

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
MINISTÈRE DE L'ENSEIGNEMENT SUPERIEUR  
ET DE LA RECHERCHE SCIENTIFIQUE



FACULTE DES HYDROCARBURES ET DE LA CHIMIE  
DEPARTEMENT D'AUTOMATISATION ET ELECTRIFICATION DES  
PROCEDES INDUSTRIELS

# MEMOIRE DE FIN D'ETUDES

En vue de l'obtention du Diplôme de Master

## THEME

Conception et réalisation d'un système de  
contrôle de température  
dans une enceinte fermée avec commande  
par une application Androïde

*Réalisé par :*  
*Ouchene Imane*

*Encadré par : Dr Meziane Nacéra*

Année universitaire : 2015/2016

## ملخص

في هذا المشروع نهتم بدراسة وتصميم نظام القياس والتحكم في درجة الحرارة. لتلبية الحاجة قمنا بدراسة موسعة و اقتصادية لمختلف الوسائل والمكونات, و أيضا تصميم نظام التحكم الذي يوفر سيطرة على مختلف وظائف الخدمة التي تم تحديدها في دفتر الشروط عبر برمجة منطق التحكم في الأداة أردوينو نانو , أخيرا نناقش مرحلة التنفيذ أو ما يعرف بالتصنيع والاختبارات و التي تهدف إلى استغلال و تحليل نتائج الاختبارات.

**كلمات مفتاحية :** أردوينو نانو- شاشة ال سي دي- مضيء صمام- صفارة- جهاز استشعار درجة الحرارة- المرحل - بلوتوث- تحكم بعرض النبضة

## Résumé

Ce projet consiste à faire l'étude, la conception ainsi que la réalisation de la partie commande d'un système de mesure et de régulation de température. Pour répondre à ce besoin, nous avons élaboré une étude bibliographique ainsi qu'une étude critique des différents instruments et composants. Ensuite, une conception du système de commande et les différents modularités à choisir et la méthode de commande utilisée, puis la programmation d'une logique dans le circuit de commande Arduino-nano qui assure les fonctions de services qui sont définies dans le cahier des charges. Enfin nous aborderons la phase dite implémentation ou bien les tests de fonctionnement, dont le but est l'exploitation des résultats d'analyse et la localisation des contraintes de fonctionnement.

**Mots clefs :** Arduino Nano - Afficheur LCD -LED-Alarme- capteur de température -Relai-Bluetooth - interface graphique

## Abstract

This project consists to study, design, and realization of the command and control part of temperature measuring and regulation system. To meet this need, we've developed a literature review and a critical study of various instruments and components, then a design of the control system and the different modularity to choose and the control method used, then programming the logical component Arduino-nano that provides service functions that are defined in the design of conditions. Finally we discussed the implementation phase or operating tests, which aims to the exploitation results of analysis and location of operating limits.

**Key words:** Arduino Nano -LCD display -LED-Buzzer- temperature sensor -Relay-Bluetooth - graphical interface

# *Remerciements*

*Je tiens à remercier le bon Dieu de nous avoir accordé toute la patience, le courage, la volonté et la motivation qui nous ont permis de mener à bien ce travail jusqu'à sa fin.*

*Je tiens à présenter ma reconnaissance à ma promotrice Mme Meziane Nacéra qui par sa confiance et ses conseils a porté une attention constante à mon travail, ainsi que pour tous ses encouragements pendant toute la durée de ce travail. Qu'elle trouve dans ce mémoire l'expression de ma profonde gratitude.*

*Je tien à remercier également les membres de jury pour avoir l'extrême gentillesse de bien vouloir évaluer ce travail.*

*Enfin, j'exprime mes remerciements à tous ceux qui ont contribué de près ou de loin à l'élaboration de ce travail.*

# Dédicace

*C'est avec profonde gratitude et sincères mots,  
que je dédie ce modeste travail de fin d'étude à mes chers parents,  
qui ont sacrifié leur vie pour ma réussite et qui ont éclairé le chemin par leurs conseils  
judicieux. J'espère qu'un jour, je puisse leurs rendre un peu de ce qu'ils ont fait pour moi,  
que dieu leur prête bonheur et longue vie.*

Je dédie aussi ce travail à mon cher frère Anouar, mes chères sœurs Kanza, Lynda ,ma sœur  
Mounia et son marie Farid ,Mes neveux Wassim et Dina ,ainsi qu'à toute ma famille.

A Mr Gherouat Ahmed Housseem Eddine

A ma chère amie Baouz Hadjer

A tous les membres de club scientifique ELECTRO de la faculté  
des Sciences de l'Ingénieur, Boumerdes.

A tous mes ami(e)s,

A tous mes professeurs qui m'ont enseigné,

Et à tous ceux qui me sont chers.

**IMANE**



# *Table des matières*

---

**Introduction générale** ..... 1

**Chapitre 1 : Choix du capteur de température**

I.1. Introduction .....2  
I.2.Définition de la température .....2  
    I.2.1.Echelle de température .....2  
    I.2.2.Mesure de température .....3  
I.3.Types de capteur de température existants.....3  
    I.3.1.Sondes RTD .....3  
    I.3.2. Thermistance .....5  
    I.3.3 .Thermocouple .....6  
    I.3.4.Thermométrie par diodes et transistors .....7  
I.5. Choix du capteur .....8  
    I.5.2. Capteur de température LM35 .....9  
    I.5.3.Avantage de LM35 .....10  
I.6. Utilisation du capteur de température .....10  
I.7.Conclusion.....11

**Chapitre 2 : Etude conceptuelle du dispositif de contrôle**

II.1. Introduction.....12  
II.2.Cahier des charges .....12  
    II.2.1Conditions d'utilisation .....12  
    II.2.2. Seuil d'action .....12  
    II.2.3.Dimensionnement .....13  
    II.2. 4 Budget .....13  
    II.2.5. Etude de faisabilité .....13  
    II.2.6. Schéma synoptique.....14  
II.3.Régulation du système .....15  
II.4. Conception du projet.....15  
    II.4.1.Capteur de température LM35 .....15  
    II.4.2.Carte Arduino Nano .....16  
    II.4.3.Module Bluetooth HC-06 .....17  
    II.4.4. Unité d'affichage .....19  
    II.4.5. Potentiomètre .....19  
    II.4.6.Alarme .....20

---

II.4.7. Diode électroluminescente .....	20
II.4.8. Circuit de puissance .....	21
II.4.9. Schéma synoptique final .....	22
II.5. Conclusion .....	23

### **Chapitre 3 : Outils de développement Software**

III.1. Introduction.....	24
III.2. Logiciel de programmation hardware ARDUINO .....	24
III.2.1. Logigramme général du programme sous Arduino.....	24
III.2.2. Interface du logiciel Arduino .....	25
III.2.3. Avantages de la carte ARDUINO .....	26
III.2.4. Développement d'un projet sous Arduino.....	26
III.3. Logiciel de programmation graphique LabView .....	27
III.3.1. Environnement de LabVIEW .....	27
III.3.2. Avantages de Labview .....	28
III.4. Logiciel de simulation Proteus .....	28
III.4.1. Présentation de l'ISIS .....	28
III.4.2. Présentation de l'ARES .....	29
III.5. Applications mobiles .....	30
III.5.1. Histoire d'Androïde .....	31
III.5.2. App Inventor pour Android .....	31
III.5.3. Instructions d'Android sous App Inventor2 .....	32
III.5.4. Avantages de l'application App Inventor2 .....	35
III.5.5. Limitations .....	35
III.6. Conclusion.....	36

### **Chapitre 4 : Programmation, réalisation et tests**

IV.1. Introduction .....	37
IV. 2. Simulation du projet avec l'outil ISIS.....	37
IV. 2. 1. Réalisation de tout le système hors service .....	37
IV. 2. 2. Programme sous Arduino .....	38
IV. 2.3. Mise en service du système dans l'intervalle [ 20°,30°].....	42
IV. 2.4 . Mise en service du système dans l'intervalle [T>32°C].....	42
IV.3. Interface graphique HMI (LABVIEW).....	43
IV.3.1. Partie enregistrement de données .....	46

---

IV.4. Conception de la carte PCB de commande (ARES) .....	46
IV.5. Commande du système à partir d'un Smartphone ANDROID via la carte Arduino Nano .....	49
IV.5.1. Code source de l'application .....	50
IV.5.2. Mise en marche de l'application .....	52
IV.6. Partie pratique .....	53
IV.6.1. Boitier .....	53
IV.6.2. Carte de commande réalisée .....	54
IV.6.3. Projet final .....	55
IV.7. Etude technico-économique .....	56
IV.8. Conclusion .....	57
<b>Conclusion générale</b> .....	<b>58</b>
<b>Références bibliographiques</b> .....	<b>59</b>
<b>Annexe</b> .....	<b>60</b>

---

# *Liste des figures*

---

## **Chapitre 1 : Choix du capteur de température**

Fig I.1 : Sonde PT100 .....	3
Fig I.2 : Thermistance .....	5
Fig I.3 : Thermocouple .....	6
Fig I.4: AD590 .....	8
Fig I.5: LM 35 .....	8
Fig. I.6 : Graphique représentatif des mesures faites entre le thermomètre et le LM35.....	9
Fig. I.7 : Capteur de température LM35.....	10

## **Chapitre 2 : Etude conceptuelle du dispositif de contrôle**

Fig II.1 : Boucle de régulation du système.....	13
Fig II.2 : Schéma synoptique.....	14
Fig II.3 : Schéma électrique de LM35DZ .....	15
Fig II.4 : Brochage de l' Arduino NANO.....	16
Fig II.5 : Figure réelle de l' Arduino NANO.....	16
Fig II.6 :Protocole Bluetooth HC 06 .....	18
Fig II.7 :Brochage et liaison bluetooth / Arduino Nano .....	18
Fig II.8. : Afficheur LCD .....	19
Fig II.09 : Potentiomètre .....	20
Fig II.10 :Buzzer.....	20
Fig II.11 : Figure représentative d'une LED .....	21
Fig II.12:Schéma électrique d'un relai ouvert et fermé.....	21
Fig II.13:Relai 5V pour la carte Arduino .....	22
Fig II.14 : Représentation du schéma synoptique final.....	22

## **Chapitre 3 : Outils de développement Software**

Fig.III.1 : Logigramme général du programme Arduino .....	25
Fig.III.2 :Interface de logiciel Arduino .....	25
Fig.III.3:Logiciel LABVIEW .....	27
Fig.III.4: Deux fenêtres de l'environnement de développement sur LabVIEW.....	28
Fig.III.5:Logiciel ISIS PROTEUS .....	29
Fig.III.6: Logiciel ARES PROTEUS .....	30
Fig.III.7 : Symbole d'Androïde.....	31

---

Fig III.8 : Symbole de MIT App Inventor.....	31
Fig III.9 :Espace pour saisir le compte Gmail.....	32
Fig III.10: Environnement d'AppInventor 2 .....	32
Fig III.11 : Ouverture d'un nouveau projet sous AppInventor2.....	33
Fig III.12 : Ecran de travail sur AppInventor2.....	33
Fig III.13 : Code QRcode.....	34
Fig III.14: Interface de dialogue.....	34
Fig III.15 :Interface de programmation .....	35

#### **Chapitre 4 : Programmation, réalisation et tests**

Fig. IV.1 : Représentation d'ensemble du système de mesure de température en simulation avec ISIS, avant sa mise en service. ....	38
Fig. IV. 2 : Représentation du fonctionnement du système à une température de 25°C.....	42
Fig. IV. 3 : Représentation du fonctionnement du système à une température de 35°C.....	43
Fig IV.4 : Représentation de la programmation de l'HMI sous LABVIEW .....	44
Fig IV.5 : Interface graphique développée.....	44
Fig IV.6 : Interface graphique du système en fonctionnement à une température de 23°C.....	45
Fig IV.7 : Interface graphique du système en fonctionnement à une température de 33°C.....	45
Fig IV.8 : Représentation du tableau Excel qui enregistre les différentes valeurs de température .....	46
Fig IV.9: Carte de commande face A.....	47
Fig IV.10 :Carte de commande face B. ....	47
Fig IV.11 :Carte de commande faces A et B.....	48
Fig IV.12: Visualisation de la carte de commande en 3D vue de dessus .....	48
Fig IV.13 : Visualisation de la carte de commande en 3D vue de profile.....	49
Fig IV.14 : Interface réalisée pour l'application sur le site App Inventor2.....	50
Fig IV.15 :Application hors connexion.....	52
Fig IV.16: Application en état de fonctionnement en mode automatique.....	52
Fig IV.17 : Application en état de fonctionnement en mode Manuel .....	53
Fig IV.18 : Boitier. ....	53
Fig IV.19 : Carte de commande réalisée .....	54
Fig IV.20 : Projet final.....	55

---

# *Liste des tableaux*

---

**Chapitre 1 : Choix du capteur de température**

Tab. I.1 : Tableau représentant les mesures de comparaison entre un thermomètre et le LM35 .....8

**Chapitre 4 : Programmation, réalisation et tests**

Tab. IV. 1 Tableau d'estimation des prix des éléments utilisés..... 56

# *Abréviations*

---

CAO : Conception Assistée par Ordinateur.

EEPROM: Electrically Erasable Programmable Read-Only Memory.

E/S : Entrée/Sortie.

HMI : Humain Machine Interface.

ISIS :Intelligent Schematic Input System.

GND :Ground .

GUI : Graphical User Interfaces.

LABVIEW :Laboratory Virtual Instrument Engineering Workbench.

LCD : Liquid Crystal Display.

LED : Light Emitter Diode.

PC : Personal Computer (ordinateur).

PCB : Printed Circuit Board.

SRAM : Static Random Access Memory.

TMR:Triple Modulaire Redondance .

USB: Universal Serial Bus.

MIT :Massachusetts Institute of Technology.

# *Introduction générale*

---

## **Introduction générale**

Dans de multiples domaines (industrie, recherche scientifique, services, loisirs ...), on a besoin de contrôler de nombreux paramètres physiques (température, force, position, vitesse, luminosité, ...).

Le capteur est l'élément indispensable à la mesure de ces grandeurs physiques. Parmi ces dernières, la température qui a une importance capitale dans les paramètres des processus industriels qu'on doit la garder sous observation afin de protéger le système et le garder en bon fonctionnement.

L'objectif primordial de cette étude est de faire la conception et la réalisation d'un système de mesure de température dans une enceinte fermée selon un cahier des charges avec des conditions qui nécessite un système de contrôle qui constitue une boucle fermée de détection, mesure puis action accomplie par un système de refroidissement, l'exécution de cette logique est faite par une carte programmable connu sous le nom d'Arduino.

De même il assure un affichage numérique sur LCD et Smartphone à travers Bluetooth. Et un autre graphique sur un ordinateur et un enregistreur de données. En utilisant un capteur moins couteux qui satisfait à nos besoins.

Pour répondre aux objectifs et aux besoins exigés dans le cahier des charges fonctionnels, nous avons bien cerné la problématique donnée et organiser notre travail en quatre (04) chapitres :

Le premier chapitre passe en revue la définition du phénomène lié à la température et les types des capteurs de température existants.

Le deuxième chapitre consiste à la présentation du cahier des charges ainsi que la proposition d'une étude conceptuelle (conception, dimensionnement,... etc) du système de contrôle et la synchronisation des tâches en s'appuyant sur les conditions demandées.

Le chapitre trois présente les outils de développement software utilisés pour la programmation et la représentation des résultats.

Le chapitre 4 est consacré à la réalisation, assemblage du projet ainsi que l'exécution de la partie commande et le passage vers la phase de test de fonctionnement et l'exploitation des résultats.

Finalement, la conclusion générale fait la synthèse de cette étude, et revient sur les résultats obtenus, avec un supplément de perspective pour le développement de ce travail pour une mesure plus fiable.

*Chapitre*

---

**1**

*Choix du capteur de  
température*

## I.1. Introduction

Il est possible de mesurer la grandeur de température de plusieurs façons différentes qui se distinguent le capteur, la technique utilisée, le coût et la précision ainsi que le temps de réponse. Les types les plus courants de capteurs sont les sondes RTD, les thermistances les thermocouples, thermométrie par diodes et transistors. Ce chapitre présente la grandeur mesurée, les différents types d'instruments et le choix de capteur approprié.

## I.2. Définition de la température [1]

La température est une grandeur physique reliée aux sensations de froid et de chaud, provenant du transfert thermique entre le corps humain et son environnement. En physique, elle se définit de plusieurs manières : comme fonction croissante du degré d'agitation thermique des particules, par l'équilibre des transferts thermiques entre plusieurs systèmes.

### I.2.1. Echelle de température [1]

La température mesurée en météorologie ne correspond pas à la température physique ressentie. Il s'agit par définition de la mesure de la température à une hauteur de 1,5 m du sol.

Le physicien anglais William Thomson, plus connu sous le nom de Lord Kelvin, introduit en 1848 une échelle absolue de température. Par cette nouvelle échelle, la notion de zéro absolu est établie comme étant la température à laquelle plus aucune chaleur ne peut être tirée du corps étudié. Il s'agit de la température la plus basse qu'il puisse exister dans l'univers et correspond à l'arrêt de l'agitation moléculaire. Par convention, le zéro absolu est égal à 0 K. L'actuelle unité internationale de température est le Kelvin. L'échelle absolue ne fait intervenir qu'un seul point de référence : la valeur particulière du point triple de l'eau 273,16 K. Le point triple de l'eau correspond aux conditions dans lesquelles coexistent les trois états (liquide, solide et gazeux). Le Kelvin est donc la fraction  $1/273,16$  de la température thermodynamique du point triple de l'eau. Afin d'obtenir une échelle centésimale et permettre d'avoir toujours le même intervalle de température pour une même valeur numérique dans les deux unités, l'unité Celsius a été définie comme égale à la température absolue diminuée de 273,15 correspondant à la température de fusion de la glace à pression atmosphérique normale.

## I.2.2.Mesure de température

Appelée thermométrie, la température est une grandeur non directement mesurable, mais repérable par la variation d'une grandeur associée selon les phénomènes reliés au changement de température (résistance, dilatation ou différence de potentiel généré entre deux points de différente température, radiation générée...).

## I.3.Types de capteur de température existants

### I.3.1.Sondes RTD (Resistance Temperature Detectors - capteurs de température à résistance) [2]

Populaires pour leur stabilité, les RTD présentent le signal le plus linéaire de tous les capteurs électroniques en matière de température. Toutefois, ils coûtent généralement plus cher que leurs équivalents à cause de leur construction plus délicate et le recours au platine. Les RTD se caractérisent aussi par un temps de réponse lent et par une faible sensibilité. Ils nécessitent une excitation en courant et peuvent mesurer des températures pouvant atteindre 850°C (Voir fig I.1).



Fig I.1 : Sonde PT100

**I.3.1.1 Principe de fonctionnement [2]**

Les RTD fonctionnent sur le principe de variations de résistance électrique des métaux purs et se caractérisent par une modification positive linéaire de la résistance en fonction de la température.

Concrètement, une fois chauffée, la résistance du métal augmente et inversement une fois refroidie, elle diminue.

Les types utilisés pour les RTD sont le nickel (Ni) et le cuivre (Cu) mais le platine (Pt) est de loin le plus courant, en raison de l'étendue de sa gamme de température (-200 °C \_850°C), de sa précision et de sa stabilité.

Faire passer le courant à travers une sonde RTD génère une tension à travers la sonde RTD. En mesurant cette tension, vous pouvez déterminer sa résistance et ainsi, sa température.

Les RTD sont habituellement classés par leur résistance nominale à 0°C. Les valeurs de résistance nominale pour les RTD à film fin en platine sont comprises entre 100 et 1000  $\Omega$ .

**I.3.1.2. Inconvénients d'utilisation [2]**

Comme chaque instrument, le capteur de température à résistance a des inconvénients qui se résument dans sa limite de performance, dont il ne fonctionne plus en capteur lorsqu'il est confronté à des conditions auxquelles il ne peut pas supporter .Nous citons :

- La non-linéarité.
- La nécessité d'une correction automatique ou manuelle due à la température de la soudure.
- La faible amplitude du signal.

### I.3.2. Thermistance [2]

En règle générale, les thermistances ont une sensibilité de mesure très élevée ( $\sim 200 \Omega/^{\circ}\text{C}$ ), ce qui les rend très sensibles aux variations de température.

Bien qu'elles présentent un taux de réponse de l'ordre de la seconde, les thermistances ne peuvent être utilisées que dans une gamme de températures ne dépassant pas  $300^{\circ}\text{C}$ .

Cette caractéristique, associée à leur résistance nominale élevée, contribue à garantir des mesures précises dans les applications à basse température (Voir fig I.2).

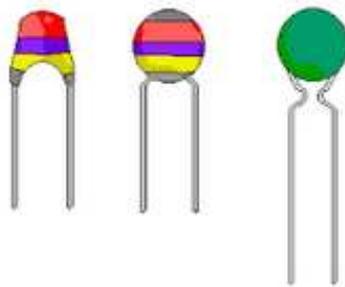


Fig I.2 : Thermistance

#### I.3.2.1 Principe de fonctionnement [2]

Les thermistances, comme les capteurs de température à résistance (RTD), sont des conducteurs thermosensibles dont la résistance varie avec la température.

Les thermistances sont constituées d'un matériau semi-conducteur d'oxyde métallique encapsulé dans une petite bille d'époxy ou de verre.

En outre, les thermistances présentent généralement des valeurs de résistance nominale plus élevées que les RTD (de  $2\ 000$  à  $10\ 000 \Omega$ ) et peuvent être utilisées pour de plus faibles courants.

Chaque capteur a une résistance nominale propre à lui qui varie de manière proportionnelle en fonction de la température selon une approximation linéaire.

Les thermistances ont soit un coefficient de température négatif (CTN), soit un coefficient de température positif (CTP).

Dans le premier cas (CTN), le plus courant, la thermistance a une résistance qui diminue lorsque la température augmente, tandis que dans le second cas (CTP), on constate une résistance accrue lorsque la température augmente.

La relation entre la résistance et la température n'est pas linéaire (exponentielle).

### I.3.2.2. Inconvénients d'utilisation [2]

Les inconvénients sont évidemment la non-linéarité, le coût très variable selon la tolérance demandée et la stabilité dans le temps (une thermistance stockée à 25°C dérive de 1% par an dans le meilleur des cas ce qui implique des réétalonnages fréquents des dispositifs qui les utilisent).

De plus leur plage de température est limitée à 200°C et il est indispensable de tenir compte de l'échauffement dû à l'effet Joule dans les résultats.

### I.3.3 Thermocouple [2]

Les thermocouples sont les capteurs les plus souvent utilisés pour la mesure de température, car ils sont tout en étant précis, et peuvent fonctionner sur une large gamme de température.

Les thermocouples présentent un taux de réponse rapide (de l'ordre de la milliseconde) (Voir fig I.3).



Fig I.3 : Thermocouple

#### I.3.3.1 Principe de fonctionnement [2]

Un conducteur génère une tension lorsqu'il est soumis à une variation de température, cette tension thermoélectrique est appelée tension Seebeck.

La mesure de cette tension nécessite l'utilisation d'un second matériau conducteur générant une tension différente pour une même variation de température (sinon la tension générée par le deuxième conducteur qui effectue la mesure annule tout simplement celle du premier conducteur).

En s'appuyant sur le principe de Seebeck, il est clair que les thermocouples ne peuvent mesurer que des différences de température entre le point de référence (soudure froide) et le point de mesure (soudure chaude).

### I.3.3.2 Inconvénients d'utilisation [2]

Les principaux inconvénients des thermocouples sont :

- La non-linéarité,
- La nécessité d'une correction automatique ou manuelle due à la température de la soudure Froide.
- La faible amplitude du signal.

### I.3.4. Thermométrie par diodes et transistors [2]

Ce type de capteur utilisant des diodes et des transistors ont l'avantage d'être simple à fabriquer et à mettre en œuvre, sont très linéaires. Ils ont une étendue de mesure limitée (-50 °C, 150 °C) .

Ces senseurs utilisent la technologie des états solides pour déterminer la température. Donc, il n'utilise pas de mercure (comme les vieux thermomètres), bilames (comme certains thermomètres ou thermomètre pour four), ni de thermistance (résistance qui varie avec la température). A la place, il utilise la caractéristique qui dit que "la tension aux bornes d'une diode augmente dans des proportions connues lorsque la température augmente". Techniquement, il s'agit de la chute de tension entre la base et l'émetteur d'un transistor. Deux exemples de capteur de température peuvent être pris comme l'AD590 ou le LM 35 (Voir fig I.4, fig I.5).



Fig I.4:AD590



Fig I.5: LM 35

### I.5. Choix du capteur

Pour faire le choix de l'instrument de mesure on a pris l'instrument de la gamme de prix faible qui est le LM35. Puis on l'a mis à l'examen par une comparaison entre le LM35 et le thermomètre classique pour avoir la différence de mesure entre les deux capteurs. On concède que la température du thermomètre est une référence.

	LM35 (°C)	Thermomètre (°C)	Différence en valeur absolu
t <sub>1</sub>	23,44	23	0.44
t <sub>2</sub>	38,09	37,3	0.79
t <sub>3</sub>	46.3	47	0.7
t <sub>4</sub>	35,64	32	3.64
t <sub>5</sub>	30,76	30	0.76
t <sub>6</sub>	29,3	29,7	0.4
t <sub>7</sub>	27,03	29	1.79
t <sub>8</sub>	24,41	27	2.59
t <sub>9</sub>	24,41	27	2.59
t <sub>10</sub>	23,93	27	3.07
t <sub>11</sub>	22,61	26	3.39
t <sub>12</sub>	22	24	2
t <sub>13</sub>	21	23,2	2.2

Tab. I.1 : Tableau représentant les mesures de comparaison entre un thermomètre et le LM35

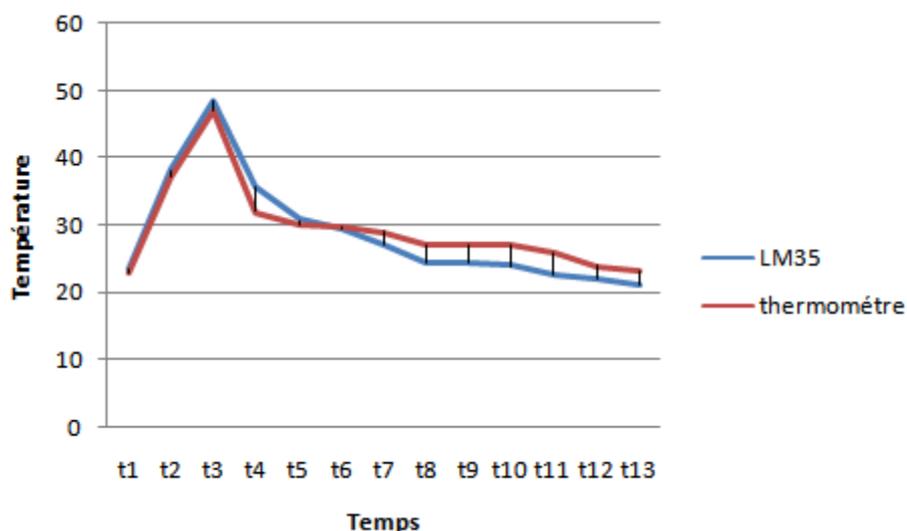


Fig. I.6 : Graphique représentatif des mesures faite entre le thermomètre et le LM35

Les valeurs représentées par les courbes respectivement en bleu et rouge pour le LM35 et thermomètre (Voir fig. I.6), ainsi les valeurs de différence entre les deux (Voir Tab. 1) nous confirment que la valeur de différence qui représente l'erreur qui a une valeur maximale de  $3,6^{\circ}\text{C}$ , cette valeur représente 3.6% sur une échelle de  $100^{\circ}\text{C}$  peut être permise pour les applications de plus haute précision où la température peut avoir un effet important, la solution est de prévoir un facteur de sécurité supérieur de la valeur de l'erreur permise pour ne jamais dépasser le seuil voulu.

### I.5.2. Capteur de température LM35 [3]

Le capteur LM35 est un capteur où la tension de sortie est linéairement proportionnelle à la température en Celsius. Ce capteur ne nécessite pas de calibrage externe pour fournir une précision sur une gamme de température de  $-55^{\circ}\text{C}$  à  $+150^{\circ}\text{C}$ . Elle augmente de 10mV pour chaque degré Centigrade et dans notre cas le capteur est alimenté en 0-5V, on ne peut mesurer par conséquent que des températures positives.

Il existe plusieurs versions de LM35, comme indiqué ci-après :

- LM35AH : boîtier métal TO46, plage  $-55^{\circ}\text{C}$  à  $+150^{\circ}\text{C}$
- LM35CH : boîtier métal TO46, plage  $-40^{\circ}\text{C}$  à  $+100^{\circ}\text{C}$
- LM35CZ : boîtier plastique TO92, plage  $-40^{\circ}\text{C}$  à  $+100^{\circ}\text{C}$
- LM35DH : boîtier métal TO46, plage  $0^{\circ}\text{C}$  à  $+100^{\circ}\text{C}$
- LM35DZ : boîtier plastique TO92, plage  $0^{\circ}\text{C}$  à  $+100^{\circ}\text{C}$



**I.7. Conclusion**

La mesure de température sert à connaître la température quelle soit utilisé juste comme indication ou pour une meilleure régulation, ainsi que son utilisation comme paramètre de sécurité pour les assemblages et mécanismes où la température ne doit pas dépasser un certain seuil.

A la fin de ce chapitre, après avoir eu une idée générale sur les différents types de capteurs de température .On a choisi le capteur de température LM35 dans ce projet vu ses avantages qui coïncident avec notre travail et surtout pour la raison économique.

Chapitre

2

*Etude conceptuelle du  
dispositif de contrôle*

## II.1. Introduction

L'étude conceptuelle se fait en référant aux conditions apparues sur un cahier des charges, pour que la réalisation soit limitée point de vue technologique, ainsi il faut prendre en charge les limites budgétaire. Donc une étude de faisabilité doit être établit avec une présentation des différents composants et organes, chacun pour accomplir une tâche selon un schéma synoptique pour avoir une régulation de température suivant un schéma de montage bien précisé.

## II.2.Cahier des charges

Le projet consiste à faire une proposition pour un système destiné à faire la mesure de température dans une enceinte considérée fermée, qui assure l'affichage de la valeur mesurée et permet la régulation automatique ou manuelle d'après des seuils bien déterminés.

### II.2.1 Conditions d'utilisation

La mesure de température doit s'effectuer à l'aide d'un capteur émettant le signal vers un système électronique (logique) non encombrant dans une boite fermée mesurant la température du climat interne d'une enceinte. Le capteur doit être monté en haut pour avoir le max de température étendue dans l'enceinte, l'affichage de la valeur mesurée est exigé sur un afficheur LCD, plus sur l'afficheur d'un Smartphone avec option de commander l'organe de régulation par l'intermédiaire d'un émetteur Bluetooth à partir d'une application androïde pour l'utilisateur qui peut être occupé à l'intérieur de cette enceinte ,ou proche du milieu de travail de ce système, ainsi que sur l'écran d'un ordinateur pour enregistrer l'historique.

Pour la régulation, le système doit être doté d'un système de contrôle de refroidissement .

L'utilisation des LED et un BUZZER pour le déclenchement des alarmes de fonctionnement.

### II.2.2. Seuil d'action

Le système marche en fonction de la température. Chaque plage de l'échelle réagit autrement que l'autre plage :

Pour des valeurs de T incluses dans l'intervalle entre 20°C et 27°C ( $20^{\circ}\text{C} < T < 27^{\circ}\text{C}$ ) la valeur est considérée bonne, donc la LED verte sera allumée.

Pour des valeurs de T incluses dans l'intervalle entre 27°C et 32°C ( $27^{\circ}\text{C} < T < 32^{\circ}\text{C}$ ) la valeur est considérée comme une valeur haute, donc la LED rouge s'allume pour indiquer « alarme haute température ».

Pour des valeurs de T supérieures à  $32^{\circ}\text{C}$  ( $T > 32^{\circ}\text{C}$ ) la valeur est considérée comme une valeur très haute, donc la LED rouge reste allumée et la logique lance le refroidissement comme procédure de protection pour indiquer « alarme très haute température », une alarme sonore doit persister jusqu'à l'élimination du danger.

### II.2.3. Dimensionnement

Le système ne doit pas être encombrant mise à part le capteur et le refroidisseur. Il doit être contenu à l'intérieur d'une boîte qui ne fait apparaître que l'afficheur LCD et les LED d'indication, un trou doit exister pour permettre au son du BUZZER pour de diffuser. Donc il ne doit pas dépasser les limites (Longueur\*Largeur\*épaisseur) = (200mm ,100mm, 150mm).

### II.2. 4 Budget

La valeur de toute l'installation de commande ne doit pas dépasser le montant budgétaire attribué à cette fonction citée , donc il ne doit pas dépasser la valeur de 6000.00 DA.

Il est nécessaire de préciser que les outils de travail tel que le PC, le Smartphone ainsi que l'outil de refroidissement ne sont pas inclus dans les charges du projet.

### II.2.5. Etude de faisabilité

La fonction principale de ce projet étant définie par la fonction de détection, mesure, raisonnement, comparaison logique et finalement l'exécution de la commande tout dépend les conditions. Sachant que la mesure régulière sert comme signal de retour qui rend le fonctionnement exécutable en boucle fermée répétitive (Voir fig II.1)

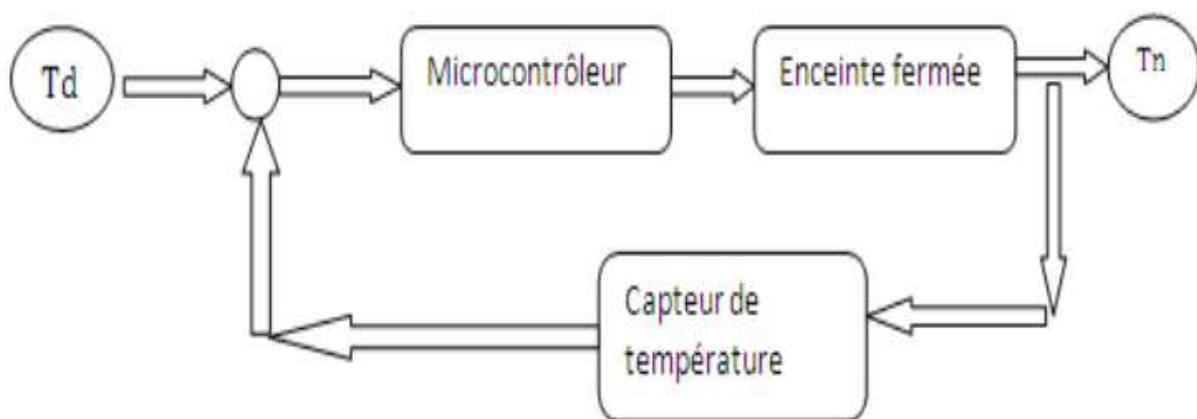


Fig II.1 : Boucle de régulation du système

### II.2.6. Schéma synoptique

Après avoir précisé que le projet est faisable, l'application de la solution technologique s'effectue selon un schéma synoptique englobant les différents modules de base de notre dispositif (Voir fig II.2).

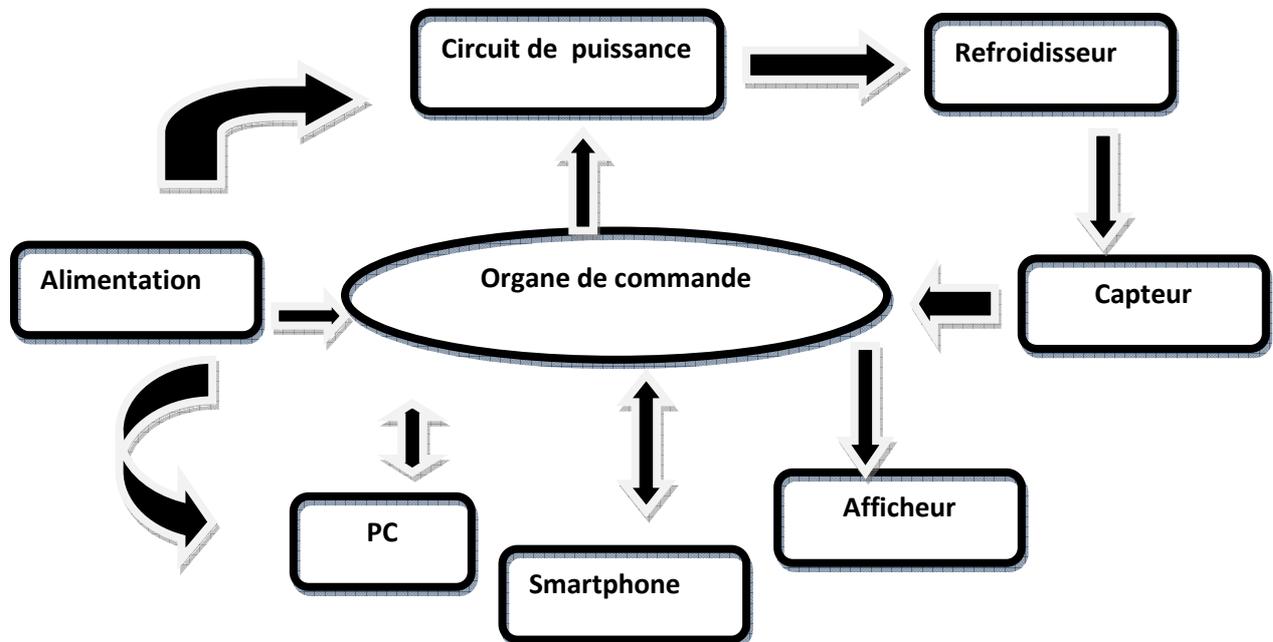


Fig II.2 : Schéma synoptique

Le schéma synoptique est constitué de :

- 1- Un bloc d'alimentations pour assurer l'alimentation d'un ordinateur, des différents composants de la partie commande (organe de commande, afficheur, etc.) et la partie puissance qui assure l'énergie pour la régulation.
- 2- Un circuit de puissance pour assurer la connexion entre l'unité de commande et le moteur.
- 3- Un refroidisseur assurant la partie régulation.
- 4- Un capteur de température.
- 5- Un afficheur pour la présentation de données nécessaires.
- 6- Un ordinateur pour lancer le cycle de fonctionnement du capteur via une interface graphique. De plus on utilisera un Smartphone pour l'affichage et la commande manuelle de l'organe de régulation.
- 7- Une unité programmable (l'organe de commande) assurant la synchronisation des différentes fonctions de service (acquisition des données, envoi des données vers l'afficheur, etc.)

### II.3.Régulation du système

Dans cette étape, notre but est de mesurer la température d'une enceinte, en se référant aux exigences du cahier des charges. L'échantillonnage se fait chaque seconde, le cycle logique fonctionne avec la comparaison de la valeur analogique mesurée avec les seuils proposés et réagit selon l'intervalle dans laquelle  $T_{mesurée}$  est incluse soit avec l'allumage de la lampe verte ou rouge ou le refroidissement, en utilisant un montage simple qui permet d'exécuter cette régulation.

### II.4. Conception du projet

Le capteur qu'on a choisi comme prévu dans le chapitre précédent est le composant LM35. L'exécution de la fonction logique de comparaison et le lancement de la commande se fait par une carte programmable de dimensions petite nommée Arduino type Nano qui possède son propre software. Le lancement du refroidissement se fait par l'intermédiaire d'un relai auxiliaire qui commande l'alimentation du refroidissement, l'affichage par un afficheur LCD et la transmission par un auxiliaire d'Arduino Bluetooth, avec d'autres composants LED, et BUZZER.

#### II.4.1.Capteur de température LM35

Dans notre projet nous avons utilisé le LM35DZ qui est disponible dans le marché

(Voir fig II.3).

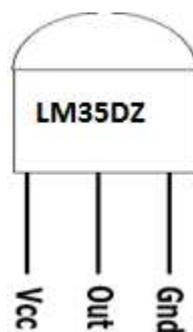


Fig II.3 : Schéma électrique du LM35DZ

##### II.4.1.1.Brochage de la puce [3]

+Vcc= alimentation positive, à relier au +5 V du montage

Vout = tension de sortie proportionnelle à la température.

GND = masse.

### II.4.1.2. Caractéristiques du LM35 [3]

- Taille: boîtier TO-92 à 3 broches (similaire à un transistor)
- Gamme de température: 0°C a 150°C / -40°F a 302°F
- Tension d'alimentation: 2.7V a 5.5V
- Courant de charge: 0.05 mA

### II.4.2. Carte Arduino Nano [4]

Arduino Nano est essentiellement un Arduino Uno conçu de taille petite pour une utilisation directe sur « breadboard ». Il embarque son propre contrôleur USB permettant une communication série très facile.

La disposition de ses pins correspond à celui d'un Arduino Pro Micro avec l'USB à l'opposé et quelques pins en plus, intelligemment placées de ce coté (Voir fig II.4, fig II.5).

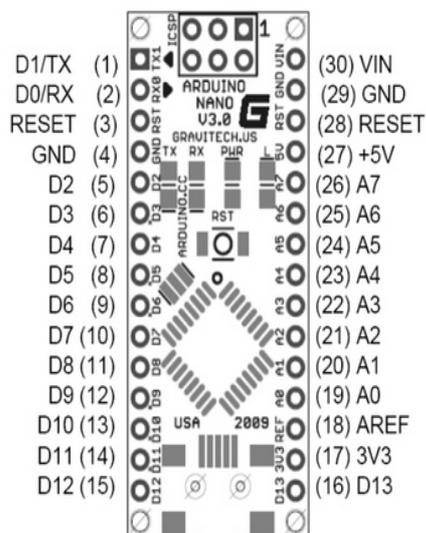


Fig II.4 : Brochage de l'Arduino NANO



Fig II.5 : Figure réelle de l'Arduino NANO

**II.4.2.1. Spécifications techniques de la carte Arduino Nano 3.0 [4]**

Les spécifications techniques de cette carte Arduino Nano sont les suivantes:

- Microcontrôleur Atmel ATmega328.
- Voltage opérationnel (au niveau logique) : 5 V.
- Voltage d'entrée recommandé : de 7 à 12 V.
- Limite de la tension d'entrée : de 6 à 20 V.
- Pins E/S digitales : 14 .
- Pins d'entrée analogique : 8.
- Courant direct par pin d'entrée/sortie : 40 mA.
- Mémoire Flash : 32 KO (ATmega328) .
- SRAM : 2 KO (ATmega328).
- EEPROM : 1 KO (ATmega328).
- Vitesse d'horloge : 16 MHz.
- Dimensions : 45 x 18 mm.

**II.4.2.2. Alimentation de la carte Arduino Nano [4]**

La carte Arduino nano est alimentée par la prise mini-USB ou bien par une source d'énergie externe non régulée, à 6-20V (pin 30), ou bien encore par une source d'énergie externe non régulée à 5V (pin 27).

**II.4.2.3. Programmation de la carte Arduino Nano [4]**

La carte Arduino Nano est programmée à l'aide du logiciel de programmation gratuit fourni par Arduino (Télécharger). Sélectionnez Nano w/ATmega168" ou "Arduino Duemilanove ou Nano w/ ATmega328" comme cartes cibles dans le logiciel.

Le microcontrôleur ATmega328 de la carte Arduino Nano possède un bootloader qui permet le téléchargement de nouveau code sans l'aide d'un programmeur matériel supplémentaire.

**II.4.3. Module Bluetooth HC-06 [4]**

Le module Bluetooth HC-06 ajoute une fonction de communication par Bluetooth à la carte Arduino. Il peut ainsi communiquer sans fil à moyenne distance de 30m avec n'importe quel autre dispositif Bluetooth (ordinateur, téléphone mobile, second module Bluetooth sur une autre carte ).

### II.4.3.1. Caractéristiques du protocole Bluetooth HC06 [4]

- Alimentation de 3.3 à 5V DC.
- Le fonctionnement de Bluetooth est sur la bande 2.4 GHz.
- LED indicatrice : statut de connexion.

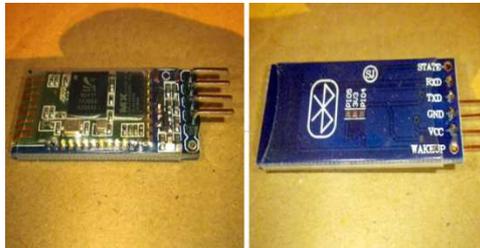


Fig II.6 :Protocole Bluetooth HC 06

### II.4.3.2. Montage du module HC06 avec la carte Arduino [4]

le module HC06 communique en mode série avec la carte Arduino, Pour cela il faut utiliser la librairie Softserial .

Le branchement est simple et est comme suit:

VCC → +5V

GND → GND

TXD → Port digital 10

RXD → Port digital 11

### II.3.3.3. Liaison Bluetooth / Arduino

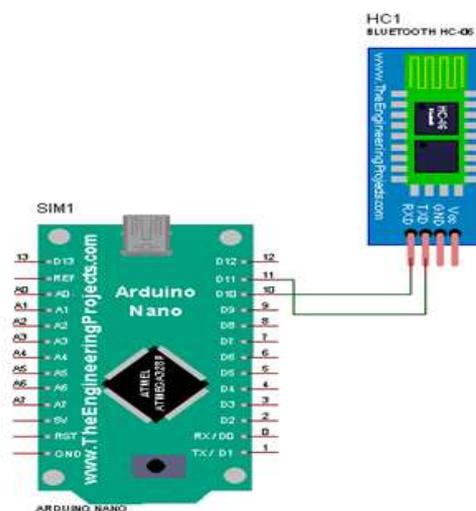


Fig II.7 : Brochage et liaison bluetooth /Arduino Nano

Quand nous avons fait la connexion de l'Arduino avec le Bluetooth HC06 et se connecter via un câble USB à un ordinateur pour la programmation, une LED sur Bluetooth clignote en permanence, cela signifie que le Bluetooth est en cours d'exécution, et elle se stabilise quand il est jumelé à un autre dispositif Bluetooth.

#### II.4.4. Unité d'affichage [5]

Tout projet qui nécessite un contrôle par l'utilisateur doit comporter un afficheur. En effet, celui-ci permet de manière très rapide de révéler n'importe quelle information qui pourrait être utile au programmeur ou à l'usagé.

Les afficheurs alphanumériques appelés LCD (Liquid Crystal Display), sont des modules compacts intelligents et nécessitent peu de composants externes pour un bon fonctionnement. Ils consomment relativement peu d'énergie (de 1 à 5 mA), sont relativement disponible dans le marché et s'utilisent avec beaucoup de facilité.

Donc, le LCD est le composant idéal pour notre application, le nombre de caractères étant limité, mais suffisant, il se contrôle aisément par un microcontrôleur.

Le modèle utilisé dans notre travail comporte deux (02) lignes de seize (16) caractères.

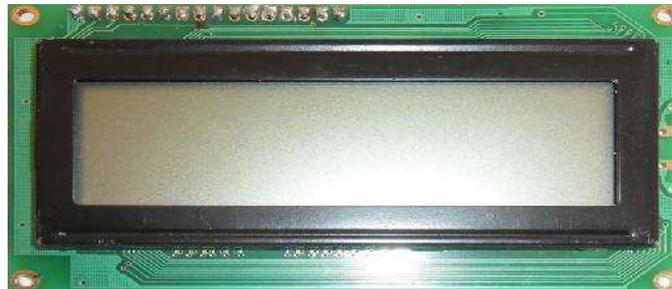


Fig II.8. : Afficheur LCD

#### II.4.5. Potentiomètre [6]

C'est un type de résistance variable à trois bornes, dont une borne est reliée à un curseur, se déplaçant sur une piste résistante terminée par les deux autres bornes. Ce système permet de recueillir, entre la borne reliée au curseur et une des deux autres bornes, une tension qui dépend de la position du curseur et de la tension à laquelle est soumise la résistance.



Fig II.09 : Potentiomètre

#### II.4.6.Alarme [7]

Appelée généralement *BUZZER*, c'est un élément électromécanique ou piézoélectrique qui produit un son caractéristique quand on lui applique une tension .

Profitant des sorties de la carte, nous avons branché un simple *BUZZER* électrique sur l'une des sorties de la carte pour faire office de Klaxon.



Fig II.10 :Buzzer

#### II.4.7.Diode électroluminescente ( *Light-Emitting Diode, LED*) [8]

C'est un dispositif capable d'émettre de la lumière lorsqu'il est parcouru par un courant électrique. Une diode électroluminescente ne laisse passer le courant électrique que dans un seul sens (le sens passant, comme une diode classique, l'inverse étant le sens bloquant) et produit un rayonnement à partir de la conversion d'énergie électrique lorsqu'un courant la traverse.



Fig II.11 : Figure représentative d'une LED

#### II.4.8. Circuit de puissance

Notre unité de commande est alimentée de 5V .Alors que la plage de fonctionnement de notre refroidisseur est du rang de 220V. Donc, il est nécessaire de prévoir un circuit de puissance pris du réseau électrique 220 V alternative de fréquence de 50Hz, pour assurer l'alimentation du refroidisseur hors le circuit de commande. Le circuit de puissance sera commandé par un relai fonctionnant sous les ordres de la logique d'organe de commande.

##### II.4.8.1. Relai [9]

C'est un organe d'exécution de commande électrique permettant de dissocier la partie puissance de la partie commande : il permet l'ouverture/fermeture d'un circuit électrique par un second circuit complètement isolé (isolation galvanique) (le circuit haute-tension n'est jamais en contact avec le circuit de commande).

##### II.4.8.1.1 Utilisation du relai [9]

Les relais sont utilisés très régulièrement, par exemple pour démarrer un moteur, commander une prise, une lumière, le chauffage, par l'intermédiaire d'une tension basse d'un circuit de commande. Dans notre cas, le relai va commander un organe de régulation de température.

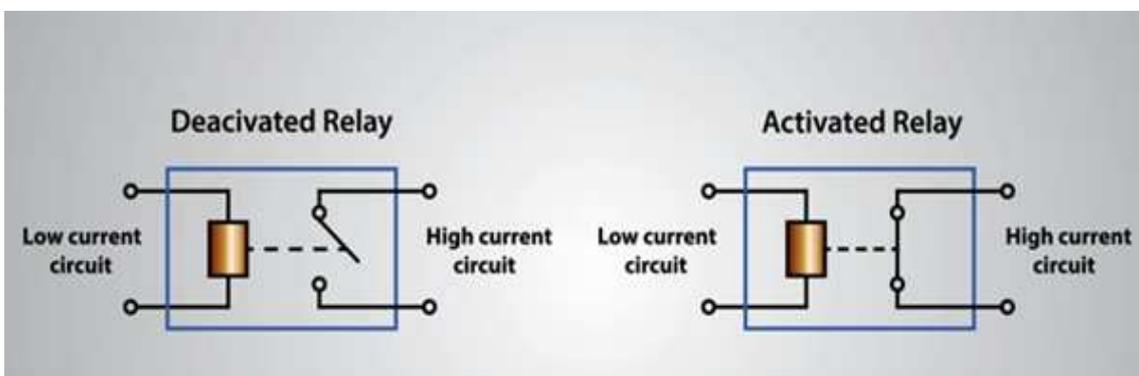


Fig II.12:Schéma électrique d'un relai ouvert et fermé

### II.4.8.1.2 Raccordement [10]

Le module relai dispose d'un connecteur de 3 broches.

- **VCC** : se branche sur une source d'alimentation 5V
- **GND** : se branche sur la masse/GND
- **DAT** : Broche de commande, à raccorder sur une sortie digitale de l'Arduino NANO.



Fig II.13:Relai 5V pour la carte Arduino

### II.4.9. Schéma synoptique final

Après l'étude conceptuelle de dispositif de contrôle on a choisit l'Arduino NANO comme organe de commande et le LM35 comme capteur de température.

La figure (Voir fig II.15) présente le schéma synoptique final, pour l'accomplissement du projet selon les obligations déterminées dans le cahier des charges.

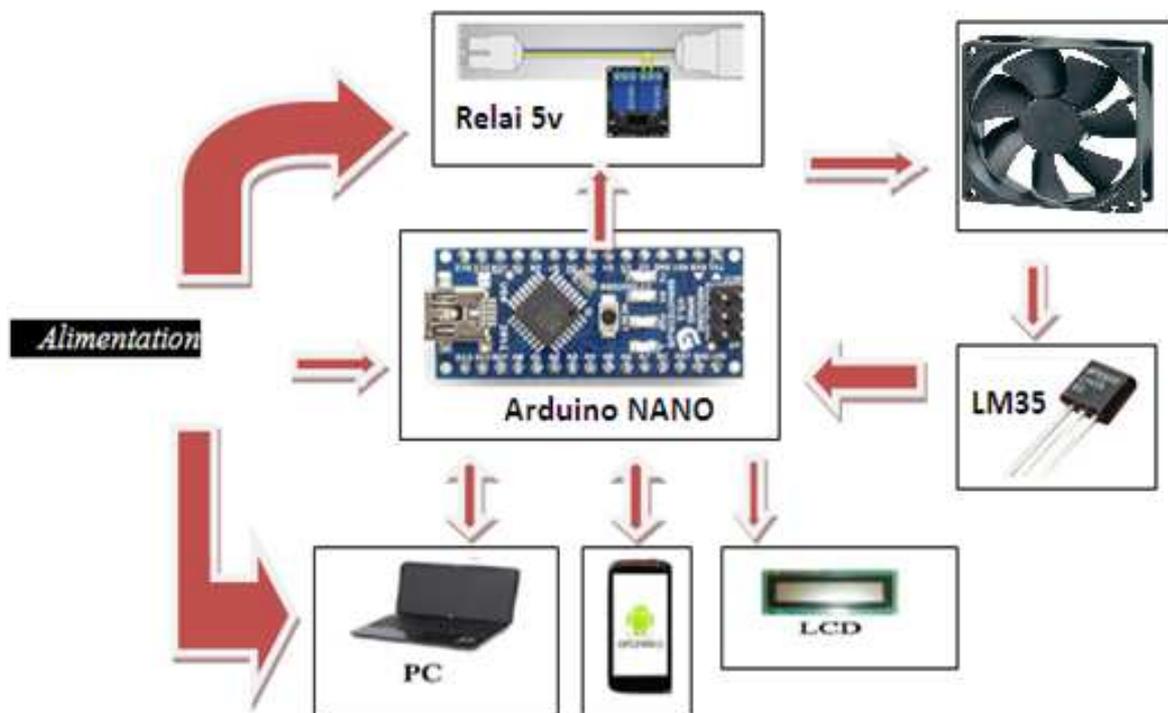


Fig II.14 : Représentation du schéma synoptique final

## **II.5.Conclusion**

La conception d'un projet peut être faite de plusieurs manières pour répondre à un même cahier des charges. Le choix de la solution adéquate est primordial et se fait en fonction de plusieurs critères.

Pour concevoir le système de régulation de température fonctionnel, il a fallu faire une étude de faisabilité dans laquelle on a déterminé le choix de tous les composants nécessaires pour le bon fonctionnement. Ainsi que la méthode de leur rassemblement afin de pouvoir assurer les fonctions de l'ensemble à moindre coût.

Dans le chapitre 03 nous allons présenter une description des différents logiciels utilisés pour le bon fonctionnement du système.

*Chapitre*

---

**3**

*Outils de développement  
software*

### **III.1. Introduction**

Le capteur LM35 qu'on a utilisé pour la mesure de la température, doit être pris en considération dans la programmation. Pour cela, on a besoin d'utiliser des instructions qui sont essentiels pour le bon fonctionnement du capteur.

Dans ce chapitre, nous allons présenter une description des différents softwares utilisés voir le langage de programmation Arduino, les logiciels de simulation ISIS et ARES , le logiciel de programmation graphique LABVIEW et le site de création des application Androïde « MIT APP INVENTOR2» .

### **III.2. Logiciel de programmation hardware ARDUINO [11]**

L'Arduino est une famille de cartes électroniques à micro-contrôleur open-source née en Italie en 2005. Ces cartes basées sur une interface d'entrée/sortie simple et sur un environnement de développement proche du langage C. Arduino peut être utilisé pour construire des objets interactifs indépendants (prototypage rapide), ou bien peut être connecté à un ordinateur pour communiquer et superviser en utilisant des logiciels de programmation graphique (Labview). Un module Arduino est généralement construit autour d'un microcontrôleur Atmel AVR (ATmega328 ou ATmega2560 pour les versions récentes, ATmega168 ou ATmega1280 pour les plus anciennes), et de composants complémentaires qui facilitent la programmation et l'interfaçage avec d'autres circuits.

#### **III.2.1. Organigramme général du programme sous Arduino**

Pour éditer un programme, on commence d'abord par faire un algorithme ou un organigramme qui montre la succession ou la chronologie des instructions et des tâches à réaliser ou à effectuer, la figure (Fig III.1) présente celui relative au projet en cours d'étude.

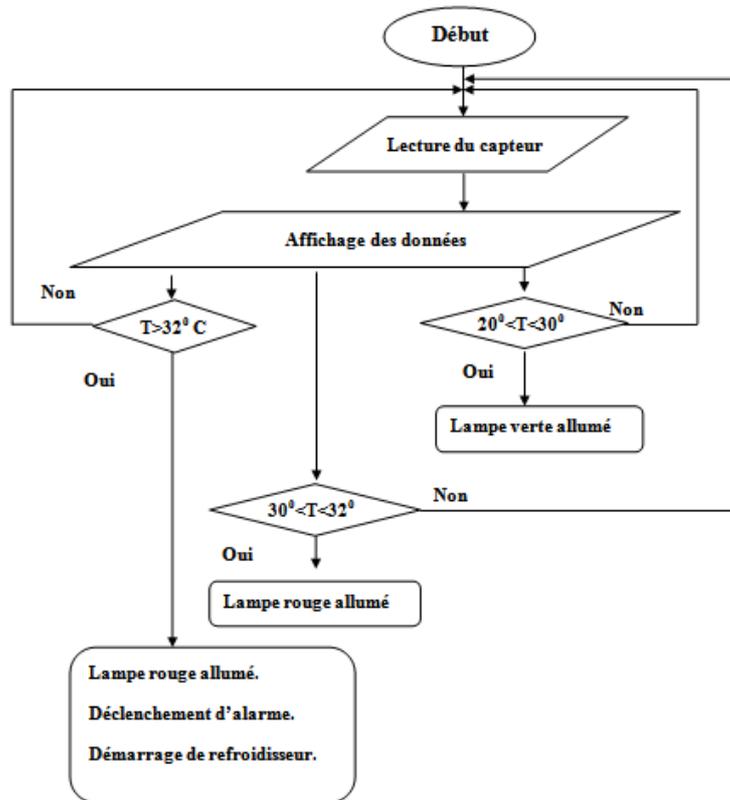


Fig III.1 : Organigramme général du programme Arduino

### III.2.2. Interface du logiciel Arduino [12]

Le logiciel Arduino comporte 04 zones montrées sur la figure (Fig.III.2) :

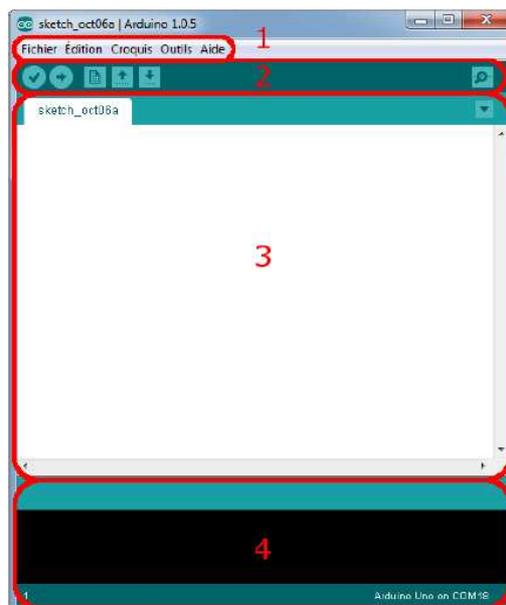


Fig.III.2 : Interface du logiciel Arduino

Zone numéro 1 : ce sont les options de configuration du logiciel.

Zone numéro 2 : il contient les boutons qui vont nous servir lorsque l'on va programmer la carte Arduino.

Zone numéro 3 : ce bloc va contenir le programme que nous allons créer.

Zone numéro 4 : celui-ci est important, car il va nous aider à corriger les fautes dans notre programme.

### **III.2.3. Avantages de la carte ARDUINO [13]**

- Il n'est pas coûteux.
- Environnement de programmation clair et simple.
- Multiplateforme : tourne sous Windows, Macintosh et Linux.
- Nombreuses bibliothèques disponibles avec diverses fonctions implémentées.
- Logiciel et matériel open source et extensible.
- Nombreux conseils, tutoriaux et exemples en ligne (forums, site personnels etc...)
- Existence de « shield » (boucliers en français) : ce sont des cartes supplémentaires qui se connectent sur le module Arduino pour augmenter les possibilités comme par exemple : afficheur graphique, interface Ethernet, GPS, Bluetooth ...etc

Arduino est utilisé dans plusieurs applications telles que : l'électronique industrielle et embarquée.

### **III.2.4. Développement d'un projet sous Arduino[12]**

Le développement d'un projet sur Arduino est très simple :

- Le code est un langage basé sur les langages C/C++, avec des fonctions et des bibliothèques spécifiques à Arduino (gestions des E/S).
- Le logiciel de programmation des modules Arduino est une application Java multiplateformes (fonctionnant sur tout système d'exploitation), servant l'éditeur de code, le compilateur peut transférer le Firmware (programme) du PC vers la carte Arduino à travers une liaison série (RS232, Bluetooth ou USB).

### III.3. Logiciel de programmation graphique LabVIEW[14]

LABVIEW (Laboratory Virtual Instrument Engineering Workbench) est un logiciel de développement d'applications de la société américaine National Instruments basé sur un langage de programmation graphique appelé langage G. Les domaines d'application traditionnels de LabVIEW sont la commande et la mesure à partir d'un PC (acquisition de données, contrôle-commande d'instruments de mesure, de dispositifs expérimentaux, de bancs de test) (Voir fig III.3).



Fig.III.3:Logiciel LABVIEW

#### III.3.1.Environment de LabVIEW

LabVIEW est centré autour du principe d'instrument virtuel (Virtual Instrument ). IL se décompose en deux parties (Voir fig III.4) :

- La première partie (partie cachée ou interne): elle contient l'algorithme du programme décrit sous la forme d'un diagramme flot de données en langage graphique.
- La seconde partie (partie visible) est constituée de l'interface utilisateur.

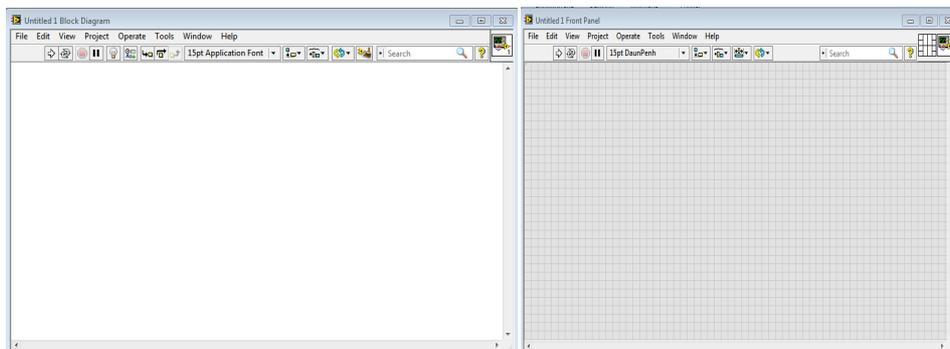


Fig.III.4: Deux fenêtres de l'environnement de développement sur LabVIEW

### **III.3.2. Avantages de Labview [14]**

Le logiciel LabVIEW contribue à rendre la complexité de la programmation bas niveau et de la configuration matérielle embarquée invisible aux yeux de l'utilisateur, ce qui permet à l'expert métier de se concentrer sur la conception (algorithmes, modèles, routines de traitement du signal, etc.) au lieu de se pencher sur des problèmes logiciels et matériels bas niveau qui ne sont pas le principal objet de ses travaux de recherche. Les experts métier y parviennent avec un niveau élevé d'abstraction en utilisant différents modèles de calculs (flux de données, diagrammes d'états, scripts, etc.).

### **III.4. Logiciel de simulation Proteus [15]**

Proteus est un logiciel destiné à l'électronique. Développé par la société Labcenter Electronics. De nombreuses entreprises et organismes de formation utilisent cette logicielle. Les logiciels inclus dans Proteus permettent la CAO (Conception Assistée par l'Ordinateur) dans le domaine électronique.

La popularité de l'outil, Proteus possède d'autres avantages :

- Pack des logiciels facile et rapide à comprendre comme ISIS et ARES.
- L'outil de création de prototype virtuel permet de réduire les coûts matériel et logiciel lors de la conception d'un projet.
- Grâce à cette suite logicielle nous pouvons mieux visualiser le bon déroulement du système ainsi que d'avoir une idée claire sur la partie matériel que ce soit références ou même conception des circuits imprimés.

Deux logiciels principaux composent cette suite logicielle nous intéressent : ISIS et ARES.

#### **III.4.1. Présentation de l'ISIS[16]**

ISIS (Intelligent Schematic Input System) est un simulateur de cartes électroniques intégrées dans le logiciel proteus. L'utilisation de ce logiciel est simple grâce à l'interface graphique qui rend claires toutes les commandes (Voir fig III.5).

L'intérêt de simuler un programme est bien pour faire visualiser le fonctionnement de l'Arduino avec tous les périphériques de la carte de commande. De plus, nous allons pouvoir vérifier le fonctionnement du programme avant de le transférer vers la mémoire du microcontrôleur.

Cela nous permettra de déceler certaines erreurs dès l'étape de conception, d'économiser du temps et également d'augmenter la durée de vie de notre Arduino.



Fig.III.5:Logiciel ISIS PROTEUS

#### **III.4.2. Présentation de l'ARES [17]**

Le logiciel ARES est un outil d'édition et de routage qui complète parfaitement ISIS. Un schéma électrique réalisé sur ISIS peut alors être importé facilement sur ARES pour réaliser le PCB de la carte électronique. Bien que l'édition d'un circuit imprimé soit plus efficace lorsqu'elle est réalisée manuellement. Ce logiciel permet de placer automatiquement les composants et de réaliser le routage de façon automatique (Voir fig III.6).



Fig.III.6: Logiciel ARES PROTEUS

### III.5.Applications mobiles [18]

Les applications mobiles sont apparues dans les années 1990, elles sont liées aux :

- Développements Internet.
- Télécommunications.
- Réseaux sans fils.
- Apparition des terminaux mobiles : Smartphones, tablettes tactiles...

Les applications mobiles sont développées sur des ordinateurs. Le langage utilisé dépend du système sous lequel l'application sera exécutée. Les applications pour les terminaux Apple sont développées dans un langage principalement dédié à ces applications mobiles, pour Windows Mobile, sont développées en C#.

Pour publier une application mobile sur les principaux App Store, le logiciel doit remplir plusieurs conditions. La combinaison de techniques qui offrent une visibilité aux applications dans les magasins est appelé App Store Optimization (ASO).

### III.5.1. Histoire d'Androïde [19]

Android, prononcé *androïde*, est un système d'exploitation mobile basé sur le noyau Linux, développé actuellement par Google. Le système a été conçu pour les Smartphones et tablettes tactiles, puis s'est diversifié dans les objets connectés comme les télévisions (Android TV), les voitures (Android Auto), les ordinateurs (Android-x86) et les smartwatch (Android Wear). En 2015, Android est le système d'exploitation le plus utilisé dans le monde avec plus de 80 % de parts de marché des Smartphones.



Fig.III.7 : Symbole d'Androïde

### III.5.2. App Inventor pour Android [20]

C'est une application développée par Google. Elle est actuellement entretenue par MIT (Massachusetts Institute of Technology). Elle simplifie le développement des applications sous Android et le rend accessible même pour les novices, et ceux qui ne sont plus familiers avec les langages de programmation. Elle est basée sur une interface graphique similaire à Scratch grâce à son interface entièrement graphique et l'absence totale de la ligne de code.

Google a publié l'application le 15 décembre 2010 et a mis fin à son activité le 31 décembre 2011. Depuis, c'est le centre d'études mobiles en MIT qui gère le support technique de cette application sous le nouveau nom "MIT App Inventor".



Fig III.8 : Symbole de MIT App Inventor

### III.5.3. Instructions d'Android sous App Inventor2

A partir de navigateur, vous créez des projets par le biais d'une interface en deux parties :

- Une page de conception d'écran (boutons, labels, etc ...).
  - Une page de programmation par blocs de type Scratch.
- On tape dans la zone d'adresse : URL <http://ai2.appinventor.mit.edu/>
  - Le système demande une connexion à un compte google.
  - On saisit le login et le mot de passe personnel d'un compte Gmail (Fig III.9)

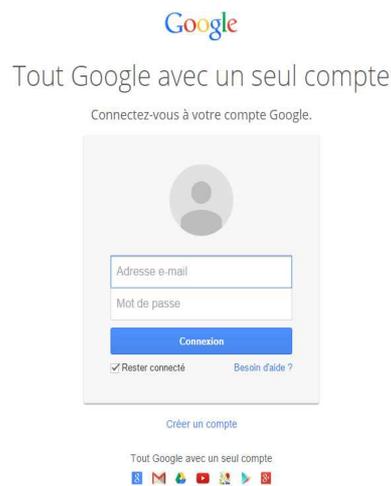


Fig III.9 : Espace pour saisir le compte Gmail

- On obtient la fenêtre de l'environnement « AppInventor 2 »

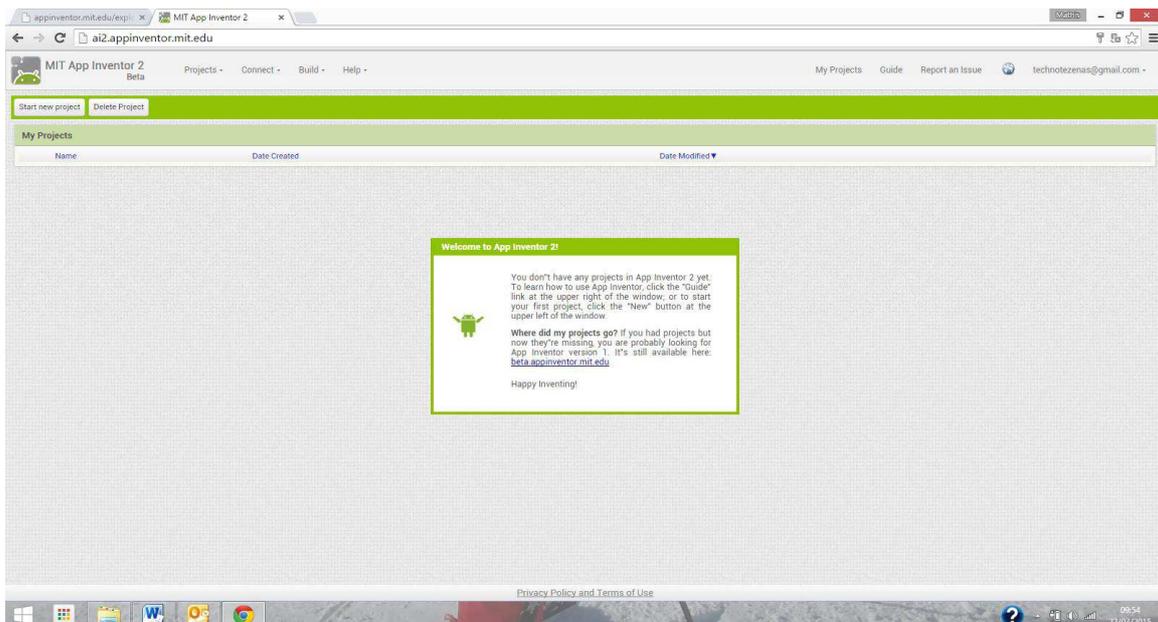


Fig III.10 : Environnement d'AppInventor

- On clique sur Projects / Start new project.

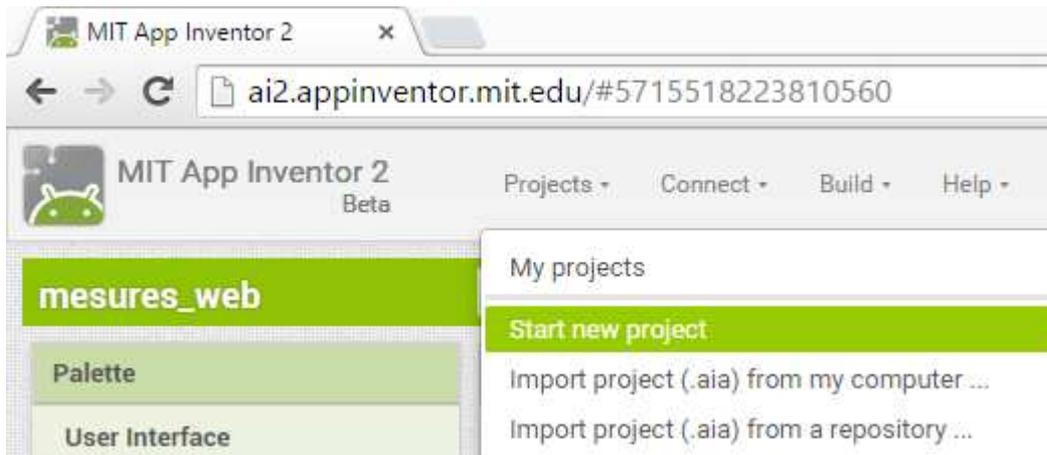


Fig III.11 : Ouverture d'un nouveau projet sous AppInventor2

- On obtient l'écran de travail suivant :

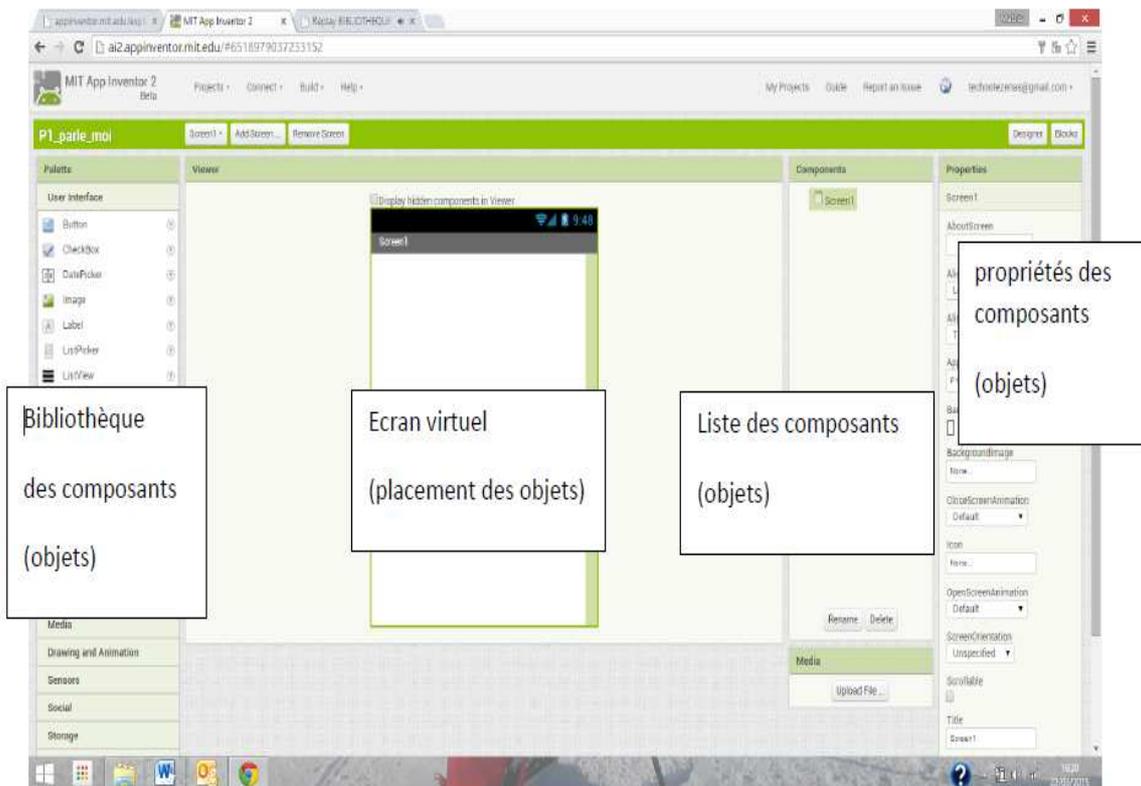


Fig III.12 : Ecran de travail sur AppInventor2

- On Vérifie le téléchargement et l'installation de l'application « MIT AI2 Companion » sur notre Smartphone ou tablette.
- On Vérifie l'accès wifi sur le Smartphone. Le PC et le Smartphone devront être connectés à un point d'accès internet.
- On lance l'application AI2 sur Smartphone.
- On flashe à l'aide d'une Cam du Smartphone le QRcode.

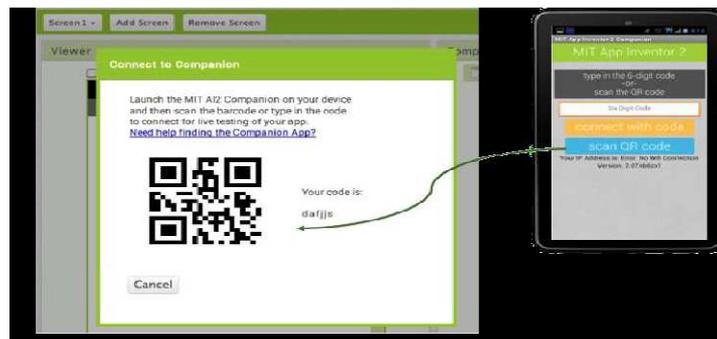


Fig III.13 : Code QRcode

- On place les éléments de l'interface de dialogue

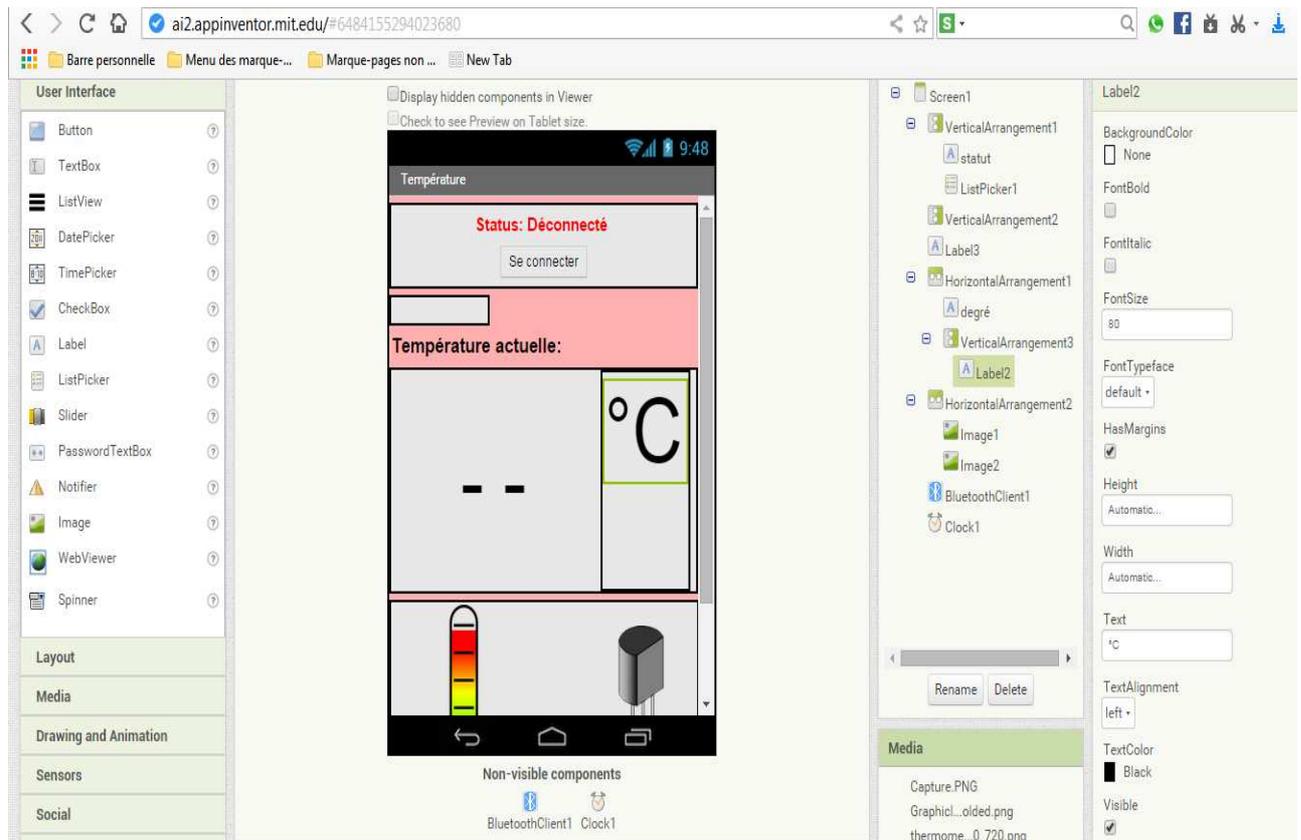


Fig III.14: Interface de dialogue

- On place les éléments de l'interface de programmation

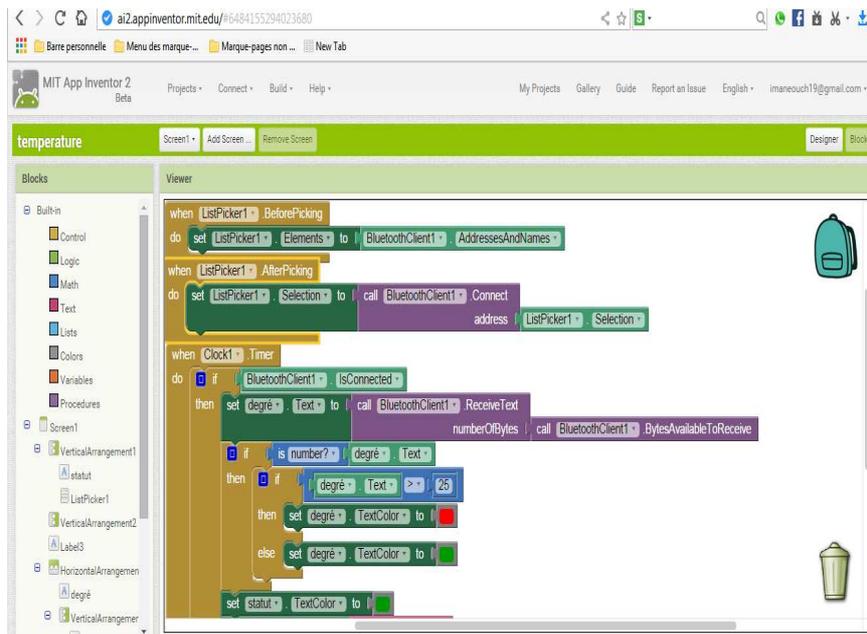


Fig III.15 : Interface de programmation

#### III.5.4. Avantages de l'application App Inventor [ 20]

L'application App Inventor permet de créer, sans connaissances en développement, une application Androïde. Les screenshots de l'interface étaient déjà présents sur le site web et le mode de fonctionnement assez bien détaillé. Une page web permet d'ajouter des widgets par glissé-déposé sur l'écran et un applet Java pour jouer avec les blocs de programmation (résultat du travail du MIT) .Le résultat s'affiche immédiatement sur le téléphone.

Pour les développeurs plus avancés. App Inventor pourrait servir à un prototypage rapide, s'il permettait d'aller un peu plus loin avec la plateforme Androïde.

#### III.5.5.Limitations [20]

- Impossible de s'interfacer avec le web et ses services.
- Ajouter un bout de code Java .

Evidemment, ces limitations se justifient complètement par le but de cette application : l'initiation des plus jeunes à une forme de contrôle de l'informatique – Ordinateur et téléphone.

### **III.6. Conclusion**

Dans ce chapitre on a présenté les logiciels utilisés pour la programmation et la simulation de notre travail.

On a présenté les logiciels de programmation tels que : Arduino, l'interface graphique LABVIEW, puis la simulation du programme pour vérifier son fonctionnement avant l'implémentation et la présentation de site de programmation de l'application App Inventor2. Comme nous venons de le voir, la phase des conceptions logicielles est une étape essentielle avant d'enchaîner l'étape de la réalisation ou l'implémentation.

Dans le chapitre 4 nous allons programmer, simuler et réaliser notre travail à l'aide de ces outils software.

*Chapitre*

---

4

*Programmation,  
réalisation et tests*

## **IV.1.Introduction**

L'étude de faisabilité nous a mené vers le choix des solutions technologiques des composants .Dans ce chapitre on va procédés aux étapes de réalisation. Donc on va faire une simulation avec l'outil informatique ISIS dans lequel on va faire toute les instructions du programme pour l'exécution, après la conformité de la simulation on fait le raccordement sur une plaque électronique à l'aide de l'outil ARES conformément aux exigences du cahier des charges du point de vue dimensionnement. L'étape suivante consiste à formuler une présentation d'une interface graphique ou HMI avec LABVIEW pour recevoir les résultats sur un PC ainsi que l'enregistrement dans une base de données sous Excel, l'affichage sur un périphérique de sortie mobile (Smartphone) et une commande a distance de refroidisseur nécessitent une application androïde .A la fin, on va survenir sur l'étape dite implémentation dans laquelle l'exécution des instructions du programme comme test pour confirmer le bon fonctionnement, dont le but d'exploiter des résultats et pouvoir faire une analyse ainsi que la localisation des contraintes de fonctionnement.

## **IV. 2. Simulation du projet avec l'outil ISIS**

### **IV. 2. 1. Réalisation de tout le système hors service**

Comme cela apparait sur la figure ( fig. IV. 1) l'outil ISIS nous a simplifié la mise en évidence du projet virtuellement pour pouvoir confirmer le fonctionnement avant sa réalisation en réel pour éviter des dégâts matériels .Le système comporte tous les composants choisis auparavant, avant la mise en fonctionnement du circuit nous devons injecter une logique dans le circuit de commande Arduino auxquelles il doit réagir.

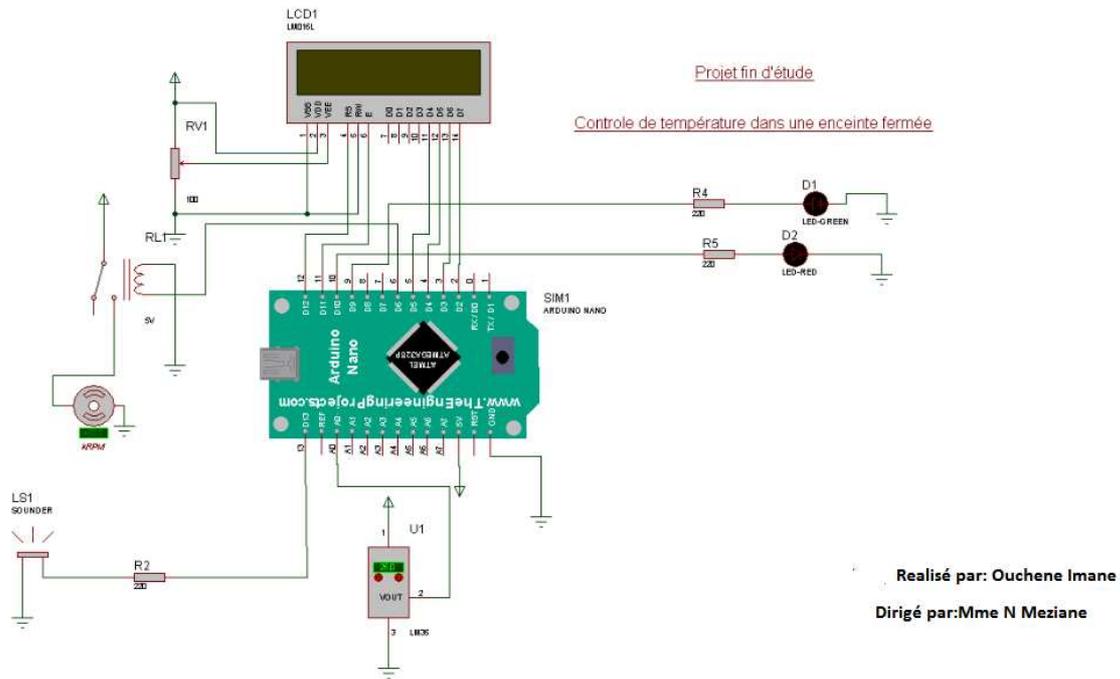


Fig. IV.1 : Représentation d’ensemble du système de mesure de température en simulation avec ISIS, avant sa mise en service.

### IV. 2. 2. Programme sous Arduino

Suivant la syntaxe spécialisée dédiée pour la programmation de l’Arduino le programme proposé pour exécuter la tâche demandée dans le cahier des charges est comme suit :

<code>#include&lt;LiquidCrystal.h&gt;</code>	}	Déclaration de la bibliothèque LCD
<code>#include &lt;SoftwareSerial.h&gt;</code>	}	Déclaration de la bibliothèque Bluetooth
<code>LiquidCrystal lcd(7,6,5,4,3,2);</code>	}	Déclaration de pins de LCD sur Arduino
<code>SoftwareSerial HC06(11,10);</code>	}	Déclaration de pins de Bluetooth sur Arduino

```
float temp;  
int pinOut=8;  
int ledgreen=9;  
int ledred=12;  
String commande;  
boolean modeAuto=true;  
int const son=13;
```

Déclaration  
des variables

```
void setup()  
{  
  Serial.begin(9600);  
  pinMode(ledgreen, OUTPUT);  
  pinMode(ledred, OUTPUT);  
  pinMode(8, UTPUT);  
  lcd.begin(16,2);  
}
```

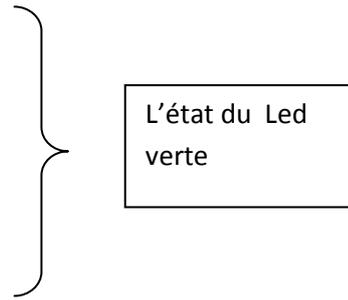
Configuration des  
ports en sortie

```
void loop()  
{  
  temp=analogRead(A0);  
  temp=(temp*500)/1024;  
  lcd.setCursor(0,0);  
  lcd.print("temp:");  
  lcd.print("c");  
  delay(100);  
}
```

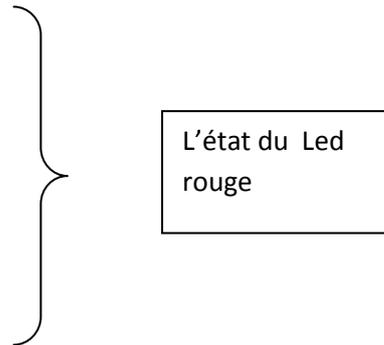
La lecture de capteur et la  
conversion de la lecture  
en degré Celsius

Affichage de la  
température sur écran  
LCD

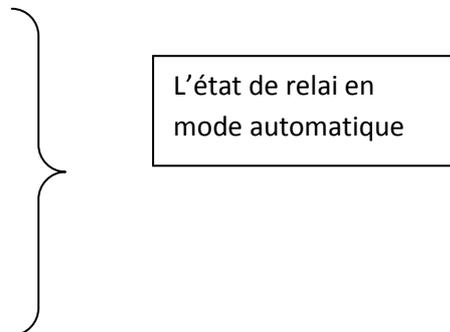
```
if (temp>20 && temp<30 )  
{digitalWrite(ledgreen, HIGH);  
}  
else {digitalWrite(ledgreen, LOW);  
}
```



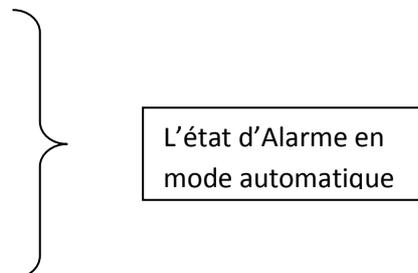
```
if (temp>30 )  
{digitalWrite(ledred, HIGH);  
}  
else  
{digitalWrite(ledred, LOW);  
}
```



```
if ( temp>32 && modeAuto==true)  
{digitalWrite(8, HIGH);  
}  
if(temp<32 && modeAuto==true)  
{digitalWrite(8, LOW);}
```



```
if ( temp>32 && modeAuto==true)  
{tone(son, 32);}   
if(temp<32 && modeAuto==true)  
{noTone(son);}
```



```

HC06.println(temp);
HC06.print(" ");
Serial.println(temp);
delay(1000);

```

Affichage de température sur Smartphone

```

while(HC06.available()){
  delay(3);
  char c = HC06.read();
  commande+=c;
}

```

La synchronisation de Bluetooth HC06

```

if(commande.length(>0){
  Serial.println(commande);
  if(commande=="A1"){modeAuto=true;}
  if(commande=="A0"){modeAuto=false;}
}

```

Le type de commande : automatique ou manuel

```

if(commande=="F0"){digitalWrite(8, LOW);}
  if(commande=="F1"){digitalWrite(8, HIGH);}

```

L'état de relai en mode Manuel

```

if(commande=="F1") {noTone(son);}
  if(commande=="F0") {noTone(son);}
  commande="";

```

L'état d'Alarme en mode Manuel

```

} } //Fin programme.

```

**IV. 2.3. Mise en service du système dans l'intervalle [ 20°,30°]**

La mise en service du système en température entre 20°C et 30°C ( $20^{\circ}\text{C} < T < 30^{\circ}\text{C}$ ) la valeur apparaît sur l'afficheur LCD et ainsi la logique fonctionne comme cela se voit sur la figure ( fig. IV. 2). La LED verte D1 est allumée, le relai est en position ouverte et le baffle ne produit aucun son.

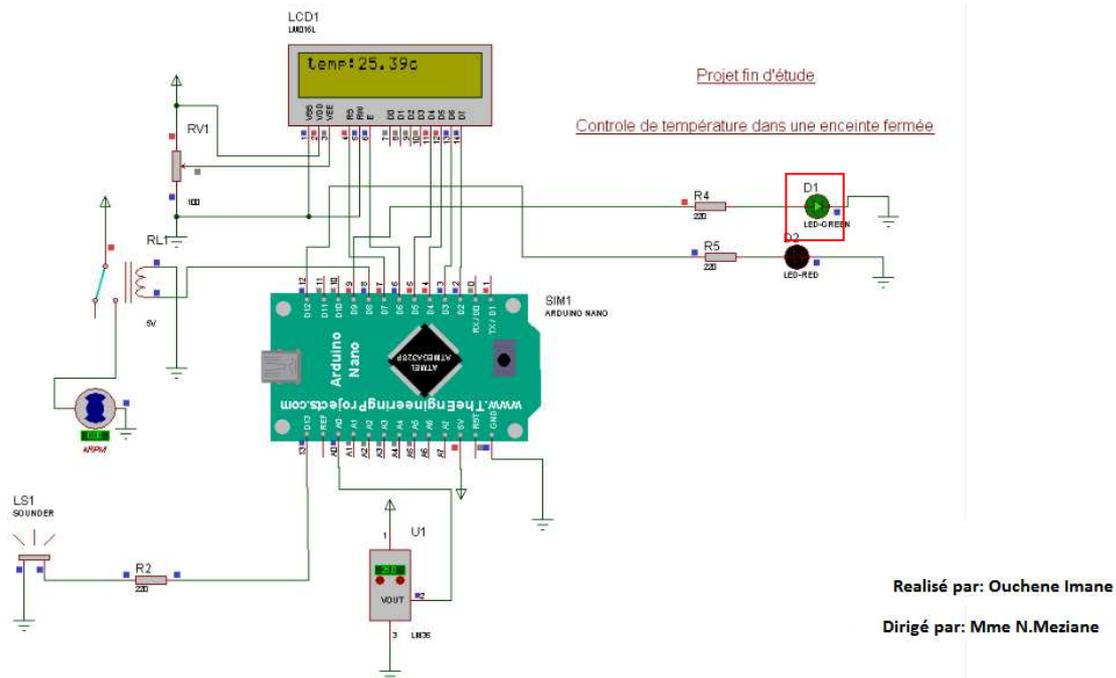


Fig. IV. 2 : Représentation du fonctionnement du système à une température de 25°C

**IV. 2.4. Mise en service du système dans l'intervalle ( $T > 32^{\circ}\text{C}$ )**

Pour une valeur de  $T = 35^{\circ}\text{C}$ , donc supérieure à  $32^{\circ}\text{C}$  la température est considérée par la logique comme une valeur très haute. On voit que la LED rouge est allumée, ainsi que le relai est excité, son contact se ferme alors le système de refroidissement commence pour diminuer la température comme c'est représenté sur la figure ( fig. IV. 3) par la rotation d'un moteur de ventilateur, sans pouvoir oublier l'alarme sonore signalant qui est faite pour ce but.

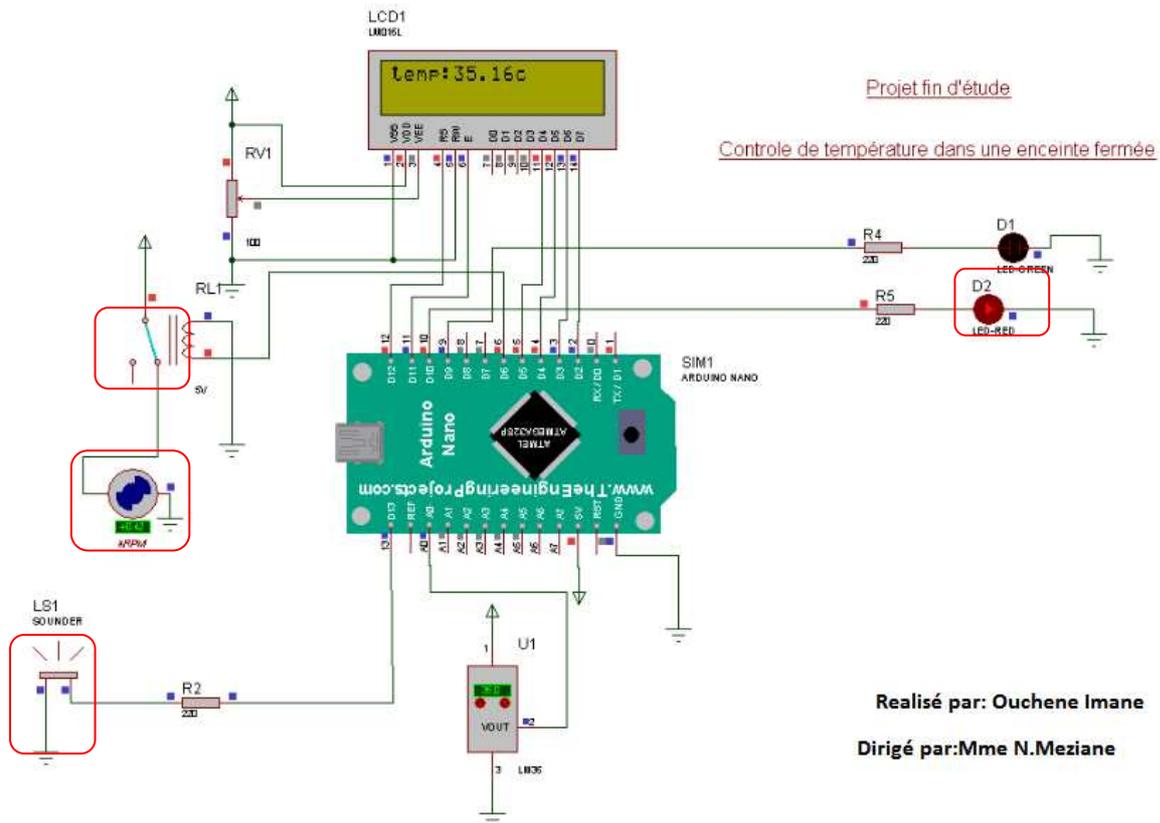


Fig. IV. 3 : Représentation du fonctionnement du système à une température de 35°C.

Donc on confirme que le système fonctionne et qu'il ne présente aucune anomalie, donc ce qui suit est la préparation de la carte PCB pour la réalisation réelle, juste un composant n'apparaît pas sur la simulation qui est le Bluetooth mais son apparition n'est pas aussi nécessaire vu la présence de l'afficheur LCD et que la transmission avec le Smartphone ne peut apparaître sur la simulation.

### IV.3. Interface graphique HMI (LABVIEW)

L'interface graphique a pour rôle de servir comme interface d'affichage pour permettre une communication entre le système de mesure et de contrôle et l'être humain qui a besoin de ces informations via un ordinateur c'est pour cela qu'on l'appelle HMI (Humain Machine Interface).

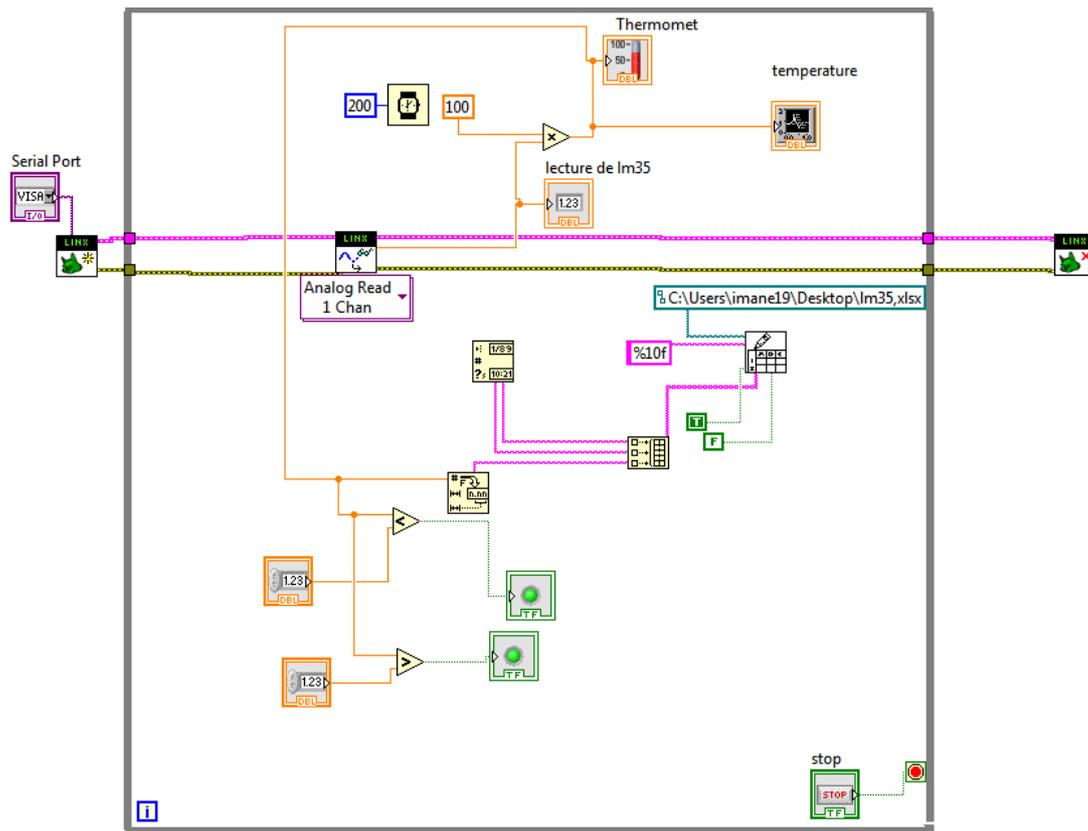


Fig IV.4 : Représentation de la programmation de l’HMI sous LABVIEW

La figure (fig.IV.4.) présente la programmation de l’interface graphique qui va transmettre les résultats par un affichage de la valeur numérique, un autre sur une barre graduée et un graphe qui montre l’évolution de la température en fonction du temps ainsi que le port de communication avec l’Arduino, en plus deux LED utilisée comme voyants sur l’état de température.

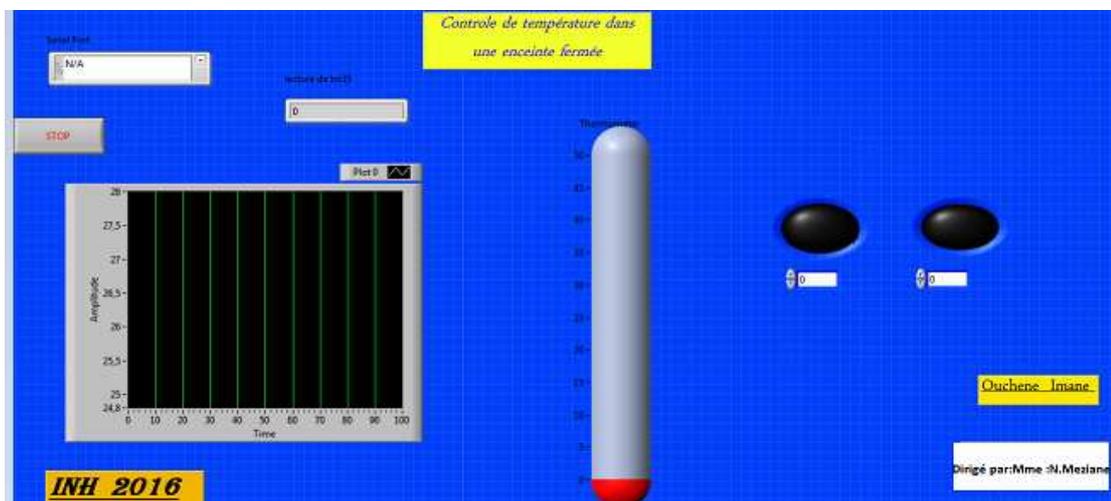


Fig IV.5 : Interface graphique développée

Une fois la configuration du port est terminée, on appuie sur le bouton START le processus démarre, il reçoit la donnée du capteur avec une période d'échantillonnage de 200ms .Le graphe affiche la variation de température en temps réel. Les figures ( fig. IV. 6, fig. IV. 7) présente l'affichage déjà programmé avec une variation des résultats numérique et graphique ainsi que l'état de la LED.

La différence entre les deux figures comme cela ce voit est dans la valeur de température mesurée et la réaction de la logique du système selon cette estimation.

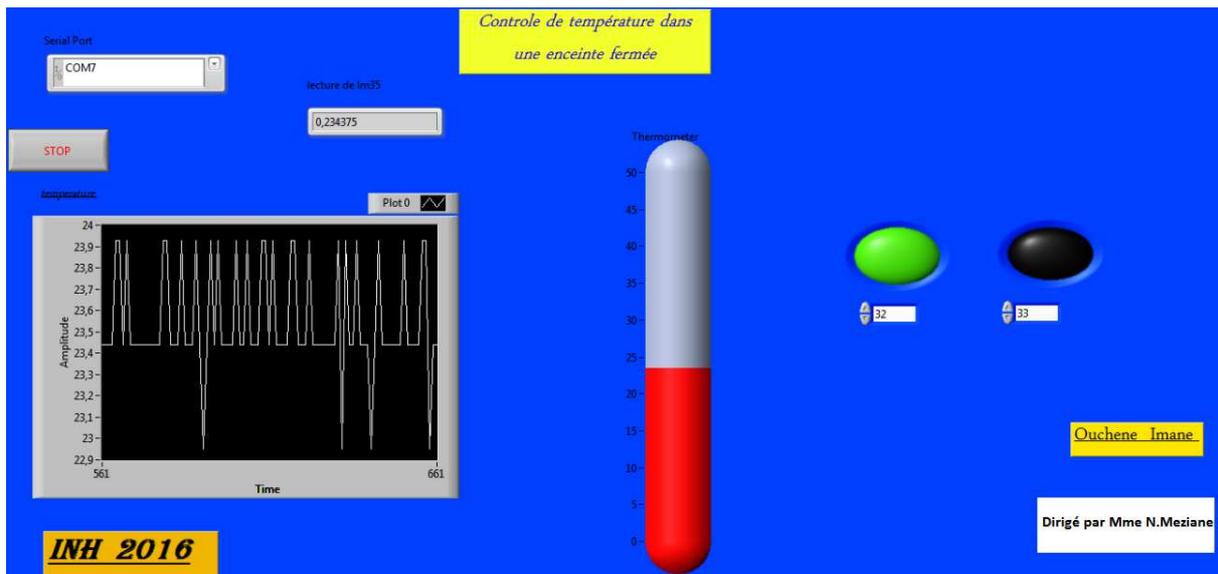


Fig IV.6 : Interface graphique du système en fonctionnement à une température de 23°C

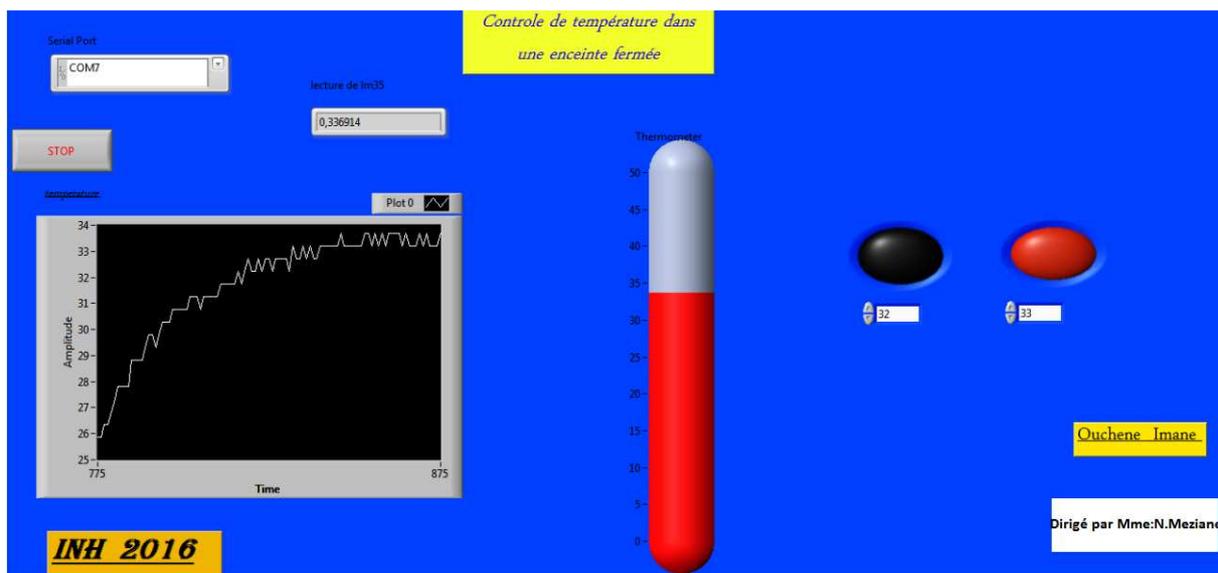


Fig IV.7 : Interface graphique du système en fonctionnement à une température de 33°C

### IV.3.1.Partie enregistrement des données

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
7308	17/04/2016	19:05	26,367187													
7309	17/04/2016	19:05	26,855469													
7310	17/04/2016	19:05	27,34375													
7311	17/04/2016	19:05	27,832031													
7312	17/04/2016	19:05	27,832031													
7313	17/04/2016	19:05	27,832031													
7314	17/04/2016	19:05	27,832031													
7315	17/04/2016	19:05	28,808594													
7316	17/04/2016	19:05	28,808594													
7317	17/04/2016	19:05	28,808594													
7318	17/04/2016	19:05	28,808594													
7319	17/04/2016	19:05	29,296875													
7320	17/04/2016	19:05	29,785156													
7321	17/04/2016	19:05	29,785156													
7322	17/04/2016	19:05	29,296875													
7323	17/04/2016	19:05	29,785156													
7324	17/04/2016	19:05	30,273437													
7325	17/04/2016	19:05	30,273437													
7326	17/04/2016	19:05	30,273437													
7327	17/04/2016	19:05	30,761719													
7328	17/04/2016	19:05	30,761719													
7329	17/04/2016	19:05	30,761719													
7330	17/04/2016	19:05	30,761719													
7331	17/04/2016	19:05	30,761719													
7332	17/04/2016	19:05														
			31,25													

Fig IV.8 : Représentation du tableau Excel qui enregistre les différentes valeurs de température

Le tableau qui apparait sur la figure ( Fig IV.8) affiche les différentes valeurs de mesure chacune attribue a un temps précis c-à-d l'évolution de la température en fonction du temps ce qui est important et dans une minute on trouve 300 valeurs, donc il y a un changement chaque 200ms avec fréquence de 0.005 HZ .

## IV.4. Conception de la carte PCB de commande (ARES)

La première figure (fig IV.9) présente la face A, elle montre la disposition des différentes composantes de la partie commande. La seconde figure (figIV.10) présente la face B et elle montre les branchements, les fils et les connexions. La troisième figure (fig IV.11) présente les deux faces A et B, elle montre la disposition et les connexions entre les différentes composantes.

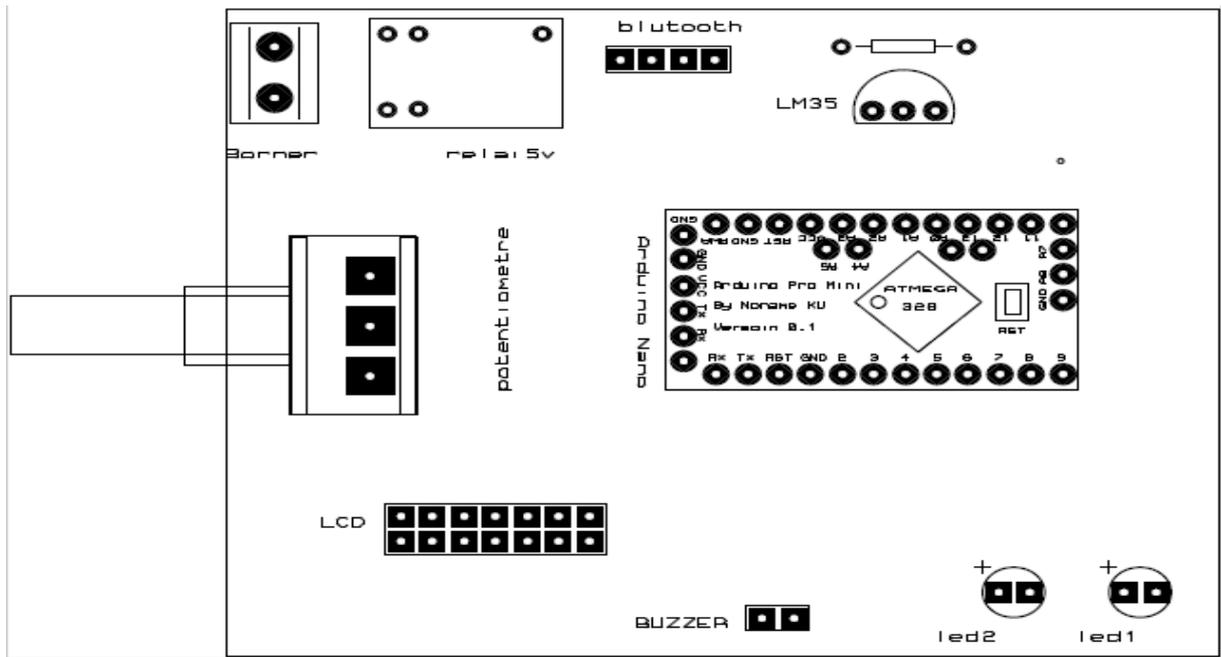


Fig IV.9: Carte de commande face A.

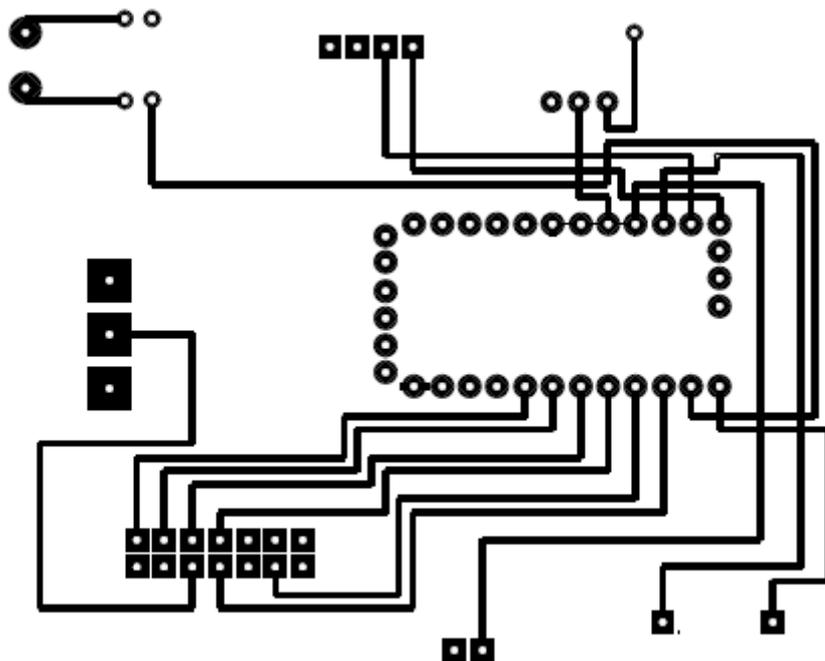


Fig IV.10 : Carte de commande face B.

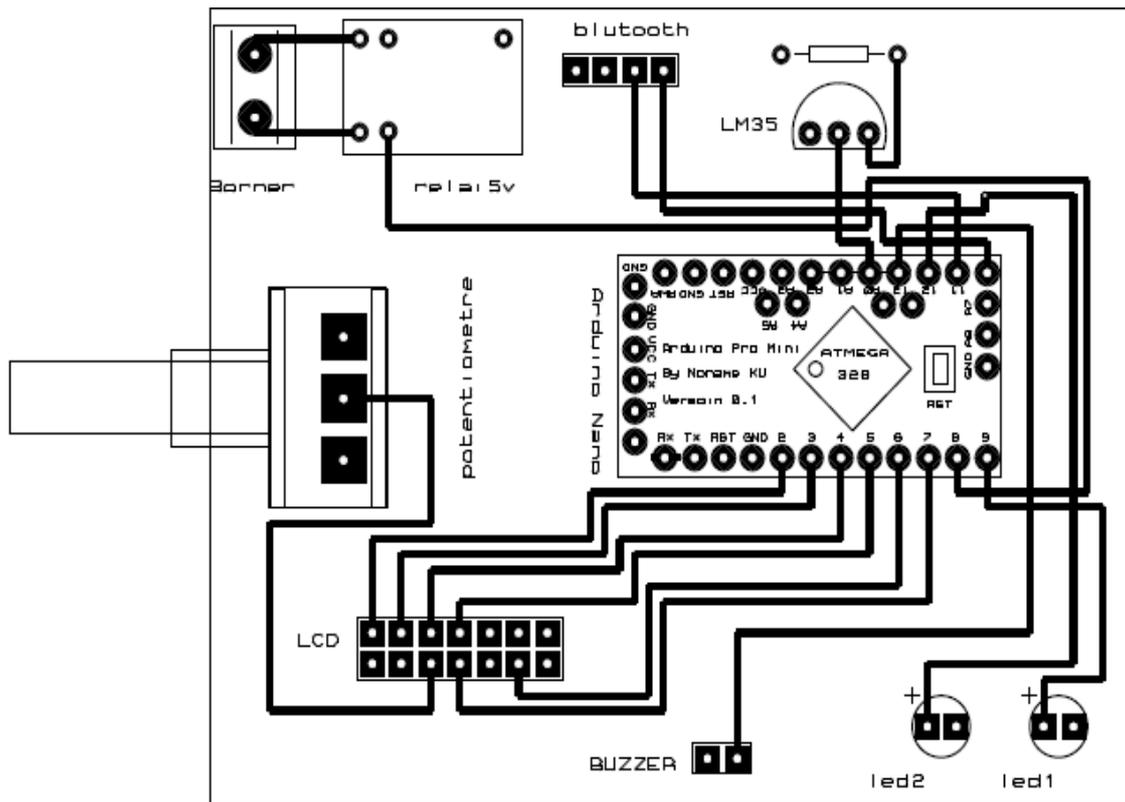


Fig IV.11 : Carte de commande faces A et B.

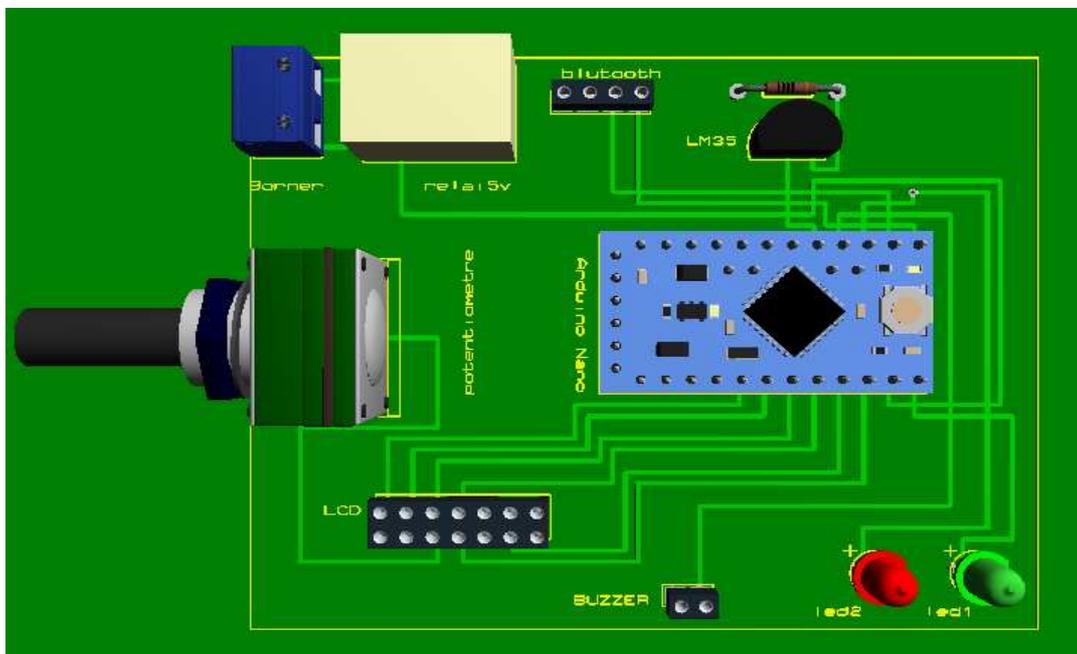


Fig IV.12: Visualisation de la carte de commande en 3D vue de dessus .

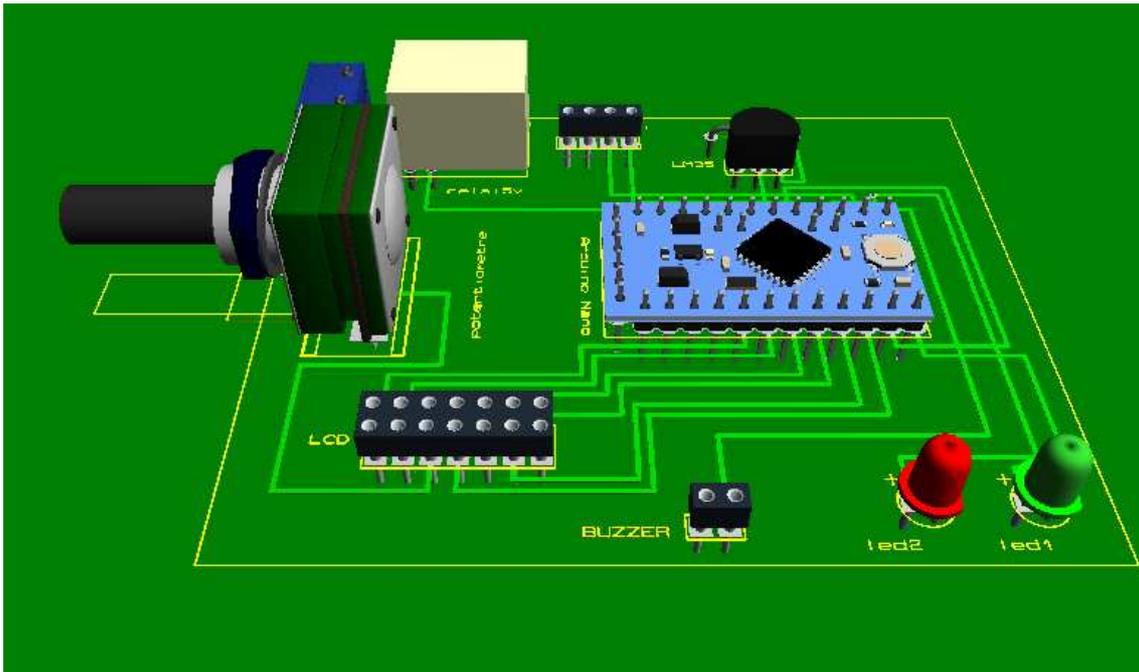


Fig IV.13 : Visualisation de la carte de commande en 3D vue de profile.

#### **IV.5. Commande du système à partir d'un Smartphone ANDROID via la carte Arduino Nano**

On veut afficher la température sur un Smartphone, et commander le système de refroidissement à partir de deux boutons : le premier pour le mode de commande automatique ou manuel et l'autre pour l'option ON/OFF pour laisser le choix à l'utilisateur de choisir l'état de refroidisseur.

La figure (fig IV.14) montre l'emplacement des différents composants de l'interface de l'application dans le site App Inventor2.

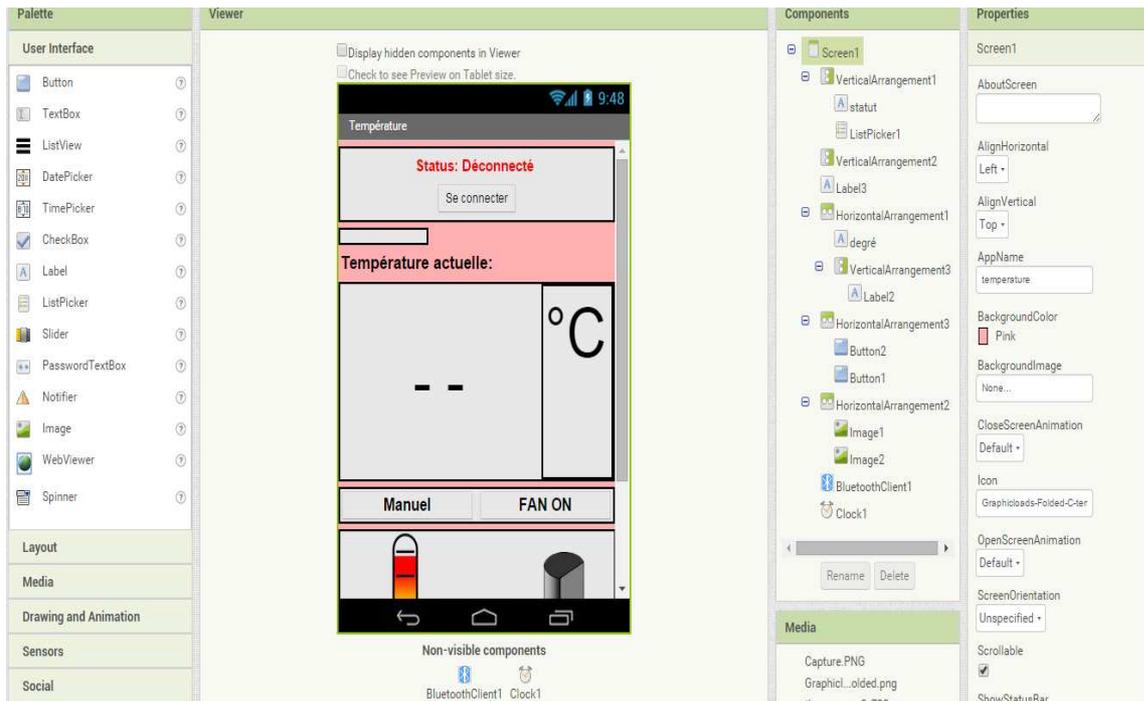


Fig IV.14 : Interface développée pour l'application sur le site App Inventor2

### IV.5.1.Code source de l'application

```

when ListPicker1.BeforePicking
do set ListPicker1.Elements to BluetoothClient1.AddressesAndNames

when ListPicker1.AfterPicking
do set ListPicker1.Selection to call BluetoothClient1.Connect
address ListPicker1.Selection

when Clock1.Timer
do if BluetoothClient1.IsConnected
then set degré.Text to call BluetoothClient1.ReceiveText
numberOfBytes call BluetoothClient1.BytesAvailableToReceive
if is number? degré.Text
then if degré.Text > 32
then set degré.TextColor to red
else set degré.TextColor to green
set statut.TextColor to green
set statut.Text to "Statut: Connecté"
set ListPicker1.Enabled to false
else set ListPicker1.Enabled to true
set statut.TextColor to red
set statut.Text to "Statut: Déconnecté"
    
```

Récupération de la liste des périphériques Bluetooth appariés sur Smartphone

Connexion au périphérique sélectionné

Les messages émis par Arduino sur le port Bluetooth

Si  $T > 32$  la lecture est en rouge si non la lecture devient en verte

L'état de la connexion

```

when Button2 . Click
do
  if Button2 . Text = "Automatique"
  then
    call BluetoothClient1 . SendText
      text "A1"
    set Button2 . Text to "Manuel"
    set Button1 . Enabled to false
  else
    call BluetoothClient1 . SendText
      text "A0"
    set Button2 . Text to "Automatique"
    set Button1 . Enabled to true
  
```

Lorsque l'évènement «automatique » apparaît Envoyer la valeur A 1 sur le port Bluetooth et l'évènement « manuel » apparaît

Lorsque l'évènement « manuel» apparaît envoyer la valeur A0 sur le port Bluetooth et l'évènement devient « automatique »

```

when Button1 . Click
do
  if Button1 . Text = "FAN ON"
  then
    call BluetoothClient1 . SendText
      text "F1"
    set Button1 . Text to "FAN OFF"
  else
    call BluetoothClient1 . SendText
      text "F0"
    set Button1 . Text to "FAN ON"
  
```

Lorsque l'évènement « FAN ON» apparaît envoyer la valeur F1 sur le port Bluetooth et l'évènement devient « FAN OFF »

Lorsque l'évènement « FAN OFF» apparaît envoyer la valeur F0 sur le port Bluetooth et l'évènement devient « FAN ON »

### IV.5.2.Mise en marche de l'application

La figure IV.15 présente l'application hors connexion.

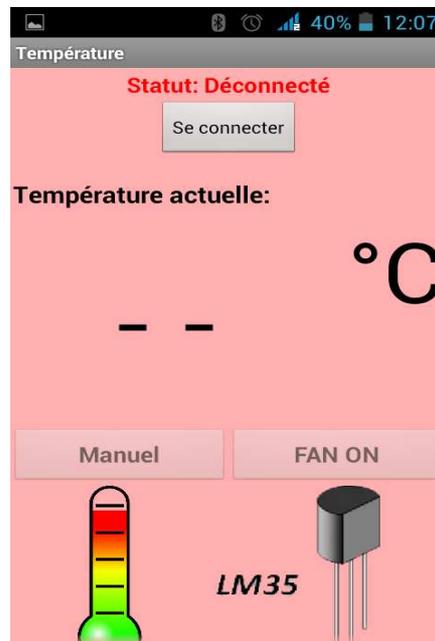


Fig IV.15 : Application hors connexion

On clique sur le bouton « Se connecter », une liste de Bluetooths appariée, on choisit notre Bluetooth HC06 .Les figures (IV.16, IV.17) représente l'application en état de fonctionnement Automatique et Manuel.



Fig IV.16: Application en état de fonctionnement en mode automatique



Fig IV.17 : Application en état de fonctionnement en mode Manuel

## IV.6.Partie pratique

### IV.6.1. Conception du boîtier

Nous avons fabriqué un boîtier (Fig IV.17) en bois dont les dimensions sont :

- Longueur 160mm.
- Largeur 80mm.
- Hauteur 100mm.

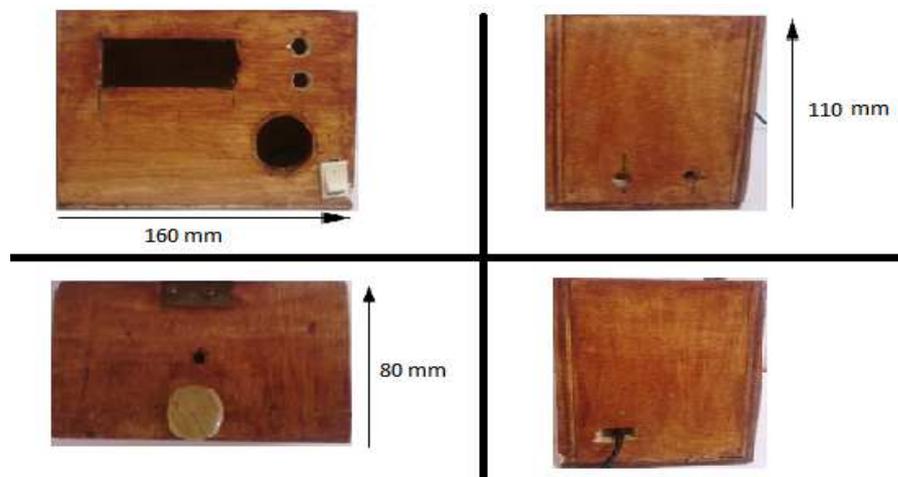


Fig IV.18 : Boîtier.

Nous avons découpé :

- La vue de dessus pour fixer le LM35.
- La vue de face pour fixer le LCD, le bouton ON/OFF les voyants (LEDS) de signalisation et l'alarme.
- La vue de gauche pour fixer le potentiomètre (celui du LCD pour modifier le contraste) et l'entrée des fils d'alimentation de la prise 220V.
- La vue de droite pour l'entrée de fil d'alimentation de boîtier.

#### **IV.6.2. Carte de commande réalisée**

Après l'étude, la conception et le choix des composants pour la carte de commande nous avons terminé la réalisation de notre carte représentée sur (Fig IV.19)

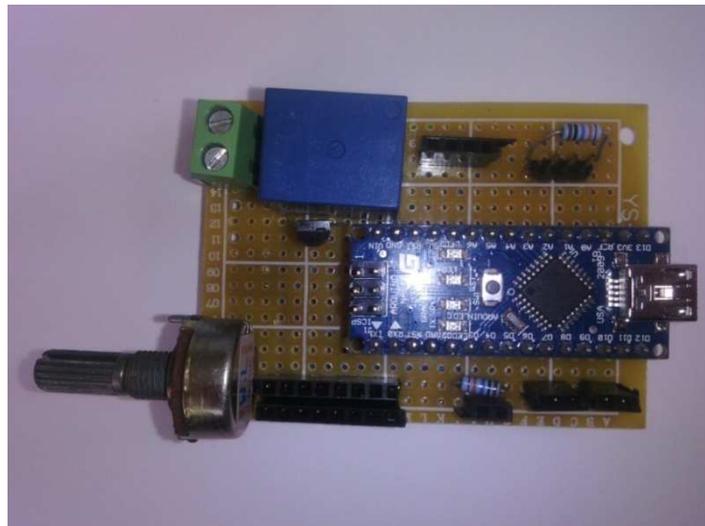


Fig IV.19 : Carte de commande réalisée

Les composants constituant la carte sont :

- Arduino NANO.
- Un Relai 5v.
- Un potentiomètre.
- un support pour le branchement du LCD.
- un support pour le branchement de Bluetooth.
- un support pour le branchement de LM35.

- un support pour le branchement d'alarme.
- 02 supports pour les LED.
- 02 Résistances.
- Un Connecteur pour l'alimentation.

Remarque : On ajoute un transistor (A2222) pour protéger l'Arduino du courant de décharges de la bobine.

### IV.6.3. Projet final

Après assemblage des différentes composantes, la figure IV.20 montre le projet final .



Fig IV.20 : Projet final

### IV.7. Etude technico-économique

Pour estimer le coût de fabrication de notre boîtier, on a fait une étude technico-économique pour calculer le coût de la partie commande excluant les équipements suivants : l'alimentation de courant, l'élément de refroidissement, le PC et le Smartphone qu'on a utilisées. Le tableau (Tab. IV.1) affiche les détails du prix unitaire, quantités et le total.

Composant	Prix unitaire (DA)	quantité	Total (DA)
Arduino NANO	2000,00	1	2000,00
LM35	200,00	1	200,00
Bluetooth	1400,00	1	1400,00
Afficheur LCD(16*2)	700,00	1	700,00
Relai 5v	200,00	1	200,00
Potentiomètre	40,00	1	40,00
LED	10,00	2	20,00
RESISTANCE	10,00	2	20,00
BUZZER	200,00	1	200,00
Carte PCB	50,00	1	50,00
<b>Total</b>			<b>4650,00</b>

Tab. IV. 1 Tableau d'estimation des prix des éléments utilisés.

Pour rester dans la marge budgétaire du cahier des charge, nous devons jouer sur le coût de main d'œuvre donc nous avons préparé la carte PCB nous même sans s'orienter vers sa préparation par une prestation de service.

L'estimation du coût de main d'œuvre inclus dans l'étude, la réalisation, la simulation et toutes les opérations software, les tests et même le transport nous arrêtons le montant des travaux de main d'œuvre à la somme de 1000.00 DA.

Donc le prix estimatif de toute l'installation est de 5650.00DA qui est une valeur convaincante qui n'est pas chère au delà de la marge demandée dans le cahier des charges vue la fonction qu'il va accomplir, et sachant que la majorité des composants sont branché sur des fiches non pas soudé sur la PCB, pour réduire le coût et la durée de la maintenance.

## **IV.8.Conclusion**

La réalisation pratique du système est réussite vue les résultats des opérations de simulation, programmation de la logique, la préparation de l'interface qui affiche les résultats sur PC à travers LABVIEW, ainsi que la réalisation .Les tests montrent la possibilité d'affichage sur PC, sur LCD, les LED s'allument suivant les intervalles déterminés auparavant. Sans oublier l'affichage de l'évolution de la température en fonction du temps et le stockage des informations pour garder l'historique dans un tableau Excel pour construire une base de données, et une application Androïde permettant d'afficher la température et commander le refroidisseur sur Smartphone conformément aux exigences du cahier des charges.

# *Conclusion générale*

---

## **Conclusion générale**

Ce projet a été à la fois très intéressant et très formateur. Nous avons pu concevoir et développer un système de contrôle de température.

Nous avons présenté les différents types des capteurs de température à fin de choisir le meilleur capteur qui satisfait nos besoins.

Par la suite, Nous avons présenté les exigences et les objectifs du cahier des charges pour mener à bien notre travail .Ensuite nous avons passé vers le choix d'une solution technologique et citer les différents composants utilisés pour notre projet. Nous avons présenté les outils de programmation et simulation que nous avons utilisé.

Nous avons aussi effectué la programmation, la simulation électronique de carte et la réalisation de l'interface homme/machine respectivement par l'utilisation de multitude de logiciel à savoir : Arduino, Proteus et le LabView. Ainsi que la mise en marche d'une application Android pour afficher la température et commander le système par Smartphone.

Enfin, la dernière phase a été consacrée à la réalisation et les tests de fonctionnement.

Ce travail nous a permis de nous mettre à l'épreuve d'une part et d'améliorer d'autre part nos connaissances théoriques et d'acquérir une bonne expérience au niveau de la réalisation pratique en : électronique, électrotechnique ainsi que la manipulation des outils tel que : la programmation, la simulation des schémas électroniques et informatique (Proteus ISIS, Arduino, LabView).

En perspective, nous proposons l'amélioration de la commande électronique, par l'augmentation de la fiabilité de la boucle de mesure par la triple redondance appliquée sur l'entrée et la sortie. Au début, les capteurs sont en redondance. Si l'un des capteurs est endommagé, la tolérance de cette panne se fait par la logique de la Triple Redondance Modulaire (TMR), ainsi pour l'application de l'action il faut installer 3 relais en parallèle dans le cas de défaillance d'un entre eux l'autre prend en main l'exécution de la commande.

*Références  
Bibliographiques*

---

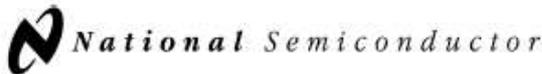
## Références bibliographiques

- [1] M. Civate et F. Mandel, « **Fiche descriptive sur les instruments de mesure météorologique** » -Version 1.0 - décembre 2008
- [2] INERIS DRA. « **pulverulent-detecter-temperature** ».vers1. Avril 2005
- [3] Site internet ,[www.alldatasheet.com/ LM35 Precision Centigrade Temperature Sensors](http://www.alldatasheet.com/LM35)
- [4] Site internet ,<https://www.arduino.cc/en/Main/ArduinoBoardNano>
- [5] **Datasheet GDM-2004** : <https://www.sparkfun.com/datasheets/LCD/GDM2004D>
- [6] Site internet ,<http://fr.wikipedia.org/wiki/Potentiomètre>
- [7] Site internet ,<http://fr.wikipedia.org/wiki/Buzzer>
- [8] Site internet ,<http://fr.wikipedia.org/wiki/Led>
- [9] Site internet ,<http://fr.wikipedia.org/wiki/Relai>
- [10] Site internet ,[http://category.alldatasheet.com/index. 5V%20RELAY](http://category.alldatasheet.com/index.5V%20RELAY)
- [11] Site internet ,<http://arduino.cc/fr/Main/DebuterPresentationLogiciel>
- [12] : S. Landrault, H. Weisslinger , « **Arduino : Premiers pas en informatique embarquée** »  
Le blog d'Eskimon Édition : 19 juin 2014
- [13] Massimo Banzi , « **Démarrer avec Arduino** », Dunod, 2011
- [14] **Manuel de l'utilisateur LABVIEW**, Edition de juin 2003
- [15] [http://www.labcenter.com/products/pcb/schematic\\_intro.cfm](http://www.labcenter.com/products/pcb/schematic_intro.cfm)
- [16] **Intelligent Schematic Input System**, User Manual Issue 6.0 - Novembre 2002
- [17] [http://www.labcenter.com/products/pcb/pcb\\_intro.cfm](http://www.labcenter.com/products/pcb/pcb_intro.cfm) .Introduction à ARES
- [18] Gonzalez. C, Huré, E. & Picot-Coupey, K. « **Usages et valeurs des applications mobiles pour les consommateurs** » ,2012, November
- [19] Garin, F. « **Android-Concevoir et développer des applications mobiles et tactiles** »2<sup>e</sup> édition. ,2011.
- [20] MIT News Office « **The MIT roots of Google's new software** », 19 août 2010.

# *Annexe*

---

Datasheet de LM35 :



November 2000

# LM35 Precision Centigrade Temperature Sensors

## General Description

The LM35 series are precision integrated-circuit temperature sensors, whose output voltage is linearly proportional to the Celsius (Centigrade) temperature. The LM35 thus has an advantage over linear temperature sensors calibrated in <sup>o</sup> Kelvin, as the user is not required to subtract a large constant voltage from its output to obtain convenient Centigrade scaling. The LM35 does not require any external calibration or trimming to provide typical accuracies of  $\pm 1/4^{\circ}\text{C}$  at room temperature and  $\pm 3/4^{\circ}\text{C}$  over a full  $-55$  to  $+150^{\circ}\text{C}$  temperature range. Low cost is assured by trimming and calibration at the wafer level. The LM35's low output impedance, linear output, and precise inherent calibration make interfacing to readout or control circuitry especially easy. It can be used with single power supplies, or with plus and minus supplies. As it draws only 60  $\mu\text{A}$  from its supply, it has very low self-heating, less than  $0.1^{\circ}\text{C}$  in still air. The LM35 is rated to operate over a  $-55^{\circ}$  to  $+150^{\circ}\text{C}$  temperature range, while the LM35C is rated for a  $-40^{\circ}$  to  $+110^{\circ}\text{C}$  range ( $-10^{\circ}$  with improved accuracy). The LM35 series is available pack-

aged in hermetic TO-46 transistor packages, while the LM35C, LM35CA, and LM35D are also available in the plastic TO-92 transistor package. The LM35D is also available in an 8-lead surface mount small outline package and a plastic TO-220 package.

## Features

- Calibrated directly in <sup>o</sup> Celsius (Centigrade)
- Linear + 10.0 mV/<sup>o</sup>C scale factor
- 0.5<sup>o</sup>C accuracy guaranteeable (at +25<sup>o</sup>C)
- Rated for full  $-55^{\circ}$  to  $+150^{\circ}\text{C}$  range
- Suitable for remote applications
- Low cost due to wafer-level trimming
- Operates from 4 to 30 volts
- Less than 60  $\mu\text{A}$  current drain
- Low self-heating,  $0.08^{\circ}\text{C}$  in still air
- Nonlinearity only  $\pm 1/4^{\circ}\text{C}$  typical
- Low impedance output,  $0.1 \Omega$  for 1 mA load

## Typical Applications

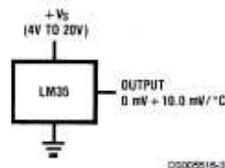
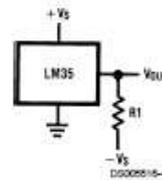


FIGURE 1. Basic Centigrade Temperature Sensor ( $+2^{\circ}\text{C}$  to  $+150^{\circ}\text{C}$ )



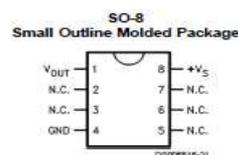
Choose  $R_1 = -V_S/50 \mu\text{A}$   
 $V_{OUT} = +1,500 \text{ mV at } +150^{\circ}\text{C}$   
 $= +250 \text{ mV at } +25^{\circ}\text{C}$   
 $= -550 \text{ mV at } -55^{\circ}\text{C}$

FIGURE 2. Full-Range Centigrade Temperature Sensor

## Connection Diagrams



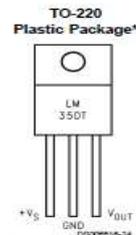
\*Case is connected to negative pin (GND)  
 Order Number LM35H, LM35AH, LM35CH, LM35CAH or LM35DH  
 See NS Package Number H03H



N.C. = No Connection  
 Top View  
 Order Number LM35DM  
 See NS Package Number M08A

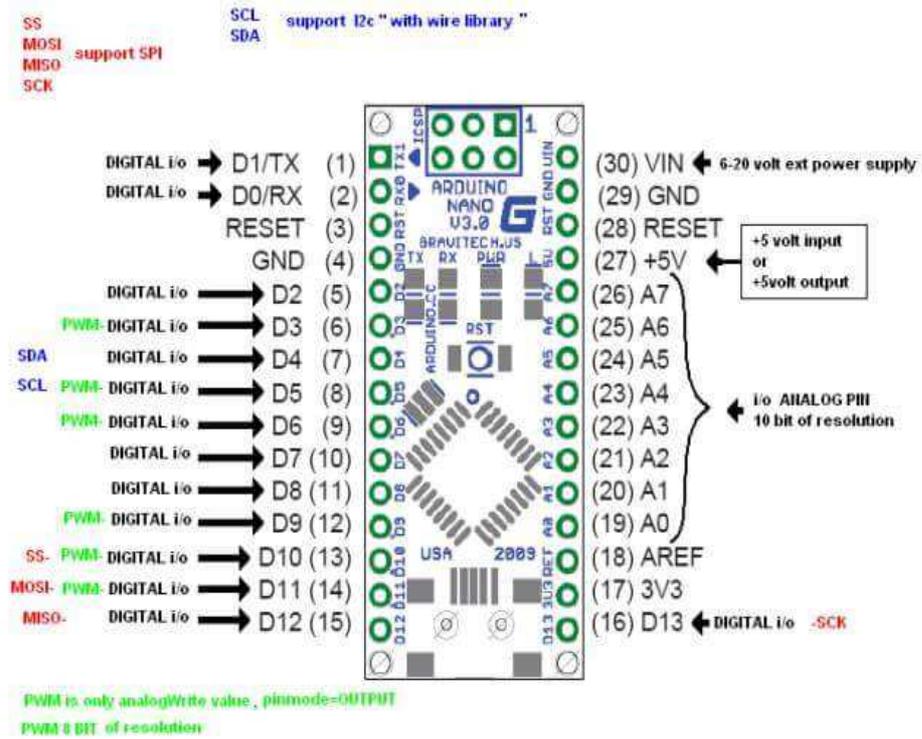
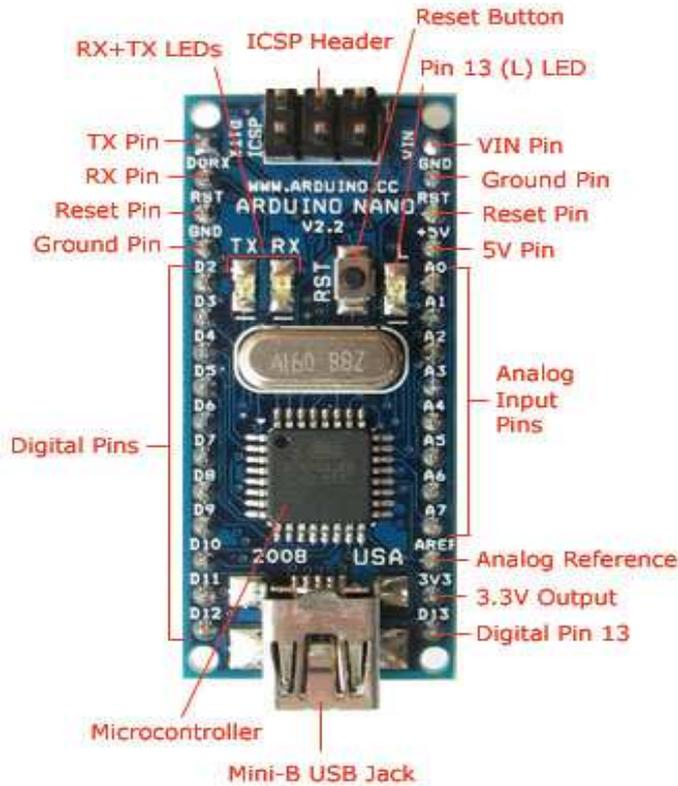


Order Number LM35CZ, LM35CAZ or LM35DZ  
 See NS Package Number Z03A



\*Tab is connected to the negative pin (GND).  
 Note: The LM35DT pinout is different than the discontinued LM35DP.  
 Order Number LM35DT  
 See NS Package Number TA03F

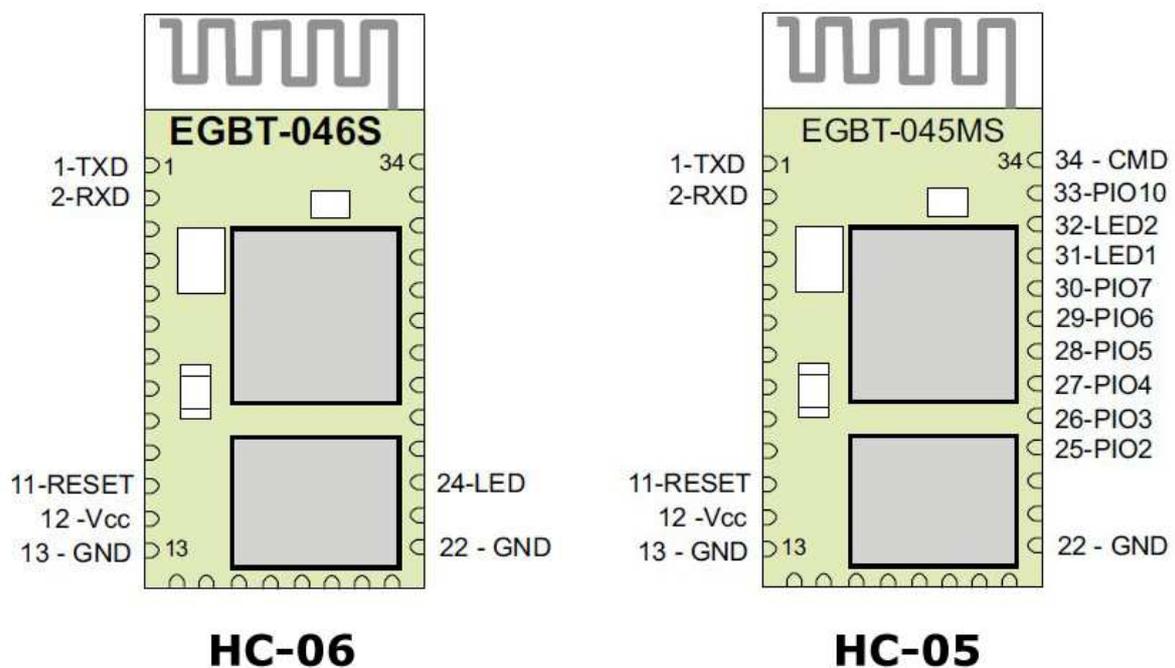
Datasheet d'ARDUINO NANO :



Datasheet de LCD (16\*2) :



Datasheet de Bluetooth HC06 :



## ملخص

في هذا المشروع نهتم بدراسة وتصميم نظام القياس والتحكم في درجة الحرارة. لتلبية الحاجة قمنا بدراسة موسعة و اقتصادية لمختلف الوسائل والمكونات, و أيضا تصميم نظام التحكم الذي يوفر سيطرة على مختلف وظائف الخدمة التي تم تحديدها في دفتر الشروط عبر برمجة منطق التحكم في الأداة أردوينو نانو , أخيرا نناقش مرحلة التنفيذ أو ما يعرف بالتصنيع والاختبارات و التي تهدف إلى استغلال و تحليل نتائج الاختبارات.

**كلمات مفتاحية :** أردوينو نانو- شاشة ال سي دي- مضيء صمام- صفارة- جهاز استشعار درجة الحرارة- المرحل - بلوتوث- تحكم بعرض النبضة

## Résumé

Ce projet consiste à faire l'étude, la conception ainsi que la réalisation de la partie commande d'un système de mesure et de régulation de température. Pour répondre à ce besoin, nous avons élaboré une étude bibliographique ainsi qu'une étude critique des différents instruments et composants. Ensuite, une conception du système de commande et les différents modularités à choisir et la méthode de commande utilisée, puis la programmation d'une logique dans le circuit de commande Arduino-nano qui assure les fonctions de services qui sont définies dans le cahier des charges. Enfin nous aborderons la phase dite implémentation ou bien les tests de fonctionnement, dont le but est l'exploitation des résultats d'analyse et la localisation des contraintes de fonctionnement.

**Mots clefs :**Arduino Nano - Afficheur LCD -LED-Alarme- capteur de température -Relai-Bluetooth - interface graphique

## Abstract

This project consists to study, design, and realization of the command and control part of temperature measuring and regulation system. To meet this need, we've developed a literature review and a critical study of various instruments and components, then a design of the control system and the different modularity to choose and the control method used, then programming the logical component Arduino-nano that provides service functions that are defined in the design of conditions. Finally we discussed the implementation phase or operating tests, which aims to the exploitation results of analysis and location of operating limits.

**Key words:** Arduino Nano -LCD display -LED-Buzzer- temperature sensor -Relay-Bluetooth - graphical interface