

NOMENCLATURE

CAN : (convertisseur analogique numérique)

UC : unité de calcul

ON/OFF : marche/arrêt

CI : circuit intégré

A.O : amplificateur opérationnel

LED : diode électroluminescente

P N : positif négatif

TTL : Transistors Transistors logic

IEEE: Electrical and Electronics Engineers

SPP: Standard Parallèle Port

EPP : port parallèle étendu

ECP : port à capacités étendues

PC : Professional Computer

LTP : Local Printer

D_i : bits des registres de données

E/S : Entrée/Sortie

ACK : Acknowledge

DMA: accès mémoire direct

OOAD: Object Oriented Analysis and Designed

UML : Unifeid Modeling Language

I.G.U : Interface Graphique Utilisateur

M.V.C : Modele Vue Controleur

E.D.I : Environnement de Développement Intégré

LISTE DES FIGURES

Figure 1.	<i>Chauffage d'un moule de brique.....</i>	<i>4</i>
Figure 2.	<i>Cycle de l'air dans l'étuve.....</i>	<i>5</i>
Figure 3.	<i>Photo représentant l'étuve.....</i>	<i>6</i>
Figure 4.	<i>Photo représentant la boîte commande.....</i>	<i>6</i>
Figure 5.	<i>Système de la boîte de commande.....</i>	<i>7</i>
Figure 6.	<i>Thermocouple... ..</i>	<i>8</i>
Figure 7.	<i>Schéma bloc simplifié.....</i>	<i>10</i>
Figure 8.	<i>Schéma bloc plus détaillé.....</i>	<i>11</i>
Figure 9.	<i>Schéma bloc d'acquisition de données.....</i>	<i>12</i>
Figure 10.	<i>Schéma bloc équivalent</i>	<i>13</i>
Figure 11.	<i>Schéma bloc du circuit de mise en forme.....</i>	<i>13</i>
Figure 12a.	<i>Schéma du circuit électronique de mise en forme.....</i>	<i>14</i>
Figure 12b.	<i>Schéma équivalent du circuit électronique de mise en forme.....</i>	<i>14</i>
Figure 13.	<i>Brochage du LM324.....</i>	<i>15</i>
Figure 14.	<i>Schéma bloc de circuit de conversion et de stockage.....</i>	<i>15</i>
Figure 15.	<i>Schéma du circuit électronique de conversion et de stockage.....</i>	<i>16</i>
Figure 16.	<i>Symbole de l'ADC avec ses composants de sécurité.....</i>	<i>17</i>
Figure 17.	<i>Symbole du registre.....</i>	<i>17</i>
Figure 18.	<i>Schéma du circuit électronique d'acquisition.....</i>	<i>18</i>
Figure 19.	<i>Schéma bloc pour la commande du ventilateur et de la résistance.....</i>	<i>19</i>
Figure 20.	<i>Commande des deux relais.....</i>	<i>19</i>
Figure 21.	<i>Schéma détaillé de la commande.....</i>	<i>20</i>
Figure 22.	<i>Symbole de la diode.....</i>	<i>21</i>
Figure 23.	<i>Circuit de commande et Circuit commuté.....</i>	<i>21</i>

Figure 24.	<i>Transistors NPN.....</i>	<i>21</i>
Figure 25.	<i>Symbole de l'optocoupleur.....</i>	<i>22</i>
Figure 26.	<i>Schéma de puissance.....</i>	<i>23</i>
Figure 27.	<i>Représentation du port parallèle.....</i>	<i>25</i>
Figure 28.	<i>Connecteur SUB-D 25male.....</i>	<i>25</i>
Figure 29.	<i>Connecteur Centronics male.....</i>	<i>25</i>
Figure 30.	<i>Principe de port parallèle.....</i>	<i>30</i>
Figure 31.	<i>Schéma bloc du kit de pilotage.....</i>	<i>31</i>
Figure 32.	<i>Schéma du cas d'utilisation.....</i>	<i>32</i>
Figure 33.	<i>Interface Graphique Utilisateur.....</i>	<i>33</i>
Figure 34.	<i>Strucure M.V.C.....</i>	<i>35</i>
Figure 35.	<i>Modèle / Moteur.....</i>	<i>35</i>
Figure 36.	<i>Diagramme des classes.....</i>	<i>37</i>
Figure 37.	<i>Diagramme définitif des classes.....</i>	<i>39</i>
Figure 38.	<i>Résultats obtenus après essais et simulation.....</i>	<i>46</i>
Figure 39.	<i>Exemple de comportement dynamique de l'étuve.....</i>	<i>48</i>

LISTE DES TABLEAUX

Tableau 1. Les brochages de connecteurs.....26
Tableau 2. Les bits des registres de commande.....27
Tableau 3. Les adresses des registres.....28
Tableau 4. Correspondance entre les objets du : I.G.U ET M.V.C.....35
Tableau 5. Les Classes.....36

Table des matières

INTRODUCTION GENERALE.....	3
CHAPITRE I : DESCRIPTION, FONCTIONNEMENT DE L'ETUVE A VONTOVORONA	4
1.1 DESCRIPTION DE L'ETUVE ET CYCLE DE L'AIR DANS L'ETUVE	5
1.2 REPRESENTATION DE L'ETUVE	6
1.3 LA BOITE DE COMMANDE MANUELLE DE L'ETUVE.....	6
CHAPITRE II. SYSTEME THERMOCOUPLE	8
2.1 GENERALITE SUR LE THERMOCOUPLE	8
2.2 PRINCIPE DE FONCTIONNEMENT	8
2.3 CHOIX DU THERMOCOUPLE	8
2.4 AVANTAGE DES THERMOCOUPLES	9
CHAPITRE III. CONCEPTION DU DISPOSTIF ELECTRONIQUE POUR LE PILOTAGE DE L'ETUVE.....	10
3.1 SCHEMA BLOC	10
3.2 Schéma bloc équivalent.....	11
3.3 CIRCUIT D'ACQUISITION DE DONNEES.....	12
3.4 CIRCUIT DE MISE EN FORME	13
3.5 CIRCUIT DE CONVERSION ET DE STOCKAGE.....	15
3.6 SCHEMA DU CIRUIT ELECTRONIQUE D'ACQUISITION	18
3.7 CIRCUIT DE COMMANDE DU VENTILATEUR ET DE LA RESISTANCE	19
3.8 ROLES DE CES COMPOSANTS DANS LA COMMANDE.....	21
3.9 CIRCUIT DE PUISSANCE DU VENTULATEUR ET DE LA RESISTANCE DE L'ETUVE	23
CHAPITRE IV : PORT PARALLELE DE L'ORDINATEUR.....	24
4.1. Introduction.....	24
4.2. REPRESENTATION DU PORT PARALLELE.....	25
4.3. LES COMPOSANTS DE BASE DU PORT PARALLELE	25
4.4. DESCRIPTION DES SIGNAUX	28
CHAPITRE V. ETUDE ET CONCEPTION DU LOGICIEL DE PILOTAGE DE L'ETUVE	31
5.1. METHODOLOGIE.....	31
5.2. LES CAS D'UTILISATIONS (USE CASES)	31
5.3. CAS D'UTILISATION	32
5.4. INTERFACE GRAPHIQUE UTILISATEUR « I.G.U ».....	33
5.5. STRUCTURE OBJET DU LOGICIEL	34
5.6. CORRESPONDANCE ENTRE LES OBJETS DU : I.G.U ET M.V.C.....	35

Promotion 2009

5.7.	MODELE / MOTEUR.....	35
5.8.	LES CLASSES.....	36
5.9.	DIAGRAMME DES CLASSES.....	37
5.10.	PROGRAMMATION.....	38
5.10.1.	CHOIX DU LANGAGE ET DE L'OUTIL DE DEVELOPPEMENT	38
5.10.2.	IMPLEMENTATION DES CLASSE.....	39
5.10.3.	REMARQUE.....	39
5.10.4.	Les fichiers headers (.h) des classes	40
CHAPITRE VI .MODE D'UTILISATION DU KIT ET SIMULATION		45
6.1.	CONFIGURATIONS EXIGEES	45
6.1.1.	configuration minimum de l'ordinateur.....	45
6.1.2.	Système de sécurité	45
6.1.3.	Système de démarrages	45
6.2.	SIMULATION.....	47
CONCLUSION GENERALE		48

INTRODUCTION GENERALE

Dans le Laboratoire de Thermique du Département Génie Mécanique et Productique (GMP) de la Filière Génie Industriel (G.I) de l'Ecole Supérieur Polytechnique d'Antananarivo (ESPA), nous disposons d'une étuve dont le fabricant et la marque sont inconnus. Pire encore, on ne dispose d'aucun manuel d'utilisation sur cet appareil. Comme conséquence, cet appareil a été manipulé manuellement depuis des années. Nous avons pensé qu'on pourrait piloter cette étuve à l'aide d'un micro-ordinateur.

Tout ceci nous a conduit à entreprendre ce mémoire qui s'intitule : **« CONCEPTION ET REALISATION D'UN SYSTEME DE PILOTAGE D'UNE ETUVE ASSISTE PAR ORDINATEUR »**.

Le présent mémoire comporte six chapitres :

- Dans le premier chapitre, on va parler de la description et du fonctionnement de l'étuve à l'ESPA;
- Dans le second, on parlera du système thermocouple ;
- Dans le troisième chapitre, on abordera la conception du dispositif électronique pour le pilotage de l'étuve ;
- Le quatrième chapitre parle du port parallèle de l'ordinateur ;
- Dans le cinquième chapitre nous allons faire l'étude et la conception du logiciel de pilotage de l'étuve.
- Et enfin dans le sixième chapitre on va parler du mode d'utilisation du kit et on terminera par une petite simulation.

CHAPITRE I : DESCRIPTION, FONCTIONNEMENT DE L'ETUVE A VONTOVORONA

L'étuve est une machine thermique qui permet de chauffer ou de sécher des éléments organiques, minéraux, etc...En réalité elle permet de chauffer les corps sous une température voulue (température de consigne) par exemple les moules, les briques, etc...



Figure 1. Chauffage d'un moule de brique

1.1 DESCRIPTION DE L'ETUVE ET CYCLE DE L'AIR DANS L'ETUVE

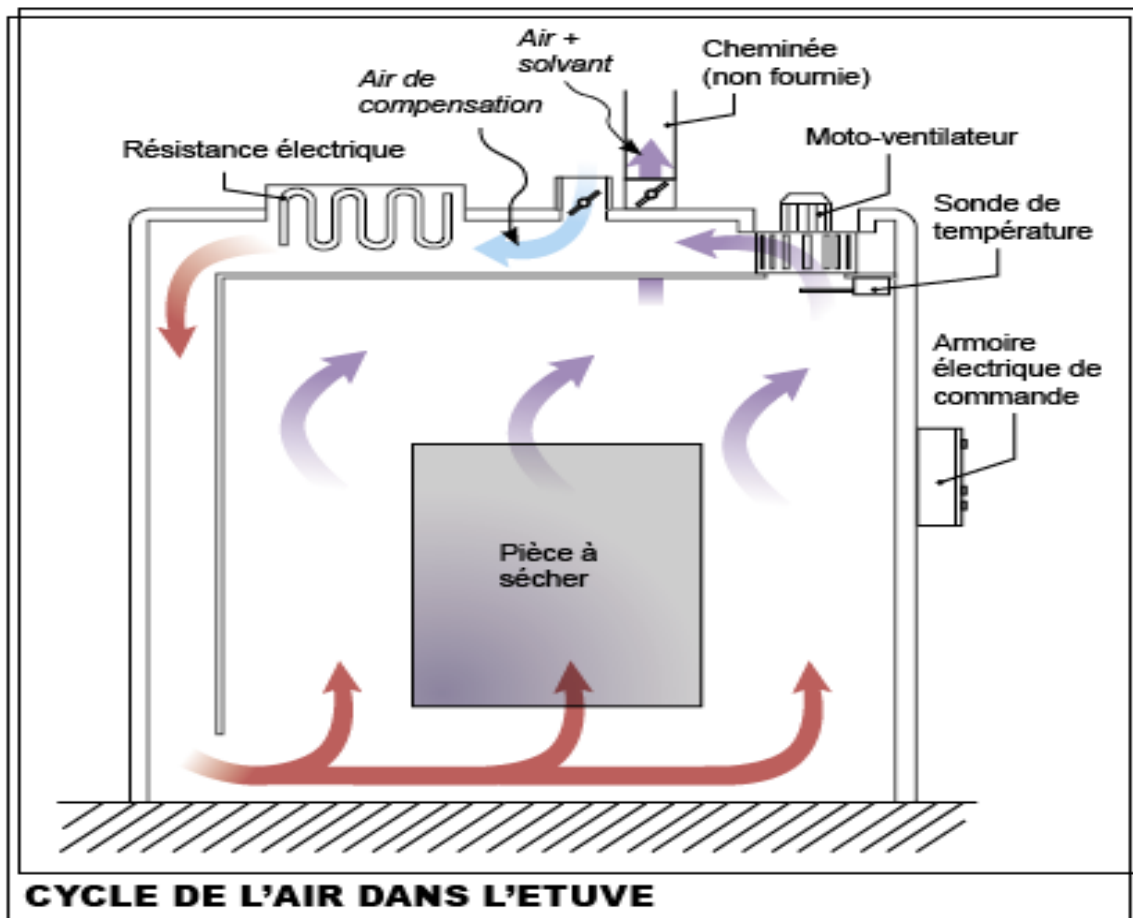


Figure 2. Cycle de l'air dans l'étuve

Le corps de l'étuve est constitué de :

- Une armoire électrique de commande,
- Une sonde de température ou thermocouple,
- Un moto - ventilateur de brassage d'air permettant un renouvellement d'air important pour la maîtrise de la température,
- Une cheminée pour le refoulement de l'air,
- Une résistance électrique pour le chauffage de l'air.

➤ **Explication du cycle :**

L'étuve électrique fonctionne comme un four à chaleur tournante : l'air circule au sein de l'étuve en circuit fermé.

- L'air est d'abord mis en circulation par le moto-ventilateur,
- Chauffé par la résistance électrique, l'air circule dans l'enceinte du bas vers le haut,
- L'air qui a refroidi est aspiré par le moto-ventilateur, et renvoyé vers la résistance électrique, maintenant un flux d'air chaud constant.

- Une partie réglable de l'air chaud s'évacue naturellement par la cheminée de sortie reliée à l'extérieur .
- Le volet de réglage doit être ouvert ,
- Une sonde de température contrôle constamment la température de l'enceinte.

1.2 REPRESENTATION DE L'ETUVE



Figure 3.Photo représentant l'étuve

1.3 LA BOITE DE COMMANDE MANUELLE DE L'ETUVE

L'armoire de commande, fixée sur l'étuve comprend :

- Tous les composants utiles à la commande du moteur et du chauffage,
- Un régulateur programmeur relié à un thermocouple ;
- Un thermostat de sécurité ;
- Une horloge de démarrage différé pour la réalisation des cycles la nuit.

a) Représentation de la boîte de commande



Figure 4.Photo représentant la boîte commande

b) Représentation du système de la boîte de commande

L'étuve électrique est gérée par l'armoire électrique.

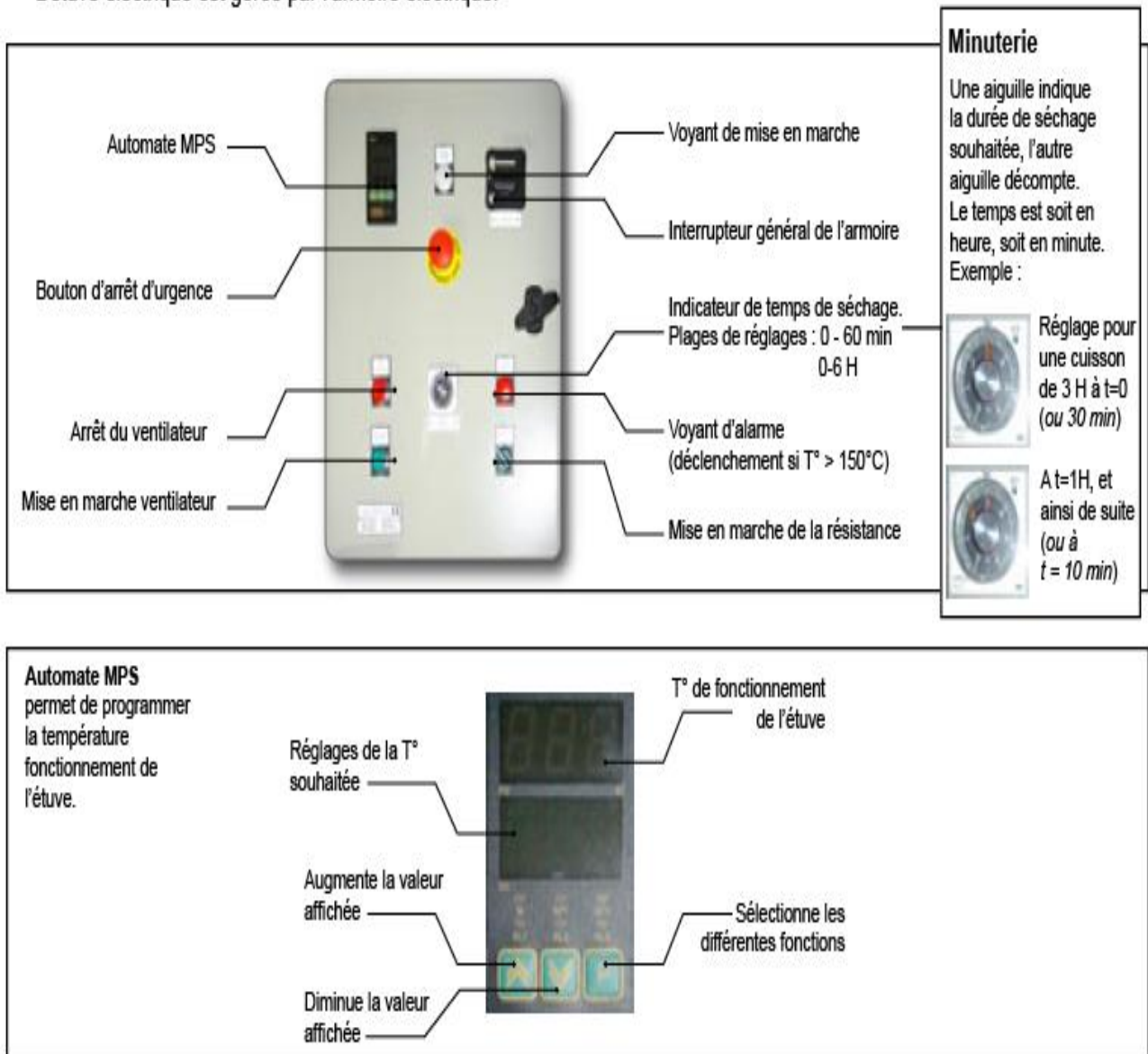


Figure 5. Système de la boîte de commande

CHAPITRE II. SYSTEME THERMOCOUPLE

2.1 GENERALITE SUR LE THERMOCOUPLE

a) But

Les thermocouples ou couples thermoélectriques transforment une température en une tension électrique mesurable. Le couple est constitué par deux conducteurs A et B de natures différentes soudés bout à bout en circuit fermé.

b) Objectif

Le thermocouple permet de capter la température dans l'étuve. C'est un élément constitutif d'un capteur.

2.2 PRINCIPE DE FONCTIONNEMENT

- Principe de mesure

Pour mesurer la tension d'un thermocouple, il suffit de le connecter à un multimètre ou à un autre système de mesure.

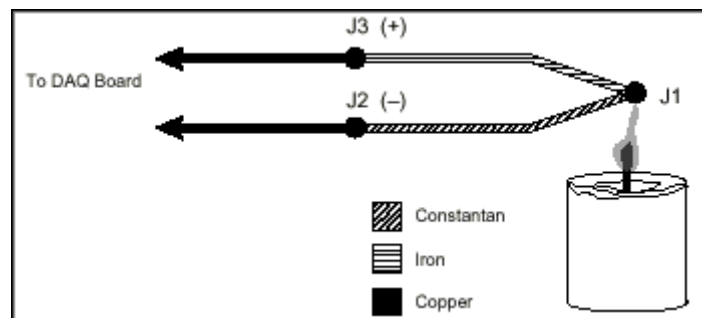


Figure 6. Thermocouple

- Observation

La figure 6 illustre le thermocouple qui est exposé à la flamme d'une bougie dont nous voulons mesurer la température. L'approche la plus pratique est de mesurer la température au niveau de la jonction par un capteur de température à lecture directe.

2.3 CHOIX DU THERMOCOUPLE

➤ La plage à mesurer

La première chose importante à prendre en compte est de faire coïncider la plage de température à mesurer avec la plage d'utilisation optimum du thermocouple.

➤ *Le milieu de mesure*

Certaines utilisations, surtout industrielles, des thermocouples conduiront à utiliser ces derniers avec des conditions de pression ou de milieu spécifiques. On retrouvera des mesures dans des milieux extrêmement basiques, acides, avec des pressions pouvant être importantes, cela aura pour conséquence de modifier la précision et la vitesse d'usure du thermocouple.

2.4 AVANTAGE DES THERMOCOUPLES

- ✓ Températures extrêmement élevées: Les métaux nobles peuvent convenir jusqu'à 1700°C (3100°F).
- ✓ Robustesse : Par leur simplicité, ils résistent aux chocs et aux vibrations.
- ✓ Faible encombrement, réponse rapide : Les thermocouples à jonctions exposée ou à la masse procurent une réponse pratiquement immédiate aux changements de température.

CHAPITRE III. CONCEPTION DU DISPOSITIF ELECTRONIQUE POUR LE PILOTAGE DE L'ETUVE

3.1 SCHEMA BLOC

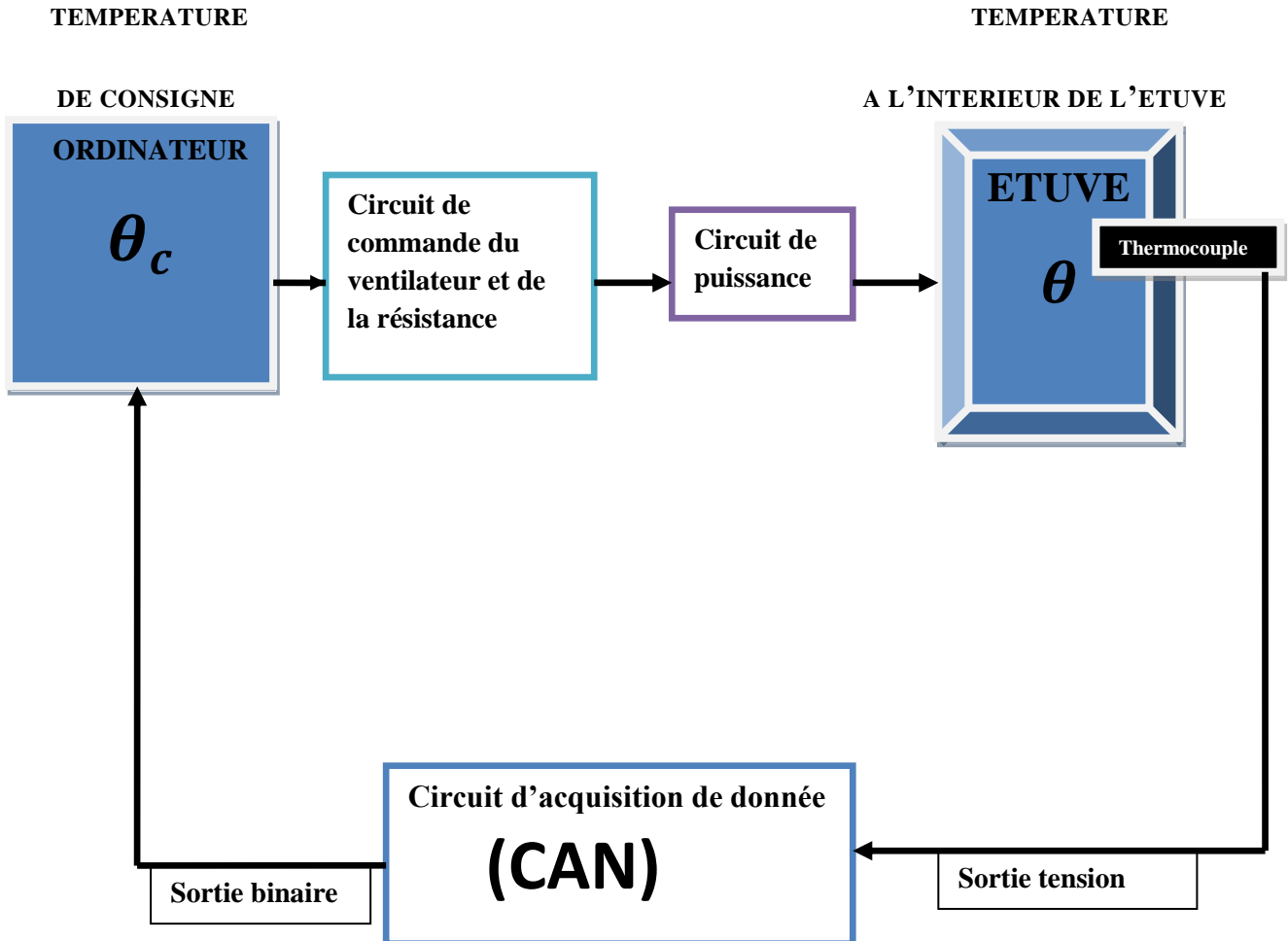


Figure 7.Schéma bloc simplifié

Le signal fourni par le thermocouple est une tension électrique qui est convertie en nombre binaire par le circuit intégré (C.I) appelé Convertisseur Analogique Numérique (CAN). A la sortie du CAN ce nombre binaire est lu par le logiciel à travers le port parallèle.

Soit θ la valeur de température dans l'étuve,

Soit θ_c la température de consigne que l'on voudrait avoir dans l'étuve.

Si $\theta < \theta_c$, alors le logiciel commande la fermeture des circuits de la moto ventilateur et de la résistance;

Si $\theta > \theta_c$, alors le logiciel commande l'ouverture des circuits de la moto ventilateur et de la résistance.

3.2 Schéma bloc équivalent

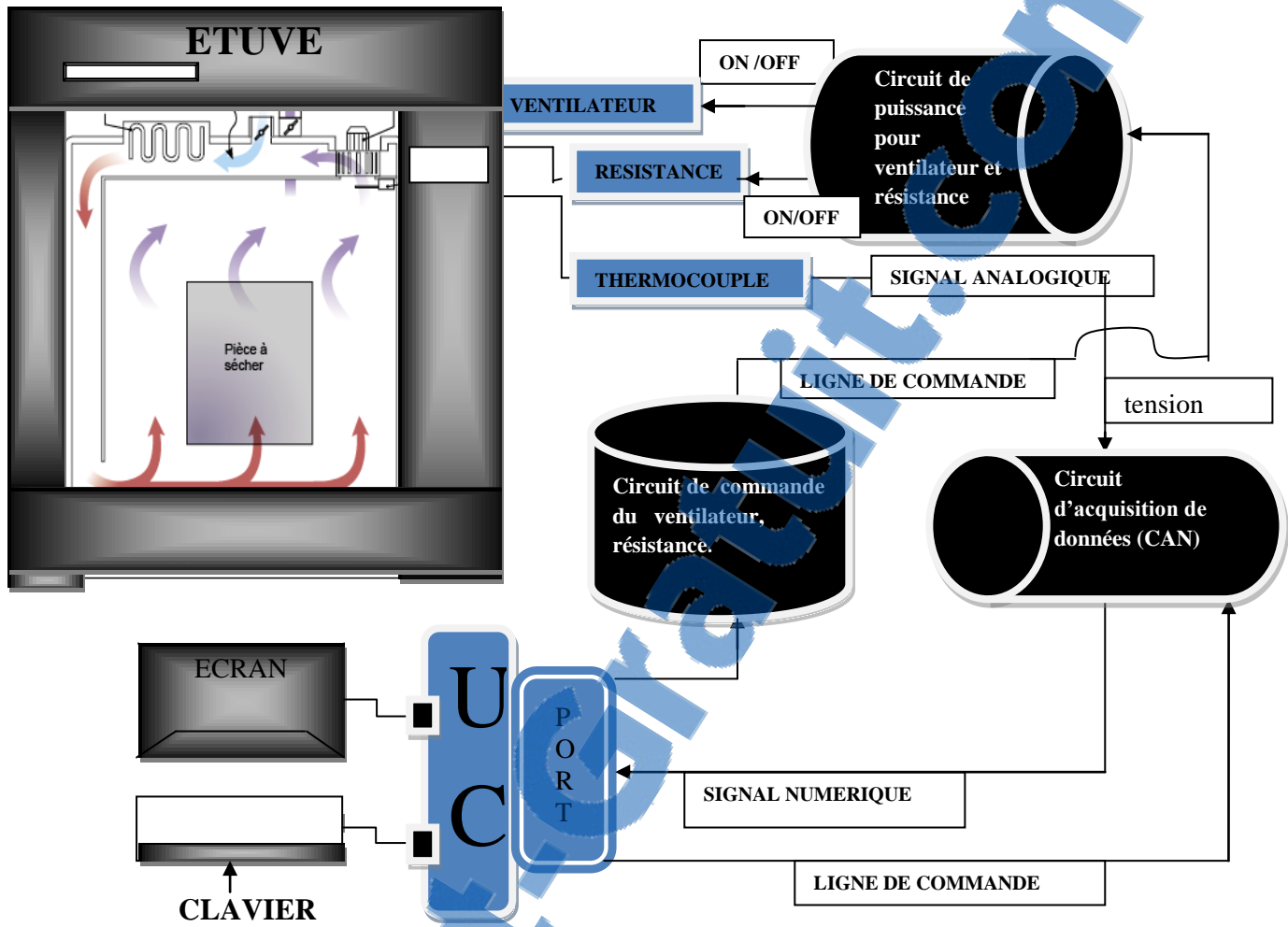


Figure 8. Schéma bloc plus détaillé

3.3 CIRCUIT D'ACQUISITION DE DONNEES

Le circuit d'acquisition de donnée se présente en deux étapes :

- **Circuit de mise en forme**
- **Circuit de conversion et de stockage.**

➤ *Expérience préalable :*

Mesure de la plage de variation de tension fournie par le thermocouple.

➤ **Résultats obtenus :**

Tension minimale $V_{\min}=0V$

Tension maximale $V_{\max} =10mV$

Le CI de conversion ADC exige une tension comprise entre 0V et 5V .Par conséquent la tension fournie par le thermocouple doit au préalable être mise en forme avant d'être convertie en binaire, d'où le schéma suivant :

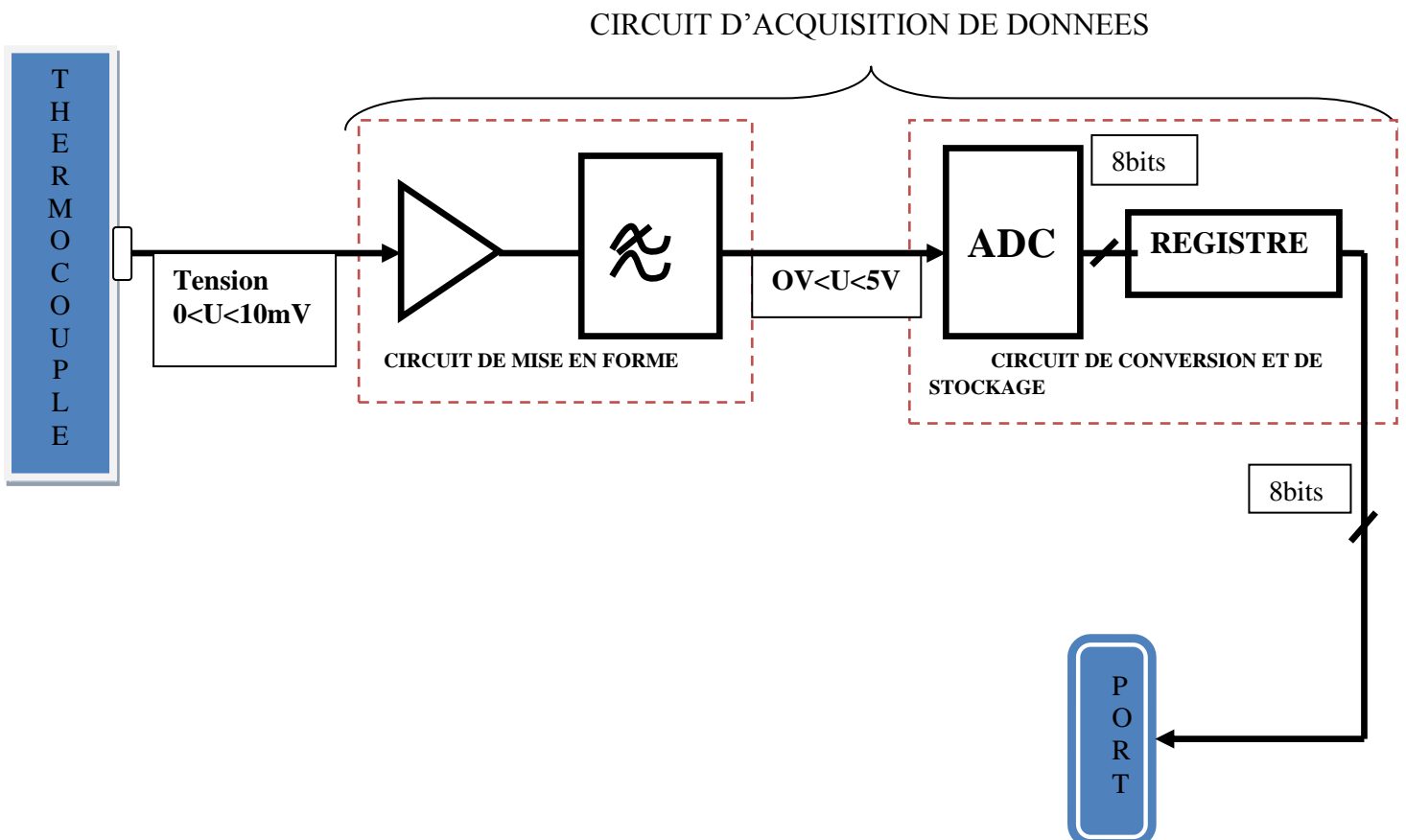


Figure 9. Schéma bloc d'acquisition de données

SCHEMA BLOC DU CIRCUIT ELECTRONIQUE D'ACQUISITION RELIE AVEC LE PORT

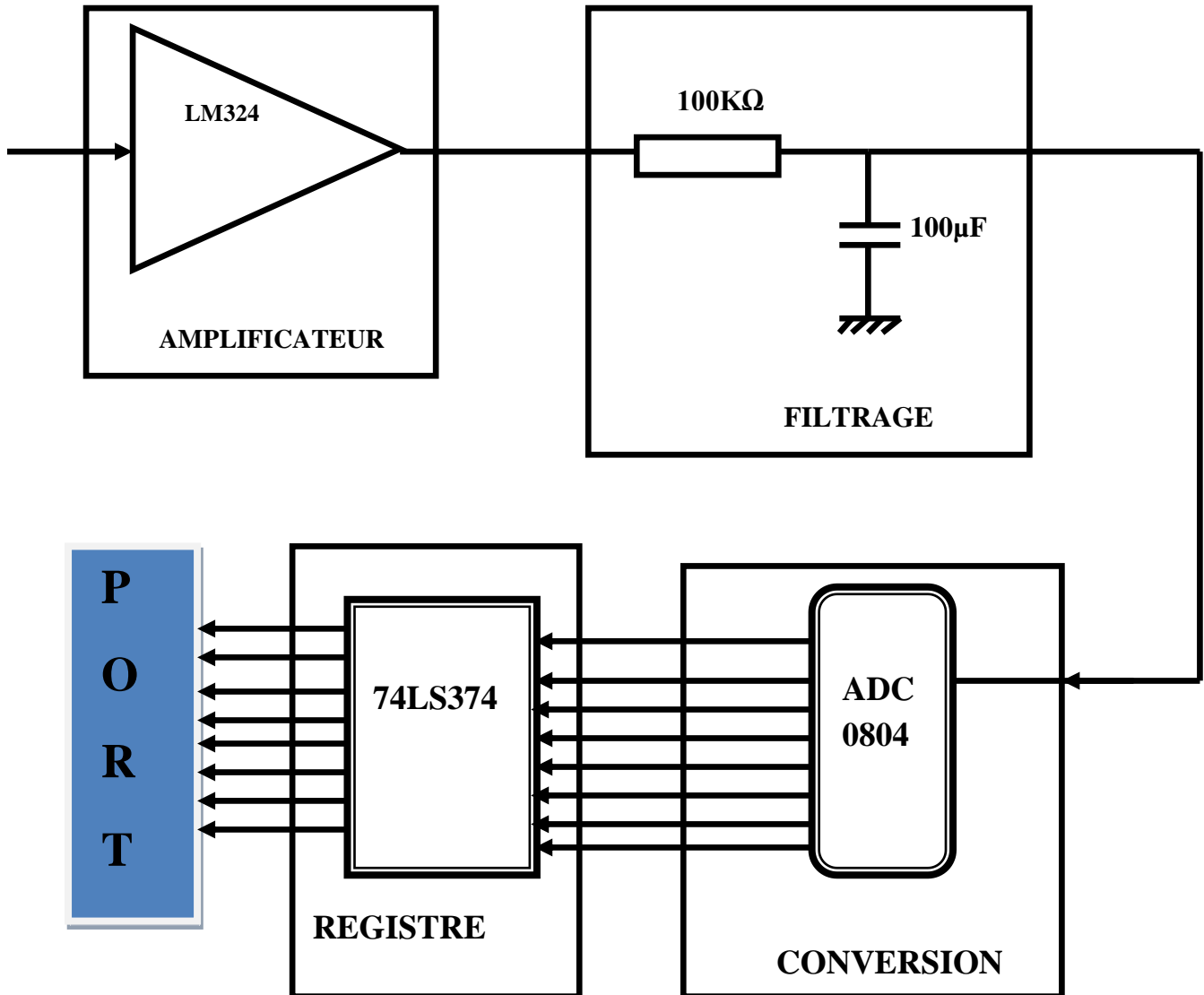


Figure 10. Schéma bloc équivalent

3.4 CIRCUIT DE MISE EN FORME

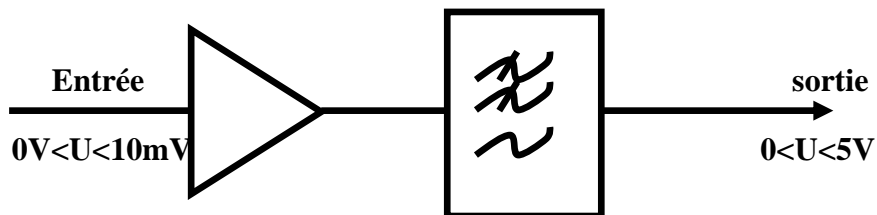


Figure11. Schéma bloc du circuit de mise en forme

Schéma du circuit électronique de mise en forme

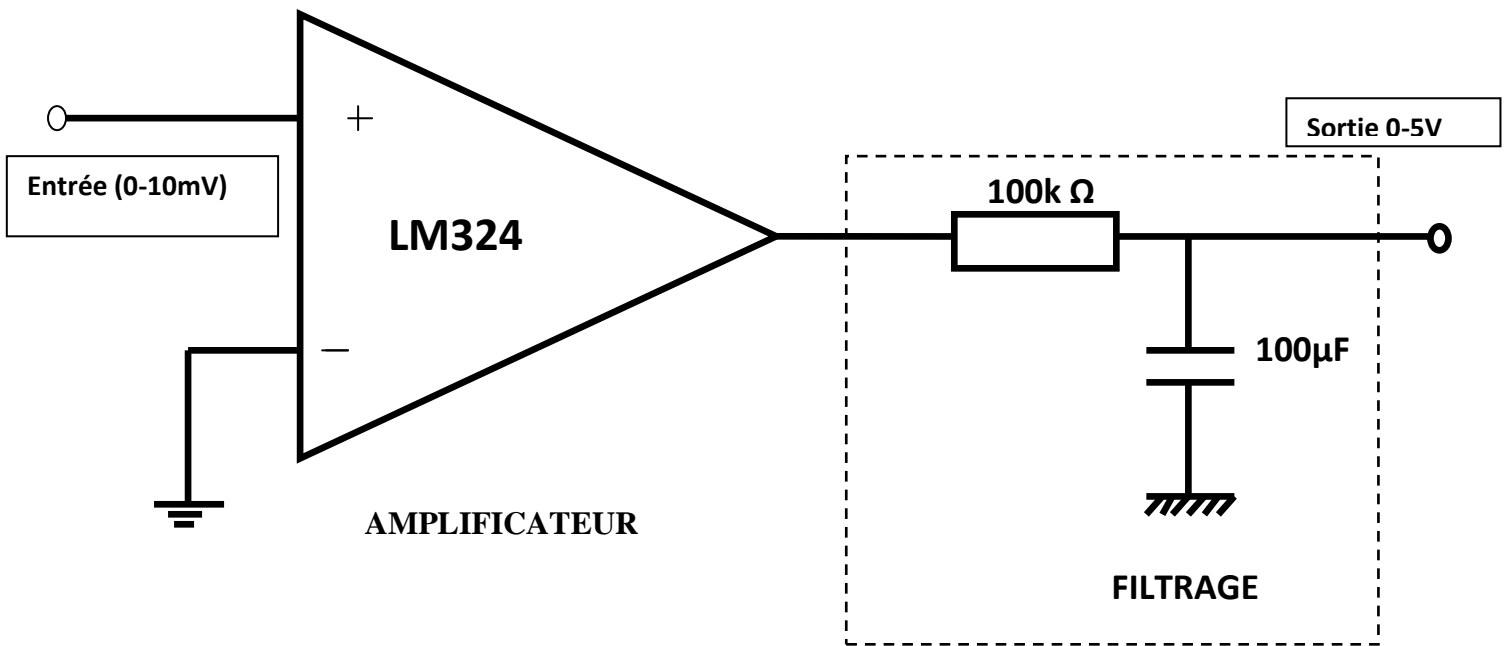


Figure12a. Schéma du circuit électronique de mise en forme

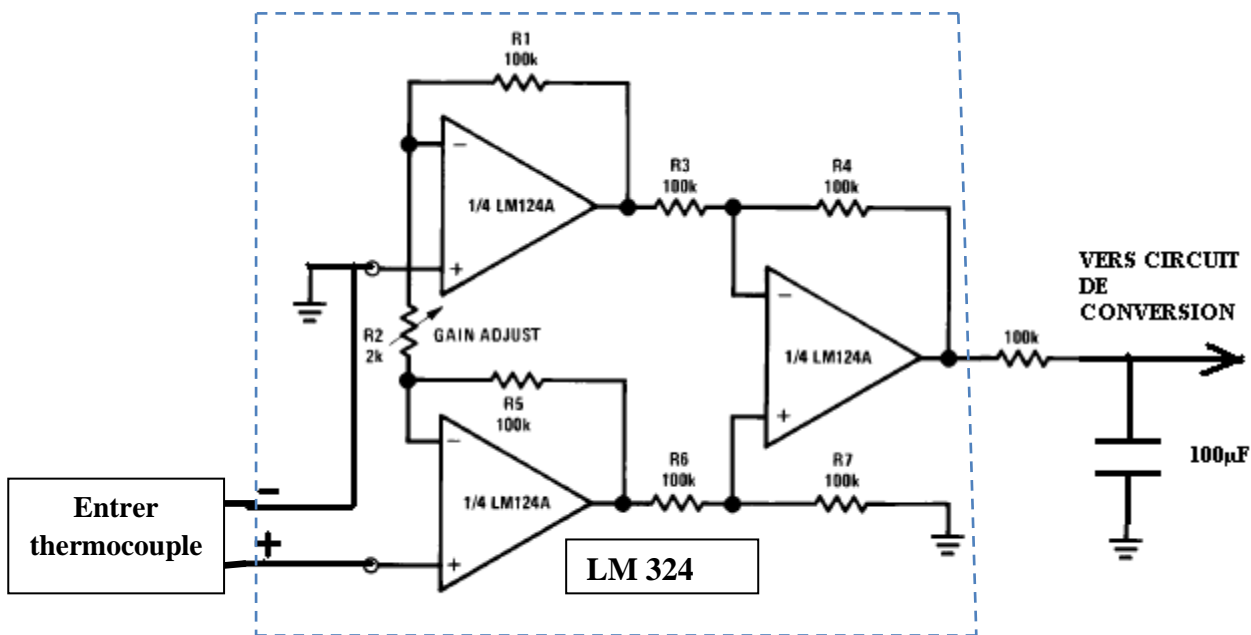


Figure12b. Schéma équivalent du circuit électronique de mise en forme

Ce dispositif comprend :

- Un amplificateur opérationnel (AO) de référence « LM324 »,
- Un filtre qui est composé d'une résistance de 100 KΩ et d'un condensateur de 100μF.

Ce dispositif a pour rôle d'amplifier la tension fournie par le thermocouple

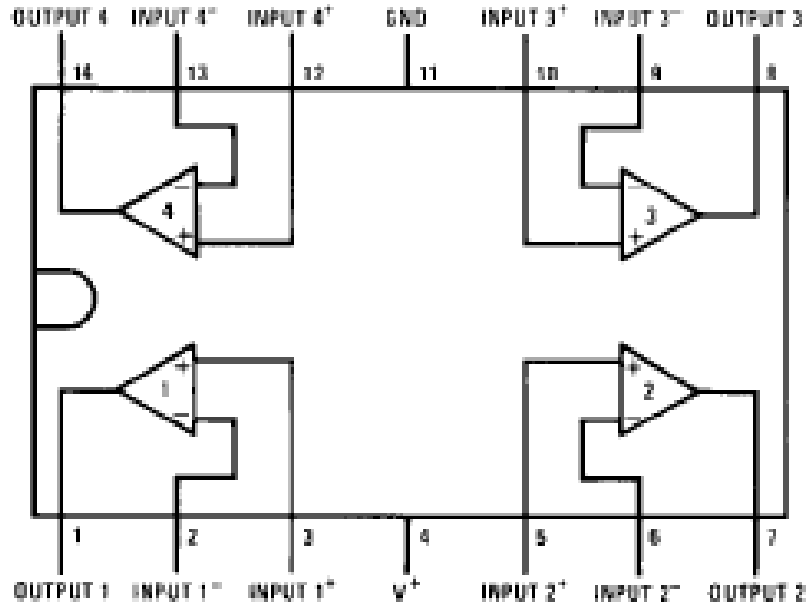


Figure 13. Brochage du LM324

3.5 CIRCUIT DE CONVERSION ET DE STOCKAGE SCHEMA BLOC

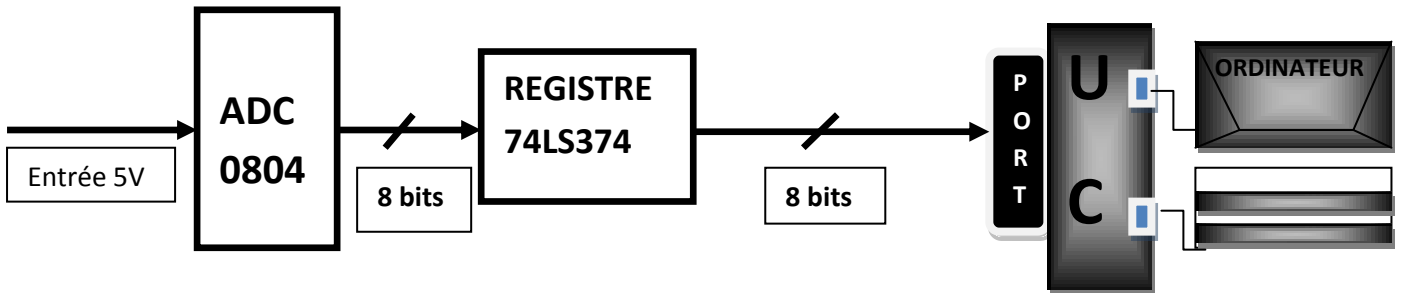


Figure 14. Schéma bloc de circuit de conversion et de stockage

Schéma du circuit électronique de conversion et de stockage relié avec le port

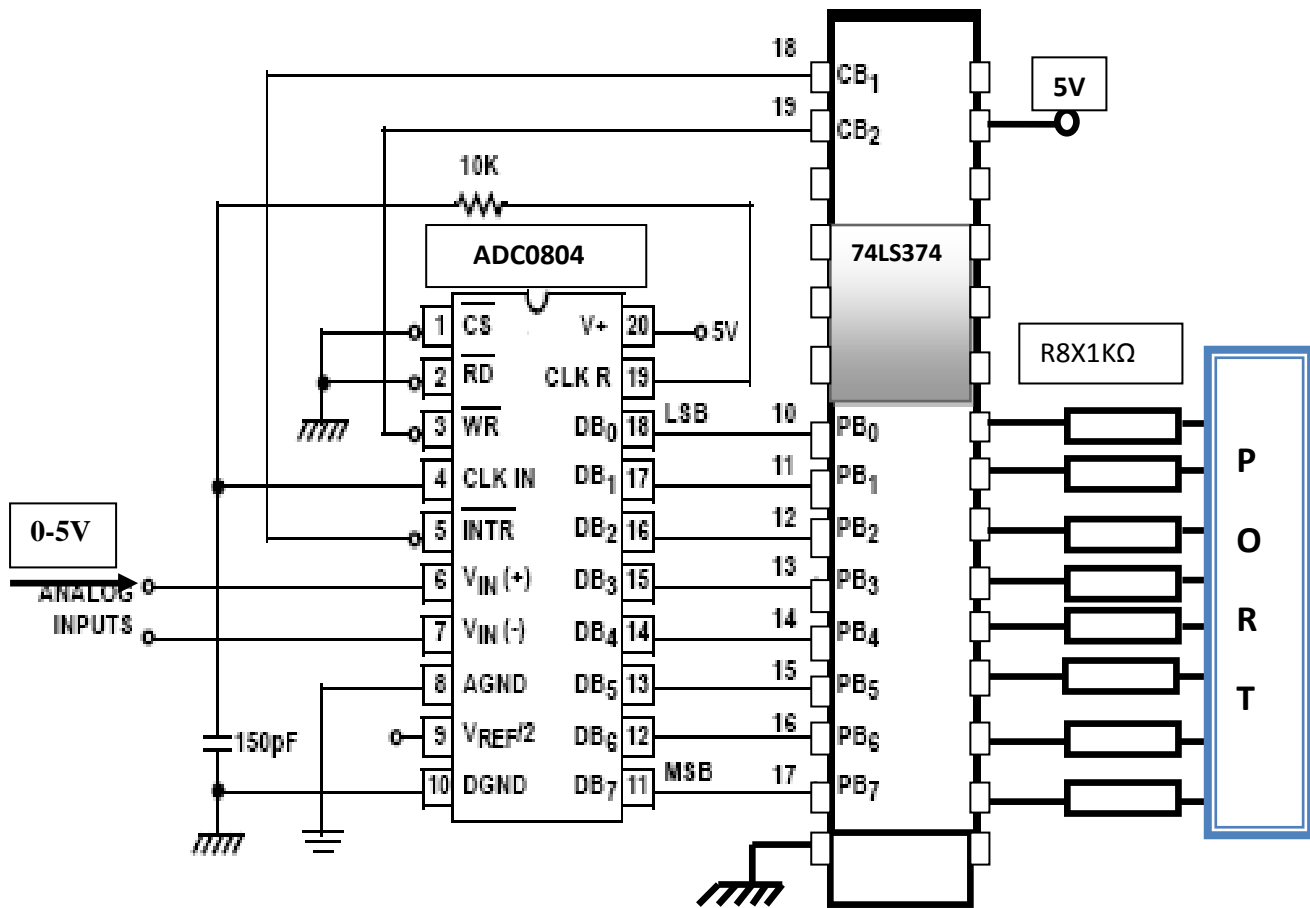


Figure 15. Schéma du circuit électronique de conversion et de stockage

Ce dispositif comprend :

- Un circuit intégré de conversion de référence « ADC 0804 »,
- Un registre de stockage de référence « 74LS374 ».

Ce dispositif a pour rôle de convertir la tension provenant du thermocouple qui est déjà amplifié par l'A.O. A la sortie de l'ADC 0804, la valeur binaire obtenue est stockée dans le registre 74LS374.

Symbole de l'ADC avec ses composants de sécurité

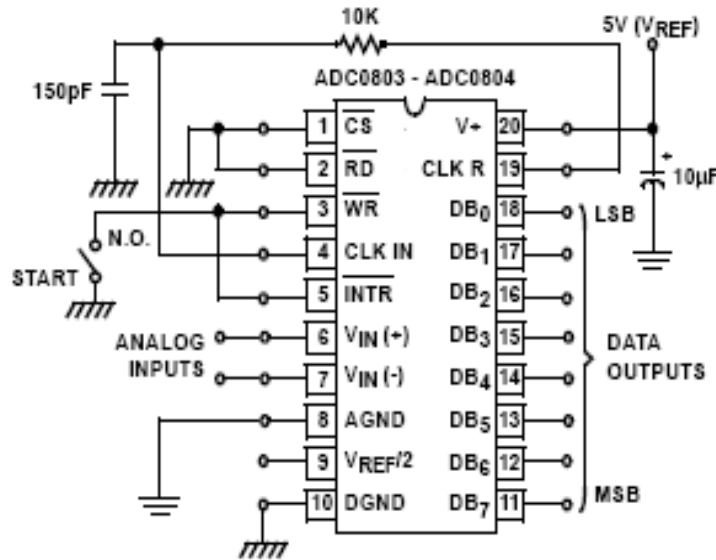


Figure 16. Symbole de l'ADC avec ses composants de sécurité

Symbole du registre

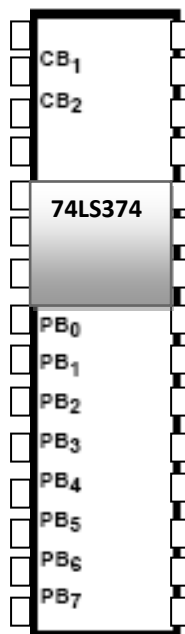


Figure 17. Symbole du registre

3.6 SCHEMA DU CIRCUIT ELECTRONIQUE D'ACQUISITION

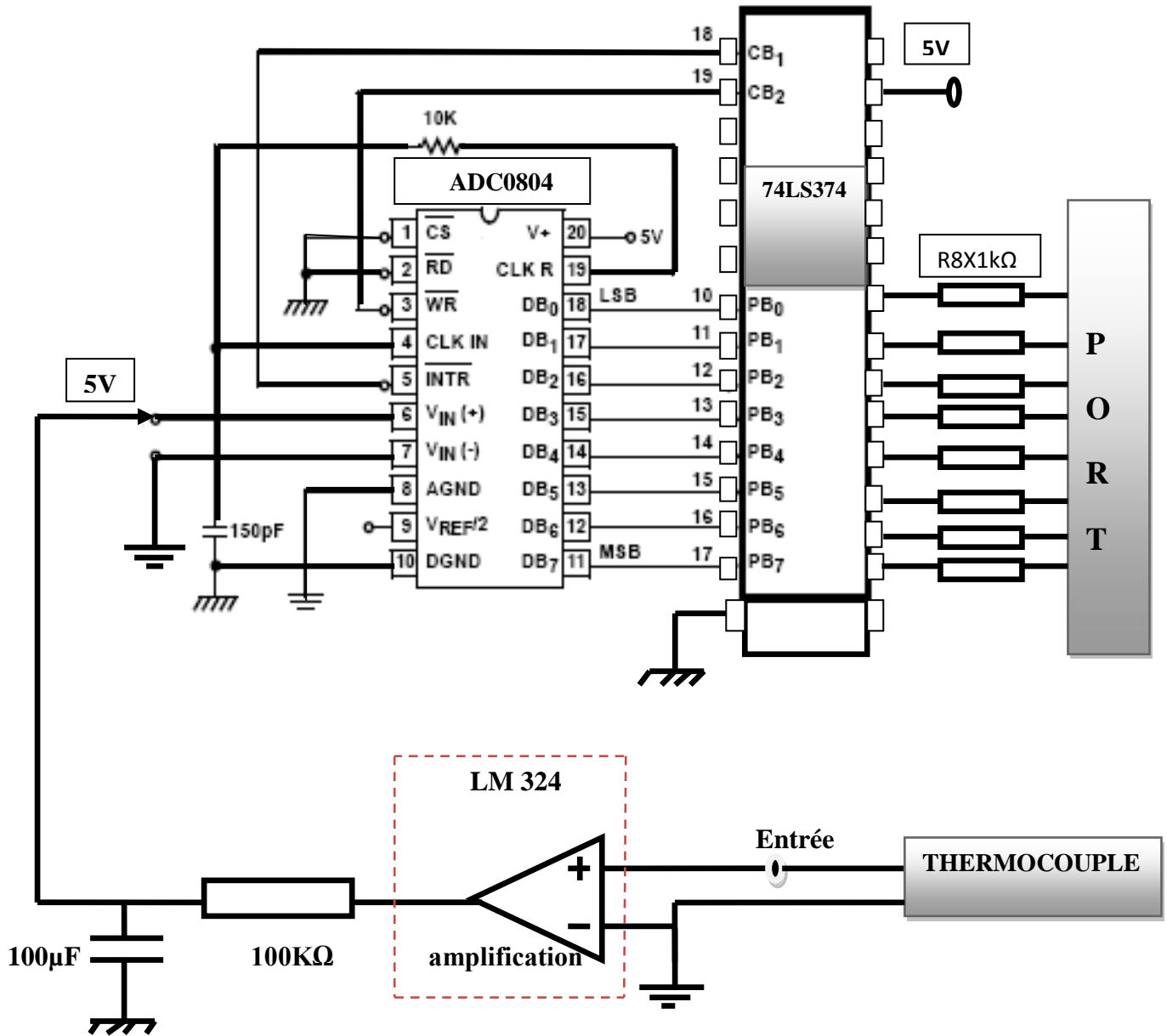


Figure 18. Schéma du circuit électronique d'acquisition

3.7 CIRCUIT DE COMMANDE DU VENTILATEUR ET DE LA RESISTANCE

a) Schéma bloc pour la commande

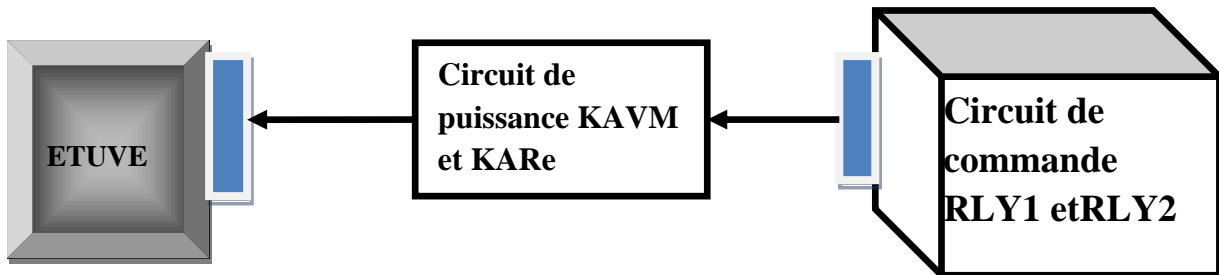


Figure 19. Schéma bloc pour la commande du ventilateur et de la résistance

Le circuit est composé de :

RLY1 : Relais de 12V qui commande le ON/ OFF du moto-ventilateur de l'étuve

RLY2 : Relais de 12V qui commande le ON/ OFF de la résistance de l'étuve.

KAVM : Circuit de puissance du moto-ventilateur

KARE : Circuit de puissance de la résistance

b) Commande de KAVM et KARE par RLY1 et RLY2

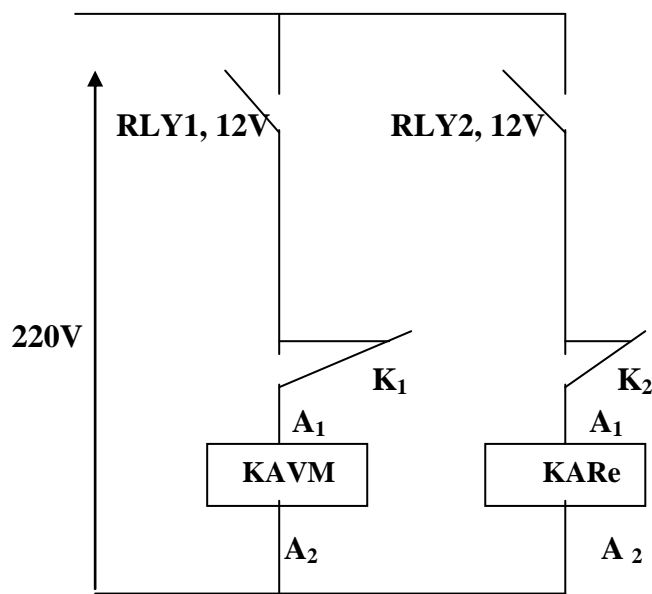


Figure 20. Commande des deux relais

RLY1 est relié avec KAVM à l'aide d'un contacteur K_1

RLY2 est relié avec KARE à l'aide d'un contacteur K_2

c) Schéma détaillé pour la commande, avec les composants électroniques, qui pilote KAVM et KARE

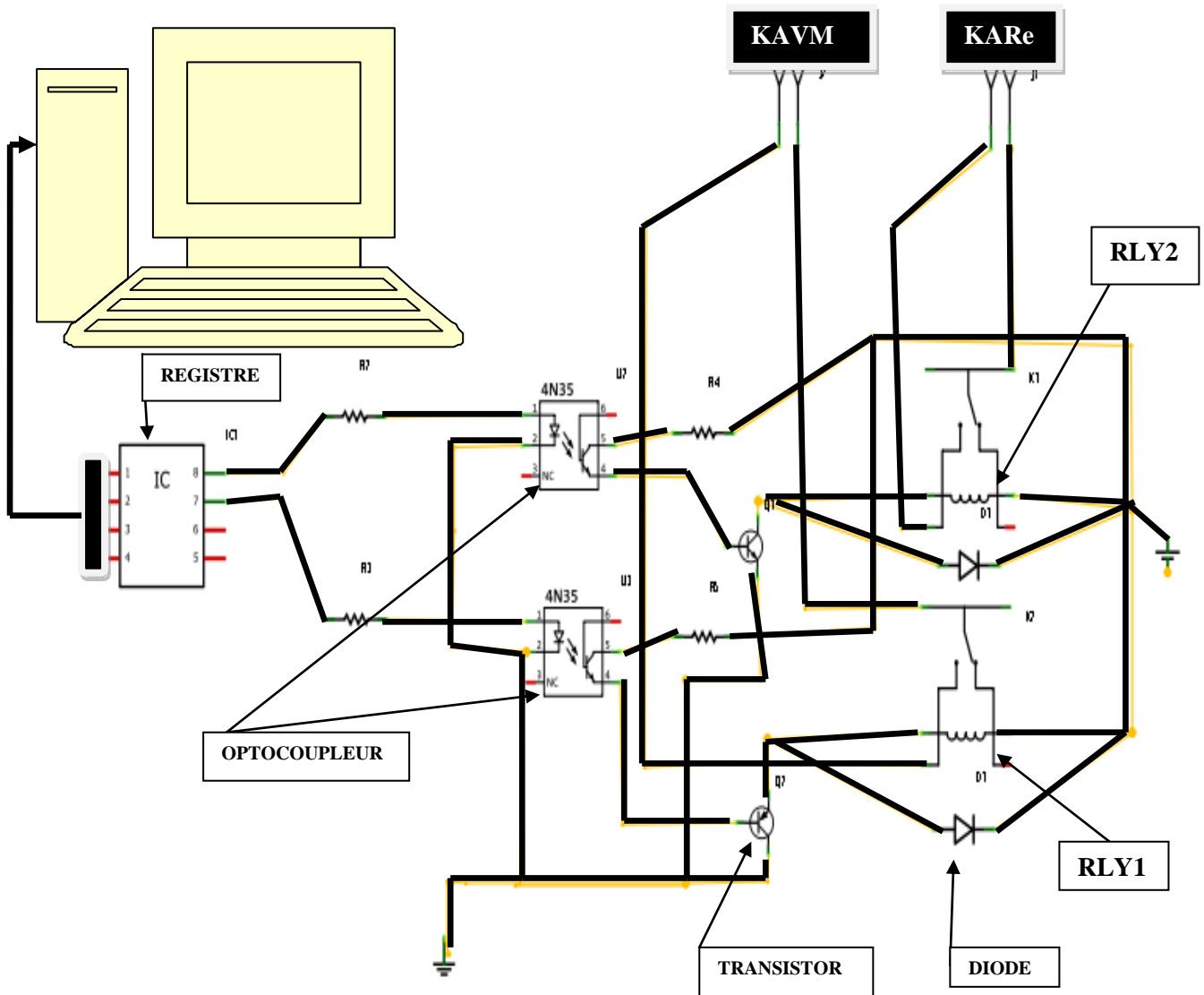


Figure 21. Schéma détaillé de la commande

Les composants électroniques pour cette commande sont :

- Deux diodes
- Deux transistors « Darlington »
- Deux optocoupleurs
- Quatre résistances
- Un registre

3.8 ROLES DE CES COMPOSANTS DANS LA COMMANDE

3.8.1. Les diodes (1N4007)

Les deux diodes protègent le circuit de commande des surtensions éventuelles, comme celles causées par l'ouverture des relais.

Symbole

La diode se compose d'une jonction **PN**, **P**ositif **N**égatif (figure 1) généralement en germanium ou en silicium.

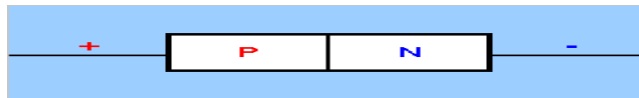


Figure 22. Symbole de la diode

3.8.2. Le transistor « Darlington »

Le transistor en commutation est utilisé afin d'ouvrir ou de fermer le circuit. Ainsi il peut commander le relais et aussi nous allons l'utiliser comme amplificateur de courant. Le montage d'un transistor en commutation peut être décomposé en deux circuits :

- ✓ **Circuit de commande ou Circuit d'entrée**
- ✓ **Circuit commuté ou circuit de sortie**

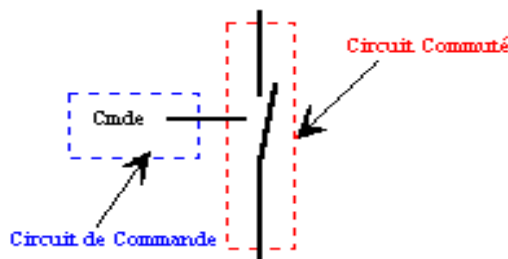


Figure 23. Circuit de commande et Circuit commuté

Transistor NPN

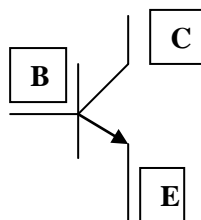


Figure 24. Transistors NPN

❖ Caractéristiques du transistor

- Type NPN
- La variation de claquage entre le collecteur et l'émetteur à $I_B = 0$ (V_{CEO})

- Le gain en continu (h_{fe}) à I_C et V_{CE} données, $G=5000$, $V_{CE}=24V$
- Courant de collecteur maximal $I_C= 4A$
- La puissance de dissipation maximale $P_{tot}=20W$

3.8.3.L'Optocoupleur

Son principal but est de protéger le thermocouple, et l'électronique de commande des courts circuits et d'autres éventuels éléments de perturbation.

❖ Caractéristiques de l'optocoupleur

- Un optocoupleur est l'association dans un seul boîtier d'une diode électroluminescente (LED) et d'un phototransistor, placés en regard l'un de l'autre, mais isolés électriquement,
- Du point de vue électrique, l'optocoupleur sera considéré comme un quadripôle dont les 2 bornes de sortie sont les bornes C et E du phototransistor et les 2 bornes d'entrées sont les bornes A et K de la LED,

Symbole de l'optocoupleur

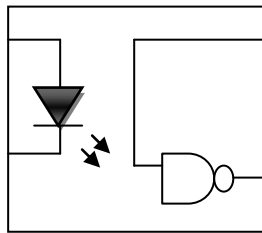


Figure 25. Symbole de l'optocoupleur

3.8.4.La résistance

C'est le support qui protège le registre et l'optocoupleur d'un retour subit de courant.

3.9 CIRCUIT DE PUISSANCE DU VENTILATEUR ET DE LA RESISTANCE DE L'ETUVE

3.9.1. SCHEMA DE PUISSANCE

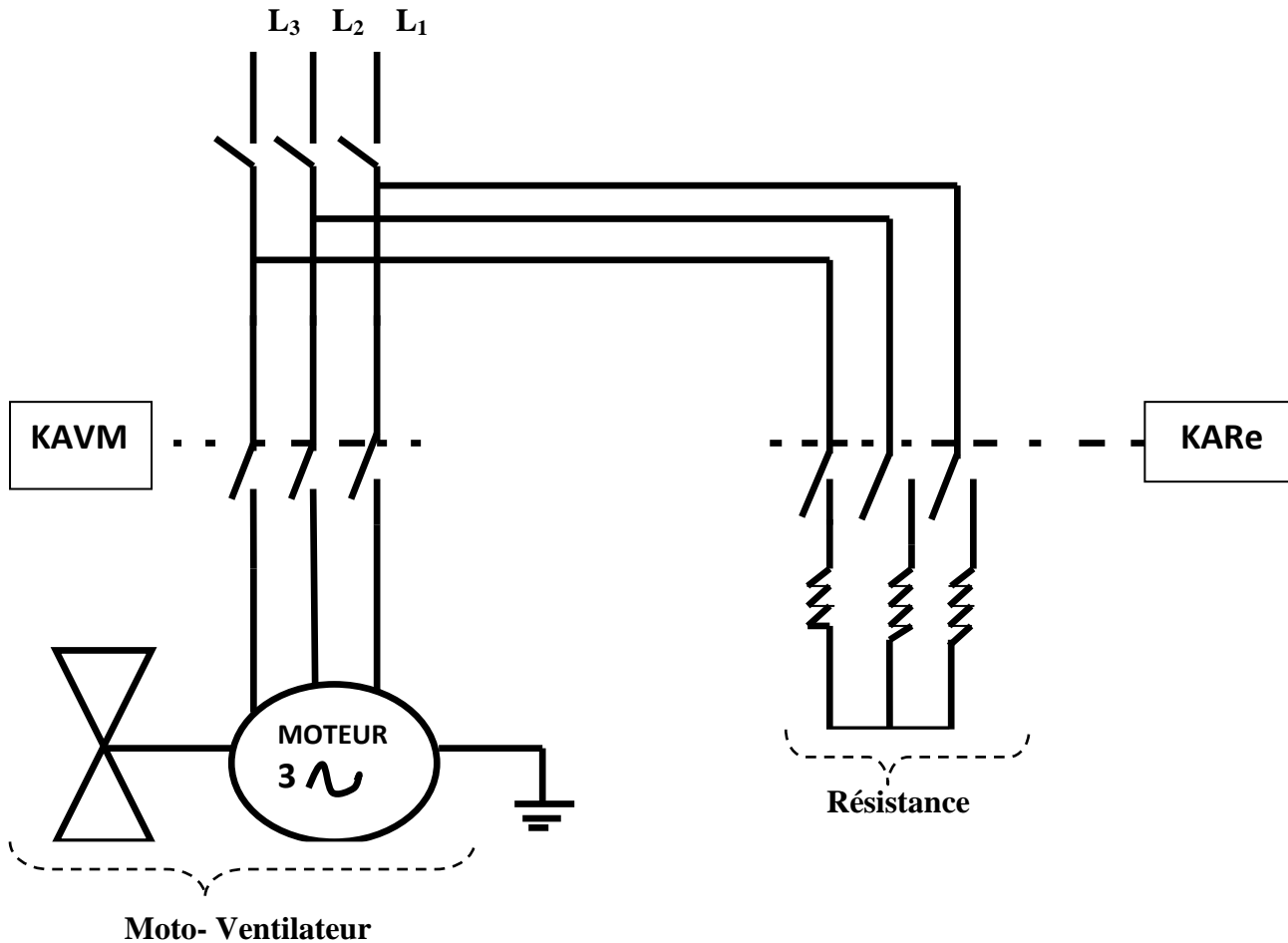


Figure 26. Schéma de puissance

3.9.2. Fonctionnement

Le moto- ventilateur est triphasé et alimenté en étoile.

3.9.3. Paramètres d'utilisation

- Tension nominale d'alimentation : 380V
- Courant nominal : 1,5A
- Puissance utile : 2,5 KW
- Facteur de puissance $\cos\varphi = 0,6$
- Vitesse nominale : 1450 tr/min
- Fréquence : 50Hz

CHAPITRE IV : PORT PARALLELE DE L'ORDINATEUR

4.1. Introduction

Le port parallèle de l'ordinateur nommé aussi LTP (Local Printer) sert à l'origine à se connecter à l'imprimante, mais peut aussi être utilisé pour se connecter à d'autres périphériques (Graveurs, Scanners, Disques durs...).

En effet il ne nécessite aucun protocole de transmission (contrairement au port série), et les niveaux électriques de ses broches varient entre 0 et 5V ce qui le rend compatible directement avec les composants de technologie TTL (Transistors Transistors Logic).

Il existe différents types de liaison parallèle, définis dans le standard IEEE 1284 (Institute of Electrical and Electronics Engineers) :

- mode compatible (SPP, Standard Parallèle Port) (unidirectionnel)
- mode 4 bits, ou NIBBLE MODE (unidirectionnel)
- mode 8 bits, ou BYTE MODE (bidirectionnel)
- mode EPP (Enhanced Parallèle Port, port parallèle étendu) (bidirectionnel)
- mode ECP (Extended Capability Port, port à capacités étendues) (bidirectionnel)

Ceux-ci sont pour la plupart paramétrables dans le BIOS (setup).

Le mode EPP est présent sur la plupart des machines actuelles, et on le préférera si l'on désire avoir les 8 bits de données en entrée et sortie, ou on choisira le mode compatible SPP présent sur toutes les machines si on désire avoir les 8 bits de données seulement en sortie.

Outre ces 8 lignes de données, le port parallèle dispose également de 4 lignes de contrôle (sortie) et 5 lignes d'état (entrée).

4.2. REPRESENTATION DU PORT PARALLELE

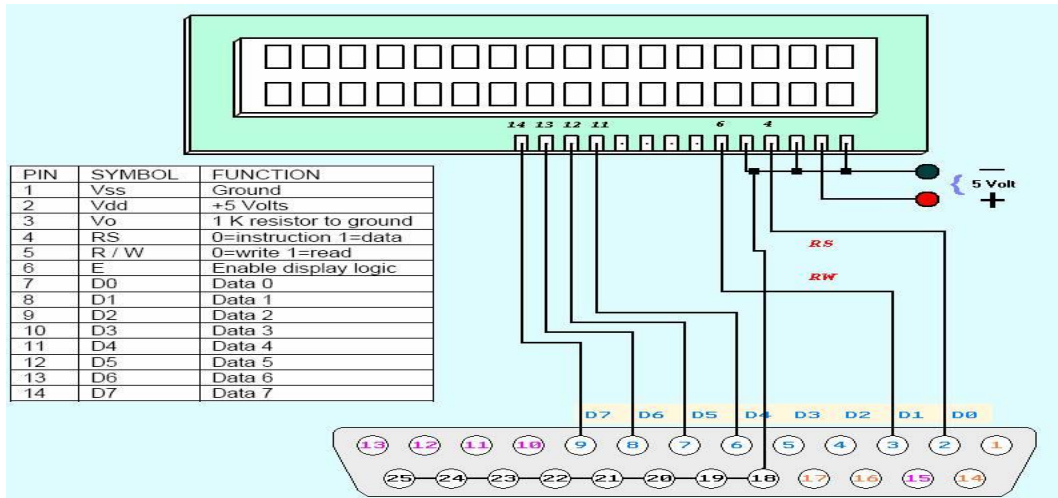


Figure 27. Représentation du port parallèle

4.3. LES COMPOSANTS DE BASE DU PORT PARALLELE

a) Connecteurs

Voici la numérotation des broches des deux principaux connecteurs parallèles, DB25 broches du côté du PC et Centronics 36 broches du côté de l'imprimante :

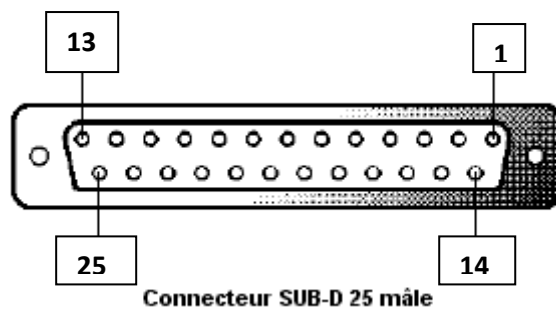


Figure 28. Connecteur SUB-D 25male

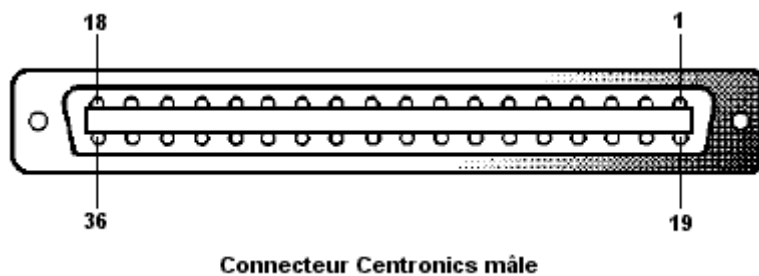


Figure 29. Connecteur Centronics male

Tableau des brochages de ces connecteurs :

connecteur SUB-D 25	connecteur Centronics	Fonction	Niveau de repos	Direction
1	1	Strobe	1	S
2	2	donnée D0	0	E/S
3	3	donnée D1	0	E/S
4	4	donnée D2	0	E/S
5	5	donnée D3	0	E/S
6	6	donnée D4	0	E/S
7	7	donnée D5	0	E/S
8	8	donnée D6	0	E/S
9	9	donnée D7	0	E/S
10	10	Acknowledge	1	E
11	11	Busy	0	E
12	12	Paper end	0	E
13	13	Select	0	E
14	14	Autofeed	1	S
15	32	Error	1	E
16	31	Initialize	1	S
17	36	Select input	1	S
18-25	17, 33, 19-29	Ground		

Tableau 1. Les brochages de connecteurs

On retrouve 5 lignes en entrée, 4 en sortie, et les 8 lignes de données en entrée-sortie dans le cas d'un mode bidirectionnel.

b) Registre du port parallèle

Le registre du port parallèle se divise en trois étapes :

- Registre de données : ce registre se fait en écriture ;

- Registre d'état : ce registre se fait uniquement en lecture au contraire du précédent. Il ne faut surtout pas écrire dans ce registre ;
- Registre de contrôle : ce dernier registre est accessible à la fois en lecture et en écriture ;

Tableau de classification de registres

Registre de données	bit 0	D0
	bit 1	D1
	bit 2	D2
	bit 3	D3
	bit 4	D4
	bit 5	D5
	bit 6	D6
	bit 7	D7
Registre d'état	bit 3	Error
	bit 4	Select
	bit 5	Paper end
	bit 6	Acknowledge
	bit 7	Busy
Registre de contrôle	bit 0	Strobe
	bit 1	Autofeed
	bit 2	Initialize
	bit 3	Select input
	bit 4	Autorisation d'interruption
	bit 5	direction données (0 = sortie, 1 = entrée)

Tableau 2. Classification de registres

c) Adressage du port parallèle

Différentes adresses peuvent exister pour les registres selon le système. Le tableau suivant en présente les différentes possibilités (0x signifie que les valeurs sont en hexadécimal) :

port de données	0x956	0x3BC	0x888	0x378	0x632	0x278
port d'état	0x957	0x3BD	0x889	0x379	0x633	0x279
port de contrôle	0x957	0x3BE	0x89A	0x37A	0x634	0x27A

Tableau 3. Adresse du port parallèle

Sur les systèmes actuels, le port LPT1 se situe toujours à l'adresse 0x378 et le port LPT2 à l'adresse 0x278.

4.4. DESCRIPTION DES SIGNAUX

STROBE : cette ligne active basse (donc à 0) indique à l'imprimante que des données sont présentées sur les lignes D0 à D7 et qu'il faut prendre en compte.

D0 à D7 : c'est le bus de données qui véhicule la valeur du caractère à imprimer. On ne peut écrire sur ce port, à moins d'avoir un port parallèle étendu (c'est le cas pour le port de type ECP/EPP).

ACK : l'imprimante met à 0 cette ligne pour indiquer à l'ordinateur qu'elle a bien reçu les caractères transmis qu'il peut continuer la transmission.

BUSY : cette ligne est mise à 0 par l'imprimante lorsque son buffer de réception est plein. L'ordinateur est ainsi averti que celle-ci ne peut plus recevoir de données. Il doit attendre que cette ligne à 1 pour recommencer à émettre.

PE : signifie "paper end". L'imprimante indique par cette ligne à l'ordinateur que l'alimentation en papier a été interrompue.

SELECT : cette ligne indique à l'ordinateur si l'imprimante est "on line" ou "off line".

AUTOFEED : lorsque ce signal est à 1, l'imprimante doit effectuer un saut de ligne à chaque caractère "return" reçue. Certaines imprimantes se contentent d'effectuer un simple retour du chariot en présence de ce caractère.

ERROR : indique à l'ordinateur que l'imprimante a détecté une erreur.

INIT : l'ordinateur peut effectuer une initialisation de l'imprimante par l'intermédiaire de cette ligne.

SELECT IN : l'ordinateur peut mettre l'imprimante hors ligne par l'intermédiaire de ce signal.

GROUND : c'est la masse du PC.

- ***Port parallèle en mode étendu***

Le Port Parallèle Étendu (EPP) est un nouveau port parfois également appelé "Port parallèle mode rapide". Il a été mis au point par INTEL, XIRCOM et ZENITH DATA SYSTEMS, et sa création a été annoncée au mois d'octobre 1991. Les premiers produits à intégrer ce type de port ont été les laptops Zenith Data Systems, les adaptateurs de réseaux locaux de poche Xircom et la puce d'E/S Intel 82360 SL. Ce port parallèle fonctionne quasiment à la vitesse d'un bus ISA et offre un débit brut dix fois plus important qu'un port parallèle conventionnel. Il est particulièrement adapté aux périphériques à port parallèle tels que des adaptateurs LAN, des disques durs et des dérouleurs de bandes. Les caractéristiques du port parallèle étendu ont été incluses dans le standard de port parallèle IEEE1284, il permet d'atteindre des débits de 1 à 2 Mo (Méga-octets) par seconde.

- ***Le port à capacité étendue***

Microsoft et Hewlett Packard ont mis au point un autre port parallèle à grande vitesse : le Port à Capacité Étendue (ECP), annoncé officiellement en 1992. Tout comme le port parallèle étendu, ce port constitue une version améliorée du port parallèle et requiert un circuit logique spécial. Depuis son lancement officiel, le port à capacités étendues a été inclus dans le standard IEEE 1284, tout comme le port parallèle étendu, mais, contrairement à ce dernier, il n'est pas conçu spécifiquement pour recevoir des périphériques de port parallèle de PC portables, son rôle étant de permettre de connecter une imprimante de hautes performances pour un coût modique. De surcroît, le fonctionnement en mode de port à capacité étendue requiert l'utilisation d'un canal d'accès mémoire direct (DMA), qui ne fait pas partie des spécifications du port parallèle étendu et qui peut provoquer des conflits avec les autres périphériques utilisant des canaux DMA. La plupart des nouveaux PC équipés de puces "super I/O" (super E/S) sont compatibles avec le mode port parallèle étendu (EPP) ou le mode port parallèle à capacités étendues (ECP). Le mode port parallèle étendu (EPP) est généralement le mieux adapté aux périphériques de port parallèle.

- *Les interruptions*

Lorsqu'une machine fonctionne, sous le contrôle d'un système d'exploitation, l'unité centrale est en permanence susceptible d'exécuter un programme. Le problème se pose alors de savoir à quel moment l'unité centrale va pouvoir prendre en compte des événements extérieurs à la séquence d'instruction qu'elle exécute : requête d'un périphérique, frappe d'une touche par exemple. A chacun de ces événements correspond une tâche à exécuter par l'unité centrale. Cette tâche est codée sous forme d'une procédure du système d'exploitation. Pour pouvoir exécuter cette procédure il faut que se produise une rupture de séquence.

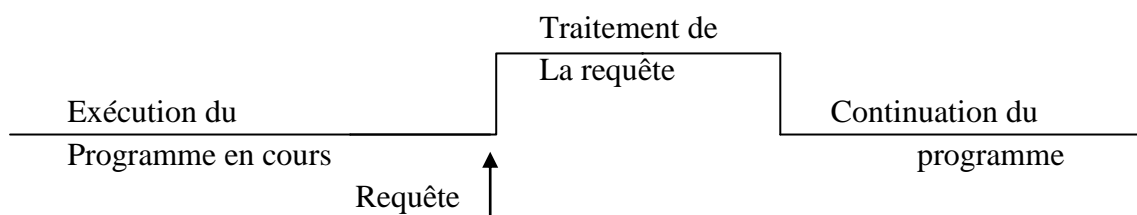


Figure 30 .Principe de port parallèle

- *Exceptions et interruptions*

Une interruption est une classe particulière d'exceptions externes (asynchrones) provoquées par un événement déclenché par du matériel extérieur au processeur qui se distingue des exceptions synchrones par la source (le processeur lui-même pour les exceptions synchrones, un matériels externe pour l'interruption). Exceptions et interruptions sont utilisées pour communiquer entre le processeur et les matériels extérieurs, erreurs internes et gestion de conditions spéciales utilisation concurrente du matériel et gestion des demandes de service.

CHAPITRE V. ETUDE ET CONCEPTION DU LOGICIEL DE PILOTAGE DE L'ETUVE

5.1. METHODOLOGIE

Comme méthodologie nous avons utilisé la méthodologie de l'Analyse et Conception Orientée Objet (OOAD : Object Oriented Analysis And Designed).

Il s'agit de l'analyse et de la conception du logiciel pour piloter les deux circuits électroniques que nous avons conçu précédemment à savoir :

- Le circuit d'acquisition
 - Le circuit de commande
- } de l'étuve

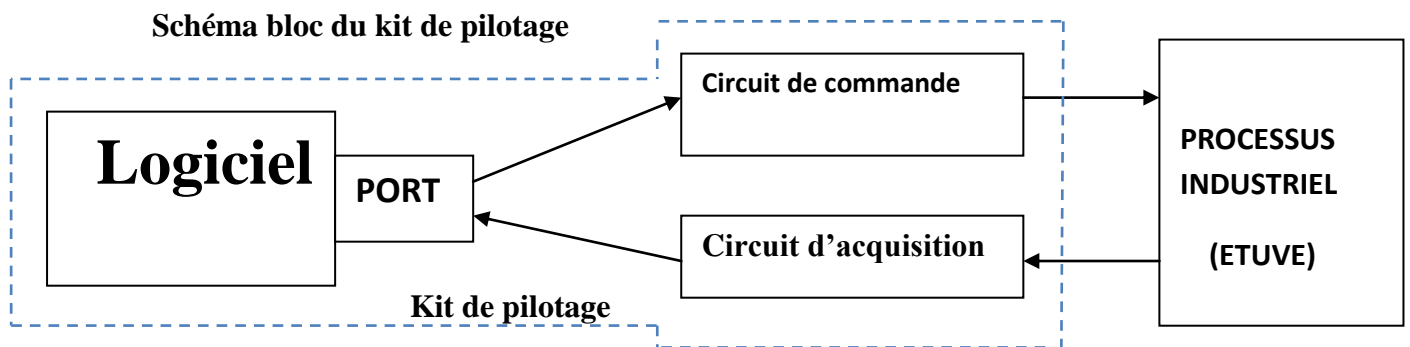


Figure 31. Schéma bloc du kit de pilotage

On a utilisé le langage UML (Unified Modeling Language)

5.2. LES CAS D'UTILISATIONS (USE CASES)

IDENTIFICATION DES DIFFERENTS UTILISATEURS DU KIT :

Ce sont les professeurs et les étudiants de l'ESPA .On peut dire que ces utilisateurs forment une seule catégorie.

Le professeur Monsieur RANARIJAONA Jean Désiré responsable du laboratoire thermique et son équipe ont représenté ces utilisateurs.

Après plusieurs entretiens avec ces utilisateurs on a pu identifier les cas d'utilisation selon le schéma suivant :

5.3. CAS D'UTILISATION

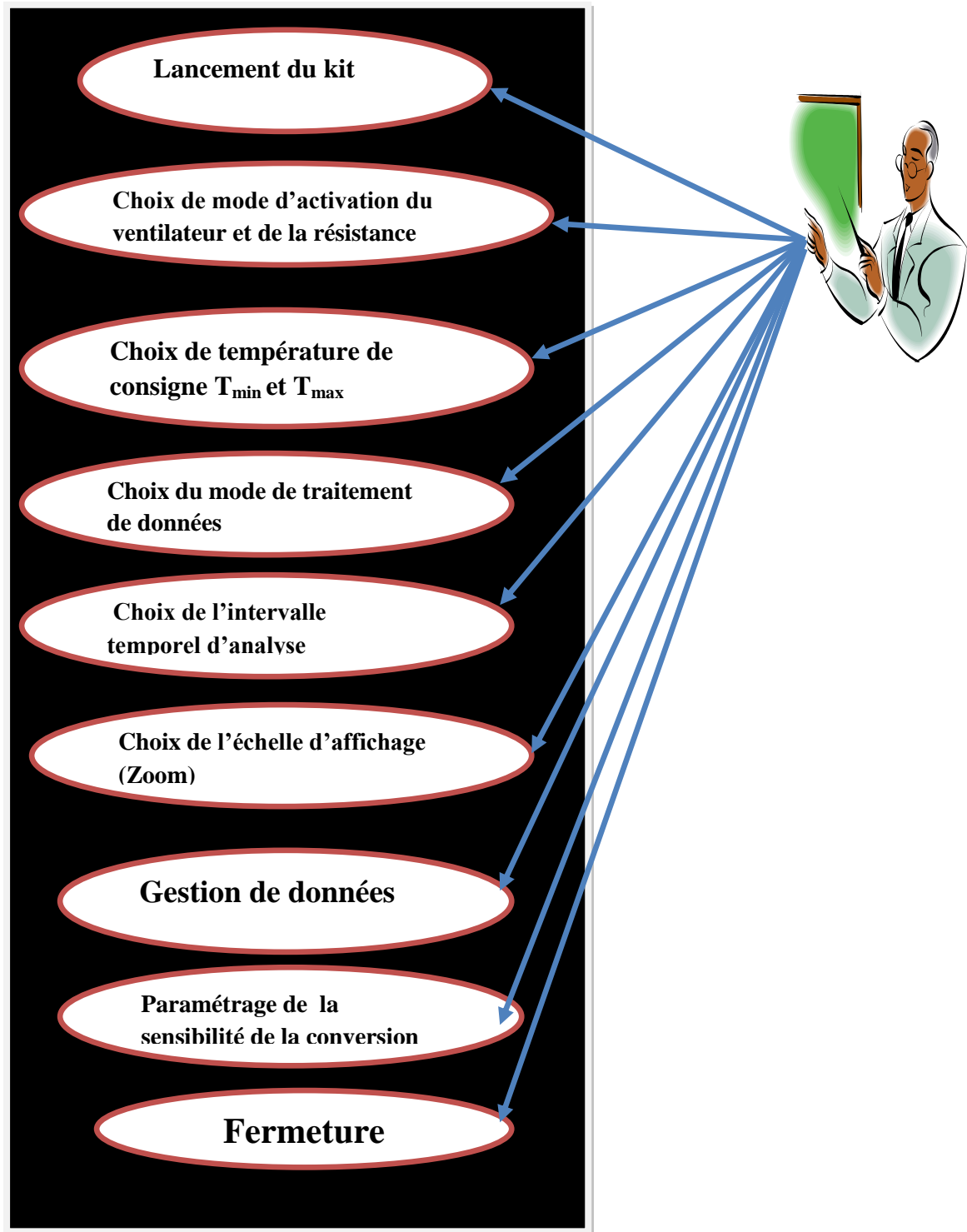


Figure 32.Schéma du cas d'utilisation

A partir de ces cas d'utilisation on a conçu l'interface graphique utilisateur suivant :

5.4. INTERFACE GRAPHIQUE UTILISATEUR « 6 J »

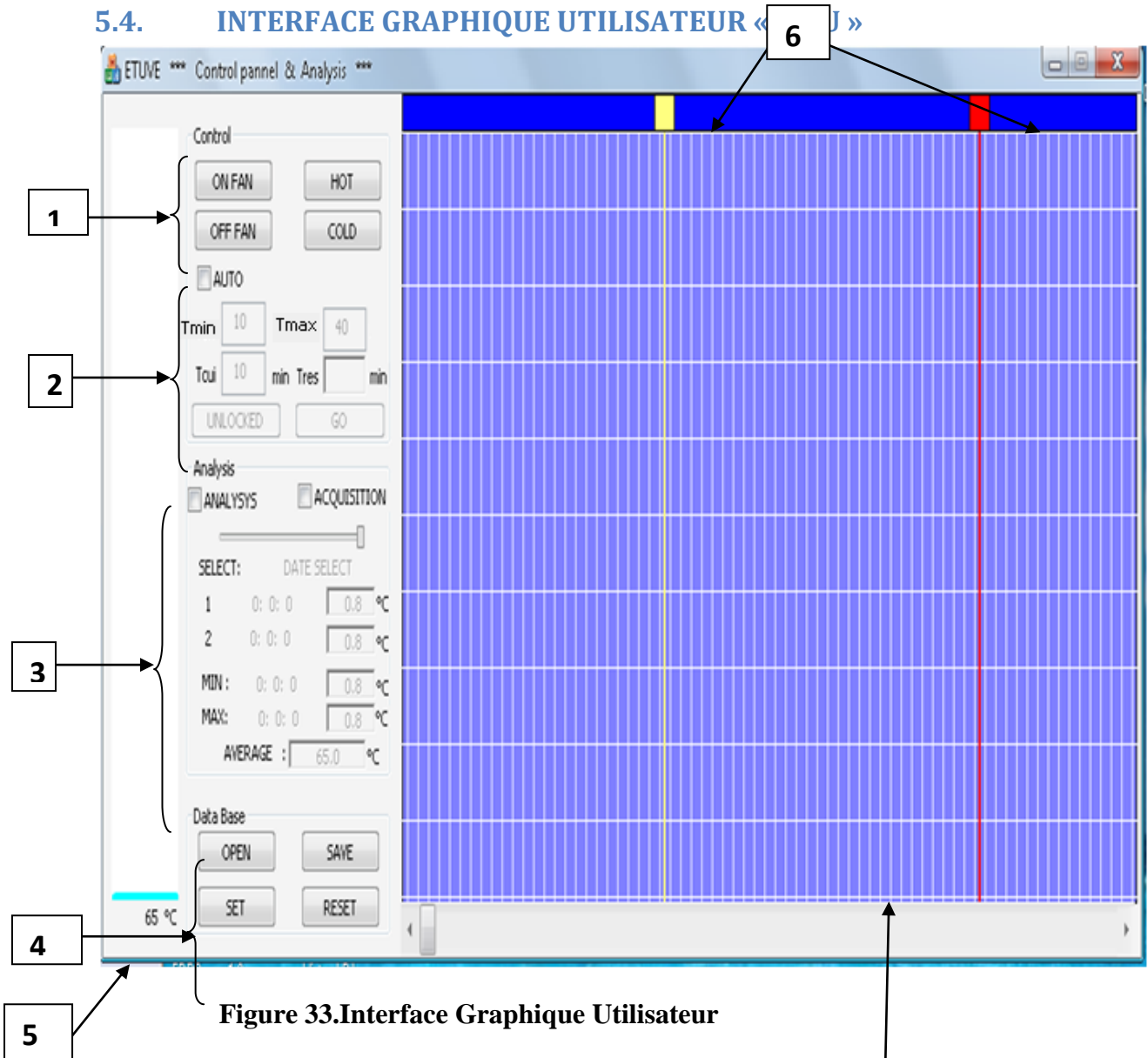


Figure 33. Interface Graphique Utilisateur

- 1 : Bouton de commande de l'étuve,
- 2 : Zone de paramètre de consigne,
- 3 : Zone de résultat d'analyse,
- 4 : Zone de l'outil de manipulation de la base de données,
- 5 : Zone de température actuelle à l'intérieur de l'étuve,
- 6 : Délimitateur de la zone à analyser,
- 7 : Ecran d'affichage graphique de l'évolution de la température.

5.5. STRUCTURE OBJET DU LOGICIEL

On adopté la structure classique M.V.C (Modele Vue Controleur) selon la figure suivante :

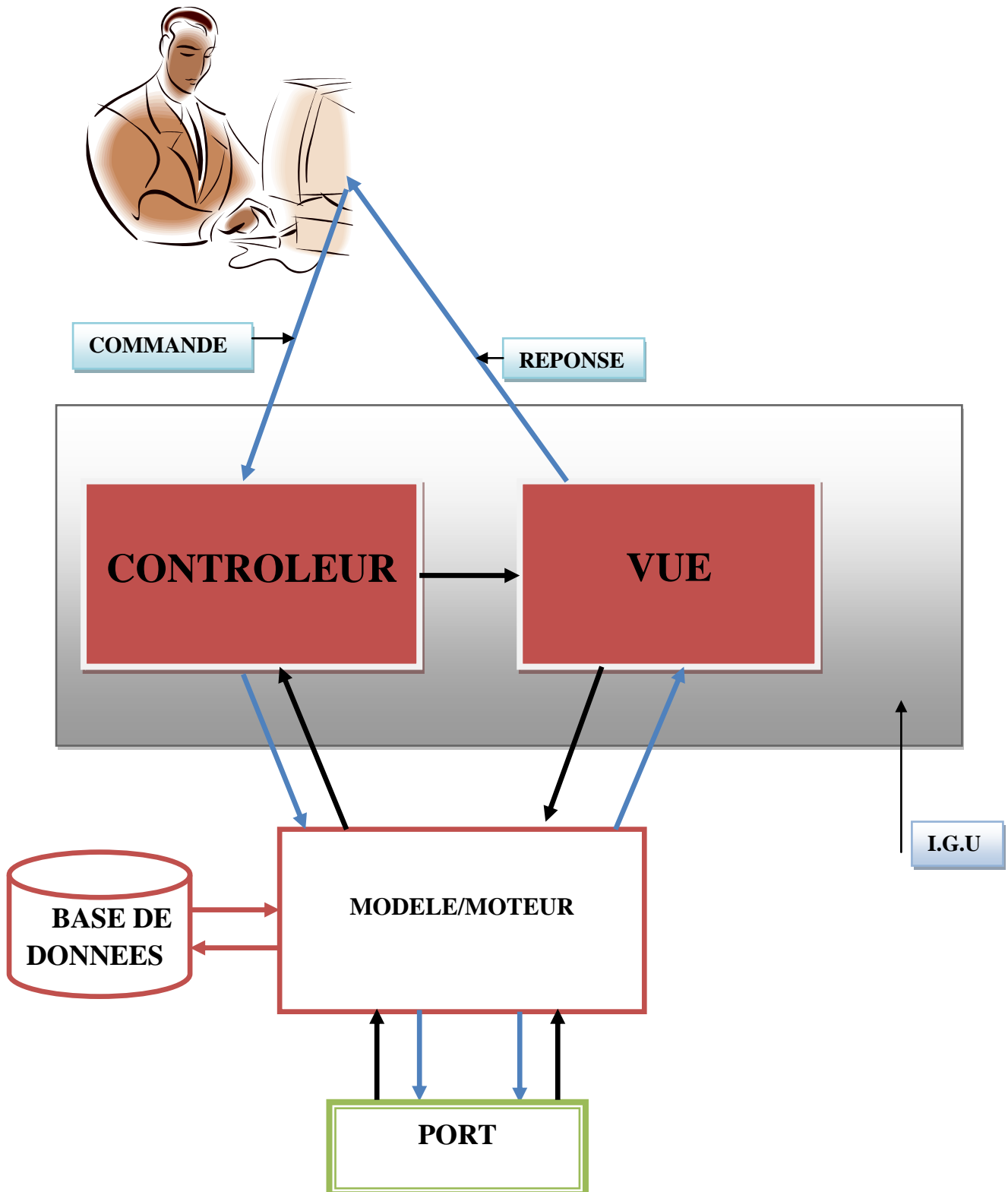


Figure 34.Structure M.V.C

5.6. CORRESPONDANCE ENTRE LES OBJETS DU : I.G.U ET M.V.C

Interface Graphique Utilisateur « I.G.U »	Model Vue Controleur « M.V.C »
[1] mode d'activation(OnFan/OffFan et Hot/Cold)	Controle
[2] choix de T_{min} et T_{max} de la temerature de consigne	Controle
[3] traitement de données,analyse,paramétrage	Controle
[4]Gestion de données	Controle
[5]echelle d'affichage de temperature	Vue
[6]intervalle temporelle d'analyse	Vue
[7] Ecran d'affichage graphique	Vue

Tableau 4. Correspondance entre les objets du : I.G.U ET M.V.C

5.7. MODELE / MOTEUR

Le modèle ou moteur de l'application est donné par le schéma suivant:

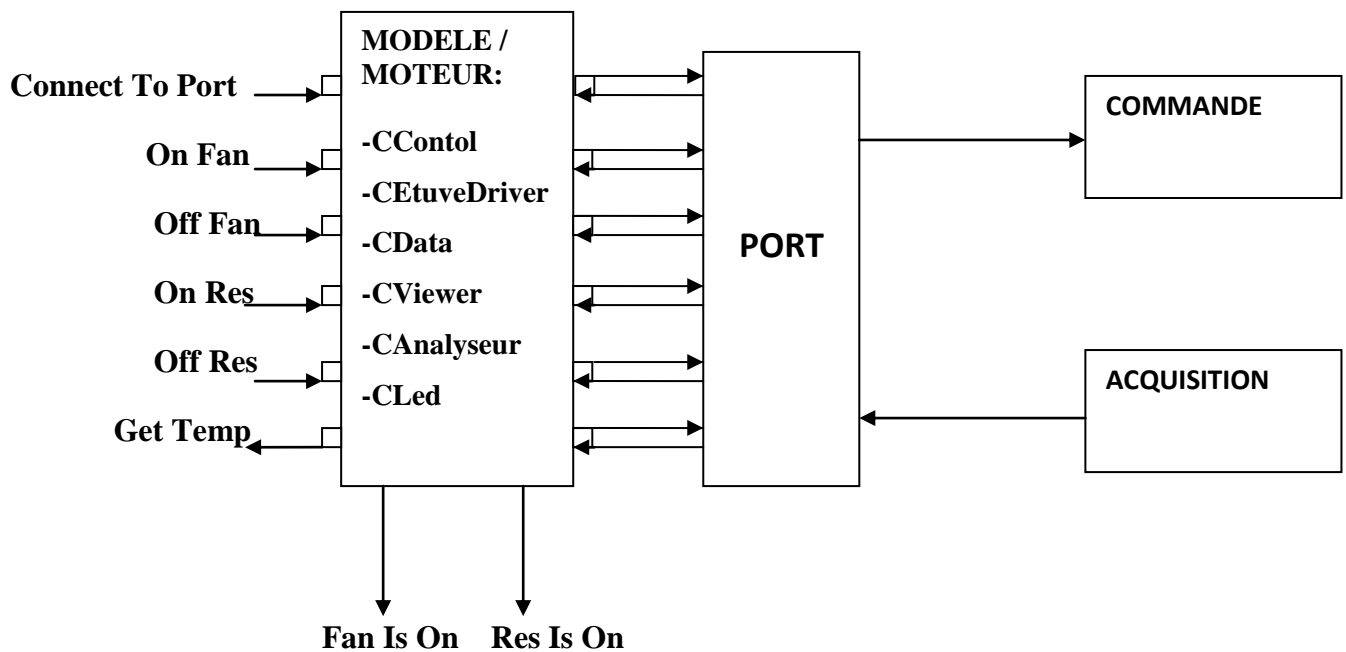


Figure 35.Modèle / Moteur

Après avoir identifié tous les objets composants le logiciels, nous avons les classes suivantes :

5.8. LES CLASSES

OBJET	CLASSE
-int ton, toff, tres, compte, min -on, go -old temp ,newtemp -hot	CContol
-c0, c1, c2, port, temp -a, b, c, d, moyenne, somme -res, fan	CEtuveDriver
getday,getmonth,getyear,gethour,getmin,getsec	CData
intx,inty,intlength,intwidth,intheight,intiepais getX _{min} , getX _{max} , getX _{moy} ,	CViewer
resmin, resmax, average int analmin, analmax, ix, xmin, xmoy, xmax	CAnalyseur
int ix, iy, ilong, iepais, ioldval	CLed

Tableau 5. Les Classes



5.9. DIAGRAMME DES CLASSES

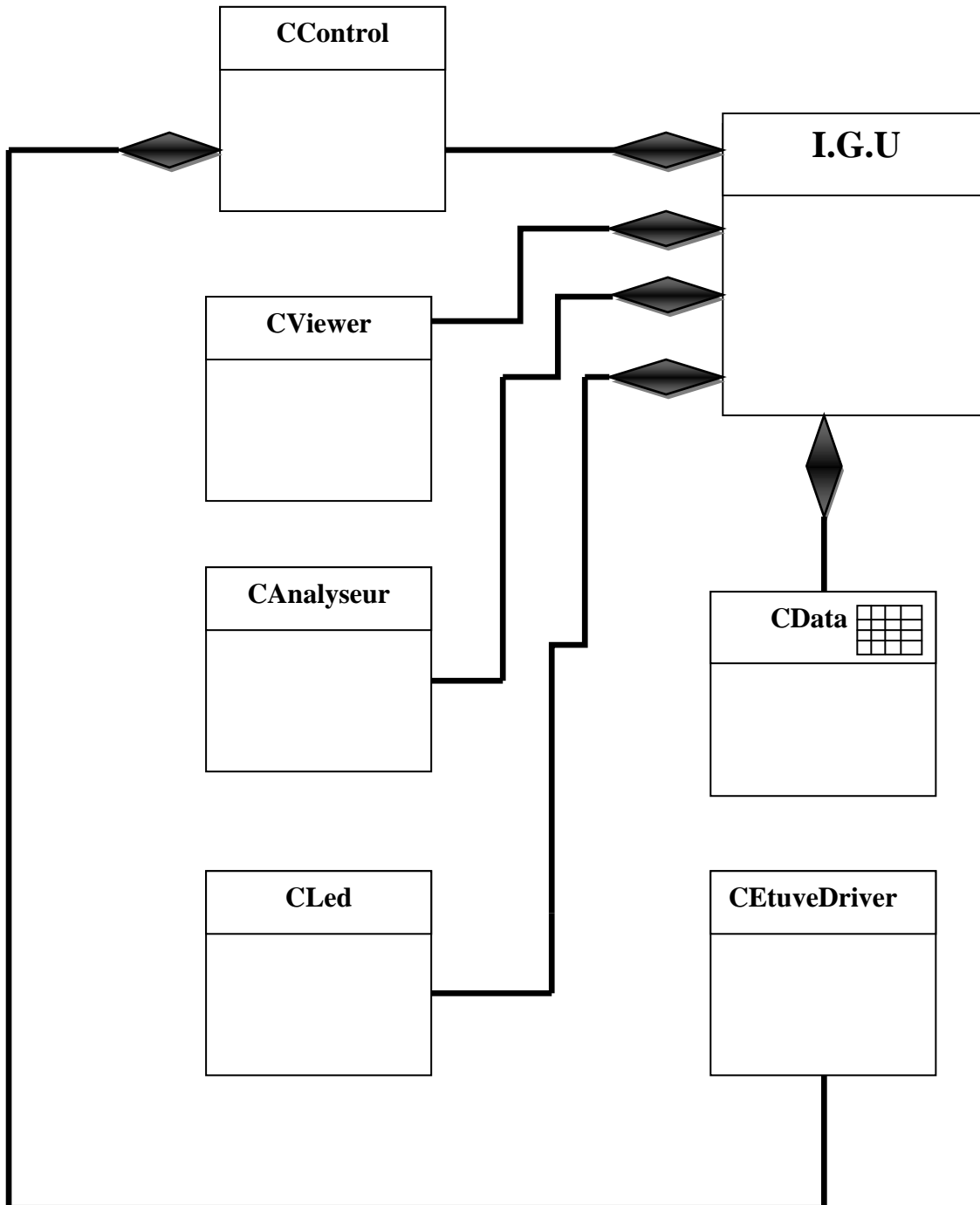


Figure 36. Diagramme des classes

5.10. PROGRAMMATION

5.10.1. CHOIX DU LANGAGE ET DE L'OUTIL DE DEVELOPPEMENT

- Langage : Le C++
- Outil de développement : Microsoft Visual Studio 2008

Les raisons de ces choix sont :

1. Le C++ est le langage que nous avons appris durant nos années de formation à l'ESPA.
2. L'Environnement de développement intégré (E.D.I) Microsoft Visual Studio 2008 est un outil très performant car il met à notre disposition toutes les bibliothèques nécessaires à la construction de l'Interface Graphique Utilisateur(I.G.U). De plus Microsoft met gratuitement à la disposition des étudiants ces produits sans licence.
3. Les autres composants essentiels au programme tels que la classe de gestion de base de données ainsi que les bibliothèques liaison dynamique (dll) pour communiquer avec le port parallèle sont écrites en C++.

5.10.2. IMPLEMENTATION DES CLASSE
LE DIAGRAMME DEFINITIF DES CLASSES

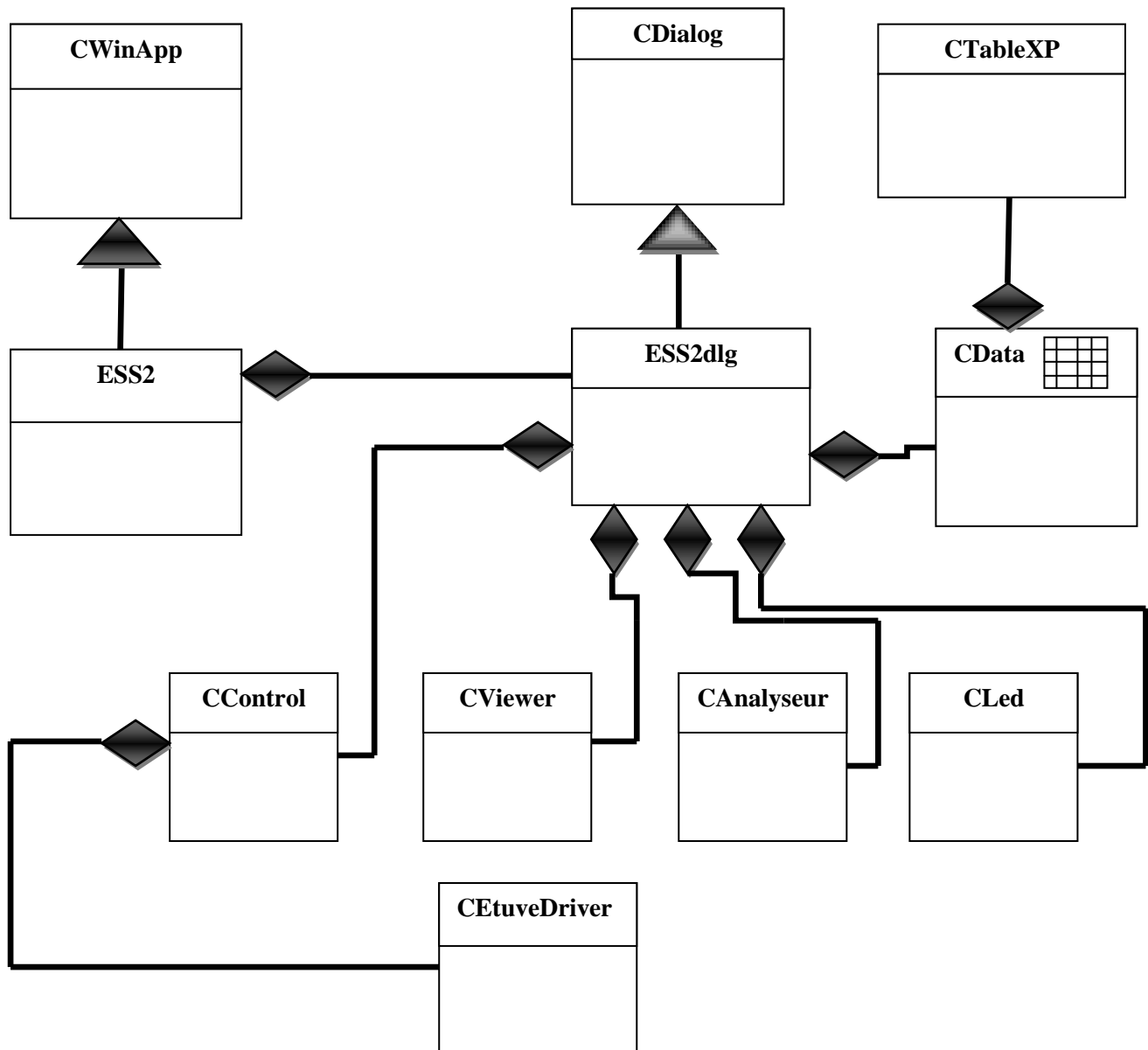


Figure 37. Diagramme définitif des classes

5.10.3. REMARQUE

1. CWinApp et CDialog sont fournies par MFC (Microsoft Foundation Class).
2. CData est fournie par la base de données qui est déjà existant à partir de FDB

5.10.4. Les fichiers headers (.h) des classes

Les headers ou déclarations sont des fichiers contenant l'interface de la classe, c'est-à-dire l'ensemble de toutes les fonctions publiques que la classe met à la disposition du programmeur utilisateur.

➤ *fichiers headers de CControl*

```
class CControl
{
private:
    CEtuveDriver driver;

    Int ton , toff , tres ,colmpte, min;
    bool mate, on , go ;
    double val;
    double delta , oldtemp,newtemp;
    bool hot;

    void Go (void);

public :
    CControl (void);
    Set Param( );
    Set Consigne ( );
    void Stop (void);
    void ModeAuto ( );
    void ModeManuel ( );
    bool Is Auto( );
    void On Fan ( );
    void OffFan ( );
    void Hot ( );
    void Cold ( );
    double Temperature ( );

};
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```

➤ **fichier header de CEtuveDriver**

```
class CEtuveDriver
{
private:
    short c0, c1, c2, port, temp;
    double a,b,c,d, moyenne,somme;
    bool valid,res,fan;
    short cont;

public:
    CEtuveDriver();
    virtual ~CEtuveDriver();
    void ToTimer();
    void Config(double xa=0,double xb=0,double xc=0.76923,double xd=63.077);
    void OnFan();
    void OffFan();
    void OnRes();
    void OffRes();
    bool Fan_IsOn();
    bool Res_IsOn();
    double GetTemp();
    double GetDataToPort();
};
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```

➤ **Le header de CViewer**

```
class CViewer
{
private:
    CWnd *support;
    int ix,iy,length,height,iepais;
    int xmin,xmax,xmoy,xsur2,scrollpos,zoom;
    int tab[700];
    COLORREF color;
    double valmin,valmax;
    double tabreal[700];
    bool miniselect,maxiselect;

    int analmin,analmax,xmaxdraw,xmindraw;
```

```
int X(int xclick);  
void Draw12();
```

```
public:
```

```
CViewer(void);  
~CViewer(void);
```

```
void ConnectDC(CWnd *dc);  
void SetZoom(int xzoom);  
void Draw();  
void Pos_Init(int x=0, int y=0,int length=500,int width=250);  
int GetXmin();  
int GetXmax();  
int GetXmoy();  
void Push(int val,int i);  
int GetScrollPos();  
int Left();//gauche  
int Right();//droite  
void ReFresh();  
void GraphSet(double xvalmin,double xvalmax);  
void SetColorFont(COLORREF xcolor);  
void SetEpais(int epais);  
void PushReal(double val,int i);
```

```
void SetXmin(int xselectmin);  
void SetXmax(int xselectmax);
```

```
double GetVal(int xselect,int yselect);
```

```
void AffSelect(int xxx);
```

```
bool Min_IsSelect(int xselect,int yselect);  
bool Max_IsSelect(int xselect,int yselect);
```

```
void Change(int xselect);
```

```
};
```

```
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```

➤ *Le header de CAnalyseur*

```
class CAnalyser
{
private:
    CViewer view;
    void Calc();
    double resmin,resmax,average;
    int analmin,analmax,ix,xmin,xmoy,xmax,xsur2;
    double tabreal[700];
    bool analyse;
    double zoom;
    int X(int click);

public:
    CAnalyser(void);
    ~CAnalyser(void);
    double GetVal_Max();
    double GetVal_Min();
    double Get_Average();
    void PushReal(double val,int i);
    void Analyse();
    void Reset();
    bool GetEtatAnalyse();
    void ReFresh();
    void ChangeZoom(int xzoom);

    void Left();//gauche
    void Right();//droite

    void Init(int xix);
};
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```

➤ **Le header de CLed**

```
class CLed
{
Private :
    CWind *support ;
    Int ix ,iy,ilong,iepais,oldval;
Public :
    CLedFan ( void);
    void DrawFont ( );
    void Connect DC( CWind*dc);
    void Draw ( int val); // val= [0;100 ]
    void Pos Init (int x=0, int y=0, int xlong=10,int xepais=1 ;

};
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```

➤ **Le header de CData**

```
class CData
{
private:
    TABLEXPDEMO tab;
    int day[700],month[700],year[700],hour[700],min[700],sec[700],i;
    double temp[700];
public:
    CData(void);
    ~CData(void);
    void Open(string nametab,string pathtab);
    void Save(string nametab,string pathtab);
    void Push();
    double GetTemperature(int i);
    int GetDay(int i);
    int GetMonth(int i);
    int GetYear(int i);
    int GetHour(int i);
    int GetMin(int i);
    int GetSec(int i);
};
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```


CHAPITRE VI .MODE D'UTILISATION DU KIT ET SIMULATION

6.1. CONFIGURATIONS EXIGÉES

6.1.1. configuration minimum de l'ordinateur

- Un processeur à 1 GHz
- RAM 512 Mo
- VISUAL STUDIO 2008
- Windows XP
- Carte graphique 64Mo
- Port parallèle

6.1.2. Système de sécurité

- Onduleur pour la coupure subite de courant
- Stabilisateur pour la protection de l'ordinateur

6.1.3. Système de démarrages

Avant d'utiliser le kit il faut suivre les étapes suivantes :

- Connecter le circuit de puissance de l'étuve avec l'électronique de commande c'est-à-dire relié KAVM avec RLY1 et KARe avec RLY2 ;
- Connecter le thermocouple avec le circuit d'acquisition de données ;
- Connecter l'électronique de commande et le circuit d'acquisition (dans une même plaquette) avec le câble du port parallèle ;
- Connecter le câble avec le port parallèle de l'ordinateur ;
- Démarrer l'étuve (tension d'alimentation 380V) ;
- Brancher l'électronique de commande et le circuit d'acquisition (tension d'alimentation 220V) ;
- Démarrer l'ordinateur (tension d'alimentation 220V)

Après ces étapes ouvrir le kit sur le bureau de l'ordinateur et l'Interface Graphique Utilisateur apparaît à l'écran comme suit :

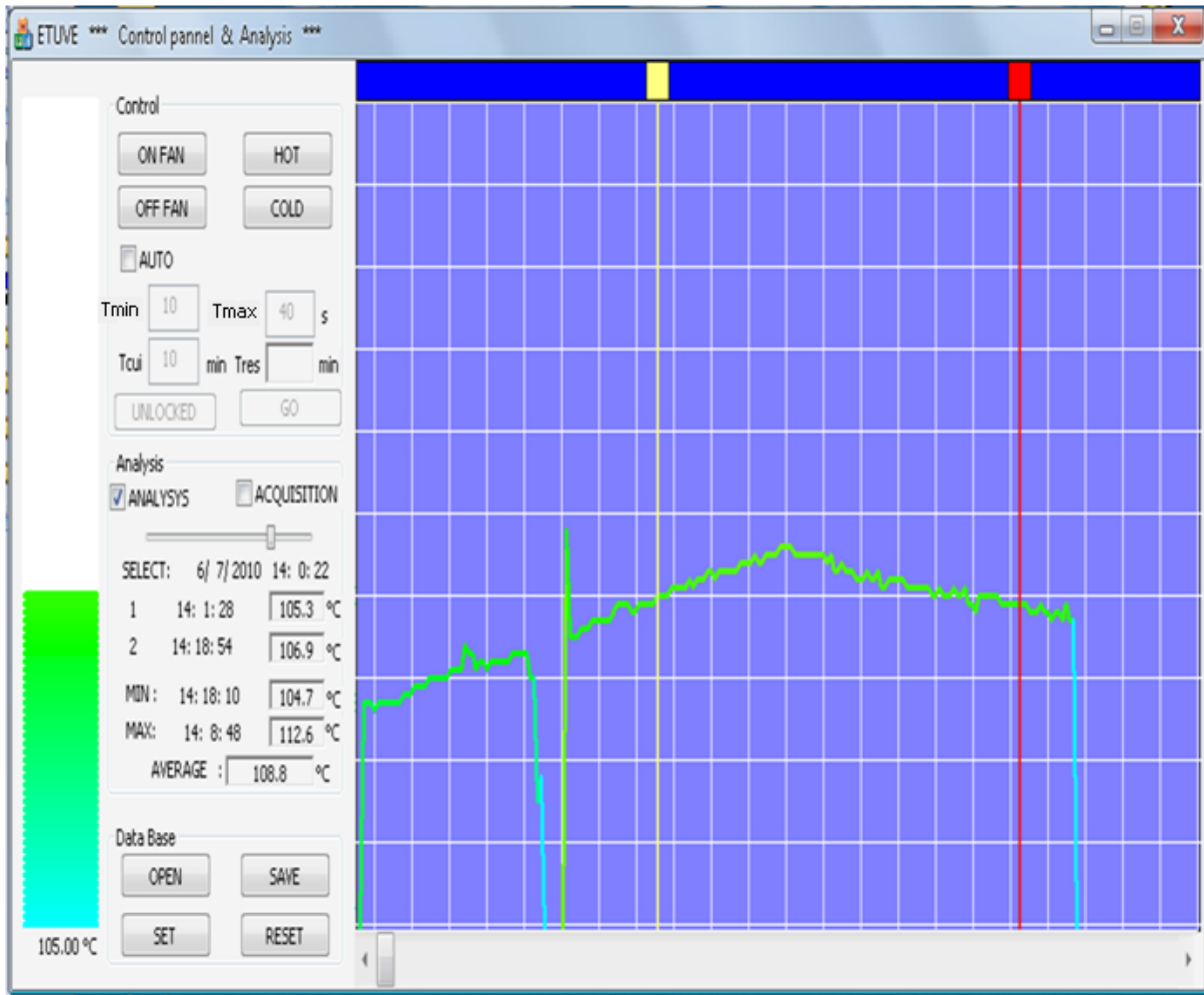


Figure 38. Résultats obtenus après essais et simulation

6.2. SIMULATION

Pour faire la simulation, on va utiliser les données obtenues le 06 Juillet 2010 enregistrées dans la base de données (DATA BASE) du logiciel.

Pour se faire il y a 4 étapes pour piloter l'étuve :

Etape 1:

- **DATA BASE (Base de données)** : Son rôle principal est de stocker les données dans un fichier et de le lire ultérieurement pour analyse.

Etape 2 :

- **CONTROL(Contrôle)** : Son but est de commander le ventilateur et la résistance de l'étuve. Le bouton ON FAN et OFF FAN c'est le bouton marche/arrêt du ventilateur, le bouton HOT et COLD c'est le bouton marche/arrêt de la résistance.

Etape 3 :

- **AUTO (Automatique)** : C'est le temps de marche et temps d'arrêt pour la résistance c'est-à-dire qu'on peut commander automatiquement le temps d'arrêt et le temps de marche de la résistance pour obtenir la température de consigne T_{min} et T_{max} dans l'étuve.

Etape 4 :

- **ANALYSIS (Analyse)** : C'est le résultat obtenu à partir du temps de fonctionnement de l'étuve. Le bouton ACQUISITION est le bouton qui affiche les valeurs de températures provenant du thermocouple. À partir de cette date à 14h 1mn 18s $T_1 = 105.3^{\circ}\text{C}$ et à 14h 18mn 54s $T_2 = 106.9^{\circ}\text{C}$. Le bouton ANALYSIS qui analyse la température minimal et maximal et directement on peut faire le moyenne(AVERAGE) entre T_1 et T_2 qui est paramétrables. À 14h 18mn 10s $T_{min} = 104.7^{\circ}\text{C}$ et à 14h 8mn 48s $T_{max} = 112.6^{\circ}\text{C}$, on a fait la moyenne entre ce deux valeur alors $T_{moy} = 108.8^{\circ}\text{C}$.

CONCLUSION GENERALE

Ce travail nous a permis d'approfondir nos connaissances sur le pilotage assisté par ordinateur d'un processus industriel tel que l'étuve.

On a utilisé le langage C++ pour envoyer les données du port parallèle.

On peut enregistrer et stocker la valeur de température suivant le temps de marche et le temps d'arrêt de l'étuve à l'aide de la base de données.

Toute fois, il faut remarquer que ce pilotage est statique. En effet les températures de consigne ne peuvent plus être modifiées que d'une façon manuelle.

A titre d'amélioration, on peut ajouter un aspect dynamique permettant d'imposer un comportement dynamique de l'étuve en précisant (consignant) les variations dans le temps des températures de consigne.

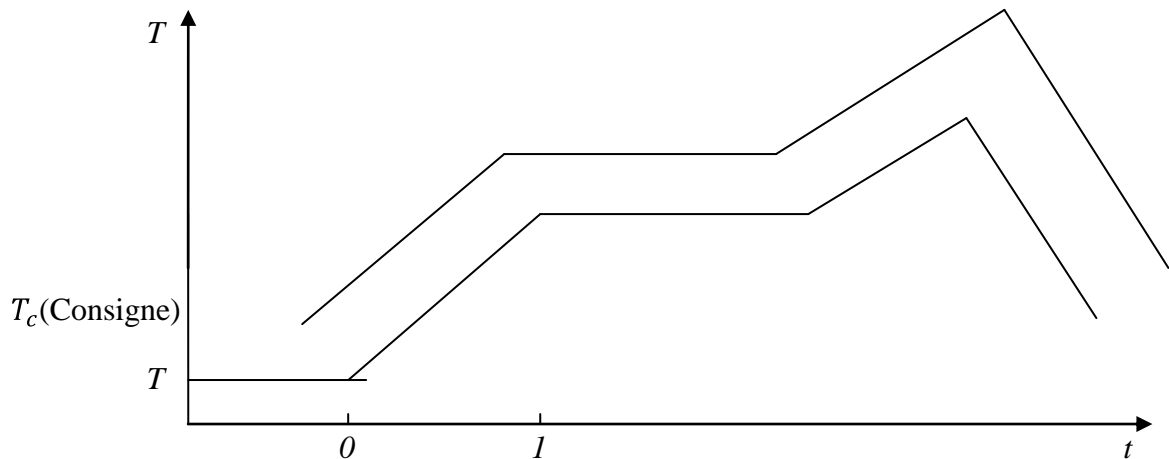


Figure 39. Exemple de comportement dynamique de l'étuve

Autre possibilité d'extension : intégrer dans le kit (c'est-à-dire dans la partie soft) un correcteur numérique.

- ✓ Enfin et non des moindres, ce kit pourrait servir de plate forme de l'étude pratique de la théorie d'asservissement et de la régulation industrielle.

BIBLIOGRAPHIE

1. *Clé pour IBM PS/2* DANIEL MARTIN et PIETTE François.
2. *The 8080A BUGBOOK Micro computer interfacing and programming*
Par LARSEN G David, Titus A, Jonathan, RONY R.
3. *Electricité, électronique de commande –puissance, électronique édition*
1997 écrit par Pierre Agati.
4. *Mesure et Essais sur Circuits Electriques et Dispositifs Electroniques*
par Pierre GAROT Professeur en Génie électrique

WEBOGRAPHIES

http://www.polytech-lille.fr / rlitwak/Cours_MuP/rlit423.html
<http://www.africa computing .org/pcours26.html?section=2>
<http://www.lvr .com/parport.htm>
<http://www.aure l32.net/elec/port parallèle .php.en>
<http://www.doc.ic.ac.uk/ih/doc/par/>
<http://www.http.download.com/>
<http://www.beyondlogic.org/spp/parallel.htm>
<http://ieeexplore.ieee.org/>
<http://www.sudoc.abes.fr>
[http://tel.archives-ouvertes.fr/ -](http://tel.archives-ouvertes.fr/)
<http://stneasy.fiz-karlsruhe.de>
<http://stneasy.fiz-karlsruhe.de/html/french/login1.html>
http://portail.scd.univ-tours.fr/screens/erm_AtoZ_ressources.html - 2008
Université de Tours.
http://portail.scd.univ-tours.fr/search/y?* - 2008 Université de Tours.
www.tricolor-industries.fr
www.lglyon.f
www.sody.fr
http://fr.wikiversity.org/wiki/Logique_s%C3%A9quentielle/Registres
<http://ni.com/legal/termsfuse/unitedstates/us/>
www.datasheetcatalog.com