

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE  
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE  
UNIVERSITÉ MOHAMED BOUDIAF - M'SILA  
FACULTÉ DE TECHNOLOGIE  
DÉPARTEMENT D'ÉLECTRONIQUE



MÉMOIRE DE FIN D'ÉTUDES EN VUE DE L'OBTENTION DU DIPLÔME  
DE MASTER EN ELECTRONIQUE

**Spécialité : Electronique des systèmes embarquée**

**THÈME**

---

Etude et réalisation d'un système de commande  
acquisition et régulation PID a base d'un api /pic

---

Proposé et dirigé par :

**Dr Lahouaoui LALAOUI**

Présenté par :

**Fares DIF**

**Ahlam BENAMARA**

Année universitaire : 2019/2020

**Résumé :**

Le travail présenté dans le cadre de ce projet concerne. Étude la régulateur PID dans le système détection niveau d'eau. Au début de ce travail, nous avons proposé un système de régulation PID. Puis nous avons présenté la différente forme de régulateur PID ainsi que les avantages et les inconvénients. Deuxièmement, on a étude les composants utilisant, ses la carte Arduino NANO, ses la partie logiciel dans cette carte, Puis nous avons présenté les différents composants utilisant dans cette projet. Finalement, nous avons consacré à une partie de simulation et résultat de simulation à chaque niveau

**Mots Clés :** Régulateur PID, système de commande, carte d'aquisition.

## Remerciement

Avant tout, mes remerciements vont à **Allah** le tout puissant qui m'a aidé à faire ce Travail et qui m'a donné le courage, la patience et la santé durant toutes ces longues années D'étude afin que je puisse arriver à ce niveau.

Ce travail de recherche a été effectué au sein de l'université de Mohamed Boudiaf -M'SILA, sous la direction scientifique de Monsieur Lahouaoui LALAOUI que je tiens à remercier très vivement pour son enthousiasme envers mon travail, sa disponibilité et son soutien scientifique et humain. Je le remercie, également, pour la confiance qu'il m'a Accordée, ses encouragements et ses précieux conseils.

Je tiens à exprimer ma gratitude envers les membres de jury : Monsieur ....., Maitre de conférence '....' à l'Université de Mohamed Boudiaf -M'SILA qui m'a fait l'honneur de présider ce jury.

J'adresse un grand merci à tous ceux qui m'ont aidé de près ou de loin durant ces Années de recherche et d'études. Je ne peux conclure sans un mot de remerciement à tous mes Enseignants tout le long de ma formation.

Enfin, un grand merci à mes parents, et à toute ma famille pour leur amour et leur soutien surtout durant les deux dernières années.

## *Dédicace*

*Je dédie ce travail :*

*A mon binôme pour tout le travail qu'on a accompli*

*À ma mère*

*Mon père*

*Mon frère NOOR ELSSALEM*

*Mon frère AYOUB ET OUSSAMA*

*Mon neveu LOAY*

*À ma famille*

*À toute la famille de Benamara et la famille Dif*

*Tous mes amis et toute la promotion 2020 et en particulier la  
promotion d'ELECTRONIQUE*

## Dédicace

*Je dédie ce travail aux personnes qui sont chères à mon cœur :*

*Mes parents sans qui, je ne suis pas là où j'en suis maintenant. Pour tous leurs sacrifices, leurs prières, leurs amours et leurs soutiens.*

*Mes frères pour leurs appuis et leurs encouragements permanents.*

*À tous ma famille pour leur soutien tout au long de mon parcours universitaire.*

*Mes profs ainsi que l'encadreur : LAHOUAOUI LALAOUI et tous mes profs qui enseignent moi au long de mon parcours universitaire.*

*Mes ami(e)s Ainsi que : DRIF ILYESS, MABRAK MAHDI et tous qui m'encouragent. Et une spéciale dédicace pour la famille BENAMARA.*

**FARES DIF**

# Sommaire

❖ Remerciement	
❖ Dédicace	
❖ Sommaire	
❖ Liste des figures.....	I
❖ Liste des tableaux.....	II
❖ Liste des symboles et abréviations .....	III

Introduction générale.....	2
<b>Chapitre I</b> .....	2
<i>Généralités sur le régulateur PID</i> .....	2
I.1. Introduction .....	3
I.2. Notions de base .....	3
I.2.a. Système .....	3
I.2.b. Régulation .....	<b>Erreur ! Signet non défini.</b>
I.2.c. Objectifs de la régulation.....	4
I.2.d. Conception des systèmes de commandes .....	4
I.2.e. Performances d'un système contrôlé :.....	6
I.2.f. Le correcteur.....	6
I.3. Régulation Proportionnelle Intégrale Dérivée PID .....	6
I.3.a. Bref historique de la régulateur PID.....	6
I.3.b. Structure d'un régulateur PID .....	7
I.3.c. Principe général d'un correcteur PID .....	7
I.3.d. Paramètres d'un régulateur PID .....	8
I.3.e. Formes de régulateur PID .....	9
I.3.f. Régulateur PID numérique .....	9
I.3.g. Domaine d'application et limites du régulateur PID .....	12
I.3.h. Les avantages et les inconvénients du régulateur PID.....	13
I.4. Conclusion.....	13
<b>Chapitre II</b> .....	15
<i>Les composants utilisant</i> .....	15
II.1 INTRODUCTION .....	16
II.2 ARDUINO.....	16
II.2.a Définition.....	16

II.2.b	Les types des cartes : il existe trois types .....	17
II.3	Arduino Nano .....	19
II.3.a	Caractéristiques principales.....	20
II.4	Comparaison entre les différentes cartes d'Arduino: .....	20
II.4.a	Partie Logicielle.....	21
II.5	LES CAPTEURS .....	23
II.5.a	Définition.....	23
II.5.b	Classification des capteurs .....	24
II.6	La LED .....	25
II.7	Les Résistances : .....	26
II.8	Condensateur :.....	<b>Erreur ! Signet non défini.</b>
II.8.a	Utilisation :.....	<b>Erreur ! Signet non défini.</b>
II.8.b	Capacité d'un condensateur :.....	27
II.9	Relais :.....	<b>Erreur ! Signet non défini.</b>
II.9.a	Symbole :.....	<b>Erreur ! Signet non défini.</b>
II.9.b	Le rôle : .....	28
II.10	Transistor :.....	28
II.10.a	LES TYPES DE TRANSISTOR.....	29
II.11	Vanne.....	29
II.11.a	Éléments d'une vanne .....	29
II.12	Pompe :.....	<b>Erreur ! Signet non défini.</b>
II.13	Conclusion :.....	33
<b>Chapitre III</b>	.....	34
<b>Simulation et résultat</b>	.....	34
III.1.	INTRODUCTION :.....	<b>Erreur ! Signet non défini.</b>
III.2.	Présentation générale de Proteus (ISIS):.....	35
III.2.a	Proteus :.....	35
III.2.b	L'éditeur de schéma ISIS : .....	35
III.2.c	Interface utilisateur :.....	36
III.2.d	Étapes de la saisie de schéma : .....	37
III.3.	Notre Carte de Commande en Puissance :.....	38
III.4.	Schéma fonctionnel :.....	38
III.5.	Fonctionnement :.....	<b>Erreur ! Signet non défini.</b>
III.5.a	Câblage de la carte : .....	39
III.5.b	Remarque : .....	<b>Erreur ! Signet non défini.</b>
III.6.	Simulation de 2 moteurs DC commandés par Arduino NANO : .....	39

III.6.a	Résultat de simulation : .....	40
III.7.	Conclusion :.....	<b>Erreur ! Signet non défini.</b>
<b>CONCLUSION GENERALE</b>	.....	44
<b>Annexe</b>	.....	<b>Erreur ! Signet non défini.</b>
<b>Références Bibliographiques</b>	.....	45

# Liste des figures

IFigure (I.1) : Les différents signaux relatifs à un système .....	3
Figure (I.2) : Schéma générale d'une commande en boucle ouverte. ....	4
Figure (I.3) : Schéma générale d'une commande en boucle fermée .....	5
Figure (I.4) : Correcteur PID .....	8
Figure (I.5) : conversion analogique numérique d'un signal.....	10
Figure (I.6) : structure approximative de régulateur PID .....	12
Figure (I.7) : comparaison entre la variation d'signal analogique et ces enchantions.....	12
Figure (II.1) : carte Arduino UNO .....	17
Figure (II.2) : Le logiciel de programmation.....	17
Figure (II.3) : Les types des cartes Arduino. ....	18
Figure (II.4) : Arduino NANO. ....	20
Figure (II.5) : IDE Arduino. ....	22
Figure (II.6) : Les boutons de l'interface Arduino. ....	22
Figure (II.7) : Un code minimal. ....	23
Figure (II.8) : Schéma d'un capteur. ....	23
Figure (II.9) : Image LED. ....	26
Figure (II.9) : La valeur d'une résistance est déterminée par ses bandes de couleurs.....	Z6
Figure (II.10) : Symbole de la résistance.....	27
Figure (II.11) : Condensateur .....	27
Figure (II.12) : Symboles des relais. ....	28
Figure (II.13) : LES TYPES DE TRANSISTOR .....	29
Figure (II.14) : Section d'une vanne à clapet.....	30
Figure (II.15) : Pompe Centrifuge 3 - 12V.....	31
Figure (II.16) : courbe caractéristique présente la relation entre HMT et débit volumique .....	32
Figure (II.17) : courbe caractéristique présente la relation entre Le rendement et le débit .....	33
Figure (III.1) : Interface utilisateur ISIS. IFigure (I.1): .....	36
Figure (III.2) : Organisation de la boite à outils ISIS.....	37
Figure (III.3) : des Etapes de la saisie de schéma.....	37
Figure (III.4) : représente la méthode de travail de microcontrôleur. ....	38
Figure (III.5) : électronique de la carte de commande de 2 moteurs connectés à Arduino .....	39
Figure (III.6) : schéma électronique présente le niveau 01 .....	40
Figure (III.7) : schéma électronique présente le niveau 02 .....	40
Figure (III.8) : schéma électronique présente le niveau 03 .....	41

# *Liste des Tableaux*

Tableau (I.1) :Différentes formes des régulateurs PID .....	9
Tableau (II.1) : les différentes cartes d'Arduino .....	21
Tableau (II.2) : Les capteurs actifs .....	24
Tableau (II.3) : Les capteurs passifs.....	25

# *Symboles et abréviations*

✓ *symboles*

<i>symbole</i>	<i>Signification</i>
$u(t)$	La sortie du contrôleur.
$u0$	La valeur d'offset.
$Kc$	Le gain du contrôleur.
$Yc(t)$	La consigne.
$Y(t)$	La mesure de la variable à réguler.
$PID$	Proportionnel Intégral Dérivé

# Introduction générale

De nos jours les techniques d'acquisition de données ont considérablement évolué en terme de rapidité, fiabilité et précision, en premier lieu, les capteurs qui bénéficient des nouvelles technologies, sont de plus en plus performants, en second lieu l'avènement des ordinateurs et microcontrôleurs qui, en permettant d'automatiser et de numériser des systèmes, offrant des gains important pour ce qui est de leurs traitement de l'information.

La commande numérique est une technique utilisant des données numériques pour représenter des instructions à la conduite d'une machine ou d'un procédé, d'une autre manière, elle est conçue pour piloter le fonctionnement d'une machine à partir des instructions d'un programme

Notre projet de fin d'étude consiste à concevoir et à réaliser un régulateur de même fonctionnalité mais facile à programmer et à maintenir. Pour ce faire notre régulateur est construit autour d'un microcontrôleur PIC qui présente un nombre important de ports d'entrée sortie et un espace mémoire très suffisant.

On s'intéresse dans le présent travail à l'étude de la régulation du niveau d'eau. Il s'agit de réaliser un système électronique qui commande la marche et l'arrêt des pompes de manière à automatiser les opérations de pompage et de remplissage.

Notre travail est présenté dans un mémoire organisé en trois chapitres :

Dans le premier chapitre, nous avons présenté des généralités sur le régulateur PID, , ses différentes structures, ainsi que des domaines d'applications, et enfin les avantages et les inconvénients de son utilisation.

Le second chapitre sera consacré à la présentation des composants utilisés, introduction sur la carte Arduino, ainsi la partie logiciel, les capteurs, LEDs, résistances, transistors...etc. Présentation générale de Proteus (ISIS et ARES), simulation par Arduino NANO, son résultat de simulation à chaque niveau d'une réponse en vitesse de moteur (pompe) et en position les LEDs (allumé ou fermé) feront l'objet du troisième chapitre.

# Chapitre I

*Généralités sur le régulateur PID*

## I.1. Introduction

Le terme de régulation est employé lorsqu'on cherche à combattre des perturbations afin de garder une valeur constante par exemple, une température, une pression, un débit ou une hygrométrie...

La régulation mesure en permanence par les capteurs le système à régler puis transmet ces informations au régulateur celui-ci compare cette mesure à la valeur désirée (la consigne) puis suivant son algorithme le régulateur va transmettre ses ordres aux actionneurs (vannes, volets, moteurs, etc.), afin de corriger les erreurs et conduire la sortie du système vers la consigne.

En industrie, les régulateurs PID répondent à plus de 90% de ces besoins. Par exemple, le nombre de régulateurs installés dans une usine industrielle se compte par milliers [3]. Dans ce chapitre nous allons présenter quelques notions sur la régulation puis nous présenterons des généralités sur le régulateur PID, ses différentes structures, ainsi que ses domaines d'applications, et enfin les avantages et les inconvénients de son utilisation.

## I.2. Notions de base

### I.2.a.1.1 Système

Un système est un ensemble de processus, physique et chimique, en évolution. Des actions sur le système (entrées) sont effectuées dans le but d'obtenir des réponses souhaitées (sorties). Les signaux qui véhiculent dans un système sont de deux types :

- ✓ **Signaux d'entrées** : Ils sont indépendants du système et peuvent être commandés (consignes) ou non commandés (perturbations).
- ✓ **Signaux de sorties** : Ils sont dépendants du système et du signal d'entrée. Pour évaluer les objectifs, ces signaux doivent être observables moyennant des capteurs.

La Figure (I.1) illustre un système à une entrée de commande, une sortie et une entrée de perturbation [2].

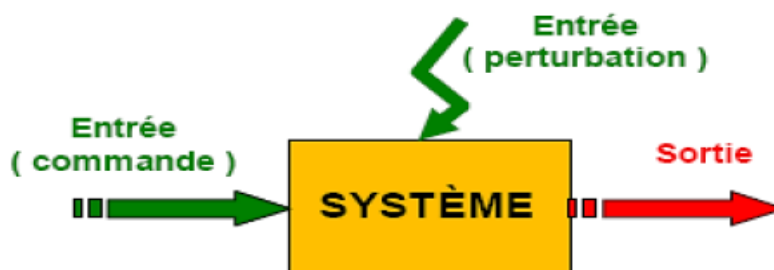


Figure (I.1): Les différents signaux relatifs à un système.

## I.2.a.1.2 Régulation

C'est l'ensemble des techniques utilisées visant à contrôler une grandeur physique. Pendant une régulation, on s'attachera à maintenir constante la grandeur réglée d'un système soumis à des perturbations. Dans un circuit de régulation, l'ensemble régulateur et système réglé doivent former une boucle fermée [5].

## I.2.a.1.3 Objectifs de la régulation

Le rôle de la régulation est multiple :

- ✓ Instrumentation du système : choisir les capteurs et actionneurs en fonction des besoins physiques, de coût et de performances demandées au système.
- ✓ Détermination des relations entrées-sorties du système,
- ✓ Modélisation pour la détermination de la structure mathématique des relations.
- ✓ Identification pour le calcul des coefficients du modèle [5].

## I.2.a.1.4 Conception des systèmes de commandes

On dit qu'un système physique est commandé lorsqu'il permet de réaliser une relation entre la grandeur d'entrée (cause) et la grandeur de sortie (effet). Quel que soit la nature du système à commander, il est toujours possible de classer les différentes structures de commande en deux grandes familles. Les structures de commande en boucle ouverte et les structures de commande à contre-réaction, appelées structures de commande en boucle fermée [2].

## I.2.a.1.5 La Commande en boucle ouverte

En l'absence d'entrées perturbatrices et en supposant que le modèle mathématique du système est parfait, il est concevable de générer un signal de commande produisant le signal de sortie souhaité (Figure (I.2)). Cela constitue le principe de la commande en boucle ouverte qui exploite la connaissance des dynamiques du système afin de générer les entrées adéquates ( $t$ ). Ces derniers ne sont donc pas influencés par la connaissance des signaux de sortie ( $t$ ). Cette solution est envisageable dans le cas où le système est parfaitement connu.

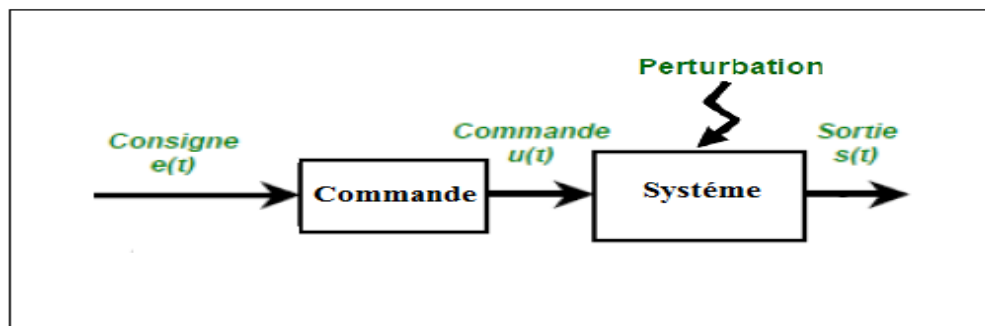


Figure (I.2): Schéma générale d'une commande en boucle ouverte.

Avec :

- **Ordre**: appelé aussi consigne ;
- **Action de commande** : action susceptible de changer l'état du système à commander, elle est élaborée en fonction des ordres
- **Sortie** : grandeur à observer, c'est une variable à contrôler
- **Perturbation** : variable aléatoire dont-on ne connaît pas l'origine.

**Avantage et inconvénient de cette conception** : Il s'agit d'un système insensible aux perturbations, rapide et stable. Ne revanche c'est un système, souvent, qualifié par aveugle, puisque c'est impossible de concevoir une correction[2].

#### I.2.a.1.6 La commande en boucle fermée

Si le système à commander n'est pas parfaitement connu ou si les perturbations affectent sensiblement le système, alors les signaux de sortie ne seront plus ceux souhaités. L'introduction d'un retour d'information sur la sortie mesurée s'avère alors nécessaire.

Un système bouclé permet, en quelque sorte, que la réponse du système correspond à l'entrée de référence. Tandis qu'un système en boucle ouverte commande sans contrôler l'effet de son action. Les systèmes de commande en boucle fermée sont ainsi préférables quand des perturbations non modélisables et/ou des variations imprévisibles des paramètres sont éventuelles. Cette structure de commande permet ainsi d'améliorer les performances dynamiques du système commandé (rapidité, rejet de perturbation, meilleur suivi des consignes, moindre sensibilité aux variations paramétriques du modèle, stabilisation des systèmes instables en boucle ouverte)[2].

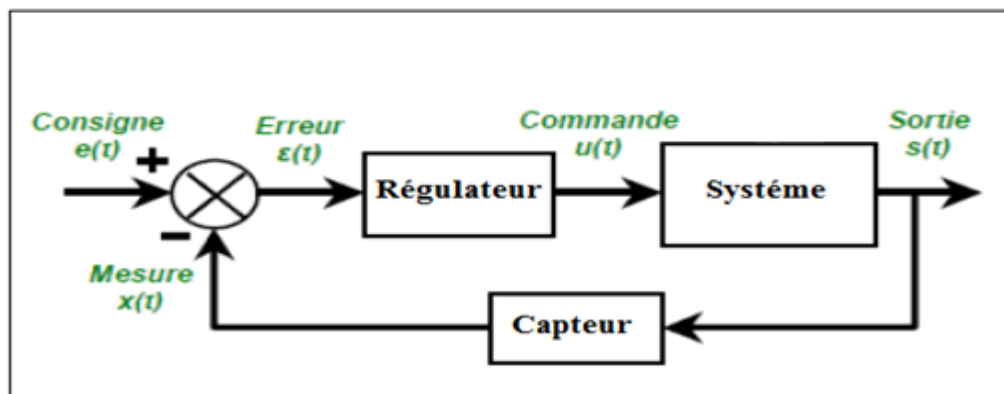


Figure (I.3) : Schéma générale d'une commande en boucle fermée.

**Avantage et inconvénient de cette conception** : Il s'agit d'un système précis, il y a une correction (sensible aux perturbations), mais peut rendre le système lent et instable.

#### I.2.a.1.7 Performances d'un système contrôlé :

Généralement le rôle d'un automaticien est de concevoir un système de régulation automatique régi par des performances exigées par un cahier de charge.

- ✓ **La Stabilité** : C'est la grandeur de sortie doit converger vers une valeur finie si le signal d'entrée est aussi limité.
- ✓ **La Précision** : La grandeur à mesurer doit être la plus proche possible de celle désirée à l'état statique.
- ✓ **La Rapidité** : Il doit répondre rapidement à une excitation.

#### I.2.a.1.8 Le correcteur

Le correcteur ou le régulateur constitue la partie « intelligente » de l'asservissement et sa détermination judicieuse confère à l'asservissement ses qualités. Aisé à modifier, le correcteur peut être muni d'une variation automatique de ses paramètres suivant la plage de fonctionnement du procédé, dans le cas où celle-ci évolue lentement. [10]

### I.3. Régulation Proportionnelle Intégrale Dérivée – PID

La commande PID est dite aussi (correcteur, régulateur, contrôleur), se compose de trois termes P, I et D, d'où le 'P' correspond au terme proportionnel, 'I' pour terme intégral et 'D' pour le terme dérivé de la commande.

Les régulateurs PID sont probablement les plus largement utilisés dans le contrôle industriel, même les plus complexes systèmes de contrôle industriel peut comporter un réseau de contrôle dont le principal élément de contrôle est un module de contrôle PID.

Le régulateur PID est une simple implémentation de retour d'information (Feedback). Il a la capacité d'éliminer la compensation de l'état d'équilibre grâce à l'action intégrale, et il peut anticiper le futur grâce à une action dérivée, [8].

#### I.3.a.1.1 Bref historique de la régulateur PID

Les régulateurs de type (Proportionnel : P, Intégral : I et Dérivé : D) ou PID se révèlent suffisants pour résoudre un grand nombre de problèmes de contrôle et ce particulièrement lorsque la dynamique du système est bénigne et que les exigences en terme de performances sont modestes. Les régulateurs PID répondent à plus de 95% des besoins industriels. Malgré l'expérience acquise au fil des ans, les valeurs choisies, pour les paramètres P, I et D, ne sont pas toujours satisfaisantes, ni adaptées au processus à régler.

L'histoire des régulateurs est déjà longue et il peut être intéressant de rappeler quelques étapes importantes. Les premiers régulateurs de type centrifuge apparaissent vers 1750 pour régler la

vitesse des moulins à vent, suivi en 1788 du fameux contrôleur de vitesse d'une machine à vapeur de James Watt. En 1942, Ziegler et Nichols ont proposé deux démarches permettant de trouver facilement les paramètres optimums pour une installation donnée.

Au fil des ans, les propositions de Ziegler et Nichols ont été adaptées ou modifiées selon les besoins. En 1963, Horowitz a ajouté un degré de liberté supplémentaire au régulateur PID afin de mieux contrôler les dépassements obtenus lors d'une réponse indicielle. Ce nouveau degré de liberté consiste, en particulier, à ne réinjecter vers le terme proportionnel qu'une partie du signal de sortie.

Au début des années 1990 et dans le but de fournir des règles d'ajustement simples mais plus performantes que celles de Ziegler-Nichols, Åström et ses collaborateurs ont analysé le comportement dynamique d'un grand nombre de processus. Cette analyse a conduit à l'établissement de tableaux servant aux calculs des paramètres P, I et D à partir de mesures simples.

#### I.3.a.1.2 Structure d'un régulateur PID

L'idée de base d'un contrôleur PID est de lire un capteur, puis de calculer la sortie souhaitée de l'actionneur en calculant les réponses proportionnelle, intégrale et dérivée, puis en faisant la somme de ces trois composants pour calculer la sortie. Avant de définir les paramètres d'un contrôleur PID [3].

La réalisation de la boucle d'asservissement par un PID comporte deux aspects essentiels :

- □ Le réglage du régulateur PID, pour lequel la connaissance d'un modèle dynamique du procédé d'une part et les performances désirées d'autre part déterminent le choix de la méthode de synthèse.
- □ L'implantation du régulateur dans une version analogique ou numérique et dans une configuration série, parallèle ou mixte.[4]

#### I.3.a.1.3 Principe général d'un correcteur PID :

L'erreur observée est la différence entre la consigne et la mesure. Le PID permet trois actions en fonction de cette erreur :

- ✓ Une action Proportionnelle : l'erreur est multipliée par un gain  $K_p$
- ✓ Une action Intégrale : l'erreur est intégrée sur un intervalle de temps  $s$ , puis multipliée par un gain  $K_i$
- ✓ Une action Dérivée : l'erreur est dérivée suivant un temps  $s$ , puis multipliée par un gain  $K_d$

Les actions dérivées et intégrales ne s'emploient jamais seules mais en combinaison avec l'action proportionnelle [7].

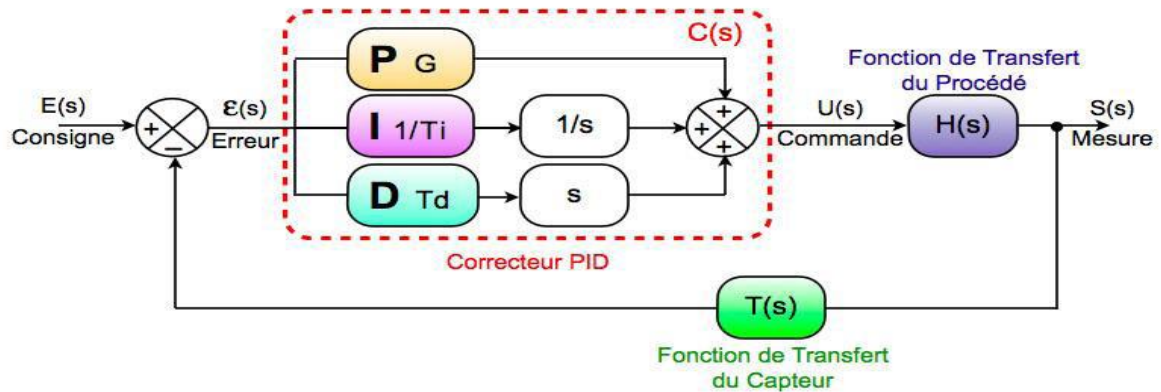


Figure (I.4): Correcteur PID.

I.3.a.1.4 Paramètres d'un régulateur PID

L'idée de base de ce régulateur est de générer une commande  $u(t)$  donnée par le régulateur PID, dans sa forme classique est décrite par l'équation (I.1). Elle est composée de la somme de trois termes :

- Le terme proportionnel « P » (proportionnel à l'erreur).

$K_P$ : est le gain proportionnel

$$P = K_P e(t) \quad (I.1)$$

- Le terme intégral « I » (proportionnel à l'intégrale de l'erreur) :

$T_i$  : est la constante de temps de l'action intégrale, intégrale, en secondes ou en minutes. Peut être réglé par son inverse (répétitions par seconde ou par minute).

$$I = K_P \frac{1}{T_i} \int_0^t e(t) dt \quad (I.2)$$

- Le terme dérivatif « D » (proportionnel à la dérive de l'erreur).

$T_d$  : est la constante de temps de la partie dérivée

$$D = K_P T_d \frac{e(t)}{T_i dt} \quad (I.3)$$

Les paramètres du régulateur PID sont le gain proportionnel  $K_P$ , le temps intégral  $T_i$ . Et le temps dérivatif  $T_d$ , les temps étant exprimés en secondes [1].

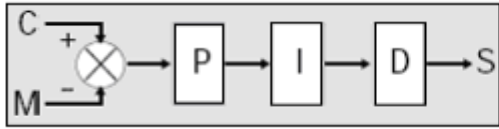
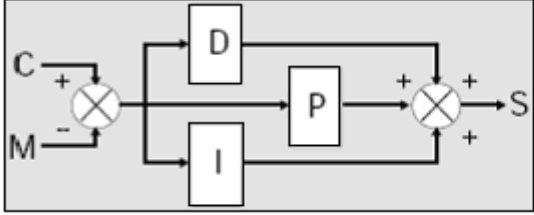
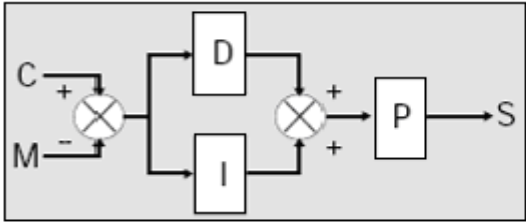
La fonction de transfert avec la transformée de Laplace du régulateur PID parallèle est la somme des trois termes :

$$C(s) = K_P + \frac{1}{s.T_i} + T_d \cdot s \quad (I.4)$$

I.3.a.1.5 Formes de régulateur PID :

Dans un régulateur PID, il existe plusieurs façons d'associer les paramètres P, I et D. en effet, le correcteur PID peut avoir une structure série, parallèle ou mixte, illustré dans le tableau suivant.(Voir le tableau I-1). [9]

Tableau (I.1) : Différentes formes des régulateurs PID.

Formes de régulateur	Schéma et fonction transfert
<i>Série</i>	 $K_P \left( \frac{T_i + T_d}{T_i} + \frac{1}{pT_i} + pT_d \right)$
<i>Parallèle</i>	 $K_P + \frac{1}{pT_i} + pT_d$
<i>Mixte</i>	 $K_P \left( 1 + \frac{1}{pT_i} + pT_d \right)$

I.3.a.1.6 Régulateur PID numérique [3]

Dans la plupart des documents sur le contrôle numérique, on s'initie rapidement à la transformation z et au plan z, par opposition à au plan s, que est principalement utilisé pour le lieu des racines dans la conception des contrôleurs PID.

Avec le contrôle numérique, nous connectons un ordinateur (microcontrôleur) au monde extérieur (analogique).

L'actionneur, l'installation et le capteur sont de véritables appareils dans le monde réel (analogique). La valeur réelle étant contrôlé (C) est détecté par un capteur qui, ici, est modélisé comme un simple gain. Habituellement, le capteur produit un signal électrique ; souvent une

tension ; qui est ensuite envoyé au calculateur via un convertisseur analogique-numérique (A / D). Là, le signal original, que ce soit la température, la pression, la position, la vitesse ou que ce soit, est récupéré sous forme numérique à l'intérieur de calculateur en divisant la tension représentative grâce au gain du capteur. Ceci est représenté par  $C_d$  dans le diagramme, le «d» indiquant que c'est la représentation numérique de la variable réelle détectée  $C$ , à ce stade, l'algorithme pour la comparaison avec la valeur souhaitée ( $R$ ) peut avoir lieu, l'erreur déterminée en format numérique et l'algorithme de contrôleur exécuté pour arriver à la version numérique de la sortie analogique à envoyer à l'actionneur. Ceci est converti par le convertisseur N / A en un signal réel ( $U$ , généralement une tension), qui démarre la chaîne d'actionnement.

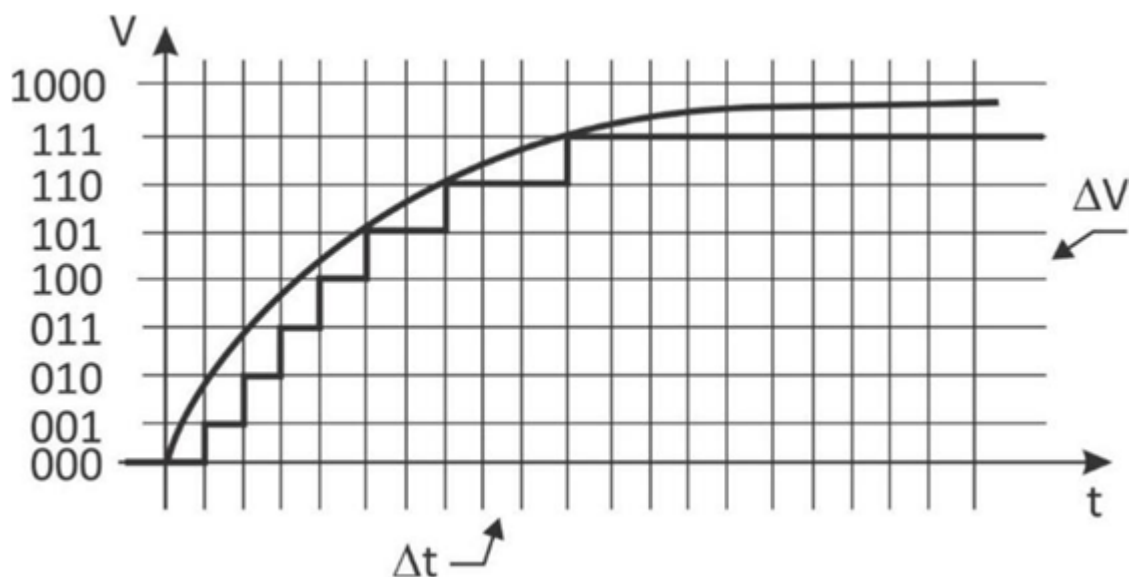


Figure (I.5): conversion analogique numérique d'un signal.

Comme on peut le voir avec une comparaison minutieuse, aux instants de temps indiqués, les bits seront activés pour représenter que la tension analogique réelle est arrivée au niveau de tension correspondant à un bit. Ainsi, la tension détectée est représentée graphiquement comme forme de courbe en escalier indiquée sous la courbe de tension analogique. Bien sûr, 3 bits ne donnent pas une bonne résolution de la courbe analogique. Trois bits binaires donnent  $2^3 = 8$  valeurs possibles. Généralement, plus de bits sont utilisés pour donner une résolution plus élevée des valeurs analogiques. Dix, par exemple, donnerait  $2^{10} = 1024$  niveaux différents, et il est facile de voir que 1024 niveaux permettraient au courbe pour être représenté avec une meilleure fidélité. Trois bits ont été utilisés ici simplement pour montrer la nature du problème de la représentation numérique d'une courbe analogique.

### I.3.a.1.7 La structure de régulateur PID numérique :

Le contrôle numérique est implémenté avec des microcontrôleurs spécialisés développés pour une utilisation industrielle renforcée. Avec l'arrivée de petits microprocesseurs qui sont peu coûteux, la réalisation des régulateurs PID adéquat pour de nombreuses applications mécaniques, thermiques, ou des systèmes de fluide est disponible.

Voici comment. Le programme de contrôle du microcontrôleur s'exécute souvent dans une boucle à pas de temps fixe. Cette boucle est très rapide, particulièrement comparé aux temps de réaction de la plupart des systèmes mécaniques. Les boucles typiques s'exécutent une fois toutes la 1-10 milliseconde. Généralement, cela suffit pour contrôler la plupart des systèmes mécaniques par exemple.

La commande numérique présente de nombreux avantages par rapport à la commande analogique (commande avec des composants électriques standards). Mais le contrôle analogique bat toujours le contrôle numérique dans cette zone critique de vitesse. Les contrôleurs analogiques répondent pratiquement sans délai, ce qui est nécessaire pour les systèmes très rapides [4].

Comme indiqué ci-dessus, les contrôleurs numériques exécutent un programme en boucle qui fournit la commande. Le schéma figure I.5 montre la structure approximative de ce programme informatique. Ces trois étapes sont effectuées, l'un après l'autre, chaque fois que l'automate tourne en mode automatique.

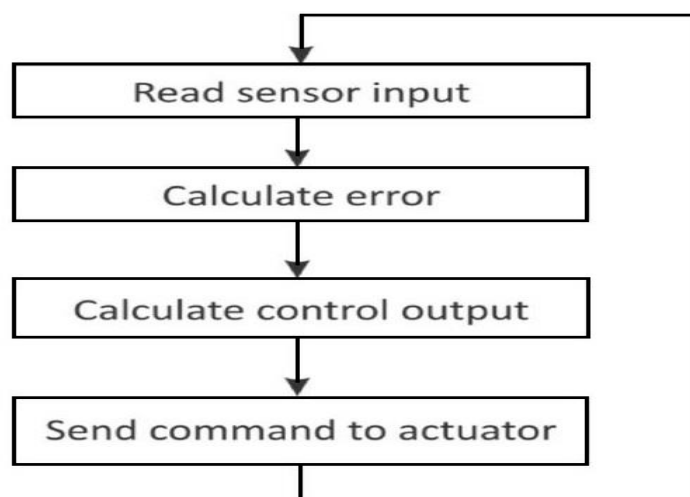


Figure (I.6) : structure approximative de régulateur PID.

### I.3.a.1.8 Problèmes rencontrés avec le régulateur numérique

Un problème inhérent au contrôle numérique est que le taux de balayage du régulateur peut être trop lent pour le processus qu'il essaie de contrôler. Les processus thermiques sont généralement lents, donc ce problème ne se rencontre pas là. Mais les processus d'actionnement hydraulique peuvent être très rapides et là, le régulateur doit suivre l'évolution de la nature du processus. Si la fréquence de balayage du contrôleur est trop lente, des changements dans le

processus peuvent être omis entre les instances d'échantillonnage du capteur. La figure illustre un scénario possible de ce type.

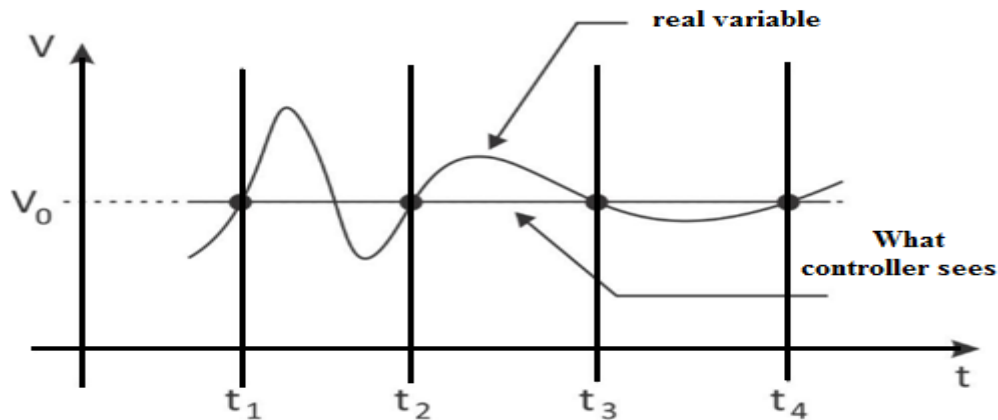


Figure (I.7): comparaison entre la variation d'un signal analogique et ses échantillons.

Dans ce cas, même si la variable échantillonnée change de haut en bas, il semble qu'elle se trouve à niveau constant parce que les instances d'échantillonnage l'attrapent alors qu'elle croise une valeur constante. Le tableau de bord affichant la valeur de  $V$ , et ne montrerait donc aucun changement de  $V$ , même si  $V$  est en fait éprouvant une excursion assez active de  $V_0$ .

Bien que ce soit un peu un exemple artificiel, c'est facile de voir que la première valeur de crête serait oubliée ou mesurée comme étant inférieure à moins que l'exemple vienne de coïncider avec l'occurrence du pic. De toute évidence, la solution à ce problème est de réduire, le taux de balayage du microprocesseur.

#### I.3.a.1.9 Domaine d'application et limites du régulateur PID

Dans le monde industriel, le régulateur PID est largement employé. Bien que relativement simple, cet algorithme soutient favorablement la comparaison avec des algorithmes plus sophistiqués, tant au point de vue de la performance que de la robustesse de la régulation vis-à-vis de non-linéarités ou variation du procédé. Par ailleurs, ce type d'algorithme est facile à implanter, ce qui plaide en sa faveur à l'époque des régulateurs pneumatiques ou même électroniques. C'est moins vrai aujourd'hui où derrière chaque régulateur se cache un microprocesseur.

Une qualité du PID est qu'il est facile à comprendre donc facile à régler et à maintenir. Ces raisons expliquent pourquoi il est si largement employé, le plus souvent sous la forme PI, l'action D est souvent mise à zéro.

Il existe cependant des cas pour lesquels le régulateur PID n'est pas bien adapté. Nous avons vu que lorsque le procédé a un retard important, les performances sont mauvaises. C'est le cas aussi lorsque le système est oscillant [4].

#### I.3.a.1.10 Les avantages et les inconvénients du régulateur PID

L'asservissement par PID est aujourd'hui l'un des asservissements les plus utilisés et ce pour plusieurs raisons. Premièrement, il est très simple à mettre en place et s'avère efficace pour la plupart des systèmes réels. De plus, le calcul des coefficients laisse le choix entre plusieurs méthodes de difficulté croissante. D'une part, une méthode expérimentale très facile à mettre en place, permet d'obtenir rapidement des coefficients corrects pour des systèmes ne nécessitant pas de très grandes précisions dans l'asservissement. D'autre part, des méthodes mathématiques avancées offrent des techniques pour obtenir les coefficients idéaux pour un système en particulier. Ainsi, la mise en place d'un asservissement PID peut-être à la fois rapide et efficace et permettra une optimisation des coefficients pour les systèmes les plus avancés. Cependant, il est important de noter que ce type d'asservissement est limité par un certain nombre de contraintes. Tout d'abord, il peut s'avérer inefficace pour certains systèmes qui contiennent du bruit (coefficient Dérivé) ou qui ne sont pas linéaires (l'asservissement PID étant linéaire, la non-linéarité d'un système entraîne des instabilités). Enfin, il est possible d'optimiser la réponse d'un système en multipliant les asservissements (Comme par exemple le double asservissement PID) [4].

## I.4. Conclusion

Dans chaque système de la vie quotidienne, les systèmes de contrôle ont l'essentiel pour gérer leurs différents appareils et environnements. En général, les régulateurs numériques sont utilisés dans presque toutes les applications de l'entreprise avec d'autres périphériques essentiels tels que les microprocesseurs, les ordinateurs et l'électronique numérique. Ensuite, nous pouvons trouver un système de contrôle numérique dans des systèmes d'application électriques, mécaniques ou chimiques. Dans ce chapitre, une étude sur les contrôleurs PID a été introduite pour explorer leurs structures avec le développement nécessaire, que ce soit dans le processus de réglage de contrôleur PID ou sur leurs modifications de structure) [3]. Le chapitre qui va suivre est consacré pour un outil d'application de ces systèmes de régulation et de contrôle.

# Chapitre II

*Les composants utilisant*

## II.1 INTRODUCTION :

Dans ce chapitre nous avons présenté une généralité sur L'Arduino ainsi ces différents types, puis ces exemples sur des modèles d'Arduino et leurs caractéristiques principales. Par la suite, on intéresse a des parties logiciel dans la carte Arduino et langage de programmation.

Finalement nous avons présentées différents composants utilisés pour réaliser ce montage électrique (capteur, résistance, condensateur, transistor, LEDs ...).

## II.2 ARDUINO

### II.2.a Définition [12] :

Le système Arduino donne la possibilité d'allier les performances de la programmation à celles de l'électronique. Plus précisément, pour programmer des systèmes électroniques. Le gros avantage de l'électronique programmée c'est qu'elle simplifie grandement les schémas électroniques et par conséquent, le coût de la réalisation, mais aussi la charge de travail à la conception d'une carte électronique.

Le système Arduino permet de :

- contrôler les appareils domestiques
- fabriquer votre propre robot
- faire un jeu de lumières
- communiquer avec l'ordinateur
- télécommander un appareil mobile (modélisme)

Le système Arduino est composé de deux choses principales : Le matériel et le logiciel.

#### II.2.a.1 Le matériel :

Il s'agit d'une carte électronique basée autour d'un microcontrôleur Atmega du fabricant Atmel, dont le prix est relativement bas pour l'étendue possible des applications.



Figure (II.1): carte Arduino UNO

### II.2.a.2 Le logiciel :

Le logiciel permet de programmer la carte Arduino. Il offre une multitude de fonctionnalités.

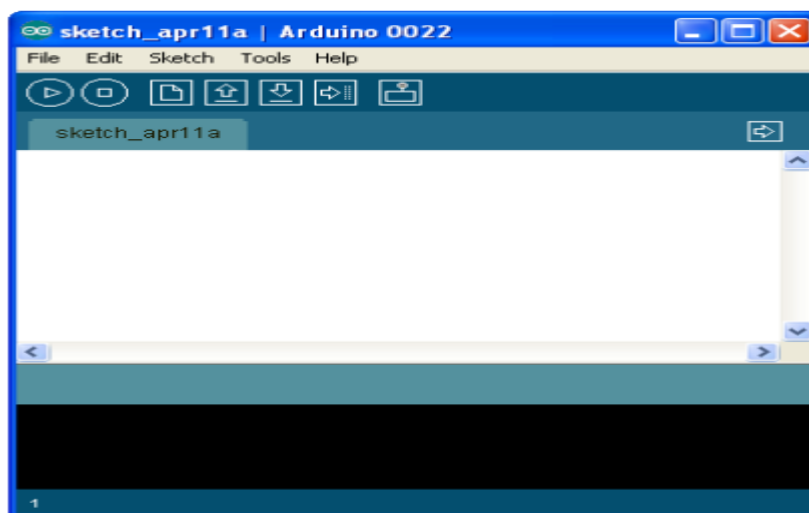


Figure (II.2): Le logiciel de programmation .

### II.2.b Les types des cartes : il existe trois types

- Les « officielles » qui sont fabriquées en Italie par le fabricant officiel : Smart Projects.
- Les « compatibles » qui ne sont pas fabriqués par Smart Projects, mais

qui sont totalement compatibles avec les Arduino officielles

- Les « autres » fabriquées par diverse entreprise et commercialisées sous un nom différent (Freeduino, Seeduino, Femtoduino, ...).

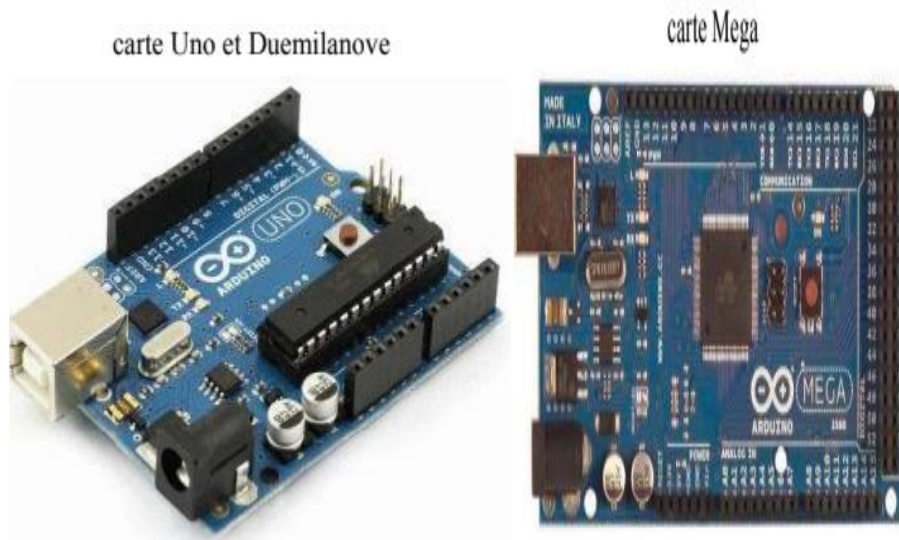


Figure (II.3): Les types des cartes Arduino.

La carte Arduino est équipée d'un microcontrôleur. Le microcontrôleur est un composant électronique programmable. On le programme par le biais d'un ordinateur grâce à un langage informatique, souvent propre au type de microcontrôleur utilisé.

Un microcontrôleur est constitué par un ensemble d'éléments qui ont chacun une fonction bien déterminée. Il est en fait constitué des mêmes éléments que sur la carte mère d'un ordinateur :

### II.2.b.1 La mémoire

Il en possède de 5 types :

- **La mémoire Flash** : C'est celle qui contiendra le programme à exécuter. Cette mémoire est effaçable et réinscriptible.
- **RAM** : c'est la mémoire dite "vive", elle va contenir les variables de votre programme. Elle est dite "volatile" car elle s'efface si on coupe l'alimentation du microcontrôleur.
- **EEPROM** : C'est le disque dur du microcontrôleur. Vous pourrez y enregistrer

des infos qui ont besoin de survivre dans le temps, même si la carte doit être arrêtée. Cette mémoire ne s'efface pas lorsque l'on éteint le microcontrôleur ou lorsqu'on le reprogramme.

- **Les registres** : c'est un type de mémoire utilisé par le processeur.
- **La mémoire cache** : c'est une mémoire qui fait la liaison entre les registres et la RAM.

### **II.2.b.2 Le processeur**

C'est le composant principal du microcontrôleur. C'est lui qui va exécuter le programme qu'on lui donnera à traiter. On le nomme souvent le CPU.

Pour que le microcontrôleur fonctionne, il lui faut une alimentation ! Cette alimentation se fait en générale par du +5V. D'autres ont besoin d'une tension plus faible, du +3,3V.

En plus d'une alimentation, il a besoin d'un signal d'horloge. C'est en fait une succession de 0 et de 1 ou plutôt une succession de tension 0V et 5V. Elle permet en outre de cadencer le fonctionnement du microcontrôleur à un rythme régulier. Grâce à elle, il peut introduire la notion de temps en programmation.

### **II.2.b.3 Le logiciel**

Au jour d'aujourd'hui, l'électronique est de plus en plus remplacée par de l'électronique Programmée. On parle aussi d'électronique embarquée ou d'informatique embarquée.

### **II.2.b.4 L'interface**

L'interface du logiciel Arduino se présente de la façon suivante. Il existe plusieurs modèles des cartes Arduino, des modèles différents pour la variation de leurs caractéristiques et leurs applications, par exemple la carte Arduino Nano que nous allons l'utiliser dans ce mémoire.

## **II.3 Arduino Nano:[16]**

Arduino NANO La carte Arduino Nano est basée sur un ATmega328 cadencé à 16 MHz. Sa mémoire de 32 KB et son grand nombre d'E/S font de ce circuit compatible DIL30 un élément idéal pour les systèmes embarqués ou pour des applications robotiques nécessitant du multitâches. La Nano 3.0 peut se programmer avec le logiciel Arduino. Le contrôleur ATmega328 contient un boot loader qui permet de modifier le programme sans passer par un programmeur.

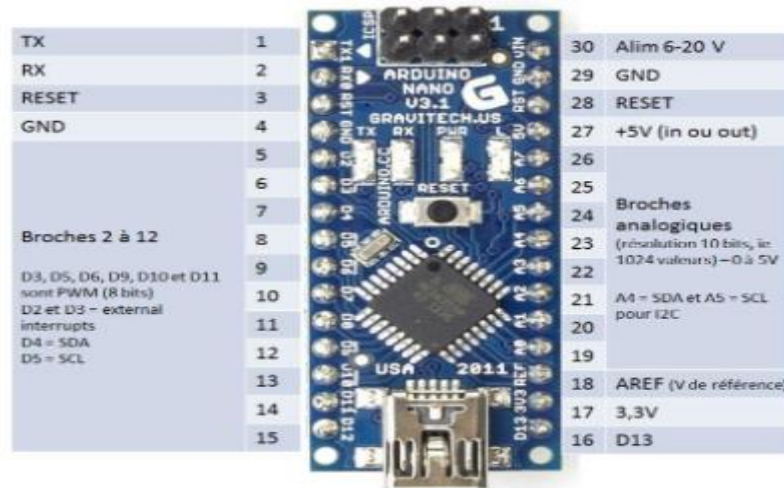


Figure (II.4): ArduinoNANO.

### II.3.a Caractéristiques principales

- Alimentation via port USB ou 5 Vcc régulée sur broche 27 ou 6 à 20 V non régulée sur broche 30
- Microprocesseur: ATmega328
- Mémoire flash: 32 kB
- Mémoire SRAM: 2 kB
- mémoire EEPROM: 1 kB
- 14 broches d'E/S dont 6 PWM
- 8 entrées analogiques 10 bits
- intensité par E/S: 40 mA
- cadencement: 16 MHz
- bus série, I2C et SPI
- gestion des interruptions
- fiche USB: mini-USB B
- boîtier DIL30
- dimensions : 45 x 18 x 18 mm

## II.4 Comparaison entre les différentes cartes d'Arduino:

Le tableau suivant nous montre des exemples sur des modèles d'Arduino et leurs caractéristiques principales

Tableau (II.1): les différentes cartes d'Arduino.

	UNO	LEONARDO	MEGA 2560	NANO		DUE	YUN
Micro-Contrôleur	ATmega 328P	ATmega 32U4	ATmega 2560	-ATmega 168	-ATmega 328P	ATSAM3X8E	ATmega32U4 AR9331 Linux
Fréquence D'horloge	16 MHz	16 MHz	16 MHz	16 MHz		84 MHz	16 MHz 400 MHz
Tension service/entrée	3.3V ; 5 V / 7-12 V	5 V / 7-12 V	5 V / 7-12 V	5 V / 7-9 V		3.3 V / 7-12 V	5 V / 5 V 3.5 V / 5 V
Ports Numériques ES	14	20	54	14		54	20
Ports Analogiques E/S	6/0	12/0	16/0	8/0		12/2	12/0
Courant max par broche d'E/S (cc)	40 mA	40 mA	40 mA	40 mA		800 mA	40mA
Courant max par broche 3.3 V	50 mA	50 mA	50 mA	-		800 mA	50 mA
Mémoire (Ko) Flash / SRAM / EEPROM	32 / 2 / 1	32 / 2.5 / 1	256 / 8 / 4	ATmega 168 : 16 / 1 / 0.512	ATmega 328 : 32 / 2 / 1	512 / 94 / 0.512	32 / 2.5 / 1 64Mo / 16Mo
Dimension (cm)	6.86 x 5.3	6.86 x 5.3	10.16 x 5.3	1.9 x 4.3		10.2 x 5.3	7 x 5.3
Prix (euro) (approximatif)	24	20	47	40		47	62

## II.4.a Partie Logicielle [13]

### II.4.a.1 IDE Arduino

Un IDE (environnement de développement) libre et gratuit est distribué sur le site d'Arduino (compatible Windows, Linux et Mac) à l'adresse. D'autres alternatives existent pour développer l'Arduino (extensions pour Code Blocks, Visual Studio, Eclipse, XCode, etc.) mais nous n'aborderons dans ce chapitre que l'IDE officiel. L'interface de l'IDE Arduino est plutôt simple (Figure II.5), il offre une interface minimale et épurée pour développer un programme sur les cartes Arduino. Il est doté d'un éditeur de code avec coloration syntaxique (1) et d'une barre d'outils rapide (2). Ce sont les deux éléments les plus importants de l'interface, c'est ceux que l'on utilise le plus souvent. On retrouve aussi une barre de menus (3) plus classique qui est utilisé pour accéder aux fonctions avancées de l'IDE. Enfin, une console (4) affichant les résultats de la compilation du code source, des opérations sur la carte, etc.

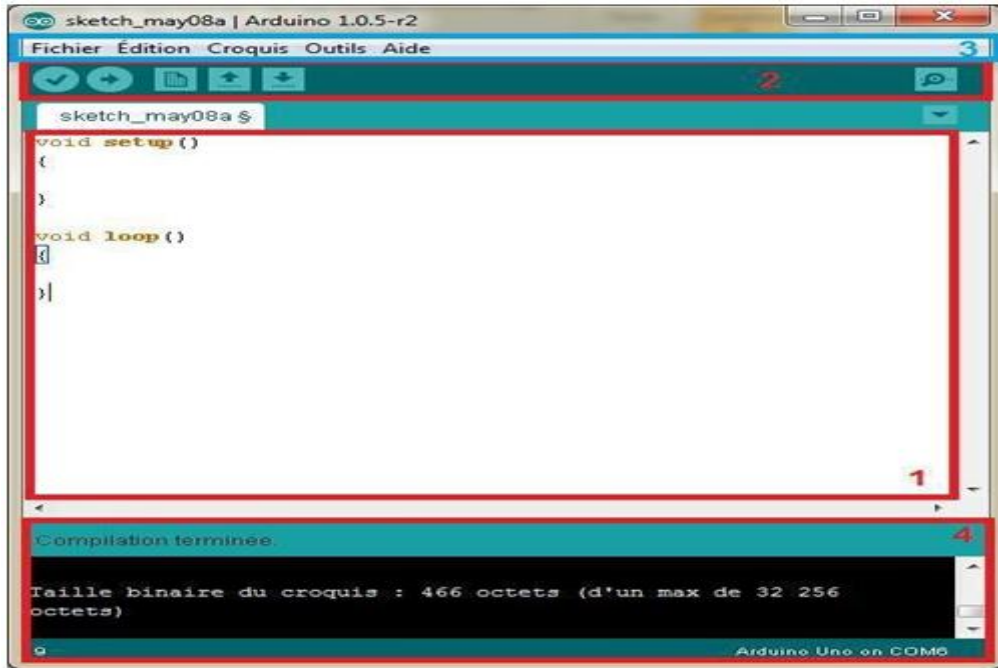


Figure (II.5) : IDE Arduino.

**II.4.a.2 LES BOUTONS : [11]**

Nous allons parler des fonctions des différents boutons du logiciel Arduino que nous avons encadrés et nommés avec des chiffres de 1 à 6.

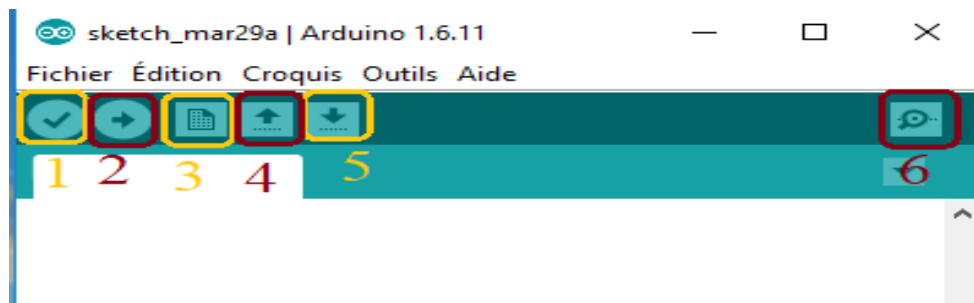


Figure (II.6): Les boutons de l'interface Arduino.

[1]	Permet de vérifier le programme, il actionne un module qui cherche les erreurs dans votre programme
[2]	Charge (télé verse) le programme dans la carte
[3]	Crée un nouveau fichier
[4]	Ouvre un fichier Bouton
[5]	Enregistre le fichier
[6]	Ouvre le moniteursérie

### II.4.a.3 Langage Arduino [13]

Le langage Arduino est inspiré de plusieurs langages. On retrouve notamment des similarités avec le C, le C++, le Java et le Processing. Le langage impose une structure particulière typique de l'informatique embarquée. La fonction `setup` (voir Fig.8) contiendra toutes les opérations nécessaires à la configuration de la carte (directions des entrées sorties, débits de communications série, etc.). La fonction `loop`, elle est exécutée en boucle après l'exécution de la fonction `setup`. Elle continuera de boucler tant que la carte n'est pas mise hors tension, redémarrée (par le bouton *reset*). Cette boucle est absolument nécessaire sur les microcontrôleurs étant donné qu'il n'y a pas de système d'exploitation. En effet, si l'on omettait cette boucle, à la fin du code produit, il sera impossible de reprendre la main sur la carte Arduino qu'exécuterait alors du code aléatoire.

```

void setup()           //fonction d'initialisation de la carte
{
    //contenu de l'initialisation
}

void loop()           //fonction principale, elle se répète
(s'exécute) à l'infini
{
    //contenu de votre programme
}

```

Figure (II.7): Un code minimal.

## II.5 LES CAPTEURS :

Un capteur est un dispositif ayant pour tâche de transformer une mesure physique observée en une mesure généralement électrique qui sera à son tour traduite en une [13]

### II.5.a Définition : [12]

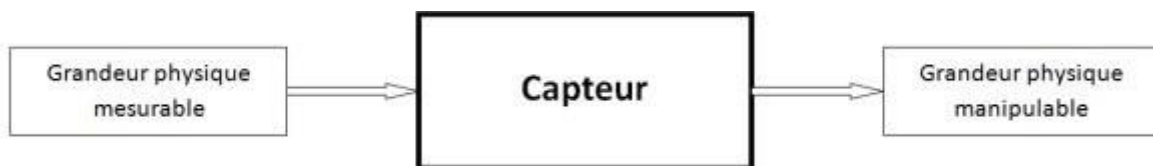


Figure (II.8): Schéma d'un capteur.

#### II.5.a.1 Principes des capteurs :

Avant de donner la définition d'un capteur, il est nécessaire de connaître quelques définitions de métrologie.

Tout d'abord, il faut savoir que mesurer une grandeur physique c'est attribuer une valeur quantitative en prenant pour référence une grandeur de même nature appelée

unité. Ensuite d'autres définitions doivent être connues comme

- ✓ Le mesura de : C'est l'objet de la mesure ou plus simplement la grandeur à mesurer.
- ✓ Le mesurage : C'est l'ensemble des opérations pour déterminer la valeur du mesurage.
- ✓ La mesure c'est le résultat du mesurage. Autrement dit c'est la valeur du mesurage.

Ces définitions permettent de donner une définition claire d'un capteur. En effet un capteur est un dispositif dont les caractéristiques physiques sont sensibles à un mesurage. Lorsque celui-ci est soumis à ce mesurage il fournit une réponse sous la forme d'une grandeur physique exploitable qui est en général de nature électrique.

## II.5.b Classification des capteurs

On peut classer les capteurs de plusieurs manières :

- ✓ par le mesurage qu'il traduit (capteur de position, de température, de pression, etc.)
- ✓ par son rôle dans le processus industriel (contrôle de produit finis, de sécurité, etc.)
- ✓ par le signal qu'il fournit en sortie qui peut être numérique ou analogique
- ✓ par leur principe de traduction du mesurage (capteur résistif, piézoélectrique, etc.)
- ✓ par leur principe de fonctionnement : capteur Actif ou Passif

### II.5.b.1 Les capteurs actifs :

Ce capteur fonctionne comme un générateur ; dès qu'il est soumis à l'action d'un mesurage celui-ci transforme celle-ci en une grandeur directement exploitable à savoir en énergie électrique.

Le tableau suivant donne les principes physiques les plus utilisés en fonction d'un mesurage :

*Tableau (II.2): Les capteurs actifs.*

Mesurande	Énergie propre du mesurande	Principe physique	Grandeur de sortie
Température	Énergie thermique	Effet thermoélectrique Effet pyroélectrique	Tension Charge
Flux lumineux	Énergie électromagnétique	Effet photoémissif Effet photovoltaïque Effet photoélectrique	Courant Tension Tension
Force Pression Accélération	Énergie mécanique	Effet piézoélectrique	Charge
Vitesse	Énergie mécanique	Effet d'induction électromagnétique	Tension
Position	Énergie mécanique	Effet Hall	Tension

### II.5.b.2 Les capteurs passifs :

Un capteur passif est considéré comme une impédance dont l'un des paramètres est sensible au mesurage. Cette impédance doit ensuite être intégrée dans un circuit

pour pouvoir retrouver une grandeur électrique en sortie. Le montage qui permet ceci est appelé conditionneur. Il existe plusieurs sortes de conditionneur comme le montage potentiométrique, le pont de Wheatstone, les circuits oscillants ou les amplificateurs opérationnels. Le tableau suivant donne différents capteurs passifs

*Tableau (II.3): Les capteurs passifs.*

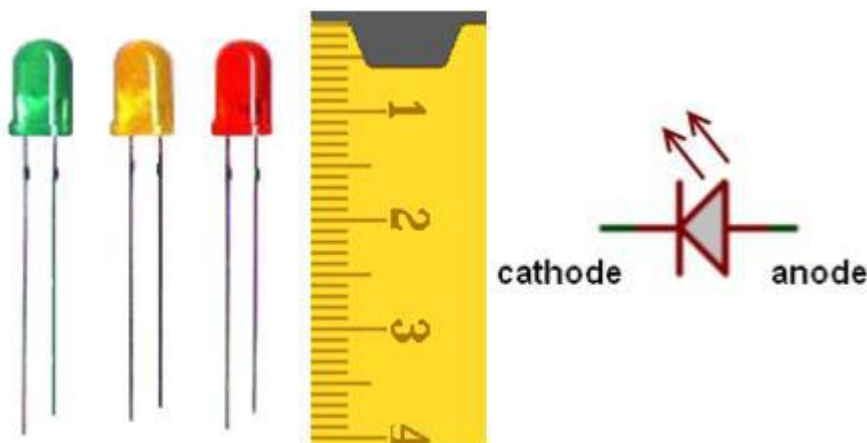
Mesurande	Caractéristiques électriques sensibles	Types de matériaux utilisés
Température	Résistivité	Métaux : platine, nickel, cuivre
Très basse température	Constante diélectrique	Verres
Flux lumineux	Résistivité	Semi-conducteur
Déformation	Résistivité Perméabilité magnétique	Alliages de nickel, silicium dopé Alliages ferromagnétiques
Position	Résistivité	Matériaux magnéto-résistants : bismuth, antimoine d'indium
Humidité	Résistivité Constante diélectrique	Chlorure de lithium Alumine, polymères
Niveau	Constante diélectrique	Liquides isolants

**II.5.c Capteur de niveau d'eau :** un capteur de reconnaissance de niveau de l'eau. Fournit un signal analogique directement par la carte Arduino

## II.6 La LED: [11]

DEL en français (diode électroluminescente) est un composant optoélectronique qui émet de la lumière lorsqu'elle est parcourue par un courant électrique dans le sens direct. C'est-à-dire le courant doit la traversée de l'anode (la tension plus élevée) vers la cathode.

Elle a une vaste gamme de couleurs en fonction du semi-conducteur (GaAsp, AlGaAS, GaN, GaP, InGaN, SiC, ...) utilisé et de la longueur d'onde (infrarouge si elle est >700nm, Ultraviolet <400nm et les autres couleurs entre les deux). La tension de seuil d'une LED dépend de sa couleur (pour le jaune est compris entre 2.10V et 2.18V).



*Figure (II.9): Image LED.*

## II.7 Les Résistances :

Une **résistance** est un composant électronique ou électrique dont la principale caractéristique est d'opposer une plus ou moins grande résistance (mesurée en ohms:  $\Omega$ ) à la circulation du courant électrique. On peut alors comparer, le débit d'eau au courant électrique **I** (qui est d'ailleurs le débit d'électrons), la différence de pression à la différence de potentiel électrique (qui est la tension **U**) et, enfin, le rétrécissement à la résistance **R**. Ainsi, pour une tension fixe, plus la résistance est faible, plus le courant la traversant est fort. Cette proportion est vérifiée par la loi d'Ohm:

$$U = R * I$$

La valeur de la résistance se mesure en Ohms ( $\Omega$ ). La valeur d'une résistance est déterminée par ses bandes de couleurs.

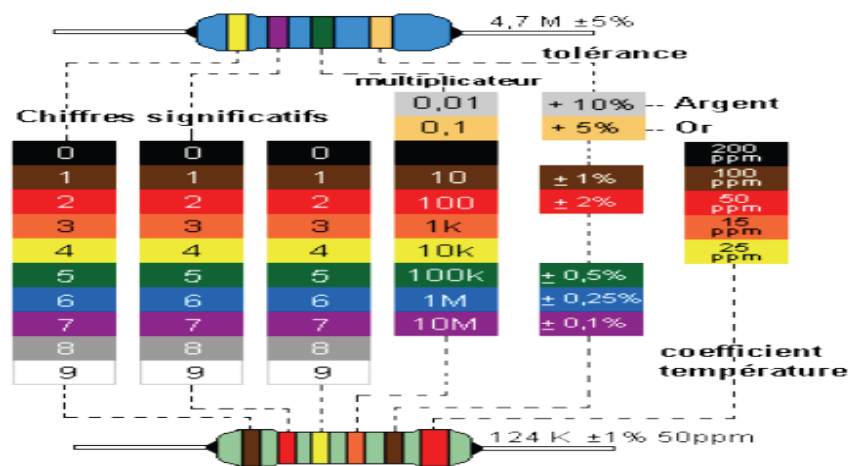


Figure (II.9) : La valeur d'une résistance est déterminée par ses bandes de couleurs.

Outre le tableau ci-dessus, on peut s'aider du petit mode d'emploi qui se trouve ici : [13]

La résistance est schématisée de ces deux manières (européenne à gauche et américaine à droite) :



Figure (II.10) : Symbole de la résistance

## II.8 Condensateur :

Un condensateur est un composant électronique ou électrique élémentaire, constitué de deux armatures conductrices (appelées « électrodes ») en influence totale et séparées par un isolant polarisable (ou « diélectrique »).

Ce diélectrique peut être gazeux, liquide ou solide

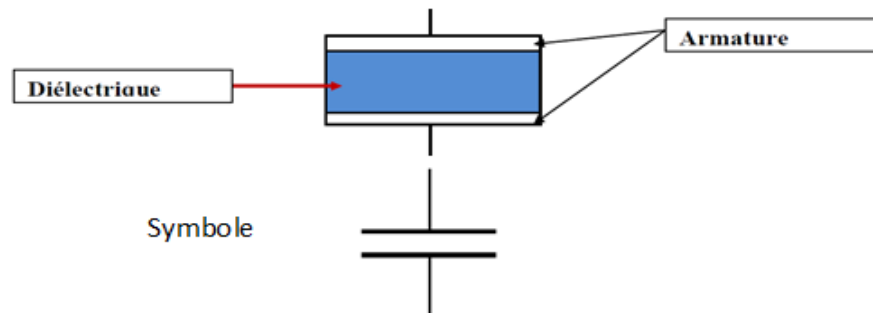


Figure (II.11) : Condensateur

### II.8.a Utilisation :

Le condensateur est utilisé principalement pour :

- stabiliser une alimentation électrique (il se décharge lors des chutes de tension et se charge lors des pics de tension) ;
- traiter des signaux périodiques (filtrage, temporisation...) ;
- séparer le courant alternatif du courant continu, ce dernier étant bloqué par le condensateur ;
- stocker de l'énergie.
- Remonter le facteur de puissance en alternative

### II.8.b Capacité d'un condensateur :

La quantité d'électricité  $Q$  que le condensateur emmagasine est proportionnelle à la différence de potentiel  $U$  entre ces deux armatures et à la capacité du condensateur  $C$  :  $Q = C \times U$

$Q$  : quantité d'électricité en coulombs

$C$  : capacité en farads (F)

$U$  : tension en volt (V)

#### II.8.b.1 Montage série :

La valeur de la capacité du condensateur équivalent est égale à la somme des inverses des capacités des condensateurs du groupement :

$$C_{1eq} = 1/C_1 + 1/C_2 + 1/C_3 + \dots$$

#### II.8.b.2 Montage en dérivation :

La valeur de la capacité du condensateur équivalent est égale à la somme des capacités des

condensateurs du groupement :  $C_{eq} = C1 + C2 + C3 + \dots$

## II.9 Relais :

Un relais électromécanique est un organe électrique permettant de distribuer la puissance à partir d'un ordre émis par la partie commande. Ainsi, un relais permet l'ouverture et la fermeture d'un circuit électrique de puissance à partir d'une information logique. Les 2 circuits, puissance et information, sont complètement isolés (isolation galvanique) et peuvent avoir des caractéristiques d'alimentation électrique différentes.

### II.9.a Symbole :

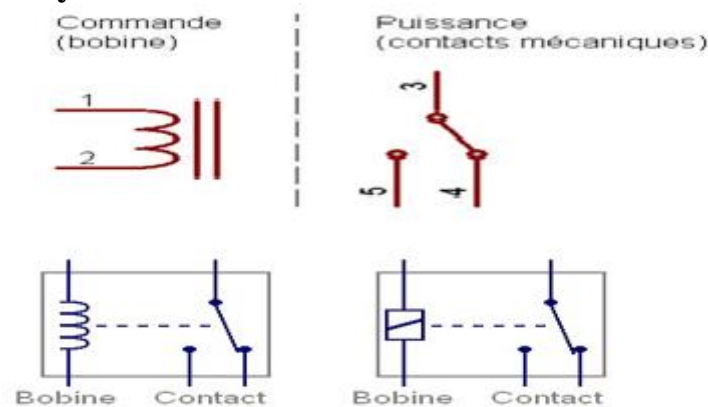


Figure (II.12): Symboles des relais.

### II.9.b Le rôle :

Le rôle de la fonction première des relais est le plus souvent de séparer les circuits de commande des circuits de puissance à des fins d'isolement, par exemple pour piloter une tension ou un courant élevé, à partir d'une commande plus faible, et dans certaines applications, assurer aussi la sécurité de l'opérateur.

## II.10 Transistor : [9]

Le transistor est un composant électronique qui est utilisé dans la plupart des circuits électroniques (circuits logiques, amplificateur, stabilisateur de tension, modulation de signal, etc.) aussi bien en basse qu'en haute tension.

Un transistor est un dispositif semi-conducteur à trois électrodes actives, qui permet de contrôler un courant ou une tension sur l'électrode de sortie (*le collecteur* pour le transistor bipolaire et *le drain* sur un transistor à effet de champ) grâce à une électrode d'entrée (*la base* sur un transistor bipolaire et *la grille* pour un transistor à effet de champ).

C'est un composant fondamental des appareils électroniques et des circuits logiques.

### II.10.a LES TYPES DE TRANSISTOR

- NPN
- PNP



**E : Emetteur, B : Base, C :Collecteur.**

*Figure (II.13) : LES TYPES DE TRANSISTOR*

### II.11 Vanne : [14]

Peut-être décrit comme compact et pratique, facilement installé et entretenu. Consommation mécanique sans consommation électrique, réglable en continu. La vanne de régulation de pression (valve à débit constant) fonctionne comme un interrupteur de débit d'eau. Une vanne est un dispositif destiné à contrôler (stopper ou modifier) le débit d'un fluide liquide, gazeux, pulvérulent ou multiphasique, en milieu libre (canal) ou en milieu fermé (canalisation). L'équivalent anglo-saxon de vanne est le mot *valve*, dont le sens est plus général puisqu'il recouvre les traductions de valve, *vanne* et de soupape en français. Le terme synonyme de vanne est robinet. Il est parfois utilisé pour des modèles de petites dimensions, couramment montés sur des canalisations domestiques.

#### II.11.a Éléments d'une vanne

Toutes les vannes sont constituées des parties fonctionnelles suivantes (voir aussi la figure ci-contre) : Toutes les vannes sont constituées des parties fonctionnelles suivantes (voir aussi la figure ci-contre) :

- ✓ Corps (body en anglais);
- ✓ Voie ou passage (port en anglais) ;
- ✓ Portée ou siège (seat en anglais);
- ✓ Axe ou tige (stem en anglais);
- ✓ Opercule, obturateur ou rotor, tels que : boisseau, papillon, guillotine, sphère ou boule... (disc, rotor ou valve member en anglais), appelé tablier lorsque la

- vanne est plate ;
- ✓ Volant ou actionneur (actuator en anglais) : élément extérieur à la vanne stricto sensu, qui permet de manœuvrer celle-ci ;
  - ✓ Chapeau ou bonnet (bonnet en anglais);
  - ✓ Garniture de presse-étoupe (packing en anglais) : qui permet de maintenir l'étanchéité ; écrou de presse étoupe (*gland nut* en anglais);

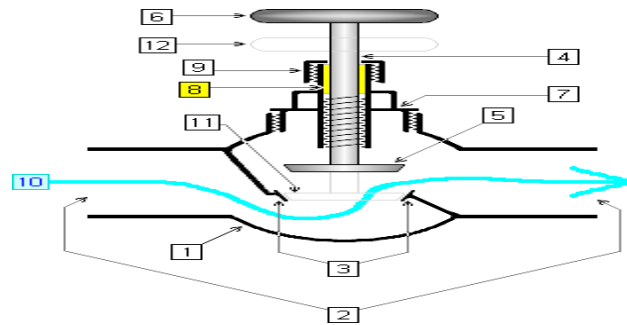


Figure (II.14): Section d'une vanne à clapet.

## II.12 Pompe :

Une pompe est un dispositif permettant d'aspirer et de refouler un liquide (les compresseurs). Les infiltrations d'eau noyant de façon continue les galeries souterraines, on utilise alors de façon régulière des pompes pour évacuer cette eau [7]



Figure (II.15) : Pompe Centrifuge 3 - 12V.

La mise en œuvre de procédés de transformation de matière nécessite très souvent le transport de fluides. L'acheminement d'un fluide d'un point à un autre peut être réalisé en utilisant les forces de gravité, de vide ou des pressions, mais ces solutions sont limitées dans leurs applications. Dans la plupart des cas, on a recours aux pompes, permettant de travailler à des débits réguliers, contrôlés et sur des distances et des

hauteurs importantes.

L'énergie fournie au moteur de la pompe (électrique ou thermique) est transformée en énergie mécanique qui est transmise au fluide. Cet apport d'énergie au fluide permet de vaincre les pertes d'énergie entre les deux extrémités d'un circuit de transport liées aux pertes de charge et à d'éventuelles différences d'altitude ou de pression.

Deux grandes catégories de pompes existent :

- Les pompes volumétriques, qui fonctionnent sur le principe du déplacement d'un volume de fluide
- Les pompes centrifuges, qui fonctionnent sur le principe du rotor et du stator

Chaque catégorie présente certains avantages et inconvénients, mais globalement on retrouve plus fréquemment les pompes centrifuges pour les applications industrielles. Lorsque l'on doit faire le choix d'une pompe pour une installation, il est nécessaire de dimensionner en utilisant les notions de hauteur manométrique totale (HMT, puissance et rendement de la pompe.

**NB** : une notion importante dans le fonctionnement des pompes, qui ne sera pas abordée dans ce cours, est la cavitation (phénomène de vaporisation du liquide à l'intérieur de la pompe).

Cette notion est à prendre en compte pour définir la zone de fonctionnement optimale d'une pompe.

### Hauteur Manométrique Totale (HMT)

La hauteur manométrique totale d'une pompe est la différence de pression du fluide entre l'entrée de la pompe (aspiration) et la sortie de la pompe (refoulement). La HMT est exprimée en mètres de colonne de fluide (mCF).

La HMT représente l'énergie nécessaire que doit transmettre la pompe au fluide pour acheminer le fluide entre deux points A et B, à savoir :

$$HMT = \underbrace{\frac{P_B - P_A}{\rho \cdot g}}_{\text{vaincre les forces de pression entre A et B}} + \underbrace{(z_B - z_A)}_{\text{vaincre les forces de pesanteur entre A et B}} + \underbrace{\frac{u_B^2 - u_A^2}{2 \cdot g}}_{\text{éventuellement amener la vitesse du fluide de } u_A \text{ à } u_B} + \underbrace{\Delta H_{AB}}_{\text{vaincre les pertes de charge entre A et B}}$$

A chaque pompe est associée sa courbe caractéristique, qui présente la relation entre HMT

et débit volumique du fluide. Les allures sont diverses en fonction des types de pompes.

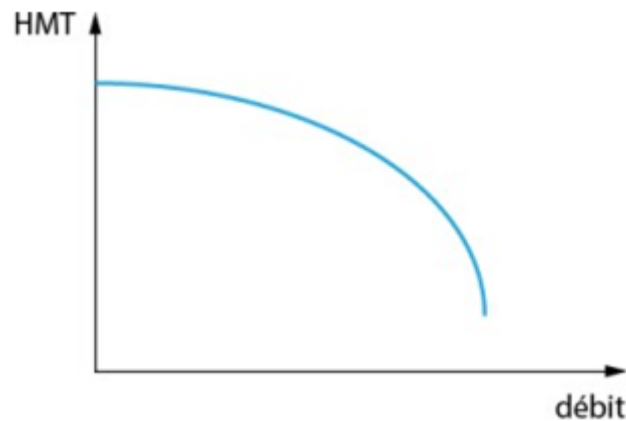


Figure (II.16) : courbe caractéristique présente la relation entre HMT et débit volumique

### Puissance et rendement d'une pompe

La puissance utile d'une pompe correspond à la puissance transmise au fluide par la pompe :

$$P_u = \rho \cdot g \cdot q_v \cdot HMT$$

$P_u$  : Puissance utile (W)

$g$  : Accélération de la pesanteur ;  $g=9,81\text{m}\cdot\text{s}^{-2}$

$\rho$  : Masse volumique du fluide ( $\text{kg}\cdot\text{m}^{-3}$ )

$q_v$  : Débit volumique du fluide ( $\text{m}^3\cdot\text{s}^{-1}$ )

**HMT**: Hauteur manométrique totale (mCF)

Le rendement d'une pompe ( $r$ ) est défini comme le rapport entre la puissance transmise au fluide ( $P_u$ ) et la puissance consommée par le moteur ( $P_{conso}$ ) :

Le rendement de la pompe est dépendant du débit du fluide (pertes d'énergie par frottement variables en fonction du débit). Ainsi, il faut se reporter aux courbes de rendement en fonction du débit fournies par le constructeur pour cibler la zone de fonctionnement optimale de la pompe.

$$r = \frac{P_u}{P_{conso}}$$

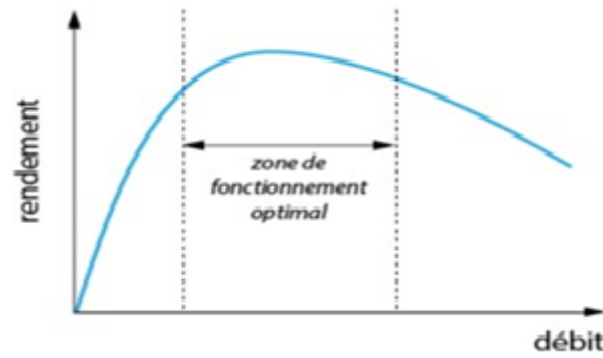


Figure (II.17): courbe caractéristique présente la relation entre Le rendement et le débit

### II.13 Conclusion:

Dans ce chapitre nous avons présenté la carte Arduino et les différentes caractéristiques et spécifiquement la carte Arduino nano, puis nous avons également présenté la partie logicielle et langage de programmation. Nous avons présenté une description détaillée pour les composants utilisés pour réaliser ce projet.

# Chapitre III

*Résultats de Simulation*

### **III.1. INTRODUCTION :**

Dans ce chapitre nous avons présenté les notions générales de Proteus (Isis) et le câblage de la carte, par la suite nous intéressons à la simulation par Arduino nano. Finalement nous démontrons en détail la conception et la mise en œuvre de la carte de contrôlée de discuter les boutons que nous avons choisi pour le contrôle manuel et aussi les capteurs que nous avons également aides aux contrôle automatique. Pour réaliser ce travail nous nous intéressons au régulateur PID pour faire la correction d'erreur de vitesse.

### **III.2. Présentation générale de Proteus (ISIS):**

#### **III.2.a Proteus :**

Est une suite logicielle destinée à l'électronique. Développé par la société Absenter Electroniques, les logiciels incluent dans Proteus permettent la CAO dans le domaine électronique. Deux logiciels principaux composent cette suite logicielle : ISIS, ARES, PROSPICE et VSM.

Cette suite logicielle est très connue dans le domaine de l'électronique. De nombreuses entreprises et organismes de formation (incluant lycée et université) utilisent cette suite logicielle. Outre la popularité de l'outil, Proteus possède d'autres avantages

- Pack contenant des logiciels facile et rapide à comprendre et utiliser
- Le support technique est performant
- L'outil de création de prototype virtuel permet de réduire les coûts matériel et logiciel lors de la conception d'un projet. [18]

#### **III.2.b L'éditeur de schéma ISIS :**

Isis est un éditeur de schémas qui intègre un simulateur analogique, logique ou mixte. Toutes les opérations se passent dans cet environnement, aussi bien la configuration des différentes sources que le placement des sondes et le tracé des courbes. [19]

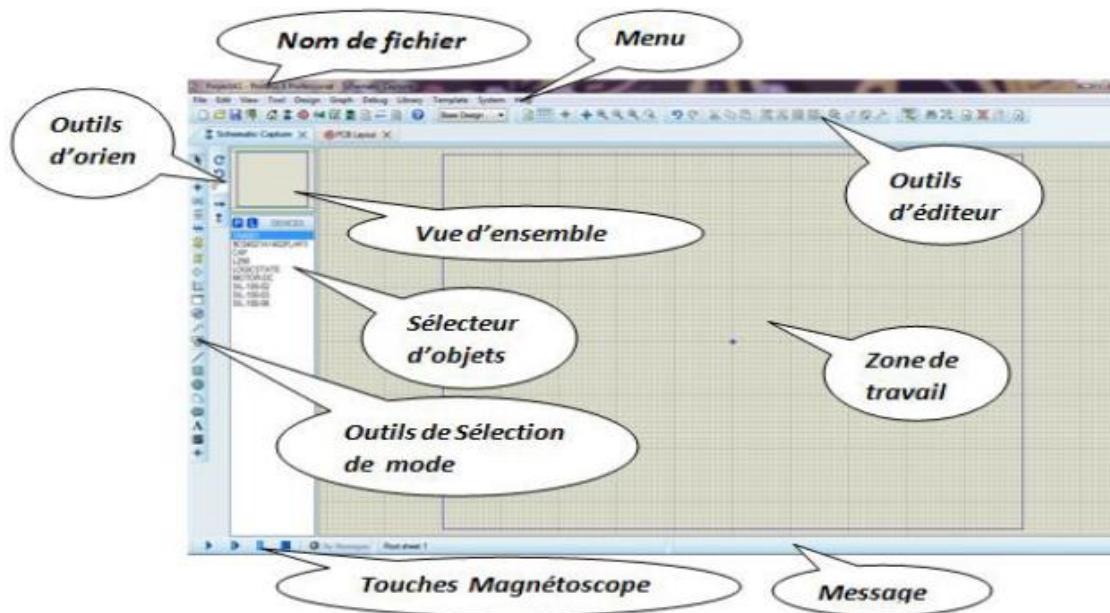


Figure (III.1):Interface utilisateur ISIS.

### III.2.c Interface utilisateur :

#### III.2.c.1 Fenêtre d'ensemble :

Le cadre en bleu délimite l'espace de travail tel qu'il a été défini par la commande 'Définir taille des feuilles' du menu 'système'.

Le cadre en vert délimite la zone de travail, c'est à dire la partie du schéma visible dans la fenêtre principale.

Vous pouvez déplacer cette zone de travail en pointant la souris sur la zone désirée de la fenêtre d'ensemble et en effectuant un clic gauche.

Vous pouvez redéfinir la zone de travail dans la fenêtre d'ensemble en appuyant sur la touche majuscule '**shift**' du clavier, associée au déplacement de la souris en maintenant appuyé le bouton gauche.

#### III.2.c.2 Fenêtre d'Edition :

C'est dans cette fenêtre que vous éditez votre circuit. Elle ne représente que la partie du circuit qui est définie par le cadre vert dans la fenêtre d'ensemble.

Vous pouvez redéfinir la zone travail à l'aide de la fenêtre d'ensemble ou en utilisant la commande '**Zoom**' du menu '**Affichage**'.

#### III.2.c.3 La boîte à outils :

Elle est composée d'un ensemble d'icônes dont les fonctions seront détaillées ultérieurement et d'un sélecteur d'objet utilisé pour choisir les boîtiers, le style des pastilles, des traces, des traversées, etc....

**III.2.c.4 Coordonnées du curseur :**

Les coordonnées déterminent la position du curseur par rapport à l'origine qui par défaut se trouve au centre de la fenêtre d'édition.

**III.2.c.5 Organisation de la boîte à outils :**



Figure (III.2) : Organisation de la boîte à outils ISIS.

**III.2.d Etapes de la saisie de schéma :**

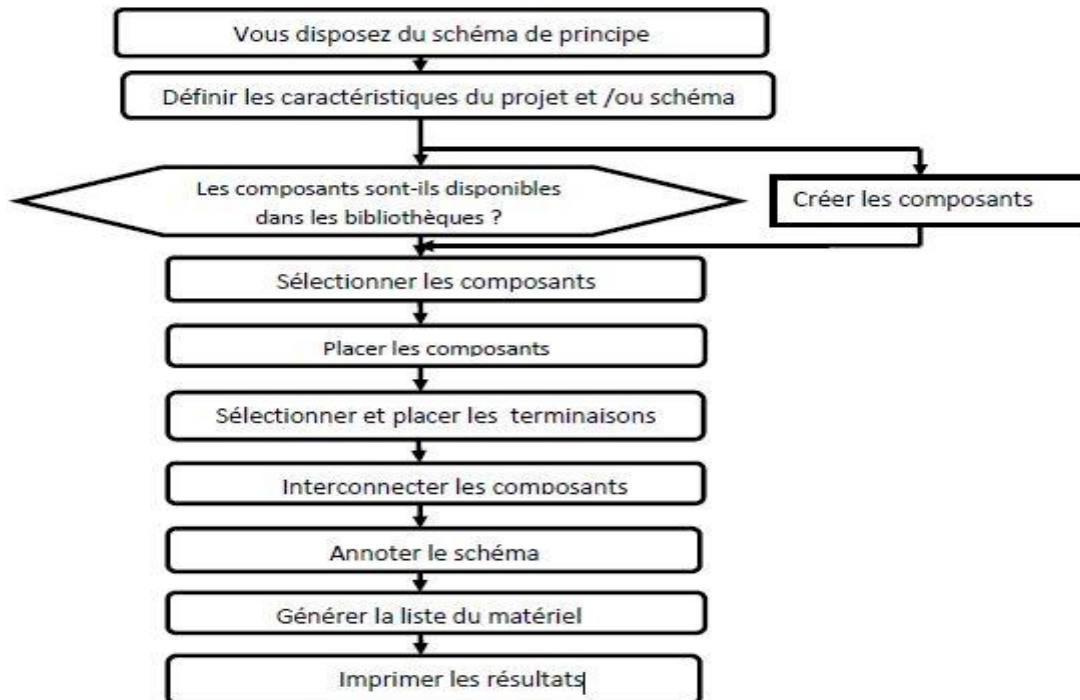


Figure (III.3) : des Etapes de la saisie de schéma.

### III.3. Notre Carte de Commande en Puissance :

Le microcontrôleur (Arduino) a des limitations de courant ; ils ne peuvent donc pas alimenter un élément à haute puissance tel qu'un moteur. L'H-Bridge résout le problème en fournissant une alimentation différente pour conduire les moteurs. C'est pour ça est utilisé RELAY (relais). Une tension d'alimentation stabilisée 12v pour alimenter les moteurs.

### III.4. Schéma fonctionnel :

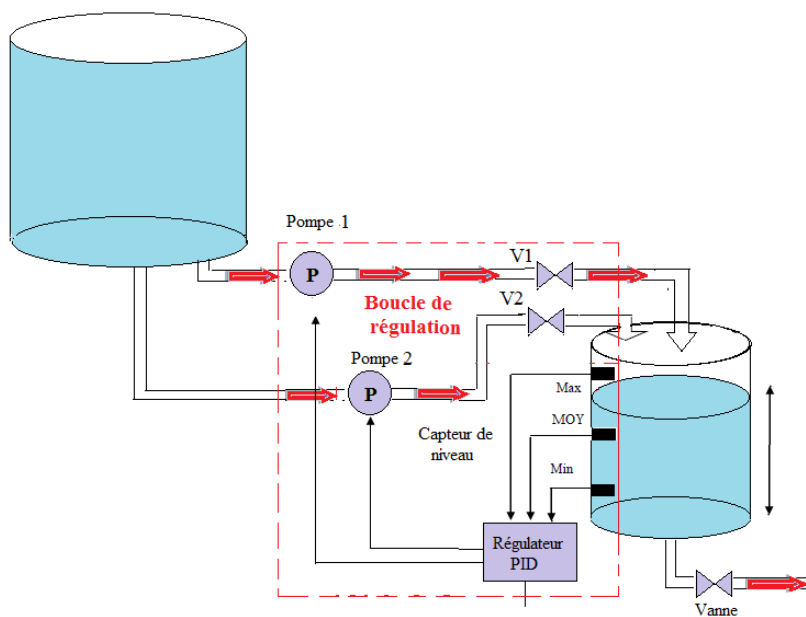


Figure (III.4) : représente la méthode de travail de microcontrôleur.

### III.5. Fonctionnement :

Le projet est une application simple du capteur de niveau d'eau avec Arduino. Le projet ne nécessite pas l'utilisation des transistors pour augmenter le courant. La mise en œuvre nécessite uniquement des capteurs, une carte Arduino. L'astuce consiste l'utilisation de la propriété de conductivité de l'eau. En effet, nous avons utilisé 3 capteurs d'eaux. La détection du niveau est assurée par le contact de l'eau avec le capteur. On distingue deux situations :

Lorsque la pointe n'a pas de contact avec l'eau : dans ce cas, l'entrée analogique Ax (A0... À2) voir la tension aux bornes de résistance 10 K, est égale à 0V.

Lorsque la pointe est en contact avec l'eau : Dans le cas où l'eau est considérée comme un conducteur parfait, le capteur dans le réservoir peut avoir une tension égale à 5V.

Nous avons utilisé trois LEDs indiquant le niveau de l'eau. Les capteurs utilisés doivent être étanches et dénudés uniquement à l'extrémité (pointe rouge). Si le capteur n'est pas étanche on

risque d'avoir une LED allumée en permanent. Le nombre de niveaux est égal à 3. Vous pouvez en ajouter d'autres.

### III.5.a Câblage de la carte:

A0 : Entrée, Niveau 3  
 A1 : Entrée, Niveau 2  
 A2 : Entrée, Niveau 1  
 D2 : Sortie, Indicateur Niveau 3  
 D3 : Sortie, Indicateur Niveau 2  
 D4 : Sortie, Indicateur Niveau 1

### III.5.b Remarque :

Quand A2 est égale 5V, Alors D4 est égale 5V.

Quand A1 est égale 5V, Alors D3 est égale 5V.

Quand A0 est égale 5V, Alors D2 est égale 5V.

(Le contraire est vrai)

En Proteus les 3 capteurs doit être remplacé par 3 boutons et 3 résistances, quand les boutons sont ouverts, donc les capteurs sont en contact avec l'eau, Alors l'entrée analogique (Ax) est égale

## III.6. Simulation de 2 moteurs DC commandés par Arduino NANO :

Afin de voir le fonctionnement et le déroulement du programme de commande des deux moteurs DC, nous avons utilisé Arduino NANO, le schéma suivant illustre le montage et le raccordement des composants.

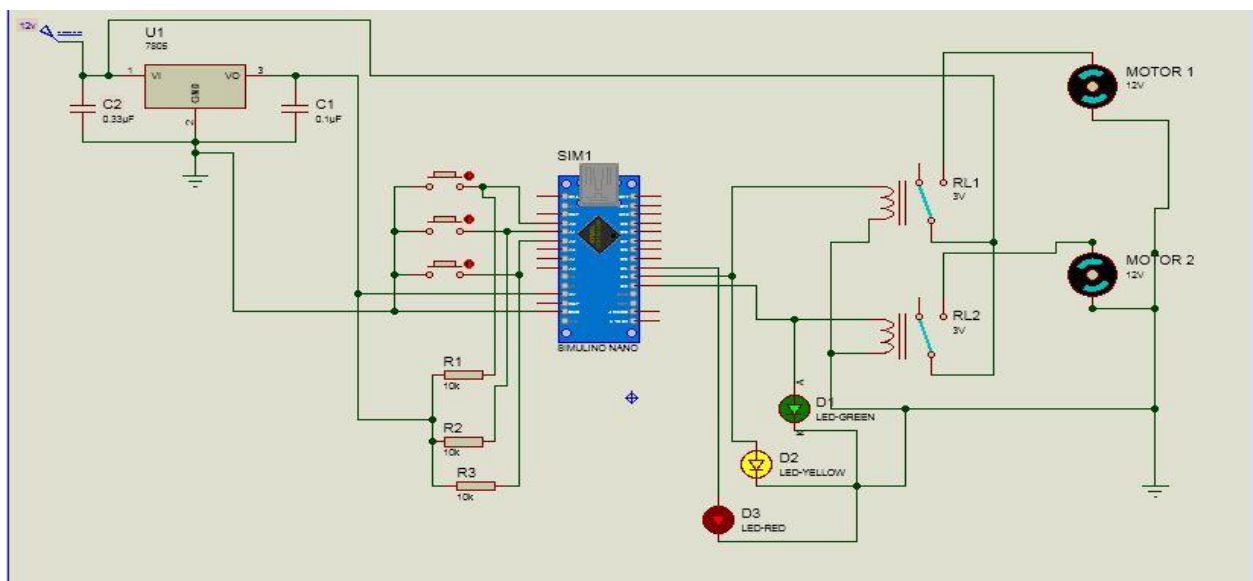


Figure (III.5) : électronique de la carte de commande de 2 moteurs connecté à Arduino.

### III.6.a Résultat de simulation :

Niveau01 :

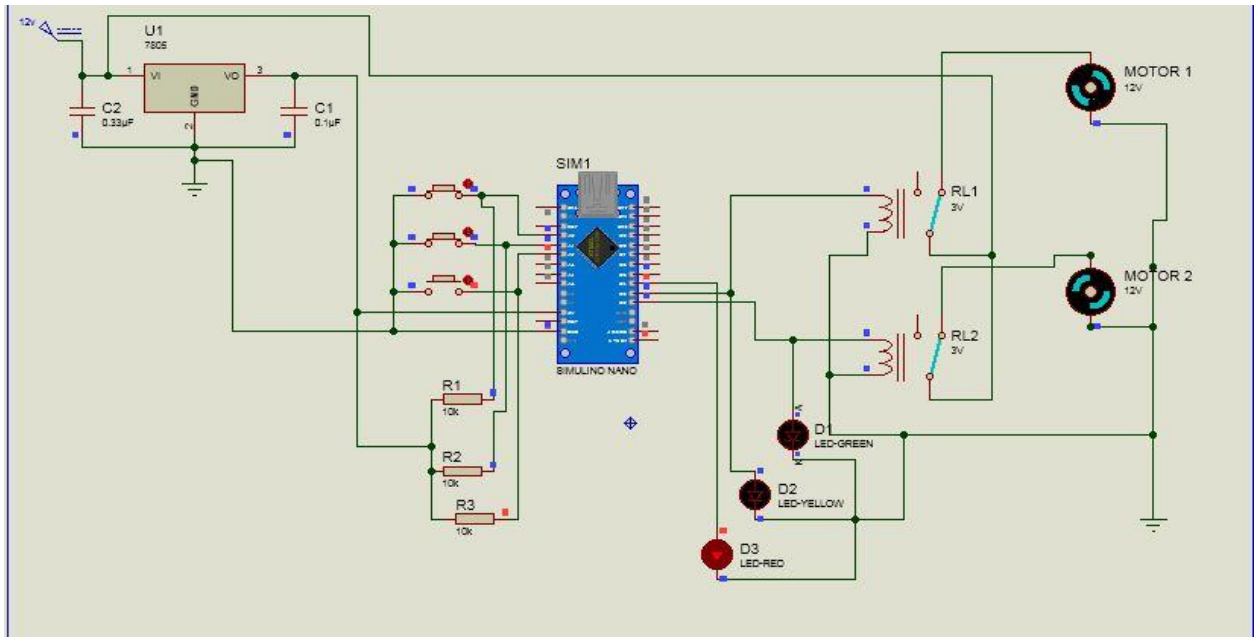


Figure (III.6) : schéma électronique présente le niveau 01.

On appuyé sur le bouton 1 et en laisse les buttons 2 et 3 fermé, on remarque que LED rouge est allumé et les deux LEDs vert et jaune restent atteintes, et les 2 moteurs marchent.

Niveau02 :

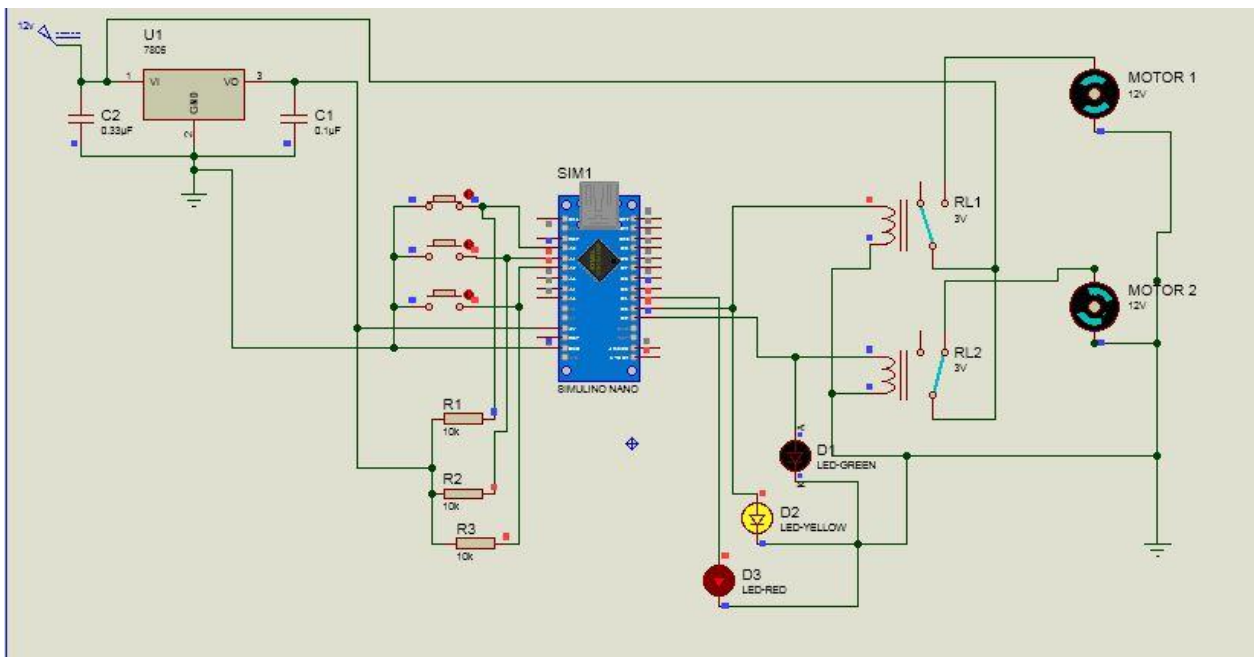


Figure (III.7) : schéma électronique présente le niveau 02.

On appuyé sur les boutons 1 ,2 et en laisse le bouton 3 fermé, on remarque que les LEDs rouge et jaune sont allumés et la LED vert est atteinte, et le 1<sup>ere</sup> moteur marche contrairement au 2<sup>eme</sup> moteur.

Niveau03 :

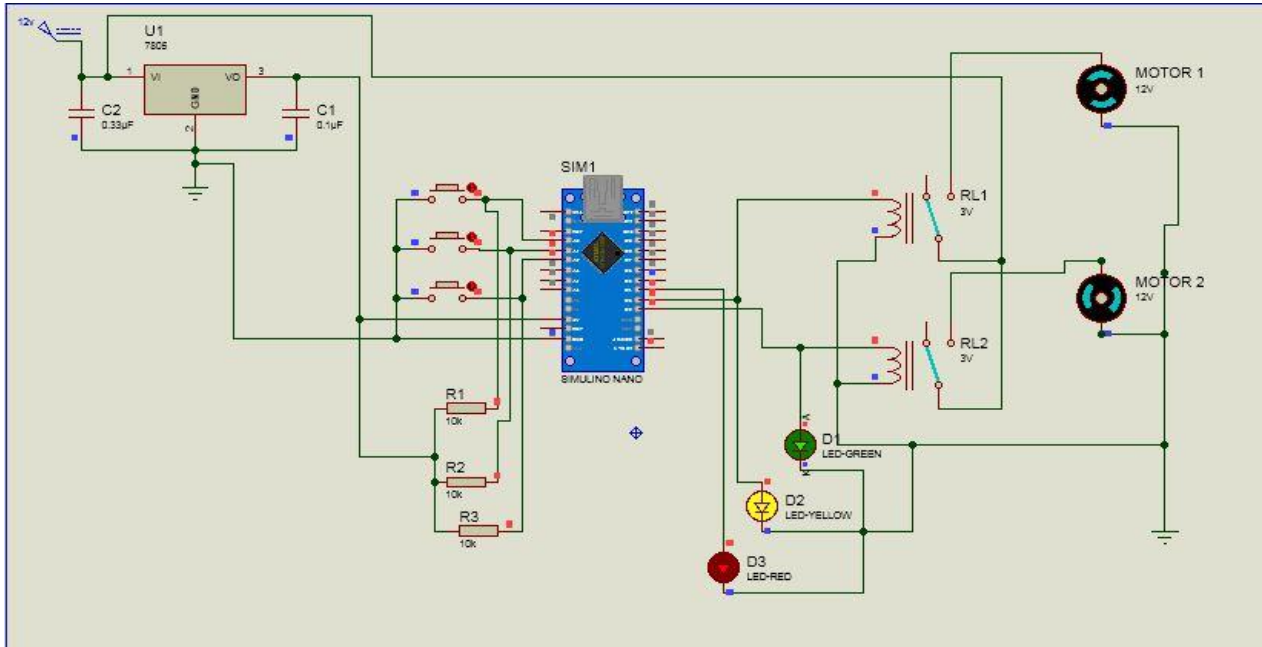


Figure (III.8) : schéma électronique présente le niveau 03.

Dans le cas au ont appuyé les 3 boutons, on remarque que les 3 LEDs rouge, jaune et vert sont allumés, et les 2 moteurs sont arrêtés.

### III.7. Conclusion :

Ce dernier chapitre a été l'occasion de démontrer en détail la conception et la mise en œuvre de la carte de contrôle. Et on a expliqué les différents boutons que nous avons utilisé pour le contrôle manuel, et aussi les capteurs que nous ont également aidés au contrôle automatique, le régulateur PID pour la correction d'erreur de vitesse.

---

# CONCLUSION GENERALE

Dans la plupart des appareils et installations industrielles, il est nécessaire de maintenir leurs grandeurs physiques à des valeurs déterminées. Le niveau d'eau, la température et la pression sont des grandeurs étant par nature variables dans la plupart des installations industrielles, et qui doivent être réglés par des actions convenables en fonction du processus considéré.

Notre contribution dans ce travail consiste à concevoir et à réaliser un système électronique pour la régulation du niveau d'eau.

Dans un premier temps, nous avons étudié le fonctionnement général du régulateur PID et rappelé le principe général de ce régulateur. Puis nous avons présenté la différente forme de régulateur PID

Le second partie sera consacré à la présentation des composants utilisés, introduction sur la carte Arduino, ainsi la partie logiciel, ses capteurs, LEDs, résistances, transistors...etc.

Présentation générale de Proteus (ISIS et ARES), Simulation par Arduino NANO, ses résultats de simulation à chaque niveau d'une réponse en vitesse de moteur (pompe) et en position les LED (allumé ou fermé) feront l'objet de la troisième partie.

---

# Références Bibliographiques

- [1] Dif NihadTrikiRazika, «Synthèse d'un contrôleur PID pour la commande d'une MCC», Mémoire de Master Option:Informatique Industrielle, Université de Larbi BenM'hidi–Oum El Bouaghi, 2010/2011.
- [2]AdneneTLILI ,Safeyiddine KALLELI, «ASSERVISSEMENT ET RÉGULATION », Cours Département Génie mécanique, Université de ISET - Nabeul, 2014/2015.
- [3] Alaa-eddineBordjiba , «Conception et réalisation d'un Régulateur PID numérique de température avec ARDUINO», Mémoire de Master Option: Instrumentation, Université 8Mai 1945 – Guelma, 2019
- [4] MENOUN Idir, REHRAH Ferhat , « Etude comparative des régulateurs PID et Flou Application à un moteur à courant continu (MCC) », Mémoire de Master Option :Automatismes Industriels, Université Abderrahmane Mira - Bejaïa, 2011/2012.
- [5]FOURAR Meriem & KEMOUCHE Meriem, « ETUDE ET REALISATION D'UN SYSTEME DE REGULATION DE NIVEAU D'EAU DANS LES CHAUDIERES A VAPEUR», Mémoire de Master Option: Électromécanique, UNIVERSITE ABDERRAHMANE MIRA - BEJAIA, 2016/2017.
- [7] AMANI Ahmed amine , «LE CONTROLE PID D'UN MOTEUR BRUSHLESS», Mémoire de Master Option Automatique et Informatique Industrielle :, Université de BADJI MOKHTAR ANNABA , 2019.
- [8] AyeB Fakhreddine , «Conception et implémentation d'une commande PID numérique d'un moteur à courant continu.», Mémoire de Master Option Informatique Industrielle :, Université de LARBI BEN M'HIDI D'OUUM EL BOUAGUI , 2013/2014
- [9] S. Tamazirt, « Etude de quelques méthodes de synthèse et de calcul des paramètres des régulateurs PID», mémoire de master, UMMTO, 2013.
- [10] Marcel NOUGARET, « Principes généraux de la correction». Edition Technique de l'ingénieur 2012.
- [11]TOURE FatoumateFousseyni REZIK Kheira, «Contrôle et comandedun système a distance via GSM», Mémoire de Magister Option : Electronique des systemes Embarqués, Université de Abdelhamide Ibn Badis de Mostaganem , 2019.
- [12]Slimane OTS ManeKadda Mansour Ahmed , «Conception et réalisation d'un système d'accès intelligent pour la domotique a base d'Arduino », Mémoire de Magister Option: Réseaux et télécommunication , Université de Belhadj Bouchaib de Ain-Temouchent, 2018/2019.

- 
- [13]Mekhafia Toufik GhadbaneToufik , «Etude et réalisation d'un système de commande a distance des installations électroniques pour la domotique», Mémoire de Magister Option : Electronique des systèmes Embarqués, Université de Msila, 2017/2018.
- [14]Zahra Dafri , Réalisation d un système basé sur internet des objets pour le contrôle des serres intelligentes.», Mémoire de Magister Option: système informatique, Université de 8 Mai 1945 de Guelma , Juillet 2019.
- [15] Cour arduinoSpt 2018
- [16]Mr Yefsah Rabah Mme ZianeLouisa , «Conception et réalisation d'un système de protection et de supervision d'un bâtiment avicole en utilisant une transmission GSM et NRF radio», Mémoire de Magister Option:Electronique industrielle , Université de Mouloud Mammeri de TiziOuzou , 2017/2018
- [17] Wikipédia
- [18] <http://www.elektronique.fr/logiciels/proteus.php>
- [19] Mr. ABDELAZIZ Mourad, Conception et réalisation de la commande d'un bras manipulation ROB3 avec Arduino ; Mémoire de fin d'études pour l'obtention du diplôme de MASTER université ; Dr MOULAY TAHAR Saïda 2015.