

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Direction Générale des Études Technologiques
INSTITUT SUPERIEUR DES ÉTUDES TECHNOLOGIQUES DE
BIZERTE

Département de Génie Électrique

RAPPORT DE
PROJET DE FIN D'ETUDES

En vue de l'obtention de : Licence Appliquée en Génie Électrique

**conception et réalisation d'un banc d'essai
des capteurs a base d'une carte arduino uno
et supervisé par labview**

Effectué à : ISET BIZERTE

Élaboré par :

Sellami Ikbel (AII)

Aouini Ichrak (AII)

Encadré par :

M^r. Ben chehida Ramzi (ISET Bizerte)

M^r. Liouane Saber (ISET Bizerte)

Soutenu le 12/06/2017 devant la commission composée de :

Président : M^{me} shaiek Yosra

Rapporteur : M^r Aouina Nourallah

Membre : M^{me} Boulehmi hela

Référence	Dép.	Génie Électrique
	A.U	2016-2017
	N°	AII 01.17

Dédicace

APRÈS AVOIR RENDU GRÂCE À **ALLAH**, NOUS PRIONS SUR SON
PROPHÈTE (PSL).

NOUS DÉDIONS CE MODESTE TRAVAIL À

NOS PARENTS

TOUTE LA PROMOTION DE TECHNICIEN SUPÉRIEURE DE CETTE
ANNÉE.

TOUTES LES PERSONNES QUI DE PRÈS OU DE LOIN ONT CONTRIBUÉES
À NOTRE RÉUSSITE.

Remerciement

En premier lieu, nous tenons à remercier et à exprimer nos gratitudes à nos encadreur Mr ben CEHIDA Ramzi et Mr Lioune Saber pour leur aides incessants, leurs conseils et les éclaircissements qu'ils ont fournis durant notre projet de fin d'études.

Nous témoignons reconnaissance et gratitudes aux aimables personnes quelque soit de près ou de loin, qui nous ont soutenu, acceptent et orientant durant notre projet de fin d'étude au sein de l'institut supérieure des études technologiques de Bizerte et spécialement dans le département génie électrique.

Enfin, nous remercions vivement les membres de jury.

Sommaire

Introduction générale.....	1
Chapitre1 :Etude de l'art.....	1
Introduction.....	3
1. Présentation ARDUINO	3
1.1. Matériel	3
Les plus utilisés sont :.....	5
1.2. Logiciel.....	6
1.3. La raison de choisir ARDUINO	7
2. Le logiciel d'interfaçage labVIEW	8
2.1. Définition	8
2.2. Domaine d'utilisation.....	8
2.2. Le composant d'un VI.....	9
2.3. Les palettes	10
3. Les capteurs.....	12
3.1. Le capteur de température LM35	12
3.2. Le capteur de température PT100	13
4. Les conditionneurs des capteurs passif et actif.....	14
4.1. Définition	14
4.2. Conditionneur de capteur passif.....	14
4.3. Conditionneur de capteur actif :.....	15
5. Les moteurs :.....	15
5.1. Le moteur pas à pas :	15
5.2. Moteur DC :	16
6. Afficheur LCD :	18
6.1. Définition :	18
6.2. Fonctionnement.....	18
6.3. Branchement avec ARDUINO :.....	19
7. Le clavier	19
7.1. Définition	19
7.2. Fonctionnement :.....	19
7.3. Branchement avec ARDUINO :.....	20
8. Les systèmes de communication.....	20
8.1. RS-232	20
8.2. Branchement avec ARDUINO :.....	22
9. Solution proposé&Cahier de charge	22
Conclusion	22

Chapitre 2 :Etude technologique de la solution proposée	2
1. Conception générale de notre solution.....	24
2. Alimentation stabilisée	25
2.1. Les étapes de stabilisé une tension	25
2.2. Composition de carte d'alimentation	26
3. Propositions des Travaux Pratiques	27
3.1. TP N°1 Conditionneur d'un Capteur Résistif.....	27
3.2. TP N°2 Régulation de température	34
3.3. TP N°3 variation de vitesse d'un moteur pas à pas.....	39
3.4. TP N°4 commande d'un moteur DC d'après Bluetooth / contrôle de buzzer d'après un clavier	45
4. Etude théorique de connexion de RS232	48
Conclusion	48
Chapitre 3 : Conception et Réalisation de la maquette	49
Introduction.....	50
1. Les logiciels utilisés dans la simulation et la conception	50
1.1. Proteus	50
1.2. LabVIEW.....	50
1.3. ARDUINO	50
1.4. GALAAD	51
2. Réalisation de la carte d'alimentation Apres avoir étudié notre besoin des TP, nous avons conclu qu'on aura besoin de quatre sources de tension continu.....	52
3. Réalisation de carte de communication série RS232 :.....	54
4. Réalisation de carte de commande moteur pas à pas	55
5. Conception du boitier	56
6. Réalisation des interfaces des TP sur labVIEW :	58
Conclusion	59
<i>Conclusion Générale Et Perspective.....</i>	<i>60</i>

Figure 1 : Carte ARDUINO	3
Figure 2 : Carte ARDUINO UNO	5
Figure 3 : carte ARDUINO MEGA	6
Figure 4 : le logiciel ARDUINO	6
Figure 5 : domaine d'utilisation	8
Figure 6 : Face avant	9
Figure 7 : Diagramme	10
Figure 8 : Icône et connecteur	10
Figure 9 : Palette de commandes	10
Figure 10 : Toutes les commandes	11
Figure 11 : Palette fonctions	11
Figure 12 : Toutes les fonctions	11
Figure 13 : Notion de capteur	12
Figure 14 : capteur de température LM35	12
Figure 15 : connexion de LM35	13
Figure 16 : capteur PT100	13
Figure 17 : branchement de PT100	14
Figure 18 : conditionneur de capteur passif	15
Figure 19 : conditionneur de capteur actif	15
Figure 20 : moteur pas à pas	16
Figure 21 : branchement de moteur pas à pas	16
Figure 22 : moteur DC	17
Figure 23 : branchement moteur DC	17
Figure 24 : schéma de l'afficheur LCD	18
Figure 25 : branchement d'afficheur LCD	19
Figure 26 : clavier alphanumérique	20
Figure 27 : branchement de clavier	20
Figure 28 : RS- 232	21
Figure 29 : liaison RS-232 et MAX232.	21
Figure 30 : branchement RS-232.	22
Figure 31 : Schéma synoptique du système	24
Figure 32 : les étapes de stabilisation d'une tension	25
Figure 33 : schéma de la carte d'alimentation sous ISIS	27
Figure 34 : Montage potentiométrique	28
Figure 35 : Simulation du montage potentiométrique sous ISIS	29
Figure 36 : Montage en pont	29
Figure 37 : Schéma sur ISIS d'un conditionneur	31
Figure 38 : Configuration INA 122	32
Figure 39 : DATASHEET CAPTEUR INA 122	32
Figure 40 : Schéma Synoptique du Montage pt100	33
Figure 41 : Schéma complet d'étude de capteur PT100	34
Figure 42 : schéma de principe	35
Figure 43 : LM35	35
Figure 44 : Montage ISIS de capteur LM35 avec Fer à souder	38
Figure 45 : Montage ISIS de capteur LM35 sans Fer à souder	38
Figure 46 : schéma synoptique de variateur de vitesse	39
Figure 47 : Pin configuration	40
Figure 48 : configuration pins	40
Figure 49 : montage de NE555 sur ISIS	41
Figure 50 : visualisation du signal de sortie sur un oscilloscope	42
Figure 51 : circuit 1297	42
Figure 52 : constitution de L297	43

Liste des figures

Figure 53 : circuit L298.....	44
Figure 54 : schéma interne de L298.....	44
Figure 55 : Schéma sous ISIS d'un variateur de vitesse.....	45
Figure 56 : Schéma synoptique.....	45
Figure 57:schéma de câblage du Bluetooth	46
Figure 58 : schéma synoptique de BUZZER	47
Figure 59:Schéma de branchement de clavier	48
Figure 60 : liaison série RS232	48
Figure 61:Le processus complet de fabrication.....	51
Figure 62:Schéma de routage de la carte d'alimentation.....	52
Figure 63 : Photo réelle de carte d'alimentation	53
Figure 64:Schéma de routage de carte RS232	54
Figure 65 : Photo réelle de la carte RS232.....	54
Figure 66 : photo réelle de la carte de commande d'un variateur de vitesse	55
Figure 67:model de valise sur Solide Works	56
Figure 68 : photo réelle du banc d'essai.....	57
Figure 69 :interface du capteur LM35 sur LabVIEW	58
Figure 70:interface du capteur PT100 sur LabVIEW.....	59
Figure 71:interface du variateur de vitesse d'un moteur pas à pas	59

Introduction générale

Dans nos jours, l'enseignement supérieur devient plus en plus moderne et pratique et ça explique l'importance des séances de TP et leurs matériels didactiques. Le système éducatif est évolutif, et va être de plus en plus pénétré par des applications de mesure et rien ne se fera sans capteur, qui est à la base de mesure. Donc, l'utilisation de ces matériels didactiques à nos universités a des incidences potentielles sur l'apprentissage des étudiants.

Le matériel didactique comprend différents instruments, dont principalement un manuel de l'étudiant et un guide de l'enseignant. Il est un outil de soutien, de médiation et de référence. Il doit également viser le rehaussement culturel et la promotion des valeurs sociétales. Il suit de près le programme ministériel accepté pour un niveau et une matière donnés.

Dans les séances de TP mesure et instrumentation, les étudiants éprouvent des difficultés lors de l'absence de dispositifs électronique didactique et sa pour plusieurs raisons telque le prix de ces matériels est élevé , délai de livraison dont le temps de transmettre est long du constructeur vers l'ISSET avec l'incertitude de leur qualité et performance et en fin l'inadaptation avec les TP dont les matériels peut être incompatible avec les montages à réaliser donc l'inadaptation entre l'équipement des écoles et les notions prescrites est ça à la raison de l'évolution de système éducatif.

Dans ce contexte et dans le cadre de notre projet fin d'étude ,nous avons réalisé une maquette de développement, autour d'une carte d'**ARDUINO**, qui permette de manipuler les applications de base et de réaliser des études des systèmes électroniques et l'outil **labVIEW** comme logiciel de supervision qui permet de simuler et tracer les caractéristiques électronique de chaque TP.

Ce rapport de fin d'études est organisé comme suit :

- Le premier chapitre est consacré pour l'état de l'art
- Le deuxième chapitre donne une description détaillée de la solution adoptée pour répondre aux spécifications de notre cahier de charge et en abordant la conception

détaillée de chaque partie de la maquette afin d'obtenir une schématisation complète et précise.

- Le troisième chapitre est dédié pour la réalisation pratique, qui commence par une présentation détaillée de la structure électronique et la simulation.

En dernier lieu, une conclusion générale résume le travail réalisé. Elle mettra en lumière les insuffisances des matériels pédagogiques, les améliorations de la maquette et ses apports ainsi que les perspectives d'amélioration de ce projet.

Chapitre1 : Etude de l'art

Introduction

Dans ce premier chapitre, nous allons consacrer à l'étude bibliographie de notre projet et nous allons définir les composants nécessaire dans la conception de la maquette en dégageons finalement notre cahier des charges.

1. Présentation ARDUINO

Dans le domaine électronique fait souvent apparaître des fonctions ayant pour rôle le traitement d'information ces fonctions peuvent être réalisées par des circuits analogiques ou logiques.

Mais, lorsque l'objet technique devient complexe, et qu'il est alors nécessaire de réaliser un ensemble important de traitements d'informations, il devient plus simple de faire appel à une carte de traitement ARDUINO et son logiciel de programmation.

1.1. Matériel

ARDUINO, et son récent synonyme Genuino3, est une marque qui couvre des cartes matériellement libres sur lesquelles se trouve un microcontrôleur (d'architecture Atmel AVR comme l'Atmega328p, et d'architecture ARM comme le Cortex-M3 pour l'Arduino Due). Les schémas de ces cartes sont publiés en licence libre.

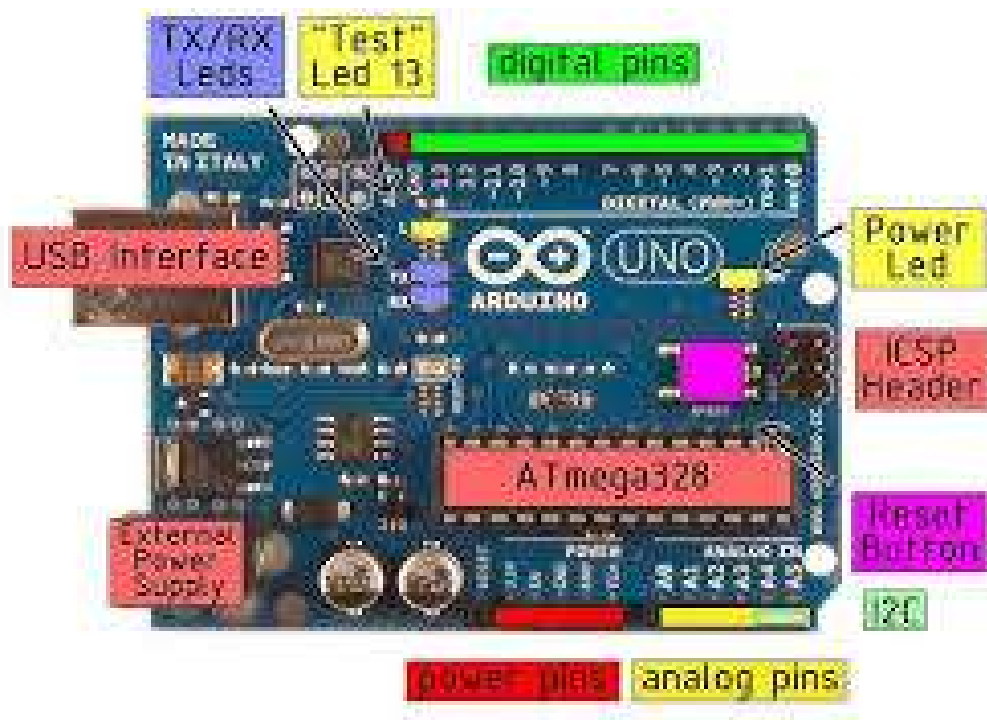


Figure 1 : Carte ARDUINO

Les différentes versions des ARDUINO fonctionnent sous le même principe général :

- En vert : ce sont les broches dites numériques (0 ou 1) ou « tout ou rien », elles offrent en sortie du 5 V et acceptent en entrée du 5 V sur le même principe.
 - fonctions **digitalWrite()** et **digitalRead()** et pour les ports PWM **analogWrite ()**.
- En jaune : ce sont les broches dites analogiques, valeur entre 0 V et 5 V
 - Fonction **analogRead ()**.
 - 40mA maximum sur une même broche et 200mA au total sur toute les broches numérique et analogique.

- En rouge: les différentes broches d'alimentation :

Il y a des variations entre les différentes cartes (par exemple : UNO, la patte 13 est équipée d'une résistance).

Certaines broches ont des fonctionnalités spéciales :

- D0 et D1 : port série TTL,
- D10, D11, D12, et D13 : bus / port SPI,
- A4 et A5 : bus I2C

1.1.1. Les types de cartes

Il y a trois types de cartes :

- Les dites « officielles » qui sont fabriquées en Italie par le fabricant officiel : Smart Projects
- Les dites « compatibles » qui ne sont pas fabriqués par Smart Projects, mais qui sont totalement compatibles avec les AARDUINO officielles.
- Les « autres » fabriquées par diverse entreprise et commercialisées sous un nom différent (Freeduino, Seeduino, etc.)

Tableau 1: comparaison entre les différentes cartes Arduino

Tableau comparatif des différentes cartes Arduino											
Cartes Arduino	UNO R3 (classique & CMS)	UNO R3 Ethernet (classique & POE)	Leonardo	Mega 2560	Mega ADK	DUE	Esplora	Mini	Nano	Yun (classique & POE)	Zero PRO
Caractéristiques											
Microcontrôleur	ATmega328P	ATmega328P	ATmega32u4	ATmega2560	ATmega2560	AT91SAM3X8E	ATmega32u4	ATmega328P	ATmega328P	ATmega32u4	ATSAMD21G18
Cadencement Horloge	16 MHz	16 MHz	16 MHz	16 MHz	16 MHz	84 MHz	16 MHz	16 MHz	16 MHz	16 MHz	48 MHz
Tension d'entrée	7 - 12V	7 - 12V	7 - 12V	7 - 12V	7 - 12V	7 - 12V	7 - 12V	7 - 9V	7 - 9V	5V	5V
Tension de fonctionnement	5V	5V	5V	5V	5V	3.3V	5V	5V	5V	5V	3.3V
Entrée/Sortie Numérique	14/6	14/4	20/7	54/15	54/15	54/12	⊗	14/6	14/6	20/7	14/12
Entrée-Sortie (PWM) Analogique	6/0	6/0	12/0	16/0	16/0	12/2 (DAC)	⊗	8/0	8/0	12/0	6/1 (DAC)
Mémoire vive (Flash)	32 Ko	32 Ko	32 Ko	256 Ko	256 Ko	512 Ko	32 Ko	32 Ko	32 Ko	32 Ko	256 Ko
Mémoire vive (SRAM)	2 Ko	2 Ko	2,5 Ko	8 Ko	8 Ko	96 Ko	2,5 Ko	2 Ko	2 Ko	2,5 Ko	32 Ko
Mémoire morte (EEPROM)	1 Ko	1 Ko	1 Ko	4 Ko	4 Ko	⊗	1 Ko	1 Ko	1 Ko	1 Ko	16 Ko
Interface USB	USB-B mâle	USB-B mâle	Micro-USB	USB-B mâle	USB-B mâle & USB-A pour Android	2 ports micro- USB (Native et programming)	Micro-USB	⊗	Mini-USB	Micro-USB	2 ports micro- USB (Native et programming)
Port UART	1	1	1	4	4	4	⊗	⊗	1	1	2
Carte SD	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗
Ethernet	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗
Wi-Fi	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗
Dimensions	68x53mm	68x53mm	68x53mm	101x53mm	101x53mm	101x53mm	165x60mm	30x18mm	45x18mm	68x53mm	68x53mm

Les plus utilisés sont :

- La carte ARDUINO UNO



Figure 2 : Carte ARDUINO UNO

▪ La carte *MEGA*

La carte ARDUINO MEGA est une autre carte qui offre toutes les fonctionnalités des précédentes, mais avec des options en plus. On retrouve notamment un nombre d'entrées et de sorties plus importantes ainsi que plusieurs liaisons séries.

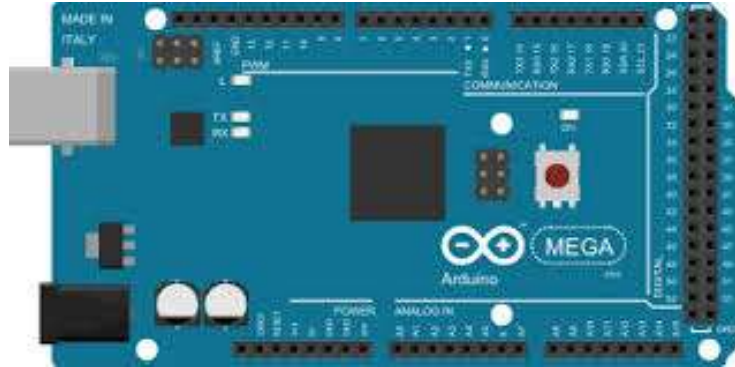


Figure 3 :carte ARDUINO MEGA

1.2. Logiciel

Le logiciel va nous permettre de programmer la carte ARDUINO. Il nous offre une multitude de fonctionnalités. Le langage de programmation utilisé est le C++, compilé avec avr-g++ 7, et lié à la bibliothèque de développement ARDUINO, permettant l'utilisation de la carte et de ses entrées/sorties. La mise en place de ce langage standard rend aisé le développement de programmes sur les plates-formes ARDUINO, à toute personne maîtrisant le C ou le C++.Voilà à quoi il ressemble :

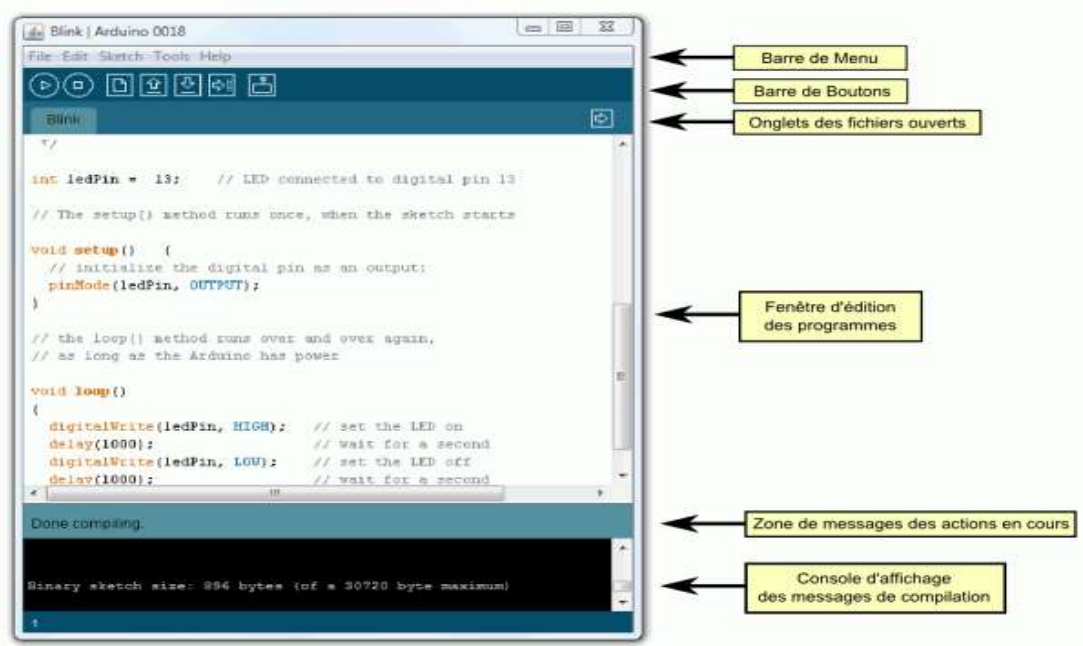


Figure 4 :le logiciel ARDUINO

1.3. La raison de choisir ARDUINO

- **Le prix**

En vue des performances qu'elles offrent, les cartes ARDUINO sont relativement peu coûteuses, le prix d'un ARDUINO officiel est d'environ 50 DT.

- **La liberté**

C'est un bien grand mot, mais elle définit de façon assez concise l'esprit de l'ARDUINO. Elle constitue en elle-même deux choses :

- Le logiciel : gratuit et open source, développé en **Java**, dont la simplicité d'utilisation relève du savoir cliquer sur la souris.
- Le matériel : cartes électroniques dont les schémas sont en libre circulation sur internet.

- **La compatibilité**

Le logiciel, tout comme la carte, est compatible sous les plateformes les plus courantes (Windows, Linux et Mac), contrairement aux autres outils de programmation du commerce qui ne sont, en général, compatibles qu'avec Windows.

La communauté ARDUINO est impressionnante et le nombre de ressources à son sujet est en constante évolution sur internet. De plus, on trouve les références du langage ARDUINO ainsi qu'une page complète de tutoriels sur le site arduino.cc.

- **leurs applications**

Le système ARDUINO nous permet de réaliser une multitude des projets, qui ont une application dans tous les domaines, l'étendue de l'utilisation de l'ARDUINO est gigantesque.

Des exemples pour leur utilisation :

- contrôler les appareils domestiques
- fabriquer votre propre robot.
- faire un jeu de lumières.
- communiquer avec l'ordinateur.
- télécommander un appareil mobile (modélisme).
- Acquisition et contrôle des données, c'est notre cas.
- etc.

Et pour ça, nous allons utiliser labVIEW pour l'interfaçage des TP sur PC.

2. Le logiciel d'interfaçage labVIEW

2.1. Définition

LabVIEW (Laboratory Virtual Instrument Engineering Workbench) est un environnement de développement graphique qui permet de créer des applications modulaires (notion de sous VI) et extensibles pour la conception d'applications, le contrôle et le test.

2.2. Domaine d'utilisation

LabVIEW est un outil d'acquisition, d'analyse et de présentation de données.

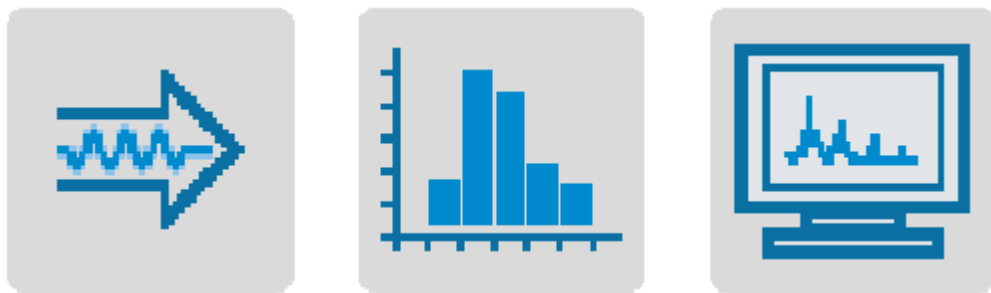


Figure 5 : domaine d'utilisation

2.2.1. Acquisition

LabVIEW permet l'acquisition de données par l'intermédiaire de diverses connectiques :

- PCI (Peripheral Component Interconnect)
- Compact Flash
- LAN (Local Area Network)
- etc,

2.2.2. Analyse

LabVIEW inclut des outils pour l'analyse des données :

- Traitement du signal : Convolution, analyse spectrale, transformées de Fourier,...
- Traitement d'images : Masque, détection de contours, profils, manipulations de pixels,...
- Mathématiques : Interpolation, statistiques (moyennes, écart-type,...), équations
- différentielles, etc.

2.2.3. Présentation des données

LabVIEW inclut des outils d'aide à la présentation (communication)des données :

- Graphiques, tableaux, images, génération de rapport,...
- Par l'intermédiaire d'Internet : outils de publication web, serveur data socket, TCP/IP, envoi d'alertes par email,...

2.2. Le composant d'un VI

Les programmes LabVIEW appelés Instruments Virtuels ou Virtual Instruments (VI).

On parle d'instruments virtuels car leur apparence et leur fonctionnement sont semblables à ceux d'instruments réels, tels que les oscilloscopes et les multimètres.

Les VI se composent de 3 éléments principaux :

▪ Face avant

Dans la face-avant nous trouverons tous les éléments interactifs du VI (commandes et indicateurs).

C'est une interface utilisateur contenant des Contrôles = entrées et des Indicateurs = sorties.

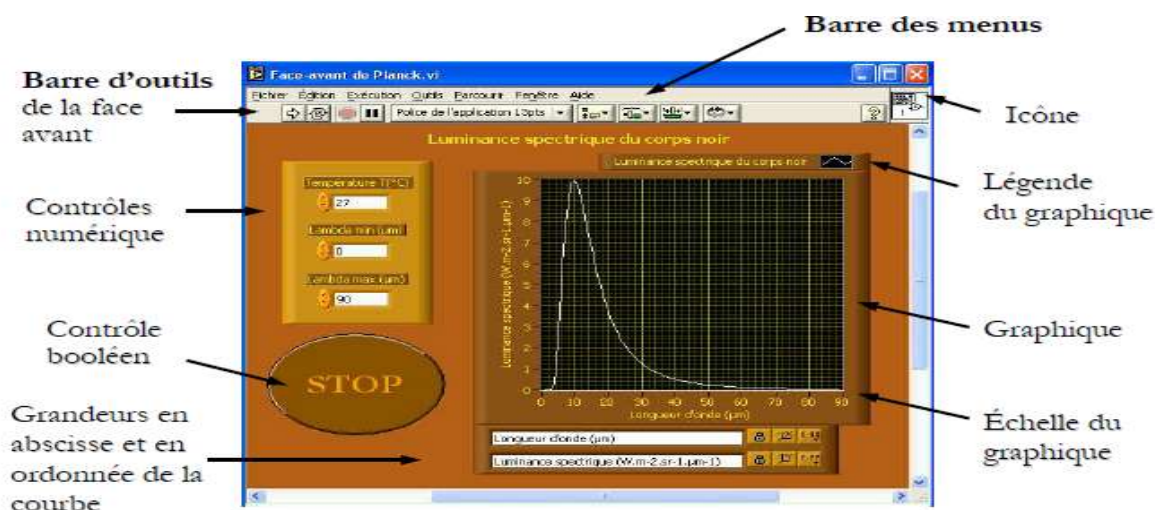


Figure 6 : Face avant

▪ Diagramme

Dans le diagramme nous trouverons tous les éléments propres au code développé (structures, fonctions, constantes, etc.).

- Fenêtre de programmation et d'affichage du code source.
- Interaction entre face avant et diagramme (Ctrl+E).

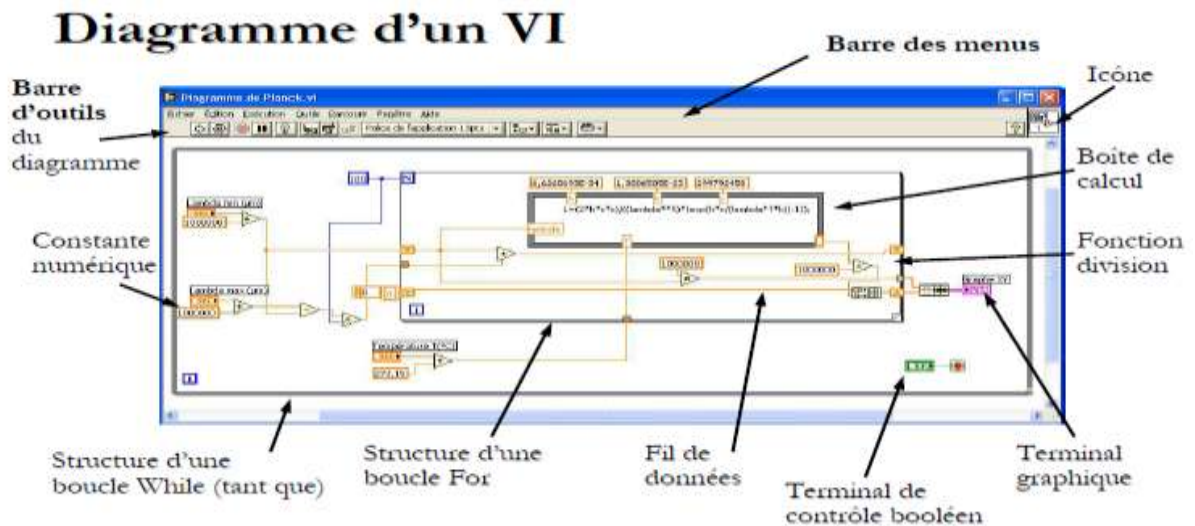


Figure 7 :Diagramme

▪ Icône/connecteur d'un VI

- Chaque VI affiche une icône, dans le coin supérieur droit des fenêtres de la face-avant et du diagramme.
- Une icône est la représentation graphique d'un VI.
- Un connecteur est un ensemble de terminaux correspondant aux commandes et aux indicateurs du VI qui sont accessibles.



Figure 8 :Icône et connecteur

2.3. Les palettes

2.3.1. Palette de commandes

Palette de commandes (disponible à partir de la fenêtre face avant par un clic droit avec la souris ou dans la barre des menus : “ Fenêtre”).

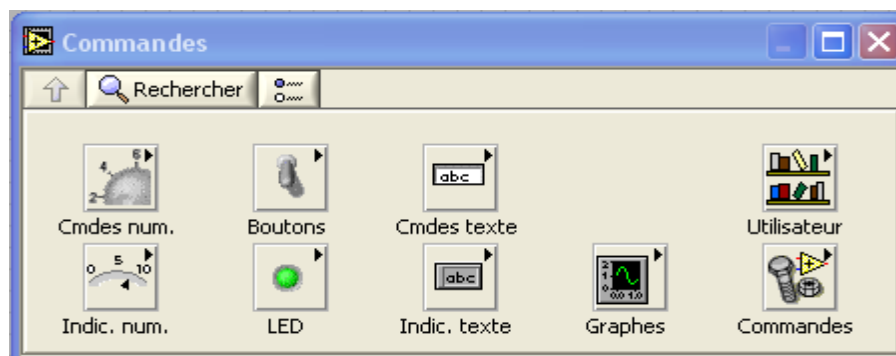


Figure 9:Palette de commandes

Dans cette palette nous trouverons tous les éléments(commands et indicateurs) nécessaires à la création de la face-avant.

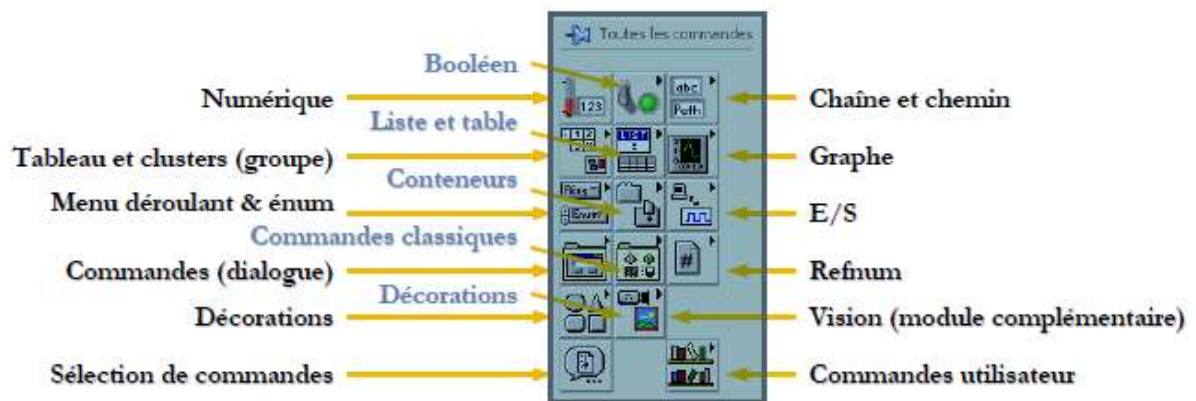


Figure 10:Toutes les commandes

2.3.2. Palette de fonctions

Palette de fonctions(disponible à partir de la fenêtre diagramme par un clic droit avec la Souris ou dans la barre des menus : “ Fenêtre”).

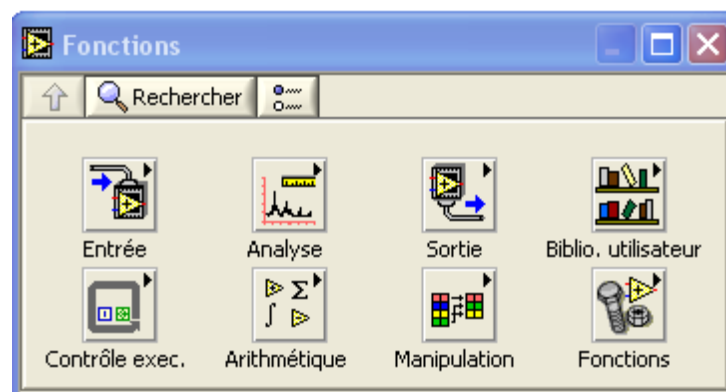


Figure 11:Palette fonctions

Dans cette palette nous trouverons tous les éléments(fonctions de base, VI Express, etc) nécessaires à la création du code graphique dans la fenêtre diagramme.

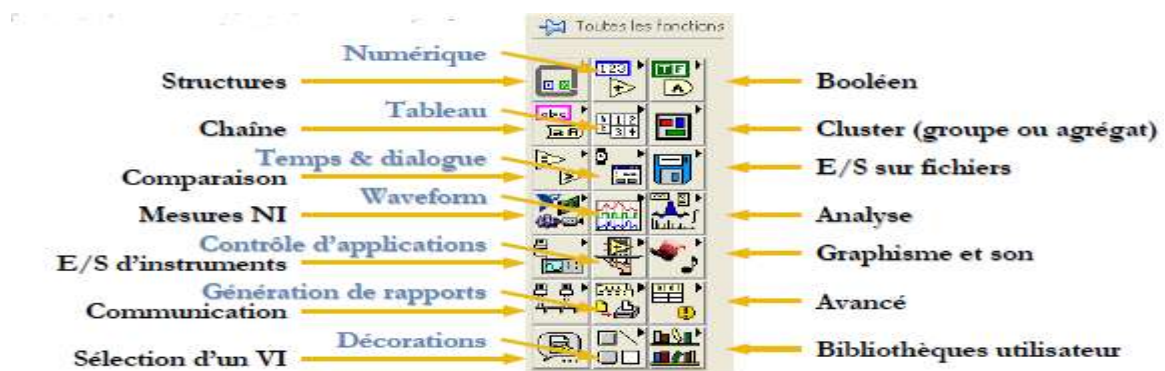


Figure 12:Toutes les fonctions

3. Les capteurs

Le capteur est un composant très important dans la chaîne de mesure et voilà les principaux étapes de leur mesure :



Figure 13:Notion de capteur

3.1. Le capteur de température LM35

3.1.1. Définition

Le LM35 est un capteur de température qui produit une tension qui varie en fonction de la température. La variation de la tension est très faible, c'est pourquoi nous utilisons un amplificateur opérationnel. Sa représentation graphique est une droite linéaire



Figure 14 : capteur de température LM35

3.1.2. Fonctionnement

Le LM35 est un capteur de température de précision qui peut être facilement étalonné. Il fonctionne comme un Zener 2-terminale et la tension de claquage est directement proportionnelle à la température absolue à $10\text{mV} / ^\circ\text{K}$. Il fournit 0 V pour 0 degré, 250 mV pour 25 degrés, etc. et un maximum de 1 volt pour 100 degrés.

Le circuit a une impédance dynamique inférieure à 1Ω et fonctionne dans une plage de courant de $450\mu\text{A}$ à 5mA sans altération de ses caractéristiques.

3.1.3. Connexion avec ARDUINO

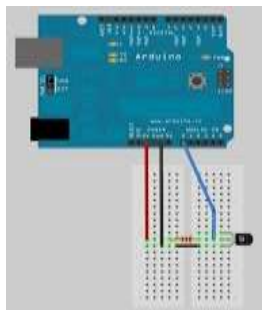


Figure 15 :connexion de LM35

3.2. Le capteur de température PT100

3.2.1. Définition

La sonde Pt 100 est un capteur de température qui est utilisé dans le domaine industriel (agroalimentaire, chimie, raffinerie...). Ce capteur est constitué d'une résistance en Platine. La valeur initiale du Pt100 est de 100 ohms correspondant à une température de 0°C .



Figure 16 :capteur PT100

3.2.2. Fonctionnement

La sonde PT100 permet de mesurer le changement de résistivité d'un filament de platine enroulé autour d'une tige de verre. La variation de résistivité est environ de $0.5\text{ ohms}/^{\circ}\text{C}$, avec une précision de $\pm 0.3^{\circ}$.

La caractéristique de la Sonde Pt100 est linéaire : Celle-ci va de -200°C à 800°C nous pouvons dire que la résistivité (en Ohm) augmente avec la Température.

3.2.3. Branchement PT100 avec ARDUINO

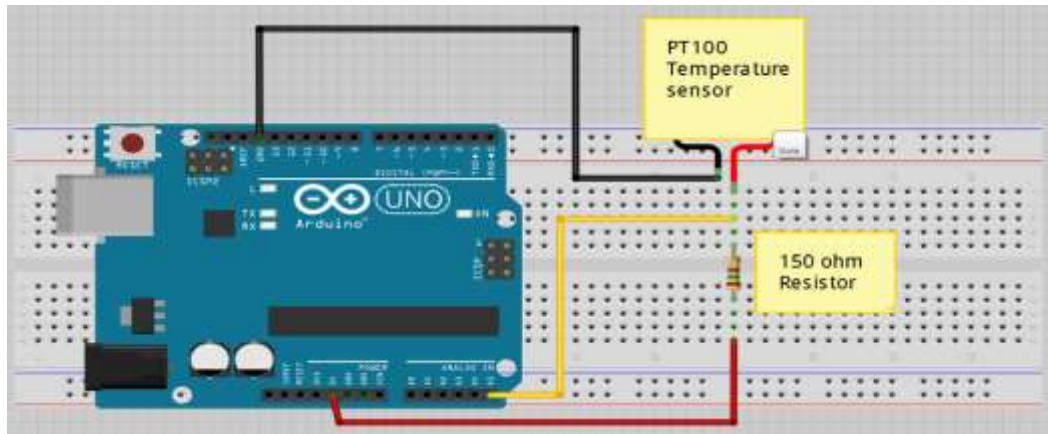


Figure 17:branchement de PT100.

4. Les conditionneurs des capteurs passif et actif

4.1. Définition

Un conditionneur est un circuit électronique de traitement du signal issu d'un capteur. Il est utilisé pour simplifier l'interface du capteur avec le système d'acquisition de données.

Dans une chaine de mesure, le conditionneur intervient entre le capteur et l'interface. Il met en forme le signal mesuré pour le traduire en une grandeur permettant le traiter par ARDUINO (par exemple : tension, courant, fréquence).

4.2. Conditionneur de capteur passif

Si le capteur est résistif (la grandeur physique fait évoluer une valeur de résistance), on pourra utiliser un pont de résistances. Si le capteur est réactif (la grandeur physique fait évoluer une valeur d'impédance), on pourra avoir recours à un montage potentiométrique ou un oscillateur. Voilà un exemple :

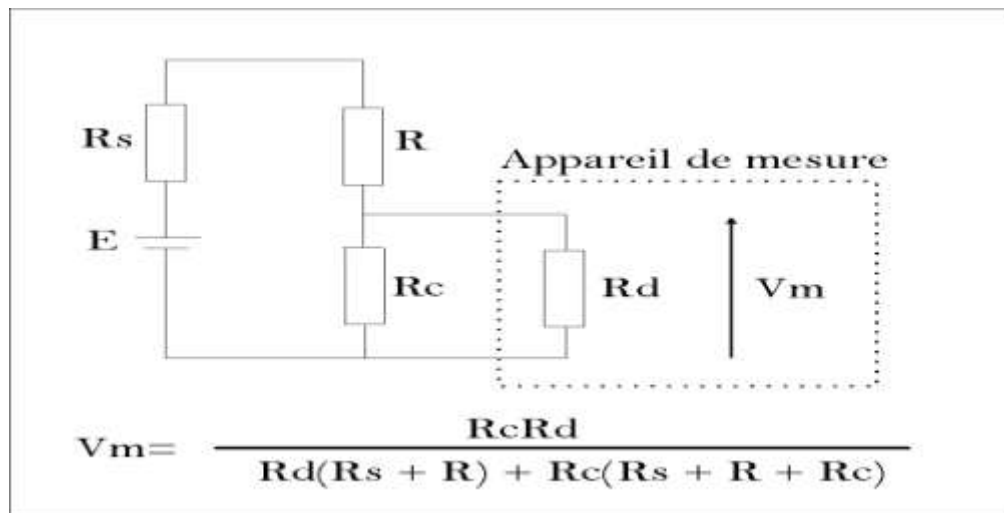


Figure 18:conditionneur de capteur passif

4.3. Conditionneur de capteur actif :

Généralement nous utilisons les conditionneurs pour convertir le courant ou la charge délivré par le capteur en une tension adapté avec l'unité de contrôle.

Il faudra généralement amplifier le signal avec un amplificateur simple ou un amplificateur d'instrumentation selon les cas. Voilà un exemple :

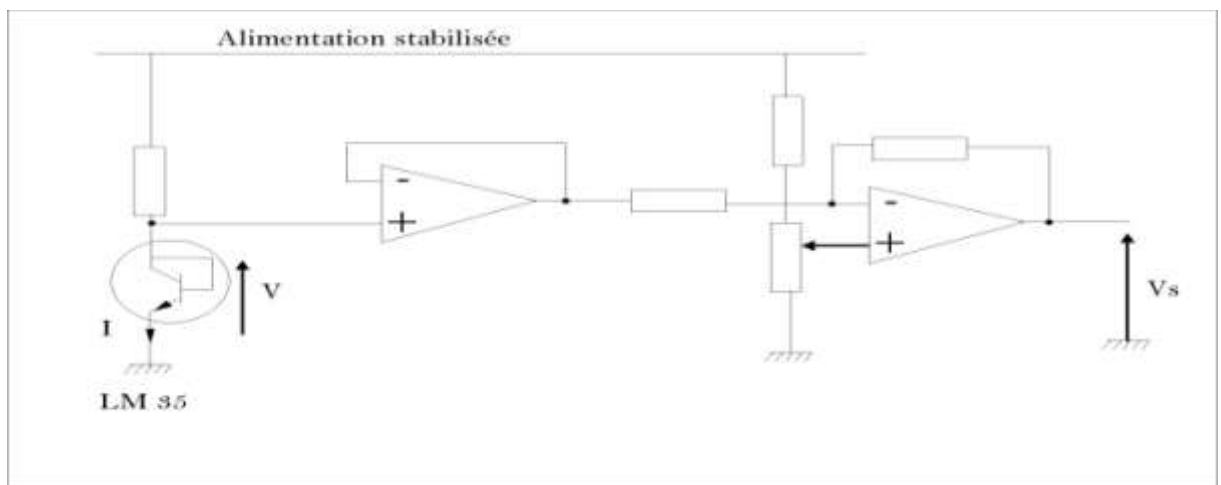


Figure 19:conditionneur de capteur actif.

5. Les moteurs :

5.1. Le moteur pas à pas :

5.1.1. Définition :

Un moteur pas à pas permet de transformer une impulsion électrique en un mouvement angulaire.

L'angle minimal entre deux modifications des impulsions électrique s'appelle un pas. On caractérise un moteur par le nombre de pas par tour.



Figure 20:moteur pas à pas

5.1.2. Fonctionnement :

Contrairement aux moteurs à courant continu, ils ne nécessitent pas de boucle d'asservissement et sont plus simples à commander. il existe trois types de moteur : à aimant permanent, à reluctance variable et hybrides.

Dans tous les types de moteur, on positionne le rotor en modifiant la direction d'un champ magnétique créé par les bobinages du stator. Ils nécessitent un circuit de commande qui comporte une partie logique et une commande de puissance. la partie logique détermine pour chaque pas quelles sont les bobines alimentées et le sens de rotation.

La fréquence de l'horloge du circuit logique détermine la vitesse de rotation.

5.1.3. Branchement avec ARDUINO :

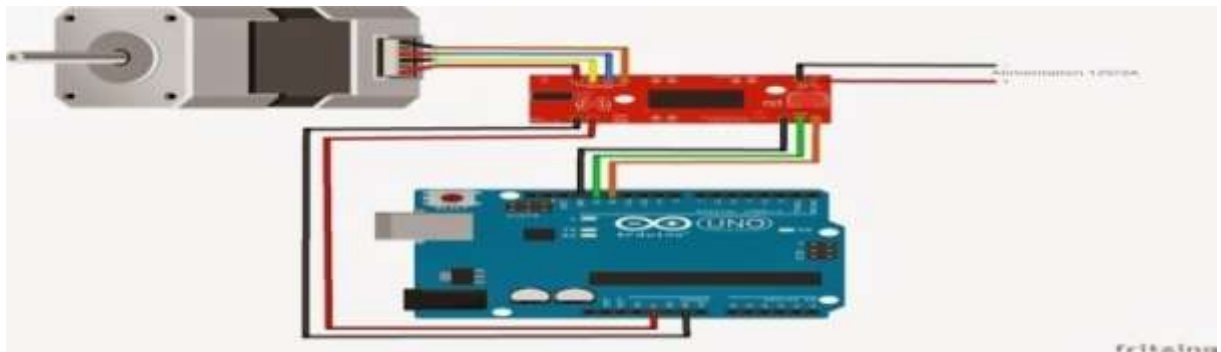


Figure 21:branchement de moteur pas à pas

5.2. Moteur DC :

5.2.1. Définition :

Le moteur à courant continu est une machine électrique transforme l'énergie électrique en énergie mécanique, il se compose de trois parties :

- de l'inducteur ou du stator.
- de l'induit ou du rotor.

- du collecteur et des balais.



Figure 22:moteur DC

5.2.2. *Fonctionnement*

Lorsque le bobinage d'un inducteur de moteur est alimenté par un courant continu, sur le même principe qu'un moteur à aimant permanent, il crée un champ magnétique de direction Nord-Sud.

Une spire capable de tourner sur un axe de rotation est placée dans le champ magnétique. De plus, les deux conducteurs formant la spire sont chacun raccordés électriquement à un demi collecteur et alimentés en courant continu via deux balais frotteur.

D'après la loi de Laplace les conducteurs de l'induit placés de part et d'autre de l'axe des balais sont soumis à des forces F égales mais de sens opposé en créant un couple moteur : l'induit se met à tourner.

5.2.3. *Branchement avec ARDUINO :*

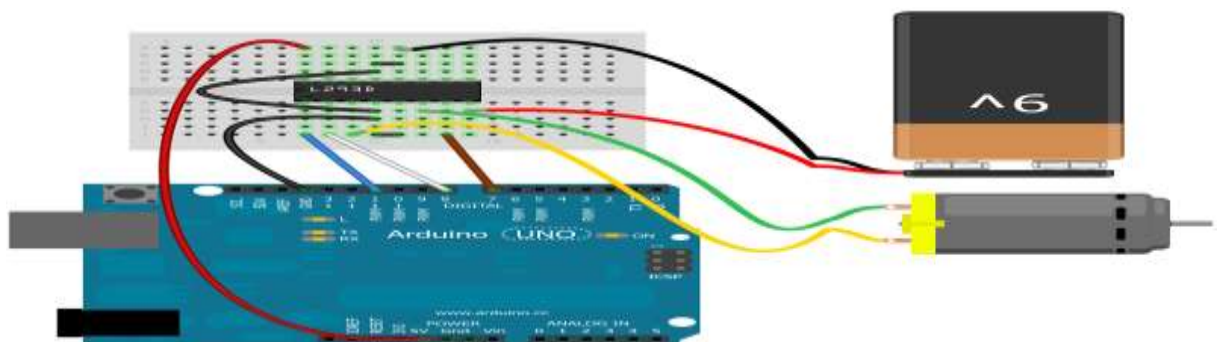


Figure 23:branchement moteur DC

6. Afficheur LCD :

6.1. Définition :

Afficheurs LCD (Liquid Crystal Display) sont des modules compacts intelligents et nécessitent peu de composants externes pour un bon fonctionnement.

Il est constitué de deux lames de verre, distantes de 20 μm environ, sur lesquelles sont dessinées les mantisses formant les caractères. L'application entre les deux faces d'une tension alternative basse fréquence de quelques volts (3 à 5 V) le rend absorbant. Un afficheur à cristaux liquides réfléchitif ne peut être utilisé qu'avec un bon éclairage ambiant. Sa lisibilité augmente avec l'éclairage.

6.2. Fonctionnement

Dans notre projet on va utiliser le mode 4 bits de l'afficheur LCD. Dans ce mode, seul les 4 bits de poids fort (D4 à D7) de l'afficheur sont utilisés pour transmettre les données et les lires. Les 4 bits de poids faible (D0 à D3) sont alors connectés à la masse.

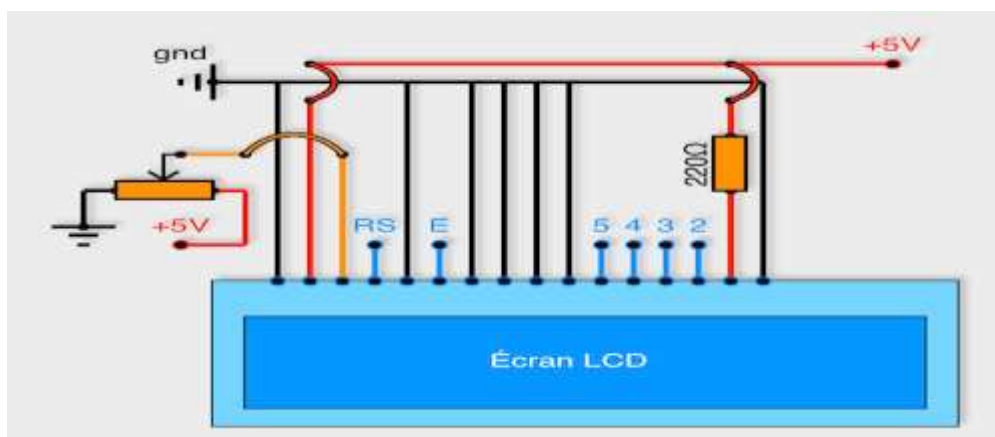


Figure 24:schéma de l'afficheur LCD

Nous trouvons une série de 14 broches aux rôles Suivantes :

- Broche 1 : masse.
- Broche 2 : Alimentation positive +5V.
- Broche 3 : luminosité.
- Broche 4 : Grâce à cette broche, l'afficheur est capable de faire la différence entre une commande et une donnée. Un niveau bas indique une commande et un niveau haut indique une donnée.
- Broche 5 : R/W : sélection du mode lecture ou écriture (Read/Write).

- Broche 6 : E : Entrée de validation (Enable) active sur front descendant,
- Broche 7-14 :(D0-D7) : utilisées pour le transfert des données ou des instructions.

Le transfert peut se faire sur 8 bits, toutes les broches sont alors utilisées, ou sur 4 bits, dans ce cas, seules les broches 11 à 14 sont utilisées.

6.3. Branchement avec ARDUINO :

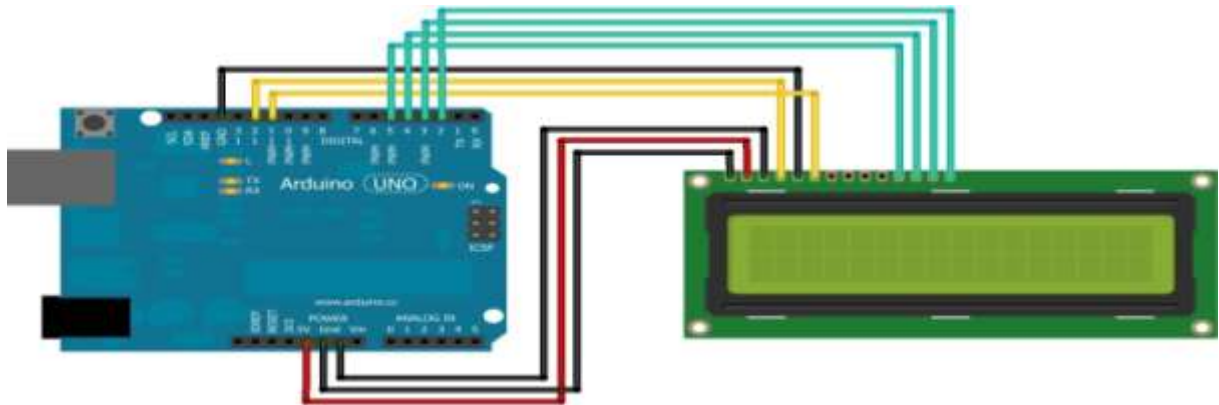


Figure 25:branchement d'afficheur LCD

7. Le clavier

7.1. Définition

Le clavier est un périphérique d'entrée le plus commode pour saisir du texte, dans notre carte on va utiliser un clavier alphanumérique à 16 touches matricées pour saisir des numéros et des lettres.

7.2. Fonctionnement :

Le clavier se compose de 16 touches réparties sur une matrice de 4 lignes 4 colonnes .le touche agit comme un bouton poussoir qui établit le contact entre une des 4 pistes horizontales et une des 4 pistes verticales.

Chaque touche est normalement en position ouverte, lorsqu'une touche est appuyée un signal électrique est envoyé vers le codeur, circuit électronique très simple qui associe à chaque signal un code .ce signal peut être utilise pour envoyer une interruption au processeur à fin qu'il traite l'information. Les codeurs réellement utilises assurent de fonctions supplémentaires comme le décodage automatique des touches appuyées longtemps.



Figure 26:clavier alphanumérique

7.3. Branchement avec ARDUINO :



Figure 27:branchement de clavier

8. Les systèmes de communication

8.1. RS-232

8.1.1. Définition

RS-232 est une norme standardisant une voie de communication de type série disponible sur presque tous les PC .il est beaucoup utilisé en industrie pour la communication entre un ordinateur via un port série et un système électronique. Les communications sont en Full duplex, c'est-à-dire capable de fonctionner dans les deux sens, ainsi la carte peut envoyer des informations et en recevoir.



Figure 28:RS- 232

8.1.2. Fonctionnement :

Les cartes électroniques à base de microcontrôleurs fonctionnent très souvent avec des niveaux TTL soit 0-5Volt, 0V pour le niveau 0 et 5Volt pour le niveau 1. On branche directement une ligne RS232 sur un microcontrôleur n'aurait donc aucun sens et pourrait aussi endommager le système en imposant des tensions de 25volt. Pour rendre compatible une ligne RS232 avec une carte de ce type il existe un composant très simple d'utilisation le max232.

Il est un composant créé par MAXIM que l'on trouve sous d'autres références chez d'autres fabricants. Il sert d'interface entre une liaison série TTL (0-5V) et une liaison série RS232 (+12 -12V) et ce avec une simple alimentation 5V.

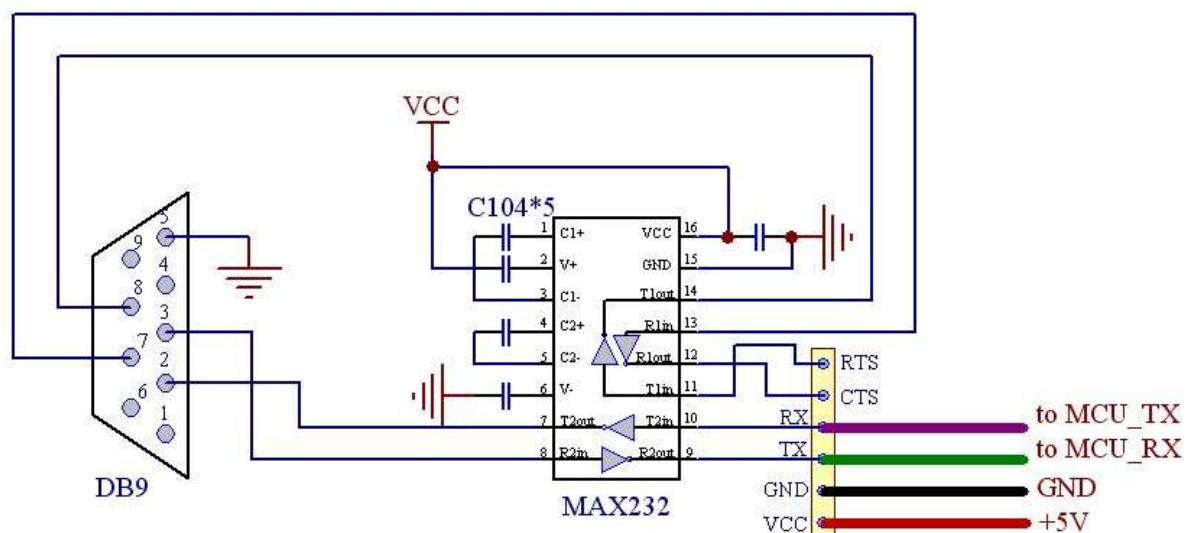


Figure 29:liaison RS-232 et MAX232.

8.2. Branchement avec ARDUINO :

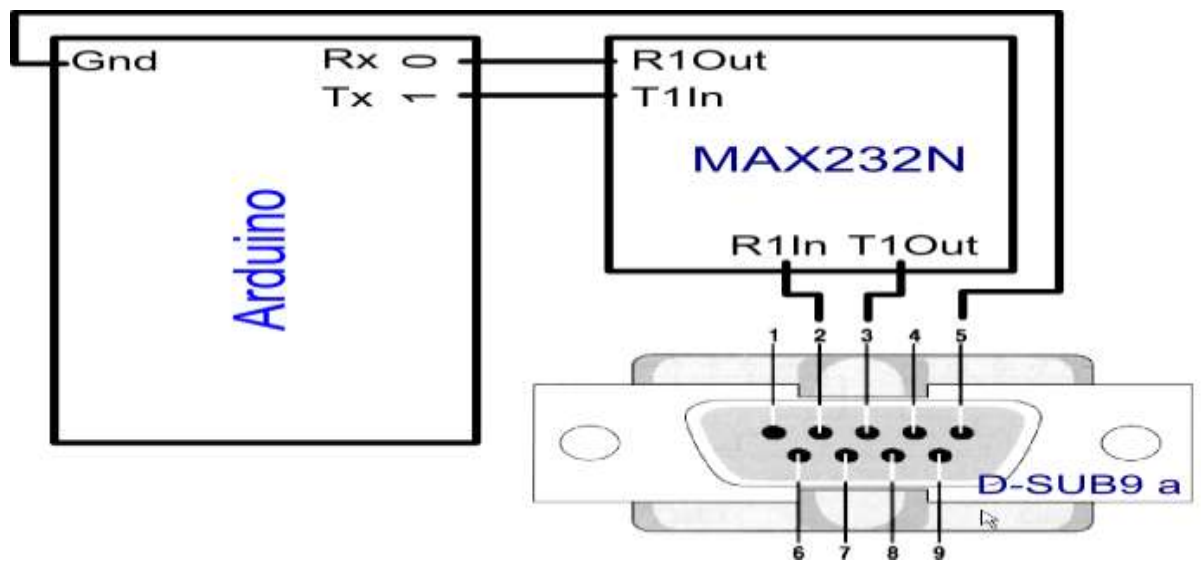


Figure 30:branchement RS-232.

9. Solution proposé&Cahier de charge

Le manque et l'inadaptation des matériels didactiques sont deux principaux problèmes dans les établissements universitaires ce qui nous ont amenés à concevoir et réaliser une maquette didactique à base d'ARDUINO permettant à l'étudiant de simuler, contrôler et mesurer des signaux et tracer ses caractéristiques à l'aide du logiciel de supervision labVIEW.

Travail à faire

- Etude bibliographique
- Etude théorique
- Choix des composants électroniques.
- Simulation de la carte électronique.
- Réalisation pratique
- Supervision par labVIEW
- Rédaction d'un rapport de projet.

Conclusion

Dans ce chapitre nous avons dégagé notre problématique et nous avons rédigé une étude globale sur les composants électronique de base.

Dans le chapitre suivant, nous allons spécifier notre besoin et nous allons proposer une solution.

Chapitre 2 :Etude technologique de la solution proposée

Dans ce chapitre nous présentons une description détaillée de la solution adoptée pour répondre aux spécifications de notre cahier de charge et en abordant la conception détaillée de chaque partie du système afin d'obtenir une schématisation complète et précise.

1. Conception générale de notre solution

Le schéma synoptique du Banc est donné à la figure :

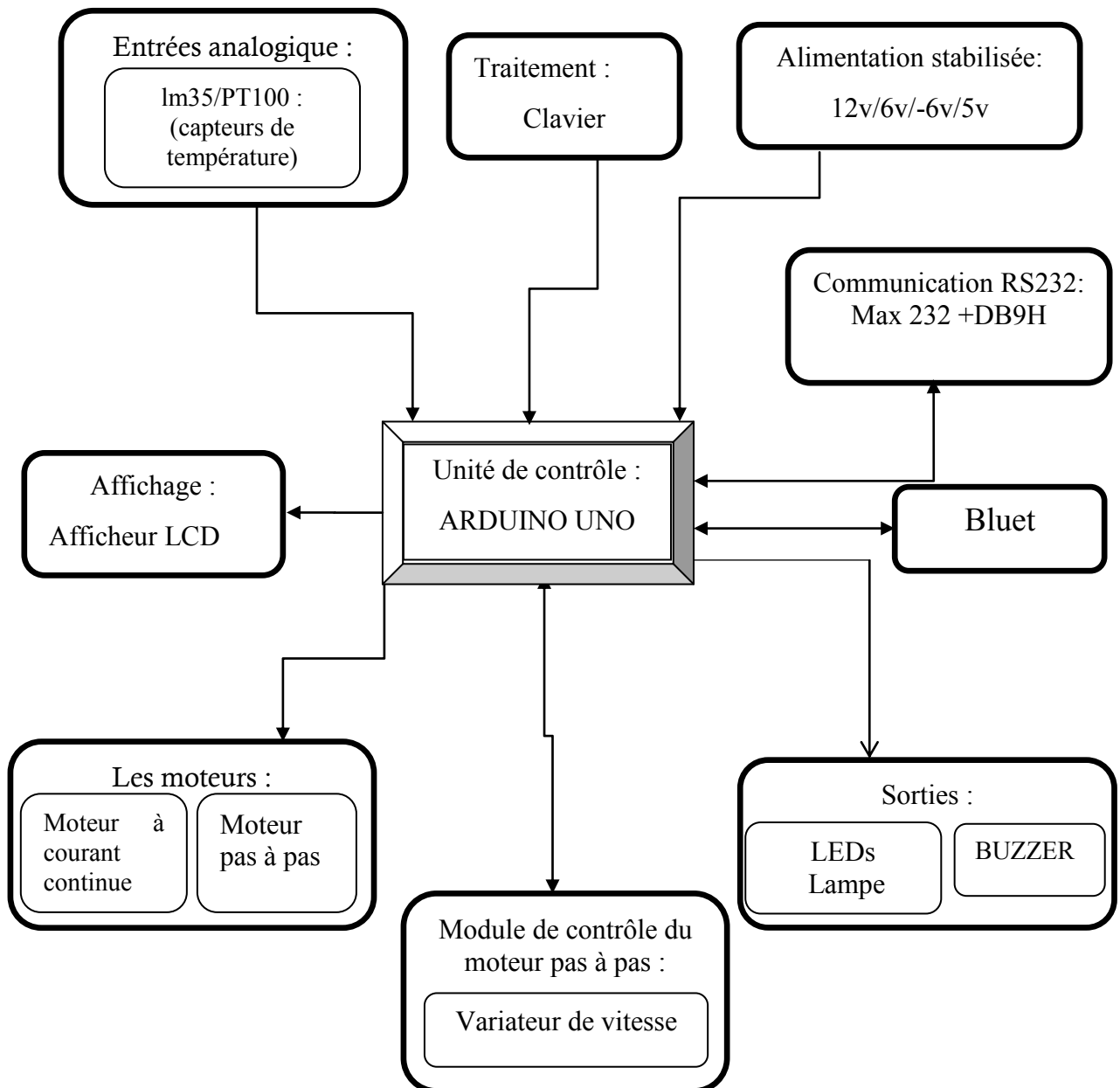


Figure 31:Schéma synoptique du système

Chapitre 2 : Etude technologique de la solution proposée

Réaliser un banc d'essai à base des capteurs et d'une carte ARDUINO UNO pour les travaux pratique « instrumentation », l'étincelle de cette idée, les difficultés rencontrées par les étudiants vue l'absence des dispositifs électroniques didactiques.

La conception de cette maquette s'articule autour de 2 axes :

- Programme officiel des ateliers des capteurs.
- L'utilisation des technologies modernes (carte ARDUINO, supervision par labVIEW, Bluetooth, etc.).

Les blocs sont choisis de façon à satisfaire les applications suivantes :

- le conditionneur du capteur,
- Commande d'un moteur DC à partir d'un capteur pt100
- Régulation de température
- variation de vitesse d'un moteur pas à pas
- commande d'un moteur à l'aide d'une application Android

Enfin, pour la communication entre la maquette et le PC nous avons utilisé la connexion RS232

2. Alimentation stabilisée

2.1. Les étapes de stabilisé une tension

Pour stabiliser une tension nous devons suivre les étapes suivantes :

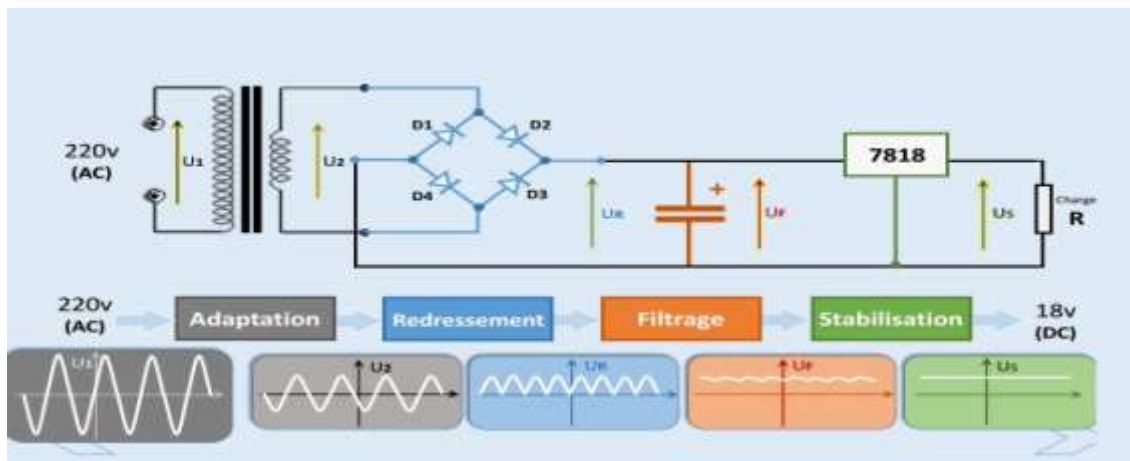


Figure 32: les étapes de stabilisation d'une tension

▪ Adaptation

On utilise un transformateur abaisseur pour obtenir la tension requise.

▪ **Redressement**

Il s'agit d'un redressement à pont de diode, il exige 4 diodes montées en pont qui fonctionnent par groupe de deux pour chaque alternance.

▪ **Filtrage**

Les condensateurs pour le filtrage, dont les valeurs devraient être suffisamment élevées pour un meilleur filtrage.

▪ **Régulation**

La régulation est l'un des critères de stabilité pour avoir un bon fonctionnement de la machine. C'est l'action de maintenir les tensions dans une plage définie pour n'importe quelle puissance demandée par la machine.

2.2. Composition de carte d'alimentation

Notre carte d'alimentation a pour but d'obtenir une tension stabilisée de 12v, 6v, -6v et 5v.

Le transformateur nous fournit une tension de 12v qui est redressé par le pont de diodes et filtré par les condensateurs C3, C5 et C6 ensuite réguler par les transistors Q1 et Q2 pour obtenir les tensions 6v et -6v.

La deuxième partie, la tension 12v est filtré par le condensateur C4 et régulier par la régulateur 7805 pour obtenir la tension 5v.

- La sortie de transformateur 12v pour alimenter le moteur à courant continue, le capteur LM35 et la lampe.
- La sortie de régulateur une tension de 5v pour alimenter ARDUINO UNO, l'afficheur LCD et la communication RS232.
- La sortie +6v/-6v assez stable pour L'AOP, 6v aussi pour alimenter le pont de capteur PT100.

La figure suivante représente la carte d'alimentation :

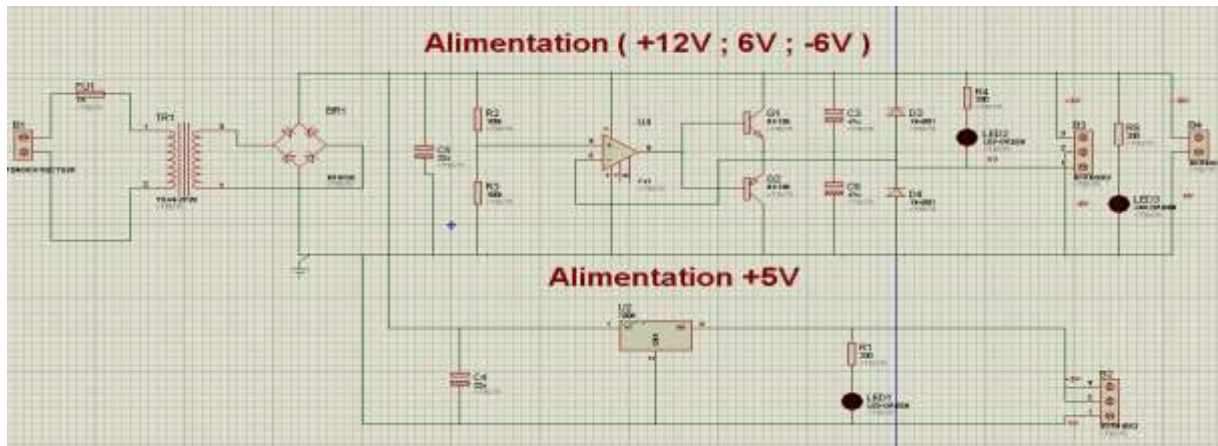


Figure 33:schéma de la carte d'alimentation sous ISIS

3. Propositions des Travaux Pratiques

3.1. TP N°1 Conditionneur d'un Capteur Résistif

Le choix d'un conditionneur est une étape importante dans la réalisation d'un ensemble de mesure. C'est, en effet, l'association capteur + conditionneur qui détermine le signal électrique. De la constitution du conditionneur dépend un certain nombre de performances de l'ensemble de mesure : sa sensibilité, sa linéarité, son insensibilité à certaines grandeurs d'influence, etc.

3.1.1. Les types des Conditionneurs des capteurs passifs

Les types de conditionneurs les plus généralement utilisés sont :

Le montage potentiométrique qui est l'association en série d'une source, du capteur et d'une impédance qui peut être ou non du même type. C'est un montage simple, dont l'inconvénient majeur est sa sensibilité aux parasites

Le pont d'impédances dont l'équilibre permet la détermination de l'impédance du capteur et /ou dont le déséquilibre est une mesure de la variation de cette impédance . c'est donc un double potentiomètre. Le caractère différentiel de la mesure permet de réduire de façon importante l'influence des parasites .

le circuit oscillant ou est inclus l'impédance du capteur qui est en fixe la fréquence.

3.1.2. Montage potentiométrique

Le capteur de résistance R_c en série avec une résistance R_1 est alimenté par une source de résistance interne R_s et de f.é.m. e_s , continue ou alternative. La tension V_m est mesurée aux bornes du capteur par un appareil de résistance d'entrée R_d .

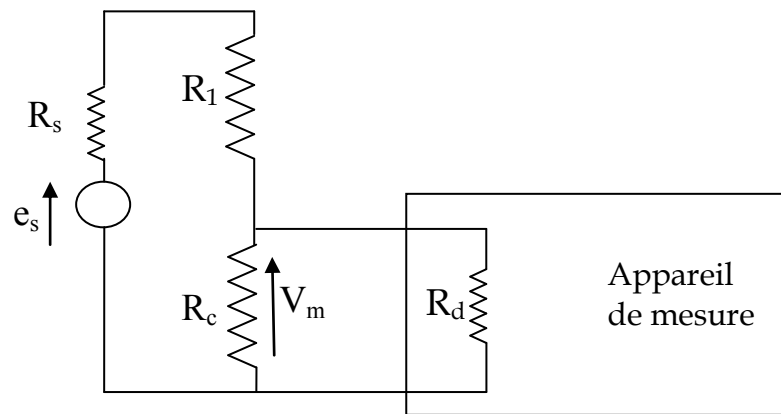


Figure 34: Montage potentiométrique

Selon les lois d'électroniques, on peut écrire :

$$e_s = (R_s + R_1 + R_c) \cdot I$$

$$V_m = R_c \cdot I$$

$$I = \frac{V_m}{R_c}$$

Donc

$$e_s = (R_s + R_1 + R_c) \cdot \frac{V_m}{R_c}$$

$$V_m = \frac{R_c}{R_s + R_1 + R_c} \cdot e_s$$

Comme R_s est négligeable donc on a :

$$V_m = \frac{R_c}{R_1 + R_c} \cdot e_s$$

La relation qui lie la tension de sortie (V_m) au paramètre image du mesurande (R_c) n'est pas linéaire. La sensibilité du montage n'est donc pas constante. On peut néanmoins faire une étude en petites variations du mesurande (étude petit signaux). Ainsi si l'on se place aux petites variations $\Delta R \ll R_c + R_1$:

$$R_c \rightarrow R_{co} + \Delta R$$

$$V_m \rightarrow V_{mo} + \Delta V_m$$

Alors on obtient :

$$\Delta V_m = \frac{\Delta R \cdot R_1}{(R_1 + R_{c0})^2} \cdot e_s$$

C'est une relation linéaire d'où on peut directement extraire la sensibilité du capteur $\Delta V_m / \Delta R_c$. Cette sensibilité est maximum pour $R_1 = R_{c0}$ soit :

$$\Delta V_m = \frac{e_s}{4 \cdot R_1} \cdot \Delta R$$

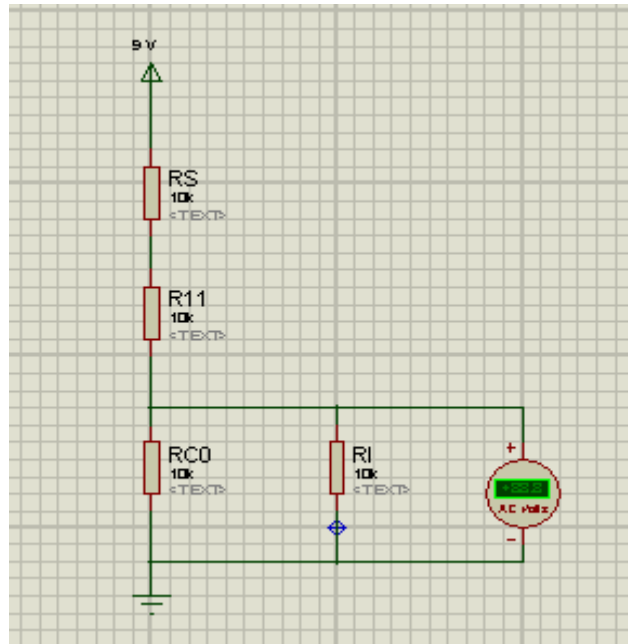


Figure 35: Simulation du montage potentiométrique sous ISIS

3.1.3. Montage en pont

L'utilisation d'un montage potentiométrique présente le défaut d'avoir en sortie la présence d'une tension continue, et ceci en l'absence de variations du mesurande. L'emploi d'un montage en pont présente l'avantage de s'affranchir de cette tension continue.

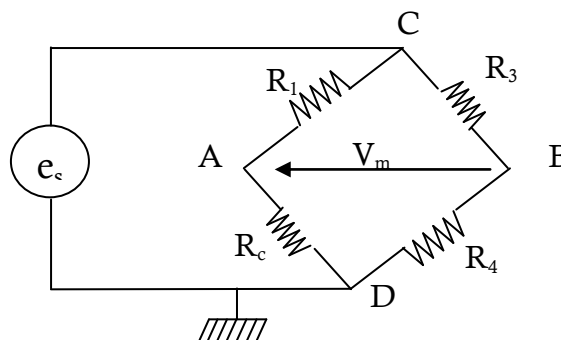


Figure 36: Montage en pont

- **Calcul des potentiels :**

En A :

$$\left\{ \begin{array}{l} e_s = (R_1 + R_c) \\ I_1 = \frac{V_A}{R_c} \end{array} \right. \quad \left\{ \begin{array}{l} e_s = \frac{(R_c + R_1)}{R_c} V_A \\ V_A = \frac{R_c}{R_c + R_1} e_s \end{array} \right.$$

En B on a :

$$\left\{ \begin{array}{l} e_s = (R_3 + R_4) \cdot I_2 \\ I_2 = \frac{V_B}{R_4} \end{array} \right. \quad \left\{ \begin{array}{l} e_s = \frac{(R_3 + R_4)}{R_4} V_B \\ V_B = \frac{R_4}{R_4 + R_3} e_s \end{array} \right.$$

$$V_m = \frac{R_c \cdot R_3 - R_4 \cdot R_1}{(R_1 + R_c) \cdot (R_3 + R_4)} \cdot e_s$$

Si on veut avoir une tension nulle en absence de mesurande, il faut avoir :

$$R_c \cdot R_3 = R_4 \cdot R_1$$

Avec une variation de ΔR de R_c on aura V_A qui prendra la forme suivante :

$$V_A = \frac{R_c + \Delta R_c}{R_1 + R_c + \Delta R_c} \cdot e_s$$

Alors V_m deviendra :

$$V_m = \left(\frac{R_c + \Delta R_c}{R_1 + R_c + \Delta R_c} - \frac{R_4}{R_3 + R_4} \right) \cdot e_s$$

On remplacera $\frac{R_4}{R_3 + R_4}$ par $\frac{R_c}{R_1 + R_c}$ parceque à l'équilibre la tension V_m est nulle, ce qui fait que ces deux éléments sont égaux. On aura donc :

$$V_m = \left(\frac{R_c + \Delta R_c}{R_1 + R_c + \Delta R_c} - \frac{R_c}{R_1 + R_c} \right) \cdot e_s$$

$$V_m = \frac{R_c \cdot R_1 + R_c^2 + \Delta R_c \cdot R_1 + R_c \cdot \Delta R_c - R_c \cdot R_1 - R_c^2 - R_c \cdot \Delta R_c}{(R_1 + R_c + \Delta R_c) \cdot (R_1 + R_c)} \cdot e_s$$

$$V_m = \frac{\Delta R_c \cdot R_1}{(R_1 + R_c + \Delta R_c) \cdot (R_1 + R_c)} \cdot e_s$$

Comme $\Delta R_c \ll R_c$, on écrit :

$$V_m = \frac{\Delta R_c \cdot R_1}{(R_1 + R_c)^2} \cdot e_s$$

Si on a $R_1 = R_c$, on aura : $V_m = \frac{e_s}{4 \cdot R_{c0}} \cdot \Delta R_c$

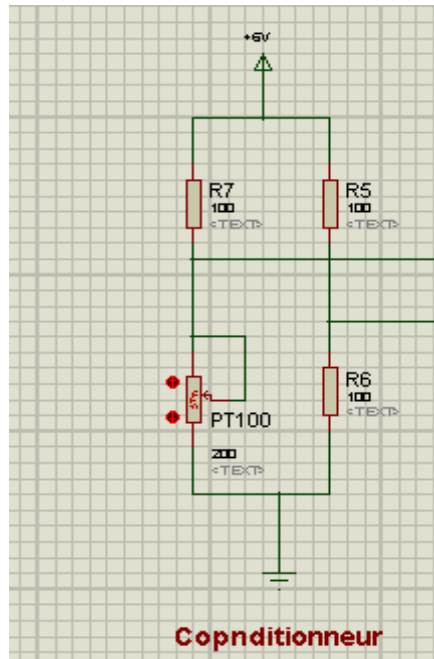


Figure 37:Schéma sur ISIS d'un conditionneur

3.1.4. Etude de capteur PT100

- Principe de mesure

$$R_t/R_0 = 1 + A t + B t^2$$

Avec :

- R_t = résistance du thermomètre à la température T .
- R_0 = résistance du thermomètre à 0°C .
- T = la température en $^\circ\text{C}$.
- $A = 3.9083 \times 10^{-3}$ et $B = -5.775 \times 10^{-7}$

Cette relation s'utilise pour les températures au-dessus de 0°C .

Pour $T < 0^\circ\text{C}$ $R_T = R_0 [1 + A t + B t^2 + C t^3 (t - 100)]$

3.1.5. Etude d'Amplificateur d'instrumentation INA 122

- Définition

Chapitre 2 : Etude technologique de la solution proposée

L'INA122 est un amplificateur d'instrumentation de précision, pour une acquisition précise et faible du signal différentiel de bruit.

Sa conception à deux ampères offre d'excellentes performances avec un courant très faible, et idéal pour les systèmes portables d'instrumentation et d'acquisition de données.

L'INA122 peut être utilisé avec des alimentations individuelles de 2.2V à 36V et le courant de repos est un simple 60 μ A.

Le circuit d'amplificateur INA 122 est représenté par le schéma interne suivant :

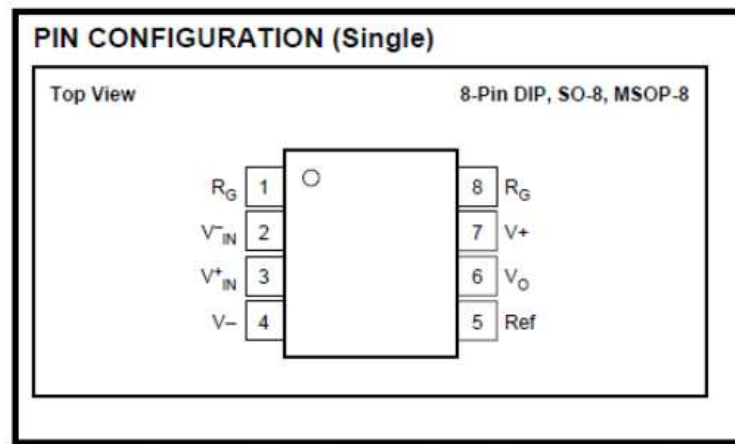


Figure 38: Configuration INA 122

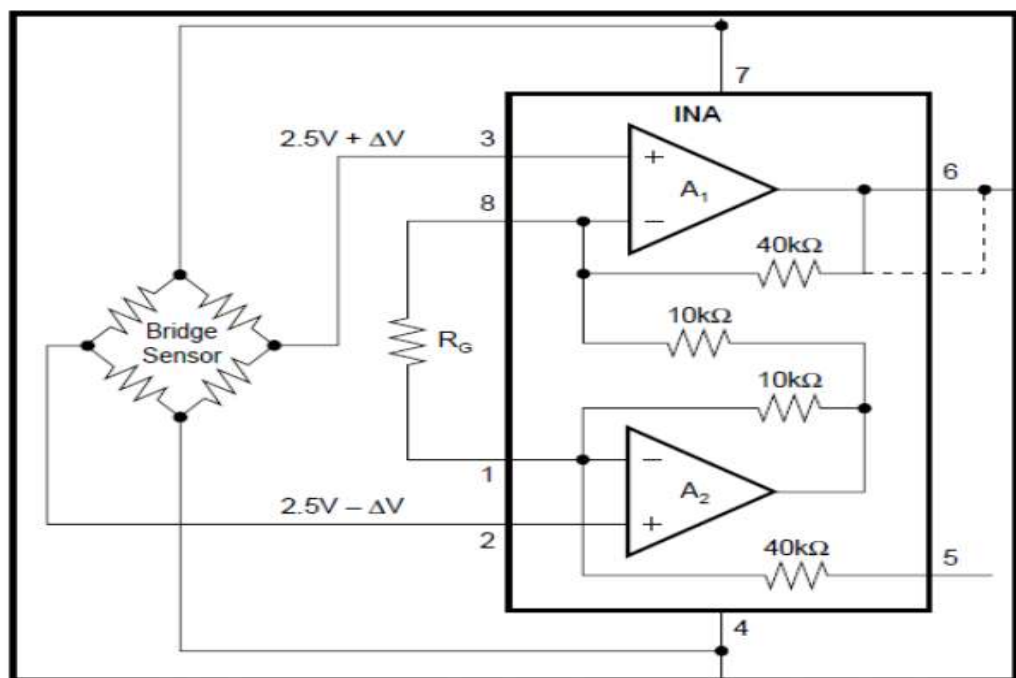


Figure 39: DATASHEET CAPTEUR INA 122

3.1.6. Application : Commande d'un moteur DC à partir un capteur pt100

▪ Introduction

Dans cette application nous allons mettre en œuvre l'étude théorique de conditionneur et l'amplificateur INA122 pour commander un moteur DC à partir d'un capteur PT100 avec l'affichage de la température et sa résistance sur LCD.

▪ Schéma synoptique et fonctionnement

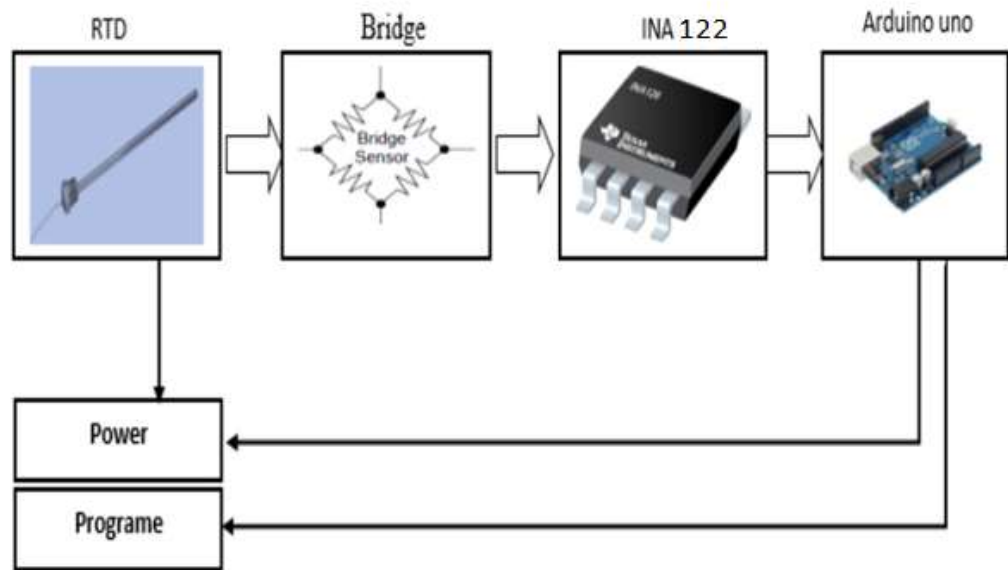


Figure 40: Schéma Synoptique du Montage pt100

Les composants utilisés sont :

- Un conditionneur,
- L'amplificateur INA 122 pour obtenir un signal compatible de celle de l'ARDUINO.
- Amplificateur LM358 pour l'adaptation de signal.

Le conditionneur de capteur pour mettre en forme le signal mesuré en une grandeur exploité par ARDUINO afin de l'amplifier avec AOP INA 122 puis le filtrer et l'adapter par le condensateur C1 et l'AOP LM358.

La figure suivante représente le branchement du capteur PT100 sous ISIS :

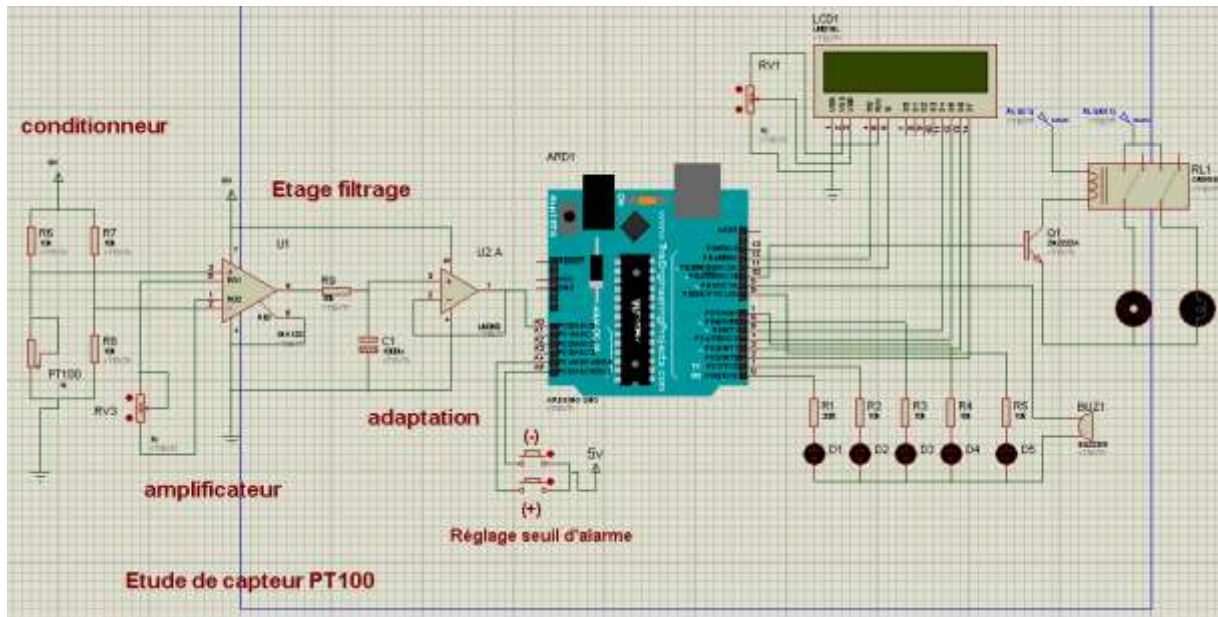


Figure 41:Schéma complet d'étude de capteur PT100

En sortie, cinq seuils d'alarmes vont être commandé par la sortie de l'ARDUINO pour :

- Indiquer le niveau de température,
- Commander un moteur (qui remplace un ventilateur) pour une température déterminée,
- Un BUZZER pour alerter.

3.1.7. Description d'un signal électrique STANDARD

En électronique, on utilise fréquemment des signaux qui transmettent une très faible quantité d'information, parfaitement connue, afin de vérifier le fonctionnement des circuits.

Dans ce schéma nous avons utilisé le conditionnement, le filtrage et l'adaptation pour obtenir à la fin un signal analogique standard de 0v à 5v lu par l'ARDUINO.

3.2. TP N°2 Régulation de température

▪ Introduction

On se propose d'étudier le système de régulation de température à base d'un capteur LM35. Le schéma de principe est donné par la figure suivante :

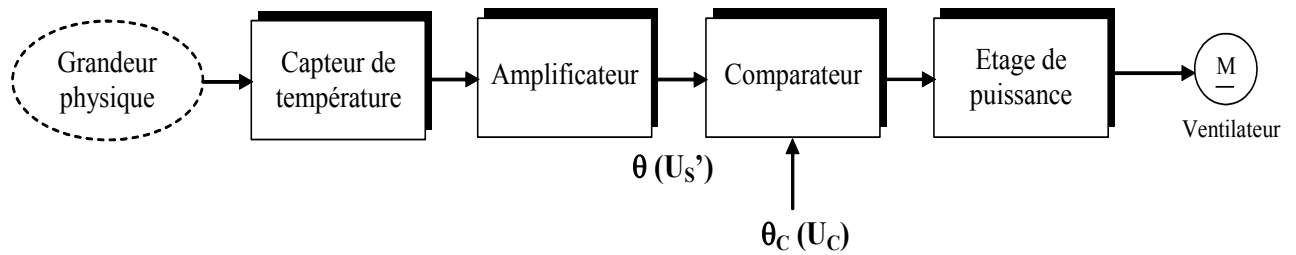


Figure 42: schéma de principe

3.2.1. Capteur de température LM35

Le **LM35** est un capteur de température qui fournit entre ses bornes une tension proportionnelle à la température mesurée. Celle-ci augmente de 10 mV par degré : Il fournit 0 V pour 0 degré, 250 mV pour 25 degrés, etc. et un maximum de 1 volt pour 100 degrés.

Il permet de faire des mesures de températures assez précises (± 0.5 degré) de 0 à 100 degrés.

Le **LM35** appartient à une famille de capteurs qui se distinguent par leurs plages d'utilisation :

- **LM35AH** : boîtier métal TO46, plage -55°C à $+150^{\circ}\text{C}$.
- **LM35CZ** : boîtier plastique TO92, plage -40°C à $+100^{\circ}$

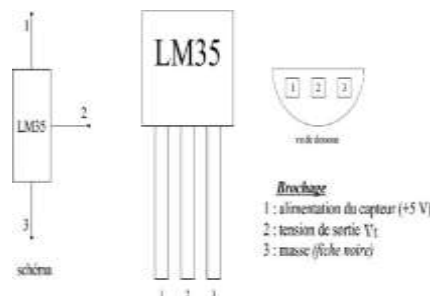
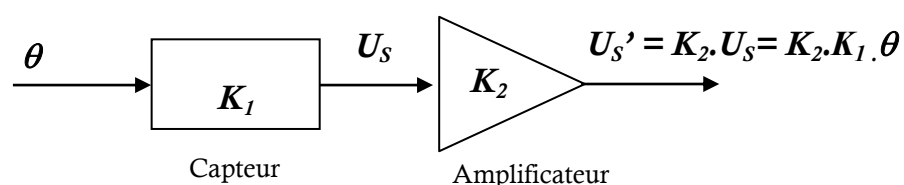


Figure 43: LM35

3.2.2. Etude de l'étage amplificateur

- **Dimensionnement des composants**



Chapitre 2 : Etude technologique de la solution proposée

Avec un **LM 35**, une variation de température de 1°C se traduit à la sortie du capteur par une variation de tension de 10 mV.

D'où la valeur de la constante de proportionnalité **$K_1 = 0,01 \text{ V}/^{\circ}\text{C}$**

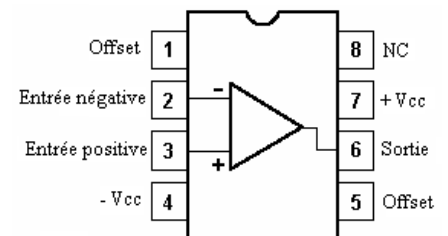
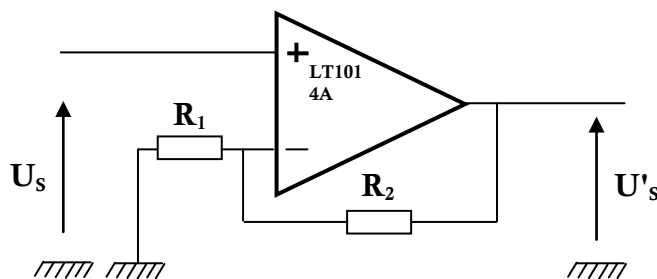
On se propose d'amplifier la tension de sortie U_S à une valeur U_S' , tel que :

- $U_S' = 5 \text{ V}$ lorsque la température θ est égale à 50°C
- $U_S' = 10 \text{ V}$ lorsque la température θ est égale à 100°C

D'où la valeur de la constante $K_2 = \frac{U_S'}{U_S} = 10$

▪ Solution proposée :

La tension de sortie U_S est amplifiée par un amplificateur opérationnel (LT1014A).



L'expression du gain K_2 de l'amplificateur non inverseur.

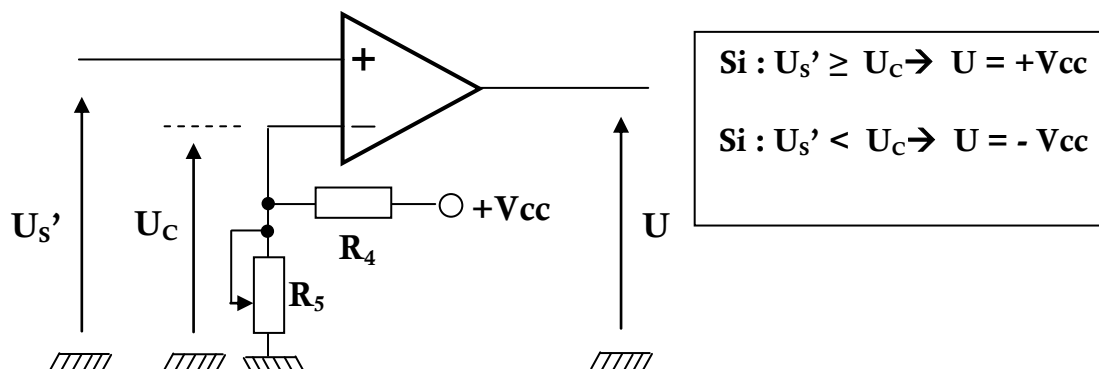
$$K_2 = \frac{U_S'}{U_S} = 1 + \frac{R_2}{R_1}$$

Si on prend $R_1 = 1 \text{ k}\Omega$ alors $R_2 = 9 \text{ k}\Omega$

3.2.3. Etude de l'étage comparateur

On utilise un second amplificateur opérationnel monté en comparateur.

La tension U_S' à la sortie de l'amplificateur est comparée à une tension de consigne U_C que l'on pourra faire varier à l'aide d'un diviseur de tension



- **Dimensionnement des composants**

$$U_C = \frac{R_5}{R_4 + R_5} V_{CC}$$

la tension de référence U_C puisse varier de 2.5 V (soit $\theta_C = 25^\circ\text{C}$) à 10 V (soit $\theta_C = 100^\circ\text{C}$) sachant que $R_4 = 3\text{ K}\Omega$.

$$R_{5min} = \frac{3 \times 2,5}{12 - 2,5} = 0.78\text{ K}\Omega$$

$$R_{5max} = \frac{3 \times 10}{12 - 10} = 15\text{ K}\Omega$$

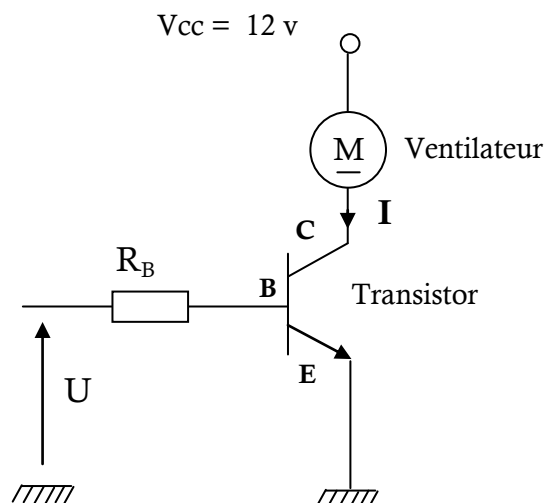
D'où on choisit la résistance réglable $R_5 = 15\text{ K}\Omega$

3.2.4. Etude de l'étage de puissance

La tension de sortie U du comparateur est appliquée sur la base d'un transistor bipolaire T. Ce dernier commande un ventilateur qui se comporte comme une résistance d'environ

$$R_M = 29,4\ \Omega$$

- Le ventilateur se met en marche dès que la température θ dépasse la valeur de consigne θ_C .
- Il refroidit l'objet surveillé et θ redevient inférieure à la consigne, a ce moment le ventilateur s'arrête.



Le transistor bipolaire NPN a les caractéristiques suivantes :

$$\beta = 100, V_{BE} = 0,7\text{ V et } V_{CEsat} = 0,3\text{ V.}$$

Le schéma suivant représente le câblage du capteur LM35 avec la carte Arduino UNO :

- **Première cas : avec Fer à souder**

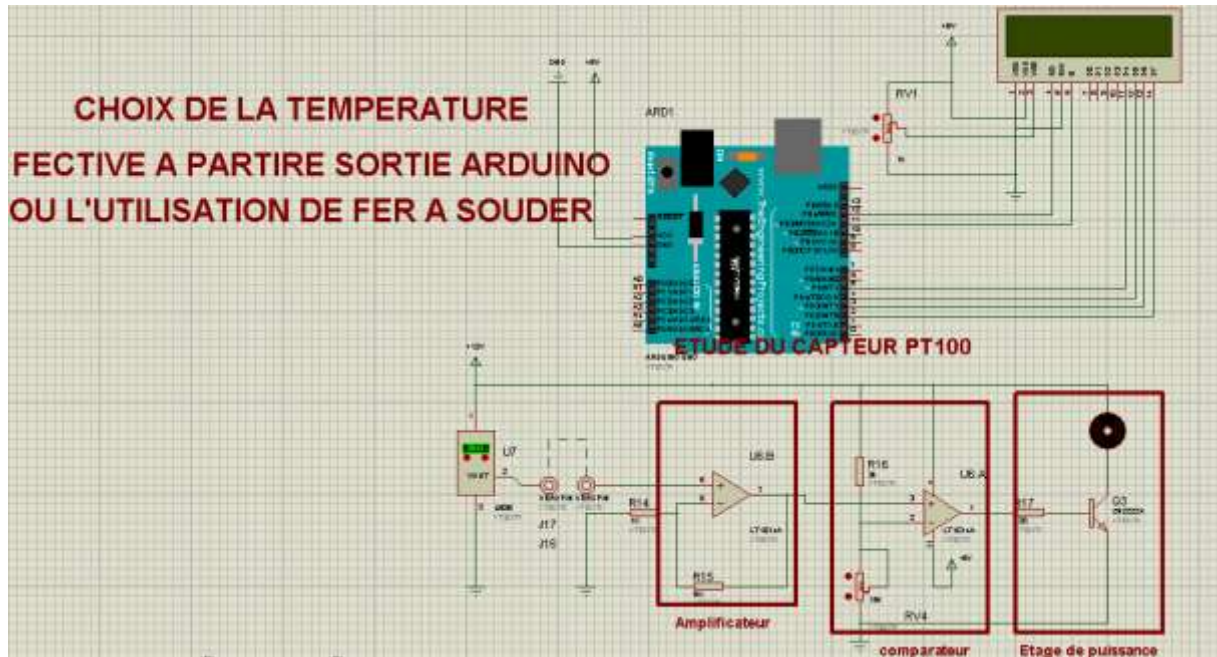


Figure 44:Montage ISIS de capteur LM35 avec Fer à souder

Le LM35 est un circuit intégré calibré en usine pour être utilisé comme capteur de température de précision. Sa principale particularité tient en ceci que sa tension de sortie est linéairement proportionnelle à la température exprimée en degrés Celsius. Plus exactement, la tension de sortie augmente de 10 mV chaque fois que la température augmente de 1°C.

- **Deuxième cas : sans Fer à souder**

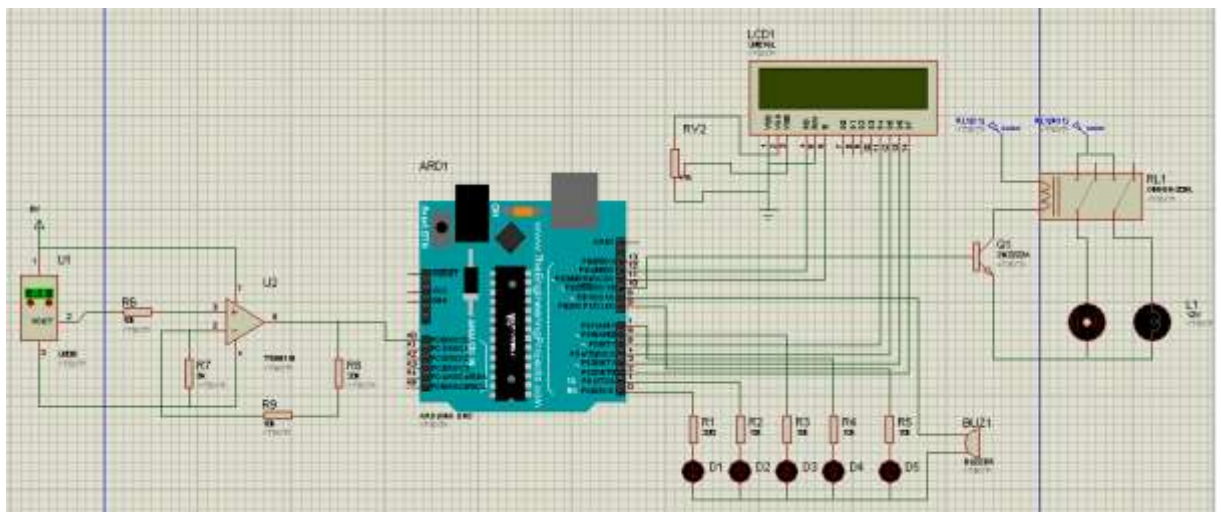


Figure 45:Montage ISIS de capteur LM35 sans Fer à souder

- **Amplificateur TS951IN**

Dans ce cas, la pleine échelle du convertisseur (0V à 5V) correspond à une plage de température de 0 à 100°C, chaque pas de mesure correspond à 0.1°C environ. Dans la pratique, une telle précision n'est pas mise à profit c'est pour cette raison on a amélioré notre montage.

L'amplificateur opérationnel U2 est un modèle de type "rail-to-rail"(modèle TS951IN), pour pouvoir bénéficier d'un affichage correct sur les premiers et derniers degrés de la plage de valeur.

3.3. TP N°3 variation de vitesse d'un moteur pas à pas

- **Introduction**

L'objectif de ce TP est de commander un moteur pas à pas.

- **Schéma synoptique**

La figure suivante représente le schéma synoptique de la variation de vitesse.

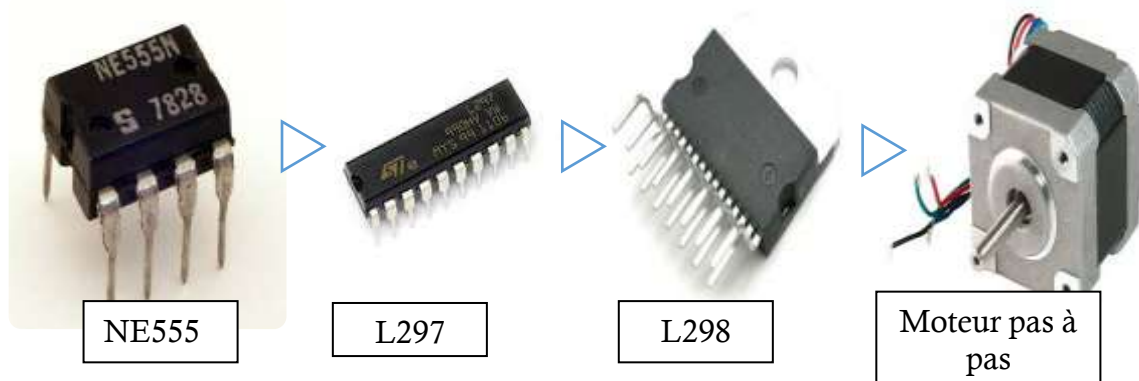


Figure 46:schéma synoptique de variateur de vitesse

Pour la commande d'un moteur pas à pas, nous avons utilisé le NE555, L297 et le L298 avec potentiomètre pour la variation de vitesse et des boutons pour freinage, modification de sens de rotation et le choix entre pas entier ou demi-pas.

Le principe de fonctionnement est comme suit :

- le NE555 fonctionne comme générateur de fréquence qui ne dépasse pas en sortie 200hz.
- Le circuit de commande de moteur pas à pas L297 est initialement prévu pour être utiliser avec un driver L298. Il reçoit les signaux de contrôle en provenance de NE555 et génère tous les signaux nécessaires pour l'étage de puissance drivé.

3.3.1. Les composants utilisés

- **NE555**

Le NE555 plus couramment nommé 555, est un circuit intégré utilisé pour la temporisation. Ce composant est toujours utilisé de nos jours en raison de sa facilité d'utilisation, son faible coût et sa stabilité.

Le NE555 contient 23 transistors, 2 diodes et 16 résistances qui forment 4 éléments :

- Deux amplificateurs opérationnels de type comparateur ;
- Une porte logique de type inverseur ;
- Une bascule SET-RESET ;

Le NE555 peut fonctionner selon deux modes : monostable, astable ou bistable.

- **Configuration des pins du circuit**

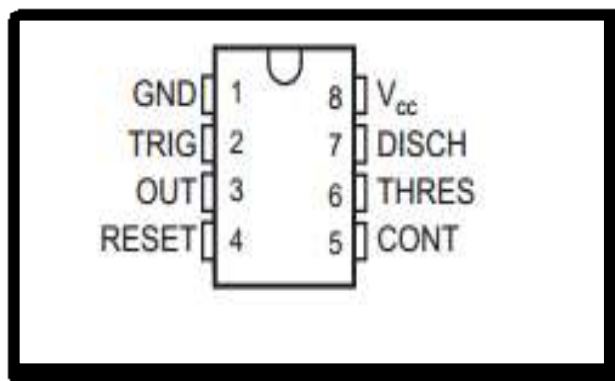


Figure 47:Pin configuration

- **Fonctionnement en mode Astable :**

La configuration astable permet d'utiliser le NE555 comme oscillateur. Deux résistances et un condensateur permettent de modifier la fréquence d'oscillations ainsi que le rapport cyclique. L'arrangement des composants est tel que présenté par le schéma ci-contre. Dans cette configuration, la bascule est réinitialisée automatiquement à chaque cycle générant un train d'impulsion perpétuelle comme ci-dessous.

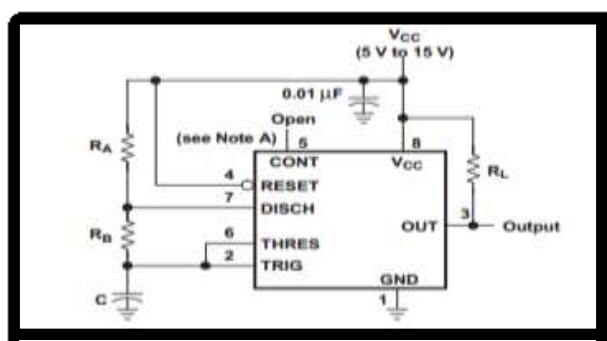


Figure 48: configuration pins

Une oscillation complète est effectuée lorsque le condensateur se charge jusqu'à 2/3 de V_{cc} et se décharge à 1/3 de V_{cc} . Lors de la charge, les résistances R_a et R_b sont en série avec le condensateur, mais la décharge s'effectue à travers de R_b seulement. C'est de cette façon que le rapport cyclique peut être modifié. La fréquence d'oscillations f ainsi que le rapport cyclique α suivent les relations suivantes :

$$f = \frac{1.44}{(R_a + 2R_b)C}$$
$$\alpha = 1 - \frac{R_b}{(R_a + 2R_b)}$$

▪ Réglage de la tension de sortie du circuit NE555 par un potentiomètre

Dans un Mode Astable on connecte les pins du circuit intégré NE555 comme le montre le schéma suivant :

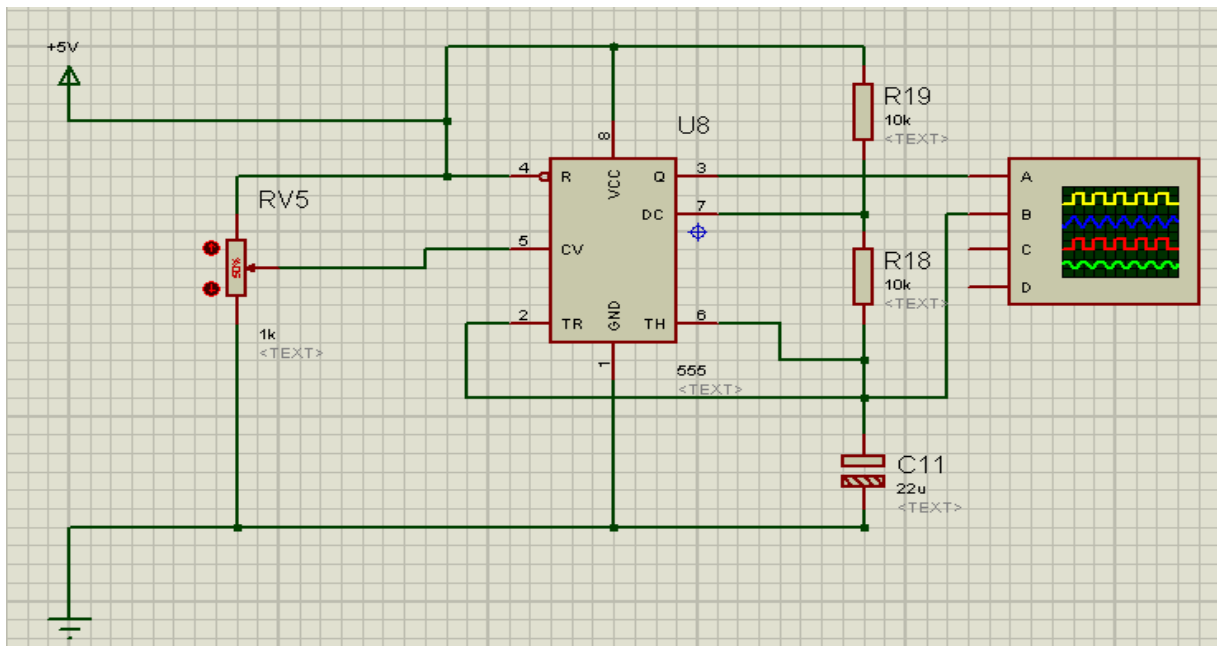


Figure 49: montage de NE555 sur ISIS

Le potentiomètre permet de régler le rapport cyclique α .

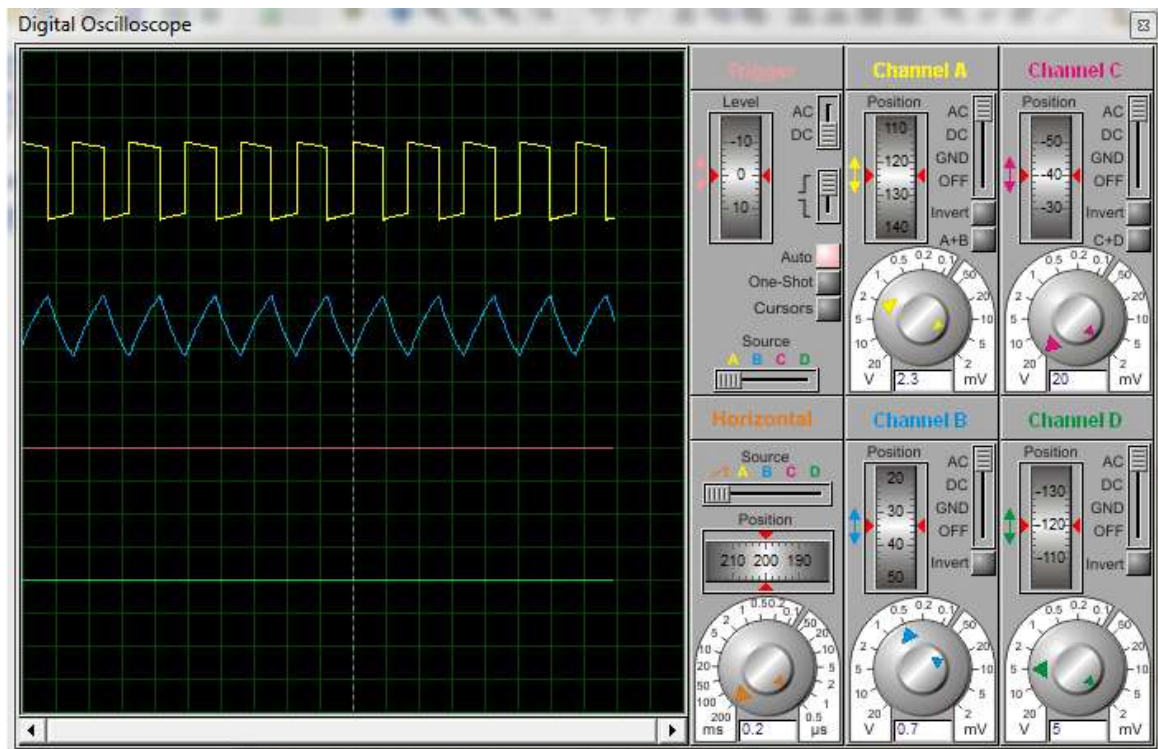


Figure 50: visualisation du signal de sortie sur un oscilloscope

- **Circuit L297**

Le L297 est un circuit qui contient toute la circuiterie de commande et de contrôle de moteurs pas à pas unipolaire et bipolaire, il est initialement prévu pour être utilisé avec un driver L298 ou L293E. Le circuit de commande de moteur pas à pas reçoit les signaux de contrôle en provenance d'un système à base de microprocesseur et génère tous les signaux nécessaires pour l'étage de puissance.

Ce circuit inclue deux régulations à découpage type PWM (modulation de largeur d'impulsion) pour réguler le courant des enroulements moteur.



Figure 51:circuit l297

- **Le séquenceur**

Le cœur du L297 est block appelé « transistor » sur le schéma-bloc.

C'est lui qui génère les séquences appropriées pour l'alimentation des phases en mode demi-pas, pas entier une phase / biphasé.

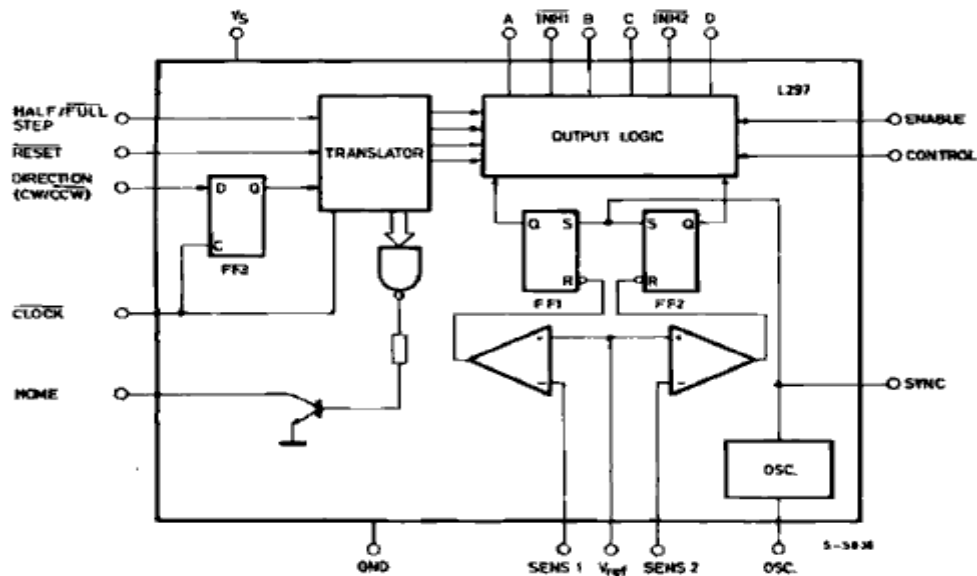


Figure 52: constitution de L297

Ce bloc est contrôlé par deux entrées de mode, le mode direction (CW / CCW) et le mode pas / demi-pas (HALF/ FULL).L'entrée clock dont les impulsions d'horloge qu'on lui applique font évoluer le séquenceur vers le prochain pas.

▪ Principales caractéristiques

- RESET INPUT & HOME OUTPUT
- Quelques composants externes
- Courant de charge programmable
- Régulation actuelle de la charge SWITCHMODE
- NORMAL / WAVE DRIVE
- Direction CLOCKWISE / ANTICLOCKWISE
- MOINS / MODES à pleine étape
- ENABLE INPUT

• Circuit L298

Le L298 est un double pont en H, c'est-à-dire un amplificateur de puissance pour moteur. Il amplifie le courant de sortie du l297 afin de fournir le courant demandé par le moteur.

le L298 avec les diodes en sorties réalisent un interfaçage de puissance, il bénéficie d'une technologie lui permettant de dissiper 160 W (avec une alimentation de 46 V et 2 A par pont).une alimentation séparée 5V pour la partie contrôle logique évite une dissipation trop importante. Elle est directement utilisable avec le 5 V du L297 ou tout autre circuit logique.



Figure 53: circuit L298

Dans un but pratique, les noms des pattes des composants (du L298N et du L297) sont souvent identiques (sur les documentations) pour une meilleure compréhension globale du montage ou de l'application.

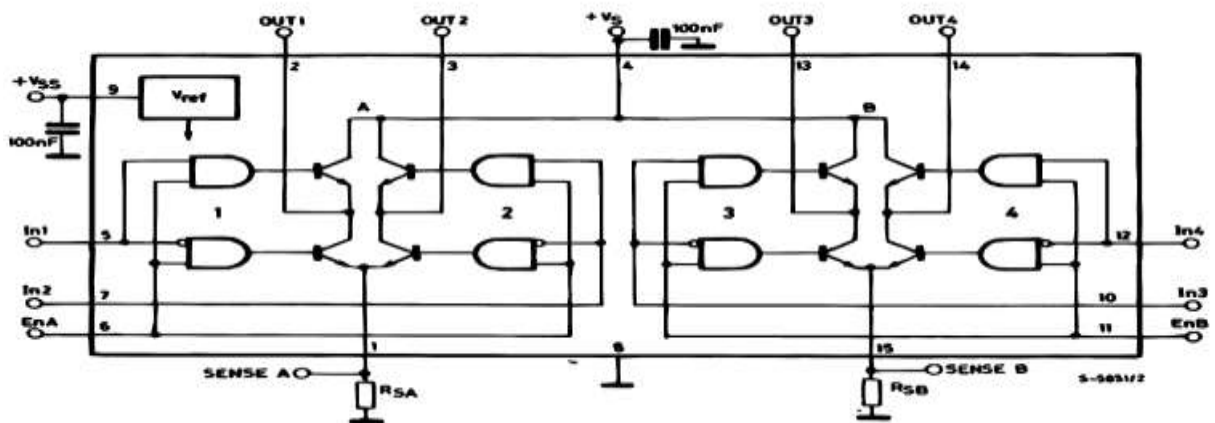


Figure 54:schéma interne de L298

3.3.2. Schéma complet sur ISIS

Nous avons utilisés trois boutons poussoirs qui sont branché avec le circuit L297 pour contrôler le pas, le sens et l'arrêt du moteur.

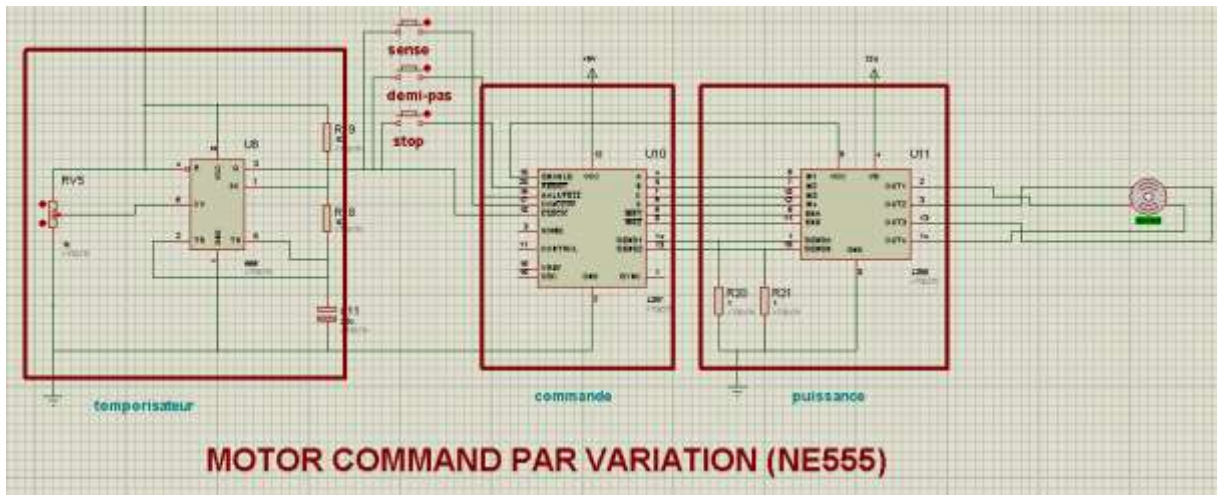


Figure 55: Schéma sous ISIS d'un variateur de vitesse.

3.4. TP N°4 commande d'un moteur DC d'après Bluetooth / contrôle de BUZZER d'après un clavier

▪ Introduction

A l'aide de Bluetooth, dans la première partie de TP nous allons commander un moteur DC à partir d'une application Android installée sur notre téléphone. Dans la deuxième nous allons commander un BUZZER d'après l'action sur un clavier 16 touches.

3.4.1. Première partie

▪ Schéma synoptique

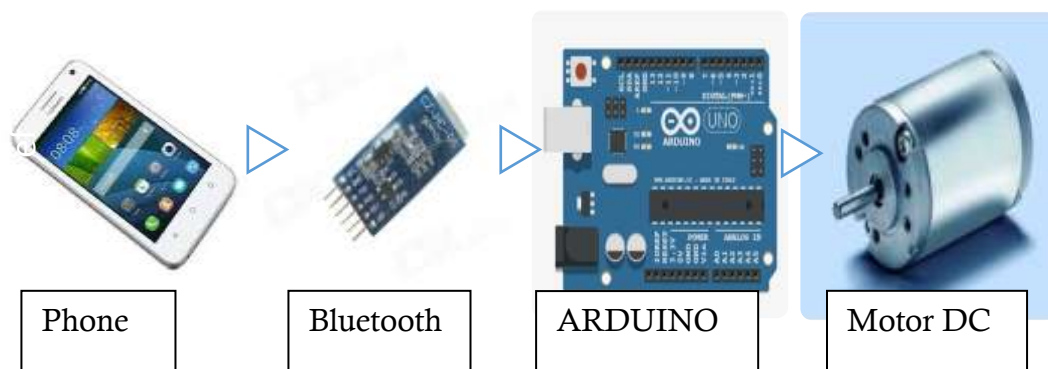


Figure 56: Schéma synoptique

Chapitre 2 : Etude technologique de la solution proposée

Pour commander un moteur DC à distance nous allons installer une application Android sur notre téléphone mobile puis le Configurer avec les paramètres de notre carte de Bluetooth, ensuite à l'aide d'un programme bien déterminé sur ARDUINO nous pouvons contrôler l'arrêt / marche, la vitesse et le sens de rotation du moteur DC.

▪ Etude sur l'utilisation du Bluetooth

Il y a 6 pattes dont 4 uniquement sont utilisées couramment. Dans l'ordre :

- KEY : non utilisé (sauf configuration du HC05).
- VCC : alimentation en 5V de l'ARDUINO.
- GND : à relier au GND de l'ARDUINO.
- TXD : à relier au RX de l'ARDUINO.
- RXD : à relier au TX de l'ARDUINO.
- STATE : non utilisé.

▪ Schéma sur ISIS

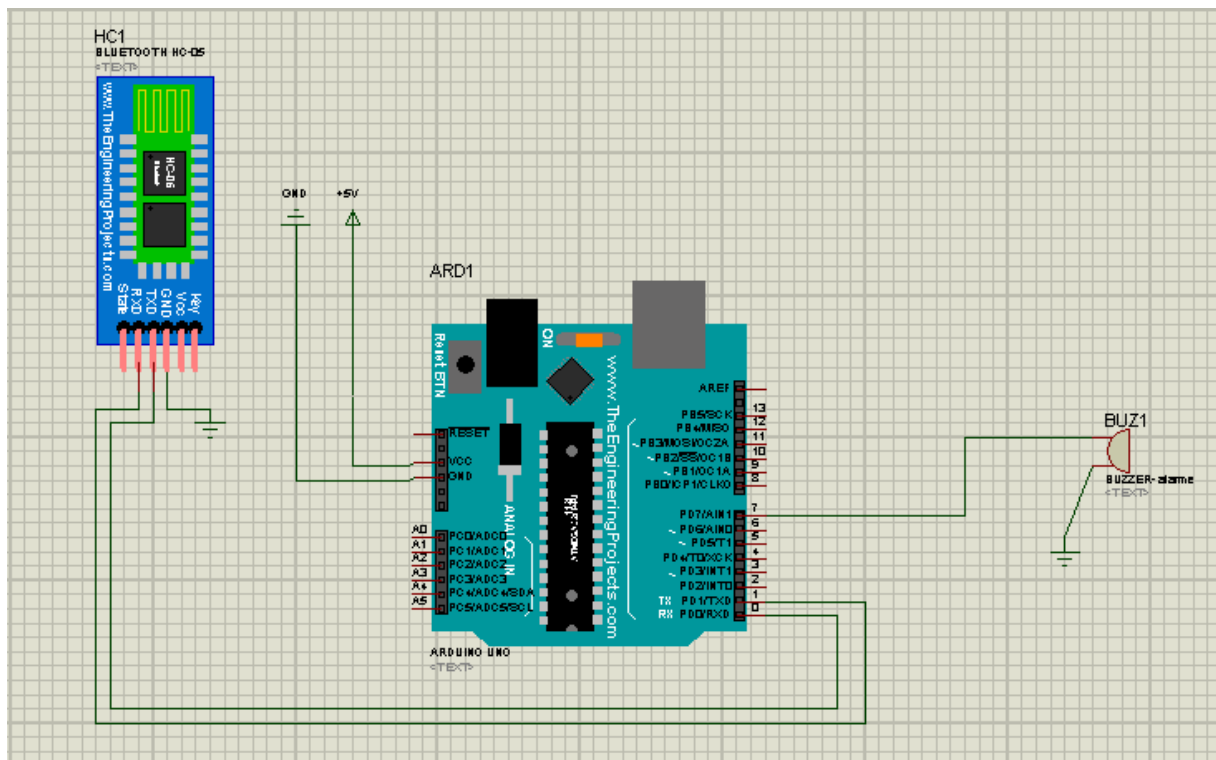


Figure 57:schéma de câblage du Bluetooth

3.4.2. Deuxième partie

- Schéma synoptique

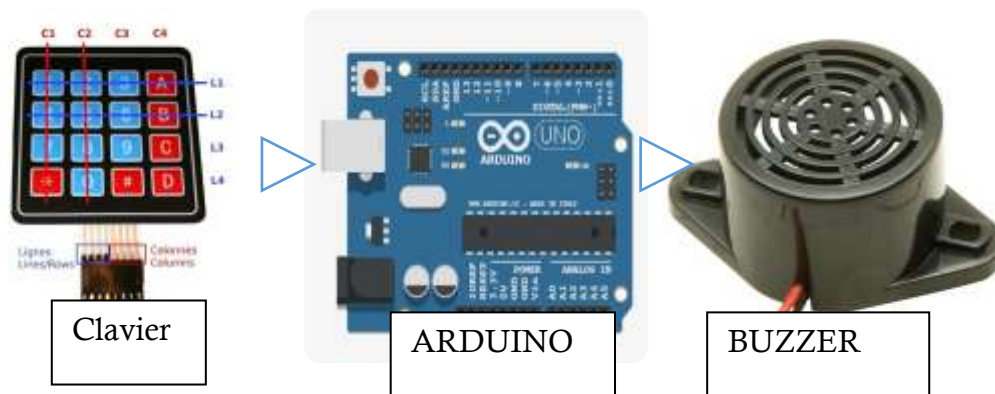


Figure 58:schéma synoptique de BUZZER

Un BUZZER mécanique se présente sous la forme d'un petit boîtier rectangulaire ou cylindrique, avec connexion électriques rigides pour fixation directe sur circuit imprimé, ou avec connexion électriques constituées de fils électriques souples. Dans ce deuxième cas, le BUZZER possède deux petites pattes de fixation. La puissance sonore d'un tel composant est de l'ordre de 85 dB/cm.

Dans cette partie nous allons commander un BUZZER à partir d'un clavier pour réaliser un piano en utilisant la carte ARDUINO pour la contrôler de niveau du son de BUZZER.

▪ Etude sur l'utilisation du BUZZER

Un BUZZER nécessite une tension continue pour fonctionner, cette dernière doit généralement être comprise entre 3 V et 28 V, selon les modèles. Il prévu pour fonctionner sous 6 V fonctionne généralement très bien pour toute tension d'alimentation comprise entre 4 V et 8 V.

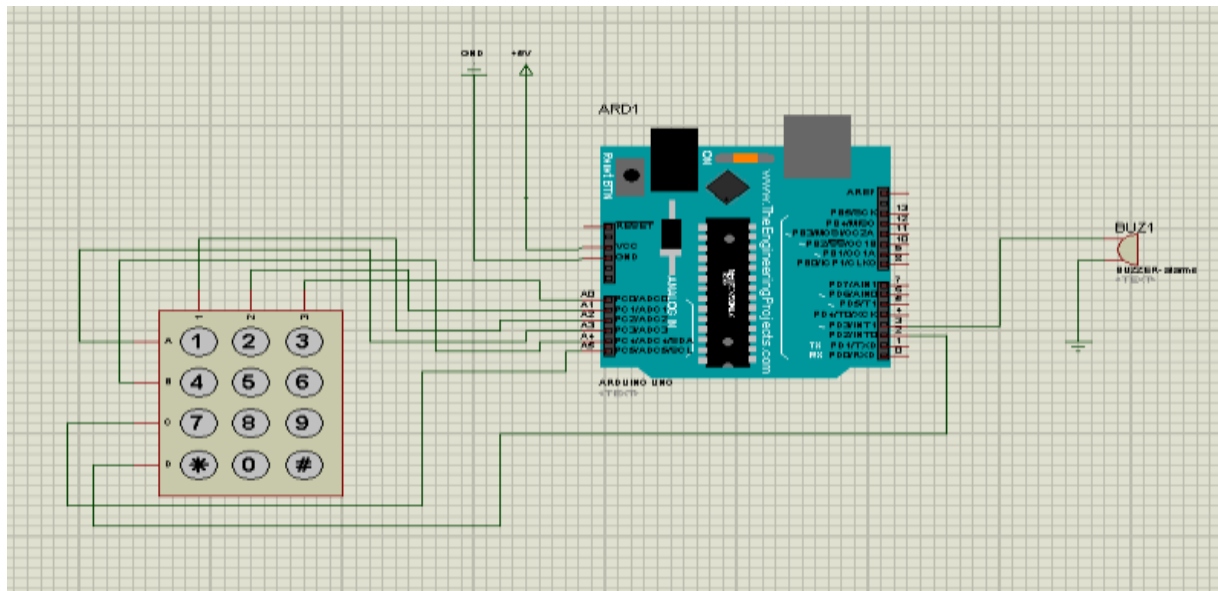


Figure 59:Schéma de branchement de clavier

4. Etude théorique de connexion de RS232

Pour assurer la liaison série entre la maquette et l'ordinateur, on a relié le MAX232 avec DB9 par l'utilisation des pines T2OUT et R1IN et on a branché les pines T2IN et R2OUT avec les pines RX et TX d'ARDUINO

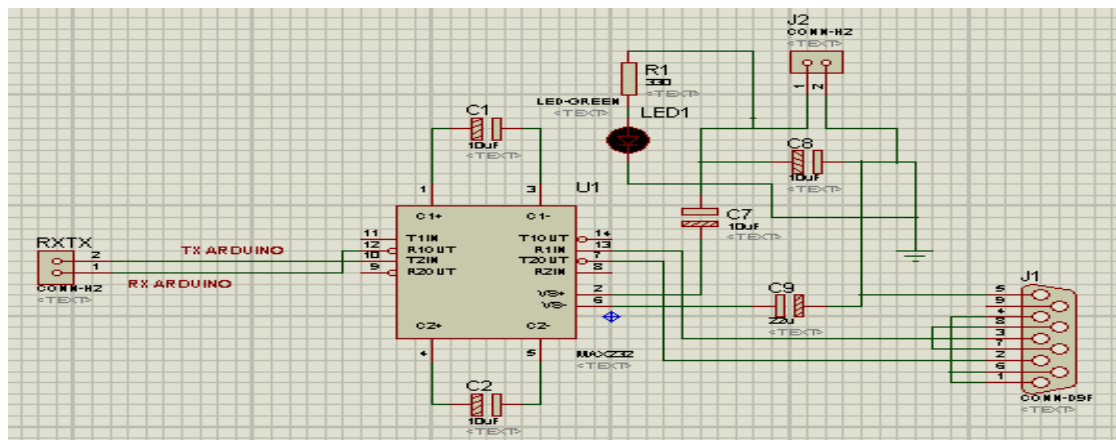


Figure 60:liaison série RS232

Conclusion

Dans ce chapitre nous avons bien étudié les solutions de nos TP et leurs schématisations. Puis nous avons consacré le troisième chapitre pour l'étude pratique dans le but de bien expliquer leur fonctionnement.

Chapitre 3 : Conception et Réalisation de la maquette

Introduction

Après une étude détaillée et une conception cohérente on va voir maintenant la réalisation de notre solution conçu. Dans cette partie on touchera aux différents logiciels et outils utilisés pour la création de notre carte, notre conception de maquette sur Galaad et l'interface graphique sous LabVIEW, en illustrant les différents résultats.

1. Les logiciels utilisés dans la simulation et la conception

1.1. Proteus

Proteus est une suite logicielle destinée à l'électronique. Développé par la société Labcenter Electronics, les logiciels incluent dans Proteus permettent la CAO dans le domaine électronique. Deux logiciels principaux composent cette suite logicielle : ISIS, ARES.

L'utilisation du logiciel « **ISIS** » permet de mieux visualiser le bon déroulement du système ainsi que d'avoir une idée claire sur la partie matérielle et la conception des circuits imprimés. Il permet de limiter les essais réels.

L'utilisation du logiciel « **ARES** » permet de router des cartes électroniques en mode automatique ou manuel. Il est possible d'utiliser « **ARES** » sans avoir créé au préalable un schéma dans ISIS.

Cette fonctionnalité permet de réaliser des circuits de faible complexité en plaçant les composants et en traçant les pistes directement. Une fois les connections établies, il est possible d'effectuer un routage automatique des pistes.

Dans ce logiciel vous pouvez également créer de nouveaux boîtiers et les placer dans une bibliothèque.

1.2. LabVIEW

LabVIEW (Laboratory Virtual Instrument Engineering Workbench) est un environnement de développement graphique qui permet de créer des applications modulaires (notion de sous VI) et extensibles pour la conception d'applications, le contrôle et le test.

1.3. ARDUINO

Une carte de traitement ARDUINO et son logiciel de programmation qui sont initialement destinés pour la programmation de carte et la production des signaux électriques, de manière à effectuer des tâches très diverses.

1.4. GALAAD

Galaad est un ensemble de logiciels 32 bits destinés au pilotage intelligent de machines à commandes numériques.

Les applications les plus courants de GALAAD sont le fraisage 2D1/2, la gravure, la découpe, le contour. De conception tout-en-un, GALAAD intègre dans un ensemble cohérent le processus complet de fabrication, et ajoute des modules annexes pour les applications spéciales.

La fabrication de pièce s'effectue par la suivre de ces 3 étapes :



Figure 61:Le processus complet de fabrication.

1.4.1. Etape 1 : Le dessin de la pièce par CAO (Conception Assistée par Ordinateur)

C'est la création d'un nouveau fichier suivant le type de matériaux utilisé, Galaad propose des paramétrages par défaut comme par exemple la vitesse de déplacement de l'outil, des menus simples : création de point, ligne, poly ligne, rectangle, cercle et texte.

1.4.2. Etape 2 : La gestion des trajectoires par FAO (Fabrication Assistée par Ordinateur)

La définition des parcours d'outils est prise en charge par le module FAO qui génère un fichier de trajectoires que les outils devront suivre.

A cette étape, Galaad offre plusieurs options sur lesquelles il est possible à tout moment d'intervenir :

- effectuer un parcours d'essai,
- répéter le même usinage,
- éteindre le système après le cycle outil,
- morceler la pièce,
- reprendre un usinage interrompu (après la casse d'un outil par exemple).

1.4.3. Etape 3 : Simulation et Usinage de la pièce CN (commande numérique)

Enfin le pilotage de la machine d'usinage à commande numérique, réalise la pièce en suivant les parcours d'outils ainsi produits.

Depuis le module de dessin, un simple clic de souris suffit à lancer le processus de fabrication sur une machine pilotée directement par Galaad.

2. Réalisation de la carte d'alimentation Après avoir étudié notre besoin des TP, nous avons conclu qu'on aura besoin de quatre sources de tension continu

- 12 V
- 6 V/-6 V
- 5 V

Le schéma ci-dessous montre le schéma de carte d'alimentation après le lancement de routage :

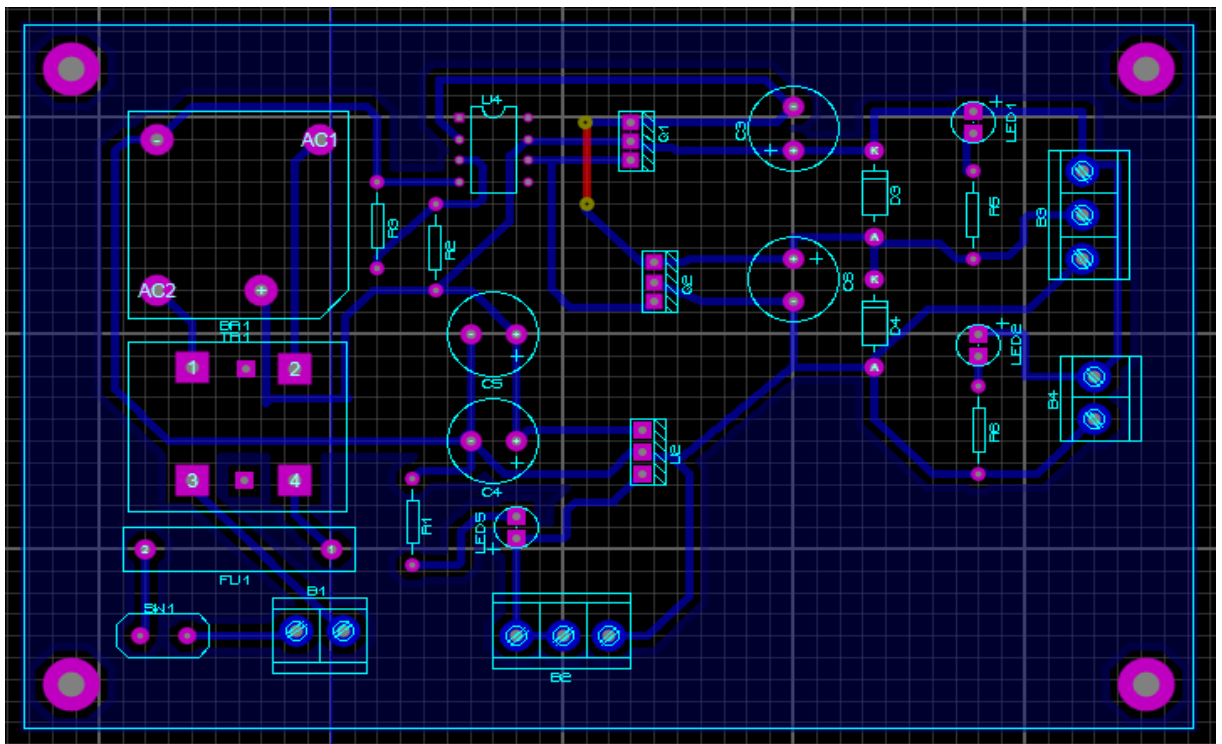


Figure 62:Schéma de routage de la carte d'alimentation

Le schéma dans la figure ci-dessous présente la carte d'alimentation après la fixation de leurs composants :



Figure 63:Photo réelle de carte d'alimentation

- Les composants utilisés dans la carte d'alimentation

Tableau 2: tableau des composants n°1

Composant	Repère	caractéristiques	Quantité	Référence
Bornier		2 pins	02	
Bornier		3 pins	02	
Fusible		1A	01	
Transformateur		230V/12V	01	
Pont diode		1.5A	01	
résistance		100k ohm	02	
résistance		330 ohm	01	
Condensateur		22 uF	02	
condensateur		47 uF	02	
Régulateur	7805	5V DC	01	
Diode		1N4001	02	
Transistor			02	BD135
Amplificateur				UA741

3. Réalisation de carte de communication série RS232 :

Pour la communication série entre la maquette et l'ordinateur nous avons réalisé la carte RS232 suivante :

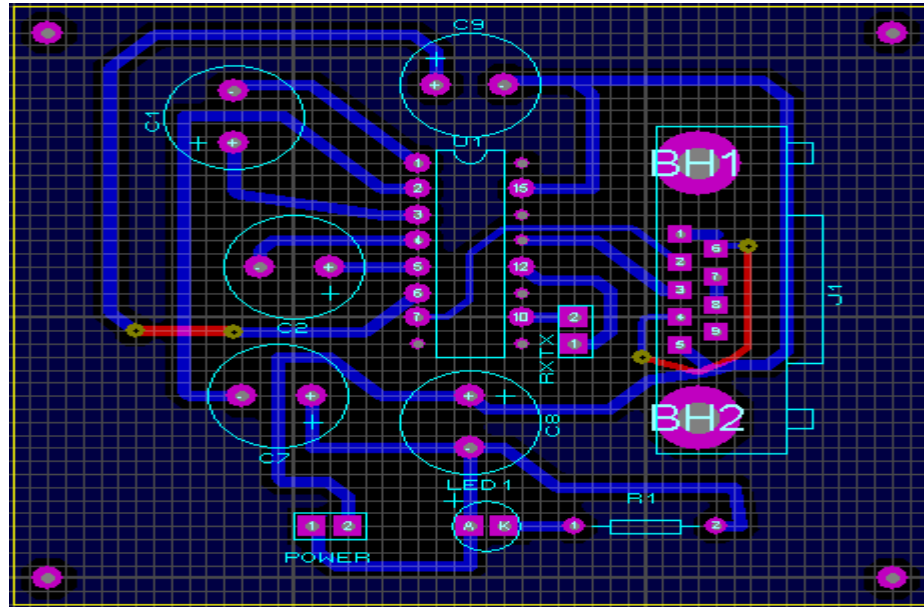


Figure 64:Schéma de routage de carte RS232

Le schéma dans la figure ci-dessous présente la carte RS232 en réelle après la fixation de leurs composants :



Figure 65:Photo réelle de la carte RS232

- Les composants utilisés dans la carte RS232

Tableau 3: tableau des composants n°2

Composant	Repère	caractéristiques	Quantité
Bornier		2 pins	01
Max232	11A9HHM		01
Résistance		330 ohm	01
Condensateur		10 uF	04
Condensateur		22 uF	01
Diode Led		Rouge	01
Connecteur	D9F		01

4. Réalisation de carte de commande moteur pas à pas

Pour le contrôle de moteur pas à pas nous avons réalisé cette carte qui contient les principaux circuits de commande NE555, L297 et L298.

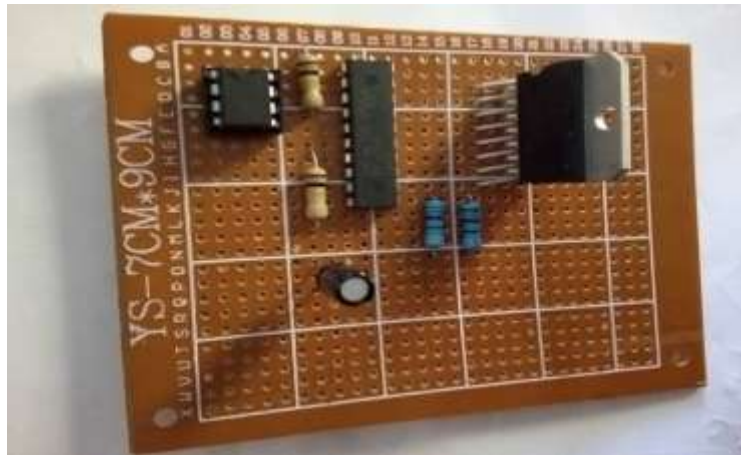


Figure 66: photo réelle de la carte de commande d'un variateur de vitesse

- Les composants utilisés dans la carte

Table 1 : les composants de carte de variation de vitesse

Composant	Repère	caractéristiques	Quantité	Référence
Potentiomètre		1 k	01	
NE555			01	555
Résistance		10 k ohm	02	
Bouton poussoir			03	
Condensateur		22 uF	01	
L297			01	
L298N			01	CZOPB501
Résistance		1 ohm	02	

5. Conception du boîtier

Nous avons fait une conception complète de la face avant de notre maquette sur GALAAD en utilisant la dimension de 210mm/297mm/5mm pour les deux plaques grands et 100mm/210mm/5mm pour les restes de deux. (Voir annexe pages 1, 2 et 3)

De plus nous avons fait un modèle similaire à la valise sur le logiciel Solid Works. La figure suivante représente ce modèle.

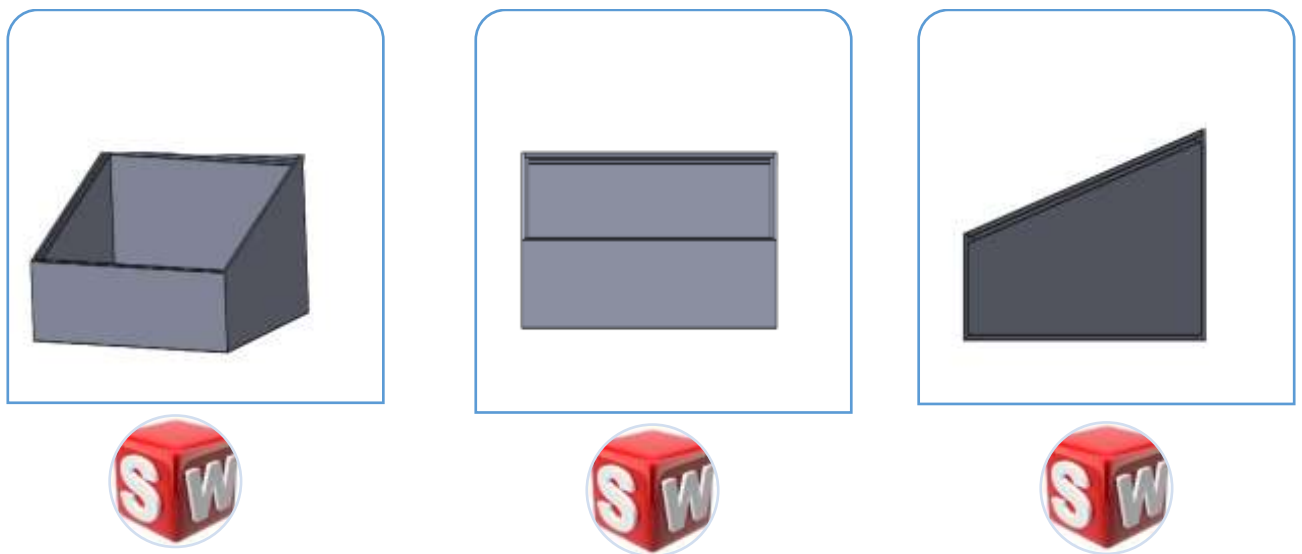


Figure 67: modèle de valise sur Solid Works

En fin voilà le banc d'essai en réelle après la fixassions de tous les composants et les cartes :

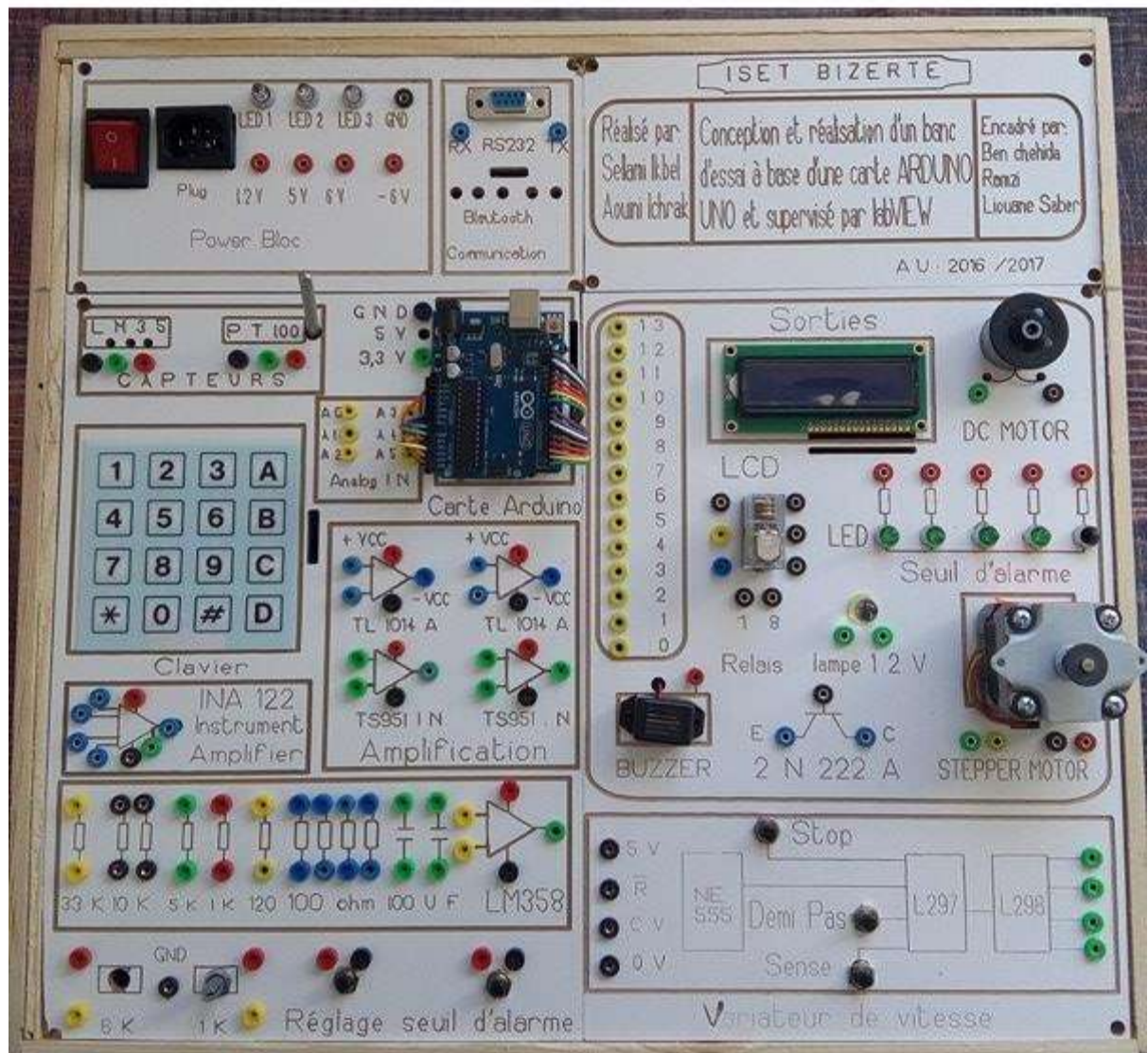


Figure 68: photo réelle du banc d'essai

6. Réalisation des interfaces des TP sur labVIEW :

Dans cette partie nous allons vous présenter les trois interfaces qu'on a faites sur LabVIEW.

La première interface c'est du capteur LM35 dont on va suivre la température mesuré et indiqué s'elle est au niveau bas, moyen ou haut.

La deuxième interface c'est du capteur PT100 dont on va mesurer et afficher la température détecté et la résistance du capteur, indiqué le seuil de la température, tourner le moteur DC 9v à une température bien déterminé et contrôler le BUZZER.

La dernière interface c'est du moteur pas à pas 12v dont on va varier sa vitesse et contrôler son sens et son pas.

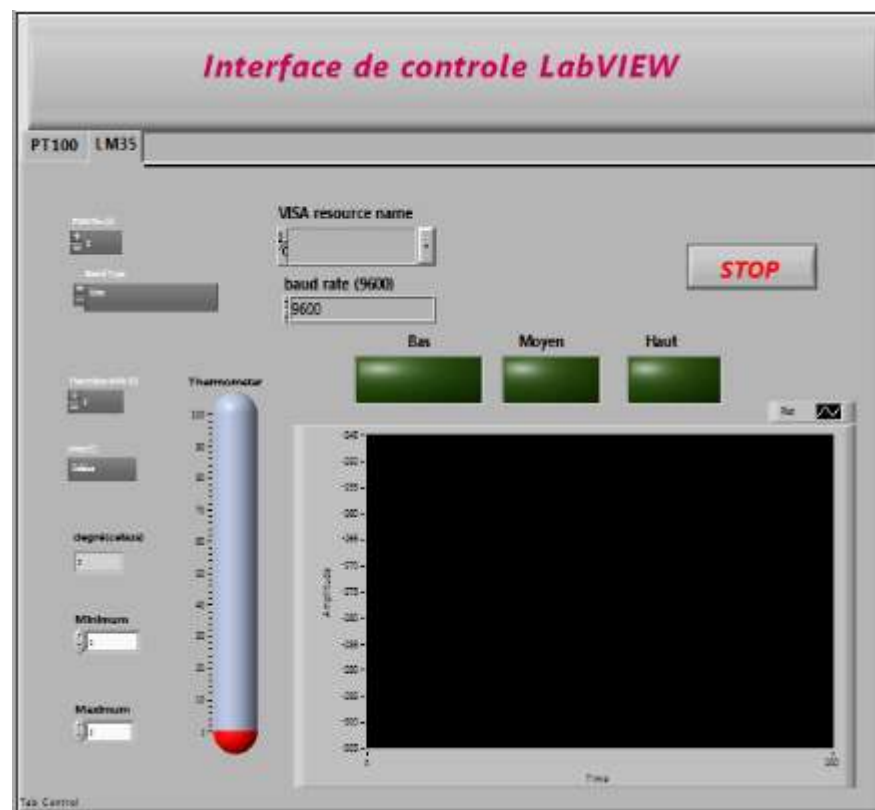


Figure 69 :interface du capteur LM35 sur LabVIEW

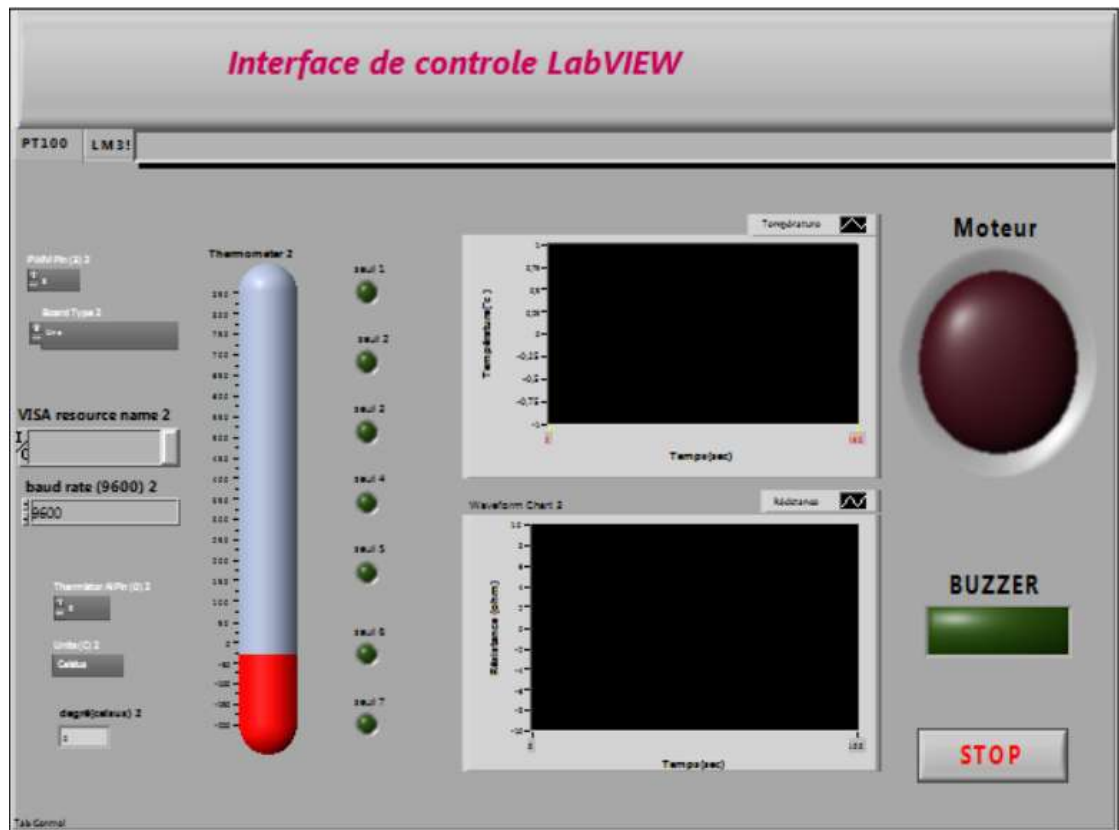


Figure 70:interface du capteur PT100 sur LabVIEW



Figure 71:interface du variateur de vitesse d'un moteur pas à pas

Conclusion

Après tout ce travail qui a procédé notre banc d'essai devenu prête à l'emploi, alors il fait préparer les montages des TP pour que ce banc d'essai fasse son rôle.

Conclusion Générale Et Perspective

Le travail mené dans le cadre de cette étude propose la conception et la réalisation d'un banc d'essai à base des capteurs et d'une carte ARDUINO UNO.

Après avoir recensé les principaux équipements utilisables, notre choix c'est porté sur La conception d'une maquette qui s'articule autour de 2 axes :

- Programme officiel des ateliers des capteurs.
- L'utilisation des technologies modernes (carte ARDUINO, supervision par labVIEW, Bluetooth, etc.).

Nous avons commencé par l'étude de l'art dans le premier chapitre ensuite dans le deuxième chapitre nous avons étudié la solution proposé pour en arriver à la fin à sa réalisation ce qui fera le sujet du troisième chapitre.

Ce projet fait partie d'un effort continu d'étude et de recherche pour développer dans un long terme l'interface sur labVIEW.

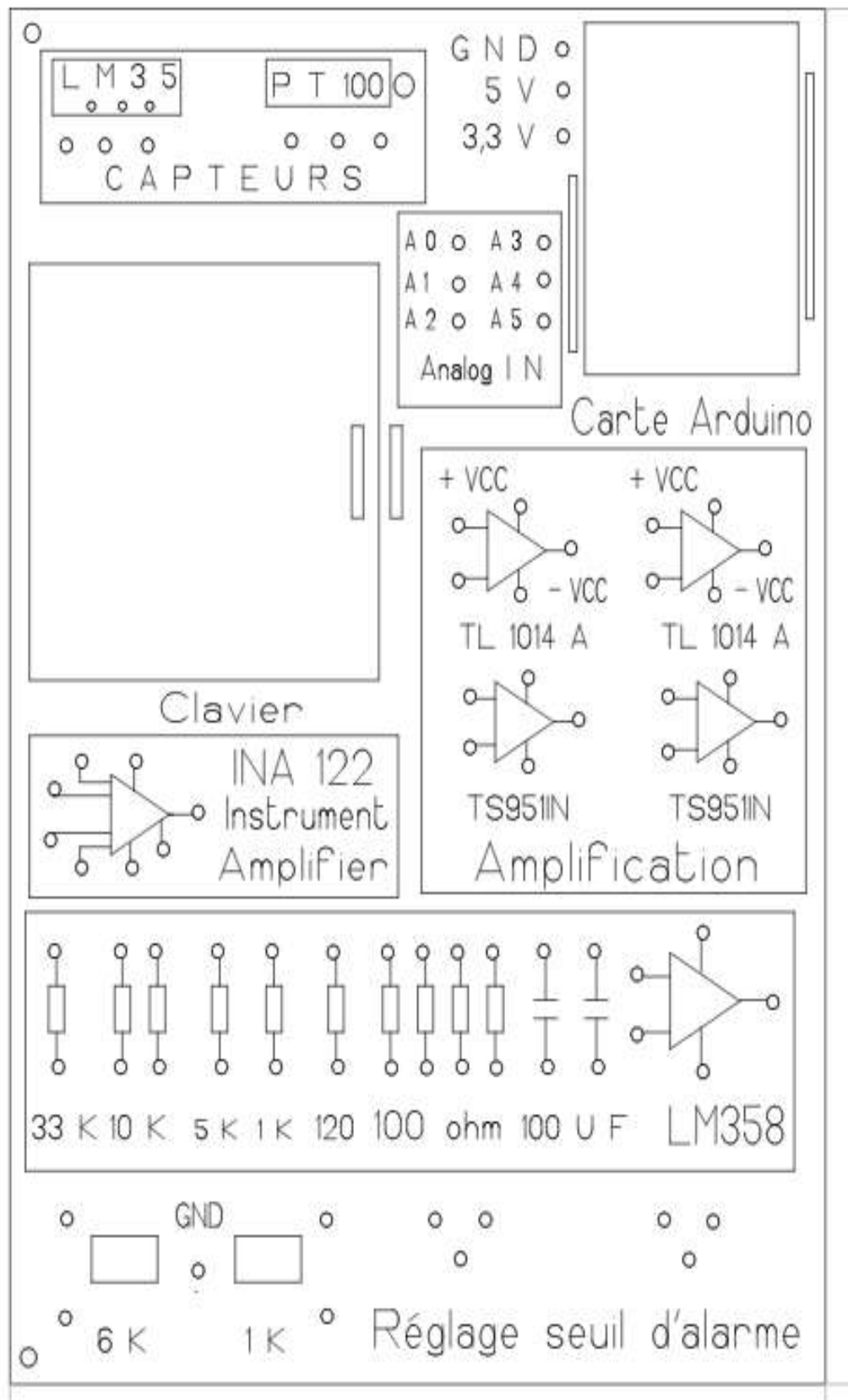
Nous avons eu l'occasion au cours de ce projet d'étudier, concevoir et utiliser une diversité de matériels et logiciels que nous ont été utiles pour notre projet ainsi que pour l'approfondissement de nos connaissances tels que ISIS, GALAAD, SOLIDE WORKS, etc.

Nous souhaitons que ce projet satisfasse à ses lecteurs et nous espérons également qu'il sera à la hauteur d'une éventuelle extension et exécution.

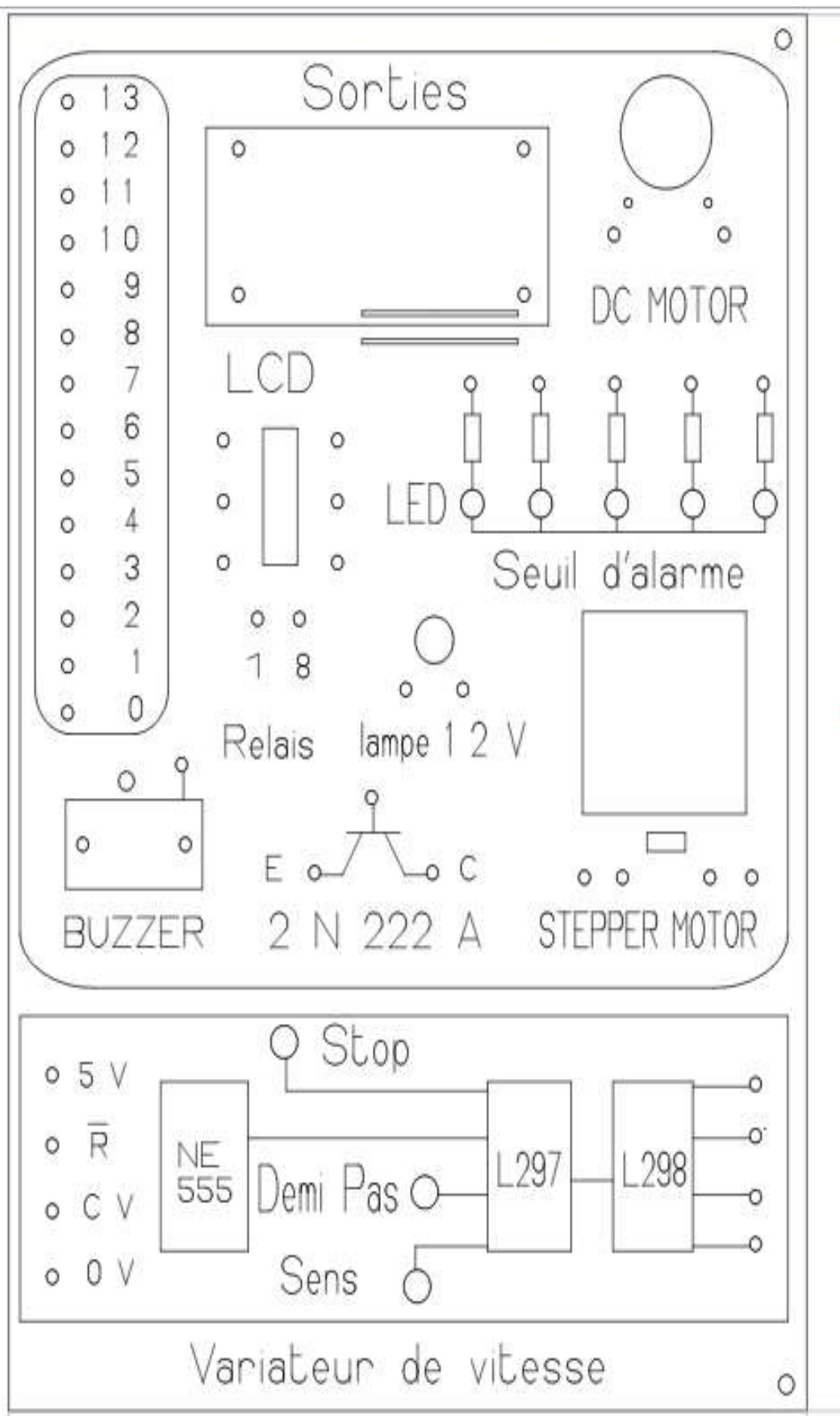
Nétographie

- <http://www.alldatasheet.com>
- <http://www.arduino.cc>.
- <http://nationalinstrument.com>
- <https://phmarduino.wordpress.com/2015/12/07/mesure-de-temperature-lm35/>
- <https://openclassrooms.com>
- http://tpil.projet.free.fr/TP_Labview/Labview_Tableaux.html
- <https://www.carnetdumaker.net/articles/mesurer-une-temperature-avec-un-capteur-lm35-et-une-carte-arduino-genuino/>

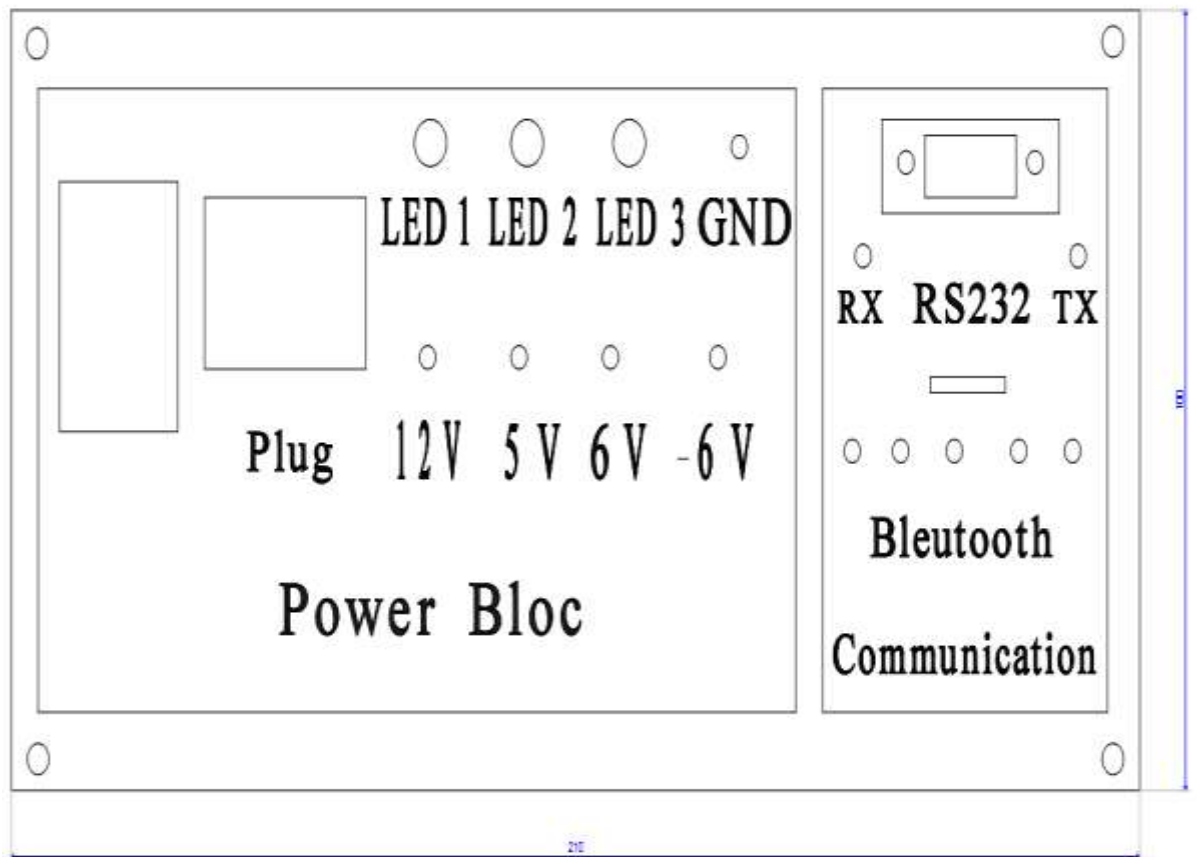
Annexe



Face avant 1 du maquette



Face avant 2 du maquette



Face avant 3 du maquette



Face avant 4 du maquette

Résumé

Notre projet consiste à réaliser un banc d'essai de travaux pratique basé sur deux capteurs de température et une carte Arduino UNO, chez l'institut supérieur des études technologique de Bizerte, consacré pour 4 TP. Nous avons testé ses deux capteurs avec un code Arduino et nous avons créé une interface sur LabVIEW pour chacun d'eux et pour la variation de vitesse d'un moteur pas à pas. Nous avons met en œuvre un banc d'essai de travaux pratique en compagnie de l'énoncé de quatre TP et permettent l'acquisition et le contrôle de donné des capteurs par leur interfaces qui sont supervisés par LabVIEW.

Mots clés : Arduino, LabVIEW, capteur de température, banc d'essai, travaux pratique.

Abstract

Our project consists of realizing a practical test bench based on two temperature sensors and a card Arduino UNO, at the higher institute of the technological studies of Bizerte, dedicated for four scenarios. We tested its two sensors with an Arduino code and we created a LabVIEW interface for each of them and for the speed variation of stepper motor.

We have implemented one practice model work with the statement of four scenarios. Enable acquisition and control of sensor data and output of Arduino with the LabVIEW interface.

Keywords: Arduino, LabVIEW, temperature sensor, test bench, practical work.