

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE DES SCIENCES ET DE LA TECHNOLOGIE
« HOUARI BOUMEDIENNE »

ECOLE NATIONALE POLYTECHNIQUE

الجامعة الوطنية للعلوم الهندسية
Département d'Electronique
الالكترونية

PROJET DE FIN D'ETUDES
ECOLE NATIONALE POLYTECHNIQUE
»O«
BIBLIOTHÈQUE

INGENIORAT D'ETAT EN ELECTRONIQUE

SUJET

*Etude et Réalisation d'un Programmeur
de Température à l'aide du Micro-
Ordinateur APPLE II Plus*

Proposé par : M. SLUSZKIEWICZ

Réalisé par : Melle GHEZAL Nadia

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE DES SCIENCES ET DE LA TECHNOLOGIE
« HOUARI BOUMEDIENNE »

ECOLE NATIONALE POLYTECHNIQUE

Département d'Electronique

PROJET DE FIN D'ETUDES

—————»O«—————

INGENIORAT D'ETAT EN ELECTRONIQUE

SUJET

*Etude et Réalisation d'un Programmeur
de Température à l'aide du Micro-
Ordinateur APPLE II Plus*

Proposé par : M. SLUSZKIEWICZ

Réalisé par : Melle GHEZAL Nadia

- R E M E R C I E M E N T S -

Je tiens à remercier Monsieur SLUSZKIEWICZ, enseignant à l'E.N.P.A pour m'avoir proposé ce sujet.

Mes plus vifs remerciements à l'équipe du laboratoire Traitement des données, en particulier Monsieur BOUDRAA pour son précieux concours et ses remarques judicieuses.

Ma sincère gratitude à tous ceux qui ont contribué de près ou de loin à l'élaboration de ce travail.

T A B L E D E S M A T I E R E S

INTRODUCTION.

I- PRESENTATION DU TRAVAIL.

A - Exposé du problème.

A-I- But de l'étude des échantillons diélectriques.

A-2- Expériences menées au cours de l'étude.

B - Cahier de Charges.

II- SIMULATION DE LA TEMPERATURE EN FONCTION DU TEMPS.

A - Présentation de l'Apple II plus.

B - Elaboration du programme de simulation.

III- ETUDE ET REALISATION DE L'INTERFACE.

A - Etude des entrées-sorties de l'Apple II plus.

A-I- Description des entrées-sorties de l'Apple II plus.

A-2- Connecteurs de périphériques.

A-3- Mémoires d'entrées-sorties.

B - L'INTERFACE.

B-I- Rôle de l'interface.

B-2- Interface d'entrée-sorties pour périphériques:PIA.

B-2-I- Organisation externe.

B-2-2- Organisation interne.

B-2-3- Sélection du boîtier et des registres internes.

B-2-4- Adressage du PIA.

B-2-5- Programmation du PIA.

- B-3- Convertisseur numérique-analogique.
- B-3-1- Principales caractéristiques du convertisseur.
- B-3-2- Description du convertisseur
et principe de la conversion.
- B-4- Transfert des données et des commandes
- B-5- Réalisation de l'interface.
- B-6- Gestion de l'interface.
- B-7- Utilisation du programmeur.
- B-8- Autres possibilités de conception du programmeur.
- B-9- Contrôle du processus par le micro-ordinateur.

CONCLUSION.

INTRODUCTION
=====

Le travail effectué au cours de ce projet de fin d'études est en rapport avec les études faites sur les échantillons diélectriques au laboratoire du même nom à l'institut de Physique de Bab Ezzouar.

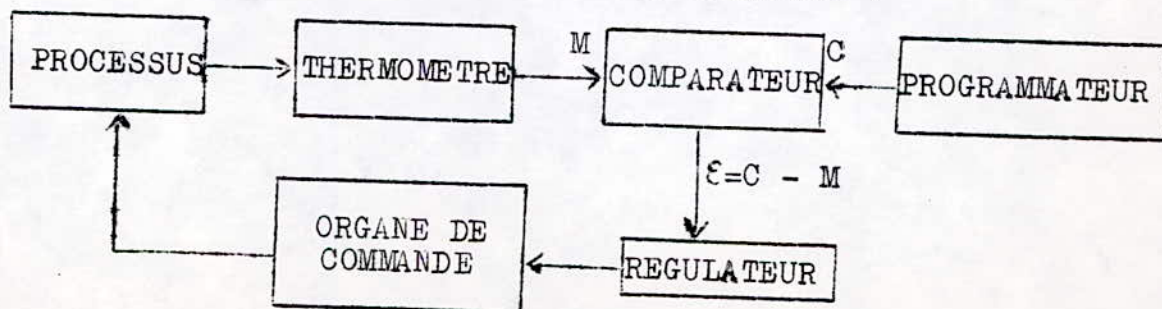
Dans le cadre de ces études, il s'est avéré que la température jouait un rôle important. Cette dernière, afin de permettre de mener à bien ces études, doit varier en fonction du temps suivant un cahier de charges bien défini.

Pour cela, la solution envisagée est d'asservir la température réelle à suivre les variations de la température de référence appelée encore température de consigne.

Cette solution comporte deux parties :

- La première partie consiste à générer ces variations de température de consigne.
- La deuxième partie consiste à forcer le système à suivre ces variations.

Le synoptique suivant résume cette solution :



La température lue par le thermomètre et la température de consigne générée par le programmeur arrivent sur un comparateur qui effectue leur différence $\xi = C - M$.

Cette valeur est régulée de manière à minimiser le plus possible cet écart c'est à dire le rendre nul d'où on aura $M = C$.

Le but de ce projet de fin d'études est d'étudier et de réaliser le programmeur qui a pour rôle de générer les variations de la température de consigne en fonction du temps et de les transmettre au comparateur sous forme de tensions analogiques.

- Le premier chapitre est consacré à la présentation du travail.
 - Le deuxième chapitre expose le programme de simulation de la courbe de température et comporte également une brève description du microordinateur utilisé : l'Apple II plus.
 - Le troisième chapitre est consacré à l'étude et la réalisation de l'interface entre le microordinateur et le comparateur; et au mode d'emploi du programmeur.
- Ce chapitre donne également une idée sur d'autres possibilités de concevoir le programmeur ainsi que sur la manière dont pourrait être contrôlé le processus par le micro-ordinateur.

A. Exposé du Problème :

Pour arriver à établir le cahier de charges qui régit les variations de la température de consigne, il est utile de donner un aperçu sur le but de l'étude des échantillons et sur les expériences menées au cours de cette étude.

A-I. But de l'étude des échantillons diélectriques :

Les diélectriques sont caractérisés par une grande résistivité; donc par un courant de conduction faible.

En présence d'un champ électrique, l'énergie dissipée dans le matériau est due non seulement à la conduction, mais aussi à des phénomènes de relaxation.

La relaxation est le phénomène de passage des groupements de charges de l'échantillon, d'un état orienté, sous l'effet du champ électrique, à leur état d'équilibre après suppression du champ.

Le but de l'étude de ces échantillons est de déterminer ces différents groupements grâce à leur température de relaxation et de pouvoir ainsi déterminer l'énergie dissipée dans le matériau.

A.-2. Expériences menées au cours de l'étude :

L'échantillon diélectrique est soumis, à la température T_p dite de polarisation, à un champ électrique \vec{E} pendant un intervalle de temps PH afin d'orienter les dipôles de l'échantillon dans la même direction que \vec{E} .

Ensuite on refroidit cet échantillon en présence du champ jusqu'à la température $T_0 = -160^\circ\text{C}$.

On bloque ainsi toute possibilité de désorientation immédiate des dipôles; on dit qu'on "gèle" la polarisation qui substituera après la suppression du champ. On maintient la température T_0 pendant un intervalle de temps PB .

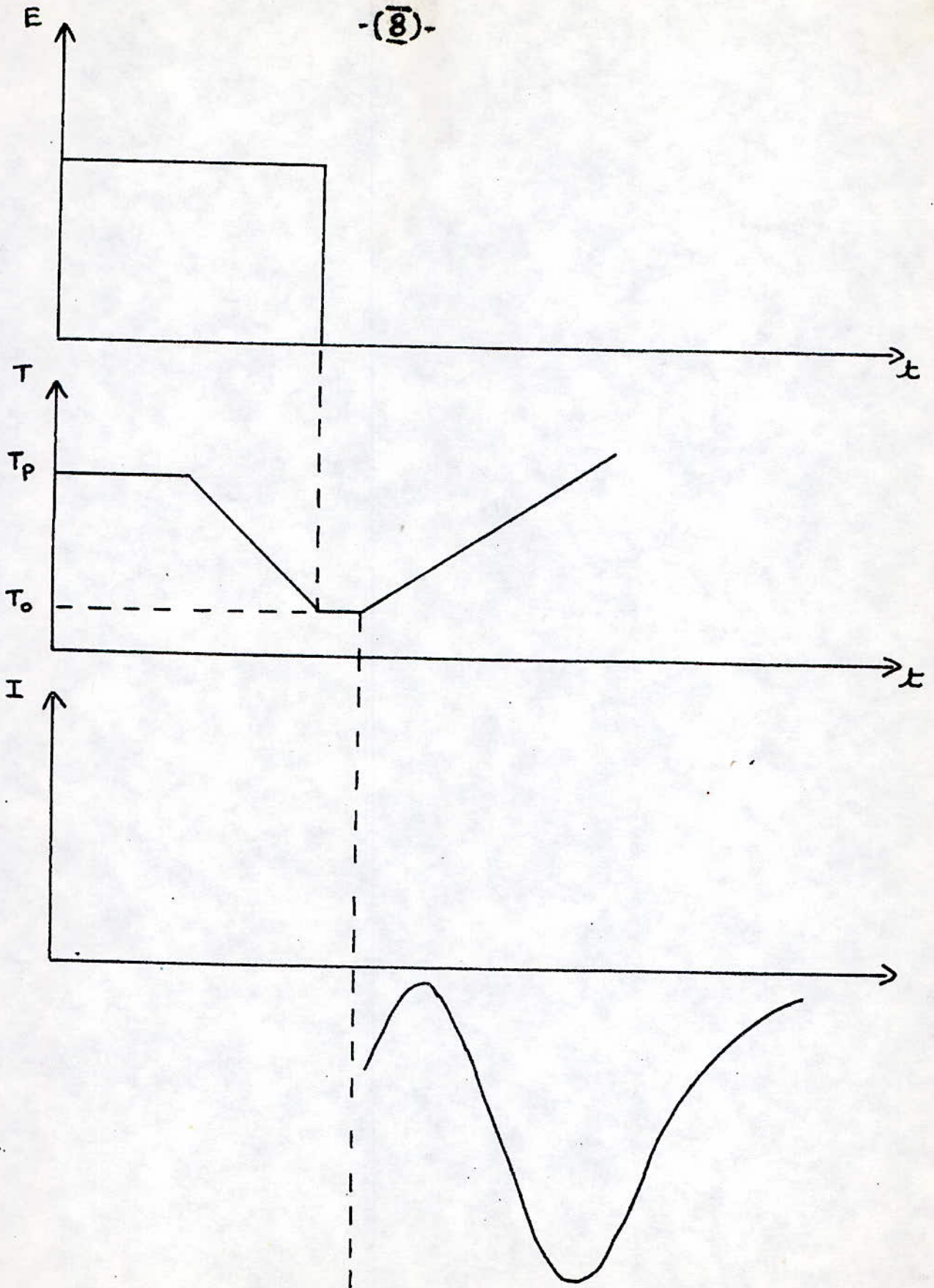
Ensuite on fait croître linéairement la température de l'échantillon en court-circuit sur un électromètre.

L'échantillon est composé de différents groupements de charges et chacun va retrouver son état d'équilibre ou état initial à une température bien définie.

Un courant de dépolarisation apparaît alors sous forme de "pics" correspondant aux différentes relaxations et expriment le retour des charges à l'équilibre.

Les différentes étapes de cette méthode sont représentées par les graphes de la page suivante.

-(8)-



On relève les variations du courant I de dépolarisation en fonction de la température : c'est le spectre global.

Mais dans la plupart des cas, on n'est pas en présence d'un temps de relaxation unique. Un même groupement de charges peut avoir plusieurs temps de relaxation.

Pour étudier cela, on fait appel à des techniques de décomposition de spectres complexes en une série de pics élémentaires à un seul temps de relaxation.

- Soit un pic du spectre global situé à la température T_M .
On considère la zone de température : $(T_M - \Delta T, T_M + \Delta T)$ située autour de la température T_M du pic.

On soumet à nouveau l'échantillon à un champ électrique à la température $T = T_M - \Delta T + 10$, on maintient constante cette température durant l'intervalle de temps PH afin d'orienter les unités qui ont un temps de relaxation inférieur à $\tau_M (T_M - \Delta T + 10)$.

Ensuite la température est abaissée jusqu'à $(T_M - \Delta T)$ et le champ électrique est annulé.

Cette température $T_M - \Delta T$ est maintenue constante pendant le temps PH ; ce qui permet le retour à l'équilibre des unités dont le temps de relaxation τ est inférieur à $\tau_M (T_M - \Delta T)$.

De ce fait seuls les dipôles dont le temps de relaxation est compris entre $\tau_M (T_M - \Delta T + 10)$ et $\tau_M (T_M - \Delta T)$ seront affectés par le champ.

Par la suite on "gèle" cette polarisation en refroidissant le système jusqu'à la température $T_0 = - 160^\circ\text{C}$.

L'échantillon est alors court circuitée sur l'électromètre et une remontée linéaire en température permet l'enregistrement du pic élémentaire relatif à la tranche de température $(T_M - \Delta T, T_M - \Delta T + 10)$.

Un travail analogue est effectué pour les bandes de température suivantes :

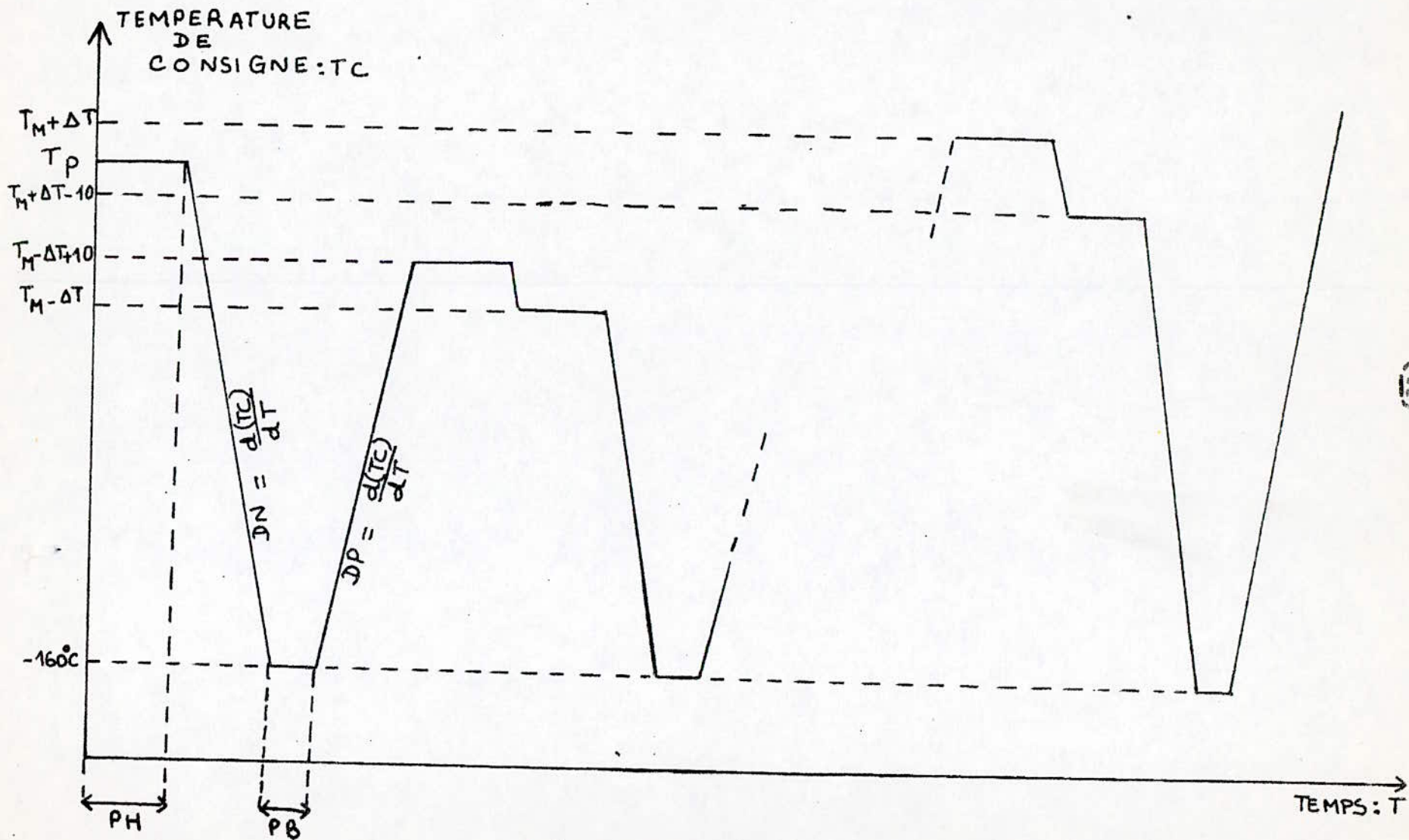
$(T_M - \Delta T + 10, T_M - \Delta T + 20)$; $(T_M - \Delta T + 20, T_M - \Delta T + 30)$
 $(T_M + \Delta T - 10, T_M + \Delta T)$.

Et à chaque bande de température on enregistre le pic élémentaire.

A partir de l'énoncé des expériences précédentes, on peut donc établir le cahier de charges du programmeur.

B. Cahier de Charges :

La température à laquelle est soumis l'échantillon doit varier suivant l'allure donnée ci-contre :



VARIATIONS DE LA TEMPERATURE
DE CONSIGNE

- Les pentes des droites croissantes DP varient entre $2^{\circ}\text{C}/\text{mn}$ et $8^{\circ}\text{C}/\text{mn}$.
- Les pentes des droites décroissantes DN varient entre $-50^{\circ}\text{C}/\text{mn}$ et $-10^{\circ}\text{C}/\text{mn}$.
- La durée des paliers PH varie entre 1 mn et 4 mn.
- La durée des paliers PB varie entre 0,5 mn et 1 mn.
- Limite des variations de TC : de -160°C à $+160^{\circ}\text{C}$.

Le programmeur, dans sa conception, doit pouvoir :

- Simuler les variations de la courbe de la température de consigne par programme.
- Transmettre les différentes températures de consigne au comparateur sous forme de tensions analogiques grâce à un interface numérique analogique entre l'Apple II plus et le comparateur.

Le chapitre II étudie la manière de simuler la courbe de la température de consigne tandis que le chapitre III expose l'étude et la réalisation de l'interface.

CHAPITRE II.

S I M U L A T I O N D E L A
T E M P E R A T U R E E N
F O N C T I O N D U T E M P S

Pour pouvoir aborder la simulation de la courbe de température en fonction du temps ainsi que l'étude de l'interface qu'assure la fonction de programmeur, la connaissance du système AppleII plus est utile.

L'étude de ce système comportera une description brève du micro-ordinateur.

A. Présentation de l'Apple II plus :

Le micro-ordinateur utilisé dans le but de cette simulation est l'Apple II plus.

C'est un micro-ordinateur conçu autour du microprocesseur MCS 6502. Il lui donne la possibilité d'adresser jusqu'à 65536 (64 K) mots de 8 bits.

Ces 64 K sont attribués aux :

- Fonctions d'entrées-sorties des données (4 K)
- Mémoires mortes : ROM (12 K)
- Mémoires vives : RAM (48 K)

Pour communiquer avec l'environnement par l'intermédiaire d'un clavier, d'un écran vidéo, d'un lecteur enregistreur de cassettes, par exemple, le système contient des interfaces spécifiques intégrés à la carte-mère.

Il permet également la connexion de cartes d'interface (huit au maximum) permettant une extension de la mémoire et la connexion à des dispositifs extérieurs.

L'alimentation du système permet de fournir quatre tensions :
-5V, +5V, -12V, +12V.

Le dialogue avec le système se fait grâce à un clavier de 64 caractères codés en A S C I I.

Le langage utilisé, pour le programme de simulation, est le BASIC APPLESOFT.

Ce langage à la fois simple et puissant s'adapte suffisamment aux exigences de cette simulation.

B. Elaboration du Programme de Simulation de la température :

La simulation de l'allure de la température va se faire en tenant compte du cahier de charges établi dans le chapitre précédent.

Ce dernier donne l'allure de la température en fonction du temps.

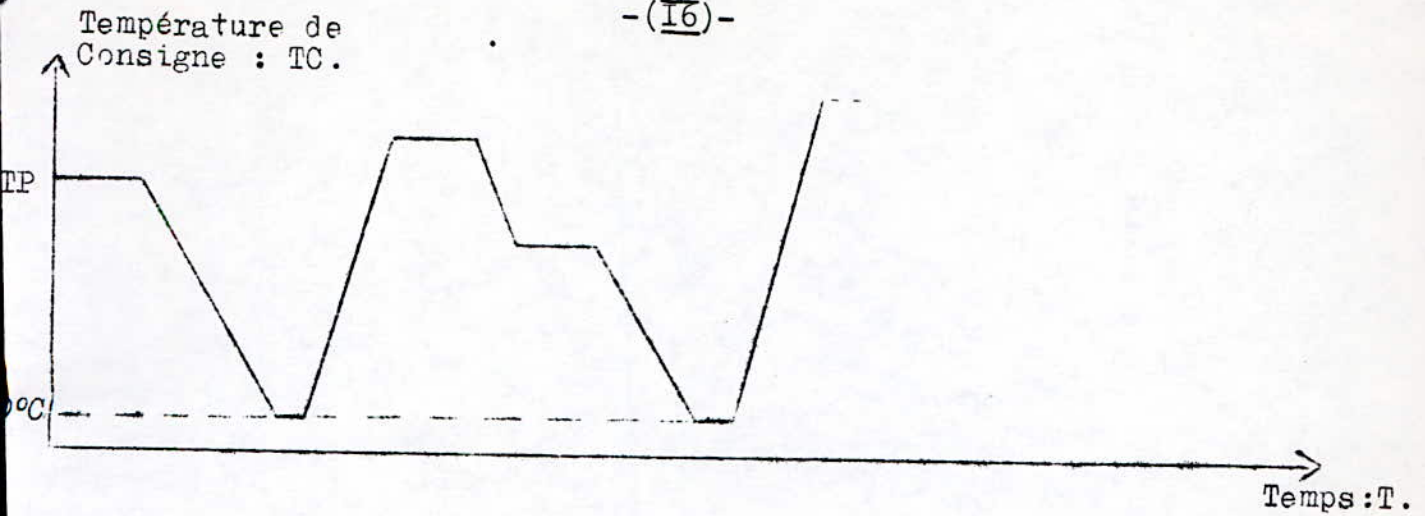
Le but est donc de générer cette allure et de la transmettre à l'interface; ce dernier se chargera de convertir les données qu'il reçoit en tensions analogiques.

Par la suite, ces tensions seront transmises au comparateur.

La première partie de cette tâche, c'est à dire l'élaboration du programme de simulation, sera traitée dans ce chapitre.

Il s'agit de programmer le microordinateur Apple II plus afin qu'il délivre sur le port A du PIA, configuré en sortie, la succession de mots numériques corrects pour que le convertisseur numérique-analogique génère le signal analogique demandé.

Pour cela, il faut considérer attentivement la courbe suivante:



Au début du programme, on doit introduire les paramètres de l'échantillon c'est à dire : DN, DP, PH, PB; ainsi que la température de polarisation TP qui est la première température de consigne. (Les variables utilisées sont données à la page 19).

Il faut également, après l'enregistrement du spectre global, définir les zones de température situées autour des "pics" afin d'introduire les valeurs des différents paliers dans le programme.

Le programme doit faire correspondre à chaque instant T, multiple de la période d'échantillonnage, la température de consigne qui lui correspond par la courbe ci-dessus.

Donc à chaque fois que T augmente de TE, TC doit augmenter de DC, DC étant négatif, nul ou positif suivant que l'on considère une rampe décroissante, un palier ou une rampe croissante.

Les différentes valeurs prises par DC sont :

- Pour un palier, la température de consigne est constante, donc l'incrément de température DC est nul.

- Pour une rampe croissante :

DP étant la pente de cette rampe, TE la période d'échantillonnage, on a :

$$DP = \frac{DC}{TE} ; \text{d'où} : DC = DP \times TE$$

- Pour une rampe décroissante :

DN : Pente de cette rampe.

$$DN = \frac{DC}{TE} ; \text{d'où} : DC = DN \times TE$$

Choix de la période d'échantillonnage :

La période d'échantillonnage ne peut pas être choisie très petite. Le langage utilisé pour le programme étant le basic (à cause des multiplications) et comme le calcul dans ce langage est assez lent, vu le temps d'interprétation des instructions, la période d'échantillonnage ne peut pas être choisie petite.

Pour remplir cette condition, la période d'échantillonnage est fixée à 0.1 minute.

Pour generer cette période d'échantillonnage, on fait appel au sous programme suivant :

```
55 L = 360  
60 FOR I = 1 TO L : NEXT I
```

Toute valeur de TC est maintenue pendant le temps d'échantillonnage TE grâce au délai generé par les deux instructions données ci-dessous.

Variables utilisées dans le programme :

- DP : Pente des rampes croissantes ($^{\circ}\text{C}/\text{mn}$)
DN : Pente des rampes décroissantes ($^{\circ}\text{C}/\text{mn}$)
PH : Durée des paliers hauts (mn)
PB : Durée des paliers bas (mn)
TE : Période d'échantillonnage (mn)
T : Temps courant
TF : Heure du prochain changement de pente (mn)
TC : Température de consigne
TI : Température du prochain palier (pendant une rampe)
ou température du palier en cours.
DC : Increment de température à chaque période d'échantil-
lonnage.
IP : IP = 0 pour un palier
IP = 1 pour une rampe
TT : Variable intermédiaire.
Pour les lignes 2070 et 2100 TT représente la durée
du palier.

Exemple d'étude d'un échantillon dont les paramètres sont les suivants :

$$\bar{DP} = 3^{\circ}\text{C}/\text{mn}$$

$$DN = 10^{\circ}\text{C}/\text{mn}$$

$$PH = 3 \text{ mn}$$

$$PB = 0.5 \text{ mn}$$

$$TE = 0.1 \text{ mn}$$

$$TP = 0^{\circ}\text{C}$$

Le spectre global donne un "pic" situé à la température $T_M = 20^{\circ}\text{C}$.

Les paliers ont pour valeurs :

0, -10, -160, 10, 0, -160, 20, 10, -160, 30, 20, -160, 150, 100

Le programme sera le suivant :

```
20 DATA 3, -10, 3, 0.5, 0.1
25 READ DP, DN, PH, PB, TE
30 TC = 0
35 T = 0
40 TF = 0
45 IP = 1
50 GOSUB 2000
55 L = 360
60 FOR I = 1 TO L : NEXT I
65 GOSUB 1000
70 GOTO 60
```

```
I000 DATA -I60, 0, - I0, -I60, I0, 0, -I60, 20,I0 -I60,30,20
I030 TC = TC + DC -I60,I50,I60.
I040 T = T + TE
I050 IF DC = 0 THEN III0
I060 IF DC > 0 THEN I090
I070 IF TC > TI THEN II30
I080 GOTO 2000
I090 IF TC < TI THEN II30
II00 GOTO 2000
III0 IF T < TF THEN II30
II20 GOTO 2000
II30 REM OUT PUT
II60 PRINT TC
I2I0 RETURN
2000 REM CHANGEMENT DE PENTE
2010 IP = I - IP
2020 IF IP = 0 THEN 2I40
2030 READ TI
2040 IF TI = I60 THEN STOP
2050 TT = TI - TC
2060 IF TT > 0 THEN 2I00
2070 TT = TT/DN
2080 DC = TE * DN
2090 GOTO 2I20
2I00 TT = TT/DP
```


2110 DC = TE * DP
2120 TF = TF + TT
2130 GOTO I000
2140 IF DC < 0 THEN 2170
2150 TF = TF + PH
2160 GOTO 2180
2170 TF = TF + PB
2180 TC = TI
2190 DC = 0
2200 GOTO I000

CHAPITRE III. ETUDE ET REALISATION
DE L'INTERFACE

A - Etude des Entrées Sorties de l'Apple II plus.

A - I. Description des Entrées Sorties de l'Apples II plus :

Dans l'espace d'adresses $\$ C \emptyset \emptyset \emptyset$ à $\$ C F F F$ sont disposés 4096 Octets exclusivement réservés aux fonctions d'entrées et de sorties d'informations.

Ces dernières sont de 2 types :

- a)- Les fonctions d'entrées-sorties intégrées sur la Carte-Mère : Elles regroupent les fonctions à usage domestique.

Ces entrées-sorties sont : - Le clavier

- Le Haut Parleur

- Le Générateur Vidéo

- L'Interface Cassette.

Elles occupent 128 Octets de $\$ C \emptyset \emptyset \emptyset$ à $\$ C \emptyset 7 F$

- b)- Les fonctions d'entrées-sorties de périphériques :

Outre les entrées-sorties intégrées sur la Carte-Mère,

le système Apple II plus possède des possibilités

d'extension sur huit périphériques différents grâce

à huit connecteurs disponibles au fond de la Carte-Mère.

A - 2. Connecteurs de Périphériques :

Le système Apple II plus peut communiquer avec des périphériques grâce à des circuits d'interface spéciaux et des programmes de gestion des signaux et des commandes particulières d'entrées-sorties, par l'intermédiaire des huit connecteurs de périphériques numérotés de 0 à 7.

Chaque connecteur comprend 50 broches permettant de :

- Disposer de signaux d'adresses, de données, de sélection d'adresses (donc de de Périphériques).
- Générer des interruptions.
- Déconnecter les ROM de la carte-mère.

Cette architecture d'entrées-sorties fait que les périphériques sont vus du microprocesseur comme des emplacements mémoire auxquels toutes les possibilités d'adressage de la mémoire sont applicables.

A - 3. Mémoires d'Entrées-Sorties :

A chaque connecteur (sauf le connecteur n°0) le système AppleII plus fait correspondre 256 adresses de son espace mémoire. Cette page de 256 positions contient un programme en ROM réalisant les fonctions d'entrées-sorties spécifiques du périphérique branché. Le début de cette page a pour adresse $\$ C S \emptyset \emptyset$ S étant le numéro du connecteur où a été branchée la carte d'interface.

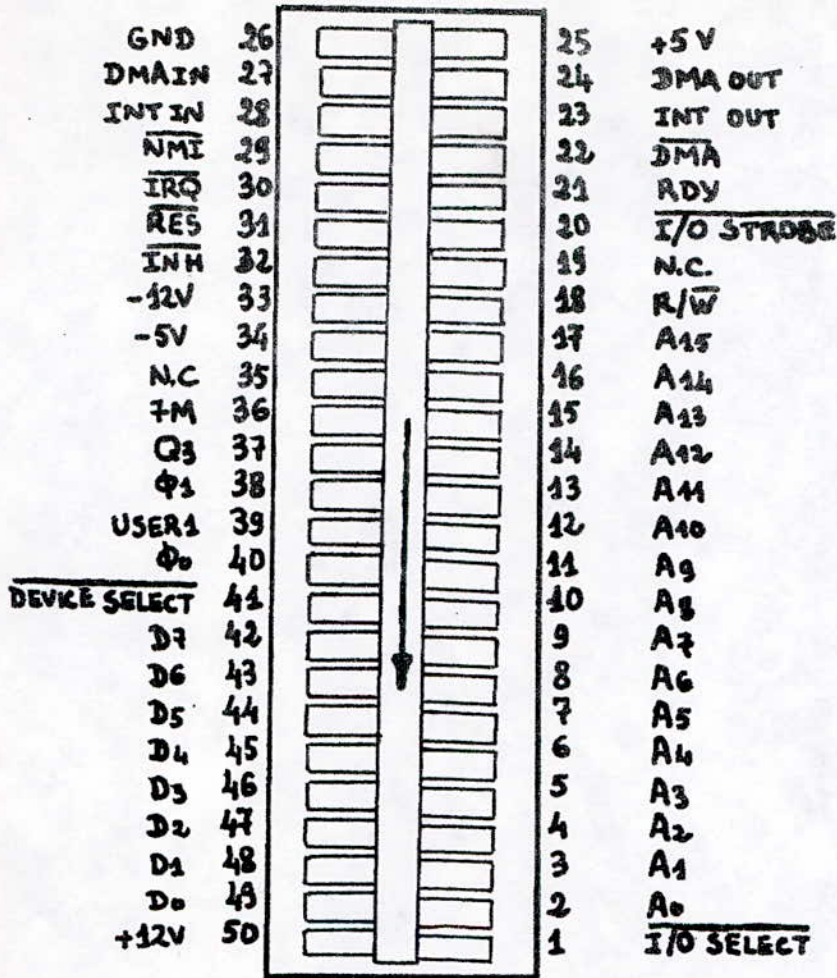
De plus chaque connecteur dispose de 16 positions Mémoire réparties entre $\$ C \emptyset 8 \emptyset$ et $\$ C \emptyset F F$ (pour l'ensemble des 8 connecteurs)

La zone Mémoire réservée pour le connecteur n°0 se situe entre $\$ C \emptyset 8 \emptyset$ et $\$ C \emptyset 8 F$.

Connecteur n°1 : de $\$ C \emptyset 9 \emptyset$ à $\$ C \emptyset 9 F$

Connecteur n°2 : de $S C \emptyset A \emptyset$ à $S C \emptyset A F$

Connecteur n°3 : de $S C \emptyset B \emptyset$ à $S C \emptyset B F$



Brochage du connecteur de peripherique

- Connecteur n°4 : de S C Ø C Ø à S C Ø C F
- Connecteur n°5 : de S C Ø D Ø à S C Ø D F
- Connecteur n°6 : de S C Ø E Ø à S C Ø E F
- Connecteur n°7 : de S C Ø F Ø à S C Ø F F

Chaque carte d'interface peut utiliser ces adresses à sa guise pour réaliser des fonctions particulières d'entrées sorties sur le périphérique branché à cette carte.

De même un espace de 2048 adresses, de § C 8 Ø Ø à § C F Ø Ø est utilisable pour tout périphérique nécessitant un programme d'entrée-sortie plus élaboré qui sera figé en ROM sur la carte d'interface correspondante.

Chaque interface peut contenir cette ROM de 2 K octets mais une seule à la fois sera en "ligne" pour le système Apple à un moment donné.

=====

TABLEAU MONTRANT L'UTILISATION DES 4 K RESERVES AUX

ENTREES - SORTIES

=====

| NOMBRE D'OCTETS | ADRESSES | UTILISATION |
|-----------------|---------------------------------------|--|
| 128 | § C Ø Ø Ø à S C Ø 7 F | Entrées-Sorties contrôlées sur la Carte-Mère. |
| 16 X 8 | § C Ø 8 Ø à § C Ø F F | 16 adresses pour chacun des 8 Connecteurs d'entrée-sorties. |
| 256 X 7 | § C S Ø Ø à § C S F F S (I + 7) | 256 adresses par carte d'interface pour un programme d'entrée-sortie en ROM. |
| 2048 | § C 8 Ø Ø à § C F F F | 2048 adresses pour d'autres programmes en ROM sur les Cartes d'interface. |

B - L'Interface :

B - I. Rôle de l'interface :

La température calculée, à chaque période d'échantillonnage par le Microordinateur, et devant être transférée vers le comparateur, est sous forme binaire.

Ce transfert se fait à chaque fois que le signal d'écriture-lecture R/\bar{W} est à zéro et durant la phase zéro du microprocesseur.

Le rôle de l'interface est de :

- Recevoir la température calculée par le Micro ordinateur
- Transformer cette donnée binaire en une tension analogique.
- La transférer vers le comparateur afin que s'effectue la comparaison avec la température lue sur le thermomètre, cette dernière étant également transmise sous forme de tension analogique.

En premier lieu, le système doit réserver au périphérique une zone mémoire.

De cette manière, toute information transférée vers cette zone de mémoires sera prise en compte par le périphérique.

Cela se fait grâce à un circuit d'interface parallèle : le PIA.

B - 2. Interface d'entrées-sorties pour Périphériques : PIA :

LE PIA utilisé pour la réalisation de l'interface est le P I A M C 6821 de Motorola.

C'est un circuit d'entrée-sortie programmable.

Il assure l'interface entre le microprocesseur et le périphérique réalisant ainsi la synchronisation de l'Apple II plus avec ce périphérique.

Le P I A est prévu pour être connecté à un bus de type 6800. Les microprocesseurs 6502 et 6800 étant compatibles (du point de vue hard), le P I A peut donc être relié à un des connecteurs de l'Apple II plus; et ne nécessite aucun circuit d'adaptation autre qu'un éventuel décodage d'adresse pour placer le P I A dans l'espace mémoire adressable par le microprocesseur.

B - 2.I. Organisation Externe :

Le P I A dispose de :

- 16 lignes d'entrées-sorties programmables individuellement et indépendamment les unes des autres en entrées ou en sorties. Cette programmation se fait par logiciel.
- 4 lignes de contrôle : Elles permettent d'envoyer des demandes d'interruption. Deux d'entre elles sont programmées en entrées, les deux autres peuvent être programmables en entrées ou en sorties.
- 3 "Chip Select" : $CS_0, CS_1, \overline{CS}_2$. Ils permettent la sélection du boîtier. Ce dernier n'est activé que lorsque $CS_0 = CS_1 = 1$ et $\overline{CS}_2 = 0$.

- Les lignes RS_0 et RS_1 : Lignes de sélection des registres internes.
- L'entrée E du P I A : Elle reçoit le signal Phase 0 du microprocesseur assurant ainsi la synchronisation des opérations dans l'interface.
- La ligne RESET : Elle est reliée au RESET du connecteur. Elle remet à zéro les registres internes du P I A.
- La ligne R/\bar{W} appliquée à R/\bar{W} du connecteur. Elle contrôle le sens de transfert des données circulant entre l'Apple II plus et le périphérique.
- Les lignes D_0 à D_7 : Elles sont appliquées au bus de données. Elles servent à véhiculer les données.

B - 2.2. Organisation Interne :

Le P I A est un circuit symétrique disposant de 2 moitiés identiques : le Port A et le Port B.

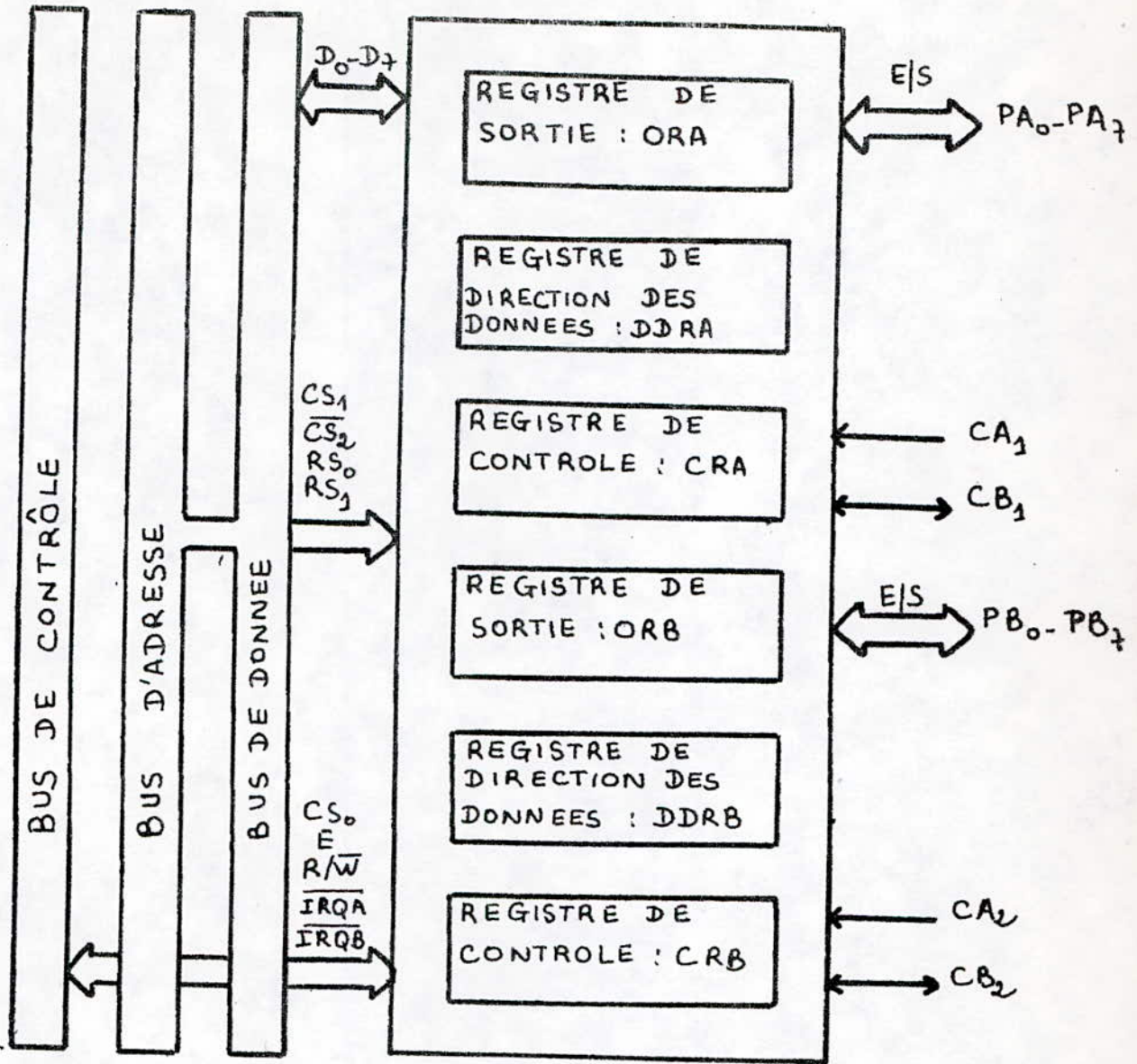
A chaque port correspond 3 registres :

- Un registre DDR : Régistre de direction des données. Il permet de définir le sens de transfert des données. Chaque bit de ce registre indique le sens de travail d'une ligne d'entrée-sortie.
- Un registre CR : registre de commande. Il permet de définir le mode de fonctionnement des lignes de contrôle ainsi que les possibilités de génération d'interruption à l'aide des lignes d'interruption.

| | | | |
|-----------------|----|----|-----------------|
| V _{SS} | 1 | 40 | CA ₁ |
| PA ₀ | 2 | 39 | CA ₂ |
| PA ₁ | 3 | 38 | IRQA |
| PA ₂ | 4 | 37 | IRQB |
| PA ₃ | 5 | 36 | RS ₀ |
| PA ₄ | 6 | 35 | RS ₁ |
| PA ₅ | 7 | 34 | Reset |
| PA ₆ | 8 | 33 | D ₀ |
| PA ₇ | 9 | 32 | D ₁ |
| PB ₀ | 10 | 31 | D ₂ |
| PB ₁ | 11 | 30 | D ₃ |
| PB ₂ | 12 | 29 | D ₄ |
| PB ₃ | 13 | 28 | D ₅ |
| PB ₄ | 14 | 27 | D ₆ |
| PB ₅ | 15 | 26 | D ₇ |
| PB ₆ | 16 | 25 | E |
| PB ₇ | 17 | 24 | CS ₁ |
| CB ₁ | 18 | 23 | CS ₂ |
| CB ₂ | 19 | 22 | CS ₀ |
| V _{CC} | 20 | 21 | R/ \bar{W} |

P.I.A
MC6821

Brochage du PIA



ORGANISATION INTERNE & EXTERNE
DU PIA

- Un registre OR : c'est le registre dans lequel le micro-
 processeur vient placer les données à faire sortir du PIA.
 Les registres OR et DDR d'un même port occupent une seule posi-
 tion mémoire. L'état du bit b2 du registre de commande permet
 la sélection entre les deux registres.

B - 2.3. Sélection du Boitier et des Registres Internes :

CS₀, CS₁ et \overline{CS}_2 sont des pattes de sélection du boitier P I A.
 Ce dernier est activé lorsque CS₀ = CS₁ = I et \overline{CS}_2 = 0.
 Une fois que le boitier est sélectionné, on adresse les regis-
 tres internes grâce à RS₀ et RS₁ et le bit b2 du registre de
 commande.

| CS ₀ | CS ₁ | \overline{CS}_2 | RS ₁ | RS ₀ | b2 | R E G I S T R E |
|-----------------|-----------------|-------------------|-----------------|-----------------|----|-----------------|
| I | I | 0 | 0 | 0 | 0 | DDRA |
| I | I | 0 | 0 | 0 | I | ORA |
| I | I | 0 | 0 | I | X | CRA |
| I | I | 0 | I | 0 | 0 | DDRB |
| I | I | 0 | I | 0 | I | ORB |
| I | I | 0 | I | I | X | CRB |

TABLEAU DE SELECTION
 =====
 DES REGISTRES INTERNES
 =====
 DU P I A.
 =====

B - 2.4. Adressage du P I A :

A chaque connecteur, le système réserve une zone mémoire de seize adresses, et pour toute adresse de cette zone la ligne DEVICE SELECT est activée.

Dans le cas de cet interface, le connecteur utilisé est le connecteur n°2.

La ligne DEVICE SELECT sera donc active pour une adresse comprise entre $\$ C \emptyset A \emptyset$ et $\$ C \emptyset A F$. Cette ligne est reliée à $\overline{CS2}$. Les registres internes du PIA représentent quatre positions mémoires et comme $\overline{CS2}$ est active pour seize positions mémoires quatre adresses seulement pourront être sélectionnées en prenant $CS0 = CSI = I$; et dans ce cas le boîtier est complètement sélectionné.

Pour les registres internes, le choix s'est porté sur les quatre dernières adresses : $\$ C \emptyset A C$, $\$ C \emptyset A D$, $\$ C \emptyset A E$, $\$ C \emptyset A F$.

Ce choix permet de connecter directement les lignes A3 et A2 du connecteur n°2 aux lignes CSI et CS0.

En effet, comme on peut le remarquer dans le tableau suivant A3 et A2 sont toujours à I, et comme CSI et CS0 sont également à I, il suffit de relier directement A3 à CSI et A2 à CS0. Pour permettre l'adressage des registres internes, RSo et RSI sont connectés à A0 et A1.

TABLEAU REPRESENTANT LES QUATRE PLUS FAIBLES BITS DES ADRESSES DES REGISTRES INTERNES

| A D R E S S E S | A3 | A2 | A1 | A0 |
|-----------------|----|----|----|----|
| C Ø A C | I | I | 0 | 0 |
| C Ø A D | I | I | 0 | I |
| C Ø A E | I | I | I | 0 |
| C Ø A F | I | I | I | I |

LES OPERATIONS DE BASE DU P I A PEUVENT ETRE RESUMEEES PAR LE TABLEAU SUIVANT.

| $\overline{CS2}$ | CS1 | CS0 | RS1 | RS0 | b2 | R/W | TYPE D'OPERATION |
|------------------|-----|-----|-----|-----|----|-----|---------------------------------------|
| 0 | I | I | 0 | 0 | 0 | 0 | Sortie:bus de données--- DDRA |
| 0 | I | I | 0 | 0 | I | 0 | Sortie:bus de données--- Canal A |
| 0 | I | I | 0 | 0 | I | I | Entrée:Canal A--bus de données |
| 0 | I | I | 0 | I | X | 0 | Sortie:bus de données--- CRA. |
| 0 | I | I | 0 | I | X | I | Entrée:CRA---bus de données |
| 0 | I | I | I | 0 | 0 | 0 | Sortie:bus de données---DDR B |
| 0 | I | I | I | 0 | I | 0 | Sortie:bus de données---Canal B |
| 0 | I | I | I | 0 | I | I | Entrée:Canal B---bus de données |
| 0 | I | I | I | I | X | 0 | Sortie:bus de données--- CRB |
| 0 | I | I | I | I | X | I | Entrée: CRB--- bus de données. |

B - 2.5. Programmation du P I A :

Le rôle que doit jouer le P I A est déterminé par programme. La définition de la configuration fonctionnelle se fait en écrivant, dans les positions mémoires attribuées aux registres du P I A, les informations indiquant la fonction que doit réaliser le P I A. Une fois que ce circuit est programmé, il est prêt à assurer l'interface entre le périphérique et le microprocesseur.

Dans le cas de cet interface, les deux ports A et B sont configurés en sortie puisque les données et la commande sont envoyées du microprocesseur au périphérique.

La programmation du P I A se fait en trois étapes :

- On accède aux registres CRA et CRB et l'on y place zéro, ce qui autorise l'accès aux registres DDRA et DDRB.
- On accède aux registres DDRA et DDRB où l'on écrit la configuration en sortie de ces registres.
- On accède de nouveau aux registres CRA et CRB et l'on met le bit b2 de ces registres à 1.

Afin d'adresser les registres de sortie des données :

```
PROGRAMME D'INITIALISATION  
1 . REM INITIALISATION PIA.  
5 . POKE 49325,0  
10 . POKE 49327,0  
15 . POKE 49324,255  
20 . POKE 49326,255  
25 . POKE 49325,4  
30 . POKE 49327,4
```


B - 3. Convertisseur Numérique Analogique :

Le convertisseur numérique analogique reçoit une information numérique sous forme de n bits et la transforme en un signal analogique.

A chacune des 2^n combinaisons possibles à l'entrée, le convertisseur numérique analogique fait correspondre une tension analogique discrète dépendant d'une tension de référence.

Le convertisseur utilisé est le MC 1408 L8 de Motorola.

C'est un convertisseur 8 bits.

B - 3.1. Principales caractéristiques de ce convertisseur :

- Résolution : Elle est donnée par le nombre de bits du convertisseur.

- Précision : C'est l'écart entre le signal de sortie réel et le signal de sortie théorique obtenu à partir de la courbe de transfert.

La précision relative du MC 1408 L8 est égale à $\pm 0,19\%$

- Temps d'établissement : C'est le temps que met la tension de sortie pour passer de 0 Volt à U Volts quand tous les bits d'entrée passent de zéro à un.

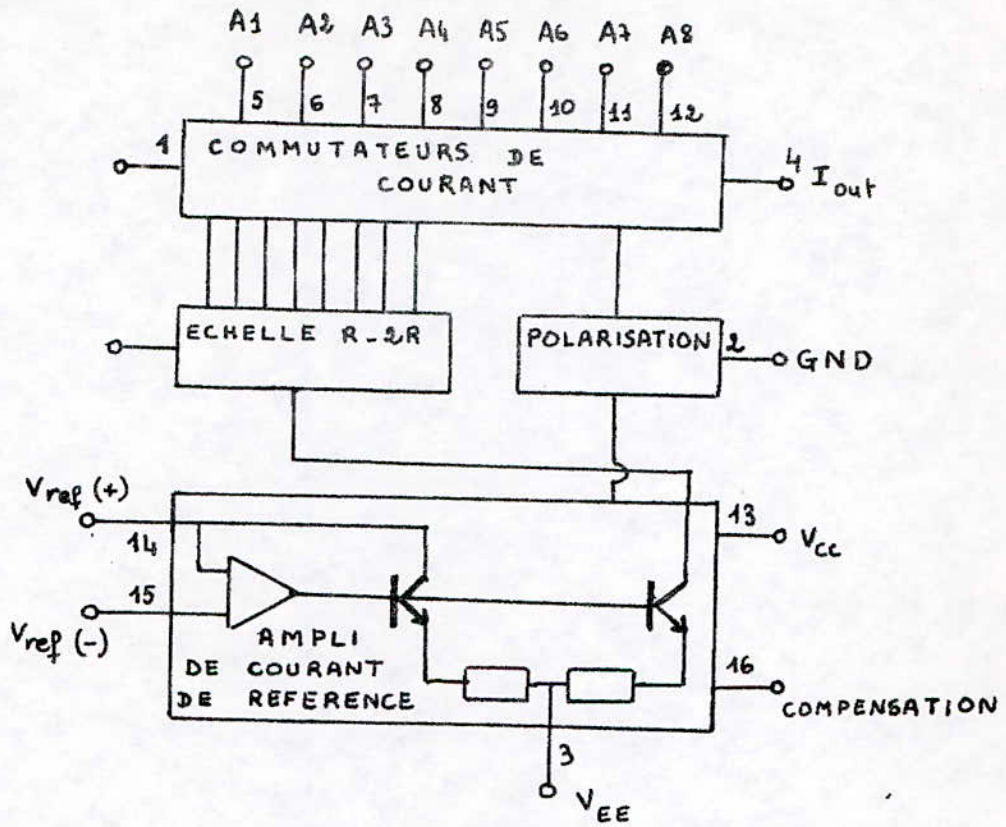
Ce temps est égal à 300 ns.

B - 3.2. Description du Convertisseur :

Le convertisseur délivre à la sortie un courant I_{out} .

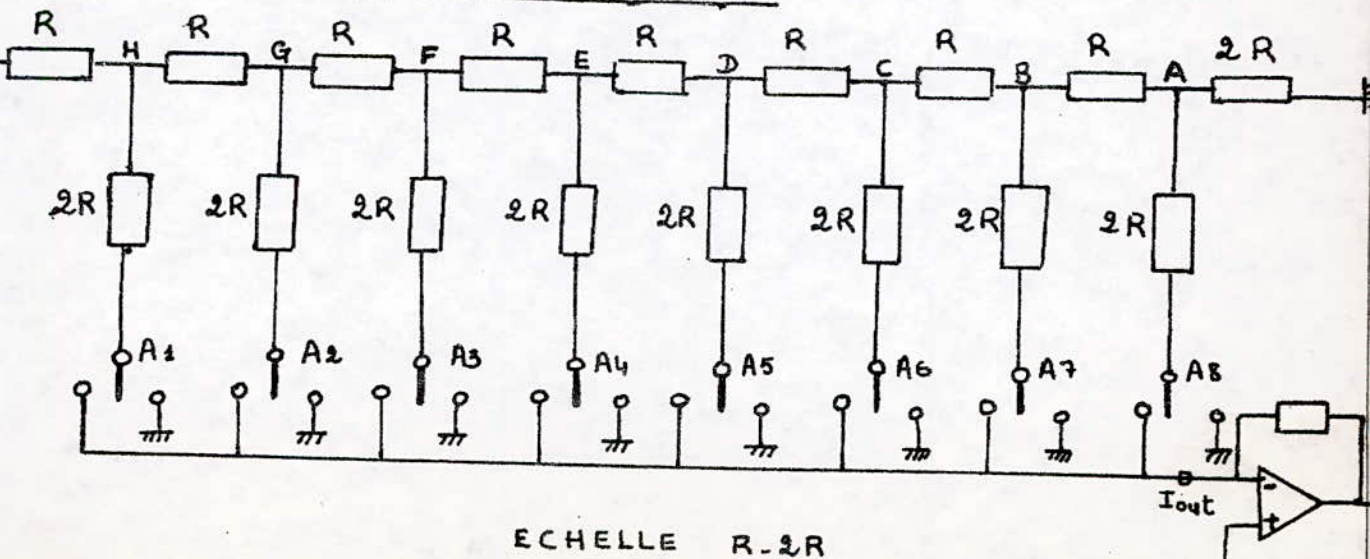
-(37)-

Ce convertisseur est composé :
- d'un amplificateur de courant de référence
- d'une échelle R-2R
- de 8 commutateurs rapides.



Synoptique du MC 1408 L8

Principe de la conversion



Chaque bit du mot d'entrée commande un commutateur connecté vers la masse pour un 0, vers la ligne de sommation pour un 1.

La tension à l'entrée de l'amplificateur opérationnel étant nulle, quelle que soit la position du commutateur, la résistance $2R$ qu'il pilote est reliée à un potentiel nul.

Ainsi si l'on se place au point noté A, et qu'on regarde vers la droite on voit apparaître deux résistances $2R$ en parallèle, l'impédance résultante est donc R . Si on se place à présent au point B en observant toujours la droite du schéma, on voit ici une résistance R en série avec la résistance R trouvée à partir de A soit $2R$ au total, le tout en parallèle sur $2R$: la résultante donne à nouveau R .

Le même raisonnement est valable pour les points C, D, E, F, G et H quel que soit le point où l'on se place on voit apparaître une résistance R qui se trouvera ainsi en série avec une dernière résistance R reliée à une source de référence.

Au point H, on a donc la tension $\frac{V_{ref}}{2}$, on en déduit, en développant le même raisonnement mais de la gauche vers la droite; les différentes tensions aux points G, F, E, D, C, B, et A qui sont respectivement :

$$\frac{V_{ref}}{4}, \frac{V_{ref}}{8}, \frac{V_{ref}}{16}, \frac{V_{ref}}{32}, \frac{V_{ref}}{64}, \frac{V_{ref}}{128} \text{ et } \frac{V_{ref}}{256}$$

Le courant I_{out} délivré à la sortie du convertisseur est égal à la somme des courants circulant dans les résistances $2R$ reliées à la ligne de sommation.

L'expression de ce courant est donc la suivante :

$$I_{out} = \frac{V_{ref}}{2} \frac{A1}{2R} + \frac{V_{ref}}{4} \frac{A2}{2R} + \frac{V_{ref}}{8} \frac{A3}{2R} + \frac{V_{ref}}{16} \frac{A4}{2R} + \frac{V_{ref}}{32} \frac{A5}{2R} + \frac{V_{ref}}{64} \frac{A6}{2R} \\ + \frac{V_{ref}}{128} \frac{A7}{2R} + \frac{V_{ref}}{256} \frac{A8}{2R}$$

A l'aide de l'amplificateur opérationnel, ce courant I_{out} va être converti en tension.

B-4 - Transfert des données et des commandes :

Le bus de données D_0 D_7 du micro-ordinateur permet d'envoyer sur le PIA les données numériques à convertir ainsi que la commande nécessaire pour obtenir une tension de même signe que la température de consigne TC qui lui correspond.

Le calcul de température et son transfert vers le périphérique se fait lorsque le signal R/\bar{W} (Lecture-écriture) est à l'état bas.

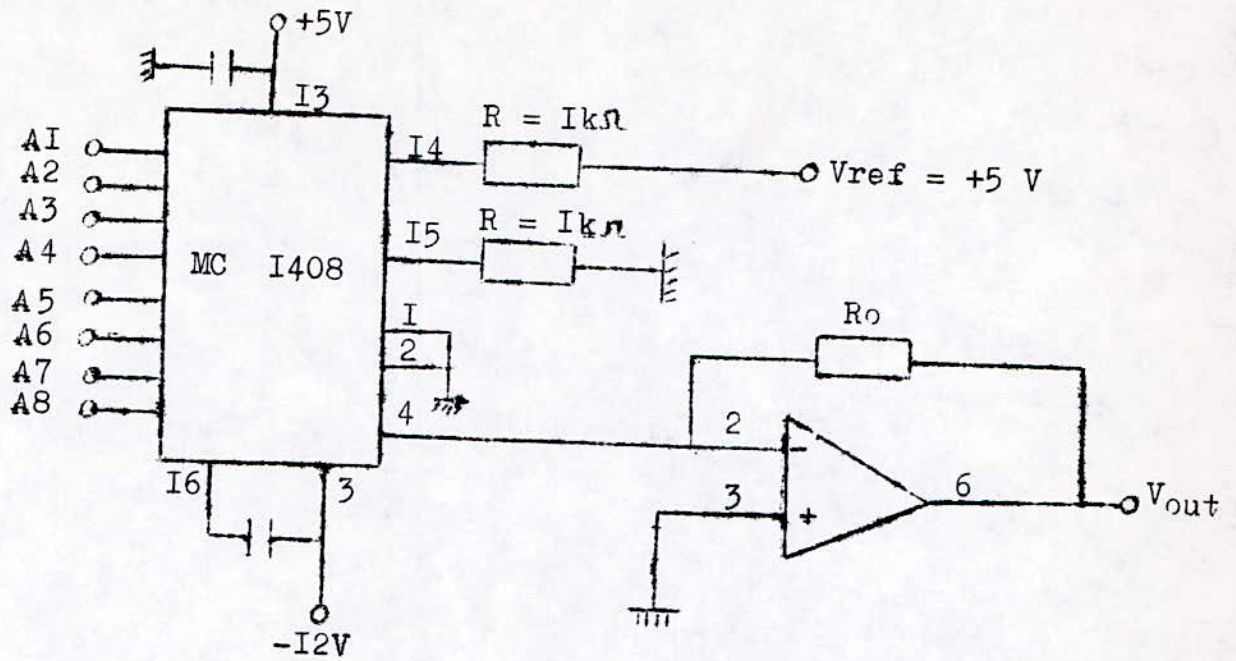
B-5 - Réalisation de l'interface :

Le PIA est directement relié au connecteur, ne nécessitant aucun circuit de décodage.

a)- Conversion en tension de la sortie du convertisseur :

La sortie 4 du convertisseur numérique analogique délivre un courant I_{out} .

La conversion en tension s'effectue grâce au montage suivant :



La tension V_{out} a pour expression :

$$V_{out} = \frac{V_{ref}}{R} R_o \left(\frac{A1}{2} + \frac{A2}{4} + \frac{A3}{8} + \frac{A4}{16} + \frac{A5}{32} + \frac{A6}{64} + \frac{A7}{128} + \frac{A8}{256} \right)$$

-(41)-

La tension maximale est obtenue quand tous les bits sont à un.

$$V_{o \text{ Max}} = \frac{+5}{10} R_o \left(\frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \frac{1}{16} + \frac{1}{32} + \frac{1}{64} + \frac{1}{128} + \frac{1}{256} \right) = 0,0049$$

Comme la température de consigne varie entre -160°C et $+160^{\circ}\text{C}$

la plus grande tension sera celle qui correspond à $T = 160^{\circ}\text{C}$.

La sensibilité du thermomètre numérique étant de $10 \text{ mV}/^{\circ}\text{C}$.

La température $T = +160^{\circ}\text{C}$ doit correspondre à une tension

$$V_{\text{out}} = +1,6 \text{ V.}$$

$$\text{D'où : } V_{o \text{ Max}} = 0,0049 R_o = 1,6$$

$$\text{On a donc : } R_o = 321,22$$

La tension V_{out} délivrée par le $\mu\text{A} 741$ est toujours positive.

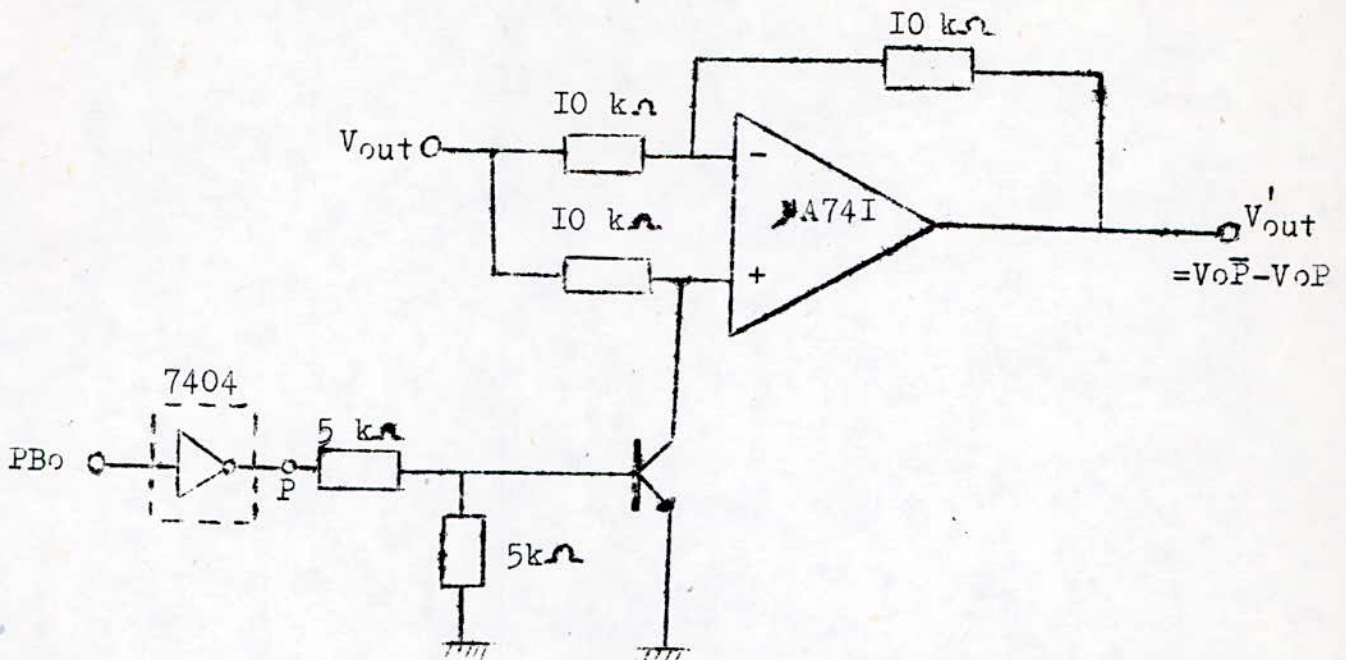
Or la température de consigne varie entre -160°C et $+160^{\circ}\text{C}$,

et aux températures négatives doivent correspondre des tensions négatives.

Il faut donc inclure dans le montage précédent un circuit qui permette d'obtenir une tension de même signe que la température de consigne qui lui est relative.

Le montage suivant réalise cette fonction.

Un transistor NPN BC 108 commutateur est commandé par un bit PBo du PIA servant à définir la polarité de la tension de sortie.



SCHEMA DE PRINCIPE

Fonctionnement de ce montage :

Suivant l'état bloqué ou saturé du transistor, la tension V'_{out} sera positive ou négative.

Le blocage ou la saturation du transistor dépendra de l'état de la ligne de commande PBo .

- Lorsque $PBo = 1$, donc $P = 0$; le transistor est bloqué.

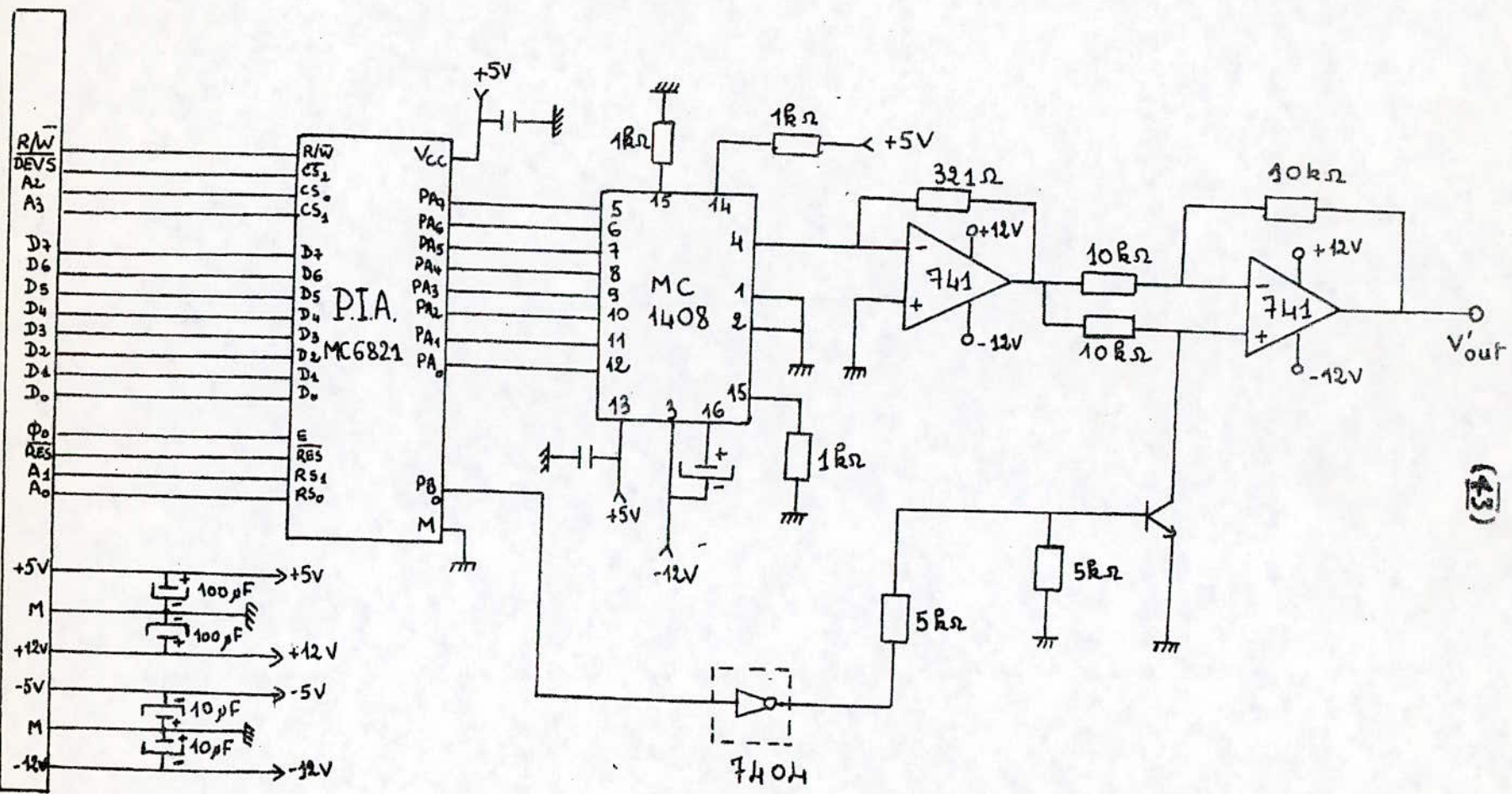
On aura pour expression de V'_{out}

$$V_o = \frac{10}{10} V_{out} + (1 + \frac{10}{10}) V_{out} = V_{out}$$

- Lorsque $PBo = 0$, le transistor est saturé, la tension VCE est négligeable, la borne + de l'amplificateur est donc mise à la masse; d'où : $V'_{out} = -\frac{10}{10} V_{out} = -V_{out}$.

Remarque : On utilise un inverseur de type 7404 afin de protéger la ligne PBo contre un éventuel retour de courant important qui risque d'endommager le PIA.

Connecteur
de
peripherique



SCHEMA ELECTRIQUE DE L'INTERFACE

B-6 - Gestion de l'interface :

Chaque fois que le microordinateur calcule une température, il l'affiche sur l'écran et l'envoie simultanément sous forme de tension analogique au comparateur.

Après avoir élaboré le programme qui permet la simulation de la température, il faut y insérer :

- le sous programme d'initialisation du PIA
- le sous programme qui permet l'envoi de la température au comparateur sous forme de tension analogique, chaque fois que l'instruction PRINT TC est utilisée.

La gestion de l'interface nécessite également un changement d'échelle pour le convertisseur numérique analogique.

Il faut donc élaborer un sous programme qui puisse réaliser cette fonction.

Le convertisseur numérique analogique a une résolution de 8 bits. Donc la plus grande valeur qu'il puisse accepter sera égale, en décimal, à 255.

La plus grande valeur de la température est égale à $T_0 = 160^{\circ}\text{C}$. Le changement d'échelle doit pouvoir faire correspondre $T_0 = \pm 160^{\circ}\text{C}$ à la valeur 255 pour pouvoir bénéficier de la meilleure résolution.

Le changement d'échelle va être le suivant :

Soit TA : valeur absolue de la température de consigne TC et

soit TI : valeur entière de l'expression $\frac{TA \times 255}{160}$.

Ce sera la valeur de TI qui sera transférée vers le périphérique.

A chaque période d'échantillonnage, le microordinateur calcule TC et TI.

Avant d'envoyer TI sur l'interface, le microordinateur effectue un test sur la température de consigne TC.

Si TC est négative le bit PBo de commande est mis à zéro, ce qui va engendrer une tension négative à la sortie du deuxième amplificateur opérationnel, après que TI soit envoyée sur l'interface.

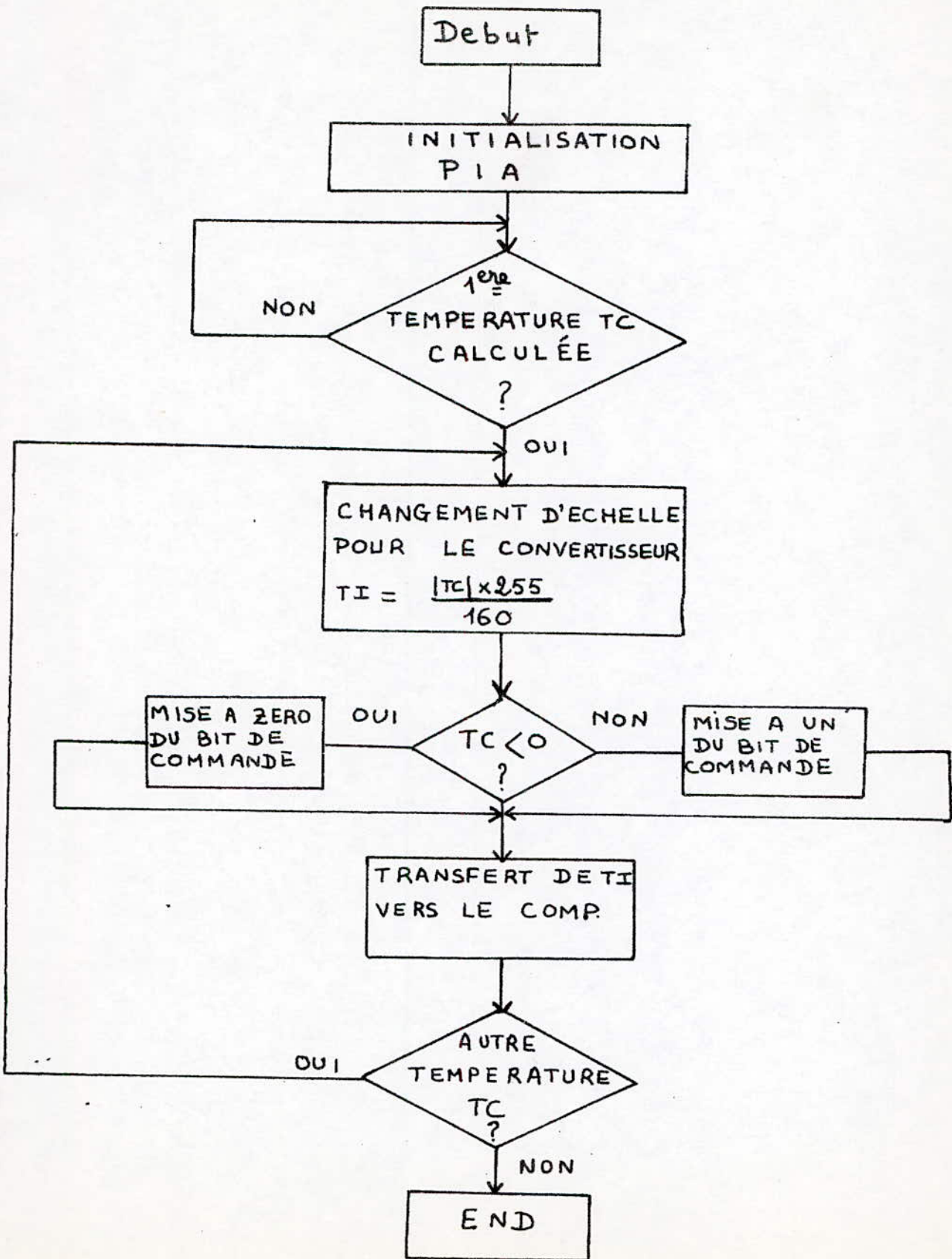
Si TC est positive, le bit PBo sera mis à un, et à la sortie du deuxième amplificateur on recueillera une tension positive.

L'insertion des différents sous-programmes nécessaires à la gestion de l'interface se fait suivant l'organigramme donné ci-contre.

B - 7 - Utilisation du Programmeur :

Considérons un échantillon diélectrique dont les caractéristiques sont les suivantes :

-(46)-



- Durée de palier bas : 1 minutes
 - Pente des droites croissantes ou pente positive : $3^{\circ}\text{C}/\text{mn}$
 - Pente des droites décroissantes ou pente négative : $-20^{\circ}\text{C}/\text{mn}$
 - Température de polarisation $T_p = 0^{\circ}\text{C}$.
 - Le spectre global donne 1 pic à la valeur $T_M = -40^{\circ}\text{C}$.
- On se propose donc de détecter les pics élémentaires compris entre -50°C et -30°C .

L'utilisation du programmeur nécessite de charger à partir d'une disquette le programme donné ci-dessous :

```

5 REM PROGRAMME DE GESTION DE L'INTERFACE
10 REM INITIALISATION
15 GOSUB 500
20 DATA 3, -20, 2, 1, 0.1
25 READ DP, DN, PH, PB, TE
30 TC = 0
35 T = 0
40 TF = 0
45 IP = 1
50 GOSUB 2000
55 L = 360
60 FOR I = 1 TO L: NEXT I
65 GOSUB 1000
70 GOTO 60
500 REM SOUS PROGRAMME INITIALISATION PIA
505 POKE 49325,0 : POKE 49327,0

```

} SOUS PROGRAMME
DE GESTION

```
510 POKE 49324,255 : POKE 49326,255 }
515 POKE 49325,4 : POKE 49327,4 } SOUS PROGRAMME
520 RETURN } DE GESTION

1000 DATA -160, -40, -50, -160, -30, -40, -160, 150, 160
1030 TC = TC + DC
1040 T = T + TE
1050 IF DC = 0 THEN 1110
1060 IF DC > 0 THEN 1090
1070 IF TC > TI THEN 1130
1080 GOTO 2000
1090 IF TC < TI THEN 1130
1100 GOTO 2000
1110 IF T < TF THEN 1130
1120 GOTO 2000
1130 REM OUT PUT
1140 TA = ABS (TC) }
1150 TI = INT ( $\frac{TA * 255}{160}$ ) } SOUS PROGRAMME
1160 PRINT TC, TI } DE GESTION
1170 IF TC < 0 THEN 1200 }
1180 POKE 49326,I : POKE 49324, TI } SOUS PROGRAMME
1190 GOTO 1210 } DE GESTION
1200 POKE 49326,0 : POKE 49324, TI }
1210 RETURN }
2000 REM CHANGEMENT DE PENTE
2010 IP = I - IP.
```



```
2020 IF IP = 0 THEN 2140
2030 READ TI
2040 IF TI = I60 THEN STOP
2050 TT = TI - TC
2060 IF TT > 0 THEN 2100
2070 TT = TT/DN
2080 DC = TE * DN
2090 GOTO 2120
2100 TT = TT/DP
2110 DC = TE * DP
2120 TF = TF + TT
2130 GOTO 1000
2140 IF DC < 0 THEN 2170
2150 TF = TF + PH
2160 GOTO 2180
2170 TF = TF + PB
2180 TC = TI
2190 DC = 0
2200 GOTO 1000
```

Le programme précédent doit, dans sa conception, pouvoir s'appliquer à l'étude de tout autre échantillon diélectrique. Il suffit, pour passer de l'étude d'un échantillon à un autre, de modifier les trois instructions suivantes :

- L'instruction 30 qui donne la première valeur de la température de consigne, c'est à dire la température de polarisation.

A chaque échantillon correspond une température de polarisation.

Donc à chaque fois que l'utilisateur entreprend d'étudier un nouvel échantillon, il doit retaper l'instruction 30 en introduisant la température de polarisation relative à ce nouvel échantillon.

- L'instruction I000 : Elle contient les valeurs des différents paliers de température. Ces valeurs se déduisent du spectre global. Ce dernier, comme on l'a déjà signalé, est obtenu lors de la première remontée linéaire de la température.

Sur ce spectre, on repère un pic se situant à une température T_M et les différents paliers seront :

$T_M - \Delta T$, $T_M - \Delta T + 10$, -160 , $T_M - \Delta T + 20$, $T_M - \Delta T + 10$, -160
....., $T_M + \Delta T - 10$, $T_M + \Delta T$, -160 , 150 , 160

La valeur de ΔT est laissée à l'appréciation de l'opérateur suivant qu'il veut ou non prendre une zone assez étendue autour de T_M .

- L'instruction 20 : Cette instruction comporte les valeurs des pente des droites croissantes, pente des droites décroissantes durée des paliers hauts, durée des paliers bas et période d'échantillonnage. Seule cette dernière valeur reste la même pour tous les échantillons.

Les autres valeurs DP, DN, PB et PH sont introduites par l'utilisateur à chaque fois qu'il change d'échantillon.

Précision de la tension de consigne :

Théoriquement à chaque température de consigne doit correspondre une tension V'_{out} tel que $V'_{out} (mv) = 10 TC$.

$$\text{Or } TI = \frac{/TC/ \times 255}{160} \quad \text{d'où : } /TC/ = \frac{160}{255} TI$$

TI en binaire s'écrit $A_1 A_2 A_3 A_4 A_5 A_6 A_7 A_8$

En décimal : $TI = A_8 + 2A_7 + 4A_6 + 8A_5 + 16A_4 + 32A_3 + 64A_2 + 128A_1$

$$\text{D'où } /TC/ = \frac{160}{255} TI = \frac{160}{255} (A_8 + 2A_7 + 4A_6 + 8A_5 + 16A_4 + 32A_3 + 64A_2 + 128A_1)$$

La tension qui lui équivaut est donc :

$$(V'_{out})_{th} = \frac{1600}{255} (A_8 + 2A_7 + 4A_6 + 8A_5 + 16A_4 + 32A_3 + 64A_2 + 128A_1)$$

La tension réelle recueillie à la sortie de l'interface est :

$$V'_{out}) \text{ réel} = V_{ref} \frac{R_0}{R} \left(\frac{A_1}{2} + \frac{A_2}{4} + \frac{A_3}{8} + \frac{A_4}{16} + \frac{A_5}{32} + \frac{A_6}{64} + \frac{A_7}{128} + \frac{A_8}{256} \right)$$

L'erreur est donc égale à : $(V'_{out}) \text{ théorique} - (V'_{out}) \text{ réel}$.

Elle est maximale quand tous les bits sont à un; et dans ce cas elle est égale à 6,25 mV; c'est à dire que l'erreur maximale sur la température de consigne est égale à 0,625°C.

B-8- Autres possibilités de concevoir le programmeur :

D'autres solutions existent pour la conception du programmeur

On va les citer brièvement :

-On peut enregistrer sur un magnétophone le signal correspondant aux variations de la température de consigne en fonction du temps.

Ce signal sera envoyé par la suite au comparateur grâce à une carte qui assurera la conversion fréquence - tension.

-On peut également enregistrer le signal sur une bande de papier.

Une "tête photoélectrique" suit les variations du signal sur la bande de papier et entraîne dans son mouvement le curseur d'un potentiomètre qui lui est relié. La tension aux bornes du potentiomètre sera envoyée au comparateur.

Ces solutions offrent l'avantage d'être simples.

Néanmoins la deuxième solution présente quelques inconvénients.

La tête photoélectrique doit suivre le signal de la température en "temps réel"; c'est à dire que si la durée du signal vaut la valeur T, la tête photoélectrique doit parcourir ce signal en un intervalle de temps égal à T. Ce réglage est assez difficile à obtenir.

Cette cause agit sur la précision de la consigne et la rend mauvaise.

Une autre solution, plus intéressante est à envisager. C'est celle qui permet, au micro-ordinateur non seulement d'assurer la fonction du programmeur mais également de contrôler tout le processus.

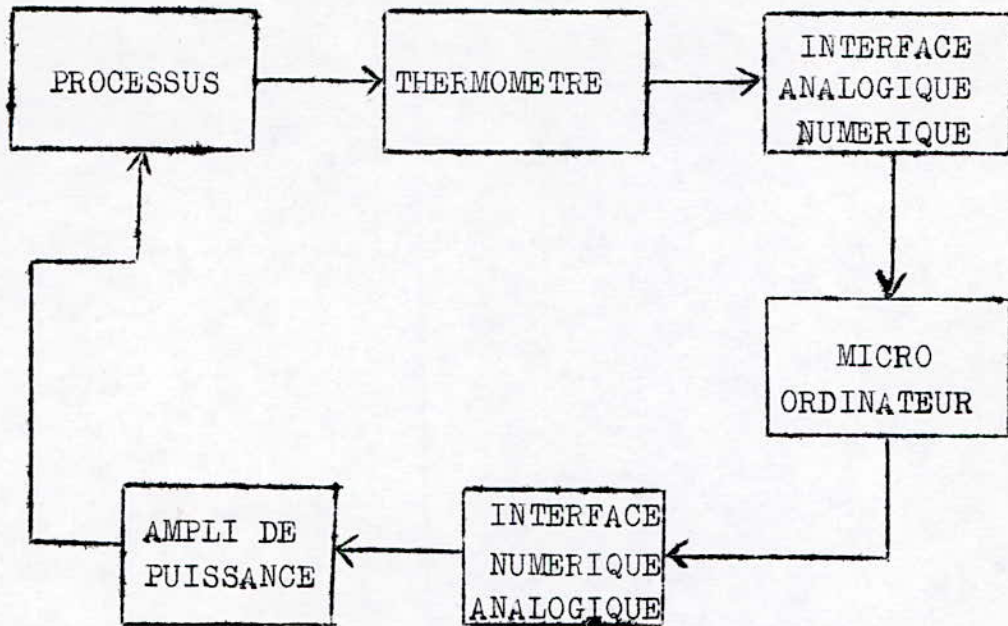
B-9- Contrôle du Processus par le Micro-ordinateur :

L'emploi de micro-ordinateur dans le contrôle des processus tend de plus en plus à s'accroître.

En effet, ceci permet :

- d'améliorer les performances de l'automatisation grâce à la souplesse du micro-ordinateur et de sa puissance.
- de développer le contrôle optimal qui est pratiquement impossible à réaliser avec un régulateur analogique.

La solution du contrôle du processus peut être résumée par le synoptique suivant :



Le micro-ordinateur joue le rôle de :

- Programmateur
- Régulateur
- Comparateur.

Cette solution, en pratique, se réalise à l'aide de deux cartes :

- Une carte conversion analogique-numérique
- Une carte conversion numérique-analogique

Le micro-ordinateur, associée à ces deux interfaces permet d'acquérir la mesure M, de calculer la consigne C, de calculer $\xi = C - M$, et de créer cette commande.

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

L'étude de ce programmeur présente une solution suffisante pour le transfert des variations de la température sous forme de tensions analogiques.

Cependant certaines améliorations peuvent être apportées :

* Du point de vue logiciel, le programme de simulation a été conçu en langage basic (à cause de l'utilisation de multiplications).

Ce langage, même s'il permet l'étude du phénomène, limite quelque peu les performances dynamiques.

En effet, le calcul en basic étant lent, la période d'échantillonnage doit être suffisamment grande par rapport à la durée d'interprétation des instructions.

La traduction du programme de simulation en langage machine éliminerait cette durée d'interprétation et permettrait donc de réaliser un échantillonnage rapide.

* Du point de vue matériel : - l'utilisation de convertisseurs à plus grande résolution donnerait de meilleurs résultats.

Les convertisseurs 12 bits ou 16 bits permettraient d'augmenter considérablement la précision de consigne.

- L'utilisation d'un autre transistor à plus faible tension de saturation que le BC 108 permettrait d'avoir dans le cas où la température est négative, une tension de consigne plus précise.

Le contrôle du processus par micro-ordinateur est une solution très intéressante vu les performances qu'elle réalise. Il serait intéressant dans l'avenir d'étudier ce cas afin d'améliorer les performances de l'automatisation.

- B I B L I O G R A P H I E -
=====

- La Pratique de l'Apple II.....Nicole Bréaud-Pouliquen
- Thèse de Magister
Spécialité: Matériaux Diélectriques...Azzedine Gourari.
- Les Microprocesseurs à l'usage des
Electroniciens.....J.P. Coquerez.
- Au Coeur des MicroprocesseursD.Girod et R. Dubois.
- N.A et A.N Convertisseurs.....R. Fontenay.
- Interface des Microprocesseurs.....R. Robin et Th. Maurin.
- Techniques d'Interface.....R. Zaks.