

25/88

الجمهورية الجزائرية الديمقراطية الشعبية  
REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

وزارة التعليم و البحث العلمي  
Ministère de l'Enseignement et de la Recherche Scientifique

ECOLE NATIONALE POLYTECHNIQUE

Département : Génie Mécanique

Projet de Fin d'Etudes

1 ex

THÈME

Etude d'un Module d'Application  
de Microcalculateur : MIC 955 pour Thermostat.

Proposé par :

M<sup>me</sup> E. ROBL

Etudié par :

M<sup>r</sup> KARIM ARSOULI

Dirigé par :

M<sup>me</sup> E. ROBL

PROMOTION Juin 1988



الجمهورية الجزائرية الديمقراطية الشعبية  
REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

وزارة التعليم و البحث العلمي

Ministère de l'Enseignement et de la Recherche Scientifique

ECOLE NATIONALE POLYTECHNIQUE

**Département : Génie Mécanique**

المدرسة الوطنية المتعددة التقنيات  
BIBLIOTHEQUE — المكتبة  
Ecole Nationale Polytechnique

Projet de Fin d'Etudes

THÈME

Etude d'un Module d'Application  
de Microcalculateur : MIC 955 pour Thermostat.

Proposé par :

M<sup>me</sup> E. ROBL

Etudié par :

M<sup>r</sup> KARIM ARSOULI

Dirigé par :

M<sup>me</sup> E. ROBL

PROMOTION Juin 1988

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ  
اقْرَأْ بِاسْمِ رَبِّكَ  
الَّذِي خَلَقَ خَلَقَ  
الْإِنْسَانَ  
الَّذِي أَحْسَنَ  
عَلَّمَ  
بِالْقَلَمِ

عَلَّمَ الْإِنْسَانَ  
مَا كَانَتْ يَدُهُ  
مُسْمُوعًا

صَدَقَ اللَّهُ الْعَظِيمُ

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

المدرسة الوطنية المتعددة التقنيات  
المكتبة — BIBLIOTHEQUE  
Ecole Nationale Polytechnique

Au terme de ce modeste travail, je tiens à exprimer mes remerciements à tous les enseignants de l'école nationale polytechnique qui ont contribué à ma formation et en particulier les professeurs du département Génie Mécanique .Je réitère ma gratitude et ma reconnaissance à Mme E.ROBL pour avoir constamment suivi ce travail; pour les conseils judicieux qu'elle m'a prodigués et pour le concours qu'elle m'a prêté. Mes remerciements s'adressent également à Mr ZERGUERAS et à Mr MADANI pour l'honneur qu'ils ont bien voulu me faire par leur présence. Que tous ceux qui ont contribué de près ou de loin à la réalisation de ce projet trouveront ici l'expression de ma profonde gratitude.

-( D E D I C A C E S )-

A

المدرسة الوطنية المتعددة التقنيات  
المكتبة — BIBLIOTHEQUE  
Ecole Nationale Polytechnique

- MES PARENTS,
- MES FRERES ET SOEURS,
- TOUTE MA FAMILLE,
- TOUS MES AMIS,
- MON CHER AMI MOHAMED ET SA FAMILLE,
- LA PROMOTION "JUIN 88"

JE DEDIE CE MODESTE TRAVAIL.

Ministère de l'Enseignement Supérieur

Ecole Nationale Polytechnique

Département: Génie Mécanique

Promotrice: Mme E. ROBL

Elève Ingénieur: Mr ARSOULI KARIM



وزارة التعليم العالي

المدرسة الوطنية المتعددة التقنيات

مصلحة المندسة الميكانيكية

الموجهة: السيدة روبل

الطالب المهندس: السيد أرسولي كريم

الموضوع : دراسة ملحق تطبيق لحاسبة مصغرة من طراز :  
MIC 955 مستعمل كمنظم لدرجة الحرارة .  
المهندس :

يتمثل هذا المشروع في دراسة ملحق تطبيق من  
سلسلة "Feedback" MIC 940 المستعمل في تطبيقات  
الحاسبة المصغرة والذي يستهدف التحكم في  
تغيرات درجة الحرارة .

Sujet: Etude d'un module d'application de microcalculateur :  
MIC 955 pour Thermostat.

RESUME: Ce projet consiste à étudier un module d'application de  
la série MICA "FEEDBACK" qui est conçu pour des applica-  
tions d'utilisation de microcalculateur portant sur la  
commande de température .

Subject: A study of a microcomputer application module:  
MIC 955 FOR THERMOSTAT.

Abstract: This subject relates to the study of the  
FEEDBACK MICA series application module which is  
designed for practical use of a microcomputer for  
Temperature Control.

ETUDE DU MODULE D'APPLICATION D'UN MICRO CALCULATEUR :  
LE MIC 955 POUR THERMOSTAT



-oOo- S O M M A I R E -oOo-

INTRODUCTION	1
I/ GENERALITES ET DEFINITIONS	3
1 - Ordinateur	3
2 - Miniordinateur	4
3 - Micro-ordinateur	4
4 - Microprocesseur	5
II/ SYSTEMES NUMERIQUES, CODAGE, DECODAGE ET TRANSCODAGE	6
III/ ELEMENTS NUMERIQUES DE BASE	13
1 - Fonctions logiques : Algèbre de BOOLE	13
2 - Réalisation des fonctions logiques à l'aide des P. logiques	15
3 - Circuits des codeurs, des décodeurs, des transcodeurs et afficheurs à sept segments.	16
4 - Multiplexeurs et démultiplexeurs	21
IV/ LES DIFFERENTS LANGAGE DE PROGRAMMATION	23
V/ LE MICROPROCESSEUR INTEL 8085 ET SA PROGRAMMATION	25
VI/ MESURE NUMERIQUE DE TEMPERATURE	29
1 - Introduction et généralités	
a) Méthodes optiques	
b) Méthodes mécaniques	
c) Méthodes électriques	
2 - Théorie et différents types de capteurs	29
a) Capteurs tout-ou-rien	
b) Capteurs analogiques	
3 - Etude des capteurs de température	30
A-Thermométrie par thermocouples	
B-Thermométrie par résistances	34



a)Sensibilité thermique et critère de choix d'un métal	34
b)Les résistances thermométriques en platine	35
c)Méthode de mesure	37
d)Amplification	38
e)C.A.N. et affichage	39
VII/ ETUDE ET MISE EN MARCHÉ DU BANC D'ESSAI	42
A - Le module d'application : MIC 955	42
B - Le micro : MAT 385	45
C - L'interface : MIC 948	48
D - Travaux pratiques	49
VIII/ C O N C L U S I O N	59
ANNEXE	



Fig 1	Chaine de mesure	2
Fig (2)	Bloc diagramme d'un ordinateur	4
Fig (3)	Bloc diagramme d'un micro-ordinateur	5
Fig (4)	Structure de base d'un microprocesseur	26
Fig (5)	Repérage des broches d'un microprocesseur	26
Fig (6)	Les registres du microprocesseur 8085	27
Fig (7)	Thermométrie par thermocouple	31
Fig (8)	Variation thermique de la f.e.m de différents types de thermocouples	33
Fig (9)	Courbe résistance-température d'une sonde thermométrique en platine	37
Fig (10)	Méthode de mesure de la résistance thermosensible en platine	37
Fig (11)	Multipléxage des voies après amplification des signaux	38
Fig (12)	Multipléxage des voies préalables à l'amplification des signaux	38
Fig (13)	Schéma de principe du convertisseur à approximation successives	41
Fig (14)	MIC 955:Commande de température	44
Fig (15)	Structure de la mémoire du MAT 385	46
Fig (16)	Description du MAT 385	47
Fig (17)	Banc d'essai	48
Fig (18)	Caractéristique de la régulation proportionnelle	52

## Présentation du sujet :

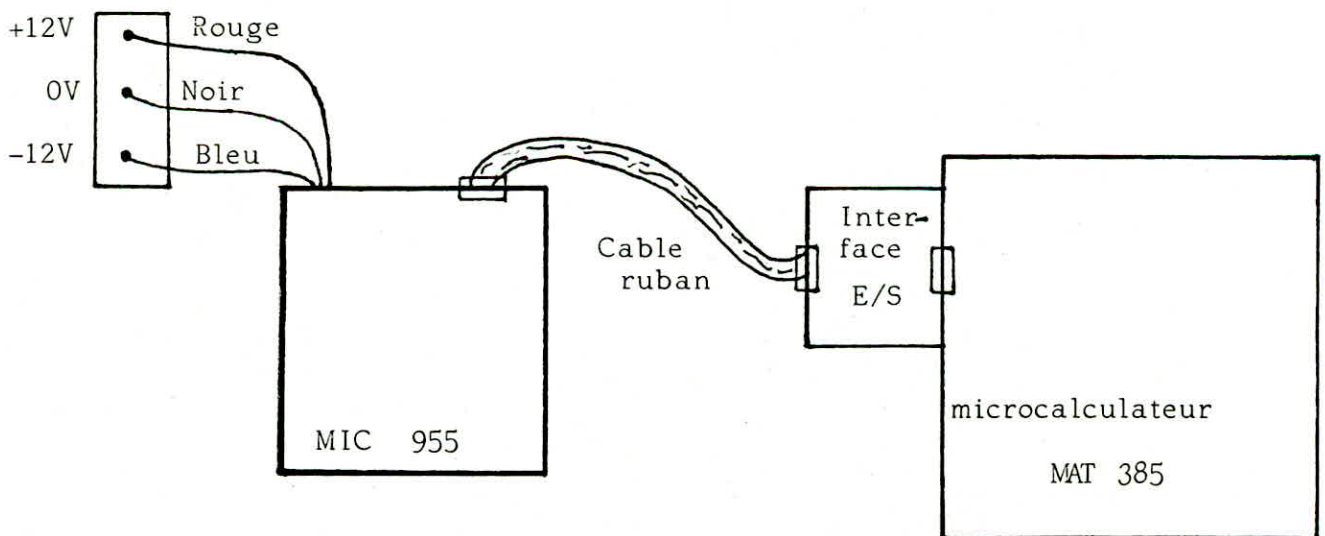
Dans ce présent sujet du projet de fin d'études, le but de notre travail est d'enrichir pédagogiquement notre école, en particulier le département Génie Mécanique.

En effet, avec l'existence des autres bancs d'essais tels que le simulateur de processus PCS327 et le simulateur de processus thermique PT 326, dont l'étude a été faite par des étudiants de la promotion précédente, les applications proposées à travers notre étude seront très bénéfiques et d'un grand intérêt à la formation des élèves ingénieurs, dans le cadre des travaux pratiques du module : Régulation et Systèmes Asservis.

L'objectif de notre travail consiste à faire l'étude du module d'application MIC 955 pour thermostat qui est commandé par le biais du microcalculateur MAT 385.

Le banc d'essai comporte essentiellement :

- \*) Le module d'application : MIC 955,
- \*) Le microcalculateur : MAT 385,
- \*) L'interface : MIC 948,
- \*) L'appareillage de l'alimentation stabilisée.



# **INTRODUCTION**

## INTRODUCTION :

Habituellement le micro-ordinateur communique avec le monde extérieur par son écran et son clavier. D'où son emploi pour le calcul, le traitement des données ou les jeux ne demande pas beaucoup de connaissances de la façon dont il commande et contrôle d'autres matériels.

Cependant, s'il est possible de dôtter le micro-ordinateur d'un certain nombre d'accessoires, le nombre d'applications envisageables est grand.

Donc le micro-ordinateur surveille, mesure, vérifie, ... puis, en fonction des informations que lui ont fournies ses accessoires, il agit, contrôle, modifie, ...

De ce fait, la nouvelle génération de technologie doit comprendre et utiliser des micro-ordinateurs pour la commande d'équipements en tous genres, tels que : machines-outils, robots, avions et beaucoup d'autres systèmes complexes. Leurs compétences doivent inclure une compréhension de l'établissement de modèles de procédés en terme de logiciel, des concepts de programmation en temps réel et des méthodes de détection et de rectification des conditions de défaut.

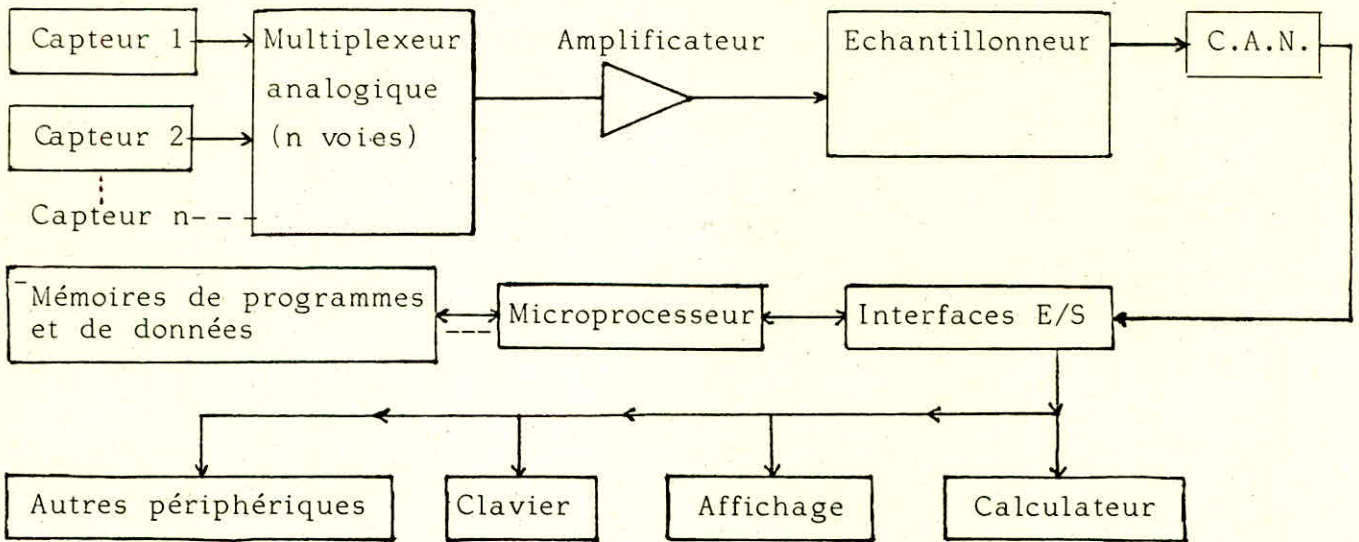
Les micro-ordinateurs actuels peuvent être utilisés pour étudier les interactions des dispositifs extérieurs et du micro-ordinateur et ces compétences peuvent à présent être enseignées et acquises grâce à une formation pratique avec un micro-ordinateur tel que : MAT 385 et les interfaces MIC 940, ainsi qu'avec un modèle des modules d'applications de la série MIC 950, tel que le MIC 955 pour thermostat.

\*La chaîne de mesure : Elle est constituée de l'ensemble des dispositifs, y compris le capteur, rendant possible dans les meilleures conditions la détermination précise de la valeur du mesurande (dans notre cas : la température).

Cependant les conditions pratiques de mesure telles qu'elles sont imposées par l'environnement d'une part et par les performances exigées pour une exploitation satisfaisante d'autre part, amènent à introduire dans la chaîne des blocs fonctionnels destinés à optimiser l'acquisition et le traitement du signal issu du capteur :

- .Circuit de linéarisation du signal délivré par le capteur,
- .Amplificateur d'instrumentation ou d'isolement destiné à réduire les tensions parasites,
- .Multiplexeur, amplificateur d'instrumentation programmable, échantillonneur, convertisseur A-N, ...

Lorsque l'information doit être traitée par ordinateur (notre cas), le schéma synoptique de la chaîne de mesure sera comme suit :



fig[1] \*\* CHAINE DE MESURE \*\*

Avant d'aborder directement l'étude du banc d'essai, certains rappels sont nécessaires.

# CHAPITRE I

## I/ GENERALITES

Un ordinateur est une machine automatique capable d'exécuter des opérations de traitement d'information, sous contrôle de séquences d'instructions préalablement fournies. Une séquence complète d'instructions est appelée un programme.

### Organisation d'un ordinateur :

La figure [2] représente les principaux sous-ensembles d'un ordinateur. Cette organisation des éléments fonctionnels est bien souvent dite "architecture" de l'ordinateur.

Le système le plus élémentaire comprend les unités suivantes :

1° Une unité centrale de traitement (ou C.P.U.: Central Processing Unit)

Très souvent on l'appelle simplement Unité Centrale. C'est l'"intelligence" de l'ordinateur. C'est elle qui exécute les opérations de traitements (arithmétiques et logiques), et gère l'utilisation des autres organes de l'ordinateur (mémoires, ...).

Pour cela, elle opère au rythme d'une horloge, suivant un cycle de travail qui se décompose comme suit:

\*Lire en mémoire une instruction à une adresse donnée, et la placer dans un "registre d'instruction".

\*Décoder l'instruction ainsi placée dans le "registre d'instruction", et exécuter l'ordre qu'elle commande.

\*Puis recommencer le même travail avec l'instruction suivante en mémoire, et ainsi de suite jusqu'à la dernière instruction du programme à réaliser.

2° Des mémoires permettant de stocker des informations:

\*Mémoires de programmes contenant les séquences d'instructions à réaliser;

\*Mémoires de données contenant les données en cours de traitement.

3° Des périphériques permettant les échanges bilatéraux entre ordinateur et utilisateur.

\*L'utilisateur fournit à l'ordinateur les instructions à exécuter, ainsi que les données à traiter. Il peut fournir également des ordres de contrôle de fonctionnement.

\*L'ordinateur fournit aux périphériques les résultats de traitements, ainsi que différents signaux de contrôles.

4° Un (ou plusieurs) bus, lignes de liaisons, véhiculant les informations entre les différentes parties de l'ordinateur.



5° Des interfaces d'Entrées/Sorties, organes d'adaptation de la logique interne de l'ordinateur aux logiques des différents périphériques.

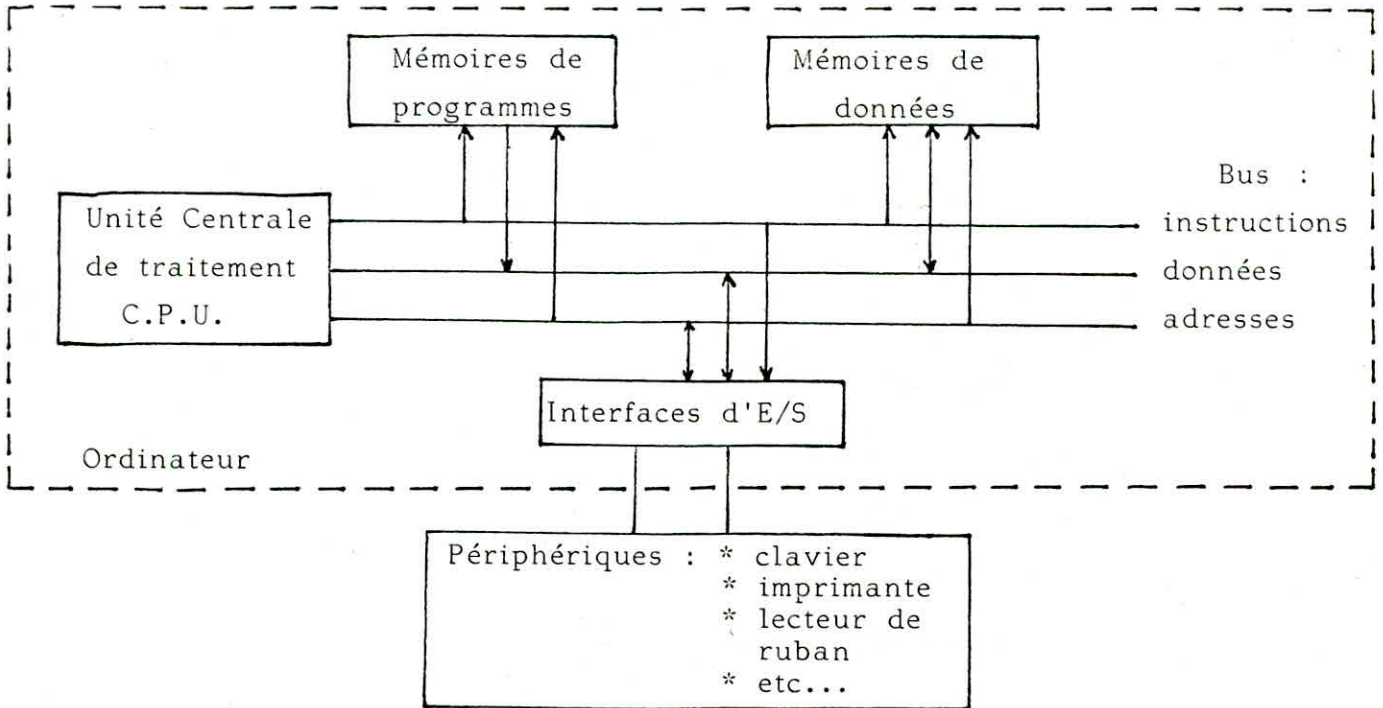


fig [2] \*\* BLOC DIAGRAMME D'UN ORDINATEUR \*\* Réf (425)

Le mini-ordinateur n'est qu'un ordinateur de taille et de prix plus réduits, grâce à l'évolution de la technologie des semi-conducteurs, qui a permis l'utilisation des circuits intégrés en M.S.I. (Medium Scale Integration) à la place d'assemblage de composants discrets.

Depuis les premiers mini-ordinateurs (Années 60), assez peu performants, l'évolution a permis d'avoir actuellement des mini-ordinateurs rejoignant en performances (principalement en capacité de traitement) le bas de la gamme des gros ordinateurs.

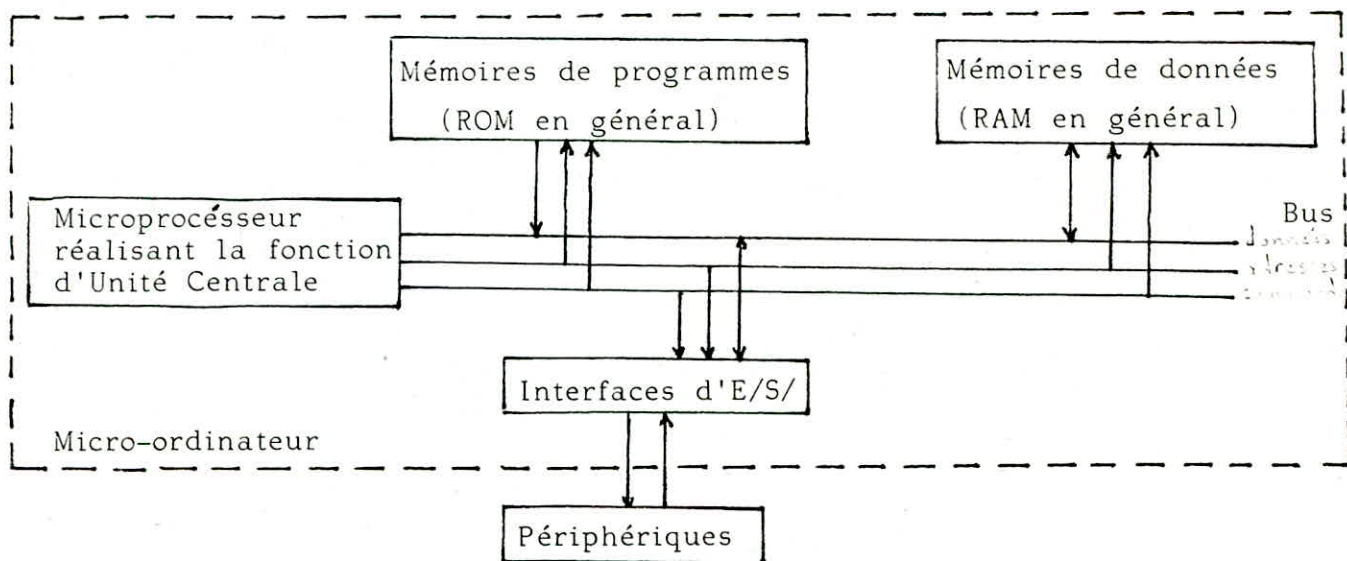
Une intégration toujours plus poussée a amené l'apparition (Années 70) des circuits intégrés en L.S.I. (Large Scale Integration) ouvrant ainsi la porte à la génération des micro-ordinateurs.

le micro-ordinateur est un ordinateur dont les principaux organes constitutifs sont réalisés en circuit L.S.I. (Voir fig. 3) :

\*Un circuit L.S.I. particulier appelé dans ce cas microprocesseur, constitue l'Unité Centrale;

\*Des RAM et des ROM constituent les mémoires. (\*\*)

\*Des circuits, initialement réalisés en logique faiblement intégrée, puis progressivement réalisés en L.S.I., constituent les interfaces d'Entrées/Sorties.



\*\* BLOC DIAGRAMME D'UN MICRO-ORDINATEUR \*\*

Réf (445)  
Fig.: [3]

Le microprocesseur est un composant L.S.I. réalisant les fonctions d'Unité Centrale d'ordinateur.

Sous contrôle de programmes lus en mémoires par l'intermédiaire des lignes de contrôle ou de commande, et au rythme des impulsions d'une horloge, il est capable de réaliser un certain repertoire d'opérations (opérations arithmétiques et logiques, contrôles des mémoires et des organes d'E/S).

(\*\*) Voir les définitions des différents types de mémoires.

**CHAPITRE**

**II**

## II/ SYSTEMES NUMERIQUES, CODAGE, DECODAGE ET TRANSCODAGE

### 1/ Systèmes numériques :

Afin d'accélérer l'exécution des programmes, il faut choisir un langage qui soit le plus rapide. Or le langage le plus rapide sur un ordinateur est celui qui est directement compréhensible par celui-ci : c'est le langage machine; il utilise le système de numération binaire (à base 2) qui ne comporte que les chiffres "0" et "1".

Ces chiffres binaires sont appelés "bits" (Binary digits) et physiquement, dans les circuits électroniques numériques, un bit "0" est représenté par une tension basse (Low) et un bit "1" par une tension haute (High).

#### a) Système binaire :

La base d'un système de numération est le nombre de chiffres différents qu'utilise ce système.

Tout nombre N peut être décomposé en fonction des puissances entières de la base de son système de numération.

Dans un nombre quelconque, le chiffre de droite s'appelle le chiffre de poids faible tandis que le chiffre de gauche s'appelle le chiffre de poids fort.

Etant donné un nombre N et un système de numération de base b, il existe une suite :  $a_0, a_1, \dots, a_{n-1}$  d'éléments de l'ensemble des chiffres de ce système, telle que :

$$N = a_{n-1} b^{n-1} + a_{n-2} b^{n-2} + \dots + a_1 b + a_0 .$$

Dans le cas du système binaire,  $b=2$ , les coefficients  $a_i$  sont pris parmi les chiffres du système binaire "0" et "1".

Exemple : Déterminer la valeur décimale de  $N=(101101)_2$

$$\begin{aligned} N=(101101)_2 &= 1.2^5 + 0.2^4 + 1.2^3 + 1.2^2 + 0.2^1 + 1.2^0 \\ &= 32 + 0.16 + 1.8 + 1.4 + 0 + 1.1 \\ &= 32 + 8 + 4 + 1 = 45 \end{aligned}$$

$$N=(101101)_2 = (45)_{10}$$

#### b) Conversion d'un nombre de base décimale en un nombre de base b quelconque :

La méthode consiste à diviser le nombre à convertir par la base du nouveau système et à conserver le reste.

On répète ce processus en considérant chaque fois le quotient obtenu. On écrit ensuite tous les restes à partir de la fin et de

gauche à droite, en les convertissant en lettres s'il y a lieu.

Exemple : Convertir le nombre  $N=(189520)_{10}$  en hexadécimal ( $b=16$ ).

Division par 16	Quotient	Reste
189520 / 16	11845	0
11845 / 16	740	5
740 / 16	46	4
46 / 16	2	14
2 / 16	0	2

$(2E450)_{16}$

D'où :

$$N=(189520)_{10} = (2E450)_{16}$$

Vérification :

$$\begin{aligned}
 N=(2E450)_{16} &= 2 \cdot 16^4 + 14 \cdot 16^3 + 4 \cdot 16^2 + 5 \cdot 16^1 + 0 \cdot 16^0 \\
 &= 2.65536 + 14.4096 + 4.256 + 5.16 + 0 \\
 &= 131072 + 57344 + 1024 + 80 \\
 &= 189520
 \end{aligned}$$

c) Nombres fractionnaires :

La conversion d'un nombre décimal fractionnaire en un nombre de base  $b$  se fait suivant le même procédé vu dans le paragraphe précédent mais au lieu d'une division on aura une multiplication.

Exemple : Convertir en binaire le nombre  $N=(0,72145)_{10}$

0,72145	x 2 =	<span style="border: 1px solid black; padding: 0 2px;">1</span>	, 44290
0,44290	x 2 =	<span style="border: 1px solid black; padding: 0 2px;">0</span>	, 88580
0,88580	x 2 =	<span style="border: 1px solid black; padding: 0 2px;">1</span>	, 7716
0,7716	x 2 =	<span style="border: 1px solid black; padding: 0 2px;">1</span>	, 54320
0,54320	x 2 =	<span style="border: 1px solid black; padding: 0 2px;">1</span>	, 08640
0,08640	x 2 =	<span style="border: 1px solid black; padding: 0 2px;">0</span>	, 1728
0,1728	x 2 =	<span style="border: 1px solid black; padding: 0 2px;">0</span>	, 34560
0,34560	x 2 =	<span style="border: 1px solid black; padding: 0 2px;">0</span>	, 69120
0,69120	x 2 =	<span style="border: 1px solid black; padding: 0 2px;">1</span>	, 38240

Donc  $N=(0,72145)_{10} = (0,101110001)_2$ ; il suffit d'écrire de gauche à droite les nombres encadrés pris de haut en bas.

d) Conversion binaire-octal et vice-versa :

Quatre (04) chiffres binaires ou bits permettent  $2^4=16$  combinaisons et donc d'écrire les seize (16) entiers de 0 à 15 selon le tableau suivant :

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

Ce code est nommé CODE BINAIRE NATUREL.

Chacun de ces chiffres représente le poids d'un bit. Nous allons illustrer comment convertir rapidement un nombre de base 2 en un nombre de base 8.

Règle : A partir de la virgule, grouper les bits par blocs de 3 en allant vers la gauche pour la partie entière et vers la droite pour la partie fractionnaire.

Convertir ensuite ces blocs en octal selon le code binaire naturel. Cette propriété découle du fait que la base 8 du système octal est une puissance entière de 2, en effet  $2^3 = 8$ .

Exemple :

$$\begin{aligned} *) N &= ( \underline{111} \underline{110} \underline{011}, \underline{011} \underline{101} )_2 \\ &= ( 7 \quad 6 \quad 3 \quad 3 \quad 5 )_8 \end{aligned}$$

\*)  $N = (567,315)_8$ , il faut écrire par blocs de 3 bits, la valeur binaire des chiffres :

$$N = (101 \ 110 \ 111 \ , \ 011 \ 101)_2$$

\*) Convertir  $N = (79182)_{10}$  en binaire :

Convertissons ce nombre en octal :

$$\begin{aligned} N &= (79182)_{10} = (232516)_8 \\ &= (010 \ 011 \ 010 \ 101 \ 001 \ 110)_2 \end{aligned}$$

Cette méthode plus rapide que la conversion directe en binaire, car le nombre de division par 2 est trois (03) fois plus grand que celui des divisions par 8.

e) Conversion binaire-hexadécimal et vice-versa :

La base du système de numération hexadécimal est aussi une puissance entière de 2, en effet  $16 = 2^4$ . On a donc les mêmes propriétés

que pour le régime octal, mais cette fois on groupe les bits par blocs de quatre (04).

Le système de numération hexadécimale utilise les 16 symboles : 0 à 9 , A , B , C , D , D , E et F.

Exemples:

$$1/ N = (\underline{1001} \ \underline{1100} \ \underline{1011} \ \underline{1010} , \ \underline{0111} )_2$$

$$N = ( 9 \quad C \quad B \quad A , \quad 7 )_{16}$$

$$2/ N = (A78, B32)_{16} = (1010 \ 0111 \ 1000 , \ 1011 \ 0011 \ 0010)_2$$

3/ Convertir en binaire le nombre décimal suivant :

$$N = (11432)_{10} = (2CA8)_{16} = (0010 \ 1100 \ 1010 \ 1000)_2$$

4/ Convertir en décimal le nombre hexadécimal suivant :

$$N = (12A3)_{16} = 1.16^3 + 2.16^2 + 10.16^1 + 3.16^0$$

$$= (1.4096) + (2.256) + (10.16) + (3.1)$$

$$= (4771)_{10}$$

f) Opérations arithmétiques en binaire :

Les opérations sur les nombres binaires telles que l'addition, la soustraction ou la multiplication, sont exécutées de manière identique à celle de l'arithmétique décimale.

+	0	1
0	0	1
1	1	10

x	0	1
0	0	0
1	0	1

Exemples :

\*)  $5 + 6 = (11)_{10}$   
 $5 = (101)_2$  et  $6 = (110)_2$

$$\begin{array}{r} 101 \\ + 110 \\ \hline 1011 \end{array}$$

Vérification :  $(1011)_2 = 1.2^3 + 0.2^2 + 1.2^1 + 1.2^0 = 2^3 + 2 + 1 = (11)_{10}$

\*)  $10 - 5 = (5)_{10}$   
 $(10)_{10} = (1010)_2$   
 $(5)_{10} = (0101)_2$

$$\begin{array}{r} \overset{0}{\uparrow} \overset{0}{\uparrow} \\ 1010 \\ - 0101 \\ \hline 0101 \end{array}$$

Vérification :  $(0101)_2 = 0.2^3 + 1.2^2 + 0.2^1 + 1.2^0 = 0 + 4 + 0 + 1 = 5$

\*)  $5 \times 6 = (30)_{10}$   
 $5 = (101)_2$   
 $6 = (110)_2$

$$\begin{array}{r} 101 \\ x 110 \\ \hline 000 \\ 101 \\ \hline 101 \\ \hline 1110 \end{array}$$

Vérification :  $(11110)_2 = 1.2^4 + 1.2^3 + 1.2^2 + 1.2^1 + 0.2^0 = (30)_{10}$

g) Nombres négatifs et positifs binaires normalisés à 8 caractères :

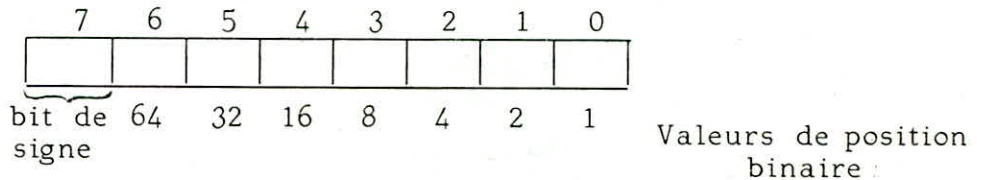
Les signes (+) et (-) ne sont pas assimilables tels quels par un ordinateur, lequel ne connaît que deux états "0" et "1".

On convient donc de les représenter par un bit qui occupera la case de gauche du jeu de case d'écriture du nombre considéré.

Ce bit est appelé : le bit du signe et il est égal à "0" si le nombre est positif et à "1" s'il est négatif.

La figure suivante représente l'aspect que pourrait présenter un registre ou une case mémoire à l'intérieur du microprocesseur.

Ce registre présente un espace de 8 bits de données. Les positions des bits sont numérotées de 0 à 7 et les poids des positions binaires sont indiqués à la base du registre.



Exemples :

	64	32	16	8	4	2	1	
0	0	1	0	0	0	0	0	+32
1	1	1	0	0	1	0	1	-27
0	1	1	1	1	1	1	1	+127

On remarque que le nombre -27 est représenté par son complément à 2.

h) Représentation par le complément à 2: Complément vrai :

Lorsqu'il est nécessaire d'utiliser des nombres signés, un code particulier, la notation en complément à 2, est mis en oeuvre.

Ce système simplifie la circuiterie de la machine.

On obtient le complément à 2 d'un nombre en ajoutant 1 au complément restreint qui s'obtient en inversant tous les bits du nombre considéré.

Exemple : Quelle serait la notation en complément à 2 de -9 ?

9  $\xrightarrow{\text{binaire}}$  0 0 0 0 1 0 0 1  
 complément restreint : 1 1 1 1 0 1 1 0  
 complément vrai : 1 1 1 1 0 1 1 1 = -9



2/ Codage, décodage et transcodage :

a) Le codage :

Le codage est une opération qui établit une correspondance entre les éléments de deux ensembles.

En terme plus mathématique, le codage peut être défini comme une application bijective d'un ensemble source vers un autre ensemble appelé "but". L'ensemble source est en général un sous-ensemble des nombres rationnels et l'ensemble but contient des combinaisons ordonnées de "0" et de "1".

A un nombre donné correspond une combinaison binaire appelée "mot-code" et l'ensemble des mots-code forme le code. D'autre part, des tentatives de normalisation de codage des caractères alphanumériques dans les calculateurs ont donné naissance à des codes très utilisés tels que :

- \* le code ASCII : American Standard Code for Information Interchange
- \* le code BCD : Binary Coded Decimal.

La figure suivante présente un extrait du code ASCII ainsi que du code BCD.

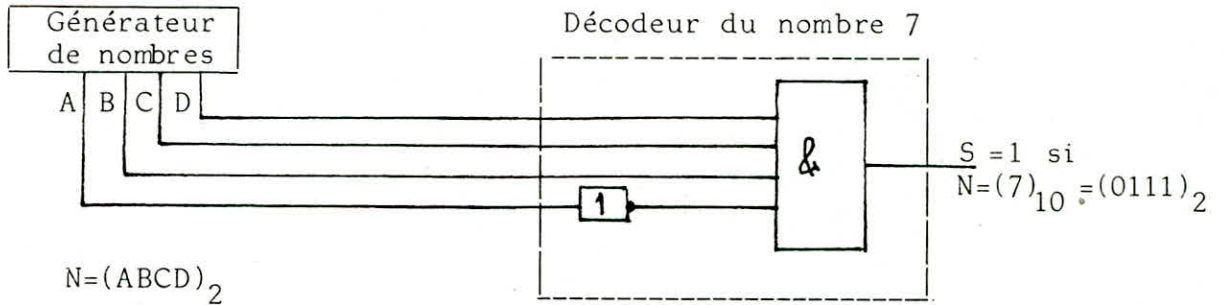
$(N)_{10}$	B C D	ASCII (à 7 bits)
0	0 0 0 0	0 1 1 0 0 0 0
1	0 0 0 1	0 1 1 0 0 0 1
2	0 0 1 0	0 1 1 0 0 1 0
3	0 0 1 1	0 1 1 0 0 1 1
4	0 1 0 0	0 1 1 0 1 0 0
5	0 1 0 1	0 1 1 0 1 0 1
6	0 1 1 0	0 1 1 0 1 1 0
7	0 1 1 1	0 1 1 0 1 1 1
8	1 0 0 0	0 1 1 1 0 0 0
9	1 0 0 1	0 1 1 1 0 0 1

b) Le décodage :

En pratique, on trouve deux types de décodage :

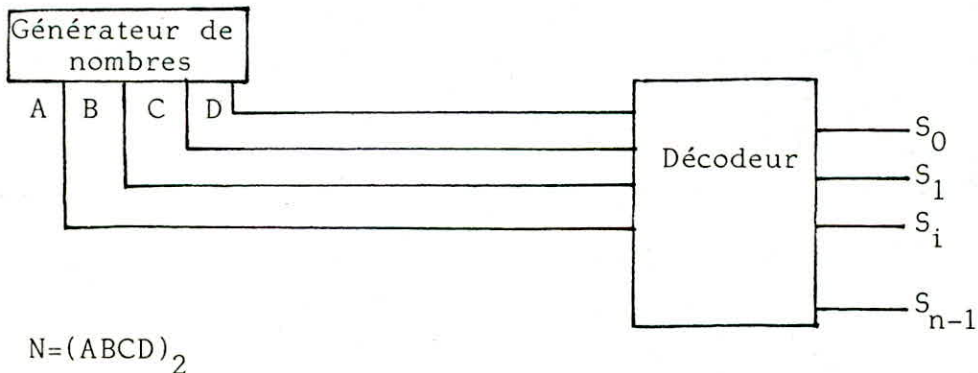
\*) Le décodage d'une configuration (ou d'un mot-code) déterminé : Dans ce cas, le décodage est obtenu à l'aide d'une simple porte "ET" vérifiant la simultanéité des variables binaires correspondantes à la configuration à décoder.

Dans l'exemple ci-dessous, la sortie S du circuit de décodage passera par 1 chaque fois que le générateur des nombres délivrera le nombre 7.



\*) Le décodage de toutes les combinaisons d'un code : Un tel système regroupe autant de décodeurs élémentaires qu'il y a de mots-code. Soit n le nombre de combinaisons. Donc ce type disposera de n sorties. Comme la combinaison des variables binaires à l'entrée du décodeur est stable et unique à un instant donné, il ne peut exister qu'une seule sortie dans un état actif.

Toutes les autres sorties sont dans un état de repos.



c) Le transcodage :

Il est défini comme étant le passage du nombre N écrit dans le code  $C_1$  au même nombre écrit dans le code  $C_2$ ; Il permet donc de passer d'un code à un autre.

**CHAPITRE III**

### III/ ELEMENTS NUMERIQUES DE BASE

En 1854 Georges Boole, mathématicien anglais, publia un essai sur les raisonnements logiques dont les objets sont des propositions auxquelles les seules réponses possibles sont oui ou non.

L'ensemble des opérations formelles appliquées à ces propositions forment une structure mathématique appelée Algèbre, dite Algèbre de Boole.

Elle est applicable à l'études des systèmes binaires, c'est-à-dire aux systèmes possédant deux états s'excluant mutuellement : c'est évidemment le cas des circuits logiques, base des systèmes numériques.

1/ Fonctions logiques :

- Les états logiques seront représentés par les nombres 0 et 1.
- Une variable logique est une grandeur, représentée par un symbole (lettre ou signe) qui peut prendre les seules valeurs 0 ou 1.
- Une fonction logique se présente comme une association de sommes et de produits de variables logiques reliées par des opérateurs logiques.

Une fonction logique, tout comme les variables logiques, ne peut prendre que deux valeurs 0 ou 1.

Il existe trois opérateurs logiques élémentaires : ET , OU et NON. Donc trois fonctions élémentaires suffisent pour définir une algèbre de Boole : la fonction complément ou NON, la fonction ET ou produit logique et la fonction OU ou addition logique.

a) La fonction complément ou NON :

C'est une fonction d'une variable qui se note par le symbole - (barre) et qui a la valeur que n'a pas la variable. Sa table de vérité est la suivante :

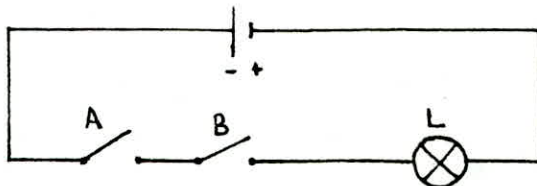
A	F(A)= $\bar{A}$
0	1
1	0

b) La fonction ET ou produit logique :

En langage logique, cette fonction, c'est-à-dire la fonction ET, de deux ou plusieurs variables prend la valeur 1 si toutes les variables sont simultanément égales à 1, ce qui donne la table de vérité suivante pour deux variables :

A	B	$F(A,B)=A.B$	
0	0	0	$F(A,B)=1$ si A et B sont égales à 1.
0	1	0	
1	0	0	Le signe de multiplication symbolise l'opération ET.
1	1	1	

Exemple : Interprétation électrique : Le circuit électrique suivant fonctionne suivant l'équation logique  $L=A.B$



On constate que la lampe L ne peut être allumée que si les deux interrupteurs A et B sont fermés simultanément. En effet, la table de vérité de la fonction  $F(A,B)=L=A.B$  est effectivement retrouvée si les conventions suivantes sont adoptées :

- L'interrupteur A est fermé :  $A=1$
- L'interrupteur B est fermé :  $B=1$
- La lampe L est allumée :  $L=1$

c) La fonction OU ou addition logique :

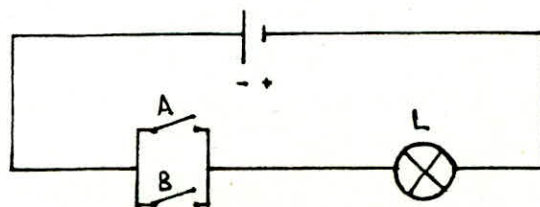
La fonction OU de deux ou plusieurs variables logiques prend la valeur 1 si au moins une des variables est égale à 1.

Sa table de vérité à 2 variables est la suivante :

A	B	$F(A,B)=A+B$	
0	0	0	$F(A,B)=1$ si A ou B sont égales à 1.
0	1	1	
1	0	1	Le signe de l'addition symbolise l'opération OU.
1	1	1	

Exemple : Interprétation électrique : Le circuit suivant traduit la fonction  $L = A + B$  pour les mêmes conventions que pour la fonction ET.

En effet la lampe L est allumée si l'on actionne indifféremment l'interrupteur A ou B ou bien les deux à la fois.



d) Simplification des fonctions logiques :

A l'image d'une fonction logique, les solutions matérielles apportées à un problème donné sont multiples. Les choix se feront avant tout sur des critères d'optimisation de coût, de complexité, de disponibilité des composants ...

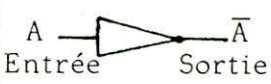


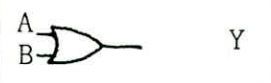
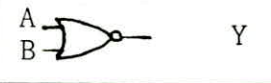

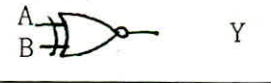
Alors, afin d'utiliser le minimum de matériel et donc de fabriquer un système à moindre coût, il faut simplifier les fonctions logiques à réaliser. Pour cela, il existe trois possibilités :

- \* La méthode algébrique;
- \*\* La méthode graphique utilisant les diagrammes de Karnaugh;
- \*\*\* Les méthodes programmables.

2/ Réalisation des fonctions logiques à l'aide des portes logiques :

Les circuits utilisés au traitement des signaux numériques sont appelés "portes logiques" et pour les identifier, on utilise des symboles logiques.

Le tableau suivant illustre les sept opérateurs logiques fondamentaux de la logique des systèmes numériques. Ces derniers sont constitués de combinaisons de portes logiques.

Fonction logique	Symbole de la porte logique	Table de vérité	Expression Booléenne																								
NON ou Complément		<table border="1"> <thead> <tr> <th>Entrée</th> <th>Sortie</th> </tr> </thead> <tbody> <tr> <td>A</td> <td><math>\bar{A}</math></td> </tr> <tr> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> </tr> </tbody> </table>	Entrée	Sortie	A	$\bar{A}$	0	1	1	0	$A = \bar{A}$																
Entrée	Sortie																										
A	$\bar{A}$																										
0	1																										
1	0																										
ET (AND)		<table border="1"> <thead> <tr> <th colspan="2">Entrées</th> <th colspan="2">Sorties</th> </tr> <tr> <th>B</th> <th>A</th> <th>ET</th> <th>NON-ET</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	Entrées		Sorties		B	A	ET	NON-ET	0	0	0	1	0	1	0	1	1	0	0	1	1	1	1	0	$A \cdot B = Y$
Entrées		Sorties																									
B	A	ET	NON-ET																								
0	0	0	1																								
0	1	0	1																								
1	0	0	1																								
1	1	1	0																								
NON-ET (NAND)		<table border="1"> <thead> <tr> <th colspan="2">Entrées</th> <th colspan="2">Sorties</th> </tr> <tr> <th>B</th> <th>A</th> <th>OU</th> <th>NON-OU</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	Entrées		Sorties		B	A	OU	NON-OU	0	0	0	1	0	1	1	0	1	0	1	0	1	1	1	0	$\overline{A \cdot B} = Y$
Entrées		Sorties																									
B	A	OU	NON-OU																								
0	0	0	1																								
0	1	1	0																								
1	0	1	0																								
1	1	1	0																								
OU (OR)		<table border="1"> <thead> <tr> <th colspan="2">Entrées</th> <th colspan="2">Sorties</th> </tr> <tr> <th>B</th> <th>A</th> <th>OU</th> <th>NON-OU</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	Entrées		Sorties		B	A	OU	NON-OU	0	0	0	1	0	1	1	0	1	0	1	0	1	1	1	0	$A+B = Y$
Entrées		Sorties																									
B	A	OU	NON-OU																								
0	0	0	1																								
0	1	1	0																								
1	0	1	0																								
1	1	1	0																								
NON-OU (NOR)		<table border="1"> <thead> <tr> <th colspan="2">Entrées</th> <th colspan="2">Sorties</th> </tr> <tr> <th>B</th> <th>A</th> <th>XOU</th> <th>XNON-OU</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> <td>1</td> </tr> </tbody> </table>	Entrées		Sorties		B	A	XOU	XNON-OU	0	0	0	1	0	1	1	0	1	0	1	0	1	1	0	1	$\overline{A+B} = Y$
Entrées		Sorties																									
B	A	XOU	XNON-OU																								
0	0	0	1																								
0	1	1	0																								
1	0	1	0																								
1	1	0	1																								
OU exclusif (XOR)		<table border="1"> <thead> <tr> <th colspan="2">Entrées</th> <th colspan="2">Sorties</th> </tr> <tr> <th>B</th> <th>A</th> <th>XOU</th> <th>XNON-OU</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> <td>1</td> </tr> </tbody> </table>	Entrées		Sorties		B	A	XOU	XNON-OU	0	0	0	1	0	1	1	0	1	0	1	0	1	1	0	1	$A \oplus B = Y$
Entrées		Sorties																									
B	A	XOU	XNON-OU																								
0	0	0	1																								
0	1	1	0																								
1	0	1	0																								
1	1	0	1																								
NON-OU Exclusif		<table border="1"> <thead> <tr> <th colspan="2">Entrées</th> <th colspan="2">Sorties</th> </tr> <tr> <th>B</th> <th>A</th> <th>XOU</th> <th>XNON-OU</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> <td>1</td> </tr> </tbody> </table>	Entrées		Sorties		B	A	XOU	XNON-OU	0	0	0	1	0	1	1	0	1	0	1	0	1	1	0	1	$\overline{A \oplus B} = Y$
Entrées		Sorties																									
B	A	XOU	XNON-OU																								
0	0	0	1																								
0	1	1	0																								
1	0	1	0																								
1	1	0	1																								

-- COMPARAISON DES SEPT FONCTIONS LOGIQUES --

3/ Circuits des codeurs, décodeurs et des transcodeurs :

a) Codeurs :

Un codeur est un dispositif qui traduit les valeurs d'une entrée dans un code choisi.

Par exemple, un clavier de console comporte n touches. Chaque touche, représentative d'un caractère, est affectée d'un numéro. L'opération de codage consiste à donner à chaque numéro un équivalent binaire.

i-Intérêt du codage : Si le nombre de boutons est de 10, codé en

éléments...

binaire pur, quatre (4) variables suffisent. Pour un clavier classique, la quarantaine de touches se codent facilement avec 6 variables binaires.

D'où le codage des informations apporte une réduction non négligeable du nombre de variables à traiter.

ii-Application : Codeur prioritaire :

Si, maladroitement, plusieurs boutons sont enfoncés simultanément, le codeur classique donne un résultat érroné car il ne sait plus quel numéro doit être codé.

Un codeur prioritaire est un dispositif réalisant le codage du numéro le plus élevé dans le cas où plusieurs touches seraient actionnées.

Si une seule commande est envoyée sur le codeur prioritaire, celui-ci fonctionne comme un codeur classique.

iii-Réalisation pratique des codeurs:

Dans sa version la plus simple, un codeur est un ensemble de circuits OU.

Soit la table de codage suivante pour les nombres de 1 à 9 :

N	est traduit par	A	B	C	D
1	-----	0	0	0	1
2	-----	0	0	1	0
3	-----	0	0	1	1
4	-----	0	1	0	0
5	-----	0	1	0	1
6	-----	0	1	1	0
7	-----	0	1	1	1
8	-----	1	0	0	0
9	-----	1	0	0	1

Dans ce qui suit, l'expression (N=i) signifie que l'entrée i du codeur est portée au niveau logique 1.

La sortie A est égale à 1 si (N=9) ou (N=8); de même :

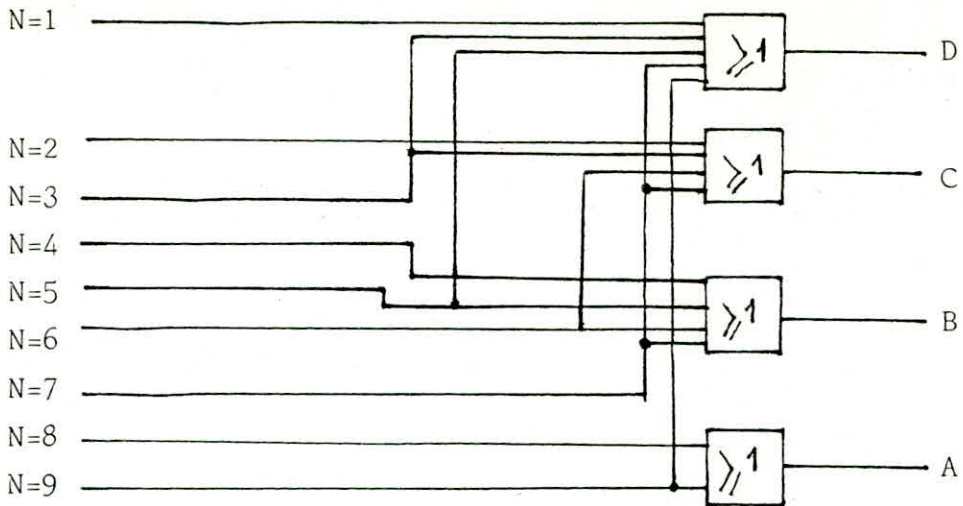
La sortie B " " à 1 si (N=4) ou (N=5) ou (N=6) ou (N=7).

La sortie C " " à 1 si N=2 ou N=3 ou N=6 ou N=7.

La sortie D " " à 1 si N=1 ou N=3 ou N=5 ou N=7 ou N=9.

Ce qui donne le schéma suivant :





(\*) Le symbole  $\geq 1$  représente l'opérateur logique "OU".

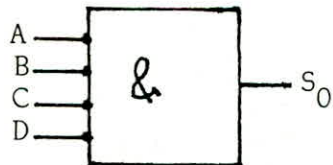
b) Décodeur : Synthèse d'un décodeur BCD :

Soit un chiffre décimal N écrit en BCD.

Si N est égal à  $(0)_{10}$ , la sortie  $S_0$  passe dans l'état actif, par exemple 1. La table de vérité de la fonction  $S_0$  est donc la suivante :

N	Code	A	B	C	D	$S_0$
0	BCD====>	0	0	0	0	1
1		0	0	0	1	0
2		0	0	1	0	0
3		0	0	1	1	0
4		0	1	0	0	0
5		0	1	0	1	0
6		0	1	1	0	0
7		0	1	1	1	0
8		1	0	0	0	0
9		1	0	0	1	0

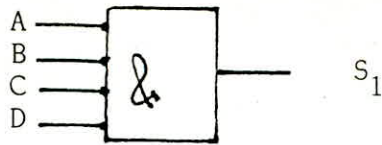
$$S_0 = \bar{A} \bar{B} \bar{C} \bar{D}$$



(\*) Le symbole "&" représente l'opérateur logique "ET"

Dans sa version la plus simple, un décodeur est un ensemble de circuits ET.

De même pour la sortie  $S_1$ , on aura :



Notons que N est compris entre 0 et 9. Les combinaisons des variables A, B, C, D traduisant les autres nombres sont impossibles, ce qui signifie que les valeurs de sorties peuvent être choisies librement pour obtenir le schéma le plus simple.

Ainsi, pour les autres nombres, et à l'aide du tableau de Karnaugh, on obtient les résultats suivants:

$$\begin{array}{lll}
 S_2 = \overline{BCD} & ; & S_3 = \overline{BCD} & ; & S_4 = \overline{BCD} \\
 S_5 = \overline{BCD} & ; & S_6 = \overline{BCD} & ; & S_7 = \overline{BCD} \\
 S_8 = \overline{AD} & ; & S_9 = \overline{AD} & & 
 \end{array}$$

c) Transcodeur :

Un transcodeur est un dispositif qui permet de passer du nombre N écrit dans le code  $C_1$  au même nombre écrit dans le code  $C_2$ .

Synthèse d'un transcodeur :

Le nombre N dans le code  $C_1$  s'exprime à l'aide des variables A, B, C, D par exemple, et dans le code  $C_2$  avec les variables X, Y, Z et T. Le nombre des variables dans chaque code n'est pas forcément identique).

Le problème de la synthèse d'un transcodeur revient à calculer chacune des sorties, c'est-à-dire les variables de  $C_2$  (ici X, Y, Z, T) en fonction des entrées ou variables du code  $C_1$  (ici A, B, C, D). Soit :

$$\begin{array}{l}
 X = f_1(A, B, C, D) \\
 Y = f_2(A, B, C, D) \\
 Z = f_3(A, B, C, D) \\
 T = f_4(A, B, C, D)
 \end{array}$$

Étudions, en exemple, un transcodeur binaire-binaire réfléchi. Les variables A, B, C, D représentent le nombre N exprimé dans le code B. réfléchi (Gray) et X, Y, Z, T représentent le même nombre en binaire pur.

N	A	B	C	D	X	Y	Z	T
0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	1
2	0	0	1	1	0	0	1	0
3	0	0	1	0	0	0	1	1
4	0	1	1	0	0	1	0	0
5	0	1	1	1	0	1	0	1
6	0	1	0	1	0	1	1	0
7	0	1	0	0	0	1	1	1
8	1	1	0	0	1	0	0	0
9	1	1	0	1	1	0	0	1
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.

X est égale à 1 si les variables A, B, C, D, prennent les valeurs 1100, 1101, ..., ce que l'on peut reporter dans un tableau de Karnaugh pour obtenir l'expression la plus simple de la fonction  $f_1$ .  
On procède de la même façon pour Y, Z et T, ce qui donne :

AB \ CD	00	01	11	10
00				
01				
11	1	1	1	1
10	1	1	1	1

$$X = A$$

AB \ CD	00	01	11	10
00				
01	1	1	1	1
11				
10	1	1	1	1

$$Y = \bar{A}B + A\bar{B} = A \oplus B$$

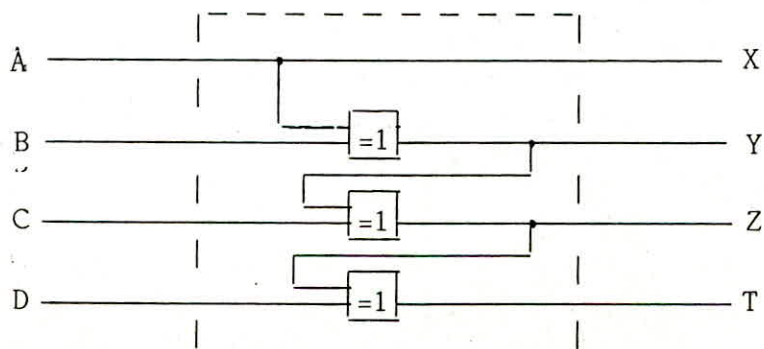
AB \ CD	00	01	11	10
00			1	1
01	1	1		
11			1	1
10	1	1		

$$Z = A \oplus B \oplus C$$

AB \ CD	00	01	11	10
00		1		1
01	1		1	
11		1		1
10	1		1	

$$T = A \oplus B \oplus C \oplus D$$

D'où le schéma sera :



(\*) Lesymbole " $\boxed{=1}$ " représente l'opérateur logique OU exclusif (XOR).

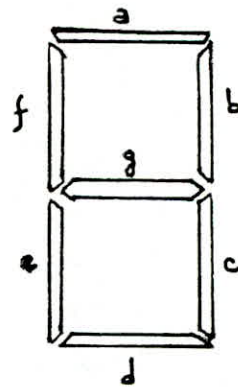
d) Application : Décodeur 7 segments :

On appelle décodeur 7 segments le dispositif de transcodage permettant de passer du code BCD au code d'affichage du chiffre. On utilise souvent de tels afficheurs pour représenter les dix chiffres décimaux.

Soient a, b, c, d, e, f et g les variables correspondant aux 7 segments. Si une variable est au niveau actif, le segment correspondant est allumé.

Le code à 7 segments correspondant est donné par la table :

N	a	b	c	d	e	f	g
0	1	1	1	1	1	1	0
1	0	1	1	0	0	0	0
2	1	1	0	1	1	0	1
3	1	1	1	1	0	0	1
4	0	0	1	0	0	1	1
5	1	0	1	1	0	1	1
6	1	0	1	1	1	1	1
7	1	1	1	0	0	0	0
8	1	1	1	1	1	1	1
9	1	1	1	1	0	1	1



4/ Multiplexeurs. et démultiplexeurs :

a) Multiplexeurs :

Dans les cas simples, on transmet les signaux directement (transmission en bande de base).

Cependant le problème se pose de plus en plus de multiplexer, c'est-à-dire de transmettre plusieurs signaux sur une seule ligne. Dans ce cas, l'intérêt du multiplexage est dans l'économie de câbles.

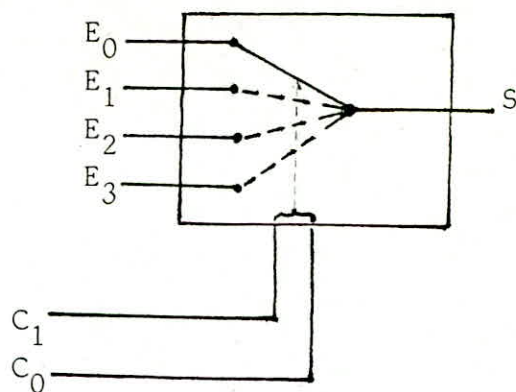
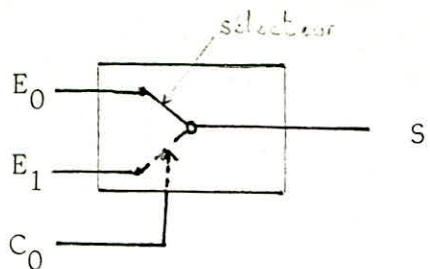
On distingue deux procédés de multiplexage :

i-Multiplexage dans le temps : Cela consiste à utiliser des signaux échantillonnés, analogiques ou numériques.

ii-Multiplexage en fréquence : Dans ce cas, chaque signal module une porteuse en amplitude ou en fréquence, et les porteuses sont décalées de façon à ce que les bandes latérales ne se recouvrent pas.

Un multiplexeur, également appelé sélecteur de données, est un circuit réalisant un aiguillage de l'une des entrées vers une sortie unique.

Il est schématisé par les figures suivantes :



$C_0$	$S$
0	$E_0$
1	$E_1$

(Validation = 0)  
(Invalidation = 1)

$C_1$	$C_0$	$S$
0	0	$E_0$
0	1	$E_1$
1	0	$E_2$
1	1	$E_3$

Le commutateur rotatif possède plusieurs entrées et une seule sortie. Par rotation du sélecteur, les données à l'une quelconque des entrées peuvent être transférés à la sortie.

La position de l'interrupteur est fixée par une commande. Lorsque la sélection se fait sur 2 entrées, un seul élément binaire de commande. Et ainsi de suite pour  $2n$ , il est nécessaire d'avoir  $n$  éléments binaires de commande.

b) Démultiplexeur :

Un démultiplexeur est un dispositif réalisant également un aigüillage d'information.

La différence entre le multiplexeur et le démultiplexeur réside dans le sens de circulation de l'information, c'est-à-dire ce dernier a pour effet de renverser l'action du multiplexeur.

Un démultiplexeur distribue l'information d'entrée vers l'une des  $2^n$  sorties; la sélection de la sortie concernée étant effectuée par  $n$  variables de commande.

Les autres sorties sont alors dans un état de repos.

Lorsque l'entrée est toujours égale à 1, le démultiplexeur fonctionne comme un décodeur binaire.

Les multiplexeurs et les démultiplexeurs peuvent être utilisés ensemble pour transmettre des données le long d'une ligne unique sous forme sérielle. Le multiplexeur constituera l'unité émettrice, le démultiplexeur, l'unité réceptrice qui remet les données sous leur forme originale.

# CHAPITRES IV ET V

#### IV/ LES DIFFERENTS LANGAGES DE PROGRAMMATION :

##### a) Langages évolués :

Ce sont des langages de haut niveau dont l'apprentissage est rapide. Parmi ces langages, on distingue :

- Le langage FORTRAN ,
- " " " BASIC ,
- " " " PASCAL ,
- " " " COBOL ...

##### b) Langages machine et d'assemblage :

i- Le langage machine : On a déjà évoqué avant, que le seul langage compréhensible par l'ordinateur était le langage machine des 0 et 1. Cependant, par exemple, le langage BASIC est incompréhensible par l'ordinateur, d'où, pour l'utiliser, l'ordinateur dispose d'une interface entre les langages machine et BASIC manipulée par l'utilisateur. Cette interface est un programme écrit en langage machine et stocké dans la ROM. On appelle cette interface aussi "Interpréteur BASIC". Ainsi tout programme en langage machine autre que l'interpréteur BASIC devra s'implanter en mémoire vive RAM.

##### \*Avantages du langage machine par rapport au BASIC :

Le langage BASIC est interprété, c'est-à-dire que durant l'exécution d'un programme, chaque instruction est examinée pour déterminer sa nature et les paramètres qu'elle contient, puis elle est exécutée.

Cette phase de recherche ralentit l'exécution d'un programme, surtout si l'instruction à exécuter se trouve dans une boucle FOR,...,NEXT, car elle est décodée à chaque passage dans la boucle.

D'où, pour diminuer le temps d'exécution d'un programme (cas du domaine de la régulation), on doit choisir le langage machine qui est directement compréhensible par le microprocesseur. Un autre avantage du langage machine sur le BASIC est que le programme en langage machine occupe moins de place en mémoire qu'un programme écrit en BASIC.

##### \*Inconvénients du langage machine :

La programmation en langage machine est réservée aux programmeurs

qui ont une certaine expérience en programmation puisqu'elle demande un minimum de connaissances de la structure interne du microprocesseur.

Les erreurs de syntaxe du BASIC peuvent être signalées par le programme moniteur, tandis que celles du programmeur en langage machine peuvent provoquer l'effacement complet de la mémoire. On peut ainsi résumer les inconvénients du langage machine comme suit :

- programmes difficiles à lire et à mettre au point,
- programmes non adaptables à d'autres ordinateurs,
- difficultés d'effectuer les calculs arithmétiques compliqués.

ii- Etude du langage d'assemblage :

Pour faire exécuter une instruction au microprocesseur, il suffit de lui envoyer son code et comme il est difficile de se rappeler à quel code correspond une instruction, d'autant qu'il y a plusieurs instructions en langage machine, on a eu l'idée alors, de substituer à chaque code un mnémonique constitué de 2 ou 3 lettres facile à retenir et qui renseigne sur le rôle de cette instruction. Ce langage a reçu le nom de "langage d'assemblage".

Exemple : J P (Jump = saut)  
          C P (Compare)  
          AND (ET)

Chaque constructeur a défini des mnémoniques pour toutes les instructions de son microprocesseur.

\*L'assembleur :

Les mnémoniques n'étant pas directement exécutables par les microprocesseurs, on utilise alors un programme appelé "Assembleur" qui effectue la conversion entre les mnémoniques et les codes. Comme le langage assembleur est écrit en langage machine, il doit être implanté en mémoire vive RAM, ceci diminue l'espace mémoire destiné au programme utilisateur.

L'inconvénient des programmes en langage assemblage, c'est qu'ils sont longs à écrire.

Insistons bien sur la confusion fréquente entre un programme assembleur (écrit en langage assembleur) et le programme ASSEMBLEUR chargé de la traduction.



## V/ LE MICROPROCESSEUR INTEL 8085 :

Le premier problème qui a retardé le développement de la commande numérique par ordinateur est celui de la très grande rapidité des phénomènes à contrôler.

Or la dernière décennie a vu une évolution considérable de l'électronique et de l'informatique, sous l'influence du développement technologique des circuits intégrés.

Ces derniers étaient au départ monofonctionnels et l'introduction de la notion propre de l'informatique de programmes enregistrés dans une mémoire, a permis de concevoir des circuits intégrés multifonctionnels : ils se comportent comme des automates électroniques à programmer enregistrés c'est-à-dire comme des ordinateurs désignés sous le nom de: microprocesseurs.

Parmi les microprocesseurs les plus répandus et qui sont largement utilisés aussi bien dans l'industrie que dans l'enseignement on trouve ceux à 8 bits. Leur capacité leur permet d'innombrables applications :

- \* Systèmes de contrôles numériques (automatismes industriels : contrôle et surveillance de processus, commandes de machines-outils, asservissements, etc...),
- \* Systèmes de traitement de données (calculatrices, micro-ordinateurs, gestion des terminaux, etc...).

Actuellement parmi les microprocesseurs à 8 bits les plus célèbres, il y a les 8080/8085 d'Intel qui sont très voisins du populaire Z80 de Zilog, mais il en existe d'autres dont les différentes améliorations portent sur la vitesse et la capacité de traitement; sur la capacité d'adressage des mémoires et des entrées/sorties.

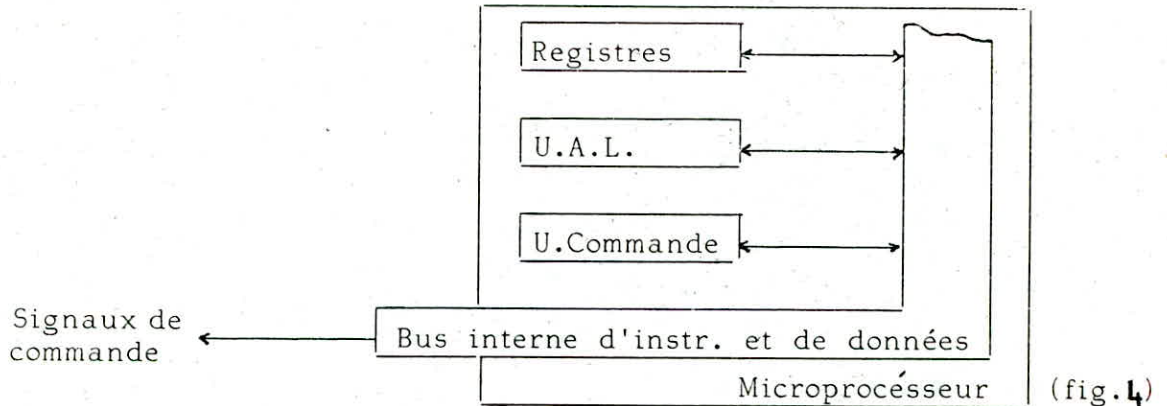
Leur domaine privilégié est la réalisation de systèmes complets de micro-ordinateurs. En effet, ils sont conçus comme de véritables unités centrales et peuvent gérer de nombreux éléments de mémoires et de périphériques.

### 1/ Architecture du microprocesseur Intel 8085 :

Le microprocesseur 8085 est un membre d'une famille de circuits compatibles comprenant des RAM, des ROM et des interfaces de communication avec des périphériques.

Il comprend essentiellement : (voir fig.4)

- \* Une unité arithmétique et logique (U.A.L.) qui effectue les différents traitements: opérations arithmétiques et logiques, tests.
- \* Une unité de commande qui va chercher dans la mémoire chaque instruction à exécuter, la décode et génère en conséquence tous les signaux nécessaires pour l'exécution correcte de l'instruction.
- \* Des registres, souvent utilisés pour des stockages provisoires d'informations.



\*\* STRUCTURE DE BASE D'UN MICROPROCESSEUR \*\*

Typiquement, le microprocesseur Intel 8085 est contenu dans un circuit intégré dit boîtier dual en ligne à 40 broches. La figure (5) illustre le mode de présentation du microprocesseur en céramique.

Le microprocesseur sur céramique est utilisable à haute température.

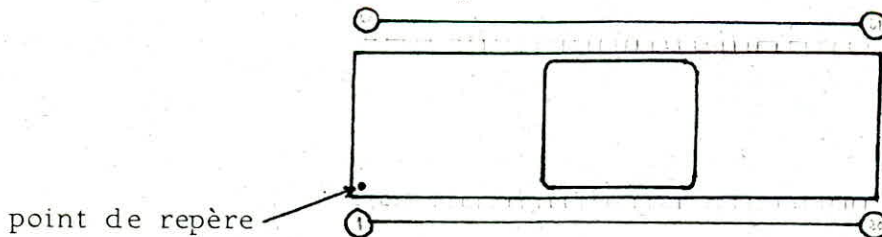


Fig.(5) : (Vue de dessus) : Repérage des broches.

La structure interne des registres du 8085 est représentée par la figure (6) :

- Il y a :
- \* un accumulateur A de 8 bits, qui :
    - .servira pour toutes les opérations arithmétiques et la plupart des opérations logiques,
    - .recevra en général les données en provenance de la mémoire et destinés à être dirigés vers un périphérique.

\*des registres auxiliaires au nombre de 6, ayant 8 bits chacun (B, C, D, E, H et L) et qui peuvent être utilisés par paires de 16 bits : (B,C) , (D,E) , (H,L).

\*un pointeur de pile de 16 bits.

La pile est une petite partie de la mémoire RAM, utilisée pour sauvegarder les contenus des registres lorsque cela est nécessaire, par exemple lors d'un appel de sous-programme ou d'une interruption. L'adresse de la prochaine position-mémoire à utiliser pour la sauvegarde de données est à tout moment fournie par le contenu d'un registre appelé : pointeur de pile.

\*un registre compteur d'instruction qui contient l'adresse de la prochaine instruction à exécuter et il est à 16 bits. Dans la littérature anglaise il est désigné par Program Counter(PC)

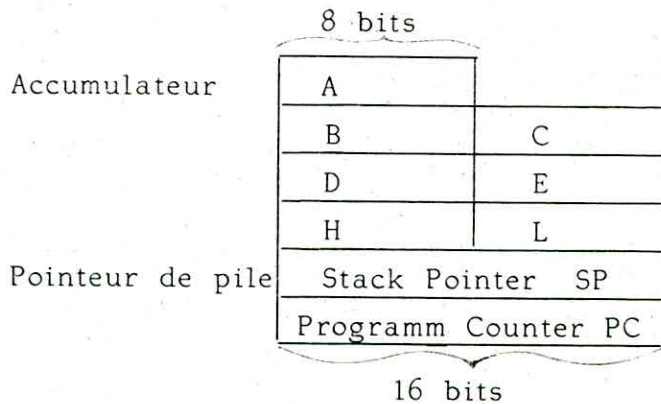


Fig. (6)

\*\* LES REGISTRE DU MICROPROCESSEUR 8085 \*\* *Rif(7&8)*

2/ Programmation :

Au niveau le plus fondamental, le microprocesseur réagit à une liste d'opérations appelée : programme machine ou binaire.

D'où une instruction, pour être admise par le microprocesseur doit comporter les informations suivantes :

- \* Le code binaire de l'instruction, établi par le constructeur (en binaire ou en hexadécimal),
- \* La valeur binaire de l'opérande à traiter,
- \* L'adresse binaire d'implantation en mémoire de l'instruction.

Donc tout programme, pour être exécutable par le microprocesseur, doit être chargé en mémoire sous cette forme.

Malheureusement, l'écriture binaire est très peu commode à manipuler et il est difficile de programmer directement en langage

machine. C'est pourquoi il existe différents langages de programmation plus facile à comprendre que le binaire. Parmi ces langages, on trouve les langages assembleurs. Ces derniers permettent au programmeur d'écrire chaque instruction du langage machine sous forme d'abréviations symboliques.

Un programme "ASSEMBLEUR" a un rôle de traduction : à partir d'un programme source, rédigé dans un langage assembleur, l'"ASSEMBLEUR" doit générer en langage machine un programme objet, exécutable par le microprocesseur.

Un langage assembleur permet de :

\*représenter un opérande sous forme symbolique :

Ex: une instruction de chargement de l'accumulateur pourra

s'écrire :                   LDA                   DATA  
                                  mnémonique           opérande symbolique

\*représenter les adresses sous une forme symbolique appelée étiquettes.

En conclusion, une instruction écrite en assembleur comportera essentiellement les éléments suivants :

Etiquette   Mnémonique   Opérande   Commentaires

Les commentaires, que peut rajouter la programmeur pour faciliter la compréhension du programme, n'interviendront pas dans l'exécution du programme.

### 3/ Jeu d'instruction (\*):

Le groupe d'instructions qu'un microprocesseur peut exécuter constitue son ensemble, ou jeu d'instructions.

Les jeux d'instructions ne sont pas normalisés. Cet inconvénient est dû aussi bien à l'individualisme des constructeurs qu'aux différences d'architecture et à l'usage prévu des microprocesseurs.

Il y a plusieurs manières de classer les instructions d'un microprocesseur et d'après la normalisation donnée par l'IEEE (Société Savante d'ingénieurs électroniciens) on trouve :

- a) Les instructions arithmétiques;
- b) Les instructions logiques;
- c) Les instructions de transfert de données;
- d) Les instructions d'appel à sous-programme;
- e) Les instructions de retour;
- f) Les instructions diverses.

(\* ) Jeu d'instruction du 8085 Intel (voir annexe)

**CHAPITRE**

**VI**

## VI/ MESURE NUMERIQUE DE TEMPERATURE :

### 1/ Introduction et généralités :

De toutes les grandeurs physiques, la température est certainement l'une de celles dont la mesure est la plus fréquente.

C'est pourquoi, en recherche comme dans l'industrie, la mesure précise et le contrôle très strict des températures sont indispensables. D'où du nombre important des propriétés de la matière et de phénomènes physiques sensibles à la température, résulte une grande diversité de méthodes de mesure :

- a) Méthodes optiques : basées sur la répartition spectrale du rayonnement émis ou l'élargissement des raies spectrales, par l'effet Doppler dû à l'agitation thermique, ...
- b) Méthodes mécaniques : fondées sur la dilatation d'un solide, d'un liquide ou d'un gaz à pression constante, ...
- c) Méthodes électriques : reposant sur la variation thermique de la valeur d'une résistance, sur l'effet Seebeck ou sur la sensibilité thermique de la fréquence d'oscillation d'un quartz, ...

\*\* Les méthodes optiques qui s'appuient sur l'observation extérieure d'une propriété du milieu dont on mesure la température n'apportent à celle-ci aucune perturbation, mais leur domaine d'emploi est cependant limité et leur mise en oeuvre est d'une certaine complexité; par contre les méthodes électriques, objet de notre étude, sont d'une grande généralité et d'une mise en oeuvre relativement simple. \*\*

### 2/ Théorie et différents types de capteurs :

Pour recueillir des informations (Entrées) et pour donner des ordres (Sorties), l'ordinateur a besoin de capteurs, d'actionneurs et d'interfaces adaptés.

- Les interfaces assurent la compatibilité entre l'ordinateur et les signaux venant des capteurs et allant vers les actionneurs.
- Les actionneurs exécutent les ordres donnés par l'ordinateur.
- Les capteurs ont pour mission de délivrer un signal correspondant à un événement ou à une situation.

Dans la pratique, on distingue deux grandes familles de capteurs: les capteurs délivrant les informations "tout-ou-rien" et les capteurs de mesure ou analogiques.

a) Capteurs "tout-ou-rien" :

Ils donnent une information binaire 0 ou 1 selon qu'un évènement s'est produit ou non. Tous les dispositifs dont le nom se termine par "STAT" entrent dans cette catégorie. Aquastat, thermostat, ... délivrent un signal 0 ou 1 selon qu'un certain seuil de comparaison a été ou n'a pas été franchi.

Viennent se joindre à cette catégorie les détecteurs de présence ou d'absence de tension ou de courant. (La tonalité du téléphone indiquant que le combiné est raccordé au réseau.)

b) Les capteurs de mesure ou analogiques :

Le terme "analogique" provient de l'analogie qui existe entre le signal fourni par un capteur et la mesure qu'il représente. Ces capteurs plus ambitieux que les précédents, délivrent une information qui est l'image à tout moment du phénomène mesuré.

Parmi ces capteurs, on distingue :

-Visuels : Le capteur fait dévier l'aiguille d'un voltmètre, la colonne d'un thermomètre, ...

Dans ce cas, on dit qu'on a affaire à de simples indicateurs et leur signal ne peut être transmis vers un automatisme;

-Pneumatiques,

-Hydrauliques,

-Capteurs de température.

3/ Etude des capteurs de température :

Dans la plupart des cas, la mesure d'une grandeur non électrique se ramène à sa transformation en une grandeur électrique qui en dépend d'une façon univoque et dont la mesure permet de déterminer la valeur de la grandeur non électrique à mesurer.

L'élément de mesure ayant pour fonction d'effectuer cette transformation porte le nom de capteur (dans notre cas capteur thermique).

En pratique, on distingue deux principaux types de capteurs de température :

A- Thermométrie par thermocouples :

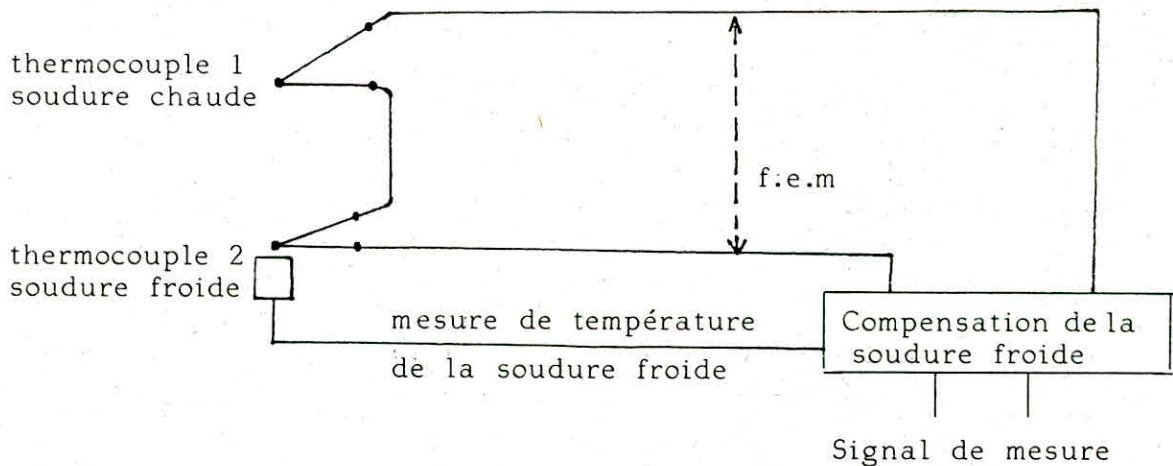
On utilise l'effet "Peltier Thomson".

Selon les lois découvertes par ce physicien, un thermocouple constitué de deux conducteurs A et B formant entre eux deux jonctions aux températures  $T_1$  et  $T_2$ , délivre une f.e.m.  $E_{A/B}^{T_2 T_1}$

qui dépend d'une part de la nature des conducteurs A et B et d'autre part des températures  $T_1$  et  $T_2$ .

En général, la température de l'une des jonctions est fixe, connue et sert de référence ( $T_1 = T_{\text{réf}}$ ); celle  $T_2$  de l'autre jonction est la température  $T_c$  qu'elle atteint lorsque placée dans le milieu étudié de température inconnue  $T_x$ ; la température  $T_c$  est fonction de  $T_x$  et de l'ensemble des échanges thermiques possibles avec d'autres milieux.

Si on connaît la température de la soudure (jonction) fixe dite "froide" et la f.e.m. entre les deux jonctions, des tables donnent la valeur de la température de la soudure "chaude".



fig[7] \*\* THERMOMETRIE PAR THERMOCOUPLE \*\*

\*Présentation des thermocouples :

Les fils du thermocouple vont par paire et sont sélectionnés suivant l'emploi car la faculté de donner un courant plus fort par une température donnée varie d'un couple à un autre. Dans l'industrie, ces thermocouples, pour être différenciés, sont affectés de couleur.

Le tableau suivant nous donne quelques types de thermocouples conformes à la norme française NF.C.42.324. réf



Nature des Conducteurs	Nature des thermocouples	C O U L E U R		
		Conducteur +	Conducteur -	Assemblage
Cuivre Cuivre nickel T	T	Jaune	Bleu	Bleu
Fer Cuivre nickel J	J	Jaune	Noir	Noir
Nickel chrome Nickel allié	K	Jaune	Violet	Violet
Cuivre Cuivre nickel V		Jaune	Brun	Brun
Cuivre Cuivre nickel S	S	Jaune	Vert	Vert

\*Thermocouples sous forme de câble bifilaire\*

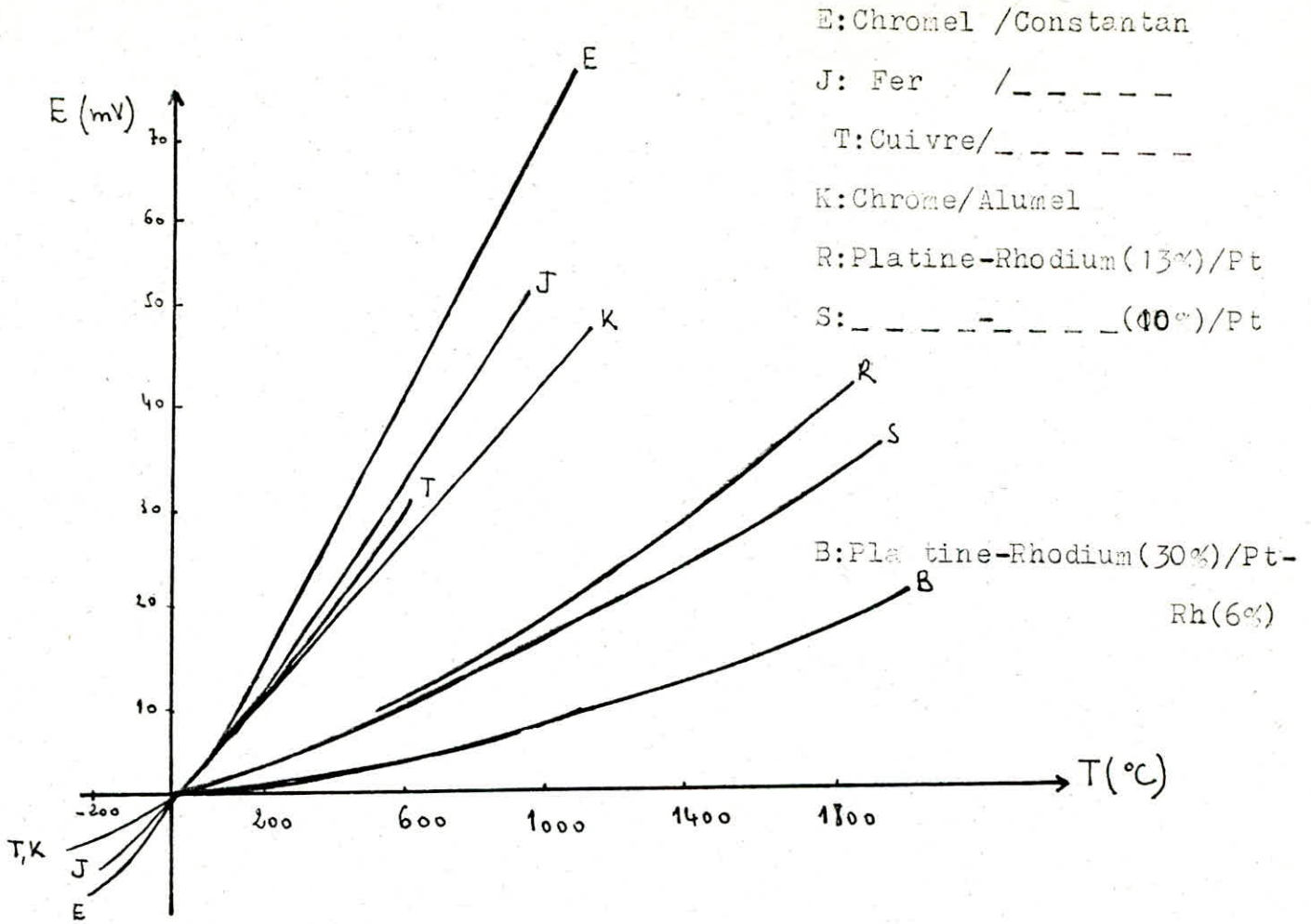
Notons que le premier nom cité représente le pôle positif et le second le pôle négatif.

Parmi les avantages des thermocouples sur les résistances, que nous allons voir dans la suite, on peut dire que :

- le thermocouple permet des mesures de température ponctuelles,
- le thermocouple délivre un signal, une f.e.m., dont la mesure ne nécessite pas la circulation d'un courant dans le capteur : il n'y a donc pas, contrairement aux résistances, aucune incertitude liée à l'autoéchauffement, ce qui peut être appréciable dans le cas de mesures sur des systèmes à basse température.

Cependant, et cela est un inconvénient du thermocouple, la mesure exige que la température de la jonction de référence soit connue, d'où toute erreur ou incertitude sur  $T_{réf}$  risque d'entraîner une incertitude du même ordre sur  $T_c$ .

Ainsi que la f.e.m. du thermocouple est, sur de grands intervalles de température, une fonction non linéaire de  $T_c$ . (Voir fig. ):



fig[1] \*\* VARIATION THERMIQUE DE LA f.e.m. DE DIFFERENTS TYPES DE THERMOCOUPLES \*\* Réf(10)

La non-linéarité de la relation entre la f.e.m. du thermocouple et la température est mise en évidence par la forme polynomiale de l'équation qui les lie.

D'où, pour chaque type de thermocouple, une norme définit :

- d'une part, une table de valeurs de la f.e.m. en fonction de la température T,
- d'autre part, une expression polynomiale qui traduit algébriquement et en conformité avec la table la relation entre la f.e.m. E et la température T.

Le tableau suivant illustre par les principaux types de couples, les températures limites d'utilisation normale pour les diamètres de fil indiqués, ainsi que la précision standard lorsqu'elle a été définie par une norme :

Thermocouples	Plage de t°	E (mv) <sub>(1)</sub>	Précision (2)
Cuivre / Constantan Type T (Ø 1,63mm)	-270°C à 370°C	- 6,258 à 19,027	+2 % de -100°C à -40°C +0,8% -40°C 100°C +0,75% 100°C 350°C
Chromel / Constantan Type E (Ø 3,25mm)	-270°C à 870°C	- 9,835 à 66,473	+3 % de 0°C à 400°C +0,75% 400°C à 1250°C
Platine-Rhodium (13%) / Platine Type R (Ø 0,51 mm)	-50°C à 1500°C	- 0,226 à 17,445	+1,4 °C de 0°C à 538°C +0,25% de 538°C à 1500°C (3)
Platine-Rhodium (30%) / P.R. (6%) - Type B (Ø 0,51 mm)	0°C à 1700°C	0,00 à 12,426	+0,5% de 870°C à 1700°C (3)
Tungstène-Rhénium (5%) / Tungstène-Rhénium (26%)	0°C à 2700°C	0 à 38,45	

\*\* PRINCIPAUX TYPES DE THERMOCOUPLES ET LEURS LIMITES D'EMPLOI \*\*

- (1) Norme NF C 42-321
- (2) Norme NF E 18-001
- (3) Norme ANSI C 96-6

*Ref (10 & 14)*

B- Thermométrie par résistance :

a) Sensibilité thermique et critères de choix d'un métal :

D'une façon générale, la valeur d'une résistance dépend de sa température T suivant la fonction suivante :

$R(T) = R_0 \cdot F(T - T_0)$ , où  $R_0$  étant la résistance à la température  $T_0$  et la fonction F une caractéristique du matériau, égale à 1 pour  $T = T_0$ . C'est ainsi que l'on a pour les métaux :

$$R(T) = R_0 (1 + AT + BT^2 + CT^3).$$

D'où la résistance électrique des métaux très purs varie rapidement avec la température (0,3 à 0,6% par °Kelvin à la température ambiante). Cependant la présence d'une impureté ou d'éléments d'alliages modifie considérablement le comportement thermique de la plupart des métaux.

Les métaux généralement employés pour la confection de résistances thermométriques sont : le cuivre, le nickel et le platine.

Le cuivre a l'avantage d'une caractéristique  $R=f(T)$  linéaire; il peut être employé aux températures les plus basses, mais à hautes températures, il risque d'être oxydé.

L'emploi des résistances en cuivre est néanmoins limité par la faible résistivité de ce métal.

.Le nickel n'est pas à conseiller pour la mesure de températures inférieures à 0°C ou supérieures à 120°C car son risque d'oxydation peut affecter sa stabilité et limite ainsi son utilisation à des températures inférieures à 250°C.

.Le platine, bien connu pour ses qualités de stabilité, permet d'obtenir des résistances thermométriques dont les caractéristiques demeurent invariables pendant de longues périodes et elles sont susceptibles de mesurer des températures dans un très large domaine.

Le tableau suivant indique, pour les métaux précédemment considérés, les valeurs d'un certain nombre de caractéristiques physiques importantes : *Re, 10)*

- . $T_f$  : température de fusion,                      .C : chaleur massique à 20°C,
- . $\lambda_t$  : conductivité thermique,                      . $\alpha_1$  : coef. dilatation linéaire,
- . $\rho$  : résistivité à 20°C,                                      . $\alpha_\rho$  : coef. température de  $\rho$  à 20°C.

Caractéristiques	Cuivre	Nickel	Platine
$T_f$ (°C)	1083	1453	1769
C (J°C <sup>-1</sup> kg <sup>-1</sup> )	400	450	135
$\lambda_t$ (W°C <sup>-1</sup> m <sup>-1</sup> )	400	90	73
$\alpha_1$ (°C <sup>-1</sup> )	16,7.10 <sup>-6</sup>	12,8.10 <sup>-6</sup>	8,9.10 <sup>-6</sup>
$\rho$ (Ω.m)	1,72.10 <sup>-8</sup>	10.10 <sup>-8</sup>	10,6.10 <sup>-8</sup>
$\alpha_\rho$ (°C <sup>-1</sup> )	3,9.10 <sup>-3</sup>	4,7.10 <sup>-3</sup>	3,9.10 <sup>-3</sup>

b) Les résistances thermométriques en platine :

.C'est en raison de sa très grande stabilité que le platine pur a été utilisé dans un étalon international de mesure de température entre -200°C et 1000°C . Dans ce but, le platine ultrapur est étiré en un fil puis bobiné en s'attachant à ne pas introduire d'impuretés dans le métal, et à éliminer toutes contraintes mécaniques internes : il s'agit là d'un capteur prototype, inutilisable en raison de sa fragilité aux vibrations et aux chocs mécaniques; dans les applications industrielles, on utilise en pratique un fil de platine sur support.

.Pour représenter la variation de la résistance du platine avec la

température, la loi de CALLENDAR-VAN DUSSEN est souvent adoptée :

$$\frac{R(T)}{R(0)} = 1 + \alpha \left[ T - \delta \left( \frac{T}{100} - 1 \right) \left( \frac{T}{100} \right) - \beta \left( \frac{T}{100} - 1 \right) \left( \frac{T}{100} \right)^3 \right]$$

où T est exprimé en °C et  $\beta=0$  pour  $T > 0^\circ\text{C}$

$\alpha$ ,  $\beta$  et  $\delta$  sont des coefficients, constants pour chaque sonde.

Quatre mesures sont nécessaires pour déterminer les coefficients  $R(0)$ ,  $\alpha$ ,  $\beta$  et  $\delta$ .

.  $R(0)$  est mesurée directement en maintenant le capteur à  $0^\circ\text{C}$ .

.  $\alpha$  s'obtient en mesurant la résistance  $R(100)$  de la sonde à  $100^\circ\text{C}$ , on a alors  $R(100)=R(0)(1+100\alpha)$  d'où  $\alpha = \frac{R(100)-R(0)}{100R(0)}$

.  $\beta$  s'obtient par une mesure à basse température, généralement la température d'ébullition de l'oxygène ( $-182,97^\circ\text{C}$ )

.  $\delta$  est mesuré par une quatrième mesure, au point d'ébullition du soufre ( $444,6^\circ\text{C}$ ), par exemple.

Pour une résistance de platine de grande pureté, un constructeur (Rosemount) donne :

$$\alpha = 0,003925 \quad ; \quad \delta = 1,492$$

$$\beta = 0,11 \text{ pour } T < 0^\circ\text{C} \text{ et } \beta=0 \text{ pour } T > 0^\circ\text{C}.$$

La formule de CALLENDAR-VAN DUSSEN peut être écrite sous la forme équivalente suivante :

$$\frac{R(T)}{R(0)} = 1 + AT + BT^2 + C(T-100)T^3,$$

$$\text{où } A = \alpha \left( 1 + \frac{\delta}{100} \right) \quad \text{et} \quad B = -\alpha \cdot \delta \cdot 10^{-4}$$

$$C = -\alpha \cdot \beta \cdot 10^{-8} \text{ pour } T < 0^\circ\text{C} \text{ et } C = 0^\circ\text{C pour } T > 0^\circ\text{C},$$

$$\text{d'où } A = 3,90802 \cdot 10^{-3} \quad ; \quad B = -5,80195 \cdot 10^{-7}$$

$$C = -4,27350 \cdot 10^{-12} \text{ pour } T < 0^\circ\text{C} \text{ et } C = 0 \text{ pour } T > 0^\circ\text{C}$$

La courbe suivante représente la variation de la résistance en fonction de la température, de la sonde à platine. (\*)

(\*) Voir tableau donnant la relation Temp./Valeur Ohmique (Annexe).

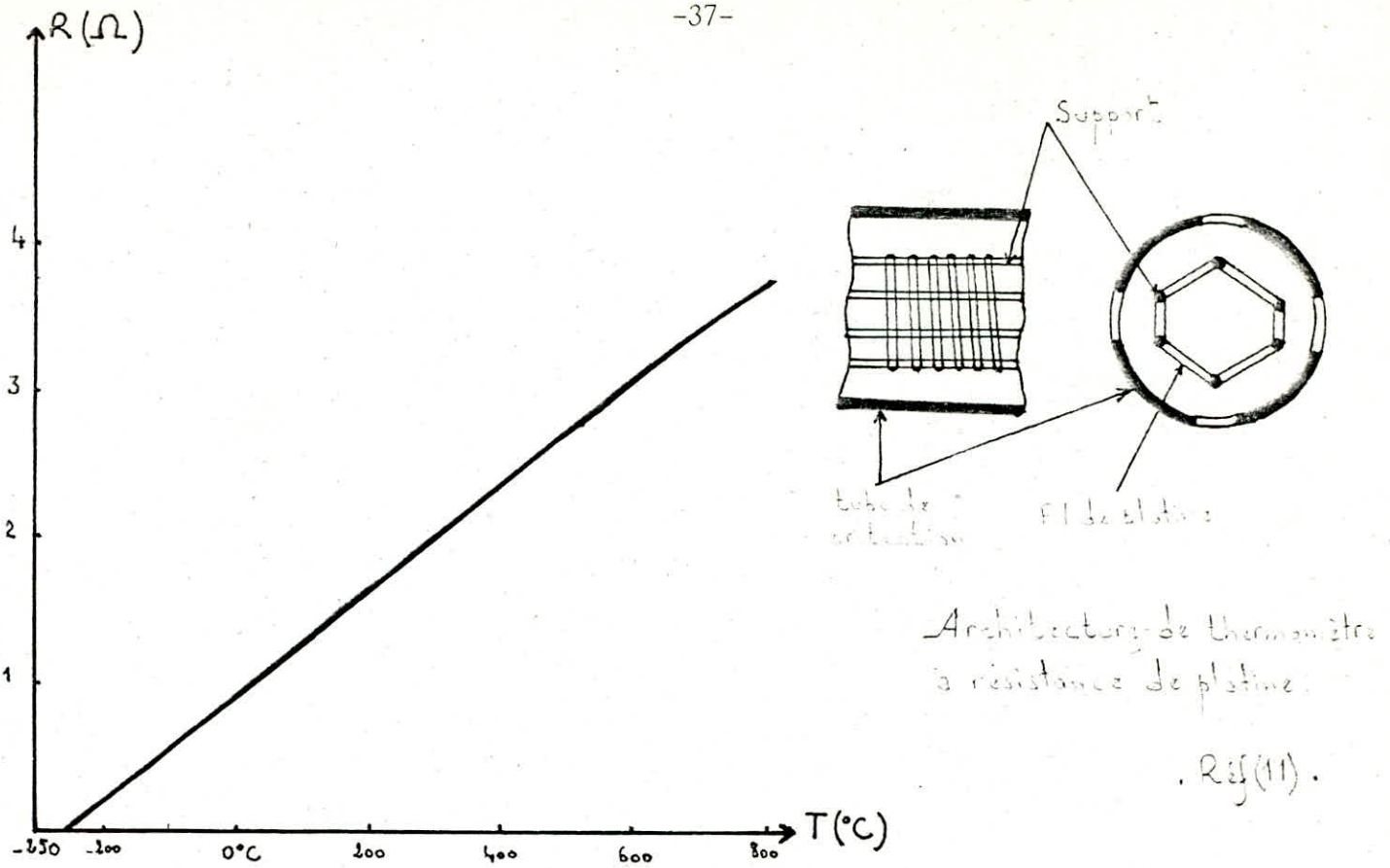
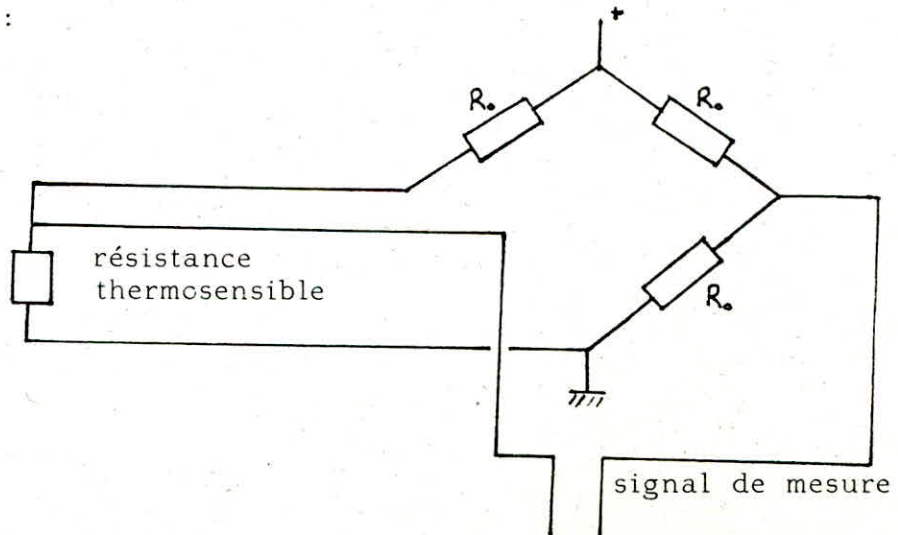


fig [9]: Courbe  $R=f(T)$  d'une sonde thermométrique en platine. Réf(11)

c) Méthode de mesure :

Le plus souvent la mesure de la résistance de la sonde thermométrique est réalisée au moyen d'un pont de Wheatstone dont l'une des branches est constituée par la résistance thermométrique et les trois autres branches sont formées par trois résistances fixes et égales à  $R_0$ , résistance thermométrique à  $0^{\circ}\text{C}$ .

(voir fig.10) :



(fig.10)

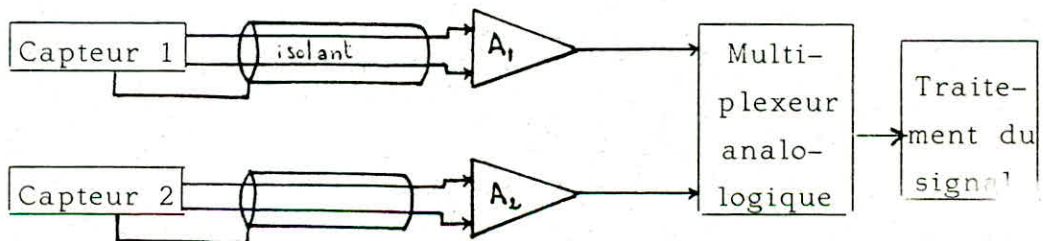
d) Amplification :

Comme le signal issu du capteur est très faible, et afin qu'il soit très bien exploité, on réalise une amplification au moyen d'amplificateur.

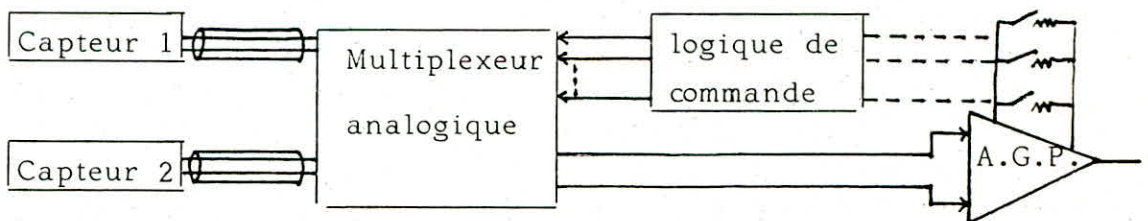
Dans des ensembles d'acquisition de données issues de plusieurs capteurs, on peut associer à chacun de ces derniers un amplificateur d'instrumentation dont le gain est fixé en fonction du niveau moyen du signal délivré et qui est localisé à proximité du capteur. Ceci permet la transmission d'un signal de haut niveau et évite d'effectuer le multiplexage à bas niveau.

(Voir fig.1A)

Une autre solution consiste à utiliser un amplificateur à gain programmable (A.G.P.). Ce dernier, placé après le multiplexeur, a un gain réglable par commutation de résistances afin d'être adapté au niveau du signal fourni par chaque capteur (voir fig.1B). Une commande par circuit logique permet à la fois le choix de la voie scrutée par le multiplexeur et la fixation de la valeur appropriée du gain de l'amplificateur.



(fig.1A) : MULTIPLEXAGE DES VOIES APRES AMPLIFICATION DES SIGNAUX



(fig.1B) :

MULTIPLEXAGE DES VOIES PREALABLES A L'AMPLIFICATION DES SIGNAUX

e) Conversion Analogique-Numérique et affichage des données :

Dans tous les appareils contemporains, l'affichage numérique des mesures faites est une technique importante car elle permet d'éliminer l'erreur de lecture qui résulte de l'utilisation des appareils à aiguilles.

Pour afficher, il faut procéder à la transformation du signal analogique au moyen d'un C.A.N. en une série de niveau haut ou bas exprimant dans un code convenu un nombre représentant l'amplitude du signal analogique. Ce signal codé est appliqué à un organe d'affichage permettant la lecture de cette amplitude.

i-Conversion analogique-numérique : elle fait correspondre une suite de nombres  $a_n$  au signal d'entrée  $X(t)$ , chaque nombre correspond au fait à l'amplitude d'un échantillon du signal  $X^*(t)$ .  $X^*(t)$  est la suite de valeurs discrètes de ce signal, prélevées à des intervalles de temps réguliers de période  $T_e$ .

ii-Rappels théoriques :

.Echantillonnage: Echantillonner une fonction, c'est prélever la valeur de cette fonction pendant un certain intervalle de temps de manière périodique. On substitue une autre fonction discontinue, obtenue à partir de la fonction continue par décomptage de cette dernière. Cette opération (échantillonnage) sera possible s'il existe une relation entre la fréquence d'échantillonnage  $F_e$  et la fréquence maximale  $F_{\max}$  du spectre du signal analogique. Cette relation porte le nom de théorème d'échantillonnage ou théorème de Shannon, qu'on peut énoncer comme suit :

"Un signal  $Z(t)$  dont le spectre est limité à la fréquence  $F_{\max}$  est entièrement déterminé par la suite complète de ses échantillons prélevés à des intervalles de temps réguliers de valeur :

$$T = \frac{1}{2F_{\max}} \text{ "}$$

.Quantification :L'information numérique est un message de longueur définie. Ainsi dans une base  $b$ ,  $n$  chiffres permettent de coder  $N$  valeurs distinctes. On ne pourra donc coder, c'est-à-dire reconnaître, que  $N$  valeurs bien définies de la grandeur analogique. L'opération qui consiste à remplacer la valeur exacte du signal par une valeur choisie par les  $N$  valeurs quantifiées est la quantification.



iii) Technique de la C.A.N.: Sue le plan pratique, quatre (4) techniques sont très utilisées :

.Convertisseur à double rampe : Il est utilisé en instrumentation, ses caractéristiques essentielles sont :

\*Convertisseur lent (quelques dizaines de ms),

\*Fible sensibilité au bruit, ...

.Convertisseur à comptage : C'est un convertisseur très voisin de celui à rampe, de réalisation simple mais présentant le désavantage d'un temps de conversion long pour une résolution donnée.

.convertisseur parallèle : C'est le type de convertisseur le plus rapide, le temps de conversion se limitant aux temps de commutation des comparateurs et de propagation à travers les portes de transcodage.

.Convertisseur à approximation successive : Ses avantages sont : une bonne résolution et une grande rapidité.

\*Principe: On compare en permanence le signal d'entrée  $V_e$  à une tension analogique précise générée par un C.N.A. Le processus de conversion est similaire à celui de la pesée à l'aide d'une balance à plateaux : on essaye les poids par valeurs décroissante tant que le poids de l'objet n'est pas équilibré ou garde tous les poids utilisés jusqu'à créer le déséquilibre. A ce moment on retire le poids ayant provoqué le déséquilibre et l'on essaie celui immédiatement inférieur jusqu'à trouver l'équilibre.

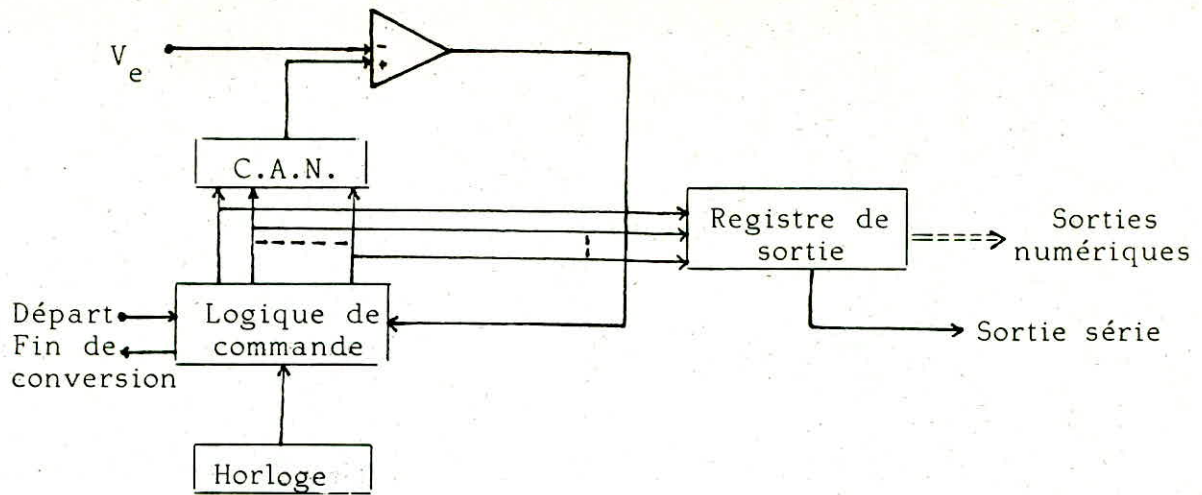
\*Fonctionnement : La tension analogique  $V_e$  est comparée par le C.A.N. au bit de plus haut poids (M.S.B.), ce qui situe immédiatement  $V_e$  par rapport à la demi-échelle ( $\frac{V_{ref}}{2}$ ) :

Si  $V_e \geq \frac{V_{ref}}{2}$ , on garde le MSB à 1 dans le registre de sortie

et le bit de rang immédiatement inférieur est essayé.

Si  $V_e < \frac{V_{ref}}{2}$ , on garde le MSB à 0 dans la position correspondante du registre de sortie et le suivant est essayé.

Si l'apport du second bit est insuffisant par rapport à  $V_e$ , on essaie le troisième bit, etc ... Après avoir essayé tous les bits, le registre de sortie contient la représentation binaire d'un signal analytique le plus proche de  $V_e$ .



\*SCHEMA DE PRINCIPE DU CONVERTISSEUR A APPROXIMATIONS SUCCESSIVES\* [13]

iv-Affichage: L'information présente à la sortie du convertisseur est en binaire. L'affichage de cette information nécessite deux transformations :

- 1-Transcodage binaire - BCD,
- 2-Décodage BCD vers afficheur sept segments.

**CHAPITRE**

**VII**

VII/ ETUDE ET MISE EN MARCHÉ DU BANC D'ESSAI :

A-- Le module d'application : MIC 955 pour thermostat :

1/ Description et données techniques :

Le module MIC 955 d'application de calculateur est conçu pour des exercices pratiques d'utilisation de microcalculateur portant sur :

- \*la mesure de température numérique,

- \*la régulation de l'alimentation électrique,

- \*la formation de boucle simple de régulation en circuit fermé.

Le module MIC 955 est un appareil miniature idéal pour l'enseignement des principes de base du fonctionnement d'un thermostat.

Le MIC 955 comprend un panneau avec une bande chauffante d'aluminium anodisé noir et un ventilateur axial qui assure un refroidissement supplémentaire. La bande supporte deux résistances de chauffage et une résistance à bande de platine décapé qui sert de capteur de température.

Les circuits électroniques correspondants sont montés sur la carte à circuit imprimé à l'arrière.

- .Les résistances de chauffage sont alimentées avec un courant stabilisé par la borne de sortie du microcalculateur avec, en tampon, un amplificateur de puissance dont la sortie peut être contrôlée à la prise de "contrôle de réchauffeur" sur le panneau.

- .Le ventilateur axial de refroidissement peut être commandé de deux façons sélectionnées par un commutateur à 3 positions "programmables". En mode "programmé", le courant vers le moteur est branché ou coupé par un autre bit de la borne de sortie du calculateur qui agit à travers un amplificateur intermédiaire.

En mode "manuel", on peut commander la vitesse du ventilateur à l'aide d'une résistance variable dont le bouton de commande s'intitule "blower speed" ("vitesse de ventilateur").

- .Le capteur de température à platine est relié à un circuit de source à courant constant à pont. Sa sortie est amplifiée et la sortie de l'amplificateur est équilibrée à zéro à 0°C; sa sensibilité est de 20 mV/°C à la prise de "contrôle de température" (Temperature monitor) sur le panneau. Le signal analogique est relié à un convertisseur analogique-numérique (C.A.N.) dans le module, ce qui permet de disposer d'un signal numérique qui sera traité par le microcalculateur.

Le fonctionnement du banc est décrit dans le manuel de l'utilisateur.

Le fonctionnement du convertisseur C.A.N. est dirigé par un signal du calculateur arrivant par la borne de sortie; il place un nombre de 8 bits sur la borne d'entrée pour que le calculateur puisse lire le nombre qui représente la mesure de la température au niveau du capteur.

.Pour la communication entre le microcalculateur et le module d'application MIC 955, les affectations de borne et de bit sont comme suit :

Numéro de bit	Borne A : Entrée	Borne B : Sortie
0	sortie CAN:bit 0=Poids+faible	
1	sortie CAN:bit 1	
.		
.		
5	sortie CAN:bit 5	Commande ventilateur(marche=1)
6	sortie CAN:bit 6	Commande réchauffeur(marche=1)
7	sortie CAN:bit 7=poids+fort	Commande début conversion A-N

Notons que le bit d'entrée de poids le plus faible a une valeur de 0,5°C.

## 2/ Données techniques :

.L'élément de thermomètre à résistance à une résistance de  $100 \pm 0,1$  Ohm à 0°C, et sa variation en fonction de la température est de 385 mOhm/°C.

.Le module utilise les alimentations suivantes :

+12 V à environ 1,2 A

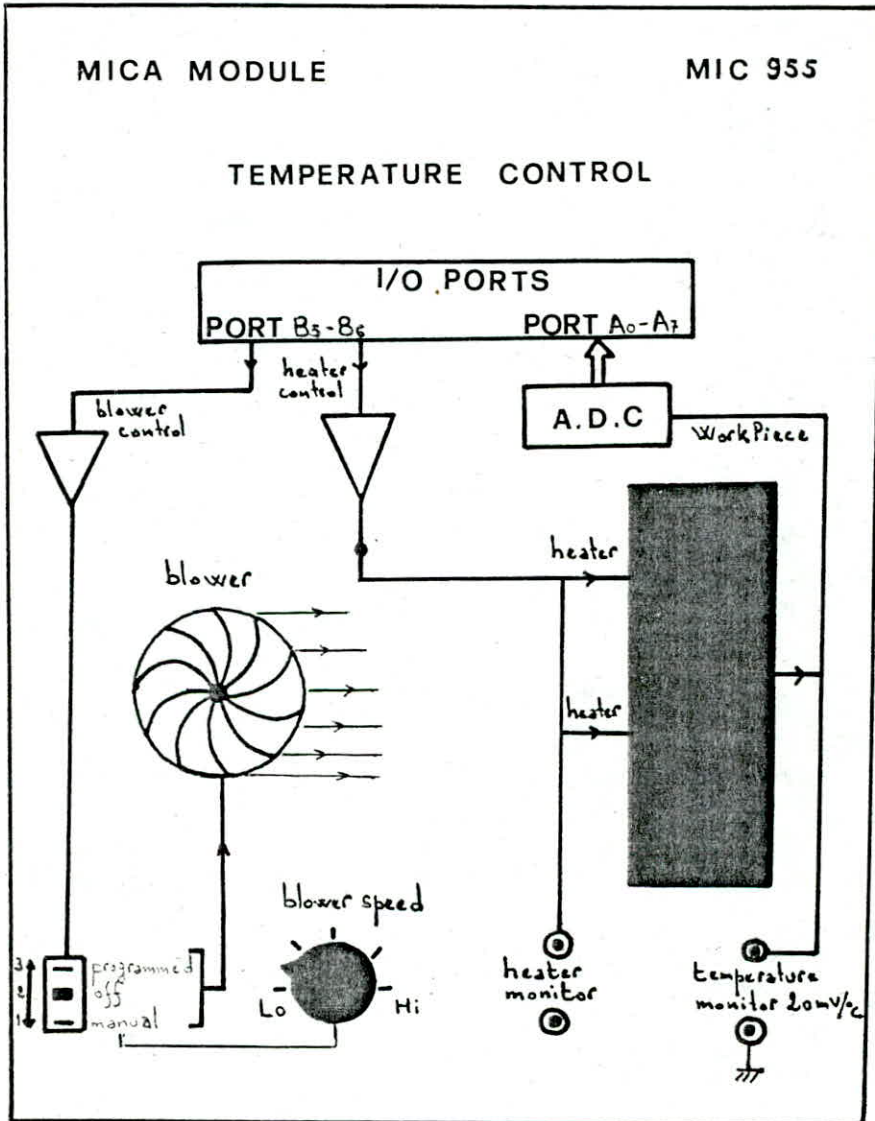
-12 V à quelques mA.

On doit raccorder les fils volants du module aux alimentations externes comme indiqué :

Fil noir : 0 V

Fil rouge : +12 V

Fil bleu : -12 V

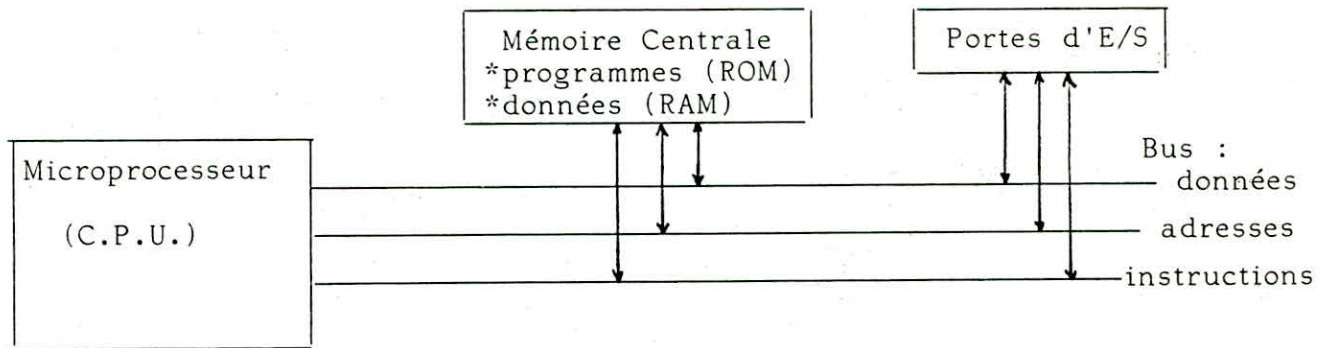


- « COMMANDE DE TEMPERATURE » - fig(14)

B/ Etude du microcalculateur MAT 385 :

Le microcalculateur MAT 385 se présente sous la forme d'un kit. Ce dernier est un ensemble de circuits imprimés, servant de supports aux modules de base (microprocesseur, quelques mémoires et les interfaces). L'ensemble dispose d'interfaces pour certains types de périphériques (lecteur de cassette, module d'applications simples : automatisme). Le microcalculateur MAT 385 possède une architecture semblable à celle décrite au paragraphe I.

En résumé, la structure interne se présente comme suit :



L'unité centrale de traitement (C.P.U.) du MAT 385 est un microprocesseur Intel 8085 à 8 bits, et est utilisé comme langage de programmation : le langage Assembleur.

Physiquement, la mémoire centrale se présente comme une juxtaposition de cases mémoires. Pour sélectionner l'une de ces dernières, il faudra connaître son implantation, c'est-à-dire son adresse.

Le bus des adresses est constitué de 16 lignes à travers lesquelles un mot binaire peut être véhiculé aux mémoires ou aux portes d'E/S. En général, on exprime la taille de la mémoire en Koctets, le nombre K valant  $1024 = 2^{10}$ . La mémoire centrale du MAT 385 a une capacité de 64 K, c'est-à-dire de 0000 à FFFF (hex) (65536 en décimal) et se présente comme c'est indiqué par la figure (15).

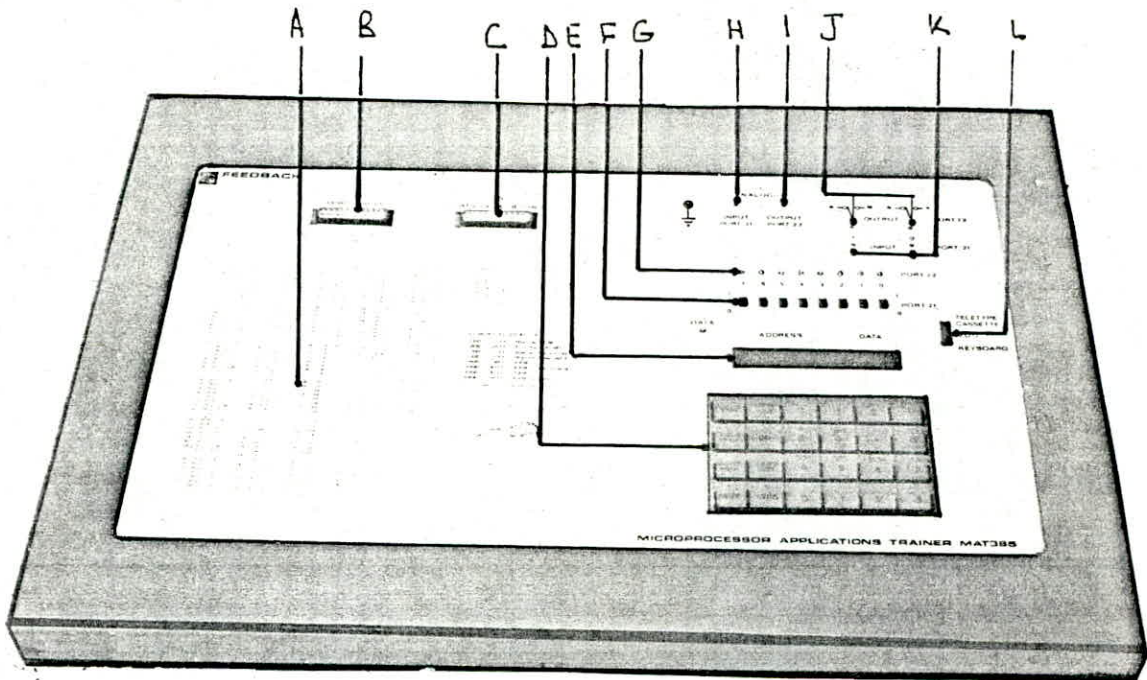
FFFF	not used
8800 87FF	R A M
8400 83FF	R A M non volatile
8000 7FFF	not used
2900 28FF	R A M
2800 27FF	not used
2100 20FF	R A M
2000 1FFF	not used
0800 07FF	R O M
0000	

fig. (15)

\*\* A MEMORY MAP - THE MAT 385 \*\*



DESCRIPTION :



- A : Aide mémoire des principales instructions
- B/C : Portes d'entrées/sortie; branchement de l'interface
- D : Touches des différentes fonctions
- E : Afficheur (en hexadécimal)
- F : Commutateur du registre d'Entrée
- G : Diodes électroluminescentes
- H : Entrée analogique bipolaire
- I : Sortie -----
- J : 2-bit relais à anche (ou à lame souple) de sortie
- K : 2-bit entrée numérique
- L : Sélecteur d'E/S.

fig(16)

N.B. : L'introduction et l'exécution des programmes seront vues dans la partie consacrée aux T.P.

C/ L'interface MIC 948 :

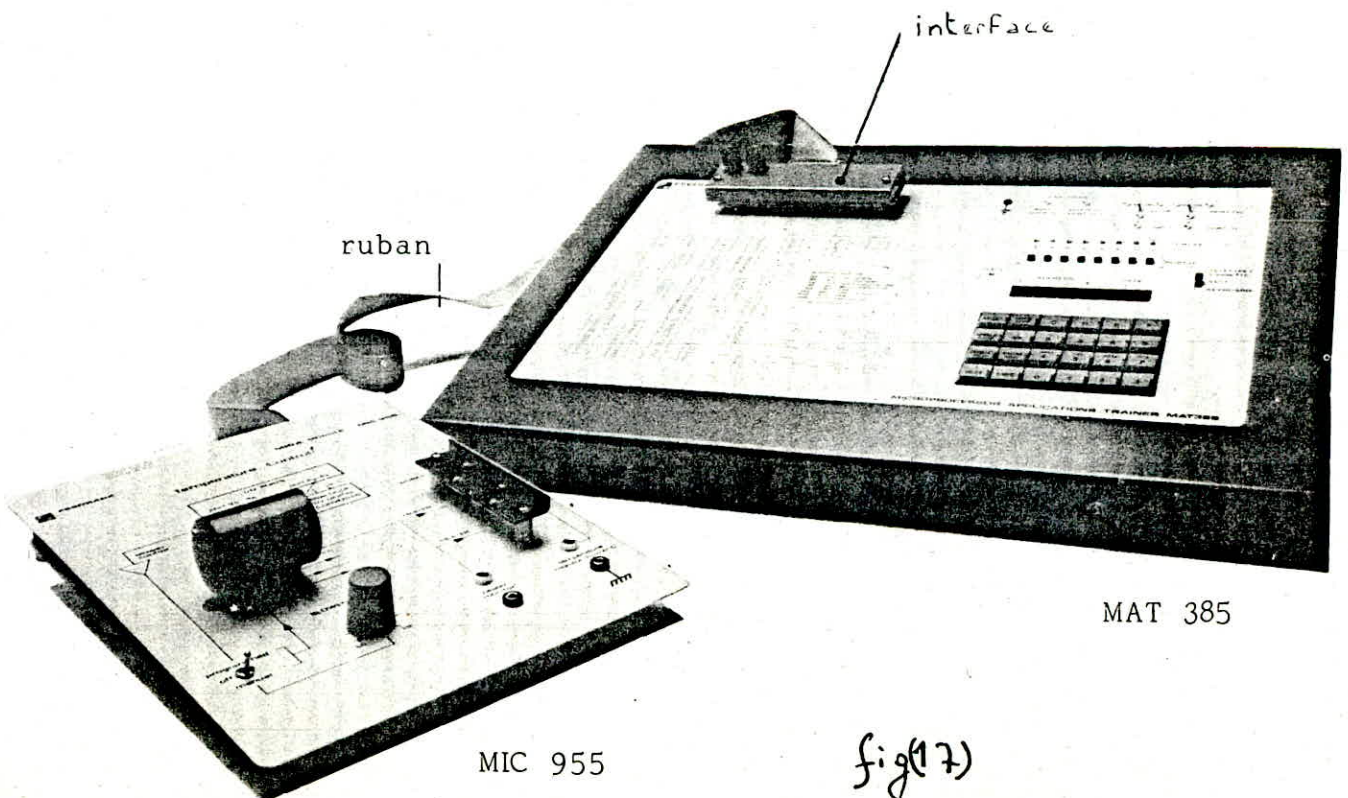
C'est un circuit d'Entrée/Sortie qui assure l'adaptation entre le microprocesseur et l'extérieur qui englobe tous les organes périphériques. D'une façon générale, on peut dire qu'une interface a un double rôle :

- a) transmettre les données en parallèle ou en série, en effectuant l'adaptation nécessaire;
- b) obéir aux signaux de contrôle envoyés soit par le microprocesseur soit par le périphérique.

D'où pour raccorder le module MIC 955 au microcalculateur MAT 385 une interface et un ruban sont nécessaires.

L'interface MIC 948 se branche directement au MAT 385 par les portes d'E/S : B/C (fig. ) et elle est livrée avec un câble ruban dont les extrémités interchangeables comportent des connecteurs. Un connecteur devra être raccordé à l'interface et l'autre au module MIC 955.

En résumé, le banc d'essai se présente comme le montre la figure suivante :



D/ Travaux Pratiques :

Application n°1 :

Addition de 2 nombres à 1 octet :

Adresse	donnée	mnémonique
2000	3E0E	MVI A, OE
2002	D320	OUT 20
2004	3E00	MVI A, 00
2006	D323	OUT 23
2008	AF	XRA
2009	3E**	MVI A, DATA1 (***:1er nombre en hexadécimal)
200B	06**	MVI B, DATA2 (***:2eme nombre)
200D	80	ADD B
(*)200E	D322	OUT 22
2010	76	HLT

.L'affichage du résultat au niveau de l'élément G sera donné en binaire. (\*) Pour que le résultat soit en hexadécimal, il faut mettre après l'instruction ADD B, l'instruction DAA dont le code objet est 27.

.Pour introduire ce programme, la procédure sera comme suit :

\*Initialiser l'adresse à 2000 en utilisant la touche "SUBST-ME ";

\*Appuyer sur la touche "NEXT"

\*Introduire les données 2 par 2. Par exemple on fait rentrer 3E, après on appuie sur la touche "NEXT" et l'écran des adresses affichera automatiquement 2001 c'est-à-dire l'adresse suivante.

\*Et ainsi de suite pour les autres données jusqu'à la fin du programme. (Après chaque introduction de donnée, il faut appuyer sur la touche "NEXT").

.Pour exécuter ce programme, on appuie successivement sur les touches "EXEC" et "GO" et on initialise l'adresse à partir de laquelle on veut exécuter le programme (par exemple: 2000). A la fin on appuie sur la touche "EXEC".

Application n°2 : Thermométrie Numérique :

.Cette application sert à introduire le module de thermostat MIC 955.

L'objectif est simplement de lire la température mesurée par le capteur et de la présenter sur l'affichage du microcalculateur.

Pour accomplir cela, il faut effectuer une conversion analogique

en numérique, lire les données et les afficher.

.Les données de base nécessaires pour cette application sont :

1-bit de déclenchement du C.A.N. inscrit sur la borne de sortie du microcalculateur.

2-octet de donnée du C.A.N. lu sur la borne d'entrée du microcalculateur pour que ce dernier puisse lire le nombre qui représente la mesure de la température au capteur.

.Procédure :

-Déclaration des données,

-Initialisation

-Lire la valeur du C.A.N.,

-La valeur de la température = 1/2 la valeur du C.A.N. car l'échelle générale de la valeur du C.A.N. est 1/2°C par bit,

-Afficher la température.

### Application n°3 : "Cycle limite"

.Cette application utilise toutes les fonctions (capteur, réchauffeur et ventilateur) du module de thermostat MIC 955.

.Objectif : faire varier la température entre deux valeurs limites qui sont définies par des entrées et ceci soit au moyen du réchauffeur pour élever la température, ou du ventilateur pour l'abaisser. On devra afficher la température.

.Les données de base nécessaires comprennent les deux limites de température, les données de l'application (2) et les signaux qui commandent le réchauffeur et le ventilateur.

.Procédure :

-Déclarations des données,

-Initialisation,

-Entrer limite haute et limite basse de température,

-Lire C.A.N. (valeur du C.A.N.: commande : réchauffeur et ventilateur),

-Température = 1/2 valeur du C.A.N.,

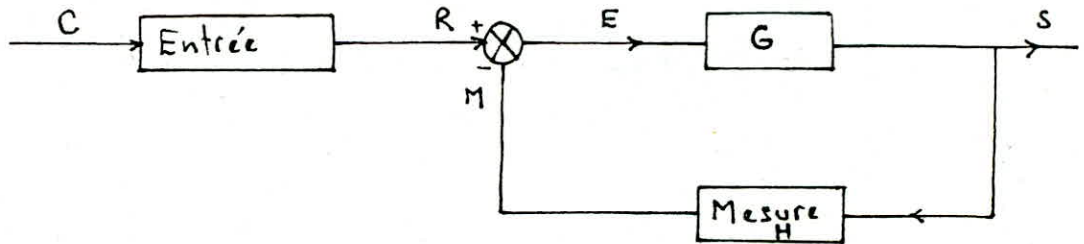
-Afficher la valeur de la température. Si  $T \gg$  haute t° : couper le réchauffeur et brancher le ventilateur,

-Si la température est  $\ll$  basse t° : couper le ventilateur et brancher le réchauffeur.

Application No 4: "Régulateur Proportionnel"

Un régulateur est un asservissement particulier dans lequel la valeur de la grandeur de référence est imposée; les variations de la grandeur de sortie en fonction de perturbations normales ou accidentelles apportées au système doivent alors être réduites le plus possible.

Le schéma d'un régulateur se présente de la façon suivante:



La régulation du processus peut être obtenue en mesurant une grandeur qui représente le résultat désiré (grandeur réglée S) et en réglant automatiquement une des autres grandeurs (grandeur E).

C est la valeur de consigne ou valeur prescrite. Un dispositif d'entrée la transforme en valeur de référence R (de nature physique généralement différente). Un dispositif de mesure transforme la grandeur réglée S en une grandeur M de même nature que R.

L'erreur  $E = R - M$  (  $M = H \cdot S$  ) agit sur le système de régulation automatique G. Ce dernier, sous l'effet du signal E, exerce une puissance sur un dispositif permettant d'ajuster la valeur de l'erreur.

On appelle écart la différence  $\xi = S - C$ .

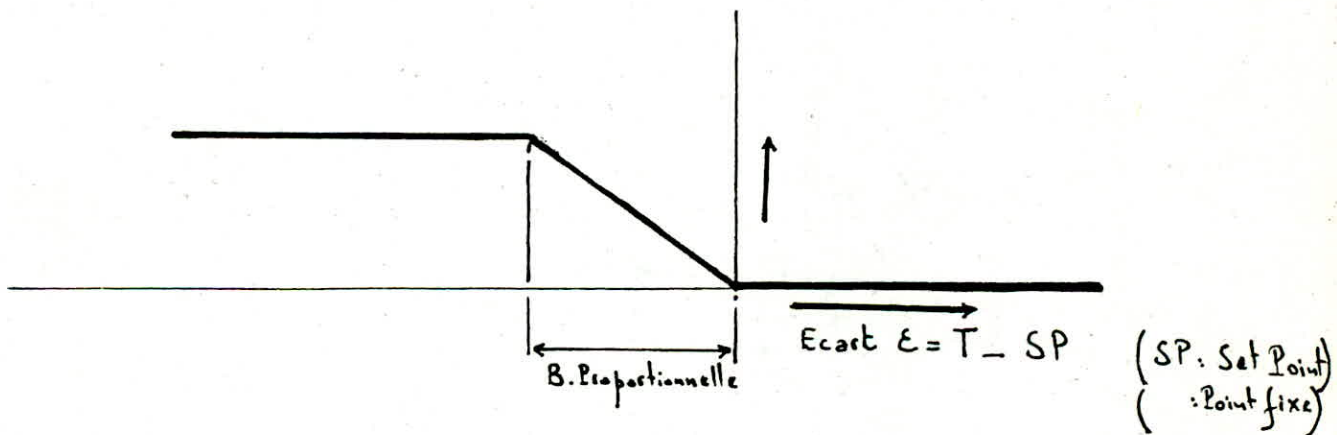
-Objectif: La température peut être maintenue à une certaine valeur en branchant et en coupant l'alimentation du réchauffeur. Mais ce type de régulation entraîne toujours des oscillations indésirables autour de la valeur fixée.

Pour avoir une température stable ,il vaut mieux faire varier progressivement le courant lorsque la température s'écarte de la valeur fixée.

La plus simple régulation convenable à ce cas est appelée :Régulation Proportionnelle dont l'équation caractéristique est : $E = K_p.S$  où  $K_p$  représente le coefficient de proportionnalité.

Dans ce cas ,le courant fourni au réchauffeur est proportionnel à l'écart de température. Il y a toujours une limite au courant disponible ,aussi cette propotionnalité n'existe que sur un intervalle de température qui est appelé "la bande proportionnelle".

La Fig(18) montre les caractéristiques de ce type de régulation.



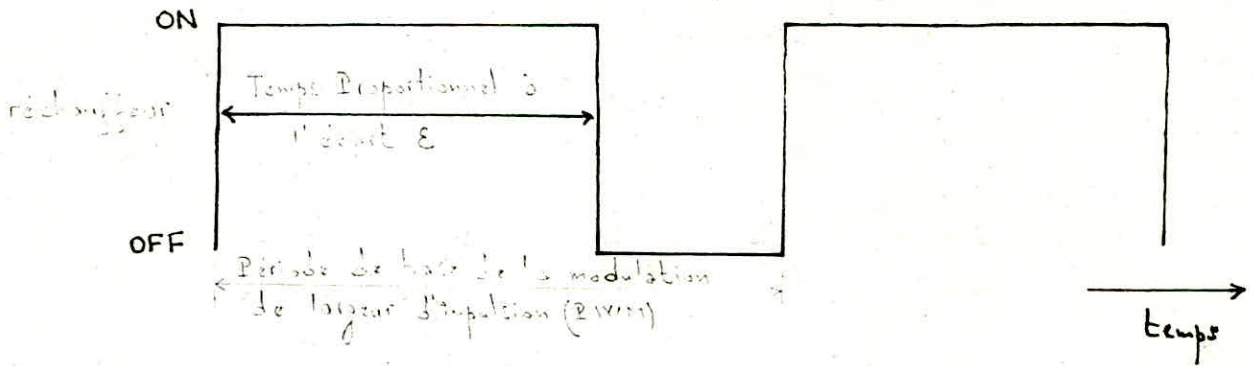
fig(18)

Réf(9)

On peut assurer la régulation proportionnelle de la sortie du réchauffeur par modulation de la largeur des impulsions(PWM)

Dans ce schéma ,le rapport marqué à espace d'une impulsion de commutation est rendu proportionnel à l'écart de température.

La fraction de temps pendant laquelle le réchauffeur est branché est une fonction linéaire de l'écart; donc, la puissance moyenne du réchauffeur est proportionnelle à l'écart. Ceci est illustré par la Fig suivante:



\*Les données de bases sont:

- 1-Bit de déclenchement du C.A.N,
- 2-Bit de commutation du réchauffeur ,
- 3-Octet de donnée du C.A.N lu sur la borne d'entrée du microcalculateur.

\*Procédure:

- \_Déclaration des données ,
- Initialisation des bornes,
- Initialisation des variables,
- Lire valeur du C.A.N et couper ou brancher le réchauffeur,
- Affichage,
- Entrer valeur fixée.

\* STRUCTURE DU PROGRAMME SUIVANT:

Le programme principal est formé d'appels de sous-programmes qui exécutent les différentes procédures, avec une instruction de saut déclenchant l'exécution jusqu'à la boucle.

Les différentes procédures que renferme le programme sont :

-Procédure d'initialisation des bornes:

Pour chaque sens de communication entre le micro et le module, on envoie à un moment donné un Octet(8bits) qui passe à travers une borne du micro. Chaque borne peut fonctionner en entrée ou en sortie, selon les indications que doit fournir le programme. C'est ce qu'on appelle l'initialisation des bornes.

-Procédure d'initialisation des variables(inutile pour le programme de thermomètre):

Elle permet de fixer l'ordre initial pour réchauffeur ou ventilateur.

-Procédure d'affichage de la valeur fixée(nécessaire que pour le programme de commande proportionnelle).

-Procédure de conversion analogique en numérique.

-Procédure de prise de décision(pour l'application de cycle limite de température).-

-Procédure de la commande proportionnelle.

Réf(9)

NB: Le détail du programme suivant se trouve dans la partie Annexe.



MODULE MIC955 THERMOSTAT

```

LOC  OBJ      LINE      SOURCE STATEMENT
1  *TITLE(' PROGRAMS FOR TEMPERATURE CONTROL MODULE 955 ')
2  NAME      TEM385
3  ;
4  ; PROGRAM BY P. D. TURNER
5  ; COPYRIGHT FEEDBACK INSTRUMENTS
6  ; APRIL 1981
7  ; -----
8  ; THIS PROGRAM IS NOT DESIGNED TO BE EXECUTED IN ITS
9  ; ENTIRETY. THE MAIN PROGRAM CALLS A NUMBER OF OPTIONAL
10 ; SUBROUTINES, ANNOTATED IN THE MAIN PROGRAM WITH '*'
11 ; FOLLOWED BY 'C' IF REQUIRED FOR CYCLE PROGRAM, AND
12 ; 'P' IF REQUIRED FOR THE PROPORTIONAL CONTROL PROGRAM.
13 ; THESE CALLS SHOULD BE REPLACED BY 'NOP' INSTRUCTIONS
14 ; (I.E 00 SHOULD BE ENTERED INTO THE THREE
15 ; LOCATIONS OF EACH CALL) WHEN THEY ARE NOT REQUIRED.
16 ; -----
17 ; DECLARATIONS
18 ;
02B7  19  OUTPT  EQU    02B7H  ; OUTPUT CHARACTERS TO DISPLAY
036E  20  UPDDT  EQU    036EH  ; UPDATE DATA DISPLAY
05F1  21  DELAY  EQU    05F1H  ; MONITOR DELAY ROUTINE
20FE  22  IBUFF  EQU    20FEH  ; INPUT BUFFER LOADED BY THE
23 ; KEYBOARD INPUT ROUTINE IN
24 ; MONITOR.
0008  25  BAND   EQU    08H    ; PROPORTIONAL BAND SETTING
26 ;
2000  27  HILIM  EQU    2000H  ; SET HIGH LIMIT
2001  28  LOLIM  EQU    2001H  ; SET LOW LIMIT
2003  29  SETVAL  EQU    2003H  ; SET VALUE (TEMP SETPOINT)
30 ; SET-VALUE DISPLAY BUFFER
2004  31  SVBUF  EQU    2004H  ; HOLDS CHARACTER 'S'
32 ; HOLDS 'P'
2006  33  MSDGT  EQU    2006H  ; HOLDS FIRST DIGIT OF SP
2007  34  LSDGT  EQU    2007H  ; HOLDS SECOND DIGIT OF SP.
35 ; PROPORTIONAL CONTROL COUNTERS
2008  36  HTRFAN EQU    2008H  ; HEATER, FAN CONTROL BITS
2009  37  CYTCT  EQU    2009H  ; CYCLE-TIME COUNTER (RESET
38 ; TO 'BAND' ONCE PER CYCLE)
39 ; HOLDS ON-TIME COUNTER
40 ; *****
41 ;
8000  42  ORG    8000H  ; START ADDRESS
43 ;
44 ; -----
45 ;

```

```

LOC  OBJ      LINE      SOURCE  STATEMENT
8000 31C220    46 START: LXI      SP,20C2H: LOAD STACK POINTER
8003 CD78E0    47          CALL     INIT      ; INITIALISE PORTS, INTERRUPT
8006 CD9380    48          CALL     INITVR   ; INITIALISE VARIABLES
8009 CD9E80    49 LOOP:   CALL     DISPSV ; DISPLAY SET-VALUE *P
800C CDBF80    50          CALL     RDADC    ; READADC
800F B7         51          ORA      A          ; CLEAR AC AND C FLAGS
8010 1F         52          RAR      A          ; TEMP = ADCVALUE/2
8011 F5         53          PUSH    PSW        ; SAVE TEMP
8012 CD6E03    54          CALL     UPDDT    ; DISPLAY TEMP
8015 F1         55          POP     PSW        ; RESTORE TEMP TO A
+ 8016 CDEE80    56          CALL     DECIDE   ; DECIDE HEATER ON/OFF *C
+ 8019 CD2D81    57          CALL     PROP     ; DECIDE HEATER ON/OFF *P
* 801C CD8781    58          CALL     ENTRSV   ; ENTER-SET-VALUE *P(OPTIONAL
                    59          ;/EXTRA FEATURE FOR USE ONLY IN
                    60          ;/PROPORTIONAL CONTROL PROGRAM)
801F 110100    61          LXI      D,01H     ; DELAY PARAMETER (NOMINAL)
8022 CDF105    62          CALL     DELAY    ; --CAN BE INCREASED AT WILL.
8025 C30980    63          JMP     LOOP      ; REPEAT FOR EVER
                    64 ;
                    65 ;***** END OF MAIN PROGRAM *****
                    66 ;
8028          67          DS      50H      ; LEAVE SPACE FOR MODIFICATION'S
                    68 ;
                    69 ;----- PROCEDURE INITIALISE PORTS, INTERRUPTS -----
8078 3E02       70 INIT:   MVI     A,02H  ; INITIALISE PORTS
807A D328       71          OUT     28H      ; 2A = O/P . 2B & 29 = I/P
807C 32FF20    72          STA     20FFH     ; REQUIRED FOR SINGLE-STEPPING
807F 3E0E       73          MVI     A,0EH    ; UNMASK RST5.5 INTERRUPT -
                    74          ;/(ONLY USED FOR OPTIONAL
                    75          ;/ENTER-SET-VALUE ROUTINE)
8081 30         75          SIM
8082 C9         76          RET
8083          77          DS      10H      ; SPACE FOR MODIFICATIONS
                    78 ;
                    79 ;----- PROCEDURE INITVARIABLES -----
80          80 ;
8093 210820    81 INITVR: LXI     H,HTRFAN; SET HL AS POINTER
8096 3E20       82          MVI     A,20H    ; TURNS FAN ON INITIALLY;
                    83          ;/SUBSTITUTE 00 FOR FAN OFF
8098 77         84          MOV     M,A       ; STORE IN HTRFAN
8099 23         85          INX     H        ; HL POINTS TO CYTCT
809A 3E08       86          MVI     A,BAND    ; SET IT FOR START OF PROP CYCLE
809C 77         87          MOV     M,A
809D C9         88          RET              ; END INITVARIABLES
                    89 ;
                    90 ;----- PROCEDURE DISPLAY-SET-VALUE-----
                    91 ; INPUTS -- SET-VALUE
                    92 ; OUTPUTS-- TWO-DIGIT DISPLAY IN ADDRESS FIELD
                    93 ;
                    94 ;
809E E5         95 DISPSV: PUSH    H          ; SAVE REGISTERS
809F C5         96          PUSH    B
80A0 D5         97          PUSH    D
80A1 210720    98          LXI     H, LSDGT  ; SET POINTER
80A4 3A0320    99          LDA     SETVAL   ; GET SET-VALUE
80A7 47        100         MOV     B,A       ; COPY SET-VALUE TO B
80A8 E50F     101         ANI     0FH      ; MASK OUT M. S. DIGIT
80AA 77        102         MOV     M,A       ; STORE L. S. DIGIT
80AB 2B        103         DCX     H        ; POINT TO MSDGT
80AC 78        104         MOV     M,B       ; RESTORE SET-VALUE
80AD 0F        105         RRC      A        ; GET MS 4 BITS TO LS POSITIONS
80AE 0F        106         RRC
80AF 0F        107         RRC
80B0 0F        108         RRC

```

```

LOC  OBJ      LINE      SOURCE  STATEMENT
80B1  E60F      109      ANI     0FH      ; DELETE UNWANTED BITS
80B3  77         110      MOV     M, A      ; MS DIGIT STORED
80B4  2B         111      DCX     H
80B5  2B         112      DCX     H      ; HL POINTS TO SVBUF
80B6  97         113      SUB     A      ; CLEAR A, USE ADDRESS FIELD
80B7  47         114      MOV     B, A      ; CLEAR B (NO DOT REQUIRED)
80B8  C0B702     115      CALL   OUTPT     ; DISPLAY BUFFER CONTENTS
80B8  D1         116      POP     D      ; RESTORE REGISTERS
80BC  C1         117      POP     B
80BD  E1         118      POP     H
80BE  C9         119      RET
120 ;
121 ;----- PROCEDURE READ-ADC -----
122 ;
123 ; THIS ROUTINE STARTS A-D CONVERSION
124 ; AND RETURNS ADCVALUE WHEN THE CONVERSION
125 ; IS COMPLETE
126 ; INPUTS -- HEATER CONTROL BIT FROM HTRFAN
127 ; OUTPUTS -- ADCVALUE IN REGISTER A.
128 ; I/O PORTS USED
129 ;
80BF  C5         130  RDADC:  PUSH    B      ; SAVE BC REGISTERS
80C0  7A0820     131      LDA     HTRFAN    ; GET HEATER-CONTROL BIT
80C3  47         132      MOV     B, A      ; SAVE COPY
80C4  3E00      133      MVI    A, 80H    ; START A-D CONVERSION
80C6  B0         134      ORA     B      ; GET HEATER-CONTROL-BIT
80C7  D32A      135      OUT    2AH      ; SEND TO OUTPUT PORT
80C8  3E00      136      MVI    A, 00     ; TRIGGER LINE LOW
80CB  B0         137      ORA     B
80CC  D32A      138      OUT    2AH
80CE  3E00      139      MVI    A, 80H    ; TRIGGER LINE HIGH.
80D0  B0         140      ORA     B
80D1  D32A      141      OUT    2AH
80D3  DB2B      142  COMPL:  IN      2BH      ; READ PORT C
80D5  E604      143      ANI    04H      ; CONVERSION COMPLETE (BIT2=1)?
80D7  CAD380    144      JZ     COMPL     ; IF NO THEN INSPECT BIT 2 AGAIN
80DA  DB29      145      IN      29H      ; ELSE GET ADCVALUE INTO A.
80DC  C1         146      POP     B      ; RESTORE OLD BC REGISTERS
80DD  C9         147      RET      ; END READ-ADC.
80DE  148      DS      10H
149 ;
150 ;----- PROCEDURE DECIDE -----
151 ; INPUTS -- HI-LIMIT, LO-LIMIT, TEMP IN A REGISTER
152 ; OUTPUTS-- HEATER-FAN CONTROL BITS STORED
153 ;
80EE  E5         154  DECIDE:  PUSH    4
80EF  210020     155      LXI    H, HILIM
80F2  9E         156      CMP     M      ; COMPARE HILIM & TEMP
80F3  D21081     157      JNC    COOL    ; IF T)=HILIM THEN COOL
80F6  23         158      INX     H      ; HL POINTS TO LOLIM
80F7  BE         159      CMP     M      ; COMPARE LOLIM AND TEMP
80F8  DA1581     160      JC     HEAT    ; IF T<LOLIM THEN HEAT
80FB  CA1581     161      JZ     HEAT    ; IF T=LOLIM THEN HEAT
80FE  E1         162  DEC1:   POP     H      ; RESTORE OLD HL
80FF  C9         163      RET      ; END DECIDE
8100  164      DS      10H
165 ;
8110  3E20      166  COOL:   MVI    A, 20H    ; HEATER OFF, FAN ON
8112  C31781     167      JMP    DEC2
8115  3E40      168  HEAT:   MVI    A, 40H    ; HEATER ON, FAN OFF
8117  320B20     169  DEC2:   STA    HTRFAN    ; STORE CONTROL BITS
811A  C3FE80     170      JMP    DEC1

```

```

LOC  OBJ      LINE      SOURCE  STATEMENT
8110                      171      DS      10H
                      172 ;
                      173 |----- PROCEDURE PROPORTIONAL CONTROL-----|
                      174 ;
                      175 ; INPUTS -- TEMP(IN A REGISTER), SET-VALUE
                      176 ;         ON-TIME-COUNT, CYCLE-TIME-COUNT
                      177 ; OUTPUTS-- HEATER ON BIT = BIT 6 OF A REGISTER,
                      178 ;         ON-TIME-COUNT, CYCLE-TIME-COUNT
                      179 ;
8120  C5      180  PROP:  PUSH   B      ; SAVE B AND C REGISTERS
812E  E5      181          PUSH   H      ; AND H AND L REGISTERS
812F  4F      182          MOV    C,A    ; SAVE TEMP IN C
8130  210920  183          LXI   H,CYCT ; HL POINTS TO CYCLE TIME CTR
8133  35      184          DCR   M      ; DECREMENT COUNTER
8134  CC5681  185          CZ    RESET ; IF END OF COUNT RESET CTRS
9137  3E00    186          MVI   A,0    ; CLEAR A REGISTER (HTR OFF)
9139  23      187          INX   H      ; HL POINTS TO ON TIME CTR
813A  35      188          DCR   M      ; DECREMENT IT
813B  FA4081  189          JM    STHF  ; IF ON TIME EXCEEDED, HTR OFF
813E  3E40    190          MVI   A,40H  ; ELSE SET HTR-ON BIT
9140  320820  191  STHF:  STA   HTRFAN ; STORE HEATER-ON BIT
8143  E1      192          POP   H      ; RESTORE SAVED HL
8144  C1      193          POP   B      ; RESTORE SAVED BC
8145  C9      194  DCEND: RET    ; END ON-OFF
8146                      195      DS      10H
                      196 ;
                      197  RESET:  ; RESET CYCLE AND ON TIME COUNTERS
                      198          ; ENTER WITH HL POINTING TO CYCLE-TIMER
                      199          ; AND TEMP IN C.
8156  3A0320  200          LDA   SETVAL ; GET SET VALUE IN REGISTER A
8159  FC7481  201          CM   SAT   ; SATURATE AT 75(HEX)
815C  91      202          SUB   C      ; CALCULATE (SV-TEMP)
815D  4F      203          MOV   C,A    ; AND SAVE IT IN C
815E  3E08    204          MVI   M,BAND ; PROPORTIONAL BAND
8160  23      205          INX   H      ; POINT TO ON-TIME CTR
8161  71      206          MOV   M,C    ; LOADED WITH (SV-TEMP)
8162  2B      207          DCX   H      ; HL AS ON ENTRY
8163  C9      208          RET    ; RETURN TO PROP ROUTINE
8164                      209      DS      10H
8174  3E7F    210  SAT:   MVI   A,7FH
8176  C9      211          RET
8177                      212      DS      10H
                      213 ;
                      214 |----- PROCEDURE ENTER-SET-VALUE -----|
                      215 ; *** THIS IS AN OPTIONAL ROUTINE NOT INCLUDED IN THE
                      216 ; EXERCISES SET OUT FOR THE STUDENT. MAKES A BETTER
                      217 ; DEMONSTRATION, AND COULD BE SET AS FURTHER WORK.
                      218 ; DATA ENTERED THROUGH THE KEYBOARD BY MEANS OF
                      219 ; THE MONITOR'S ININT ROUTINE IS SHIFTED LEFT
                      220 ; INTO THE SET VALUE.
                      221 ;
8187  FB      222  ENTRSV: EI      ; ENABLE INTERRUPTS
8188  3AFE20  223          LDA   Ibuff ; GET INPUT BUFFER
818B  B7      224          ORA   A      ; TEST IT
818C  FA9F81  225          JM    ENT1   ; IF BIT 7 IS SET (BUFFER IS
                      226          ; EMPTY) THEN RETURN.
818F  C5      227          PUSH  B      ; ELSE SAVE B,C
8190  4F      228          MOV   C,A    ; SAVE DIGIT IN C
8191  3A0320  229          LDA   SETVAL ; GET OLD SETVALUE
8194  07      230          RLC   ; SHIFT LEAST SIGNIFICANT HALF
8195  07      231          RLC   ; INTO MOST SIGNIFICANT
8196  07      232          RLC   ; POSITION
8197  07      233          RLC

```

```

LOC  OBJ      LINE      SOURCE  STATEMENT
8198  E6F0      234      ANI     0F0H      ; DELETE UNWANTED BITS
819A  B1         235      ORA     C        ; BRING IN NEW L.S DIGIT
819B  J20320     236      STA     SETVAL  ; STORE NEW SET-VALUE
819E  C1         237      POP     B
819F  3E80      238 ENT1:  MVI     A,80H   ; SET EMPTY FLAG OF Ibuff
81A1  J2FE20     239      STA     Ibuff
81A4  C9         240      RET
      241 ;
      242 ;
      243 ;
      244      END
*****

```

PUBLIC SYMBOLS

EXTERNAL SYMBOLS

USER SYMBOLS

BAND	A 0008	COMPL	A 80D3	COOL	A 8110	CYTCT	A 2009
DCEND	A 8145	DEC1	A 80FE	DEC2	A 8117	DECIDE	A 80EE
DELAY	A 05F1	DISPSV	A 809E	ENT1	A 819F	ENTRSV	A 8187
HEAT	A 8115	HILIM	A 2000	HTRFAN	A 2008	IBUFF	A 20FE
INIT	A 8078	INITVR	A 8093	LOLIM	A 2001	LOOP	A 8009
LSDGT	A 2007	MSDGT	A 2006	OUTPT	A 02B7	PRDP	A 812D
RDADC	A 80BF	RESET	A 9156	SAT	A 9174	SETVAL	A 2003
START	A 8000	STHF	A 8140	SVBUF	A 2004	UPDDT	A 035E

ASSEMBLY COMPLETE, NO ERRORS

# CONCLUSION

## VIII/ CONCLUSION :

L'étude du module d'application MIC 955 conçu comme exercice pratique microcalculateur, et la manipulation d'un matériel diversifié, nous ont permis d'approfondir nos connaissances théoriques, d'élargir nos horizons pratiques et d'avoir ainsi une meilleure idée sur les applications d'un thermostat.

Si ce n'était pas la défaillance du microcalculateur MAT 385, qui nous a posé de sérieux problèmes lors de la réalisation de certaines applications, les résultats de ces dernières auraient pu être d'un grand intérêt dans un ordre pédagogique car elles permettraient l'initiation aux systèmes de régulation et de commande numérique d'une manière simple mais efficace.

Cependant, malgré les résultats que nous estimons appréciables, dans la mesure où nous avons pu voir beaucoup de choses (comment mesurer numériquement une  $t^\circ$ , structure d'un kit,...), un perfectionnement du système est à entreprendre afin d'exploiter le MIC 955 au maximum de ses capacités.

L'étude faite nous a montré que la principale amélioration porte sur l'utilisation d'un autre type de microcalculateur dont le langage de programmation est plus utilisé par les mécaniciens tels que le BASIC ou le FORTRAN et dont les erreurs de programmation peuvent être signalées, car l'inconvénient du langage assembleur c'est qu'il est propre à un seul type de micro et demande une étude très approfondie pour son assimilation, chose qui n'est pas enseignée aux étudiants mécaniciens.

Nous sommes persuadés qu'avec cette amélioration, un tel module d'application peut être exploité de manière très utile.

Pour cette raison, nous souhaitons vivement que ce travail soit poursuivi car le module d'application MIC 955 pour thermostat peut être adapté à d'autres types de microcalculateurs pour lesquels existent des interfaces de la série MIC 940 et des programmes en BASIC. (9)

Signalons aussi que la MAT 385 peut être utilisée pour commander un certain nombre de simulateurs tels que : le simulateur de processus PCS 327 et le simulateur de processus thermique PT 326 (7).

**BIBLIOGRAPHIE  
ET  
ANNEXES**



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

- (1) "Introduction aux circuits logiques" (518.5 LETOCHA)
- (2) "Electronique Numérique" (Marcel Gindre; Mc Grawhill)
- (3) "Instrumentation industrielle Tome 1" (Michel Cerr)
- (4) "L'emploi des microprocesseurs" (681.3 Amiaux)
- (5) "Microprocesseurs et microordinateurs" (J.M. Crozet)
- (6) "Les microprocesseurs T1 et T2" (Roger L. Tokheim)
- (7) Les manuels de la firme Feedback: MAT 385
- (8) "The MCS 80/85 Family : User's Manual"
- (9) "Module d'application dumicrocalculateur MIC 955 pour thermostat"  
(Feedback)
- (10)"Les capteurs en instrumentation industrielle" (621.336 G.Asch)
- (11)"Mesures et contrôles industriels" (621.7.08 FERRITTI)
- (12)"Thermomètre Numérique 16 voies pour thermocouple"  
(Thèse de MM. GUEDIMI R.,BOUKHTACHE, département Génie Elec-  
tronique, 84)
- (13)"Logiciel pour la C.N. par calculateur" (Thèse de Mr DJOUDER(86)  
département Génie Electrotechnique)
- (14)"Catalogue général des capteurs de température (Pyro-contrôle)
- (15) Cours de régulation.

--ooOOoo-- ANNEXE --ooOO--

\* Définition des différentes mémoires :

-RAM (Random Access Memory)ou mémoire à accès aléatoire: Elle permet:  
.l'enregistrement des données mot par mot,.une mémorisation globale des mots inscrits,.la lecture de mots.

Le maintien des informations impose la permanence de source d'alimentation.

Elle comporte des bornes d'adresse du mot choisi,d'entrée écriture, de sortie lecture,d'ordres de lire ou d'écrire et d'alimentation.

- ROM(READ ONLY MEMORY)ou mémoire à lecture contenant les programmes non modifiables par l'utilisateur.

Cette dernière peut également être une PROM c'est-à-dire une ROM programmable par l'utilisateur ou une REEPROM programmable et reprogrammable après effacement.Elle est donc intermédiaire entre une RAM et une PROM.

- Pour lire ou écrire un mot dans la RAM il faut :

.Envoyer l'adresse par le bus d'adresse,

.Envoyer un signal de lecture ou d'écriture.

C'est le microprocesseur qui effectue cette dernière fonction lorsqu'il a reconnu une instruction de lecture ou d'écriture.

.La donnée est véhiculée par le bus de données.

- Dans le cas d'une ROM le processus est le même mais la seule opération possible est une lecture.

SIGNIFICATION DES MNEMONIQUES ET DUREE DES INSTRUCTIONS  
(durée exprimée en périodes d'horloge)  
Microprocesseur INTEL 8080 A ou 8085 A

ACI	Add immediate to A with carry	7	MVI M	Move immediate memory	10
ADC M	Add memory to A with carry	7	MVI r	Move immediate register	7
ADC r	Add register to A with carry	4	MOV M, r	Move register to memory	7
ADD M	Add memory to A	7	MOV r, M	Move memory to register	7
ADI	Add immediate to A	7	MOV		
ANA M	And memory with A	7	r1, r2	Move register to register	5
ANA r	And register with A	4	NOP	No-operation	4
ANI	And immediate with A	7	ORA M	Or memory with A	7
CALL	Call unconditional	17	ORA r	Or register with A	4
CC	Call on carry	11/17	ORI	Or immediate with A	7
CM	Call on minus	11/17	OUT	Output	10
CMA	Compliment A	4	PCHL	H & L to program counter	5
CMC	Compliment carry	4	POP B	Pop register pair B & C off stack	10
CMP M	Compare memory with A	7	POP D	Pop register pair D & E off stack	10
CMP r	Compare register with A	4	POP H	Pop register pair H & L off stack	10
CNC	Call on no carry	11/17	POP PSW	Pop A and Flags off stack	10
CNZ	Call on no zero	11/17	PUSH B	Push register pair B & C on stack	11
CP	Call on positive	11/17	PUSH D	Push register pair D & E on stack	11
CPE	Call on parity even	11/17	PUSH H	Push register pair H & L on stack	11
CPI	Compare immediate with A	7	PUSH		
CPO	Call on parity odd	11/17	PSW	Push A and Flags on stack	11
CZ	Call on zero	11/17	RAL	Rotate A left through carry	4
DAA	Decimal adjust A	4	RAR	Rotate A right through carry	4
DAD B	Add B & C to H & L	10	RC	Return on carry	5/11
DAD D	Add D & E to H & L	10	RET	Return	10
DAD H	Add H & L to H & L	10	RLC	Rotate A left	4
DAD SP	Add stack pointer to H & L	10	RM	Return on minus	5/11
DCR M	Decrement memory	10	RNC	Return on no carry	5/11
DCR r	Decrement register	5	RNZ	Return on no zero	5/11
DCX B	Decrement B & C	5	RP	Return on positive	5/11
DCX D	Decrement D & E	5	RPE	Return on parity even	5/11
DCX H	Decrement H & L	5	RPO	Return on parity odd	5/11
DCX SP	Decrement stack pointer	5	RRC	Rotate A right	4
DI	Disable Interrupt	4	RST	Restart	11
EI	Enable Interrupt	4	RZ	Return on zero	5/11
HLT	Halt	7	SBB M	Subtract memory from A with borrow	7
IN	Input	10	SBB r	Subtract register from A with borrow	4
INR M	Increment memory	10	SBI	Subtract immediate from A with borrow	7
INR r	Increment register	5	SHLD	Store H & L direct	16
INX B	Increment B & C registers	5	SPHL	H & L to stack pointer	5
INX D	Increment D & E registers	5	STA	Store A direct	13
INX H	Increment H & L registers	5	STAX B	Store A indirect	7
INX SP	Increment stack pointer	5	STAX D	Store A indirect	7
JC	Jump on carry	10	STC	Set carry	4
JM	Jump on minus	10	SUB M	Subtract memory from A	7
JMP	Jump unconditional	10	SUB r	Subtract register from A	4
JNC	Jump on no carry	10	SUI	Subtract immediate from A	7
JNZ	Jump on no zero	10	XCHG	Exchange D & E, H & L, registers	4
JP	Jump on positive	10	XRA M	Exclusive Or memory with A	7
JPE	Jump on parity even	10	XRA r	Exclusive Or register with A	4
JPO	Jump on parity odd	10	XRI	Exclusive Or immediate with A	7
JZ	Jump on zero	10	XTHL	Exchange top of stack, H & L	18
LDA	Load A direct	13			
LDAX B	Load A indirect	7			
LDAX D	Load A indirect	7			
LHLD	Load H & L direct	16			
LXI B	Load immediate register Pair B & C	10			
LXI D	Load immediate register Pair D & E	10			
LXI H	Load immediate register Pair H & L	10			
LXI SP	Load immediate stack pointer	10			



Ligne	Source	Instruction
1		Titre ( Programmes pour Module MIC955 Thermostat)
8		Ce programme n'est pas destiné à être exécuté dans son intégralité. Le programme principal appelle plusieurs sous-programmes optionnels, annotés dans le programme principal avec " +" suivi par "C" si ils sont nécessaires pour le programme de cycle, et par "P" s'ils sont nécessaires pour le programme de commande proportionnelle. Ces appels devront être remplacés par des instructions "NOP" (c.a.d. <del>00</del> seront entrés aux trois emplacements de chaque appel) quand ils ne sont pas nécessaires.
17		déclarations
19		sortie caractères à afficher
20		mise à jour données affichées
21		moniteur sous-programme retard
22		circuit tampon d'entrée chargé par le sous-programme entrée au clavier dans le moniteur
25		réglage bande proportionnelle
27		fixer limite haute
28	" "	basse
29		fixer valeur (réglage de température)
30		valeur fixée affichage circuit tampon
31		conserver caractère "S"
32		conserver "P"
33	"	premier chiffre de SP
34	" "	second " de SP
35		compteurs de commande proportionnelle
36		bits de commande du réchauffeur et du ventilateur
37		compteur du temps de cycle (réinitialisation à "bande" une fois par cycle
39		conserver compteur de temps de branchement
42		début adresse
46		charger indicateur de pile
47		initialiser bornes, interruption
48		initialiser variables
49		afficher valeur fixée
51		effacer drapeaux AC et C
53		ranger temp
54		afficher temp
55		remettre temp à A

décider arrêt/marche du ventilateur  
 " "  
 entrer valeur fixée P (optionnel, à utiliser uniquement dans le programme de  
 commande proportionnelle)  
 paramètre de retard (nominal)  
 peut être augmenté à volonté  
 répéter sans limite  
 fin du programme principal  
 laisser espace pour modifications  
 procédure initialisations bornes, interruptions  
 initialiser bornes  
 nécessaire pour pas simple  
 démasquer RST5.5 interruption (utilisé seulement pour le sous-programme optionnel  
 d'entrée de valeur fixée)  
 fin initialisation  
 espace pour modifications  
 procédure d'initialisation des variables  
 fixer HL comme indicateur  
 commencer à mettre en route le ventilateur; substituer 00 pour l'arrêter  
 mis en mémoire dans HTRFAN  
 HL indique CYTCI  
 le fixer pour démarrer le cycle prop  
 fin initialisation variables  
 procédure affichage valeur fixée  
 entrées - valeur fixée  
 sorties - affichage à 2 chiffres dans zone d'adresse  
 ranger registres  
 fixer indicateur  
 obtenir valeur fixée  
 reproduire valeur fixée dans B  
 démasquer le chiffre de poids + fort  
 mettre en mémoire le chiffre de poids le plus faible

- 103 indiquer MSDGT
- 104 réinitialiser valeur fixée
- 105 obtenir 4 bits de poids + fort dans positions poids + faible
- 109 effacer bits inutiles
- 110 chiffre poids + fort mis en mémoire
- 112 HL indique SVBUF
- 113 effacer A, utiliser zone adresse
- 114 effacer B (le point n'est pas nécessaire)
- 115 afficher contenu des mémoires tampons
- 116 réinitialiser registres
- 121 procédure lecture ADC (readADC)
- 123 ce sous-programme débute la conversion A-N et retourne les valeurs ADC quand la conversion est terminée.
- 126 entrées - bit de commande du réchauffeur venant de HTRFAN
- 127 sorties - valeurs ADC dans registre A
- 128 bornes E/S utilisées
- 130 ranger registres BC
- 131 obtenir bit de commande de réchauffeur
- 132 ranger une copie
- 133 débiter conversion A-N
- 134 obtenir bit de commande de réchauffeur
- 135 transmettre à borne de sortie
- 136 ligne de déclenchement à bas
- 139 " " haut
- 142 lire borne C
- 143 conversion terminée (bit 2=1)?
- 144 si non, inspecter de nouveau bit 2
- 145 autrement, obtenir valeur ADC en A
- 146 réinitialiser anciens registres BC
- 147 fin lecture ADC
- 150 procédure décision
- 151 entrées - limite H, limite B, temp dans registre A
- 152 sorties - mise en mémoire des bits de commande du réchauffeur, ventilateur
- 156 comparer HILIM & temp
- 157 si  $T > HILIM$ , alors refroidir
- 158 HL indique LOLIM
- 159 comparer LOLIM et temp
- 160 Si  $T < LOLIM$ , alors chauffer
- 161 si  $T = LOLIM$ , alors chauffer
- 162 réinitialiser ancienne valeur de HL
- 163 fin décision



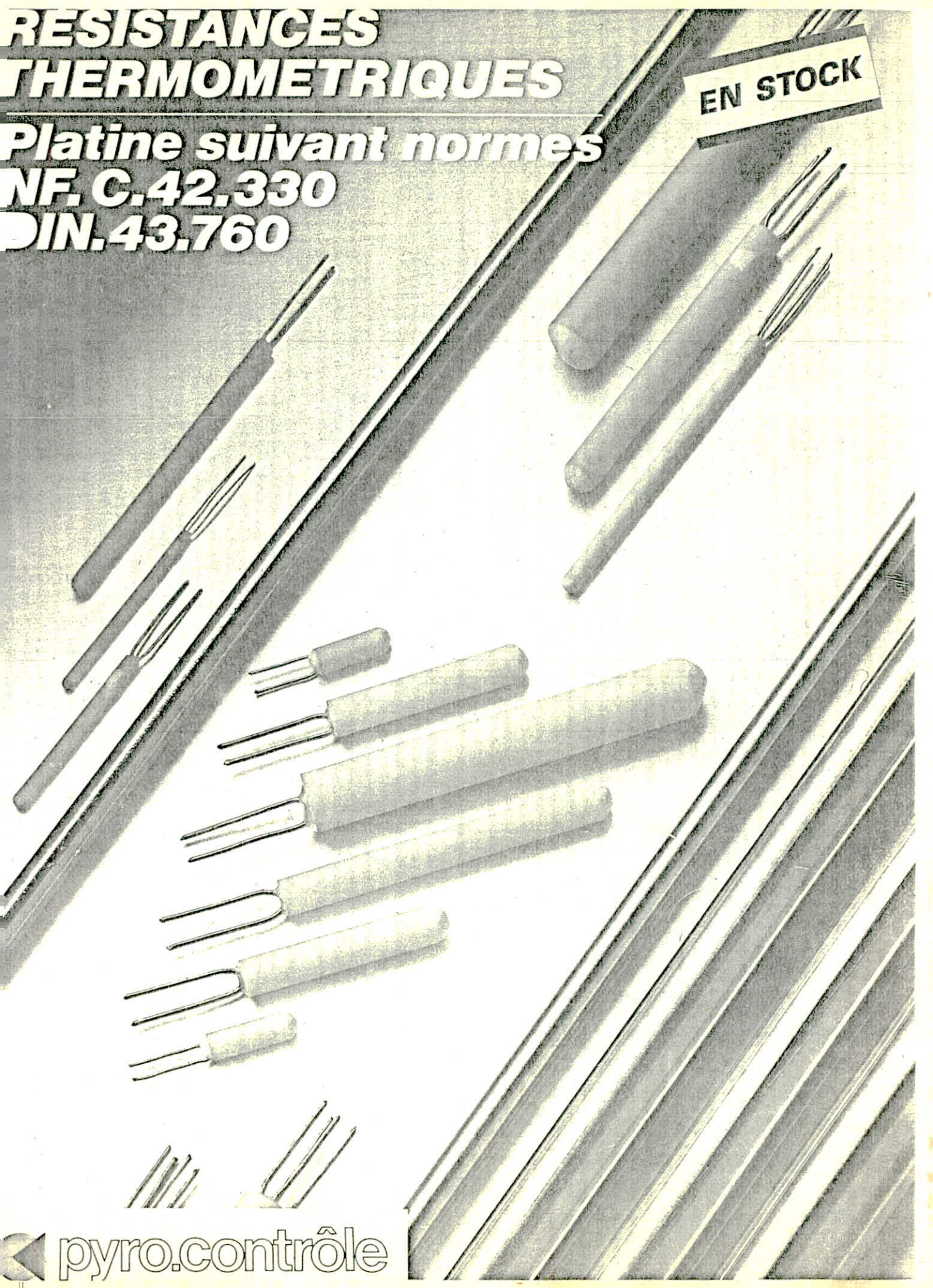
arrêt réchauffeur, marche ventilateur  
 arrêt ventilateur, marche réchauffeur  
 mise en mémoire bits de commande  
 procédure commande proportionnelle  
 entrées - temp (dans un registre) valeur fixée, compteur de temps de branchement  
 compteur de temps de cycle  
 sorties - réchauffeur sur bit = bit 5 d'un registre, compteur de temps branchement,  
 compteur de temps de cycle  
 ranger registres B et C  
 et registres H et L  
 ranger temp en C  
 HL indique temps de cycle au compteur  
 diminuer le compteur  
 si fin du décompte, réinitialiser compteurs  
 effacer registre A (RECHcoupé)  
 HL indique temps de branchement au compteur  
 le diminuer  
 si temps de branchement dépassé, couper réchauffeur  
 autrement fixer <sup>bit</sup> réchauffeur branché  
 mise en mémoire bit réchauffeur branché  
 réinitialiser HL rangé  
 réinitialiser BC rangé  
 fin arrêt/marche  
 réinitialiser compteur de temps de branchement et de cycle  
 entrer HL indiquant le compteur de cycles et temp dans C  
 obtenir valeur fixée dans registre A  
 saturer à 7E(hex)  
 calculer (SV-temp)  
 et la ranger dans C  
 bande proportionnelle  
 indique compteur temps de branchement  
 charger (SV-temp)  
 HL en entrée  
 retour au sous-programme prop  
 procédure entrée valeur fixée  
 Ceci est un sous-programme optionnel non inclus dans les exercices proposés à  
 l'étudiant. Il assure une meilleure démonstration et pourra servir de travail  
 complémentaire.  
 Les données entrées au clavier à l'aide du sous-programme ININT du moniteur  
 sont décalées à gauche dans la valeur fixée.

- 222 valider interruptions
- 223 obtenir mémoire tampon d'entrée
- 224 la tester
- 225 Si bit 7 est mis (tampon vide), retour
- 227 autrement, ranger B,C
- 228 ranger chiffre en C
- 229 obtenir ancienne valeur fixée
- 230 décaler moitié de poids le plus faible à la position de poids le plus fort
- 234 effacer bits inutiles
- 235 amener un nouveau chiffre de poids le plus faible
- 236 mise en mémoire de la nouvelle valeur fixée
- 238 mettre drapeau vide de Ibuff
- 240 fin entrée valeur fixée

# RESISTANCES THERMOMETRIQUES


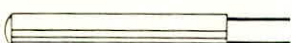





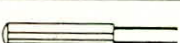



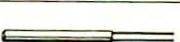



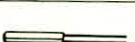
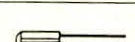
Platine suivant normes  
NF. C.42.330  
DIN.43.760

EN STOCK



# RÉSISTANCES THERMOMÉTRIQUES PLATINE

## 100 Ω À 0°C. suivant normes NF.C 42330 et DIN 43.760 (octobre 80)

	Constante de temps en s*	Corps céramique		Longueur sensible (mm)		N° identification
		Diamètre (mm)	Longueur (mm)			
	0,3	2,4 <sup>+0</sup> <sub>-0,03</sub>	50 <sup>+0</sup> <sub>-0,5</sub>	47 ± 1	1 × 100 2 × 100	310 023-110 310 024-110
	0,7	4,5 <sup>+0</sup> <sub>-0,03</sub>	30 <sup>+0</sup> <sub>-0,5</sub>	27 ± 1	1 × 100 2 × 100	310 008-110 • 310 009-110 •
	0,4	3,2 <sup>+0</sup> <sub>-0,03</sub>	25 <sup>+0</sup> <sub>-0,5</sub>	22 ± 1	1 × 100 2 × 100	310 025-110 310 026-110
	0,4	2,8 <sup>+0</sup> <sub>-0,03</sub>	25 <sup>+0</sup> <sub>-0,5</sub>	22 ± 1	1 × 100 2 × 100	310 001-110 • 310 002-110 •
	0,3	2,4 <sup>+0</sup> <sub>-0,03</sub>	25 <sup>+0</sup> <sub>-0,5</sub>	22 ± 1	1 × 100 2 × 100	310 027-110 310 028-110
	0,1	1,5 <sup>+0</sup> <sub>-0,03</sub>	25 <sup>+0</sup> <sub>-0,5</sub>	22 ± 1	1 × 100 2 × 100	310 003-110 • 310 004-110 •
	0,7	4,5 <sup>+0</sup> <sub>-0,03</sub>	15 <sup>+0</sup> <sub>-0,5</sub>	12 ± 1	1 × 100 2 × 100	310 029-110 310 030-110
	0,4	3,2 <sup>+0</sup> <sub>-0,03</sub>	15 <sup>+0</sup> <sub>-0,5</sub>	12 ± 1	1 × 100 2 × 100	310 031-110 310 032-110
	0,4	2,8 <sup>+0</sup> <sub>-0,03</sub>	15 <sup>+0</sup> <sub>-0,5</sub>	12 ± 1	1 × 100 2 × 100	310 011-110 310 040-110
	0,3	2,4 <sup>+0</sup> <sub>-0,03</sub>	15 <sup>+0</sup> <sub>-0,5</sub>	12 ± 1	1 × 100 2 × 100	310 033-110 310 034-110
	0,25	2,0 <sup>+0</sup> <sub>-0,03</sub>	15 <sup>+0</sup> <sub>-0,5</sub>	12 ± 1	1 × 100 2 × 100	310 035-110 310 036-110
	0,15	1,6 <sup>+0</sup> <sub>-0,03</sub>	15 <sup>+0</sup> <sub>-0,5</sub>	12 ± 1	1 × 100 2 × 100	310 037-110 310 038-110
	0,1	1,5 <sup>+0</sup> <sub>-0,03</sub>	15 <sup>+0</sup> <sub>-0,5</sub>	12 ± 1	1 × 100 2 × 100	310 012-110 • 310 042-110
	0,08	1,2 ± 5%	15 <sup>+0</sup> <sub>-0,5</sub>	12 ± 1	1 × 100	310 005-110 •
	0,03	0,9 ± 5%	15 <sup>+0</sup> <sub>-0,5</sub>	12 ± 1	1 × 100	310 006-110 •
	0,1	1,5 <sup>+0</sup> <sub>-0,03</sub>	8 <sup>+0</sup> <sub>-0,5</sub>	8 <sup>+0</sup> <sub>-1</sub>	1 × 100	310 041-110
	0,25	2,0 ± 3%	6 <sup>+0</sup> <sub>-0,5</sub>	6 <sup>+0</sup> <sub>-1</sub>	1 × 100	310 007-110 •

\* La constante de temps est le temps requis par l'élément pour atteindre 63 % de l'écart total de température. Ces valeurs sont mesurées dans une circulation d'eau en mouvement V = 1 m/s à 60°C.

Chaque élément est livré avec une sortie de fils platine de longueur 8 mm (le contrôle est réalisé à 5 mm du corps de sonde).

— Courant de mesure recommandé : pour éléments  $\varnothing$  4,5 - 3,2 - 2,8 et 2,4  $\leq$  5 mA  
 $\varnothing$  1,6 et 1,5  $\leq$  3 mA  
 $\varnothing$  1,2  $\leq$  2 mA  
 $\varnothing$  0,9  $\leq$  1 mA

— Stabilité meilleure que  $\pm 0,05$  % après 10 chocs thermiques consécutifs de 0°C à -220°C ou de 0°C à +600°C.

— Vibration : résiste à 30 G à une fréquence de 10 à 1.000 Hz.

— Auto-échauffement : moins de 0,3°C pour 10 mW dissipés dans un bain de glace fondante, acitée.

Températures °C	CLASSE A		CLASSE B	
	Tolérances en °C	Tolérances en Ω	Tolérances en °C	Tolérances en Ω
-200	0,55	0,24	1,3	0,56
-100	0,35	0,14	0,8	0,32
0	0,15	0,06	0,3	0,12
100	0,35	0,13	0,8	0,30
200	0,55	0,20	1,3	0,48
300	0,75	0,27	1,8	0,64
400	0,95	0,33	2,3	0,79
500	1,15	0,38	2,8	0,93
600	1,35	0,43	3,3	1,06
650	1,45	0,46	3,6	1,13
700	—	—	3,8	1,17
800	—	—	4,3	1,28
850	—	—	4,6	1,34

**TOLÉRANCES EN °C ET EN Ω  
LES CLASSES A ET B**

**RELATIONS TEMP./VALEUR OHMIQUE  
de -200 à 0°C**

°C	-0	-1	-2	-3	-4	-5	-6	-7	-8	-9
-200	18,49	—	—	—	—	—	—	—	—	—
-190	22,80	22,37	21,94	21,51	21,08	20,65	20,22	19,79	19,36	18,93
-180	27,08	26,65	26,23	25,80	25,37	24,94	24,52	24,09	23,66	23,23
-170	31,32	30,90	30,47	30,05	29,63	29,20	28,78	28,35	27,93	27,50
-160	35,53	35,11	34,69	34,27	33,85	33,43	33,01	32,59	32,16	31,74
-150	39,71	39,30	38,88	38,46	38,04	37,63	37,21	36,79	36,37	35,95
-140	43,87	43,45	43,04	42,63	42,21	41,79	41,38	40,96	40,55	40,13
-130	48,00	47,59	47,18	46,76	46,35	45,94	45,52	45,11	44,70	44,28
-120	52,11	51,70	51,29	50,88	50,47	50,06	49,64	49,23	48,82	48,41
-110	56,19	55,78	55,36	54,97	54,56	54,15	53,74	53,33	52,92	52,52
-100	60,25	59,85	59,44	59,04	58,63	58,22	57,82	57,41	57,00	56,60
-90	64,30	63,90	63,49	63,09	62,68	62,28	61,87	61,47	61,06	60,66
-80	68,33	67,92	67,52	67,12	66,72	66,31	65,91	65,51	65,11	64,71
-70	72,33	71,93	71,53	71,13	70,73	70,33	69,93	69,53	69,13	68,73
-60	76,33	75,93	75,53	75,13	74,73	74,33	73,93	73,53	73,13	72,73
-50	80,31	79,91	79,51	79,11	78,72	78,32	77,92	77,52	77,13	76,73
-40	84,27	83,88	83,48	83,08	82,69	82,29	81,89	81,50	81,10	80,70
-30	88,22	87,83	87,43	87,04	86,64	86,25	85,85	85,46	85,06	84,67
-20	92,16	91,77	91,37	90,98	90,59	90,19	89,80	89,40	89,01	88,62
-10	96,09	95,69	95,30	94,91	94,52	94,12	93,73	93,34	92,95	92,55
0	100,00	99,61	99,22	98,83	98,44	98,04	97,65	97,26	96,87	96,48

**de 0° à 850°C**

°C	0	1	2	3	4	5	6	7	8	9
0	100,00	100,39	100,78	101,17	101,56	101,95	102,34	102,73	103,12	103,51
10	103,90	104,29	104,68	105,07	105,46	105,85	106,24	106,63	107,02	107,40
20	107,79	108,18	108,57	108,96	109,35	109,73	110,12	110,51	110,90	111,28
30	111,67	112,06	112,45	112,83	113,22	113,61	113,99	114,38	114,77	115,15
40	115,54	115,93	116,31	116,70	117,08	117,47	117,85	118,24	118,62	119,01
50	119,40	119,78	120,16	120,55	120,93	121,32	121,70	122,09	122,47	122,86
60	123,24	123,62	124,01	124,39	124,77	125,16	125,54	125,92	126,31	126,69
70	127,07	127,45	127,84	128,22	128,60	128,98	129,37	129,75	130,13	130,51
80	130,89	131,27	131,66	132,04	132,42	132,80	133,18	133,56	133,94	134,32
90	134,70	135,08	135,46	135,84	136,22	136,60	136,98	137,36	137,74	138,12
100	138,50	138,88	139,26	139,64	140,02	140,39	140,77	141,15	141,53	141,91
110	142,29	142,66	143,04	143,42	143,80	144,17	144,55	144,93	145,31	145,68
120	146,06	146,44	146,81	147,19	147,57	147,94	148,32	148,70	149,07	149,45
130	149,82	150,20	150,57	150,95	151,33	151,70	152,08	152,45	152,83	153,20
140	153,58	153,95	154,32	154,70	155,07	155,45	155,82	156,19	156,57	156,94
150	157,31	157,69	158,06	158,43	158,81	159,18	159,55	159,93	160,30	160,67
160	161,04	161,42	161,79	162,16	162,53	162,90	163,27	163,65	164,02	164,39
170	164,76	165,13	165,50	165,87	166,24	166,61	166,98	167,35	167,72	168,09
180	168,46	168,83	169,20	169,57	169,94	170,31	170,68	171,05	171,42	171,79
190	172,16	172,53	172,90	173,26	173,63	174,00	174,37	174,74	175,10	175,47

°C	0	1	2	3	4	5	6	7	8	9
200	175,84	176,21	176,57	176,94	177,31	177,68	178,04	178,41	178,78	179,14
210	179,51	179,88	180,24	180,61	180,97	181,34	181,71	182,07	182,44	182,80
220	183,17	183,53	183,90	184,26	184,63	184,99	185,36	185,72	186,09	186,45
230	186,82	187,18	187,54	187,91	188,27	188,63	189,00	189,36	189,72	190,09
240	190,45	190,81	191,18	191,54	191,90	192,26	192,63	192,99	193,35	193,71
250	194,07	194,44	194,80	195,16	195,52	195,88	196,24	196,60	196,96	197,33
260	197,69	198,05	198,41	198,77	199,13	199,49	199,85	200,21	200,57	200,93
270	201,29	201,65	202,01	202,36	202,72	203,08	203,44	203,80	204,16	204,52
280	204,88	205,23	205,59	205,95	206,31	206,67	207,02	207,38	207,74	208,10
290	208,45	208,81	209,17	209,52	209,88	210,24	210,59	210,95	211,31	211,66
300	212,02	212,37	212,73	213,09	213,44	213,80	214,15	214,51	214,86	215,22
310	215,57	215,93	216,28	216,64	216,99	217,35	217,70	218,05	218,41	218,76
320	219,12	219,47	219,82	220,18	220,53	220,88	221,24	221,59	221,94	222,29
330	222,65	223,00	223,35	223,70	224,06	224,41	224,76	225,11	225,46	225,81
340	226,17	226,52	226,87	227,22	227,57	227,92	228,27	228,62	228,97	229,32
350	229,67	230,02	230,37	230,72	231,07	231,42	231,77	232,12	232,47	232,82
360	233,17	233,52	233,87	234,22	234,56	234,91	235,26	235,61	235,96	236,31
370	236,65	237,00	237,35	237,70	238,04	238,39	238,74	239,09	239,43	239,78
380	240,13	240,47	240,82	241,17	241,51	241,86	242,20	242,55	242,90	243,24
390	243,59	243,93	244,28	244,62	244,97	245,31	245,66	246,00	246,35	246,69
400	247,04	247,38	247,73	248,07	248,41	248,76	249,10	249,45	249,79	250,13
410	250,48	250,82	251,16	251,50	251,85	252,19	252,53	252,88	253,22	253,56
420	253,90	254,24	254,59	254,93	255,27	255,61	255,95	256,29	256,64	256,98
430	257,32	257,66	258,00	258,34	258,68	259,02	259,36	259,70	260,04	260,38
440	260,72	261,06	261,40	261,74	262,08	262,42	262,76	263,10	263,43	263,77
450	264,11	264,45	264,79	265,13	265,47	265,80	266,14	266,48	266,82	267,15
460	267,49	267,83	268,17	268,50	268,84	269,18	269,51	269,85	270,19	270,52
470	270,86	271,20	271,53	271,87	272,20	272,54	272,88	273,21	273,55	273,88
480	274,22	274,55	274,89	275,22	275,56	275,89	276,23	276,56	276,89	277,23
490	277,56	277,90	278,23	278,56	278,90	279,23	279,56	279,90	280,23	280,56
500	280,90	281,23	281,56	281,89	282,23	282,56	282,89	283,22	283,55	283,89
510	284,22	284,55	284,88	285,21	285,54	285,87	286,21	286,54	286,87	287,20
520	287,53	287,86	288,19	288,52	288,85	289,18	289,51	289,84	290,17	290,50
530	290,83	291,16	291,49	291,81	292,14	292,47	292,80	293,13	293,46	293,79
540	294,11	294,44	294,77	295,10	295,43	295,75	296,08	296,41	296,74	297,06
550	297,39	297,72	298,04	298,37	298,70	299,02	299,35	299,68	300,00	300,33
560	300,65	300,98	301,31	301,63	301,96	302,28	302,61	302,93	303,26	303,58
570	303,91	304,23	304,56	304,88	305,20	305,53	305,85	306,18	306,50	306,82
580	307,15	307,47	307,79	308,12	308,44	308,76	309,09	309,41	309,73	310,05
590	310,38	310,70	311,02	311,34	311,67	311,99	312,31	312,63	312,95	313,27
600	313,59	313,92	314,24	314,56	314,88	315,20	315,52	315,84	316,16	316,48
610	316,80	317,12	317,44	317,76	318,08	318,40	318,72	319,04	319,36	319,68
620	319,99	320,31	320,63	320,95	321,27	321,59	321,91	322,22	322,54	322,86
630	323,18	323,49	323,81	324,13	324,45	324,76	325,08	325,40	325,72	326,04
640	326,35	326,66	326,98	327,30	327,61	327,93	328,25	328,56	328,88	329,19
650	329,51	329,82	330,14	330,45	330,77	331,08	331,40	331,71	332,03	332,34
660	332,66	332,97	333,28	333,60	333,91	334,23	334,54	334,85	335,17	335,48
670	335,79	336,11	336,42	336,73	337,04	337,36	337,67	337,98	338,29	338,61
680	338,92	339,23	339,54	339,85	340,16	340,48	340,79	341,10	341,41	341,72
690	342,03	342,34	342,65	342,96	343,27	343,58	343,89	344,20	344,51	344,82
700	345,13	345,44	345,75	346,06	346,37	346,68	346,99	347,30	347,60	347,91
710	348,22	348,53	348,84	349,15	349,45	349,76	350,07	350,38	350,69	350,99
720	351,30	351,61	351,91	352,22	352,53	352,83	353,14	353,45	353,75	354,06
730	354,37	354,67	354,98	355,28	355,59	355,90	356,20	356,51	356,81	357,12
740	357,42	357,73	358,03	358,34	358,64	358,95	359,25	359,55	359,86	360,16
750	360,47	360,77	361,07	361,38	361,68	361,98	362,29	362,59	362,89	363,19
760	363,50	363,80	364,10	364,40	364,71	365,01	365,31	365,61	365,91	366,22
770	366,52	366,82	367,12	367,42	367,72	368,02	368,32	368,63	368,93	369,23
780	369,53	369,83	370,13	370,43	370,73	371,03	371,33	371,63	371,93	372,22
790	372,52	372,82	373,12	373,42	373,72	374,02	374,32	374,61	374,91	375,21
800	375,21	375,81	376,10	376,40	376,70	377,00	377,29	377,59	377,89	378,19
810	378,48	378,78	379,08	379,37	379,67	379,97	380,26	380,56	380,85	381,15
820	381,45	381,74	382,04	382,33	382,63	382,92	383,22	383,51	383,81	384,10
830	384,40	384,69	384,98	385,28	385,57	385,87	386,16	386,45	386,75	387,04
840	387,34	387,63	387,92	388,21	388,51	388,80	389,09	389,39	389,68	389,97
850	390,26	—	—	—	—	—	—	—	—	—

Parmi nos fabrications : Sondes étalon fournies avec table - Etalonnages - Appariages de capteurs.

Egalement sur stock : Capteurs Pt 100 - Doigts de gant - Capteurs à poignée - Thermocouples gainés - Connecteurs compensés - Câbles de compensation.

