$$X_3 \text{ max} = \theta_0$$

$$X_4 \text{ max} = X'_3 \text{ max} = \theta_0 \cdot w_2$$

$$X'_4 \text{ max} = \theta_0 \cdot w_2^2$$

Equations machine : le systeme (II) peut s'ecrire

$$\left\{ \begin{array}{l} X'_1 = X_2 \\ \frac{X'_2}{X'_2 \text{ max}} \cdot X'_2 \text{ max} = -C_2 \frac{X_1}{X_1 \text{ max}} + C_1 \frac{X_3}{X_3 \text{ max}} \\ X'_3 = X_4 \\ \frac{X'_4}{X'_4 \text{ max}} \cdot X'_4 \text{ max} = -C_2 \frac{X_3}{X_3 \text{ max}} + C_1 \frac{X_1}{X_1 \text{ max}} \end{array} \right.$$

Compte tenu des valeurs numeriques on obtient :

$$\left\{ \begin{array}{l} X'_1 = X_2 \\ X'_2 = -1,058 X_1 + 0,05036 X_3 \\ X'_3 = X_4 \\ X'_4 = 0,0536 X_1 - 1,058 X_3 \end{array} \right.$$

Ce systeme d'equations admet pour pulsations  $w'_1$  et  $w'_2$  telles que :  $w'_1 = \frac{w_1}{w_2}$  et  $w'_2 = \frac{w_2}{w_2} = 1 \text{ rd/s}$

La solution correspond donc a un mouvement oscillatoire de pulsation  $w'_2 = 1 \text{ rd/s}$  (pulsation reduite)

Le schema synoptique correspondant a ce systeme est donne (fig IV.5)

On obtient bien deux sinusoides en phase de meme periode

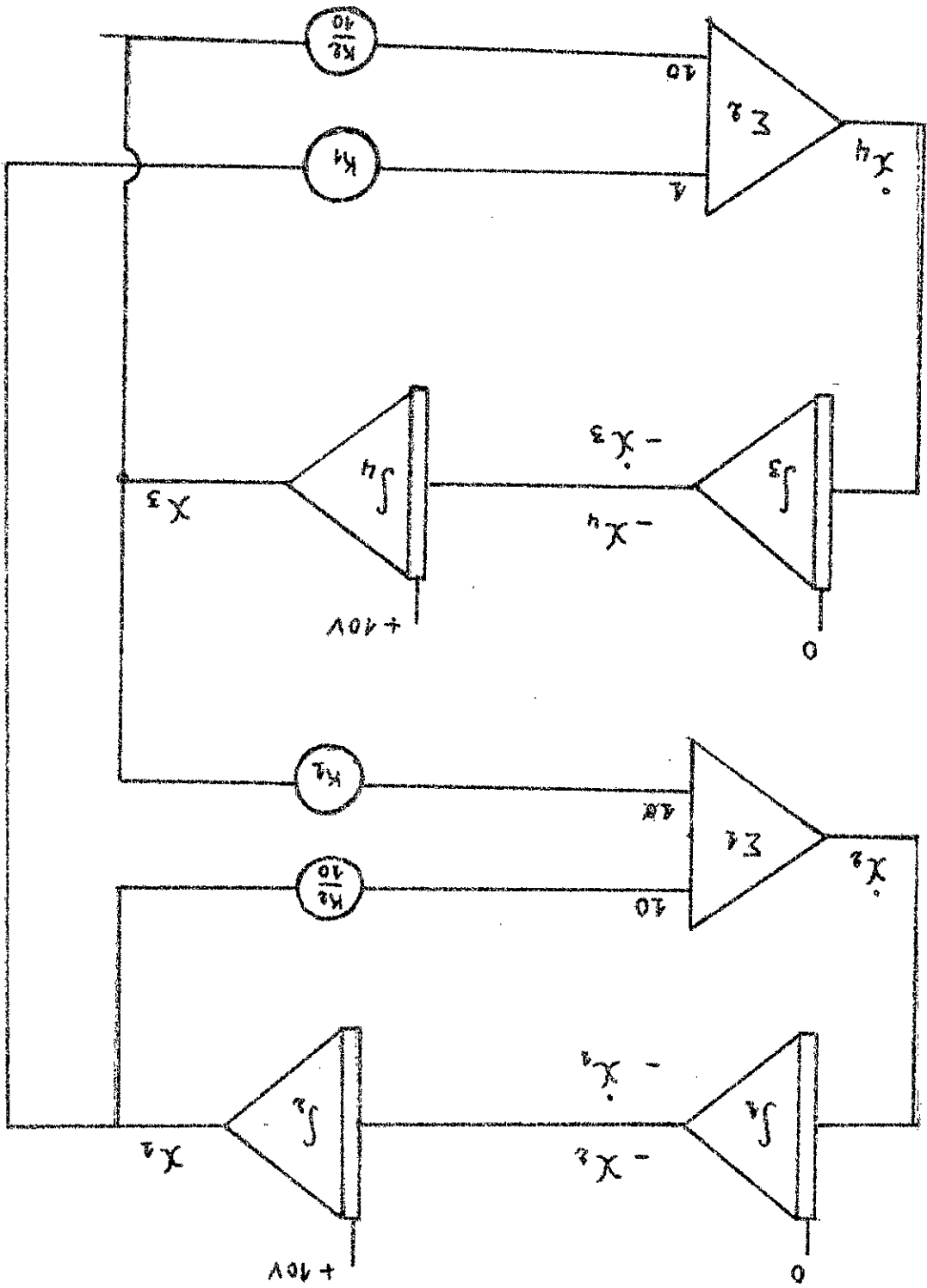
$$T = 6,2 \text{ s}$$

$$\text{donc } w = \frac{2 \cdot \pi}{T} \approx 1 \text{ rd/s} \quad (\text{fig IV.6})$$

- fig - 1V-5

$K_2 = -0,0536$

$K_2 = 1,058$



DATE \_\_\_\_\_  
 DESIGNED BY \_\_\_\_\_  
 OPERATOR \_\_\_\_\_

**SOLUTION DU SYSTEME**

$$\dot{X}_1 = X_2$$

$$X_1 = \theta_1$$

$$\dot{X}_2 = -C_2 X_1 + C_1 X_3$$

$$X_3 = \theta_2$$

$$\dot{X}_3 = X_4$$

$$\dot{X}_4 = C_1 X_1 - C_2 X_3$$

$$\theta_1(0) = \theta_2(0) = -0,314 \text{ rd.}$$

CONDITIONS DE RESEA :

	X	Y
Lo	Lo	Lo
Hi	Hi	Hi
DECAIAGE		
NECESSITE	A B C A B C	
SEC DE	0,15	X

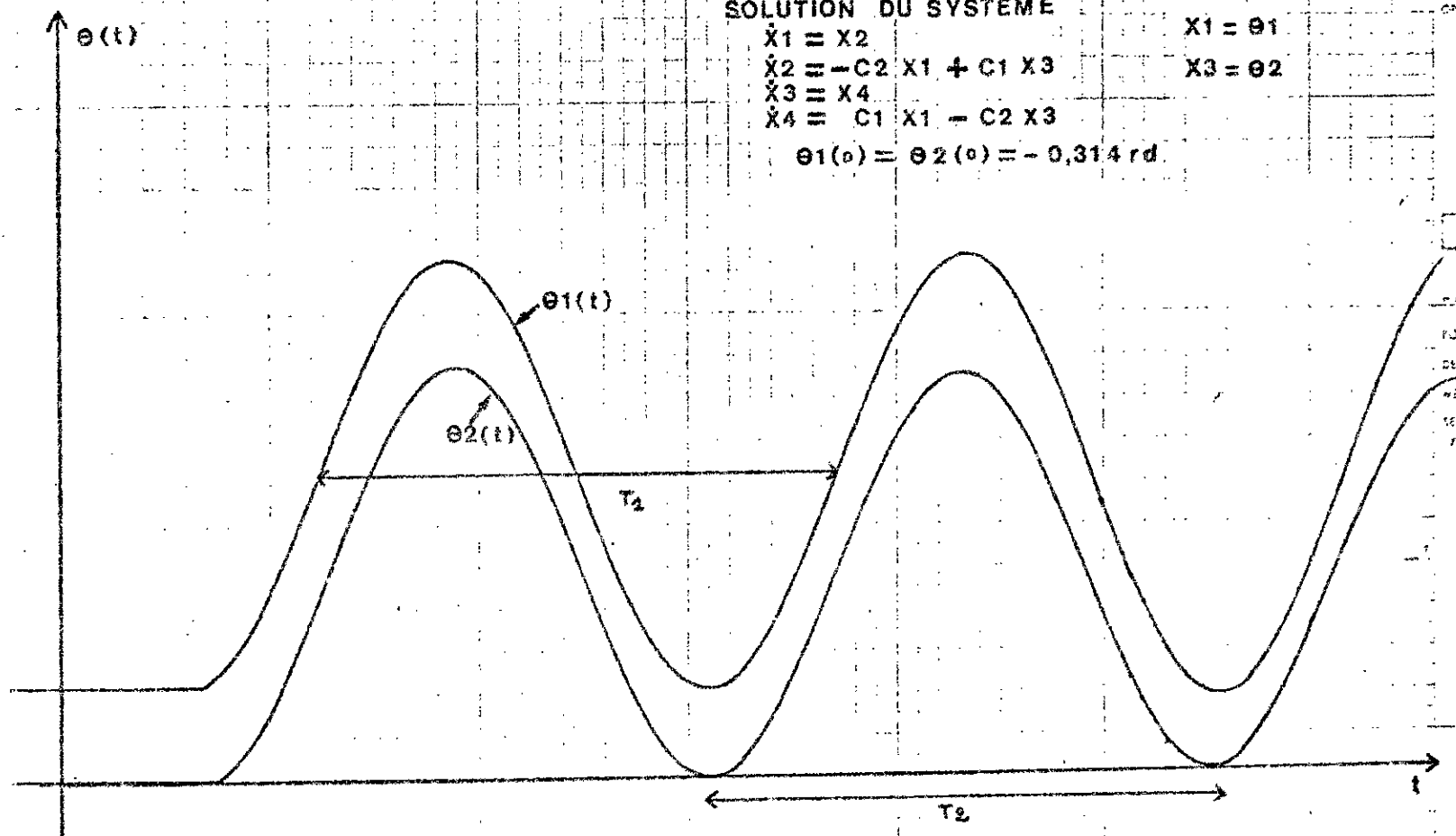


fig IV 6

2 ieme cas : les deux pendules sont laches sans vitesse  
initiale avec des elongations opposees

$$\theta_1(0) = -\theta_2(0) = 0,314 \text{ rd}$$

Solution analytique

$$\begin{cases} \theta_1(t) = \theta_0 \cos \omega t \\ \theta_2(t) = -\theta_0 \cos \omega t \end{cases}$$

Les deux pendules battent en opposition de phase.

De la meme facon que precedemment on determine les  
equations machines

$$\begin{cases} X'1 = X2 \\ X'2 = -0,9497 X1 + 0,0481 X3 \\ X'3 = X4 \\ X'4 = 0,0481 X1 - 0,9497 X3 \end{cases}$$

Le schema synoptique est donne (fig IV.7)

Les resultats sont donnes (fig IV.8)

Les resultats obtenus montrent bien que les deux pendules  
sont en opposition de phase.

La periode  $T = 6,25 \text{ s}$  correspond a  $\omega = 1,004 \text{ rd}$

Ceci correspond bien a la pulsation reduite  $\omega'1 = \frac{\omega_1}{\omega_2}$ .

$$\frac{\omega_1^2}{\omega_2^2} = \frac{15,07}{15} = 1,005 \implies \frac{\omega_1}{\omega_2} = 1,002$$

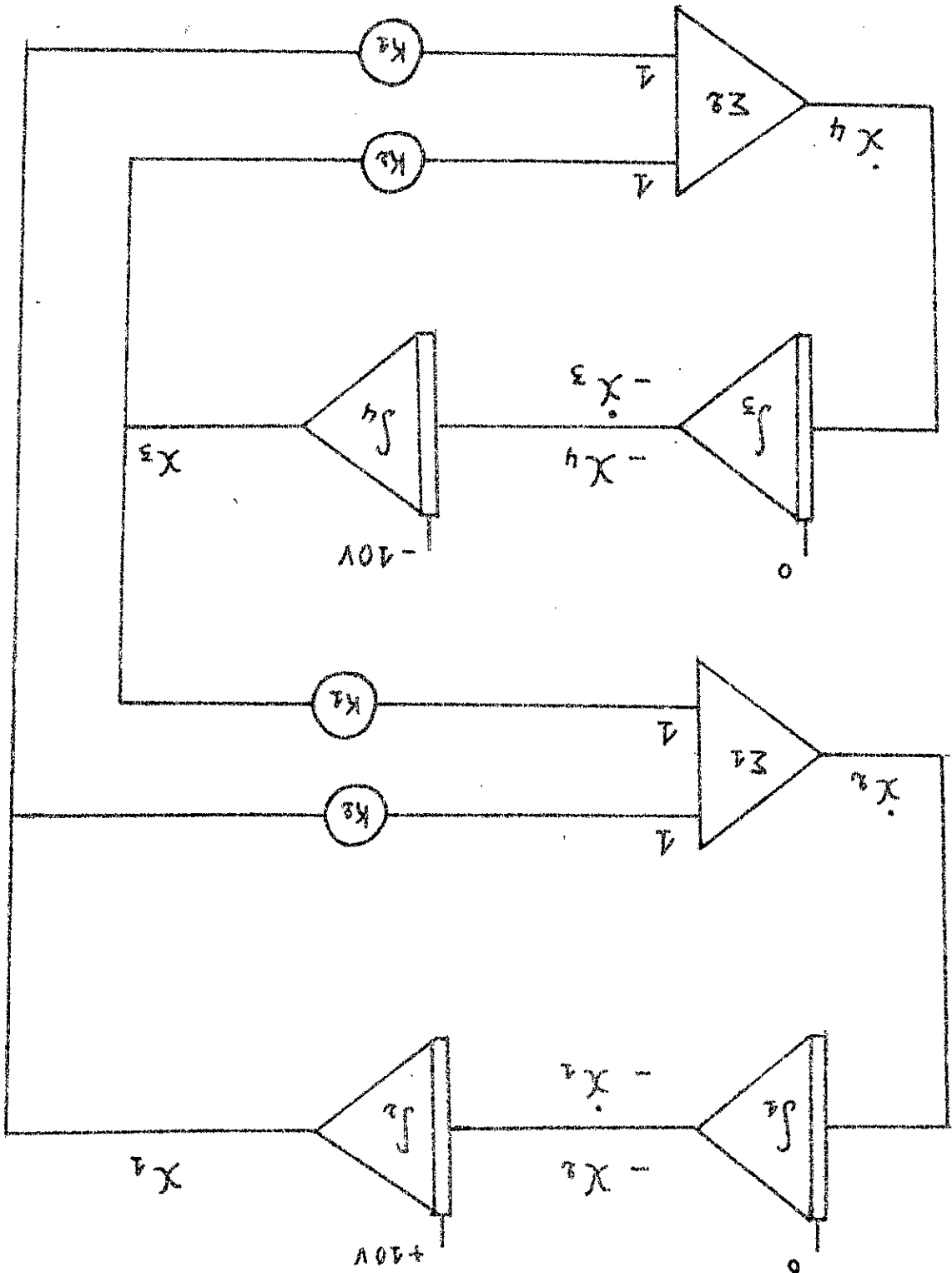
Conclusion

Les exemples traites montrent que le calculateur donne  
des resultats conformes a la theorie.

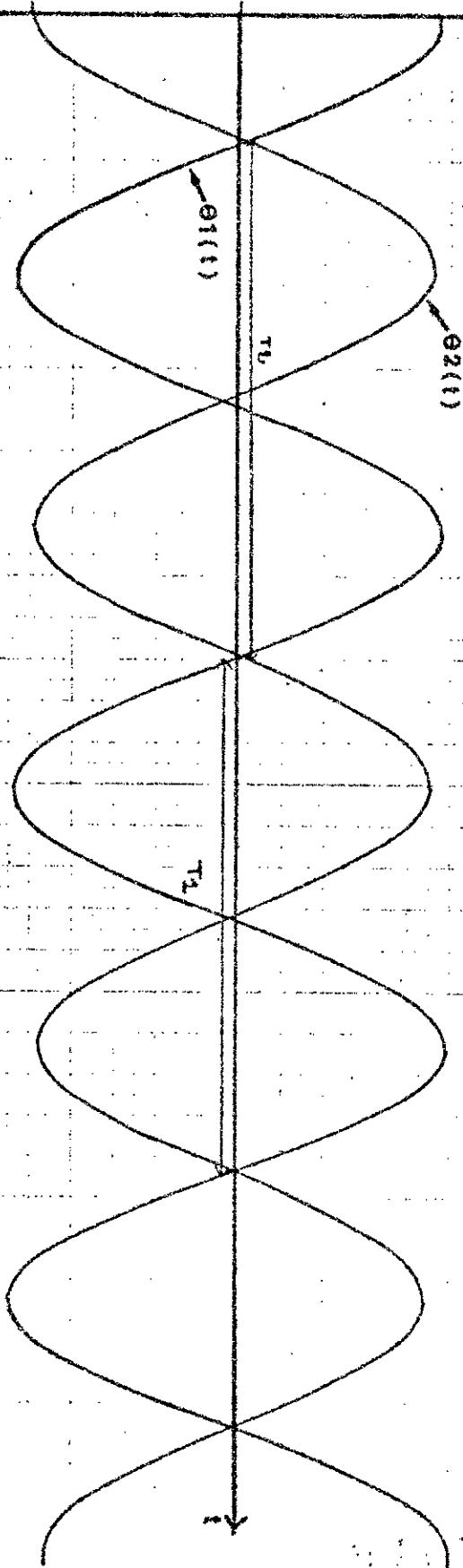


- fig - IV - 7 -

$K_2 = 0.9497$   
 $K_1 = -0.0482$



$\theta(t)$



$\theta_1(0) = \theta_2(0) = 0.314 \text{ rad}$

Fig. IV 8

DATE: \_\_\_\_\_  
 TIME: \_\_\_\_\_  
 95 55  
 0.15

Ces résultats nous permettent d'affirmer le bon fonctionnement du calculateur hybride. Cependant, ils ne font pas apparaître les imperfections du calculateur.

En effet, celles-ci se manifestent au niveau des opérateurs analogiques en particulier des intégrateurs, sommateurs, potentiomètres numériques ....

Les performances de ces derniers sont conditionnées par celles des amplificateurs utilisés (courant et tension de décalage, dérive, bande passante...) ainsi que par la qualité des circuits de calcul associés à ces amplificateurs (résistances, capacités...)

Les résultats d'une simulation peuvent être considérablement influencés par les erreurs dues aux amplificateurs opérationnels.

Une analyse de ces erreurs permet d'évaluer les performances du calculateur et d'en déduire les précautions à suivre sur le choix des composants.

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

La réalisation de ce travail a nécessite un certain nombre d'etapes :

- Mise au point des differents operateurs analogiques existants.
- Conception et realisation d'un microsysteme adapte a la structure initiale pour l'obtention d'un calculateur hybride. Ce microsysteme a ete bati autour d'un microprocesseur 8 bits (MC 6800 MOTOROLA)
- Ecriture des programmes de gestion et d'exploitation du calculateur hybride. L'exploitation du calculateur a necessite l'analyse de la structure des interfaces logiques existants en vue de l'adressage et du chargement des operateurs hybrides du calculateur.

Les differents problemes que nous avons rencontres au cours de cette etude ont ete lies essentiellement a l'interfacage entre le calculateur numerique et le calculateur analogique, ceci aussi bien du point de vue logiciel que materiel. Le logiciel de gestion que nous avons developpe, nous a permis par la meme occasion de nous familiariser avec certaines fonctions de base du noyau d'un systeme d'exploitation multitaches temps reel telles que :

- Ordonnancement des taches.
- Primitives P et V de synchronisation entre taches.

Les fonctions relatives a l'exploitation du calculateur hybride dans le cas de la resolution de systemes

d'equations differentielles ordinaires ont ete developpees.

Les tests de mise en oeuvre de ces fonctions ont ete realises a l'aide de deux exemples :

- Une equation differentielle du second ordre a coefficients constants.

- Un systeme de quatre equations differentielles ordinaires du premier ordre.

Les resultats obtenus permettent d'affirmer que le calculateur hybride est, dans sa version actuelle, adapte a la resolution de systemes d'equations differentielles.

L'ordre du systeme d'equations differentielles que l'on peut resoudre est limite par le nombre d'integrateurs que l'interface logique est capable de commander.

Une extension est neanmoins possible.

Il est a noter que notre systeme n'offre pas les possibilites optimales d'un calculateur hybride. En effet, le microsysteme etant realise a base d'un microprocesseur 8 bits est destine a la commande et a la gestion du calculateur hybride. Il n'est pas bien adapte au traitement numerique. Il est cependant, possible d'agir a deux niveaux pour obtenir une amelioration de notre systeme :

- Au niveau technologique : il est plus interessant d'utiliser un microprocesseur 16 bits qui est plus adapte au traitement numerique d'autant plus que les constructeurs ont prevu des coprocesseurs arithmetiques

tres puissants associes a cette famille de microprocesseurs.

- Au niveau logiciel : a partir de la definition des fonctions du noyau du systeme d'exploitation, il est possible de reecrire les programmes correspondants s'adaptant a une structure a microprocesseur 16 bits. Ceci est d'autant plus interessant que les microprocesseurs 16 bits sont dotes d'un jeu d'instructions plus puissant que celui des microprocesseurs 8 bits.

Il serait aussi, interessant d'utiliser un langage evolue tel que le langage C.

L'adjonction de nouvelles primitives dans le noyau peut etre envisagee.

Avec les modifications que nous venons de citer, le calculateur numerique pourrait, moyennant le developpement de taches de calcul numerique, prendre part au calcul au cours de la simulation.

Les modifications que l'on pourrait donc envisager relevent beaucoup plus de la partie numerique du calculateur hybride.

B I B L I O G R A P H I E

- (1) G de Grandi D Flash Distributed operating system for microprocessors. Institut of Atomic energy - Pecking
- (2) Analog / Hybrid Computation and Digital Simulation  
G.A.Korn - and R.VICNEVETSKY - IEEE Transactions on computers - vol C25 NO 12 Dec 1976.
- (3) Digital Continuous System Simulation  
G.Korn - J.Wait  
Prentice - Hall ENGLEWOOD CLIFFS, New Jersey 1978
- (4) Interactive Simulation With direct - executing language systems.  
G.A.Korn - Simulation Juillet 1981
- (5) Calculateurs analogiques et hybrides  
H.Cheraton  
Techniques de l'ingenieur - H1 - informatique (1984)
- (6) Structure et fonctionnement des ordinateurs  
J.P.Meinardier - Ed Larousse - Serie informatique
- (7) Les superordinateurs  
Levine Ronald - revue pour la science - Mars 1982
- (8) Hybrid Computation  
Thomas.D.Trutt IEEE June 1964
- (9) Hybrid methods for partial differential equations  
R.VICHNEVETSKY - Simulation - Avril 1971

- (10) Electronic Analog and hybrid Computers  
G.A.Korn, T.Korn Mc Graw Hill 1964.
- (11) Simulation hybride  
A.Bonnemay - Ecole Superieure d'electricite -  
NO 2274 1973
- (12) Cours de calcul analogique CEA.  
Departement d'electronique generale (Novembre 1967)
- (13) Cours de calcul hybride CEA  
Departement d'electronique generale (Fevrier 1969)
- (14) Calcul analogique et hybride  
J.HEILEN - cours Supelec NO 2155 Paris 1969
- (15) Comparison of analog and digital simulation methods  
for a vehicule dynamics problem  
S.Bloor , A de Pennington, RS Sharp, JS Swift  
revue simulation - Septembre 1975
- (16) Real time analog digital simulation -  
M.Connely. IRE Transaction on electronic computers  
Fevrier 1962.
- (17) Analog computers  
M.Brand and T.Brand - Contemporay mathematics 1970
- (18) System modeling and control  
J SCHARZENBACH and K.F.Gill Ed Arnold 1984



- (19) Analog simulation of a hybrid gasoline - Electric vehicle  
D.B gilmore . Departement of Mechanical Engineering  
Simulation 1982.
- (20) Conception et realisation d'un calculateur hybride a  
microprocesseur.  
these de 3 ieme cycle - Electronique - U.S.T.H.B Alger  
R.OUIGUINI - juin 1981
- (21) Calcul analogique et calcul hybride  
W.Steiberg - NO 2074 cours E.S.E - universite de Paris  
- 1967
- (22) Elements de calcul analogique - serie automatique  
Moreau - Gobbey - ENET - 1973
- (23) Hybrid computation - Bekey Ga, Karplus,  
J Wiley - 1968
- (24) Analog and Analog / Hybrid computer programming  
HAUSNER, A Prentice Hall, Englewood Cliffs  
NJ, 1971
- (25) A Hybrid method for the solution of the first order partial  
differential equations arising in non linear estimation  
M.I.El Zorhany, Balq Suhamanian  
Trans I.N.A.C.S vol XVIII NO 4 october 1976
- (26) Simulation of non linear differential equation systems  
- M R Belmont, W. Andrew, L.Stewart,  
Simulation Juin 1984

- (27) dynamics simulation of the Slopopoke 3 nuclear heating reactor  
C.Min Tseng, R M Lepp - Simulation Avril 1985
- (28) Modified basic for all digital hybrid computing  
M J Morse, B R Gilbert - Simulation Septembre 1977
- (29) Modeling processing systems on analog / hybrid computers  
J L Guy - revue chemical Engineering - Rec 1981
- (30) Hybrid techniques applied to optimisation problems  
. Hans S Witsenhausen - May 1962 - CAL vol 21 IEEE  
Spectrum June 1964
- (31) Corrected inputs - A method for improved hybrid simulation  
R. Gelman - Nov 1963. Vol 24 IEEE Spectrum June 1964
- (32) Calculateur hybride en simulation de systemes continus complexes par methodes de decomposition et coordination par relaxation.  
L.RAFFORT, B.Lang  
- mathematics and computers in simulation XXVI 1984
- (33) Real time microcomputer simulation for space shuttle centaur avionics.  
G P Szatkowski, HC Nelandes - simulation novembre 1982
- (34) Using a multiprocessing hybrid computer for flight simulation  
J P Landauer - Simulation Juillet 1982

- (35) Early Desire : a floating point equation language simulation system for minicomputers and microcomputers  
G.Korn simulation Mai 1982.
- (36) Interactive simulation on a microcomputer  
D Ellison, I Herschdofer, J Tunnicliffe  
Simulation Mai 1982
- (37) La simulation des systemes  
C.V. Feuvrier  
Maitrise d'informatique - Dunod Universite 1971
- (38) The microcomputer and the analog computer  
A perfect hybrid computing couple  
R H Spiess - Simulation Octobre 1980

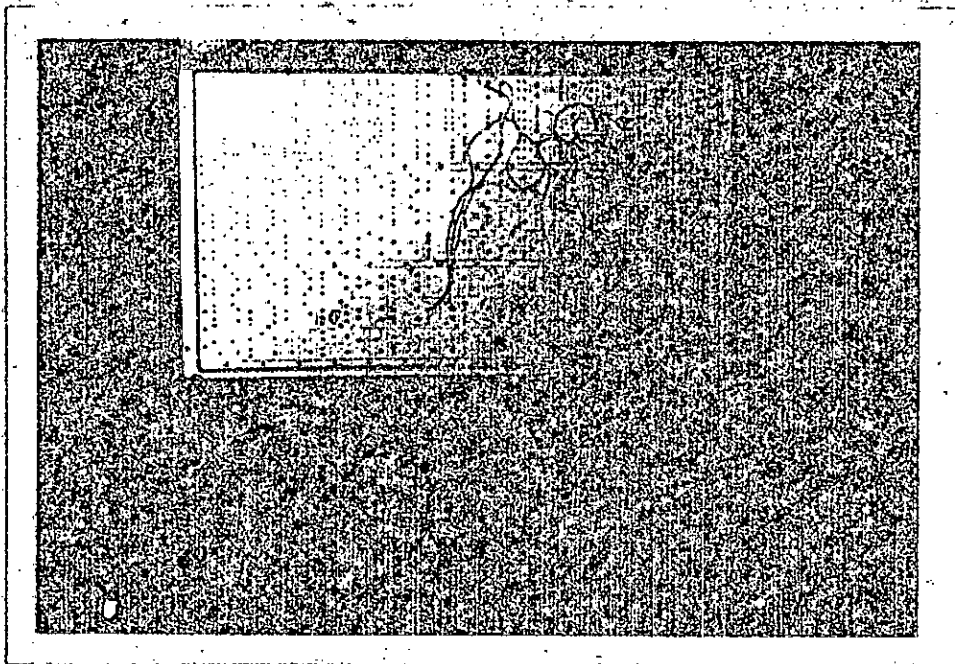


FIG A1 : Le panneau de câblage du calculateur hybride

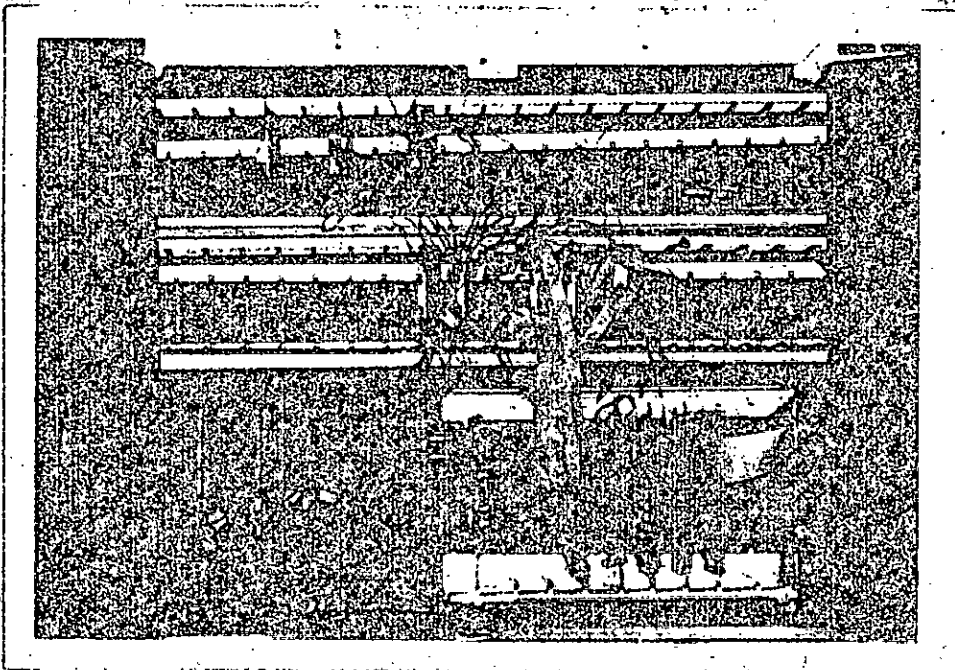


FIG A2 : Le Back du calculateur hybride

Fig 4 - Interface analogique

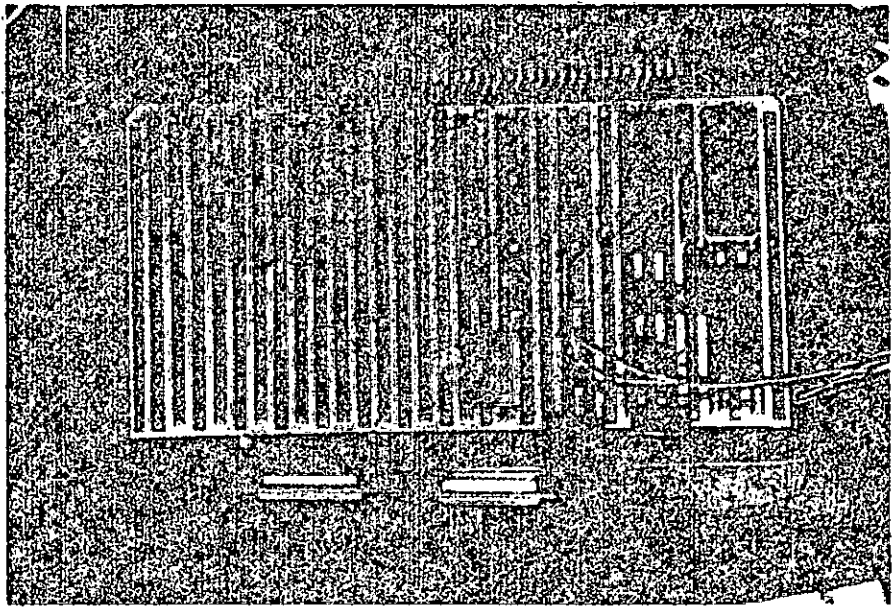
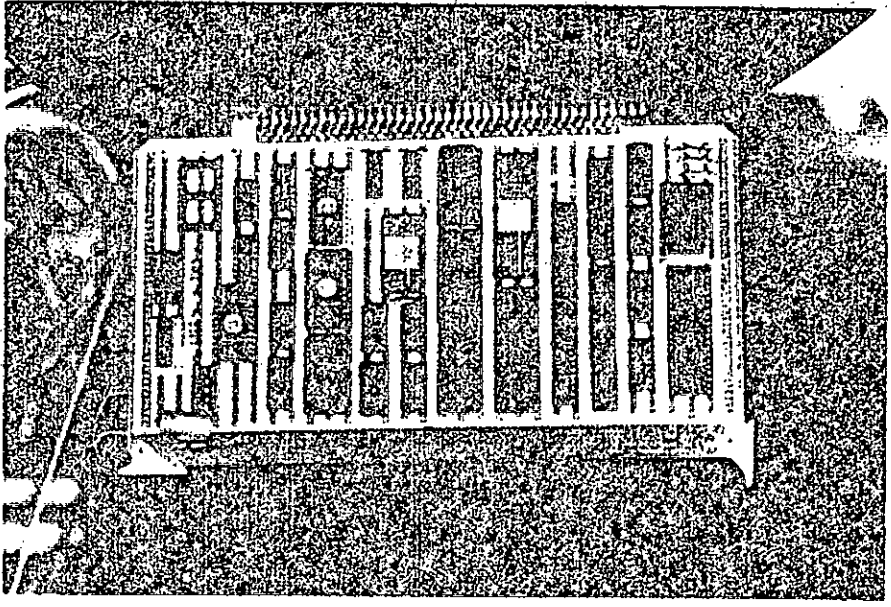
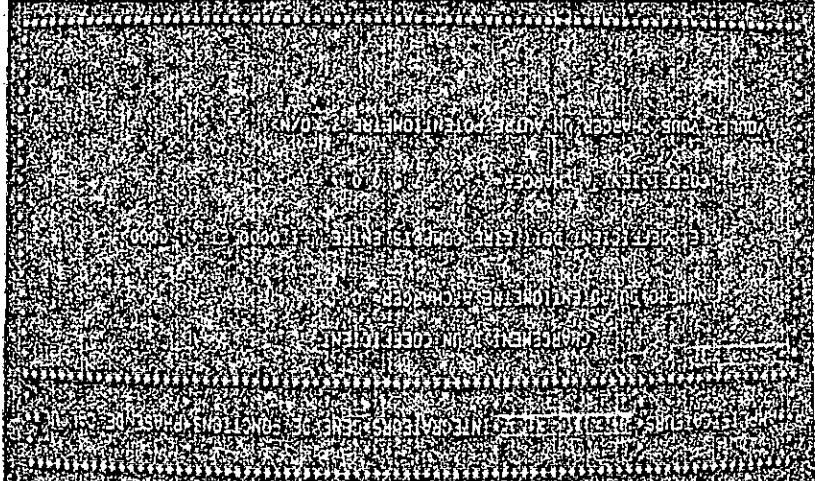
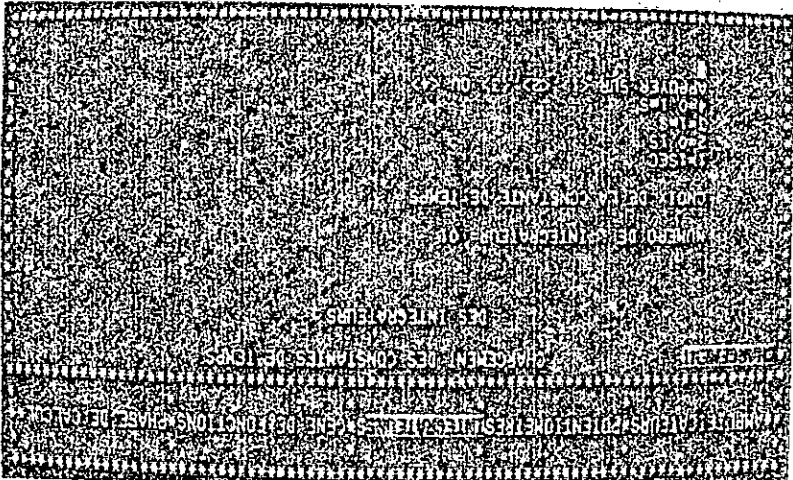
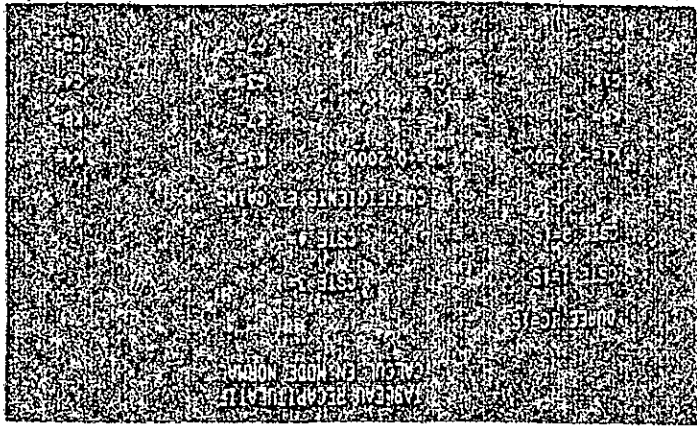


Fig 3 - la carte PROCESSEUR





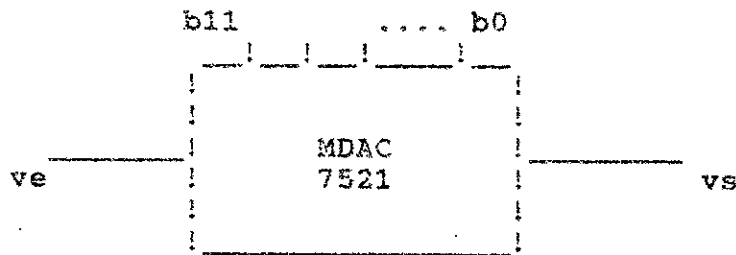
EXEMPLES DE CHARGEMENT EN DE LECTURE DE PARAMETRES

ANNEXE I

-----  
 Description des PTN et des APR  
 -----

1 ) Potentiometres numeriques  
 -----

N



N est l'entree binaire du PTN

Ve est la tension d'entree analogique du PTN

Dans le cas des PTN , Ve est une tension de reference  
 egale a 10 V

Si l'on note

N valeur binaire programme

n le nombre de bits

Ve tension d'entree analogique

Vs tension de sortie

En posant

$$D = \frac{N}{2^n} \quad 0 < D < 1$$

La relation liant Ve et Vs sera :

$$Vs = ( 2D - 1 ) Ve$$

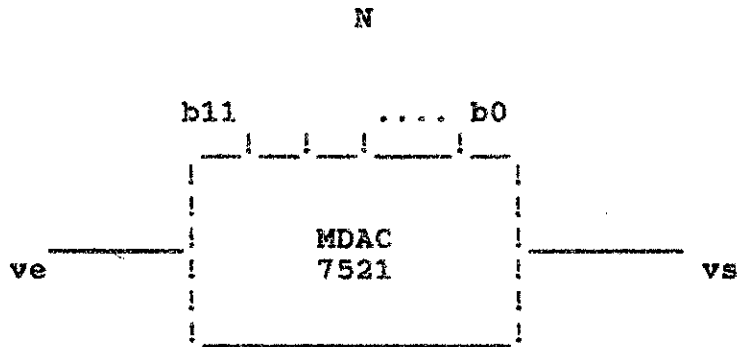
Que l'on peut mettre sous la forme suivante

$$Vs = - K Ve \quad \text{avec} \quad -1 < K < 1$$

Il est important de noter que lorsque K est fixe , Vs est une tension constante . De ce fait nous pouvons dire que la sortie d'un PTN represente un parametre du modele.

Remarque : les GDF sont concus de la meme facon .

2 ) Amplificateurs programmables :



Dans ce cas , Ve est la tension de sortie d'un operateur analogique du calculateur hybride .

Cette tension de sortie est appliquee a l'entree analogique de l'APR .

La relation liant Ve et Vs dans le cas d'un APR est :

$$Vs = - \frac{1}{D} Ve$$

ou  $Vs = - A Ve$  avec  $1 < A < 100$

Ve est donc une tension variable . Il en est de meme pour Vs . Vs represente une variable du modele .



ANNEXE II Adresses des operateurs a commande numerique

---

1 ) La carte d'interface logique permettant l'adressage et le chargement des operateurs a commande numerique (APR , PTN , GDF) contient deux PIA .

Le PIA I est destine a la selection d'un des operateurs a commande numerique (voir tableau).

Potentiometres	PORT B	PORT A
Potentiometre 01 :	02	00
02	02	01
03	//	02
04	//	03
05	//	04
06	//	05
07	//	06
08	//	07
Amplificateurs		
Amplificateur 01	//	08
02	//	09
03	//	0A
04	//	0B
05	//	0C
06	//	0D
07	//	0E
08	//	0F

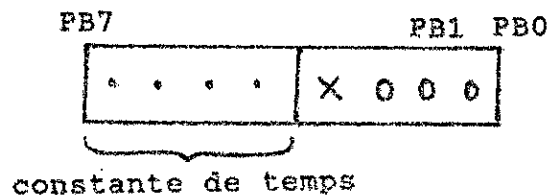
2 ) La carte d'interface logique d'adressage des integrateurs ne contient qu'un seul PIA d'adresse A730 .

Le port A de ce PIA permet la selection d'un integrateur le port B permet de fixer la valeur de la constante de tps de l'integrateur selectionne .

Les configurations du port A en sortie pour la selection d'un integrateur sont les suivants :

Port A : 10 - 11 - 12 - 13 - 14 - 15 - 16 - 17  
-----  
          20 - 21 - 22 - 23 - 24 - 25 - 26 - 27  
          30 - 31 - 32 - 33 - 34 - 35 - 36 - 37

Les constantes de temps sont fixees par une configuration contenue dans le port B de la facon suivante



Les commutateurs utilises pour le choix de la constante de temps sont en logique negative ..

On obtient les configurations suivantes :

PB7	PB6	PB5	PB4	
1	1	1	0	-----> 1S
1	1	0	1	-----> 0,1S
1	0	1	1	-----> 1 mS
0	1	1	1	-----> 0,1 mS

## A N N E X E III

---

### Specifications techniques du calculateur

#### Tension de reference

---

Tension de sortie : + 10 V

Courant de sortie : 600 m A

Resolution : 10<sup>-4</sup> (unite machine)

#### Operateurs Analogiques

---

Tension de sortie : + 11 V

Courant de sortie : 20 m A

#### Sommateur

---

3 entrees : gain 1

2 entrees : gain 10

Precision : 0,5 %

Bande passante : 400 Khz

#### Integrateur

---

1 entree

3 modes : (IC, OP, HL)

4 Constantes de temps : 1 s; 0,1 s; 1 ms; 0,1 ms

Temps de commutation des switches

Derive : 100 V / sec

Precision : 0,9 %

#### Generateur non lineaires

---

3 types d'operateurs : multiplieur, diviseur,  
racine carree.

Tension de sortie : + 10 V

Detection de saturation :  $(X + Y) > 20 \text{ V}$

Erreur statique : 0,1 % de la pleine echelle

$(X + Y) < 200$

Bande passante : 300 Khz

Memoire analogique :

-----  
Temps de commutation : 1,2 s

Derive : 500 V / sec

Erreur a 1 Khz : 1 %

Comparateur :

-----  
Nombre d'entree : 2

Tension d'entree : + 10 V

Tension de sortie : + 5 V (TTL)

Sensibilite : 10 mV

Temps de commutation : 1 s

Commutateur analogique :

-----  
Tension d'entree : + 10 V

Erreur statique : 0,1 %

Temps de commutation : 1 s

Generateur de fonction :

-----  
sin / cos

Tension d'entree : + 10 V

Erreur statique :  $5 \text{ V} < V_e < 10 \text{ V}$  15 mV

$V_e = 0$  8 mV

bande passante : 50 Khz

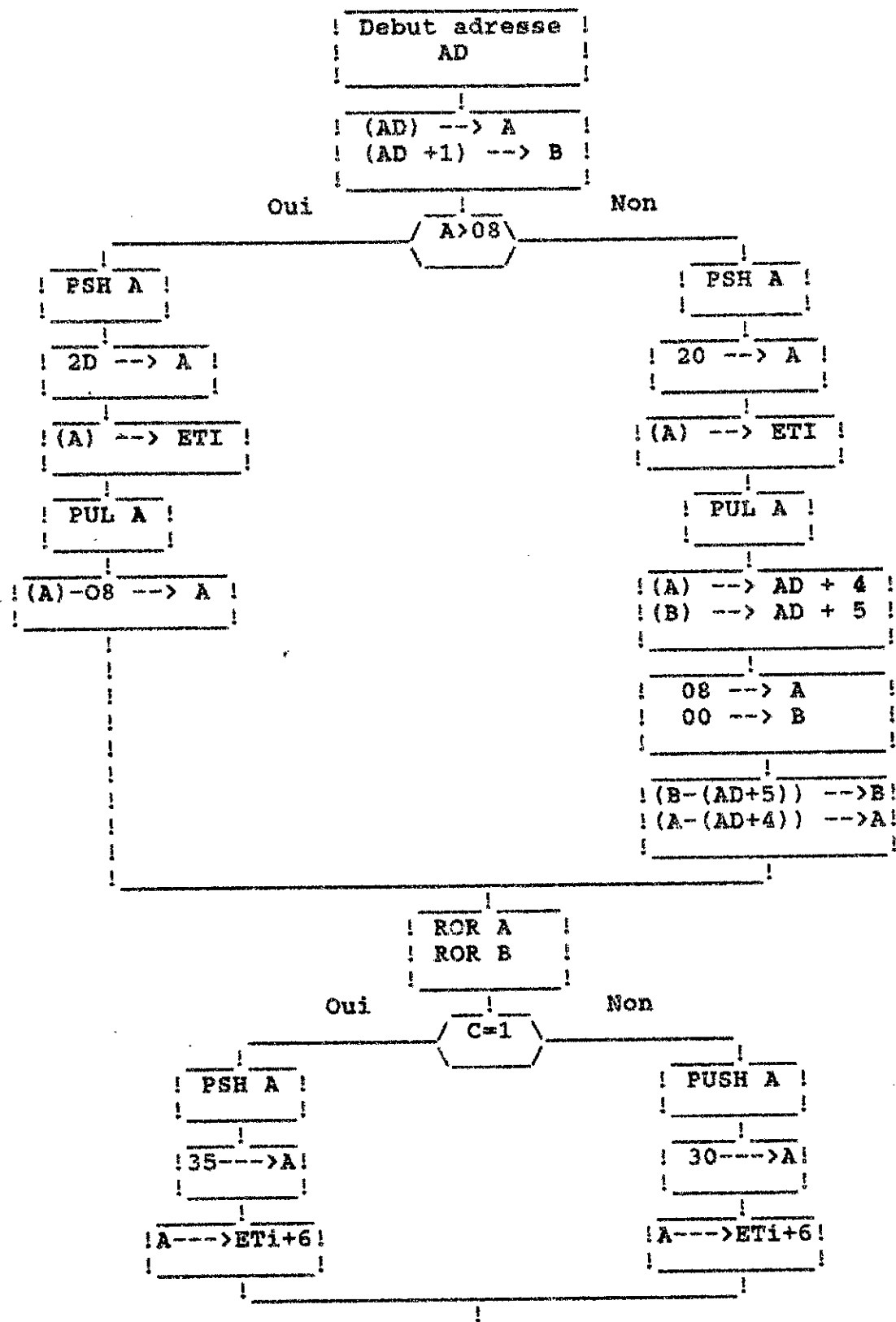
Fonction logarithme

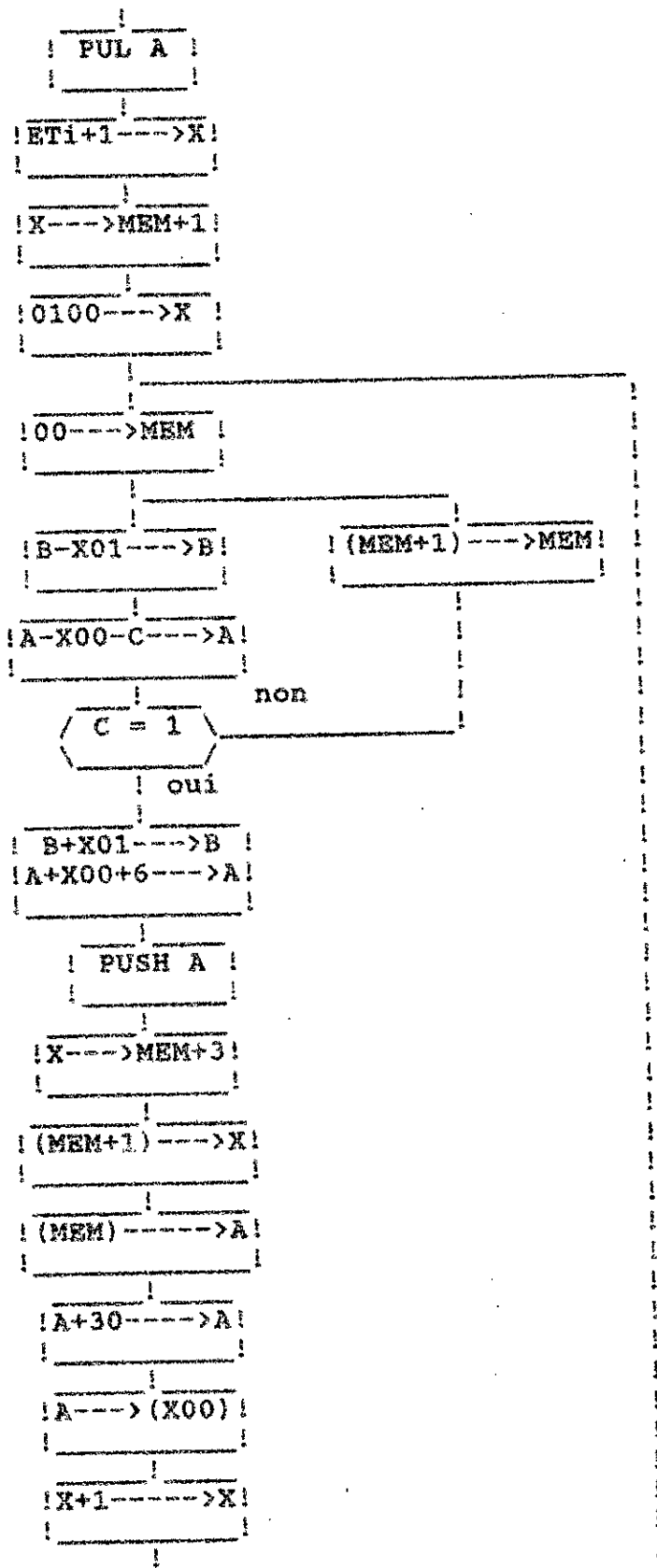
-----  
Tension d'entree : 1 mV + 10 V  
Erreur statique :  $0 < V_e < 7$  V + 16 mV  
Bande passante : 100 Khz

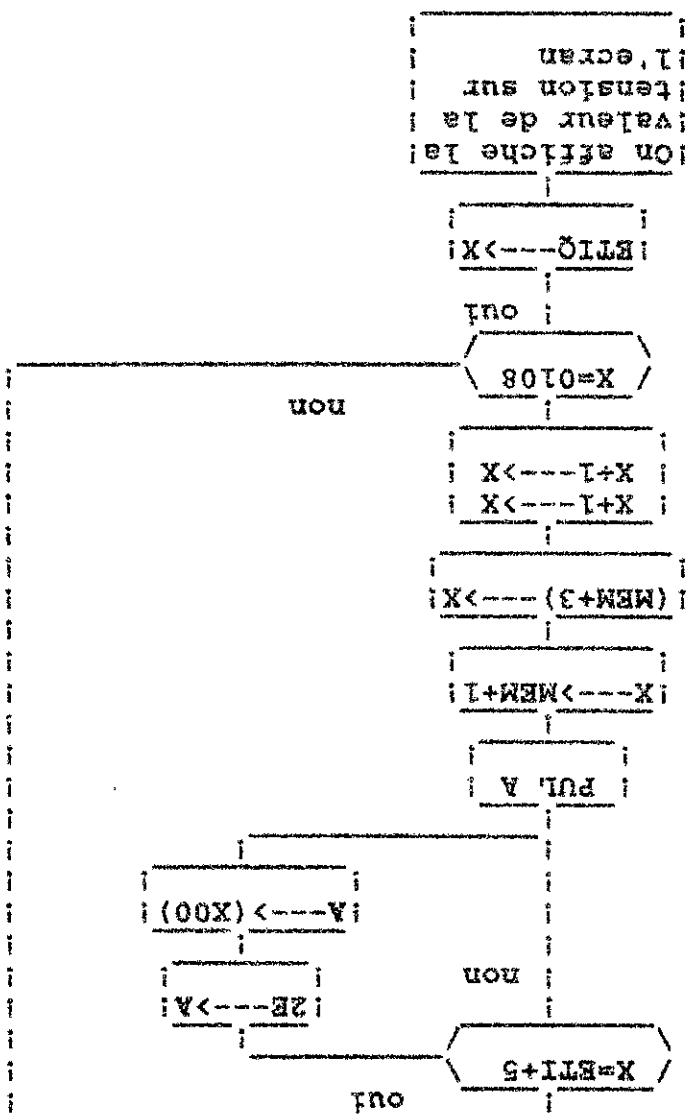
Potentiometre numerique :

-----  
Resolution : 12 bits  
Code binaire decale  
Tension de sortie : + 10 V

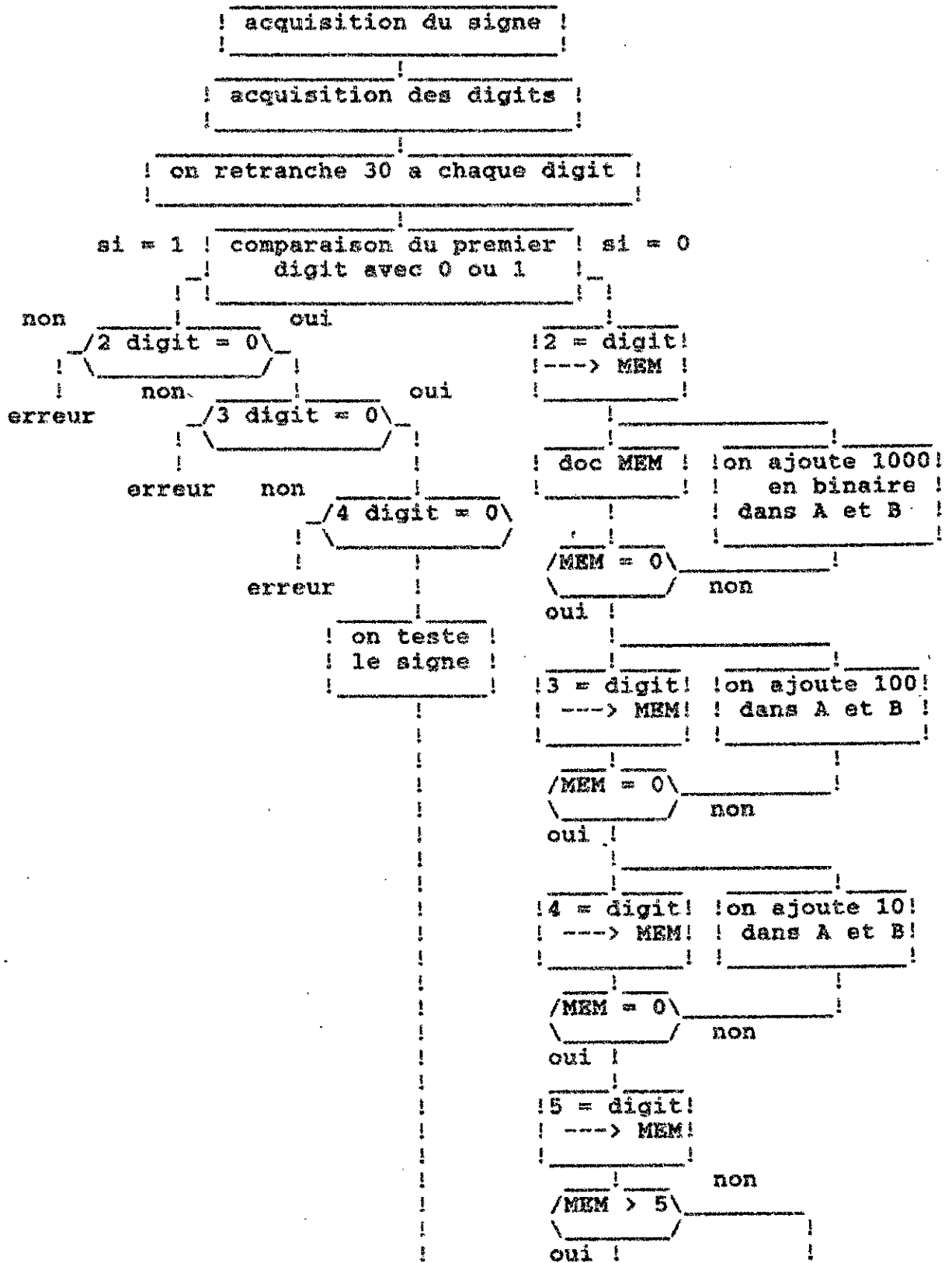
Programme de conversion  
Binaire 12 bits ----> Decimal ASCII











on met 0  
dans b0

on met 1 dans  
b0 de la  
configuration  
A et B

on charge le  
potentiometre  
concerne de la  
configuration  
contenue