

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

UNIVERSITE MOULOU D MAMMERI DE TIZI-OUZOU



FACULTE GENIE ELECTRIQUE ET D'INFORMATIQUE

DEPARTEMENT D'ELECTRONIQUE

Mémoire de Fin d'Etude De Master Professionnel

Domaine : Sciences et Technologies

Filière: Génie Electrique

Spécialité : Electronique Industrielle

Présenté par :

M^{elle} : MERABTI Djahida

Thème

**Conception et réalisation d'une commande automatique d'une
station de pompage à base d'une carte ARDUINO Mega2560**

Mémoire soutenu publiquement le: 04/07/2017

Devant les jurys composé de :

Président : M^r ATTAF Youcef

Promoteur : M^r ZIRMI Rachid

Examineur : M^r HAMICHE Hamid

Année universitaire : 2016/2017

Remerciement

A mon bon dieu qui m'a donné la force et le courage et la patience nécessaire pour réaliser ce modeste travail.

Mes remerciements :

À mon promoteur Mr ZIRMI. Rachid qui ne m'a ménagé aucun effort pour mener ce travail au but attendu, pour son aide, son orientation et ses conseils durant l'accomplissement du projet.

A Melle MAMMAR Lamia et Talem Mohand pour leur aide et leur soutien tout au long de mon travail.

Je remercie vivement les membres de jury qui ont accepté de juger mon travail.

À tous les enseignants de l'Université Mouloud MAMMERRI.

Enfin je remercie toutes les personnes qui m'ont aidé de près et de loin.

A decorative border of red maple leaves surrounds the text. The leaves are arranged in a repeating pattern along the top, bottom, and sides of the page.

Dédicace

*Avec l'aide et la grâce de dieu j'ai achevé ce
modeste travail dont je le dédie*

*A mes très chers parents qui m'ont toujours soutenu avec leurs
encouragements et qui m'ont donné un magnifique modèle de
labeur et de persévérance*

A mes chères sœurs.

A mes oncles et mes tantes

A mes amis

*Je ne prendrai pas le pari de vous les citer tous tant la liste est
longue mais je sais que vous vous reconnaitrez dans ces lignes.
Merci pour vos conseils et votre soutien de tous les jours.*

Djahida

Sommaire

Sommaire

Liste des abréviations

Liste des figures et tableaux

Introduction générale

Chapitre I : Généralité

I.1 : Les pompes centrifuges	2
I.1.1. Définition	2
I.1.2. Constitution de la pompe centrifuge.....	2
I.1.3. Principe de fonctionnement	3
I.1.3.a. L'aspiration :	3
I.1.3.b. L'accélération :	3
I.1.4. Caractéristiques d'une pompe	3
I.1.5. Avantages et inconvénients	4
I.1.5.1 : Avantage :	4
I.1.5.2 : Inconvénients :	4
I.1.6. Les différents types de pompes centrifuges	4
I.2. Moteurs asynchrones triphasés	4
I.2.1.Présentation	4
I.2.2. Constitution	5
I.2.2.1.Le stator (inducteur)	5
I.2.2.2.Le rotor	6
I.2.2.2.a. Rotor bobiné :	6
Symbole :	7
Description de différentes pièces d'un moteur asynchrone à rotor à cage :	7
I.2.2.2.b. rotor à cage d'écureuil :	8
Symbole :	9
Description de différentes pièces d'un moteur asynchrone à rotor à cage :	9
I.2.3. Principe de fonctionnement	10
I.2.3.2. La vitesse du synchronisme :	10
I.2.4. Plaque signalétique d'un moteur asynchrone	11
I.2.5. La plaque à borne	12
I.2.5.1.Couplage des enroulements :	12
I.2.6. Démarrage des moteurs asynchrones triphasés.....	13

Sommaire

I.2.6.1. Démarrage direct :.....	13
I.2.6.2. Démarrage étoile-triangle :	15
I.2.6.3. Démarrage statorique :	14
I.2.6.4. Démarrage rotorique :	14
I.2.6.5. Démarrage par autotransformateur :	14
I.2.7. Grandeurs électromécaniques caractéristiques ^[9]	14
I.2.7.1. Puissance et rendement	14
I.2.8.2. Grandeurs mécaniques : ^[14]	15
I.3. Dispositifs industriels de Commande.....	16
I.3.1. Sectionneur ^[18]	16
I.3.2. Disjoncteurs manuels ^[17]	17
I.3.3. Commutateurs à cames	18
I.3.4. Boutons poussoir	18
I.3.4. Relais de phase.....	18
I.3.5. Relais thermique.....	19
I.3.5-1. Fonction	19
I.3.6. Le Contacteur	20
I.3.6.1. Fonction	20
I.3.6.2. Constitution	20

Chapitre II : Arduino et capteurs

II.1. Définition du module Arduino.....	24
II.1.1. Présentation de la carte ARDUINO Mega2560.....	24
II.1.1.1. Description de la carte Arduino Mega2560	25
II.1.1.2. Le microcontrôleur.....	26
II.1.1.3. Présentation du microcontrôleur ATmega2560	27
II.1.1.3.1 : Architecture :	27
II.1.1.3.2 : Brochage :	28
II.1.1.4. description des broches	28
II.1.1.5. Caractéristiques du microcontrôleur ATMEGA 2560	30
II.1.1.6. Plan mémoire	31
II.1.2. Alimentation de la carte ArduinoMega2560	33
II.1.3. Entrées et sorties numériques	34

Sommaire

II.1.4. Broches analogiques	34
II.1.5. Communication.....	34
II.1.6. Protection du port USB contre la surcharge en intensité ^[24]	35
II.2. Capteurs	35
II.2.1. Définition du capteur	35
II.2.2. Types de capteurs :(apport énergétique) ^[13]	35
II.2.2.a. Capteurs passifs	35
II.2.2.b. Capteurs actifs.....	36
II.2.3. Type de détection	36
II.2.4. Caractéristiques d'un capteur.....	36
II.2.5. Nature de l'information fournie par le capteur : (types de sortie) ^[29]	36
II.2.5.1. Analogique	37
II.2.5.2. Numérique.....	37
II.2.2.3. Logique	37
II.2.2.3.a. Capteur de pression.....	38
➤ pressostat	38
II.2.2-3.b. Capteur de niveau	38
➤ Flotteur :	38

Chapitre III : Conception et réalisation

III.1. Cahier de charge	40
III.1.1. Enoncé du problème	40
III.1.2. La solution envisagée.....	40
III.1.3. Structure d'une station de pompage.....	41
III.1.4. Bronchement des différents modules à la carte ARDUINO	41
III.1.5. L'organigrammes de fonctionnement.....	43
III.1.6. Le temps de fonctionnement d'une pompe.....	44
III.1.7. Défauts capteurs.....	44
III.1.7.1. La table de vérité	45
III.1.7.2. Tableau de karneugh.....	45
III.1.7.3. Logigramme de défauts capteur	46
III.1.8. Démarrage des moteurs	46
III.1.8.1. Démarrage étoile/triangle	47

Sommaire

III.1.8.1.a. Principe	46
III.1.8.1.b. Analyse du fonctionnement au démarrage	47
III.1.8.1.c. Circuit de puissance	48
III.1.8.1.d. Les étapes de fonctionnement de circuit de puissance	49
III.2. Matériels utilisés.....	49
III.2.1. Carte Arduino	50
III.2.2. Contacteur tripolaire NC1-32	50
III.2.3 Le relais.....	50
III.2.4. Le transistor	51
II-5. La diode.....	52
III.2.8. Les commutateurs On/Off.....	52
III.2.7. LED : diode électroluminescente	53
III.2.11.Circuit électrique	54
III.3. Réalisation	56
Conclusion générale	
Bibliographie	

*Figures et
Tableaux*

Figure et tableaux

Figures

Chapitre I :

Figure (I-1) : constitution d'une pompe centrifuge :	2
Figure (I-2): Vue en coupe d'un moteur asynchrone triphasé et identifie ses principaux composants.....	5
Figure (I-3) : vue d'un stator d'un moteur asynchrone triphasé.....	5
Figure (I-4) : Schéma de principe d'un rotor bobiné (à bagues) d'un moteur asynchrone triphasé.....	6
Figure (I-5) : Symbole d'un moteur asynchrone triphasé à rotor bobiné (à bague).	7
Figure (I-6) : description de différentes pièces d'un moteur asynchrone triphasé à rotor bobiné (à bague).....	7
Figure (I-7) : Vue d'un rotor à cage d'écureuil d'un moteur asynchrone triphasé.....	8
Figure (I-8) : Symbole d'un moteur asynchrone triphasé à rotor à cage.	8
Figure (I-9) : Description de différentes pièces d'un moteur asynchrone triphasé à rotor à cage.	9
Figure (I-10) : Principe d'un moteur asynchrone triphasé.....	10
Figure (I-11) : Plaque signalétique d'un moteur asynchrone.....	11
Figure (I-12) : bornes du moteur asynchrone sur la plaque à borne.....	12
Figure (I-13) : couplages des enroulement en étoile.....	13
Figure (I-14) : couplage des enroulements en triangle.	13
Figure (I-15) : Symboles-en circuit de puissance et en circuit de commande.....	16
Figure (I-16) : symbole d'un interrupteur sectionneur.	16
Figure (I-17) : Symbole d'un fusible.	17
Figure (I-18) : Symbole d'un disjoncteur manuel.....	17
Figure (I-19) : Symbole d'un commutateur à came.....	18
Figure (I-20) : Symbole d'un bouton poussoir.	18
Figure (I-21) : Symbole d'un relais de phase.....	19
Figure (I-22) : Symbole d'un relais thermique.	19
Figure (I-23) : Symbole d'un contacteur.	20
Figure (I-24) : circuit de puissance d'un contacteur.....	21
Figure (I-25) : circuit de commande d'un contacteur.....	21
Figure (I-26) : l'organe mécanique.....	21

Figure et tableaux

Chapitre II :

Figure (II-1) : Architecture de la carte Arduino Mega2560 :	25
Figure (II-2) : Architecture interne de l'Atmega2560.	27
Figure (II-3) : Les différents pins de microcontrôleur.....	28
Figure (II-4) : Alimentation de l'Arduino par un microcontrôleur.....	34
Figure (II -6) : principe général des capteurs.....	35
Figure(II-7) : Schéma synoptique d'un capteur de pression.....	37
Figure (II-8) : photo d'un flotteur.....	39

Chapitre III :

Figure (III-1) : Structure de la station de pompage.....	41
Figure (III-2) : Branchement des différents modules à la carte ARDUINO.....	42
Figure (III-3) : l'organigramme de remplissage	43
Figure (III-4) : l'organigramme de vidage.....	44
Figure (III-5) : Le fonctionnement des pompe en fonction la de pression du réservoir.	44
Figure (III-6) : Schéma du temps fonctionnement d'une pompe.....	45
Figure (III-7) : Logigramme de défauts capteur.	46
Figure (III-8) : Couplages étoile et couplage triangle. [12].....	47
Figure (III-9) : Schéma de puissance de démarrage étoile triangle pour le moteur asynchrone triphasé.	48
Figure (III-10) : les étapes de fonctionnent d'un circuit de puissance.....	49
Figure (III-11) : Circuit électrique de module relais.....	52
Figure (III-12) : Le circuit électrique des commutateurs.....	53
Figure (III-13) : LED de signalisation de défauts capteur.	54
Figure (III-14) : Circuit électrique du système.	55
Figure (III-15) : L'alimentation de la carte Arduino	56
Figure (III-16) : Branchement des transistors à la carte Arduino.....	57
Figure (III-17) : Les modules des relais.....	57
Figure (III-18) : Les commutateurs avec test de défaut capteur.	58
Figure (III-19) : Photo de la réalisation.	59

Figure et tableaux

Les tableaux

Chapitre I

Tableau (I-1) : de comparaison de la tension de réseau et de la plaque signalétique.....12

Chapitre II

Tableau (II-1) : les caractéristiques de la carte Arduino Mega2560.....26

Tableau (II-3) : temps d'accès en écriture.....32

Chapitre III

Tableau (III-1) : Table de vérité de défaut capteurs.45

Tableau (III-2) : Tableau de karneugh de défauts capteur.45

Abréviations

η : Le rendement [%].

P_h : puissance hydraulique de la pompe [Watt].

ρ : masse volumique de l'eau (=1000kg/m³).

Q : débit volumique [mètre³/h] ou [mètre/seconde].

H : hauteur manométrique [mètre].

N_s : Vitesse de synchronisme [tr/min].

P : Nombre de paires de pôles.

g : glissement [%].

Ω_s : Vitesse angulaire de synchronisation

Ω : La vitesse angulaire de rotation du rotor

N : La vitesse de rotation du rotor [tr/min].

f : fréquence de la tension [Hz].

ω : pulsation de la tension [rad/s].

P_a : puissance absorbée.

U : tension entre les bornes du moteur (tension entre phases) [Volt].

I : courant en ligne [Ampère].

$\cos\varphi$: facteur de puissance.

P_u : puissance utile [Watt].

C_u : Couple mécanique utile [N.m].

gF : fusible à usage domestique.

gG : fusible à usage industriel.

PWM: largeur d'impulsion modulée.

USB: Universal Serial Bus.

SRAM: Static Random Access Memory.

EEPROM: Electrically-Erasable Programmable Read Only Memory.

LED: Light Emitting Diode.

ALU : l'unité arithmétique et logique.

Abréviations

AC : Alternative Current.

DC : Direct Current

ADC : Convertisseur Analogique/numérique

UART : (Universel Asynchrones Receveur Transiter.

A : Analog (analogique).

D : Digital (numérique).

IDE : Integrated Developement Environnement.

I2C : Inter Integred Circuit.

M : Moteur.

Introduction
Générale

Introduction

L'eau, en tant que ressource stratégique et vitale dans la vie, nécessite une bonne gestion afin d'optimiser son exploitation. Il est alors indispensable de faire appel à des techniques efficaces de supervision au niveau des installations hydrauliques et de leurs accessoires.

L'idée de ce projet a pour but est de concevoir un système qui permet la gestion et l'optimisation du fonctionnement des stations de pompage.

Parmi les éléments utilisés dans les installations de la commande industriels sont : les pompes hydraulique, représente l'une des éléments les plus importants dans une station de pompage, elle génère la puissance hydraulique, débit de liquide, un moteur électrique pour actionner la pompe, et les dispositifs de commande tels que les contacteurs, relais thermique, etc....

En effet il existe sur le marché des milliers de modèles de cartes de commande comme la carte Arduino, cette approche est basée sur l'utilisation des ordinateurs. D'une part, leurs disponibilités et leur faible coût nous permettra une conception à budget minime, et d'autre part, la puissance et la rapidité d'exécution des processeurs du moment pourra prendre en charge le traitement des données lors d'une communication entre les modules de commandes. ^[32]

Dans ce cas, Arduino est le cerveau du notre système qui nous permette de réaliser des programmes de commande, et gérer le fonctionnement des deux pompes en fonction de niveau et de la pression mesurés à l'aide des capteurs comme les flotteurs et les pressostats.

L'objectif de ce travail est d'assurer le bon fonctionnement des pompes, et le remplissage de réservoir on évitant le débordement de ce dernier. Nous avons alors commencé, après une introduction, par le premier chapitre consacré à la définition et la description des composants constituant notre système. Dans le deuxième chapitre nous somme intéressé à la présentation de la carte Arduino et les capteurs.

Dans le troisième chapitre on a deux parties, la première consacrée à la conception de notre système après avoir défini le cahier des charges, et la deuxième partie à la réalisation de ce dernier est présentée avec les détails que nous avons vus intéressantes à montrer. Nous avons terminé par une conclusion dans laquelle nous avons mis en évidence l'essentiel de notre réalisation.

Chapitre I :

Généralité.

Introduction :

L'hydraulique industriel est un domaine très vaste, alors on s'intéresse d'abord aux composants essentiels participant à la réalisation des circuits hydrauliques : pompes, moteurs, dispositifs de commande, Etc....

Ce chapitre présente la description, fonctionnement et caractéristiques de ces composants.

I.1 : Les pompes centrifuges

I.1.1. Définition

Une pompe centrifuge est une machine rotative qui transforme la puissance mécanique d'un moteur électrique en une puissance hydraulique fournie au fluide ^[1]

I.1.2. Constitution de la pompe centrifuge

Les pompes se composent de deux éléments essentiels :

- Une roue qui impose au liquide un mouvement de rotation. Celle-ci est montée sur un arbre porté par des paliers et entraîné par un moteur.
- Un corps de pompe (enveloppe extérieure de la machine) qui dirige l'écoulement vers la roue et l'en éloigne à nouveau sous plus haute pression. Le corps de pompe comprend une tubulure d'aspiration et une tubulure de refoulement, supporte les paliers et ensemble du rotor. ^[1]

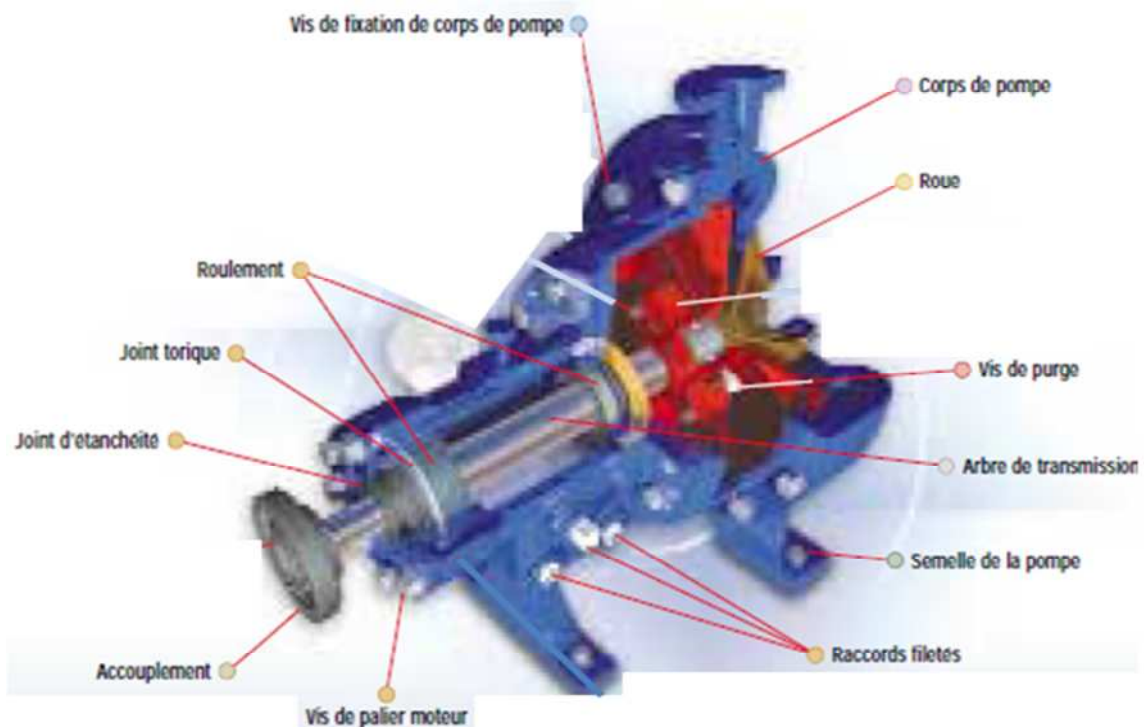


Figure (I-1) : constitution d'une pompe centrifuge ^[2]

I.1.3. Principe de fonctionnement

On peut décomposer le fonctionnement d'une pompe centrifuge en deux étapes : ^[3]

I.1.3.a. L'aspiration :

Le liquide est aspiré au centre du rotor par une ouverture appelée distributeur dont le rôle est de conduire le fluide depuis la conduite d'aspiration jusqu'à la section d'entrée du rotor.

La pompe étant amorcée, c'est à dire pleine de liquide, la vitesse du fluide qui entre dans la roue augmente et par conséquent la pression diminue et engendre ainsi une aspiration et maintient l'amorçage.

I.1.3.b. L'accélération :

Le rotor transforme l'énergie mécanique appliquée à l'arbre de la machine en énergie cinétique. A la sortie du rotor, le fluide se trouve projeté dans la volute dont le but est de collecter le fluide et de le ramener dans la section de sortie. La section offerte au liquide étant de plus en plus grande, son énergie cinétique se transforme en énergie potentielle (pression).

I.1.4. Caractéristiques d'une pompe

D'un point de vue hydraulique, les trois principaux paramètres qui caractérisent une pompe centrifuge sont :

- Le débit volumique Q : C'est le volume refoulé pendant l'unité de temps. Il s'exprime en mètres cubes par seconde (m^3/s), ou plus pratiquement en mètres cubes par heure (m^3/h). ^[6]
- La hauteur manométrique H : On appelle Hauteur manométrique H d'une pompe, l'énergie fournie par la pompe à l'unité de poids du liquide qui la traverse. C'est la somme de la hauteur géométrique dans les niveaux du liquide et les pertes de charge causées par de frottements intérieurs qui se forment au passage du liquide dans les tuyaux, dans la pompe et les accessoires hydrauliques. ^[6]
- Le rendement η : Le rendement d'une pompe est le rapport de la puissance utile « P_h » (puissance hydraulique) communiquée au liquide pompé à la puissance absorbée « P_a » par la pompe ou par le groupe (aux bornes du moteur).

$$\eta = \frac{P_h}{P_a} \quad (\text{I-1})$$

L'expression générale de la puissance hydraulique est donnée comme suit :

$$P_h = \rho g H Q \quad (\text{I-2})$$

Ph : puissance hydraulique de la pompe(en Watt),
 ρ : masse volumique de l'eau (=1000kg/m³).
Q : débit volumique. m³/h
H : hauteur manométrique. ^[1]

I.1.5. Avantages et inconvénients

I.1.5.1 : Avantage :

- ✓ Ces machines sont de construction simple et demande peu d'entretien.
- ✓ Prix modérés et coût de maintenance faible.
- ✓ Matériaux de construction très variés (fluide corrosif possible).
- ✓ Pompes compactes et peu encombrantes.
- ✓ Bons rendements.
- ✓ Le débit est continu.
- ✓ En cas de dysfonctionnement du circuit de refoulement (colmatage), la pompe ne subit aucun dommage. ^[3]

I.1.5.2 : Inconvénients :

- ✓ Elle n'est pas auto-amorçant.
- ✓ Elle ne fonctionne pas avec des fluides trop visqueux.
- ✓ Elle nécessite des dispositifs d'équilibrage. ^[3]

I.1.6. Les différents types de pompes centrifuges

Il existe différentes formes de roues et d'aubes qui induisent une classification de ces machines

- Des rotors fermés ou rotors ouverts.
- On peut monter plusieurs roues sur le même arbre ce qui permet d'augmenter la pression de refoulement

On les appelle pompes multicellulaires ou multi-étagées. ^[3]

I.2. Moteurs asynchrones triphasés

I.2.1. Présentation

Cette partie est consacrée à la présentation des moteurs asynchrones triphasés, moteurs les plus utilisés pour l'entraînement des machines.

Ces moteurs s'imposent en effet dans un grand nombre d'applications en raison des avantages qu'ils présentent : ils sont robustes, simples d'entretien, faciles à mettre en œuvre et de faible coût. Ils sont utilisés pour transformer l'énergie électrique en énergie mécanique grâce à des phénomènes électromagnétiques.

I.2.2. Constitution

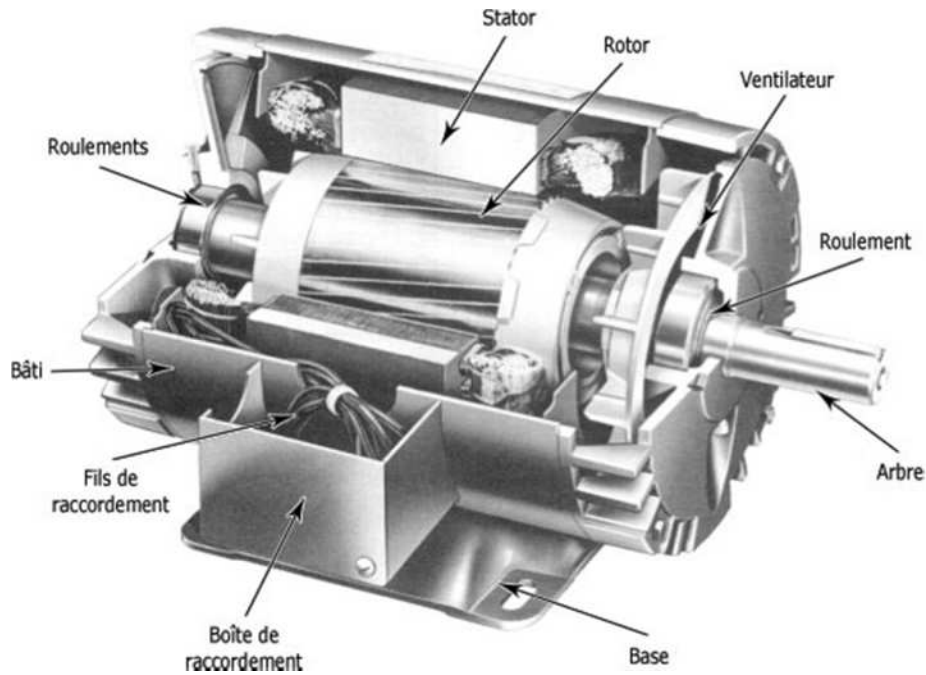


Figure (I -2): Vue en coupe d'un moteur asynchrone triphasé et identifie ses principaux composants. ^[5]

I.2.2.1. Le stator (inducteur)

C'est la partie fixe du moteur, la carcasse sur laquelle est fixée une couronne de tôle d'acier muni d'encoches. Des bobines de section appropriée sont réparties dans ces derniers et forment un ensemble d'enroulement qui est d'un nombre correspondant aux nombres de phases du réseau d'alimentation.

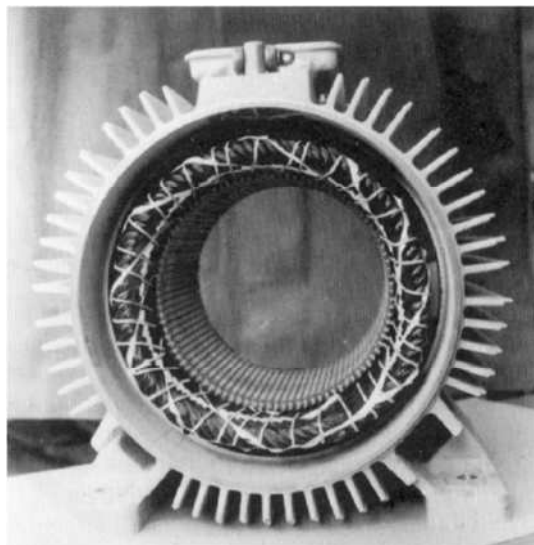


Figure (I-3) : vue d'un stator d'un moteur asynchrone triphasé. ^[9]

I.2.2.2. Le rotor

C'est la partie mobile du moteur, c'est lui qui transmet la force mécanique à la machine à mettre en mouvement, il est placé à l'intérieur du stator, il est constitué d'un empilement de tôles d'acier formant un cylindre fixée sur l'arbre du moteur.

Il existe deux types de rotor :

- ❖ Rotor bobiné (à bagues).
- ❖ Rotor à cage d'écureuil (rotor en court-circuit).

I.2.2.2.a. Rotor bobiné :

Le rotor comporte un enroulement bobiné à l'intérieur d'un circuit magnétique constitué de disques en tôles empilés sur l'arbre de la machine. Cet enroulement est obligatoirement polyphasé, même si le moteur est monophasé. Les encoches, découpées dans les tôles sont légèrement inclinées par rapport à l'axe de la machine de façon à réduire les variations de réluctance liées à la position angulaire rotor/stator et certains pertes dues aux harmoniques. Les extrémités des enroulements rotorique sont sorties et reliées à des bagues montées sur l'arbre, sur lesquelles frottent des balais en charbon. On peut ainsi mettre en série avec le circuit rotorique des éléments de circuit complémentaires (résistances, électronique de puissance...) qui permettent des réglages de la caractéristique couple/vitesse. Ce type de moteur est utilisé essentiellement dans des applications où les démarrages sont difficiles et /ou nombreux. ^[9]

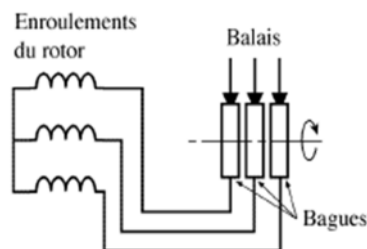


Figure (I-4) : Schéma de principe d'un rotor bobiné (à bagues) d'un moteur asynchrone triphasé. ^[7]

➤ Symbole :

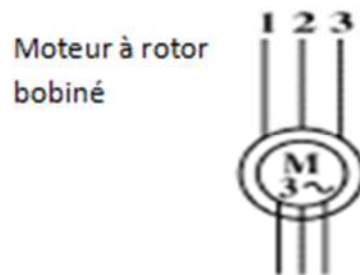


Figure (I-5) : Symbole d'un moteur asynchrone triphasé à rotor bobiné (à bague).^[7]

➤ Description de différentes pièces d'un moteur asynchrone triphasé à rotor bobiné :

La figure ci-dessous montre les différents composants d'un moteur asynchrone triphasé à rotor bobiné.

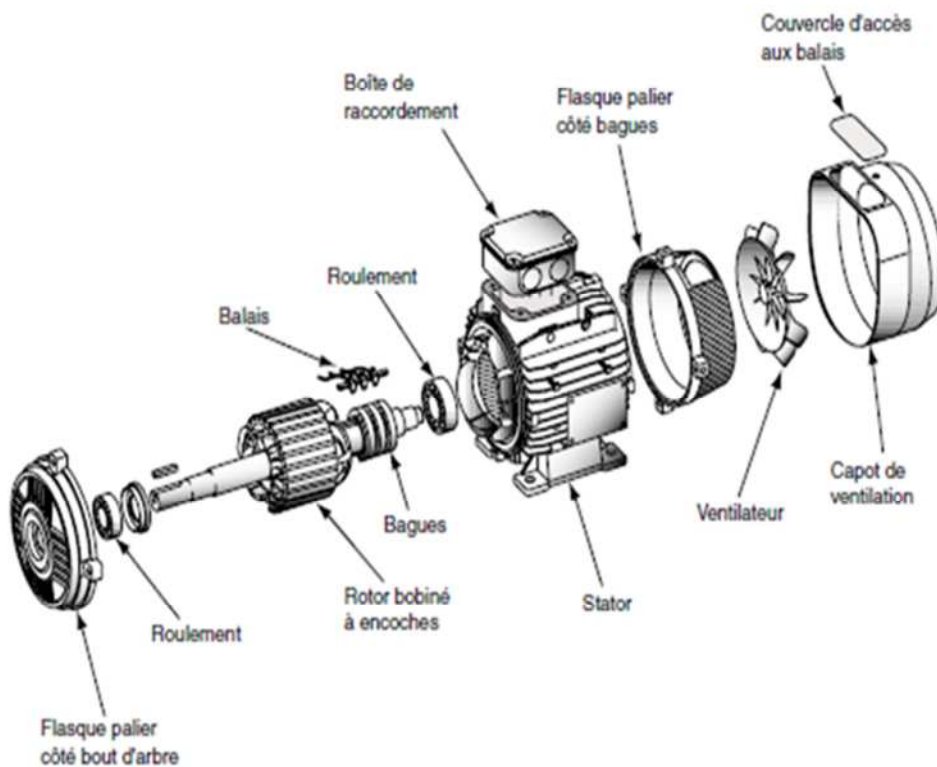


Figure (I-6) : description de différentes pièces d'un moteur asynchrone triphasé à rotor bobiné (à bague).^[8]

I.2.2.2.b. rotor à cage d'écureuil :

Le circuit du rotor est constitué de barres conductrices régulièrement réparties entre deux couronnes métalliques formant les extrémités, le tout rappelant la forme d'une cage d'écureuil, cette cage est insérée l'intérieur d'un circuit magnétique analogue à celui du moteur à rotor bobiné. Les barres sont faites en cuivre, en bronze ou en aluminium, suivant les caractéristiques mécaniques et électriques recherchées par le constructeur. Dans certaines constructions, notamment pour des moteurs à basse tension, la cage est réalisée par coulée et centrifugation d'aluminium. On démontre que. Ce type de moteur, beaucoup plus aisé à construire que le moteur à rotor bobiné est par conséquent d'un prix de revient inférieur et a une robustesse intrinsèquement plus grande. Il n'est donc pas étonnant qu'il constitue la plus grande partie du parc des moteurs asynchrones actuellement en service. Son inconvénient majeur est qu'il a, au démarrage, de mauvaises performances (courant élevé et faible couple). C'est pour remédier à cette situation qu'ont été développés deux autres types de cages (rotor à double cage et rotor à encoches profondes).^[9]

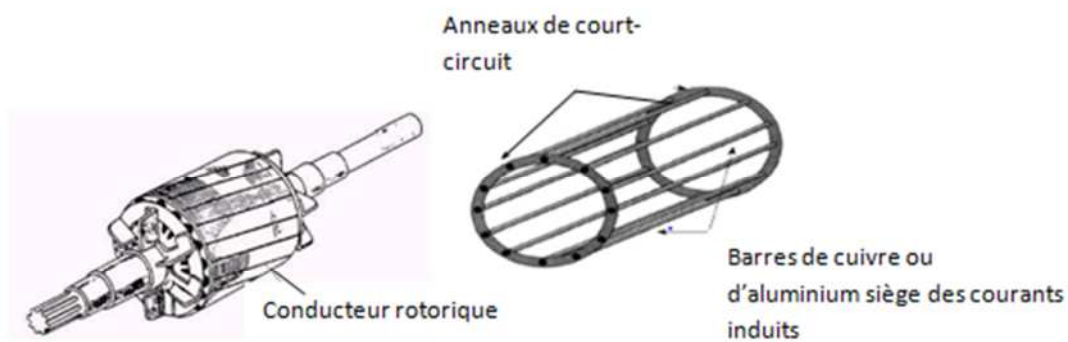


Figure (I-7) : Vue d'un rotor à cage d'écureuil d'un moteur asynchrone triphasé.^{[7] [17]}

➤ **Symbole :**

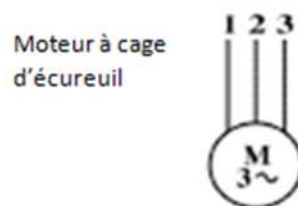


Figure (I-8) : Symbole d'un moteur asynchrone triphasé à rotor à cage.^[7]

➤ Description de différentes pièces d'un moteur asynchrone à rotor à cage :

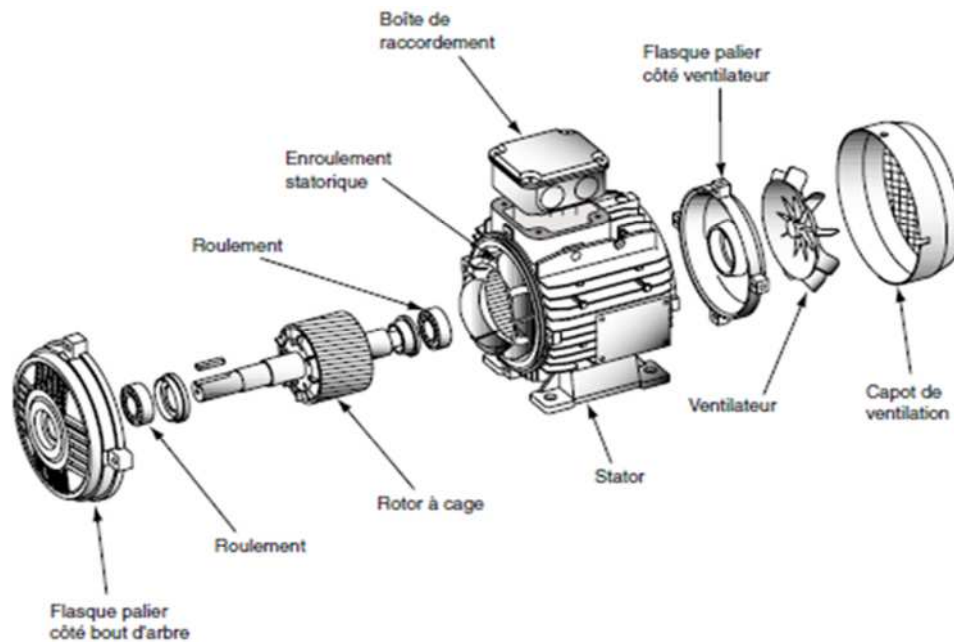


Figure (I-9) : Description de différentes pièces d'un moteur asynchrone triphasé à rotor à cage. [8]

I.2.3. Principe de fonctionnement

- Le stator supporte trois enroulements, décalés de 120° , alimentés par une tension alternative triphasée.
- Ces trois bobines produisent un champ magnétique variable qui à la particularité de tourner autour de l'axe du stator suivant la fréquence de la tension d'alimentation, ce champ magnétique est appelé champ tournant.
- Le champ tournant (statorique) vient induire des courants dans le rotor.
- Leur interaction entraîne la rotation du rotor à une fréquence légèrement inférieure à celle du champ tournant [9]

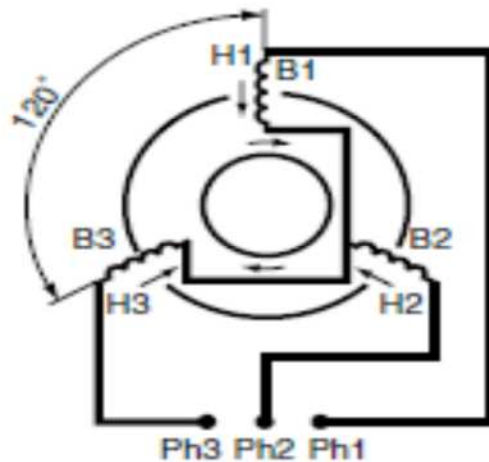


Figure (I-10) : Principe d'un moteur asynchrone triphasé. ^[8]

I.2.3.2. La vitesse du synchronisme :

Est nommée Ω_s en tours/secondes on a alors :

Sur les moteurs asynchrones, la vitesse de synchronisme dépend de la fréquence d'alimentation et du nombre de paires de pôles (P) constituant le stator :

$$N_s = \frac{60f}{P} \quad (I-3)$$

Avec :

N_s : Vitesse de synchronisme [tr/min].

f : Fréquence [Hz].

P : Nombre de paires de pôles. ^[8]

I.2.4. Plaque signalétique d'un moteur asynchrone

La plaque signalétique d'un moteur asynchrone est la carte d'identité du moteur délivrée certifiée par le constructeur. Elle contient les caractéristiques nominales électriques du moteur, la figure ci-dessous montre un exemple d'une plaque signalétique d'un moteur asynchrone triphasé. ^[33]



Figure (I-11) : Plaque signalétique d'un moteur asynchrone. ^[33]

- **Type** : (LS90Lz) référence propre au constructeur.
- **Puissance** : (1.5 Kw) puissance utile délivrée sur l'arbre du moteur.
- **Facteur de puissance ou cos phi** : (0.78) permet le calcul de la puissance réactive consommée.
- **Rendement (76%)** : permet de connaître la puissance électrique consommée ou absorbée.
- **Tension** : (230V/400V) la première indique la valeur nominale de la tension aux bornes d'un enroulement. Elle détermine le couplage (étoile ou triangle) à effectuer en fonction de la tension du réseau d'alimentation.
- **Intensités** : (6.65 A/3.84 A) Elles représentent l'intensité en ligne (dans chaque phase) pour chacun des couplages.
- **Vitesse** : (1440 Tr/min) indique la vitesse nominale du rotor. On dit aussi vitesse réelle. On connaît alors la vitesse de synchronisation **ns** du moteur (ici 1500 tr/min).
- **Classe d'isolement** : (non indiquée).
- **Température ambiante** : (40°C) utilisation recommandée maximum.
- **Fréquence** : (50Hz) fréquence du réseau d'alimentation.
- **Nombre de phases** : (Ph 3) moteur triphasé.
- **Service** : (S1) utilisation en marche continue, intermittente.
- **Indice de protection IP** : (non indiquée) défini par trois Chiffres le degré de protection du moteur à la poussière, à eau et aux chocs mécaniques.
- **Détermination de couplage** : A partir des indications donnée par la plaque et le réseau d'alimentation l'utilisateur doit coupler adéquatement les enroulements du stator soit en triangle soit en étoile.
 - Si la plus petite tension de la plaque signalétique du moteur correspond à la tension entre phases du réseau on adopte le couplage triangle.
 - Si la plus grande tension de la plaque du moteur correspond à la tension entre la phase du réseau on adopte le couplage étoile.
 -

Tableau (I-1) : de comparaison de la tension de réseau et de la plaque signalétique. ^[17]

Réseau d'alimentation		Plaque signalétique		Couplage
230V	400V	230V	400V	Y
230V	400V	400V	690V	Δ
Tension simple	Tension composée	Tension d'un enroulement	Tension entre deux enroulements	

I.2.5. La plaque à borne

La plaque à borne du moteur comporte six bornes sur lesquelles sont connectés les enroulements du stator (par construction du fabricant). L'image ci-dessous décrit la disposition "croisée" des enroulements et donne l'appellation des bornes, soit X, Y, Z et U, V, W sur lesquelles l'arrivée de l'alimentation électrique se fait.

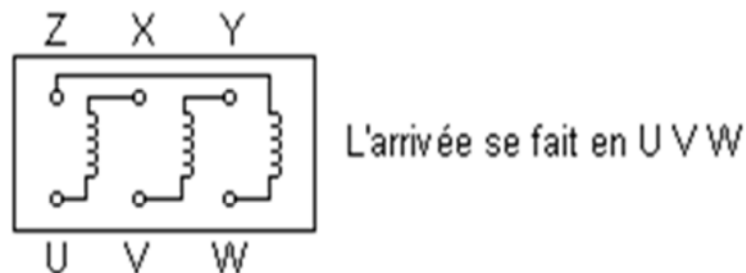


Figure (I-12) : bornes du moteur asynchrone sur la plaque à borne.

I.2.5.1. Couplage des enroulements :

La plaque à bornes d'un moteur asynchrone est toujours couplée soit en étoile ou en triangle ^[11]

- Couplage étoile :

Les trois enroulements ont un point commun. Le couplage en étoile des enroulements est réalisé en mettant en place des barrettes de couplage entre les bornes X, Y et Z.

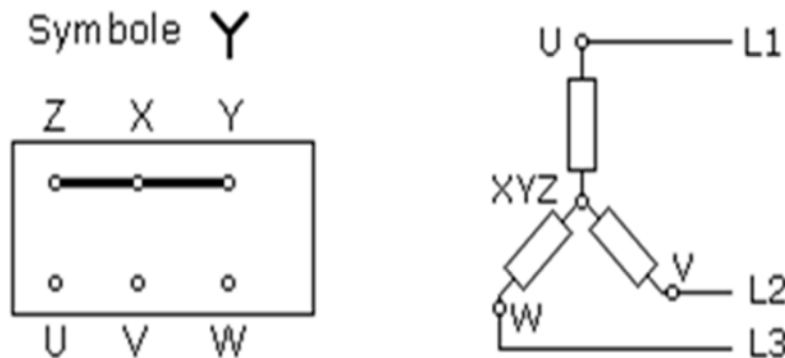


Figure (I-13) : couplages des enroulement en étoile.

➤ Couplage triangle :

Les trois enroulements sont montés en série et les points communs relient à chaque phase du réseau. Pour le couplage en triangle, trois barrettes sont installées pour coupler les bornes UZ, VX et WY.

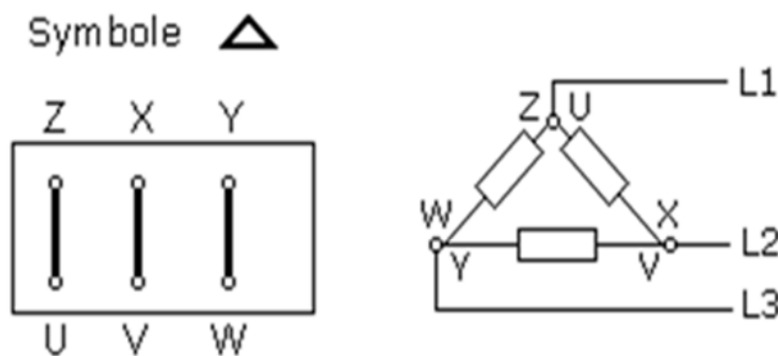


Figure (I-14) : couplage des enroulements en triangle. ^[11]

I.2.6. Démarrage des moteurs asynchrones triphasé

Le moteur asynchrone d'induction qui est le moteur de base de l'industrie actuelle convient surtout pour les commandes à vitesse constante. Sa mise en vitesse, au moment du démarrage, est liée à la surintensité admissible par l'installation électrique au couple nécessaire à la machine entraînée et à la durée de démarrage admissible. ^[12]

I.2.6.1. Démarrage direct :

C'est le plus simple qui ne peut être exécuté qu'avec le moteur asynchrone à rotor à cage. Les enroulements du stator sont couplés directement sur le réseau.

I.2.6.2. Démarrage étoile-triangle :

Ce type de démarrage est réservé aux machines démarrant à vide ou dont le couple résistant est faible. L'intensité de démarrage est divisé par trois, mais le couple de démarrage aussi (proportionnelle au carré de la tension d'alimentation des enroulements). Lors du démarrage étoile-triangle, la machine est d'abord connecté au réseau avec un couplage étoile, puis une fois démarré, on passe au couplage triangle. Ce démarrage est très économique puisqu'il est réalisé simplement à l'aide de contacteurs.

I.2.6.3. Démarrage statorique :

Au moment de démarrage lorsque l'on diminue la tension aux bornes des enroulements du moteur on diminue l'intensité et le couple au démarrage du moteur, pour cela, on place une (ou plusieurs) résistances en série dans la ligne du moteur.

I.2.6.4. Démarrage rotorique :

Lors d'un démarrage rotorique, des résistances de puissance sont insérées en série avec les enroulements du rotor. Ce type de démarrage permet d'obtenir un fort couple de démarrage avec des courants de démarrage réduits mais il ne peut être mis en œuvre qu'avec des machines à rotor bobiné muni de contacts glissants (bagues et balais) permettant les connexions électriques des enroulements rotoriques.

I.2.6.5. Démarrage par autotransformateur :

Il consiste à utiliser un autotransformateur qui est un appareil dont le circuit primaire est alimenté par le réseau et qui délivre à son secondaire une tension pouvant varier linéairement de 0 à 100% de la tension primaire. Ce démarrage s'effectue en alimentant le moteur par une tension réduite à travers l'autotransformateur puis en alimentant le moteur par une tension pleine de fonctionnement ; la tension est progressivement augmentée, l'intensité du courant ne dépassant pas la valeur maximale désirée.

I.2.7. Grandeurs électromécaniques caractéristiques ^[9]

I.2.7.1. Puissance et rendement

✓ La puissance électrique consommée (absorbée) sur le réseau en triphasé est :

$$P_a = UI \sqrt{3} \cos \varphi \quad (I-4)$$

Le $\cos \varphi$ du moteur est une caractéristique indiqué sur la plaque signalétique.

P_a : puissance absorbée.

U : tension entre les bornes du moteur (tension entre phases) (V).

I : courant en ligne (A).

$\cos\varphi$: facteur de puissance.

✓ La puissance mécanique utile :

$$P_u = C_u * \Omega \quad (I-5)$$

P_u : puissance utile.

C_u : Couple mécanique utile (N.m).

Ω : vitesse de rotation angulaire (rad/s) ; $\Omega = 2\pi n$ (n en tr/s).

✓ Rendement :

$$\eta = P_u / P_a \quad (I-6)$$

Remarque : Le rendement et le facteur de puissance sont liés à la charge du moteur. Plus le moteur fonctionne proche de ces caractéristiques nominales meilleures sont ces grandeurs. ^[13]

I.2.8.2. Grandeurs mécaniques : ^[14]

✓ Vitesse de synchronisme/vitesse de rotation :

Le champ magnétique tournant « tourne » à la vitesse de rotation N_s .

Le rotor tend à rattraper le champ tournant et tourne à une vitesse N ($N < N_s$).

✓ Glissement (g):

Le rotor tourne à la vitesse Ω plus petite que la vitesse de synchronisme Ω_s .

On dit que le rotor glisse par rapport au champ tournant.

Glissement « g » va dépendre de la charge, s'exprime en % de la vitesse de synchronisme.

$$g = \frac{N_s - N}{N_s} \quad (I-7)$$

Ω_s : Vitesse angulaire de synchronisation : $\Omega_s = \frac{2\pi N_s}{60}$ (rad/s).

$$g = \frac{\Omega_s - \Omega}{\Omega_s} \quad (I-8)$$

N_s : La vitesse du synchronisme du champ tournant (tr/min).

Ω : La vitesse angulaire de rotation du rotor : $\Omega = \frac{2\pi N}{60}$ (rad/s).

N : La vitesse de rotation du rotor (tr/min). ^[8]

I.3. Dispositifs industriels de Commande

La commande industrielle désigne l'ensemble des méthodes qui permettent de contrôler les performances d'un appareil électrique, d'une machine ou d'un système. Appliquée aux moteurs, la commande industrielle contrôle le démarrage, l'accélération, le sens de rotation, la vitesse, la décélération et l'arrêt des parties tournantes.

Tout circuit de commande comprend quelques composants de base raccordés entre eux de façon à assurer le contrôle désiré du moteur. Leurs dimensions peuvent varier selon la grosseur du moteur à commander, mais leur principe de fonctionnement reste le même. Avec seulement une dizaine de dispositifs de base, on réalise des montages de commande très complexes. ^[17]

Voici les principaux dispositifs :

I.3.1. Sectionneur ^[18]

Les sectionneurs isolent le circuit du moteur de celui de la source. Ils doivent pouvoir supporter indéfiniment le courant nominal ainsi que les courants de court-circuit pendant de courtes périodes. Ils comportent des contacts à couteaux et des fusibles de protection. Ils s'ouvrent et se ferment manuellement.

Le sectionnement n'a pas de pouvoir de coupure, il doit manipuler à vide.

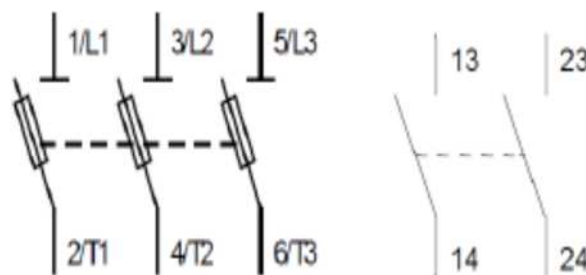


Figure (I-15) : Symboles-en circuit de puissance et en circuit de commande.

- **Interrupteur sectionneur :**



Figure (I-16) : symbole d'un interrupteur sectionneur.

L'interrupteur sectionneur a un pouvoir de coupure, peut être manipulé en charge.

- **Fusible :**

C'est l'élément comportant un fil conducteur, grâce à sa fusion, il interrompt le circuit électrique lorsqu'il est soumis à une intensité du courant qui dépasse la valeur maximale supportée par le fil.



Figure (I-17) : Symbole d'un fusible.

Il existe plusieurs types de fusibles :

- 1- **gF** : fusible à usage domestique, il assure la protection contre les surcharges et les courts-circuits.
- 2- **gG** : fusible à usage industriel. Protège contre les faibles et fortes surcharges et les courts-circuits.

Utilisation : Eclairage, four, ligne d'alimentation,...

- **aM** : cartouche à usage industriel, pour l'accompagnement moteur, commence à réagir à partir de $4 \times I_n$ (I_n est le courant prescrit sur le fusible), protège uniquement contre les courts-circuits.

Utilisation : Moteurs, transformateurs,...

I.3.2. Disjoncteurs manuels ^[17]

Les disjoncteurs sont conçus pour ouvrir et fermer manuellement le circuit d'un moteur et pour ouvrir le circuit automatiquement si le courant dépasse une limite prédéterminée. On peut réenclencher le disjoncteur après une ouverture anormale. Souvent, on utilise le disjoncteur manuel au lieu d'un sectionneur.

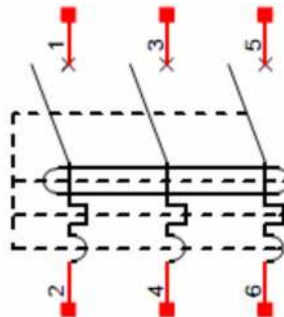


Figure (I-18) : Symbole d'un disjoncteur manuel.

I.3.3. Commutateurs à cames

Ces commutateurs comprennent une série de contacts fixes et autant de contacts mobiles actionnés par la rotation manuelle d'un arbre à cames. On les utilise pour la commande manuelle des moteurs de grues, calandres, pompes, etc.

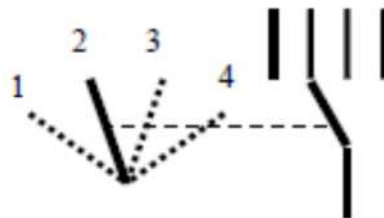


Figure (I-19) : Symbole d'un commutateur à came.

I.3.4. Boutons poussoir

Les boutons poussoirs sont des commutateurs actionnés par une pression du doigt et qui ouvrent ou ferment deux ou plusieurs contacts. Habituellement, ils ouvrent ou ferment momentanément un circuit.

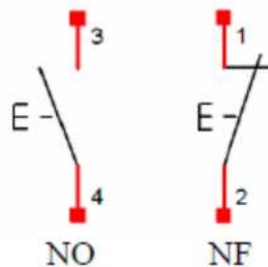


Figure (I-20) : Symbole d'un bouton poussoir.

I.3.4. Relais de phase

C'est un organe de commande qui se déclenche en cas de coupure d'une des trois phases ou dans le cas d'un déphasage prolongé. Dans un réseau triphasé, il surveille simultanément l'ordre des phases.

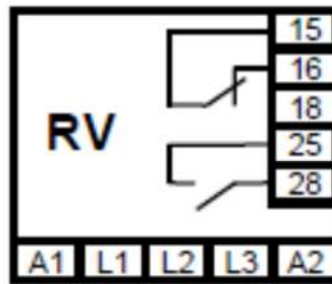


Figure (I-21) : Symbole d'un relais de phase.

I.3.5. Relais thermique

Le relais thermique est un relais de protection contre les surcharges.

I.3.5-1. Fonction

Relais thermiques permet de détecter les surcharges faibles et prolongées, contre les incidents d'origines mécaniques, chute de tension, déséquilibre des phases, manque d'une phase et de couper le circuit de commande de contacteur.

Le relais thermique est utilisable en courant continu et alternatif, Ils sont conçus pour couper l'alimentation si le moteur tire trop de courant pendant une période de temps prolongée. Pour ce faire, relais thermiques contiennent un (NC) Relais normalement fermé. Quand un courant excessif circule dans le circuit du moteur, le relais s'ouvre en raison de l'augmentation de température du moteur, la température de l'équipement, ou le courant de surcharge détecté, en fonction du type de relais.

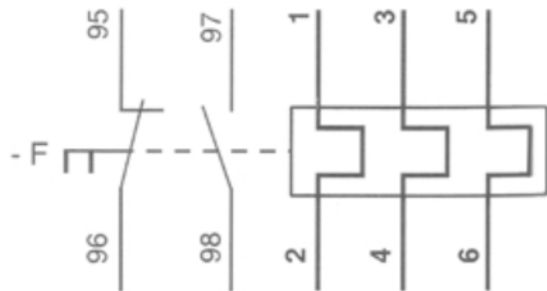


Figure (I-22) : Symbole d'un relais thermique.

I.3.6. Le Contacteur

I.3.6.1. Fonction

Le contacteur est un appareil de connexion à commande électrique actionnée à distance et automatiquement.

Le contacteur réalise les fonctions suivantes

- Commande des récepteurs, le contacteur est capable de supporter et d'interrompre des courants dans les conditions normales de fonctionnement du récepteur.
- Protège des chutes de tensions et des coupures de courant : le contacteur est un appareil monostable. En cas de coupure de courant ou de tension, le contacteur s'ouvre et seule une commande volontaire permet de le refermer.
- Protège les récepteurs des surcharges : le contacteur associé à un dispositif de détection (relais thermique) est capable de supporter et d'interrompre des courants de surcharge.

I.3.6.2. Constitution

Le contacteur de puissance comporte 4 ensembles fonctionnels : Le circuit principal ou circuit de puissance, Le circuit de commande, L'électro-aimant, Le circuit auxiliaire (bloc supplémentaire).

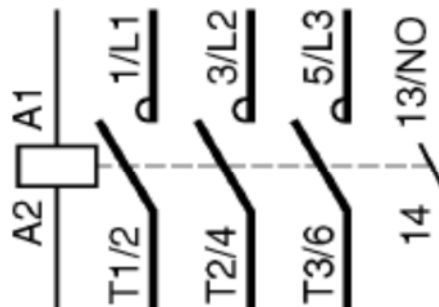


Figure (I-23) : Symbole d'un contacteur.

✓ Le circuit principal ou circuit de puissance :

C'est un ensemble de pièces conductrices du courant principal du contacteur. Il est constitué de : Contacts principaux (1/L1- T1/2, 3/L2-T2/4, 5/L3-T3/6).

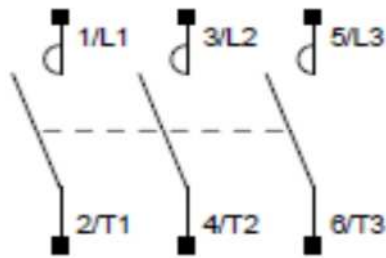


Figure (I-24) : circuit de puissance d'un contacteur.

✓ **Le circuit de commande :**

Il comprend un ou deux contacts de commande pour par exemple l'auto maintien. Le contact de commande (NO/13-14 ; NC/21-22).

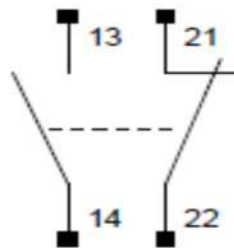


Figure (I-25) : circuit de commande d'un contacteur.

✓ **L'organe moteur :**

L'électro-aimant est l'élément moteur du contacteur. Il comprend :

Une bobine alimenté sous une tension alternative ou continue en 24V ; 48V ; 110V ; 230V ; 400V. Elle est repérée par les bornes A1, A2.

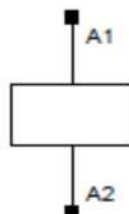


Figure (I-26) : l'organe mécanique.

✓ Le circuit auxiliaire :

Le circuit auxiliaire qui est réalisé par l'addition d'un bloc auxiliaire est destiné à remplir des fonctions autres que celles assurées par les deux premiers circuits :

-verrouillage électrique, signalisation, temporisation.

Il comporte essentiellement des contacts auxiliaires instantanés et temporisés.

Ils ont la particularité de s'installer sur la face.

Conclusion :

Dans ce chapitre, on a présenté et définit les différents éléments d'une station de pompage, tel que la pompe (centrifuge) et sa machine d'entraînement qui est le moteur asynchrone triphasé, ainsi que les dispositifs industriels de commande.

Chapitre II :

Arduino et les capteurs.

Introduction :

L'électronique est de plus en plus remplacée par l'électronique programmée, pour la réalisation de notre travail, On a utilisé une carte d'acquisition de type Arduino afin d'assurer la communication entre les deux parties à savoir, la partie commande (ordinateur) et les pompes afin de pouvoir commander et piloter le système.

Dans ce chapitre, on présente la description de la carte Arduino qui sera utilisée pour notre application. En effet, les différentes parties constituant cette carte seront détaillées, à savoir un microcontrôleur, des ports entrées/sorties, des mémoires. Aussi la présentation des capteurs.

II.1. Définition du module Arduino

Le module Arduino est un circuit imprimé en matériel libre, dont les plans de la carte elle-même sont publiés en licence libre dont certains composants de la carte : comme le microcontrôleur et les composants complémentaires qui ne sont pas en licence libre. Un microcontrôleur programmé peut analyser et produire des signaux électriques de manière à effectuer des tâches très diverses. Arduino est utilisé dans beaucoup d'applications comme l'électrotechnique industrielle et embarquée ; le modélisme la domotique mais aussi dans des domaines différents comme le pilotage d'un robot, commande des moteurs et faire des jeux de lumières, communiquer avec l'ordinateur, commander des appareils mobiles (modélisme). Chaque module d'Arduino possède un régulateur de tension +5 V et un oscillateur à quartz 16 MHz (ou un résonateur céramique dans certains modèles).^[19]

II.1.1. Présentation de la carte ARDUINO Mega2560

Le système de contrôle est réalisé grâce à une carte Arduino, on a choisi ce contrôleur pour son faible coût et sa facilité de développement de programme. Figure (II-1). Le modèle choisi est le 2560 Mega R3 car il présente plusieurs entrées et de sorties. La carte Arduino Mega2560 est une carte à microcontrôleur basée sur ATmega2560.^[32]

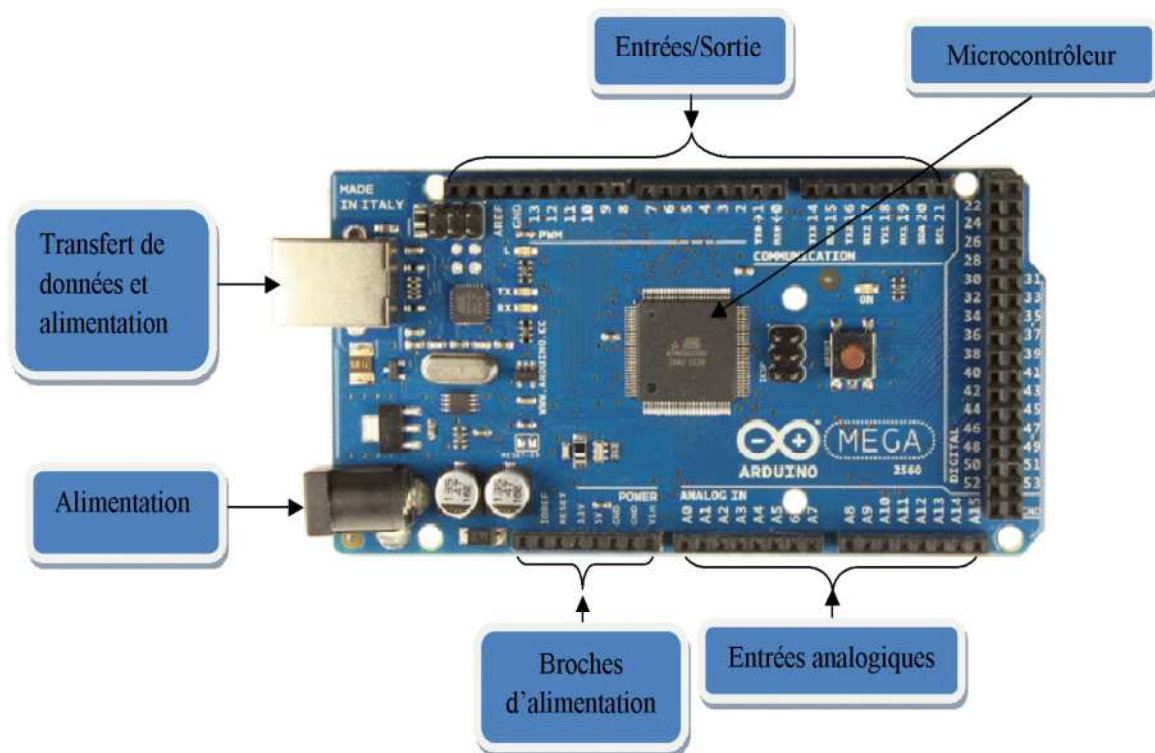


Figure (II-1) : Architecture de la carte Arduino Mega2560^[32]

II.1.1.1. Description de la carte Arduino Mega2560

Cette carte dispose :^[20]

- De 54 broches numériques d'entrées/sorties (dont 14 peuvent être utilisées en sorties PWM (largeur d'impulsion modulée)).
- De 16 entrées analogiques (qui peuvent également être utilisé en broches entrées/sorties numériques).
- De 4 UART (port série matériel).
- D'un quartz 16 Mhz.
- D'une connexion USB.
- D'un connecteur d'alimentation jack.
- D'un connecteur ICSP (programmation in-circuit).
- Et d'un bouton de réinitialisation (reset).

L'intérêt principal des cartes Arduino est leur facilité de mise en œuvre, Arduino fournit un environnement de développement s'appuyant sur des outils open source. Le chargement du programme dans la mémoire du microcontrôleur se fait de façon très simple par le port USB. En outre, des bibliothèques de fonction clé en main sont également fournies pour l'exploitation d'entrée-sorties courants : gestion des E/S TOR, gestion des convertisseurs ADC, génération de signaux PWM, exploitation de bus 12C, exploitation de servomoteurs...etc.

La carte Arduino Mega2560 contient tout ce que est nécessaire pour le fonctionnement du microcontrôleur, pour pouvoir l'utiliser et se lancer, il suffit simplement de la connecter à un ordinateur à l'aide d'un câble USB (ou de l'alimenter avec un adaptateur secteur ou une pile, mais ceci n'est pas indispensable, l'alimentation étant fournie par port USB).

La carte Arduino Mega2560 est compatible avec les circuits imprimés prévus pour les cartes Arduino Uno ,Duemilanove ou Décimila.

Tableau (II-1) : les caractéristiques de la carte Arduino Mega2560 ^[32]

Microcontrôleur	ATmega2560
Tension de fonctionnement	5 V
Tension d'alimentation (recommandée)	7-12 V
Tension d'alimentation (limites)	6-20 V
Broches d'entrées /sorties numériques	54 (dont 14 disposent d'une sortie PWM)
Broches d'entrées analogiques	16(utilisables en broches E/S numériques)
Intensité maxi disponible par la broche E/S (5V)	40 mA
Intensité maxi pour la sortie 3.3 V	50 mA
Mémoire Programme Flash	256 KB dont 8 KB sont utilisés par boot loader
Mémoire SRAM (mémoire volatile)	8 KB
Mémoire EEPROM (mémoire non volatile)	4 KB
Vitesse d'horloge	16 MHz

II.1.1.2. Le microcontrôleur

C'est le cerveau de la carte, permettant de recevoir le programme crée et stocker dans sa mémoire, puis l'exécuter. Grâce aux instructions de ce programme, il peut accomplir des taches souhaitées, qui peuvent être: faire clignoter une LED, afficher des caractères sur un écran, commander des moteurs comme dans mon système, envoyer des données à un ordinateur...etc.

II.1.1.3. Présentation du microcontrôleur ATmega2560

II.1.1.3.1 : Architecture :

Le cœur AVR combine un riche jeu de 13 instructions avec 32 registres spéciaux travaillant directement avec l'unité arithmétique et logique (ALU). Ces registres spéciaux permettant à deux registres indépendants d'être en accès direct par l'intermédiaire d'une simple instruction et d'être exécuté en un seul cycle d'horloge. Cela signifie que pendant un cycle d'horloge ALU exécute l'opération et le résultat et stocker en arrière dans le registre de sortie, le tout dans cycle d'horloge.

Les registres spéciaux sont dit aussi registre d'accès rapide et 6 des registres peuvent être employés comme trois registres d'adresse 16 bits pour l'adressage indirects d'espace de données (X, Y et Z), le troisième Z est aussi employé comme indicateur d'adresse pour la fonction de consultation de table des constantes. [23]

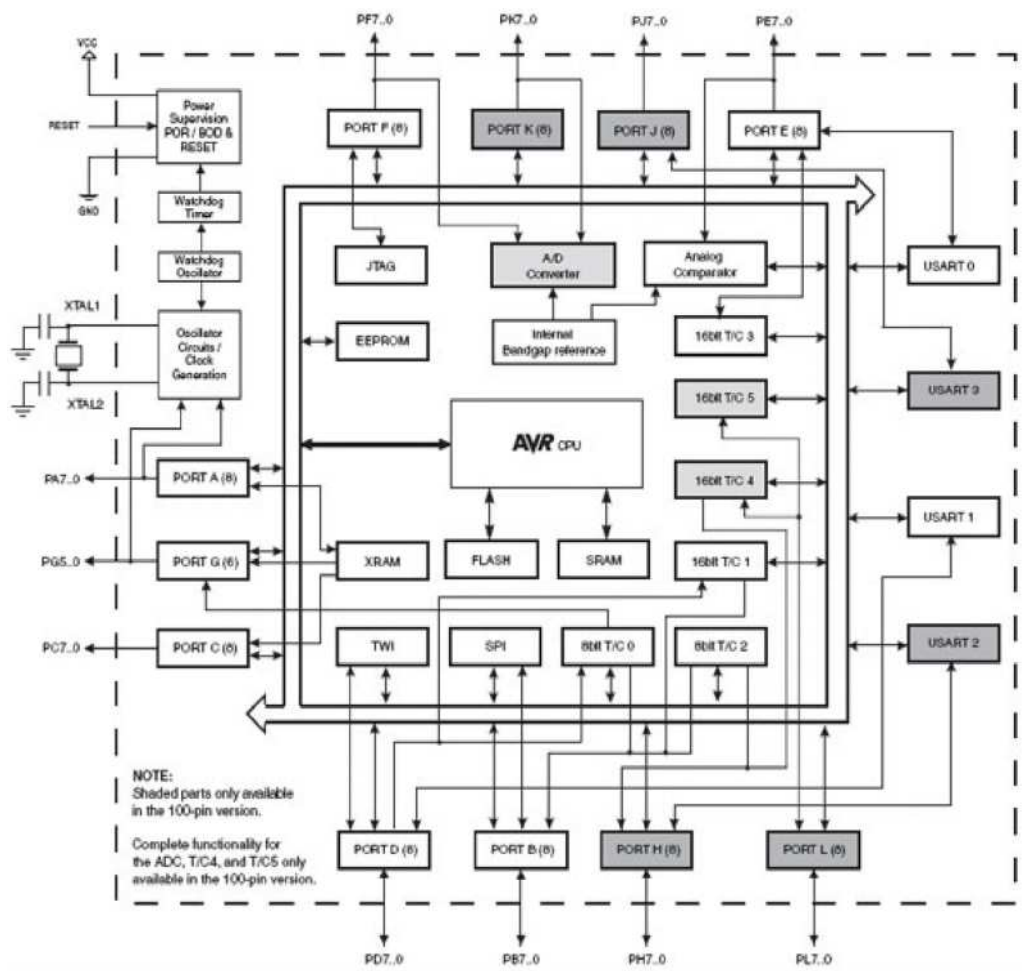


Figure (II-2) : Architecture interne de l'Atmega2560.

II.1.1.3.2 : Brochage :

Le brochage du microcontrôleur Atmega2560 est montré sur la figure (II.3) et dans cette figure on visualise le label de chacun des 100 pins ainsi que leurs fonctions alternatives.

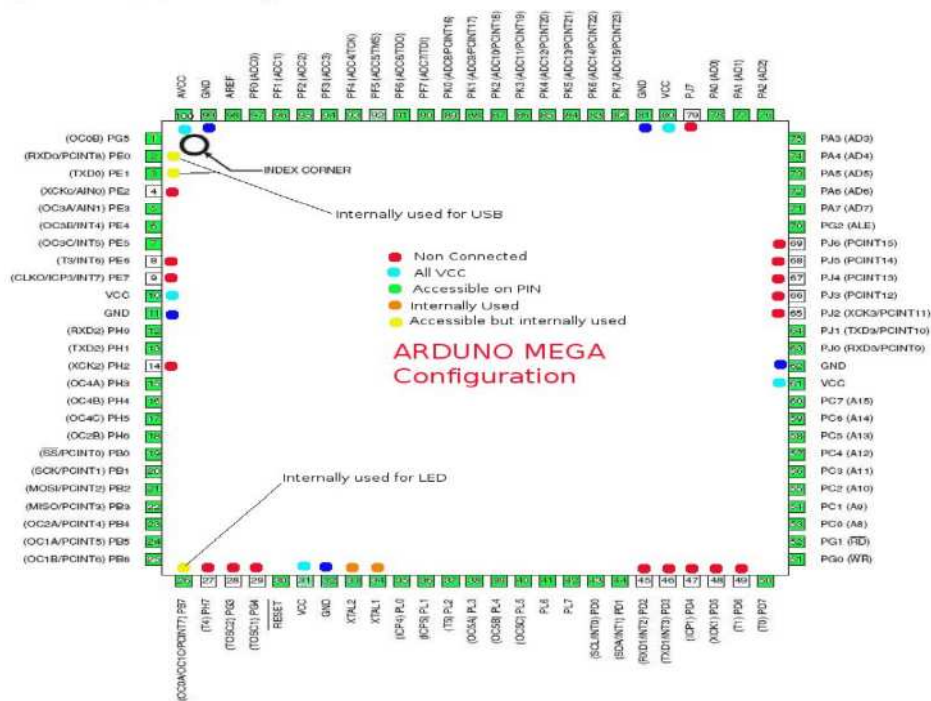


Figure (II-3) : Les différents pins de microcontrôleur.

II.1.1.4. description des broches [19]

Ports : Le microcontrôleur dispose de 11 ports (port A...port L) chaque port est un port d'entrée-sortie à 8 bits bidirectionnel avec des résistances internes de tirage (pull-up) (choisi pour chaque bit).

- **Port A (PA7...PA0) :** le port A est un port d'entrée-sortie à 8 bits bidirectionnel avec des résistances internes de tirage (choisi pour chaque bit). Il sert aussi pour les entrées analogiques du convertisseur A/D. Le port A (comme le B, C et D) est en position trois états quand une condition de reset devient active, même si l'horloge ne court pas.
- **Port B (PB7...PB0) :** le port B est un port d'entrée-sortie à 8 bits bidirectionnel avec des résistances internes de tirage (choisi pour chaque bit) .Il sert aussi de comparateur analogique (sortie sur PB2, PB3), ou de SPI.
- **Port C (PC7...PC0) :** le port C c'est un port d'entrée-sortie à 8 bits bidirectionnel avec des résistances internes de tirage (choisi pour chaque bit) .Il sert aussi comme oscillateur pour le Timer/Compteur 2 et d'interface 12C.

- **Port D (PD7...PD0) :** Le port D c'est un port d'entrée-sortie à 8 bits bidirectionnel avec des résistances internes de tirage (choisi pour chaque bit) .Il sert aussi d'USART et d'entrée pour les interruptions externes.
- **Port E (PD7...PD0):** Est un port de 8 bits bidirectionnel E/S avec des résistances pull-up interne (sélectionné pour chaque bit). Les tampons de sortie port E ont des caractéristiques d'entraînement symétriques à la fois haute évier et la capacité de la source. Comme entrées, les broches de port E qui sont tiré faible seront la source de courant si les résistances pull-up sont activées. Les broches de port E sont tri-indiqué lorsqu'un état reset devient actif, même si l'horloge ne fonctionne pas.
- **Port F (PD7...PD0) :** Sert des entrées analogiques au convertisseur A/N. F port sert également d'orifice 8 bits bidirectionnel E/S, si le convertisseur A/N n'a pas été utilisée. Broche de port peuvent fournir une résistance interne pull-up (sélectionnés pour chaque bit les tampons de sortie F du port ont des caractéristiques d'entraînement symétriques à la fois haute évier et la capacité de la source. Comme entrées, port F broches qui sont à l'extérieur au niveau bas seront la source de courant si le pull-up résistances activées.
- **Port G (PD5...PD0) :** Est un 6-bit port I/O avec des résistances de pull-up internes (sélectionnés pour chaque bit). Comme sortie le port G symétriques ont conduire des caractéristiques à la fois haute évier et la capacité de la source. Comme entrées, les broches de port G qui sont à l'extérieur très faible seront la source de courant si les résistances de tirage sont activées. Les broches de port G sont tri-indiqué lorsqu'une condition de remise à zéro devient actif même si l'horloge ne fonctionne pas.
- **Port H (PD7...PD0) :** Est un port de 8 bits bidirectionnel E/S avec des résistances pull-up interne (sélectionnés pour chaque bit). Les tampons de sortie port H ont des caractéristiques d'entraînement symétriques à la fois haute évier et la capacité de la source. Comme entrées, port H broches qui sont extérieur tiré faible sera la source de courant si les résistances pull-up sont activés. Les broches de port H sont tri indiqué lorsqu'un état reset devient actif, même si l'horloge ne fonctionne pas.
- **Port J (PD7...PD0) :** Est un port de 8 bits bidirectionnel E/S avec des résistances pull-up internes (sélectionnés pour chaque bit). Les tampons de sortie port J ont des caractéristiques d'entraînement symétriques à la fois haute évier et la capacité de la source. Comme entrées, port J broches qui sont extérieur tiré faible sera la source de courant si les résistances pull-up activées. Les broches port J sont tri-indiqué lorsqu'un état reset devient actif, même si l'horloge ne fonctionne pas.
- **Port K (PD7...PD0) :** Sert des entrées analogiques au convertisseur A/N. Port K est un port de 8 bits bidirectionnel E/S avec des résistances internes de tirage (sélectionnés pour chaque bit). Les tampons de sortie port K ont des caractéristiques d'entraînement symétriques à la fois haute évier et la capacité de la source. Comme entrées, port K broches qui sont extérieur tiré faible sera la source de courant si les résistances de pull-up sont activés. Les broches de port K sont tri-indiqué lorsqu'un état reset devient actif, même si l'horloge ne fonctionne pas.

- **Port L (PD7...PD0)** : Est un port de 8 bits bidirectionnel E/S avec des résistances pull-up internes (sélectionné pour chaque bits) .Les tampons de sortie de port L ont des caractéristiques d'entraînement symétrique à la fois haute évier et la capacité de la source. Comme entrées port L broches qui sont extérieur tiré faible sera la source de courant si les résistances interne de tirage ont activés. Les broches de port L sont tri-indiqué lorsqu'un état reset devient actif, même si l'horloge ne fonctionne pas.
- **RESET** : Déclenche par un front descendant maintenue plus de 50 ns il produira le Reset du microcontrôleur, même si l'horloge ne court pas.
- **TAL1** : Entrée de l'oscillateur externe ou libre pour l'horloge interne
- **XTAL2** : Production de l'amplificateur d'oscillateur.
- **AVCC** : Est une broche de tension d'alimentation pour le convertisseur A /D qui doit être connectée à VCC via un filtre passe-bas pour éviter les parasites.
- **AREF** : Est l'entrée de références analogique pour le convertisseur A/D avec une tension dans la gamme de 2 V à AVCC avec filtre passe bas.
- **AGND**: Masse analogique. Si la masse analogique est séparé de la masse générale, brancher cette broche sur la masse analogique, sinon, connecter cette broche à la masse générale GND.
- **VCC** : Broche d'alimentation du microcontrôleur (+3 à +5V).
- **GND** : Broche d'alimentation.

II.1.1.5. Caractéristiques du microcontrôleur ATMEGA 2560

Le microcontrôleur ATmega2560 dispose :

- 135 instructions puissantes, la plupart d'exécution simple de rythme.
- 32x8 registres d'usage universel de fonctionnement.
- Opération entièrement statique.
- Jusqu'à 16 MIPS de sorties à 16 Mhz.
- Multiplicateur de cycle de Sur-Morceau2.
- Mémoires non-volatile de programme et de données.
- Bytes 64K/128K/256K de flash Individu-Programmable de Dans-Système.
- Résistance : 10.000 écrire/cycles d'effacement.
- 4K Bytes EEPROM.
- Serrure de programmation pour la sécurité de logiciel.
- Quatre canaux à 8 bits de PWM.
- Puissance sur la remise et la détection programmable d'arrêt partiel.

Certaines broches peuvent avoir plusieurs fonctions différentes choisies par programmation, on distingue :

Interface série UART :

L'USART est l'abréviation de (Universal Synchronous and Receiver and Transmitter), l'application principale de ce périphérique est la communication entre le microcontrôleur et l'ordinateur.

L'ATmega2560 à quatre port séries (UART0, UART1, UART2, UART3), chaque port à une entrée R_x et une sortie T_x.

Interface série SPI :

L'interface SPI est l'abréviation (Serial Peripheral Interfacel Synchronous), soit Interface Série Synchrone. Contrairement à l'UART et comme son nom l'indique, ce type de périphérique génère les signaux d'horloge de synchronisation.

L'interface SPI permet le transfert de données ultrarapide synchrone entre l'ATmega et des périphériques sous entre plusieurs dispositifs.

Interface série 12C ou TWI :

L'interface à deux conducteurs périodiques TWI est dérivé de l'interface 12C de Philips, c'est une nouveauté du modèle ATmega, il n'existe pas dans les versions précédentes AT89 et AT90.

Le protocole TWI permet de connecter jusqu'à 111 systèmes différents employant seulement deux lignes de bus bidirectionnelles, un pour l'horloge SCL (horloge de synchronisation du bus) et un pour les données SDA (Donnée transmise ou reçus du bus) , le seul matériel externe nécessaire pour la mise en œuvre du bus est un simple jeu de résistance pour chacune des lignes (R1,R2).

Tous les systèmes connectés au bus ont des adresses individuelles et les mécanismes de contrôle sont inhérents au protocole 12C.

II.1.1.6. Plan mémoire

Trois types de mémoires sont utilisés dans la série ATmega2560, la mémoire de programme

FLASH, la mémoire de donnée SRAM et la mémoire morte de type EEPROM.

- **La mémoire morte :**

La mémoire morte est de type EEPROM d'accès plus complexe contiendra la configuration du programme et les données importantes qui seront sauvé pendant l'absence de courant électrique. On peut écrire jusqu'à 100.000 fois dans EEPROM. La taille de l'EEPROM est en fonction du modèle de microcontrôleur ATmega (de 256 bits à 8 Ko).

EEPROM : Est une mémoire programmable est effaçable électriquement. La particularité de cette mémoire et de pouvoir garder les informations stockées longtemps même hors tension.

L'ATmega2560 contient 512 bytes de données dans la mémoire EEPROM. Elle est organisée comme un espace de données séparé, dans lequel des octets simples peuvent être lus et écrits.

L'EEPROM a une endurance d'au moins 100,000 cycles d'écriture/effacement.

L'espace mémoire d'EEPROM est accessible par l'utilisation de registres spéciaux d'accès. Le temps d'accès en écriture dans l'EEPROM est donné dans la table suivante :

Tableau (II-3) : temps d'accès en écriture.^[20]

Symbole	Nbre de cycle d'oscillateur RC interne	Temps typique de programmation
Ecriture EEPROM	8448	8.5 (ms)

- **La mémoire de donnée :**

La mémoire de donnée contient les 32 registres de travail, les 64 registres de commande et la mémoire SRAM pour les variables du programme.

- **La mémoire de programme :**

La mémoire programme permet de stocker et de faire fonctionner le microcontrôleur, il contient de 4 à 256 Ko de programme selon le modèle du microcontrôleur. Le nombre d'écriture sur cette mémoire est limité à 10.000, largement suffisant pour la majorité des applications.

Convertisseur Analogique/numérique (ADC) :

Le convertisseur analogique/numérique ADC intégré dans l'ATmega est doté de caractéristique très intéressante avec une résolution sur 10 bits, 8 entrées simultanées avec une non linéarité inférieur à 1/2 LSB, une erreur à 0V inférieur à 1 LSB, le temps de conversion est réglable de 65 à 260 micro seconde plus le temps est long, plus le résultat est précis. Prés de 1500 échantillons/seconde avec le maximum de résolution sont possible. Le convertisseur possède 7 entrées différentielles normales et 2 entrées différentielles avec gain optionnel de 10x (et 200x sur les boitiers carré TQFP et MLF uniquement). La tension de référence peut être externe (conversion de 0 à AREF analogique, le maximum étant interruption étant VCC) ou peut être interne avec la tension de référence de 2,56 V. l'ADC à une interruption sur conversion complète. La limitation du bruit en mode de sommeil est possible.

II.1.2. Alimentation de la carte ArduinoMega2560 ^[32]

La carte Arduino Mega2560 peut être alimentée soit via la connexion USB (qui fournit 5V jusqu'à 500 mA) ou à l'aide d'une alimentation externe. La source d'alimentation est sélectionnée par la carte.

L'alimentation externe (non-USB) peut être soit un adaptateur secteur (pouvant fournir typiquement de 3V à 12 V sous 500 mA) ou des piles (ou des accus). L'adaptateur secteur peut être connecté en branchant une prise 2.1 mm positif au centre dans le connecteur jack de la carte. Les fils en provenance d'un bloc de piles ou d'accus peuvent être insérés dans les connecteurs des broches de la carte appelées GND (masse ou 0V) et Vin (tension positive en entrée) connecteur d'alimentation.

La carte peut fonctionner avec une alimentation externe de 6 V à 20 V. cependant, si la carte est alimentée avec au moins de 7 V, la broche 5V pourrait fournir moins de 5 V et la carte pourrait être instable. Si on utilise plus de 12 V, le régulateur de tension de la carte pourrait chauffer et endommager la carte. Aussi, la plage idéale recommandée pour alimenter la carte 7V et 12 V.

La carte Arduino Mega2560 diffère de toutes les cartes précédentes car elle n'utilise pas le circuit intégré FTDI USB-vers-séries. A la place, elle utilise un Atmega8U2 programmé en convertisseur USB-vers-série.

Les broches d'alimentation sont les suivantes :

- VIN : la tension d'entée positive lorsque la carte Arduino est utilisée avec une source de tension externe (à distinguer du 5 V de la connexion USB ou autre source 5 V régulée). Vous pouvez alimenter la carte à l'aide de cette broche, ou, si l'alimentation est fournie par le jack d'alimentation, accéder à la tension d'alimentation sur cette broche.
- 5 V : la tension régulée utilisée pour faire fonctionner le microcontrôleur et les autres composants de la carte (pour info : les circuits électroniques numériques nécessitent une tension d'alimentation parfaitement stable dite (tension régulée) obtenue à l'aide d'un composant appelé un régulateur et qui est intégré à la carte Arduino). Le 5 V régulé fourni par cette broche peut donc provenir soit de la tension d'alimentation VIN via le régulateur de la carte, ou bien de la connexion USB(qui fournit du 5 V régulé) ou de tout autre source d'alimentation régulée.
- 3V3 : Une alimentation de 3.3V fournie par le circuit intégré FTDI (circuit intégré faisant l'adaptation du signal entre le port USB de votre ordinateur et le port série de l'Atmega) de la carte est disponible : ceci est intéressant pour certains circuit cette tension au lieu du 5V).L'intensité maximale disponible sur cette broche est de 50 mA. Figue II-4
- GND : Broche de masse(ou 0V). ^[33]

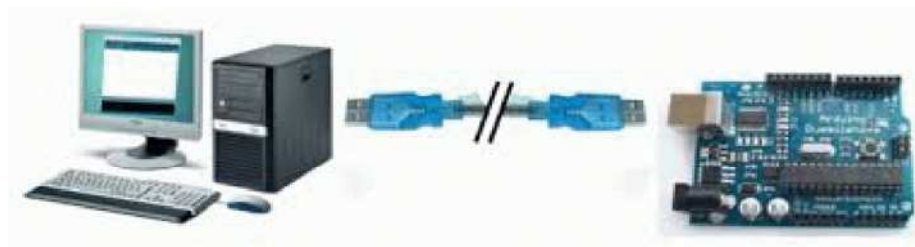


Figure (II-4) : Alimentation de l'Arduino par un microcontrôleur

II.1.3. Entrées et sorties numériques ^[24]

Chacune des 54 broches numériques de la carte Mega peut être utilisées soit comme une entrée numérique, soit comme une sortie numérique, en utilisant les instructions `pinMod()`, `digitalWrite()` et `digitalRead()` du langage Arduino. Ces broches fonctionnent en 5V. Chaque broche peut fournir ou recevoir un maximum de 40mA d'intensité et dispose d'une résistance interne de rappel au plus (pull-up) (déconnecté par défaut) de 0-50 kOhms. Cette résistance interne s'active sur une broche en entrée à l'aide de l'instruction `digitalWrite(broche, HIGH)`.

II.1.4. Broches analogiques

La carte Mega2560 dispose de 16 entrées analogiques, chacune pouvant fournir une mesure d'une résolution de 10bits (c.-à-d. sur 1024 niveaux soit de 0 à 1023) à l'aide de très utile fonction `analogRead()` du langage Arduino. Par défaut, ces broches mesurent entre le 0V (valeur 0) et le 5V (valeur 1023), mais il est possible de modifier la référence supérieure de la plage de mesure en utilisant la broche AREF et l'instruction `analogReference()` du langage Arduino.

II.1.5. Communication

La carte Arduino Mega2560 dispose de toute une série de facilités pour communiquer avec un ordinateur, une autre carte Arduino, ou avec d'autres microcontrôleurs. L'Atmega2560 dispose de quatre UARTs (Universel Asynchrones Receveur Transiter ou émetteur asynchrone universel en français) pour communication série de niveau TTL (5 V) et qui est disponible sur les broches 0(R_X) et 1(T_X). Un circuit intégré Atmega8U2 sur la carte assure la connexion entre cette communication série de l'un des ports séries de L'Atmega2560 vers le port USB de l'ordinateur qui apparaît comme un port COM virtuel pour les logiciels de l'ordinateur. Le code utilisé pour programmer l'Atmega8U2 utilise le river standard USB COM, et aucun autre driver externe n'est nécessaire. Cependant, sous Windows, un fichier. Inf. est requis.

Le logiciel Arduino inclut une fenêtre terminal série (ou moniteur série) sur l'ordinateur et qui permet d'envoyer des textes simple depuis et vers la carte Arduino. Les LEDs RX et TX sur la carte clignote lorsque les données sont transmises via le circuit intégré Atmega8U2 utilisé en convertisseur USB-vers-série et la connexion USB vers l'ordinateur (mais pas pour les communications série sur les broches 0et 1).

Une librairie série logicielle permet également la communication série (limitée cependant) sur n'importe quelle broche numérique de la carte Mega.

II.1.6. Protection du port USB contre la surcharge en intensité ^[24]

La carte Arduino Mega2560 intègre un poly fusible réinitialisable qui protège le port USB de votre ordinateur contre la surcharge en intensité (le port USB est également limité à 500mA en intensité). Bien que la plupart des ordinateurs aient leur propre protection interne, le fusible de la carte fournit une couche supplémentaire de protection. Si plus de 500mA sont appliqués au port USB, le fusible de la carte coupera automatiquement la connexion jusqu'à ce que le court-circuit ou la surcharge soit stoppé.

II.2. Capteurs

II.2.1. Définition du capteur

Un capteur est un dispositif transformant l'état d'une grandeur physique observée en une grandeur utilisable, telle qu'une tension électrique, une hauteur de mercure, une intensité ou la déviation d'une aiguille. ^[13]

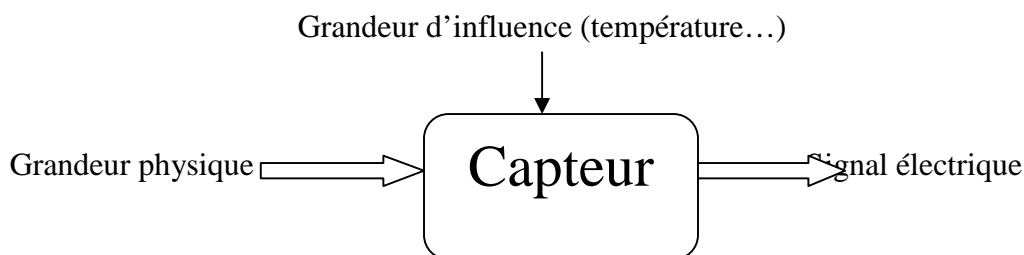


Figure (II-6) : principe général des capteurs.

II.2.2. Types de capteurs ^[13]

Les capteurs se classent selon leur principe de fonctionnement, on distingue deux types :

II.2.2.a. Capteurs passifs

Ils ont besoin, dans la plupart des cas, d'apport d'énergie extérieure pour fonctionner ex. : thermistance, photorésistance, potentiomètre, etc...). Ce sont des capteurs modélisable par une impédance. Une variation du phénomène physique étudié (mesuré) engendre une variation de l'impédance. Il faut leur appliquer une tension pour obtenir un signal de sortie.

II.2.2.b. Capteurs actifs

Ils se comportent comme des générateurs. Ils fournissent une grandeur active (tension, courant, charge électrique) sans nécessiter une source d'alimentation externe, on cite par exemple : le thermocouple, les capteurs piézoélectriques,

II.2.3. Type de détection

- Détection avec contact (le capteur doit entrer en contact physique avec un phénomène pour le détecter).
- Détection sans contact (le capteur détecte le phénomène à proximité de celui-ci).

II.2.4. Caractéristiques d'un capteur

Certains paramètres sont communs à tous les capteurs. Ils caractérisent les contraintes de mise en œuvre et permettent le choix d'un capteur :

- L'étendue de la mesure.
- La sensibilité.
- Le temps de réponse.
- La précision.
- Sa bande passante.
- Sa gamme de température.

II.2.5. Nature de l'information fournie par le capteur ^[29]

Les capteurs et leurs conditionneurs peuvent aussi faire l'objet d'une classification par type de sortie.

II.2.5.1. Analogique

La sortie est une grandeur électrique dont la valeur est une fonction de la grandeur physique mesurée par le capteur.

La sortie peut prendre une infinité de valeurs continues. Le signal de capteurs analogiques peut être du type :

- Sortie tension.
- Sortie courant.
- Etc.

II.2.5.2. Numérique

La sortie est séquence d'état logique qui, en se suivant, forment un nombre. La sortie peut prendre une infinité de valeurs discrètes. Le signal des capteurs numériques peut être de type :

- Train d'impulsion, avec un nombre précis d'impulsions ou avec fréquence précise.
- Code numérique binaire.
- Etc.

II.2.2.3. Logique

Ou capteur TOR (tout ou rien). La sortie est un état logique que l'on note 1 ou 0. La sortie peut prendre ces deux valeurs. Le signal des capteurs logiques peut être du type :

- Courant présent/absent dans un circuit ; Potentiel, souvent 5V / 0V ;
- Etc.

Quelques capteurs logiques typiques : capteur de fin de course, capteur de rupture d'un faisceau lumineux, divers capteurs de position.

II.2.2.3.a. Capteur de pression

Le capteur de pression (ou sonde de pression) est un dispositif destiné à convertir les variations de pression en variation de tension électrique.

Sont utilisés pour le contrôle et le suivi de nombreuses applications, ils permettent également être utilisés pour mesurer indirectement d'autres variables comme le débit de fluide/gaz, la vitesse, le niveau d'eau et l'altitude ...etc^[28]

Tous les capteurs de pression peuvent se ramener au schéma synoptique ci-dessous.

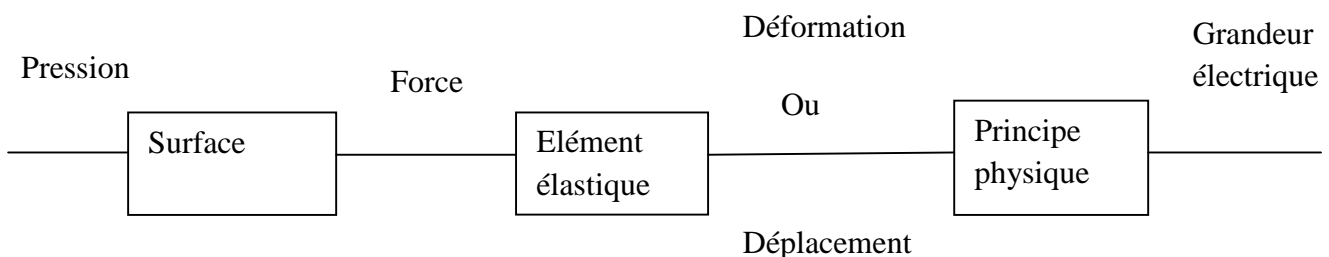


Figure (II-7) : Schéma synoptique d'un capteur de pression.

Le corps d'épreuve est l'élément mécanique qui, soumis aux variations de grandeurs à mesurer a pour rôle de transformer celle-ci en grandeur physique mesurable.

➤ pressostat (les capteurs de seuil de pression)

Le pressostat est un appareil électromécanique permettant d'automatiser la fourniture d'eau courante, dans une plage de pression définie par l'utilisateur. Le pressostat peut prendre aussi le nom de "contacteur manométrique", "manostat", ou "manocontact".

Les pressostats sont utilisés dans de nombreuses applications de systèmes de contrôle ou de régulation par exemple en provoquant le démarrage d'un compresseur d'air ou d'une pompe, Contrôler le niveau ou la présence d'eau, la pression d'un fluide.

Si la pression du circuit contrôlé descend au-dessous d'une limite déterminée. L'information ainsi transmise dépend de la comparaison entre la valeur ou consigne prédéfinie et la mesure réelle de l'équipement. Lorsque cet écart Consigne / Mesure dépasse un seuil, le pressostat envoie un signal qui prend une position de sécurité, et dans le cas contraire, la position revient à la normale, on parle alors de tout ou rien (TOR).

Il s'agit d'un contact électrique qui change d'état sous la pression. Le contrôle de niveau d'eau par pressostat ne peut se réaliser sans une chambre de compression. Ces deux organes sont alors reliés ensemble par une durite, formant ainsi un système totalement étanche capable d'emprisonner l'air nécessaire au bon fonctionnement du montage. Le tout est relié à la cuve de l'appareil (élément contenant l'eau).

Les pressostats ont une grande robustesse, ainsi qu'une excellente tenue dans les réglages malgré leur utilisation dans les environnements assez contraignants.

II.2.2-3.b. Capteur de niveau

Un dispositif électronique qui permet de mesurer la hauteur du matériau, en général du liquide, dans un réservoir ou un autre récipient. Une partie intégrante du contrôle de procédé dans de nombreuses industries, les capteurs de niveau se divisent en deux types principaux. Les capteurs de niveau de mesure de point est utilisé pour marquer une seule hauteur de liquide discrète - une condition de niveau prédéfinie. Généralement, ce type de capteur de niveau fonctionne comme une alarme haute, pour signaler une condition de débordement, ou en tant qu'indicateur pour une condition d'alarme basse. Le capteur de niveau continu est plus sophistiqué et peut assurer une surveillance de niveau de tout un système. Il mesure le niveau de liquide dans une plage, plutôt qu'à un seul point, ce qui produit une sortie analogique qui est directement corrélée au niveau de la cuve.

Flotteur :

Un détecteur de niveau à flotteur est utilisé pour la simple détection d'un seuil de niveau dans un liquide. Le flotteur du détecteur reste à la surface du liquide. Le détecteur est fixé à l'aide de son propre câble à un niveau adéquat pour l'application. Le processus de commutation est déclenché par les mouvements de basculement du détecteur, le câble servant d'articulation au dispositif.

Il s'agit d'un interrupteur à flotteur, en nylon, totalement étanche et neutre. Le flotteur est constitué d'un cylindre, qui coulisse librement le long de la tige principale. Par un jeu d'aimants, la position de ce cylindre permet d'ouvrir ou de fermer le circuit électrique enfermé dans la tige principale. En butée contre la tête du capteur, le courant passe. Dès que le flotteur s'éloigne de la tête, le courant ne passe plus. Ce type de capteur s'installe verticalement, de manière fixe, dans la cuve que l'on souhaite piloter. La position « tête en haut » ou « tête en bas », permet de choisir le mode de fonctionnement « normalement ouvert » ou « normalement fermé ». La sensibilité lors des mouvements, est tout à fait adaptée à nos besoins. Un léger clapotis en surface, n'entraîne pas un cycle d'ouverture / fermeture intempestif. ^[26]



Figure (II-8) : photo d'un flotteur

Conclusion :

Dans ce chapitre, notre étude est divisée en deux parties :

La 1^{ère} s'est focalisée sur la présentation de la carte Arduino, et ça commence par un bref historique et quelques détails concernant la création de cette carte. Puis on a présenté son côté matériel. La compréhension de données exposées dans ce chapitre on offre des bases de connaissances pour l'usage de la carte Arduino.

La 2^{ème} : Les généralités sur les capteurs, présentation de capteur de pression (pressostat), capteur de niveau (à flotteur).

Introduction

Le contrôle de pompes et surveillance de niveau de liquides dans des réservoirs et fosse se fait à l'aide de capteurs de niveau et de pression, et la carte de commande Arduino pour gérer le fonctionnement des pompes et éviter le débordement.

Dans ce chapitre on va définir la structure de mon réseau de remplissage plus définir l'ensemble des éléments constituant ce système qui est complètement automatisé. Ce chapitre consiste en deux parties : la première est consacrée à la conception alors que la deuxième est la réalisation.

III.1. Cahier de charge**III.1.1. Enoncé du problème**

On doit permettre de piloter deux pompes contrôlant le niveau d'un réservoir ou d'une fosse. En fonction du niveau d'eau, une pompe ou les deux doivent fonctionner. Le niveau en question est capté par trois sondes de niveau. Pour équilibrer leurs temps de fonctionnement, les pompes fonctionnent en alternance pendant une durée prédéfinie.

III.1.2. La solution envisagée

Si les deux premières sondes de niveau, sonde1 et sonde2 réagissent, la pompe 1 doit se mettre en marche après un bref retard de 2 secondes. Si la troisième sonde de niveau 3 réagit à son tour, la seconde pompe doit elle aussi se mettre en marche (au bout de 2 secondes).

Après 2 heures de fonctionnement, la première pompe est automatiquement remplacée par la seconde. La commutation s'effectue entre les deux pompes. Il en va de même lorsque la seconde pompe a fonctionné pendant 2 heures. Les comptages d'heures de service totalisent la durée complète de fonctionnement des pompes.

Avant que l'une des pompes soit démarrée, la pression du système est vérifiée par un capteur de pression. Si elle n'est pas établie dans les 5 secondes, les deux pompes fonctionnent jusqu'à ce que le niveau haut (sonde de niveau 1) soit atteint. Dans le même temps, une anomalie est signalée par témoin en feu fixe

Les pompes peuvent être réinitialisées manuellement par l'intermédiaire boutons poussoir.

III.1.3. Structure d'une station de pompage

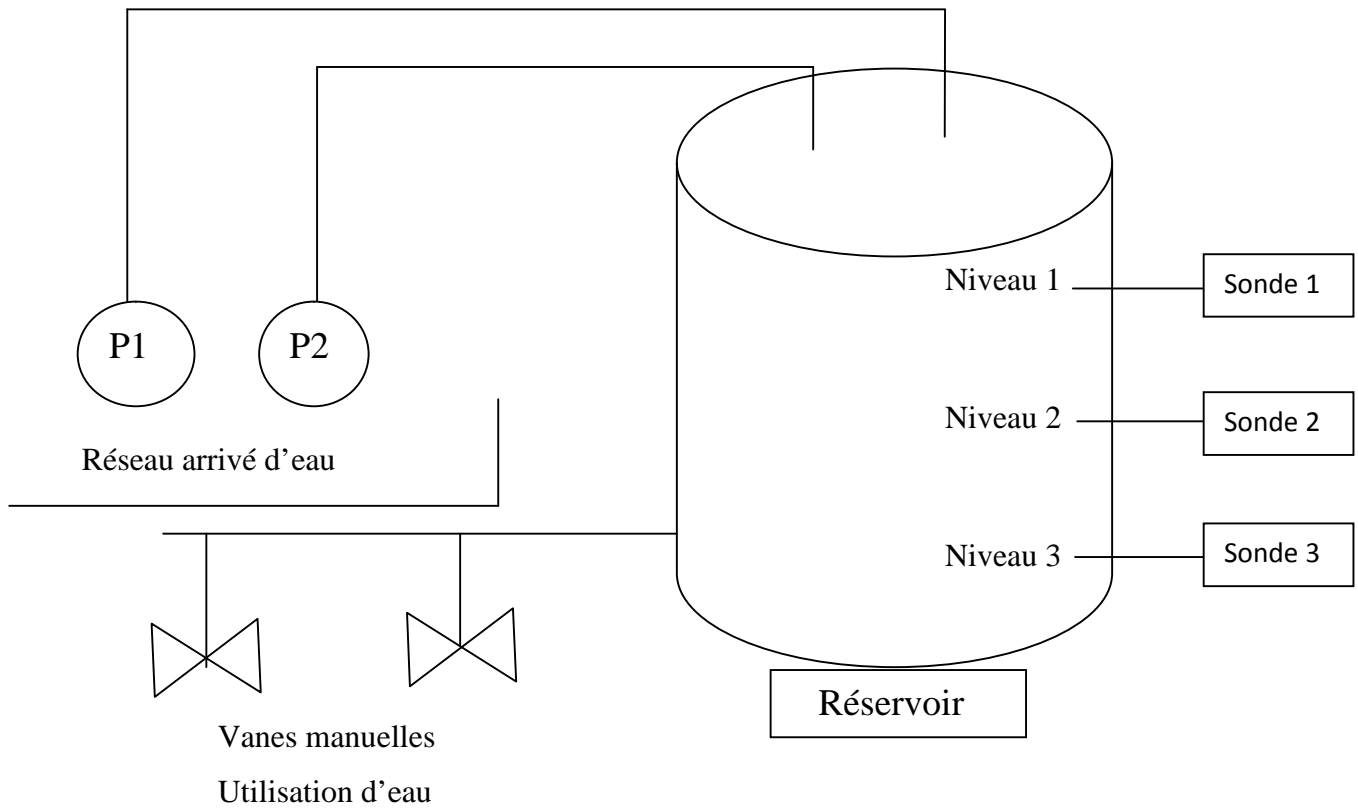


Figure (III-1) : Structure de la station de pompage.

III.1.4. Bronchement des différents modules à la carte ARDUINO

Broche D2 : Flotteur 1 (A).

Broche D3 : Flotteur 2 (B).

Broche D4 : Flotteur 3 (C).

Broche D5 : Pressostat (P).

Broche D6 : Commutateur automatique/manuel.

Broche D7 : Bouton marche/arrêt pompe 1 (P1).

Broche D22: Bouton marche/arrêt pompe 2 (P2).

Broche D23: LED de défauts capteurs

Broche D8 : Contacteur de ligne KM1 de la pompe 1.

Broche D9 : Contacteur étoile KM2 de la pompe 1.

Broche D10 : Contacteur triangle KM3 de la pompe 1.

Broche D11 : Contacteur de ligne KM1 de la pompe 2.

Broche D12 : Contacteur étoile KM2 de la pompe 2.

Broche D13 : Contacteur triangle KM3 de la pompe 2.

GND : 0 V.

VCC : 5V.

Ces modules sont connectés à la carte Arduino Méga via les broches numériques(D : Digital).

La figure ci-dessous résume les différentes connexions de la carte .

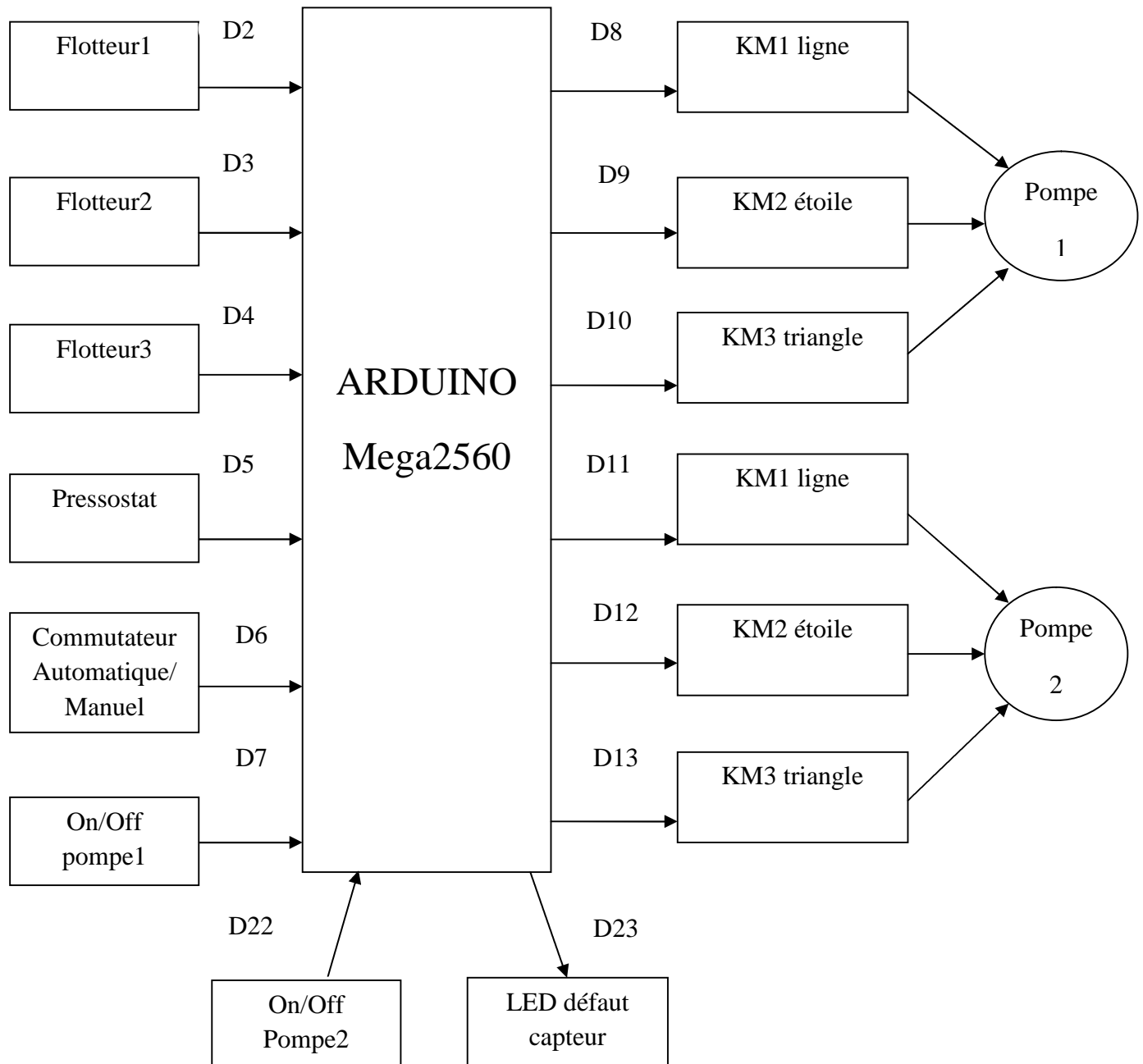


Figure (III-2) : Branchement des différents modules à la carte ARDUINO.

III.1.5. L'organigrammes de fonctionnement

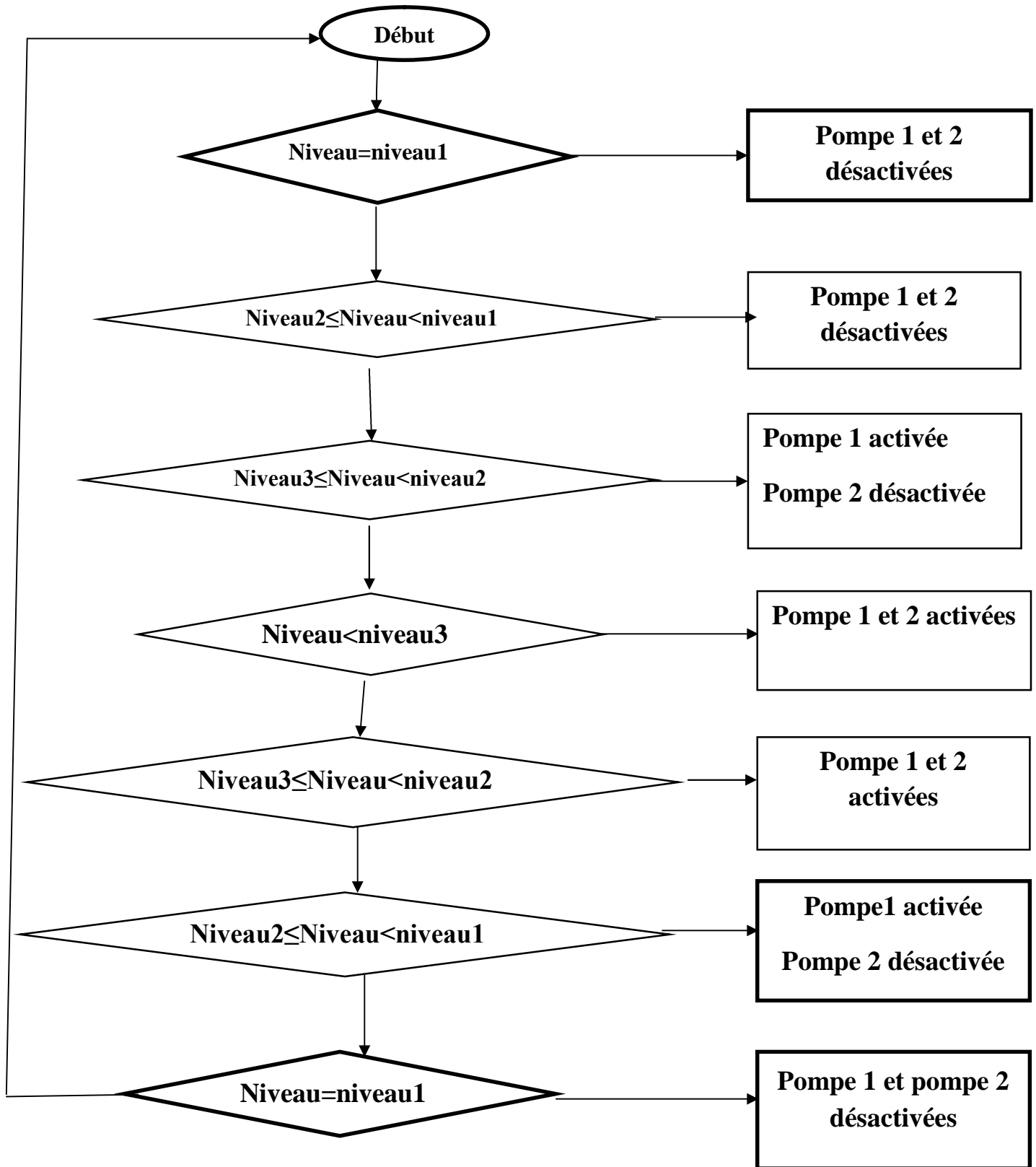


Figure (III-3) : l'organigramme de fonctionnement

III.1.6. Le temps de fonctionnement d'une pompe

Comme c'est mentionné dans le cahier de charge chaque pompe fonctionne 2 heures du temps pour se commuter à la deuxième et ça sera automatiquement avec la carte Arduino.

On a : 2 heures= 7200 secondes.

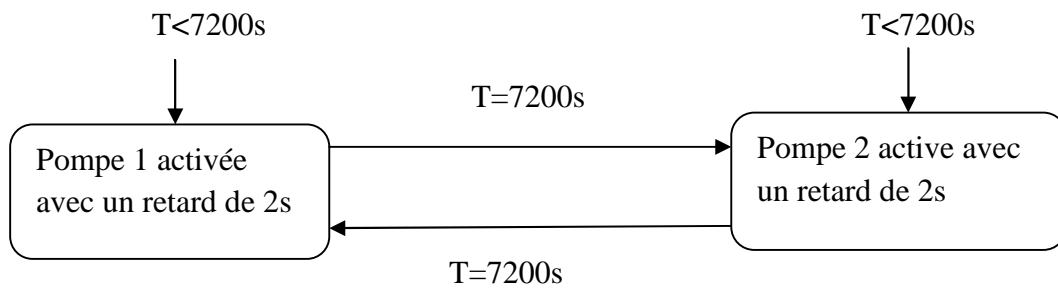


Figure (III-6) : Le temps fonctionnement d'une pompe.

III.1.7. Défauts capteurs

A : Capteur de niveau à flotteur 1.

B : Capteur de niveau à flotteur 2.

C : Capteur de niveau à flotteur 3.

X : LED.

0 : Absence d'eau.

1 : Présence d'eau.

III.1.7.1. La table de vérité

Tableau (III-1) : Table de vérité de défaut capteurs.

A	B	C	X
0	0	0	0
0	0	1	0
0	1	0	1

0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

III.1.7.2. Tableau de karneugh

Tableau (III-2) : Tableau de karneugh de défauts capteur.

BC \ A	00	01	11	10
0	0	0	0	1
1	1	1	0	1

$$X = B \overline{C} + A \overline{B}$$

III.1.7.3. Logigramme de défauts capteur

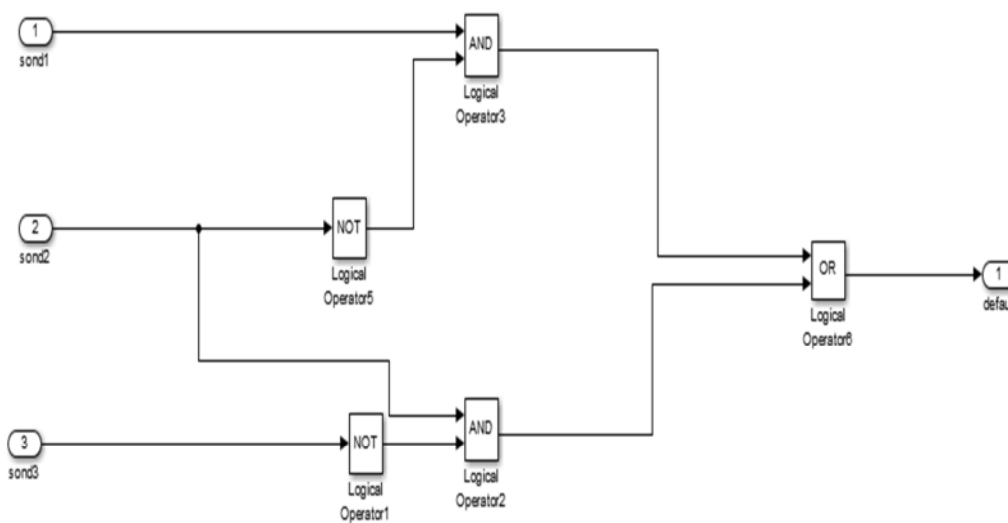


Figure (III-7) : Logigramme de défauts capteur.

III.1.8. Démarrage des moteurs

Le moteur asynchrone d'induction possède un fort couple de démarrage, mais il a l'inconvénient d'absorber de 4 à 8 fois son intensité nominale. Sur les gros moteurs ces courants importants entraînent des surcharges sur les lignes d'alimentations (fils, câbles, bornes) et sur les appareils de protection et de commande (fusible, sectionneur, contacteur, relais thermique...) d'où une usure, voir une destruction, prématurée des composants du démarreur. Pour réduire cet appel de courant et le démarrage brutal on utilise le démarrage étoile-triangle.

III.1.8.1. Démarrage étoile/triangle

III.1.8.1.a. Principe

Ce procédé de démarrage consiste à changer le couplage des enroulements du stator pour limiter l'appel de courant.

Le démarrage s'effectue en deux temps :

- ✓ 1^{er} temps : chaque enroulement du stator est alimenté sous une tension réduite (couplage étoile Y).
- ✓ 2^{ème} temps : chaque enroulement du stator est alimenté sous sa tension nominale (couplage triangle Δ).

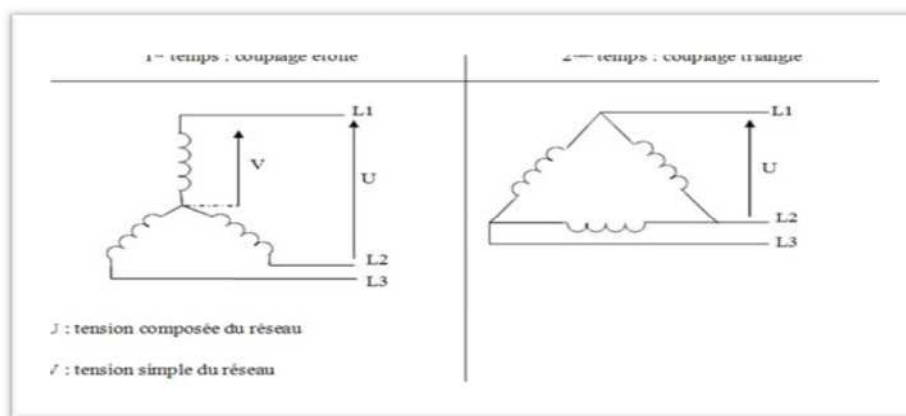


Figure (III-8) : Couplages étoile et couplage triangle. [12]

III.1.8.1.b. Analyse du fonctionnement au démarrage

Chaque enroulement est d'abord alimenté sous une tension V , puis sous une tension U :

$$U = \sqrt{3}V \quad (\text{III-1})$$

Au démarrage le moteur est couplé en étoile. La tension appliquée sur une phase est réduite, soit :

$$V = U / \sqrt{3} \quad (\text{III-2})$$

L'intensité absorbée (proportionnelle à la tension appliquée) est le 1/3 de celle qu'absorberait le moteur s'il démarrait directement en triangle. Le couple au démarrage (proportionnel au carré de la tension appliquée) et le couple maximum en étoile sont ramenés au 1/3 des valeurs obtenues en démarrage direct. La valeur du couple de démarrage atteint en général 0,5 fois le couple nominal.

Coupure (passage étoile triangle):

Le temps de passage entre les deux couplages doit être très bref.

Couplage triangle:

Un deuxième appel de courant se manifeste ; il est en fonction de la durée du couplage étoile et peut atteindre la valeur de pointe du démarrage direct. Cette pointe de courte durée provient du fait que les forces électromotrices qui subsistent au stator lors du couplage triangle ne sont pas en opposition de phase avec les tensions de ligne.

Le couple subit une forte pointe pour retomber rapidement à sa valeur nominale. [11]

III.1.8.1.c. Circuit de puissance

On dispose pour ce démarreur de trois contacteurs : $KM2$ qui couple le moteur en étoile au début du démarrage, $KM3$ qui le couple en triangle à la fin du démarrage et $KM1$ contacteur de ligne, dont le rôle est de commander le moteur.

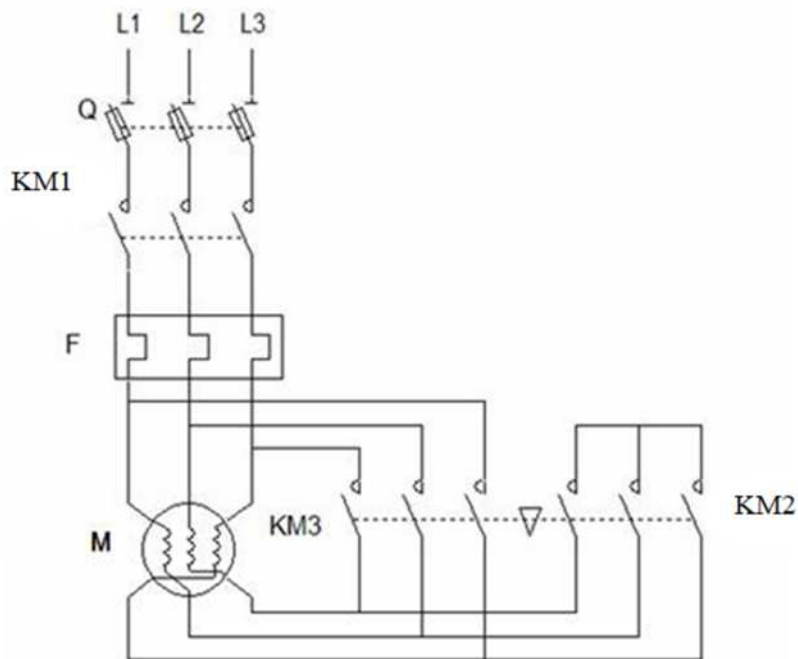


Figure (III-9) : Schéma de puissance de démarrage étoile triangle pour le moteur asynchrone triphasé. ^[12]

L1, L2, L3 : Réseau triphasé.

Q1 : Sectionneur tripolaire triphasé porte-fusibles

KM1 : Contacteur tripolaire "ligne".

KM2 : Contacteur tripolaire "étoile".

KM3 : Contacteur tripolaire "triangle".

F : Relais thermique de protection.

M : Moteur.

III.1.8.1.d. Les étapes de fonctionnement de circuit de puissance

Le contact KM1 est à fermeture différée, imposant un retard de 1s environ à KM3 au moment de la commutation étoile afin d'assurer un temps de coupure suffisant à KM2. Cela évite le risque de court-circuit entre deux fils de phase.

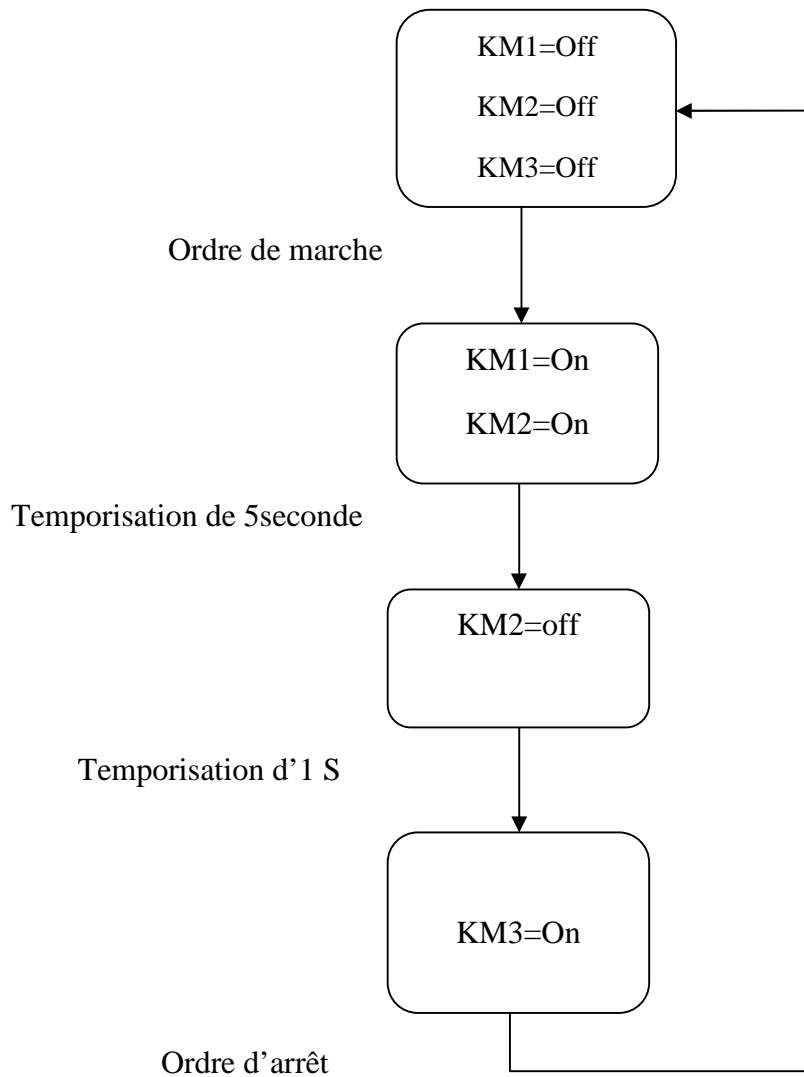


Figure (III-10) : les étapes de fonctionnent d’un circuit de puissance.

III.2. Matériels utilisés

Programmer la carte Arduino c’est bien, mais pouvoir la faire interagir avec le monde extérieur c’est encore mieux. L’Arduino ne peut pas commander du 230V directement, il faut pour cela passé par un relais. ^[13]

Voici la liste des composants utilisés durant ce travail :

Carte ARDUINO, Commutateurs, Contacteurs, Relais électromagnétique, Transistors, iodes, Résistances, LED.

III.2.1. Carte Arduino

Elle représente le composant essentiel, c'est le cerveau qui gère le système : Arduino Mega2560 (représenter dans le 2^{ème} chapitre).

III.2.2. Contacteur tripolaire NC1-32

Caractéristiques techniques de la bobine du contacteur:

Résistance de la bobine : 293 Ohms.

Tension de la bobine : 220V/230V.

Le calcul de l'intensité maximale de la bobine du contacteur :

On a, La loi d'Ohm $U=R*I$

Donc:

- $I=U/R$

$$I=230/293$$

$$I=0,78A$$

L'intensité bobine du contacteur NC1 est : 0,78A.

III.2.3 Le relais

Pour choisir le relais, il faut qu'il soit pilotable par le 5V du circuit de commande (Arduino). Il faut également qu'il supporte l'intensité maximale de la bobine du contacteur. Supportant au moins 0,78A/230V : comme le relais D005.

Caractéristiques du relais D005 :

Tension de bobine: 5VDC

Résistance de la bobine : 60 Ohm

Une sortie d'un Arduino supporte 20 mA au maximum. Pour exciter la bobine du relais on utilise un transistor entre les deux; il a la particularité de laisser passer un courant fort venant de l'alimentation entre son collecteur et son émetteur alors qu'on envoie un petit courant sur la base.

III.2.4. Le transistor

Le transistor est un composant à trois pattes : la base, le collecteur et l'émetteur. Pour l'utiliser en interrupteur télécommandé, appelé en commutation, la base est la commande, et l'interrupteur se situe entre le collecteur et l'émetteur. La jonction base – émetteur se comporte comme une diode.

Lorsque $V_{be} = 0V$, le transistor est bloqué, et comme $I_c = I_e = 0A$, le transistor se comporte comme un interrupteur ouvert entre le collecteur et l'émetteur.

Lorsque $V_{be} = V_{be_max}$, alors le transistor est passant. Pour que le transistor soit saturé, il faut que $I_b > I_c / H_{fe}$; (H_{fe} : gain). Il se comporte alors comme un interrupteur fermé.

Choix du transistor :

Il faut donc que l'intensité maximale admise par le collecteur soit supérieure à l'intensité parcourue par la bobine du relais. Il faut calculer I_c .

La résistance de la bobine de relais vaut 60 Ohms, et on va dire qu'en théorie le transistor consomme 0V ($V_{ce} = 0V$, ce qui donne une marge supérieure dans le calcul de l'intensité parcourant la bobine) donc: On a $U_{bob} = I_{bob} * R_{bob}$ (Loi d'Ohm), donc :

- $I_{bob} = U_{bob} / R_{bob}$
 $I_{bob} = (V_{cc} - V_{ce}) / R_{bob}$
 $I_{bob} = (5 - 0) / 60$
 $I_{bob} = 0,083A$

L'intensité de la bobine du relais est : 0,083 A

Il faut aussi qu'il supporte une tension de 5V entre la base et l'émetteur (tension de sortie de l'Arduino V_{be}), et également une tension de 5V entre le collecteur et l'émetteur (tension du circuit V_{ce}).

On a choisi un transistor NPN TIP122 puisque son $I_{c_max} = 5A$ (donc supérieur à $I_{bob} = 0.083A$), $V_{ce_max} = 100V$ (supérieur à 5V de circuit) et $V_{be_max} = 5V$ (égale à 5V de tension de commande V_{be}).

II-5. La diode

Choix de la diode :

La diode de roue libre va être branchée en parallèle avec la bobine du relais. Elle va devoir encaisser une tension de 5V et une intensité de 0,083A. Je choisis la 1N4007 et surtout elle supporte 1000V et un courant de 1A

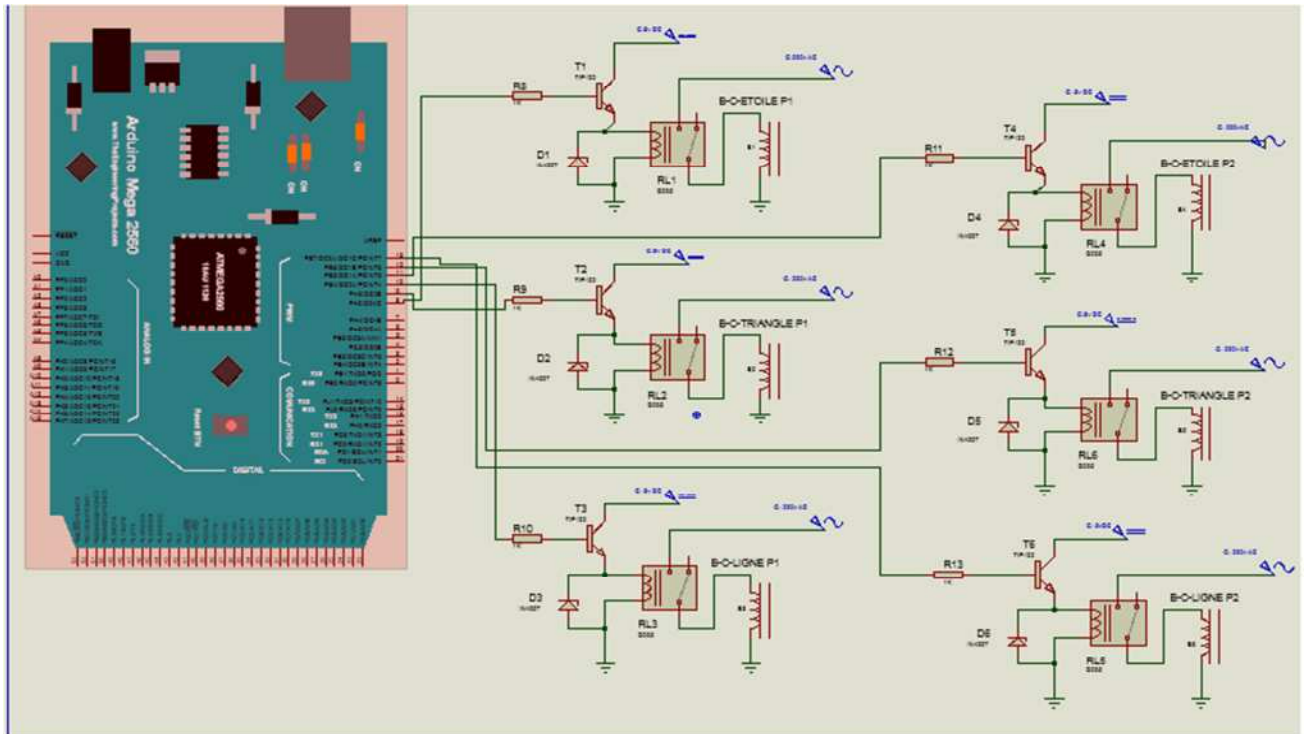


Figure (III-11) : Circuit électrique de module relais.

III.2.8. Les commutateurs On/Off

Lorsqu'on utilise une entrée numérique, il est important de s'assurer que le potentiel de l'entrée « Au repos » est bien celui auquel on s'attend. En effet, si on laisse l'entrée « libre », c'est-à dire câblée à rien, le potentiel qu'elle prendra ne sera pas nécessairement 0 V. (potentiel flottant) car l'électricité statique ambiante ou les perturbations électromagnétiques peuvent faire apparaître des valeurs très fluctuantes. Pour s'assurer du bon fonctionnement, l'on utilise une liaison protégée par une résistance qui va « tirer vers le haut » (5 V) ou « tirer vers le bas » (0 V) le potentiel au repos, comme une sorte d'élastique. On utilise en général une résistance de 10 kOhms. ^[33]

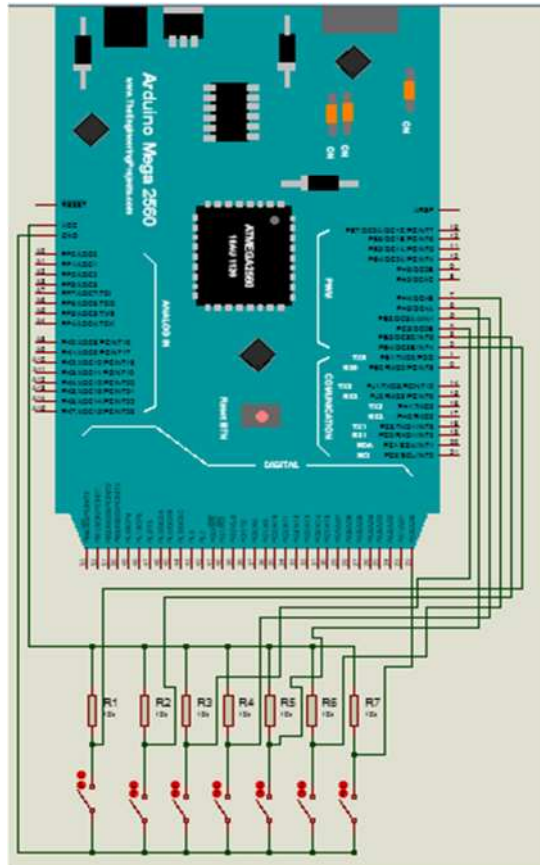


Figure (III-12) : Le circuit électrique des commutateurs.

III.2.7. LED : diode électroluminescente

Les LEDs sont polarisées et ont besoin d'une résistance pour limiter le courant.

Il n'est pas bon d'alimenter une LED directement en 5V (via une carte ARDUINO) car elle est en surtension : même si elle fonctionne elle brulera rapidement

Pour protéger la LED, il faut utiliser le principe du pont diviseur de tension, en la câblant en série avec une résistance, dont la valeur se calcule de la manière suivante :

$$R = (\text{tension d'alimentation} - \text{tension de seuil}) / \text{courant} \quad [33]$$

Couleur : Rouge.

Tension de seuil(Vf) : 1,6 V à 2 V 6 à 20 mA.

Longueur d'onde : 650 à 660 nm.

- $R = (5 - 1,6) / 0,02 = 170 \text{ Ohms}$

Pour une LED rouge : $R = 170 \text{ Ohms}$.

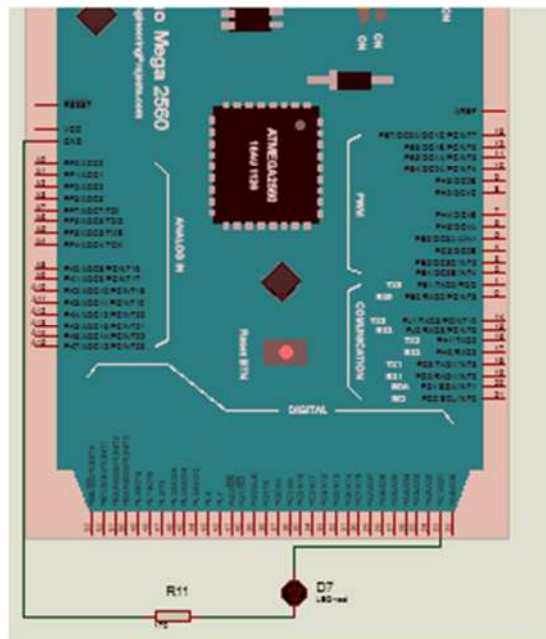


Figure (III-13) : LED de signalisation de défauts capteur.

III.2.11.Circuit électrique

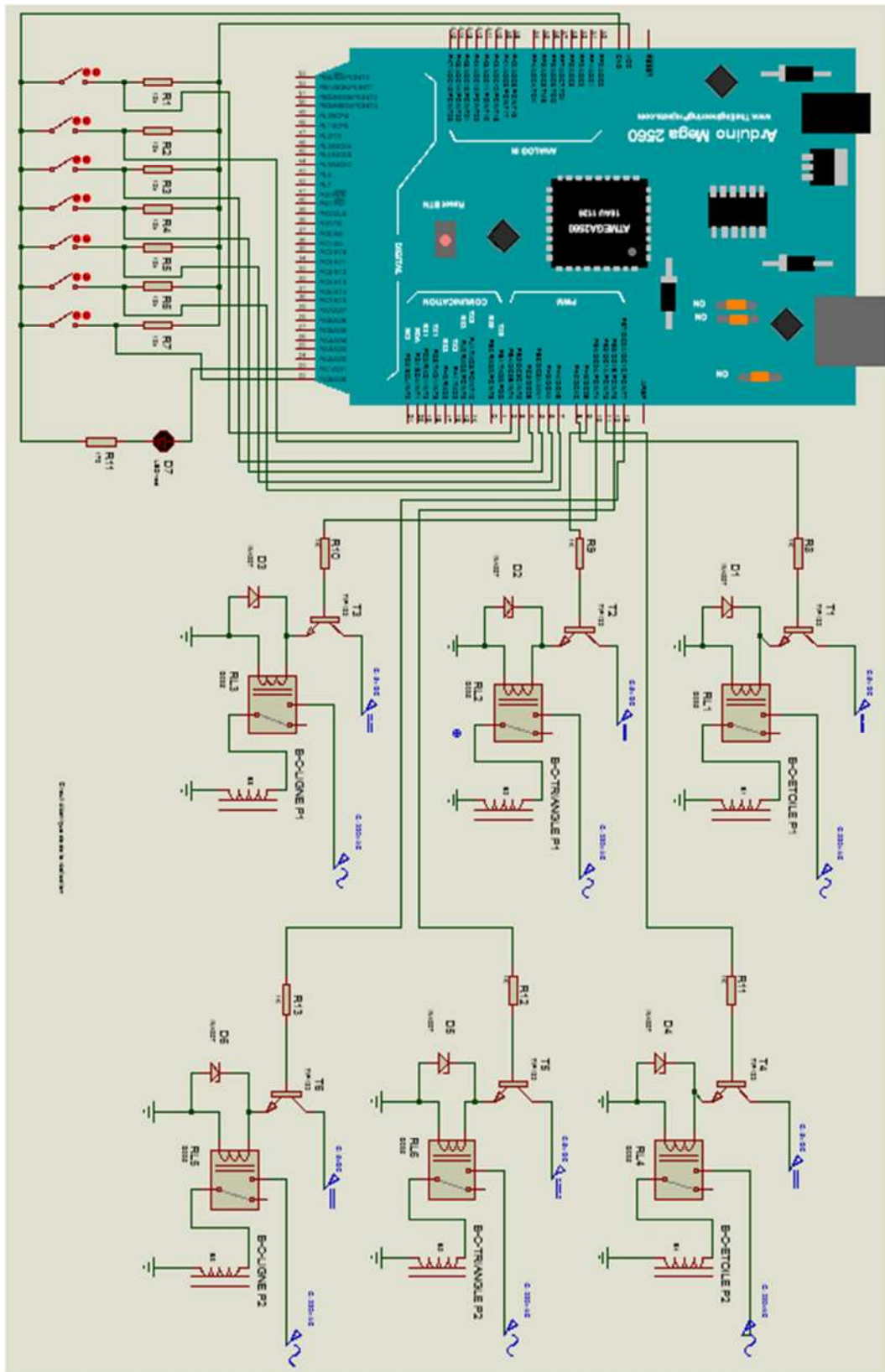


Figure (III-14) : Circuit électrique du système.

III.3. Réalisation

- **L'alimentation de la carte Arduino :**

Alimentation de la carte Arduino avec une pile 9V par le connecteur externe (jack).



Figure (III-15) : L'alimentation de la carte Arduino

- **Transistor TIP122**

La base du transistor est commandée par une sortie numérique d'Arduino. Il va commander la bobine du relais D005.

Le transistor est utilisé en interrupteur télécommandé, en mode saturé et en mode bloqué. Le mode saturé correspond à un interrupteur fermé, et le mode bloqué correspond à un interrupteur ouvert. Une résistance à la sortie d'Arduino en série avec la base du transistor pour limiter le courant : $R=1\text{Kohms}$.

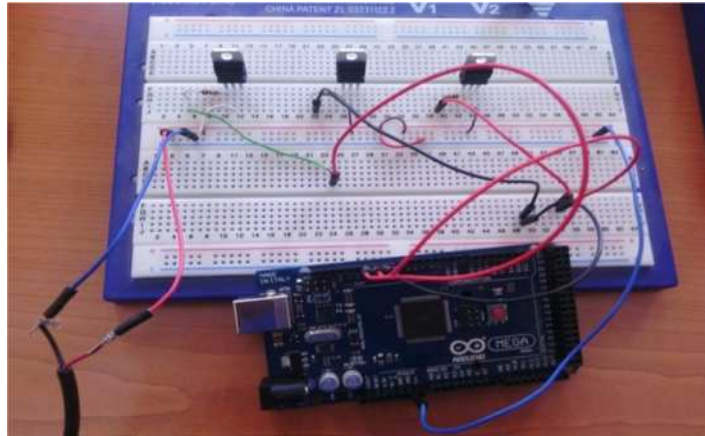


Figure (III-16) : Branchement des transistors à la carte Arduino.

- **Le relais D005**

Le modules relais D005 permet de relier les pompes à la carte Arduino afin d'être commandées.

Chaque relais est excité par l'émetteur du transistor TIP122. IL est utilisé é comme une interface pour exciter la bobine du contacteur avec 220V.

Une diode dite de roue libre 1N4007. Elle supprime le pic de tension aux bornes de la bobine du relais lors de l'ouverture du circuit, et par conséquent protège le transistor, et laisse passer le courant dans un seul sens.

La figure ci-dessous représente la photo de modules relais D005 avec la diode de roue libre 1N4007.



Figure (III-17) : Les modules des relais

- **Le contacteur NC1**

Est utilisé pour démarrer et protéger des moteurs asynchrones triphasés.

Le NC1, accouplé au relais thermique NR2, compose un démarreur électromagnétique, avec protection thermique du circuit contre les surcharges. Il est fiable, compact, léger, consomme peu et dure longtemps.

- **Les capteurs logiques ou TOR (Flotteur et Pressostat)**

Ce type de capteur renvoie une information de type logique c'est-à-dire 0 ou 1. En d'autres termes, la sortie peut prendre uniquement deux états. Comme mon cas les flotteurs présence ou absence d'eau, et le pressostat par rapport à la pression de seuil.

L'Arduino est relié à ce type de capteurs par les connexions les entrées/sorties numériques qui reçoivent ou envoient des signaux « 0 » ou « 1 » traduit par 0 ou 5V.

Dans ma réalisation je vais utiliser des commutateurs au lieu des capteurs.

La photo ci-dessous présente le branchement des commutateurs à la carte Arduino avec un test de défauts capteur.

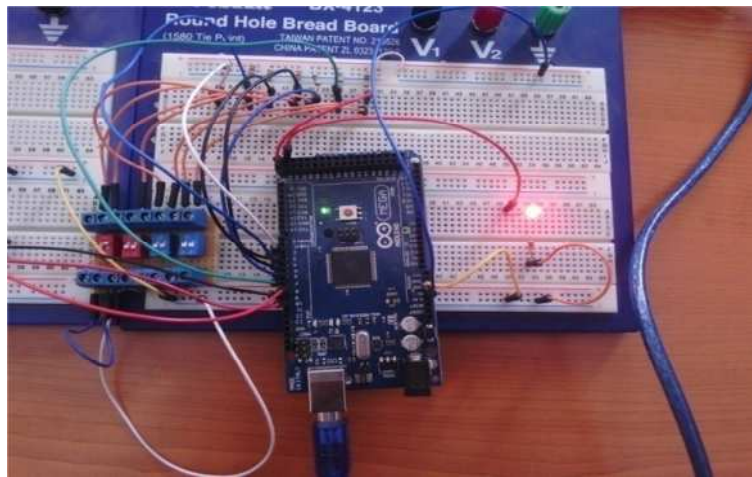


Figure (III-18) : Les commutateurs avec test de défaut capteur.

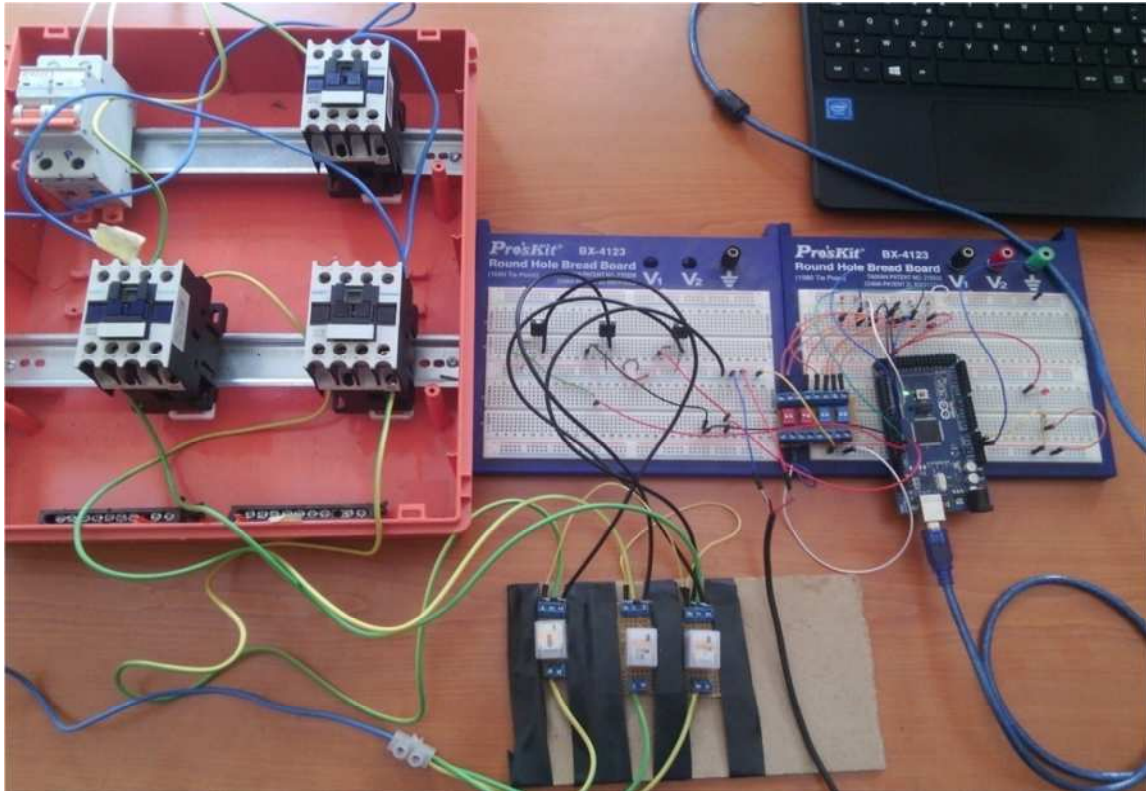


Figure (III-19) : Photo de la réalisation.

Dans ce chapitre, j'ai fait la réalisation pratique après la conception d'un système de remplissage d'un réservoir à eau.

Pour l'affectation des tâches, tout est assuré par la carte de commande, elle permet de contrôler le niveau dans le réservoir et de gérer le fonctionnement des deux pompes, aussi la détection de défauts capteur de niveau.

L'ensemble des modules qui permettant de faire les différent mesure sont reliés au broches du microcontrôleur Arduino Méga, qui le cœur de la carte de commande.

*Conclusion
Générale*

Conclusion générale

L'objectif de notre mémoire était de concevoir et réaliser un système de commande de remplissage d'un réservoir ou une fosse d'eau dans le cadre du projet fin d'étude.

La problématique de ce travail était de trouver des solutions à plusieurs posés. La gérance de fonctionnement des pompes, déduire le niveau d'eau dans les réservoirs et éviter le débordement de ces derniers, ainsi que de détection de défauts de capteurs.

Concevoir un système permettant l'amélioration de l'arrivée en eau dans les réservoirs était le point important de la réalisation. Pour cela on a utilisé une carte électronique programmable pour effectuer les différentes tâches du système afin d'apporter des solutions aux problèmes posés dans le cahier de charge.

Ce système permet d'exécuter des tâches industrielles avec une intervention humaine très réduite. A l'issue de notre travail, on peut conclure que :

- Facilité d'utilisation des cartes Arduino.
 - Facile à programmer.
 - Il existe plusieurs librairies pour les modules électroniques (capteur à flotteur, à pression, ...).
 - L'inconvénient des cartes c'est qu'elles ne supportent pas des tensions élevées.
-

Bibliographies

Bibliographie

Sites Internet:

- [1] Besson_S_miniaire_HEIG-VD_Topmoto.pdf-Adobe Acrobat Reader DC.
- [2] 92155_Brochure_Pompes.pdf-Adobe Acrobat Reader DC.
- [3] technologie des pompes.pdf-Adobe Acrobat Reader DC.
- [4] 2-Hydraulique-Industriel.pdf-Adobe Reader
- [5] <http://www.ardepi.fr>
- [6] Théorie des pompes.html.
- [7] [http://www .Cours d'Electrotechnique - Moteurs asynchrones triphasés Maxicours.com.html](http://www.Maxicours.com/html)
- [8] <http://www.schneider-electric.com>.
- [9] moteur_asynhorne.pdf-Adobe Acrobat Reader DC.
- [10] Moteur- asynchrone(2) .pdf-Adobe Acrobat Reader DC
- [11] <http://www.norme-standard.com/tag/installation-electrique-industrielle>
- [12] <http://Procédés de démarrage des moteurs asynchrones triphasés.htm>
- [14] [http : //Le moteur asynchrone triphasé ~ Automatisme Industriel ®.htm](http://Le moteur asynchrone triphasé ~ Automatisme Industriel ®.htm)
- [16] <http://www.leroy-somer.com>
- [18] Le moteur asynchrone triphasé-1.pdf-Adobe Acrobat Reader DC
- [20] arduino-mega2560-reference-design.zip, arduino-mega2560-schematic.pdf
- [21] premiers_pas_en_informatique_embarquée.pdf-Adobe Acrobat Reader DC
- [22] http://www.mon-clubelec.fr/pmwiki_reference_arduino/pmwiki.php.
- [23] Arduino_Mega.pdf-Adobe Acrobat Reader DC
- [24] Référence Arduino français Main_Materiel Mega 2560.html
- [33] Arduino .pdf-Adobe Acrobat Reader DC
- [26] Capteurs_et_pompes.pdf-Adobe Acrobat Reader DC
- [27] <http://www.Détecteur de niveau - RSM - WIKA France.htm>
- [28] Schneider - Catalogue contrôle industriel 99 .pdf-Adobe Acrobat Reader DC

Bibliographie

[29] capteur de pression/Capteur — Wikipédia.html.

[30] capteur de pression/Tout-electromenager.fr-Pièces détachées-Pressostat.html.

[31] <http://LOCODUINO> - Comment alimenter l'Arduino sans ordinateur.htm

[33] (Le moteur asynchrone triphasé) Alain .Charbonnel@ac-caen.fr Edition 2010.

Thèse :

-[13] Thèse : Détermination de la répartition des pertes de puissances dans un moteur asynchrone triphasé, département électronique, Université Mouloud Mammeri Tizi-Ouzou, Etudié par M^{elle} Zourdani Samia et M^{elle} Akmoussi Karima.

-[15] Thèse : Réalisation d'une station de caractérisation de capteurs de pression, département électronique, Université Mouloud Mammeri Tizi-Ouzou, Etudié par : Melle Bennour Yamina.

-[17] Thèse : INSTALLATIONS INDUSTRIELLES, Institut Supérieur des Etudes Technologiques de Nabeul Département : Génie Electrique, Préparé par : Adel SAID et Yassine JEMAI Technologues à l'I.S.E.T de Nabeul Année universitaire 2013 /2014.

-[19] Conception et réalisation d'un répondeur téléphonique programmable à base d'Arduino, département électronique, Université Mouloud Mammeri Tizi-Ouzou, Etudié par Nor Belarbi et Djamel Douaidi.

[32] Conception et réalisation d'un système autonome de distribution d'eau potable département électronique, Université Mouloud Mammeri Tizi-Ouzou, Etudié par Messaoui Oualid, Mezaguer Karima.

Ouvrage :

[25]-Simon Landrault (Eskimon), Hippolyte Weisslinger(Olyte), Arduino : Premiers pas en informatique embarquée, le blog d'Eskimo, Edition, juin 2014.

-Physique Appliquée, terminale électrotechnique - collection R. Mérat et R. Moreau – édition Nathan technique 1994.

I- L'environnement Matlab/Simulink

C'est un logiciel de calcul mathématique pour les ingénieurs et les scientifiques créé par Mathworks.

I-1 : MATLAB :

Est un environnement de programmation pour le développement d'algorithme, d'analyse de données, de visualisation, et de calcul numérique. En utilisant Matlab, la résolution des problèmes de calcul complexes se fait plus rapidement qu'avec des langages de programmation traditionnels, tels que C, C++, et le Fortran.

I-2 : SIMULINK :

Est un environnement pour la simulation multi-domaine. Il fournit un environnement graphique interactif et un ensemble de bibliothèques de bloc qui permettent de concevoir, simuler, mettre en application, et examiner une variété de systèmes, tel que les systèmes de communications, de commandes, de traitement des signaux, de traitement visuel, et de traitement d'image.

I-3: L'interfaçage Arduino Matlab/Simulink

Il existe trois possibilités d'interfacer la carte Arduino avec Matlab/Simulink, à savoir :

1. Programmation de la carte Arduino MEGA comme une carte d'interface.
2. Utilisation du package Arduino I/O.
3. Utilisation du package Arduino Target.

➤ Arduino I/O

La bibliothèque Arduino I/O, permet d'utiliser la carte Arduino comme une interface d'entrée/sorties, le principe est de charger un programme dans la carte pour qu'elle fonctionne comme serveur, écoutant les requêtes envoyées via la liaison série (USB) et de répondre à ces requêtes en envoyant l'état d'une entrée ou en modifiant l'état d'une sortie. Ces mêmes entrées/sorties sont vues dans Matlab comme des entrées logiques ou analogiques (utilisant du CAN) ou des sorties analogique en mode PWM.

➤ Installation du support package Arduino I/O :

Matlab Simulink ne viennent pas avec des paquets préinstallés pour arduino.il nous suffit de télécharger depuis le site officiel de Mathworks .pour 2013 et versions supérieures.

- 1- Exécuté le programme MATLAB.
- 2- Dans la barre de menu allez dans le menu HOME. ('1' dans la figure 1)
- 3- Il est bloc pour RESSOURCES ayant une option d'ad-ons('2'dans la figure 1)

Annexe

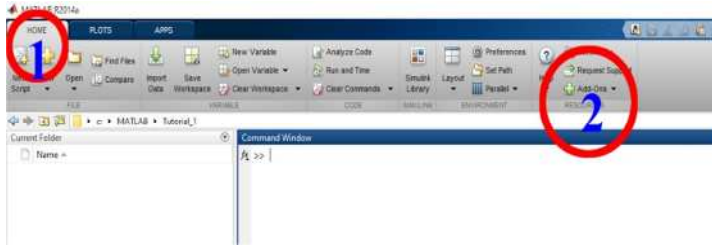
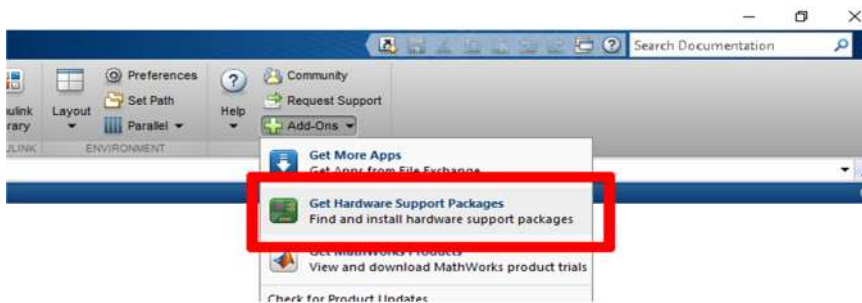


Figure (1) : barre de menu.

- 4- Cliquer sur les paquets de Support matériel get.



Figure(2) : Support Get.

- 5- Installer à partir d'internet pour installer le package matériel à partir d'internet sur votre ordinateur

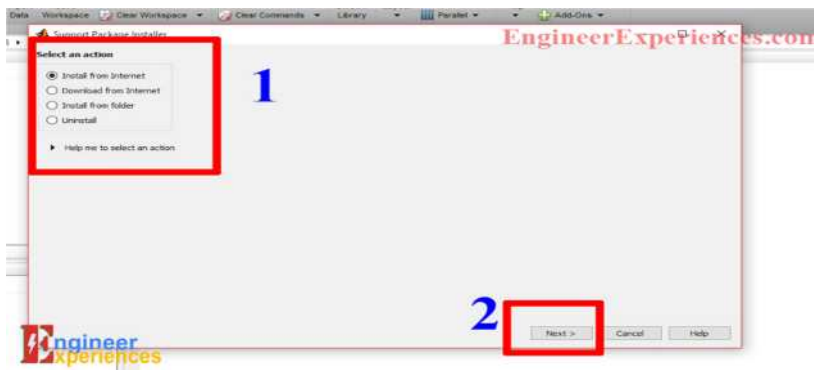


Figure (3) : support package d'installation

- 6- Cliquant sur Suivant dans la fenêtre d'installation de Support Package à droite (Figure 3-2)

La fenêtre suivante va ouvrir montrant tous les paquets de support matériel disponible (figure 4)

- 7- cliquant sur Arduino et Suivant (figure 4-1 et 4-3).

Annexe

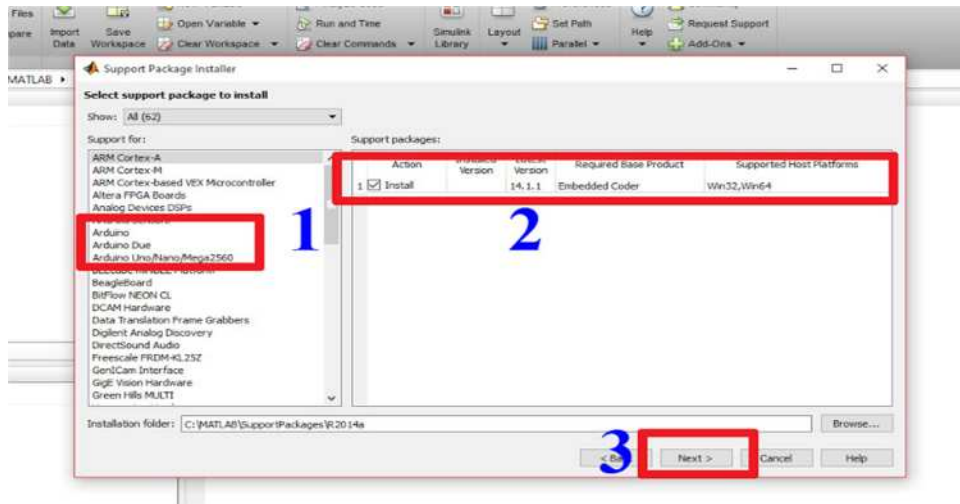


Figure (4) : Sélectionner Support Package pour installer.

- 8- Nous devons connecter au compte mathworks comme la montre la figure 5 à la figure.
- 9- Après la connexion par notre adresse e-mail et mot de passe cliquer sur connexion et cliquez sur suivant et à côté et il faudra peu de temps (selon la connexion internet) pour télécharger et installer. Puis cliquez sur Terminer.

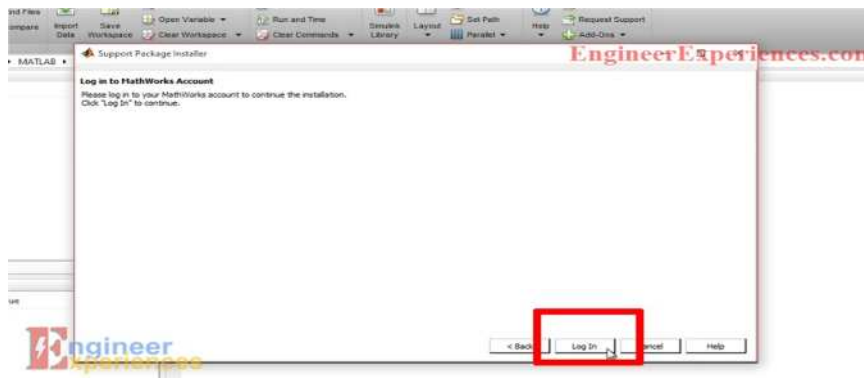


Figure (5) : connexion au compte mathworks

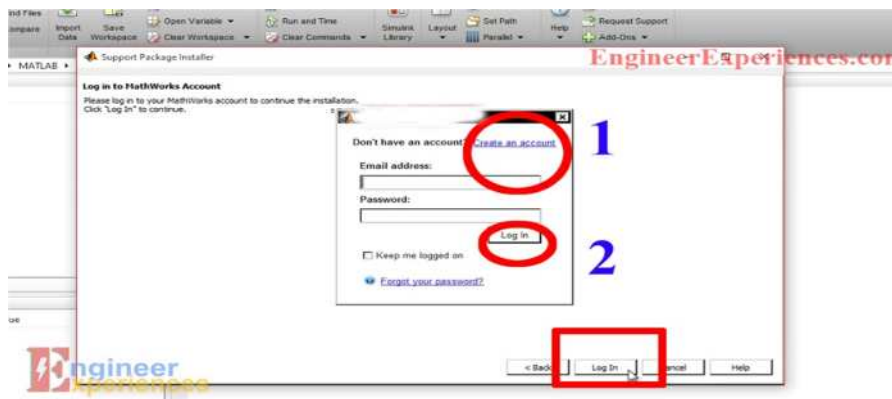
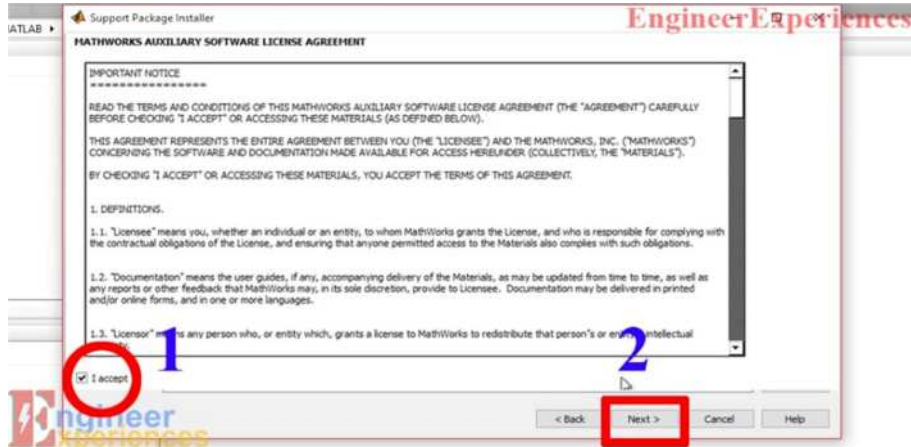


Figure (6) : Connexion à compte mathworks.

Annexe



Figure(7) : Accepter les termes et conditions.

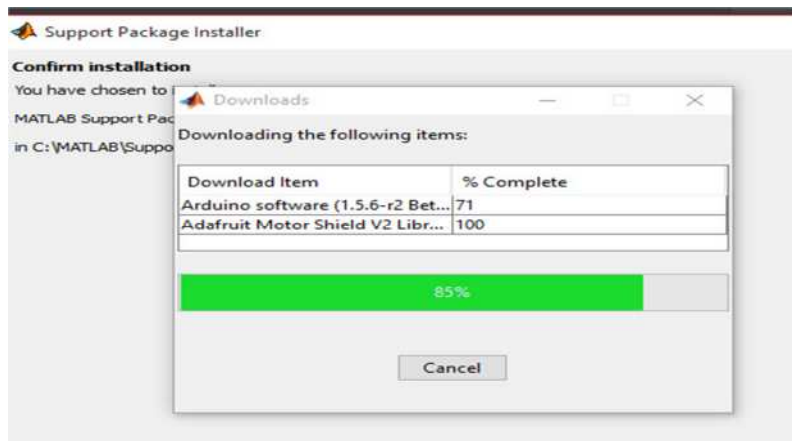


Figure (8) : Télécharger et installation en cours

Après le succès de l'installation du Support Package, les bibliothèques Arduino dans MATLAB Simulink seront ajoutées.



Figure (9) : Support package dans Simulink matlab.

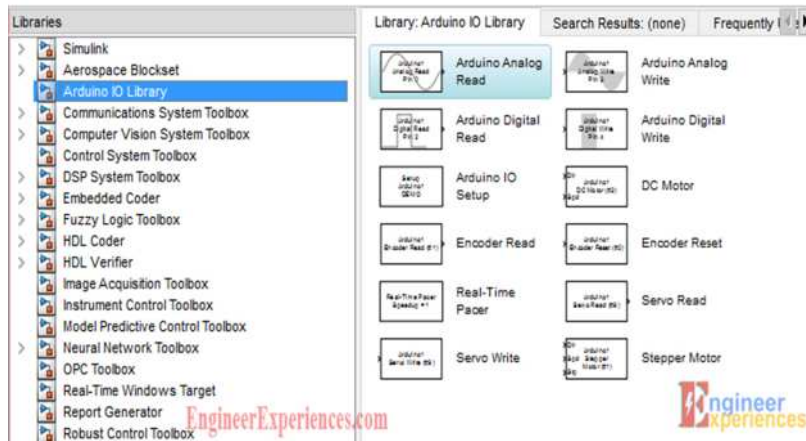


Figure (10) : Arduino I/O library

Cette bibliothèque présentée dans la figure 10 contient 12 autres blocs / opérations que vous pouvez faire avec Simulink et Arduino connecté. Cette bibliothèque permet d'envoyer les informations du Matlab au Conseil Arduino connecté ainsi que de recevoir du conseil d'administration.

II- Stateflow :

Stateflow est un module développé par la société américaine MathWorks qui permet la simulation de machines d'état. Une machine d'état comporte un nombre fini d'états. Elle modélise le comportement de systèmes qui passent d'un état à un autre en réponse à des événements. On parle alors de systèmes à événements discrets.

Comme son nom l'indique, ce module permet de tracer des diagrammes d'état (State Chart) et des diagrammes de flux (Flow Chart).

Stateflow est intégré à Matlab et Simulink. Les modèles construits pourront par conséquent comporter des blocs des différentes « toolboxes » de Simulink et/ou appeler des fonctions Matlab et/ou des fonctions Simulink.

Ainsi le modèle global d'un système complexe pourra comporter des modèles linéaires continus construits avec Simulink sous la forme de schéma-blocs, des machines à état construites avec Stateflow ou encore des modèles acausaux réalisés en utilisant Simscape.

Aussi Stateflow permet de simuler le comportement de systèmes hybrides c'est-à-dire à événements discrets et continus.

Stateflow est un environnement de modélisation et de simulation de logique de décision combinatoire et séquentielle à partir de machines d'état et de diagrammes de flux. Stateflow vous permet de combiner des représentations graphiques et tabulaires, dont des diagrammes de transition d'états, des diagrammes de flux, des tables de transition d'états et des tables de vérité, pour modéliser la manière dont votre système réagit à des événements, des conditions temporelles et des signaux d'entrée externes.

Stateflow permet de concevoir la logique de commandes de surveillance, de planification des tâches et des applications de gestion d'incidents. Stateflow inclut une animation des

Annexe

diagrammes d'états ainsi que des contrôles statiques et d'exécution pour tester la cohérence et l'exhaustivité de la conception avant sa mise en œuvre.

Le programme du fonctionnement du système cité dans le mémoire sur stateflow :

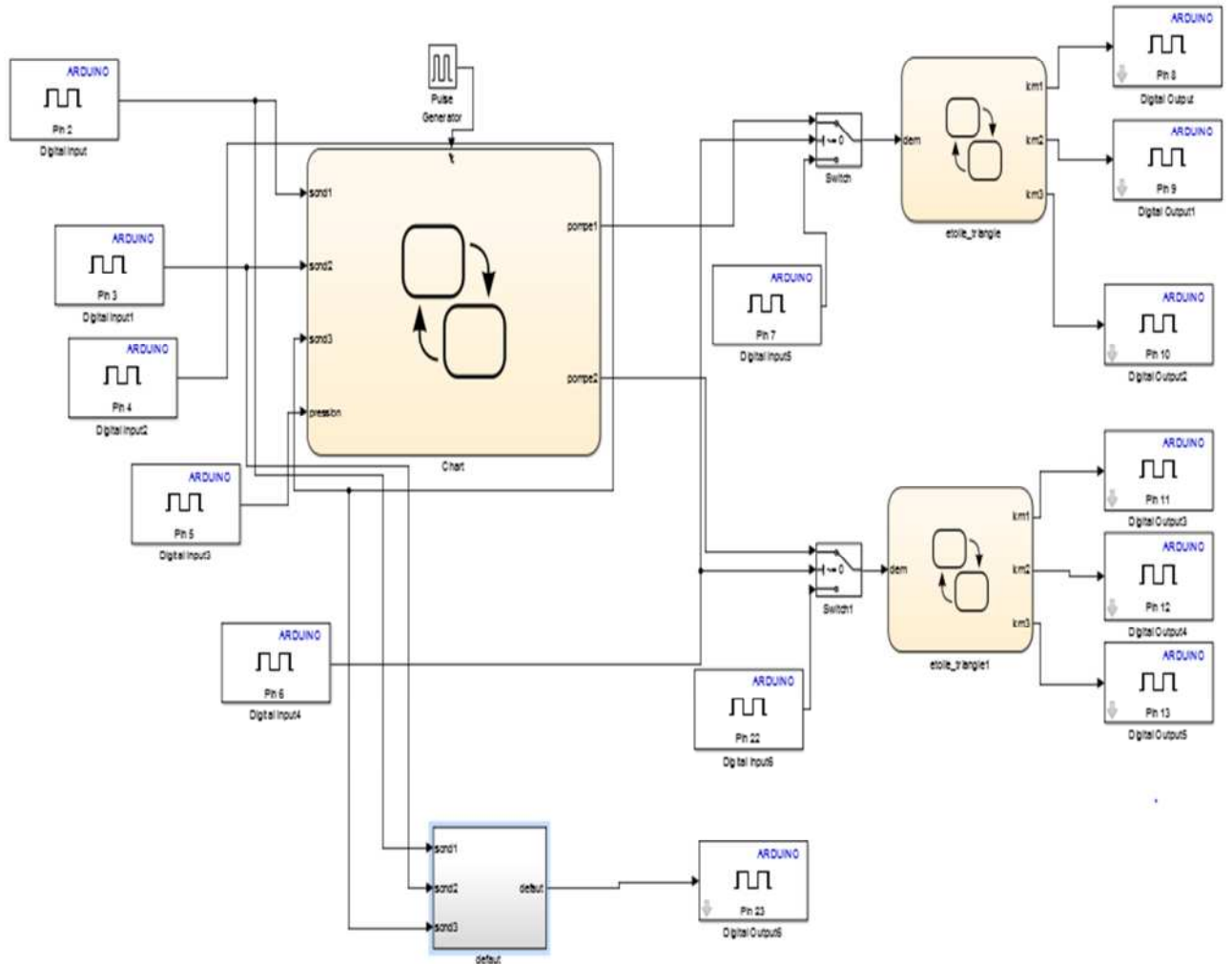


Figure (11) : branchement de la carte Arduino et les différents composants.

Annexe

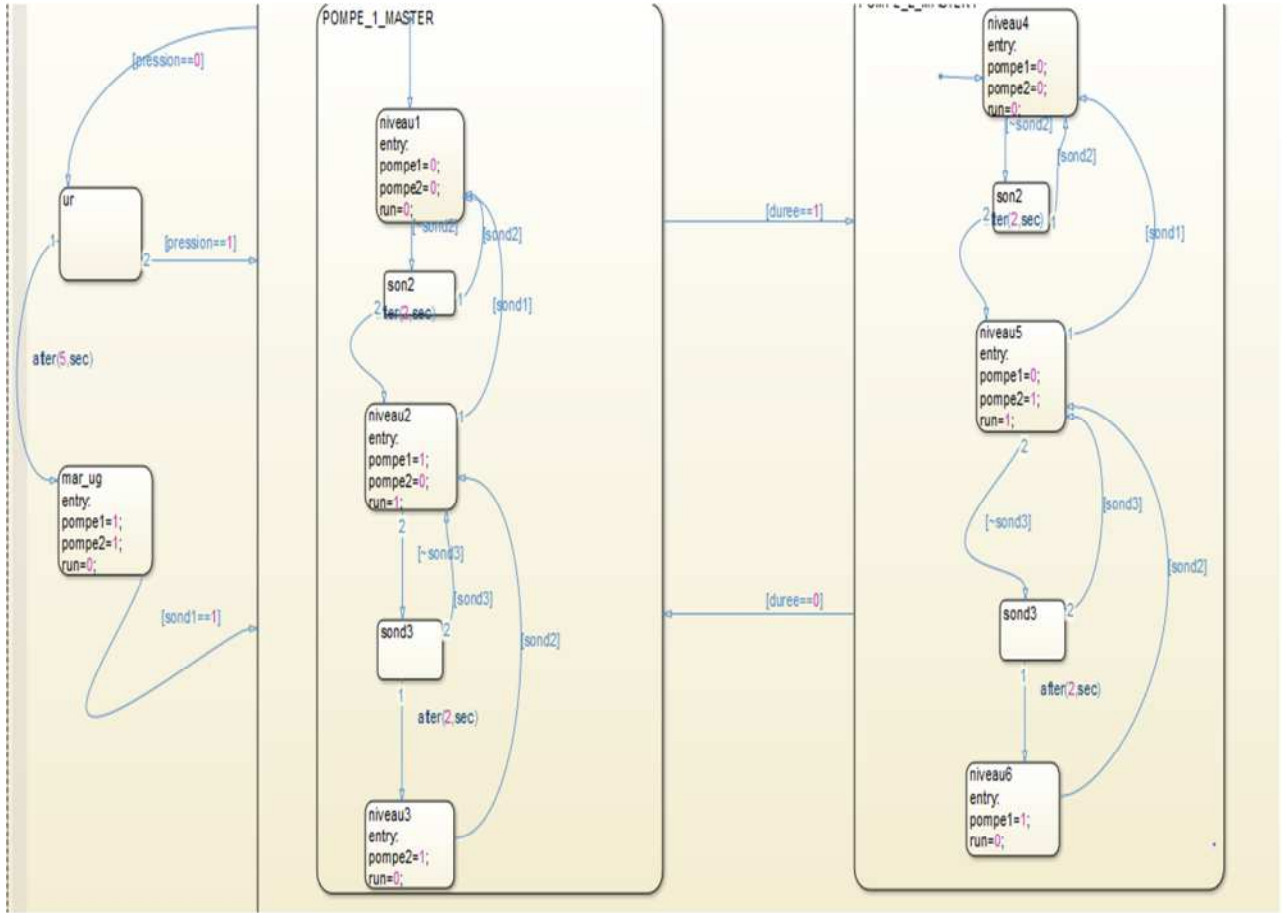


Figure (12) : L'enchaînement des étapes de fonctionnement en fonction des capteurs.

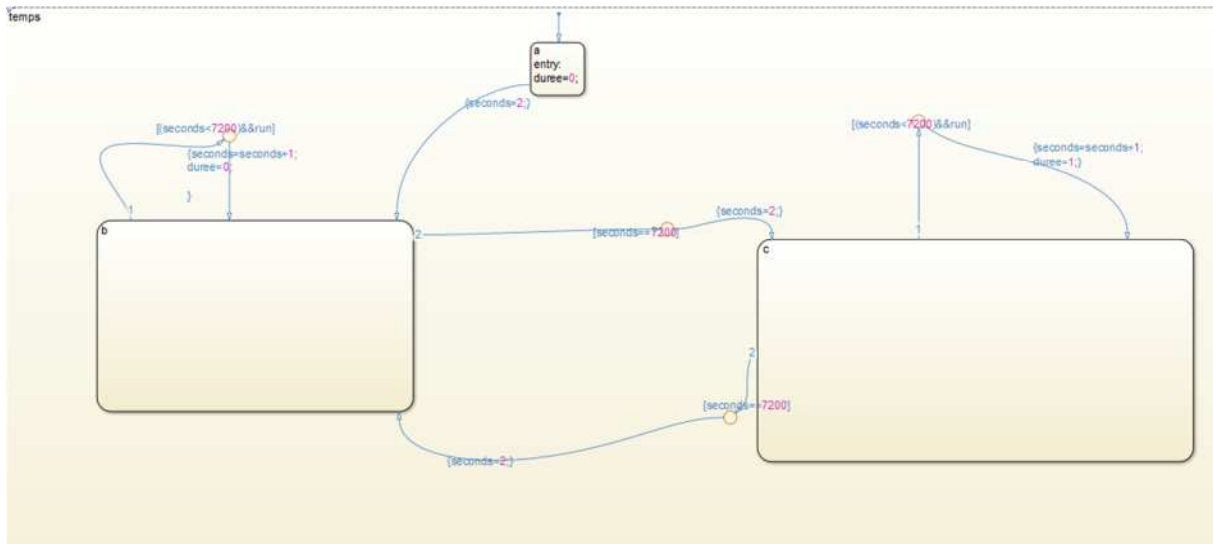
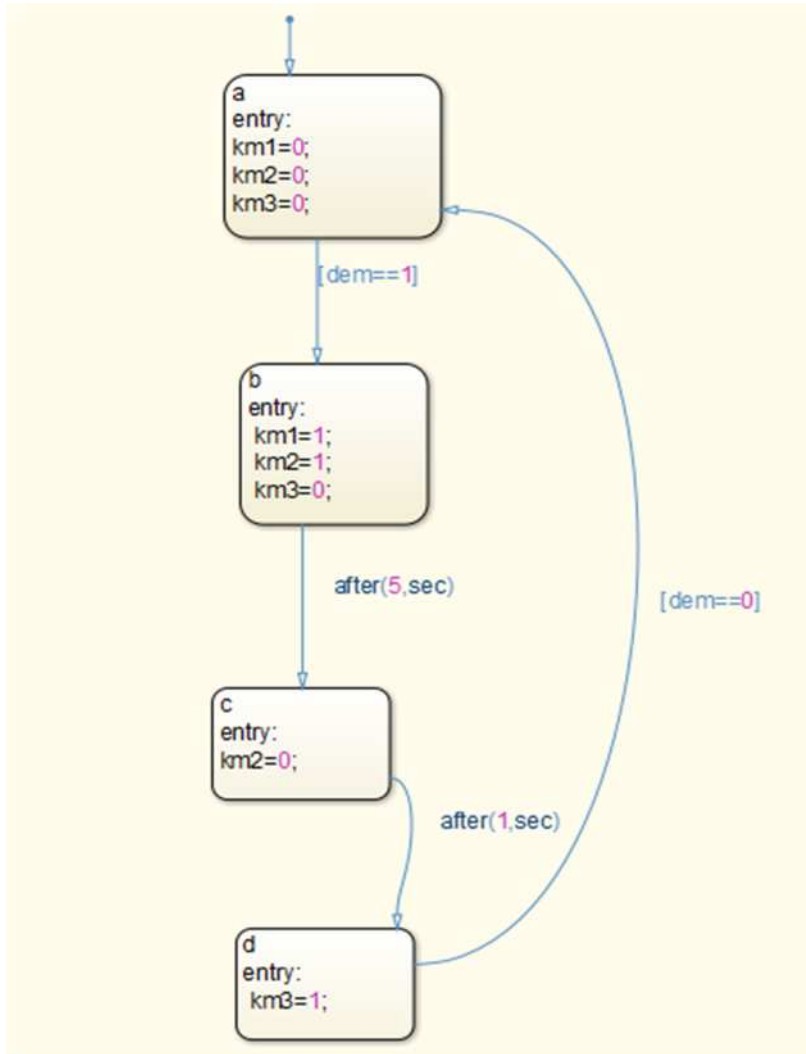
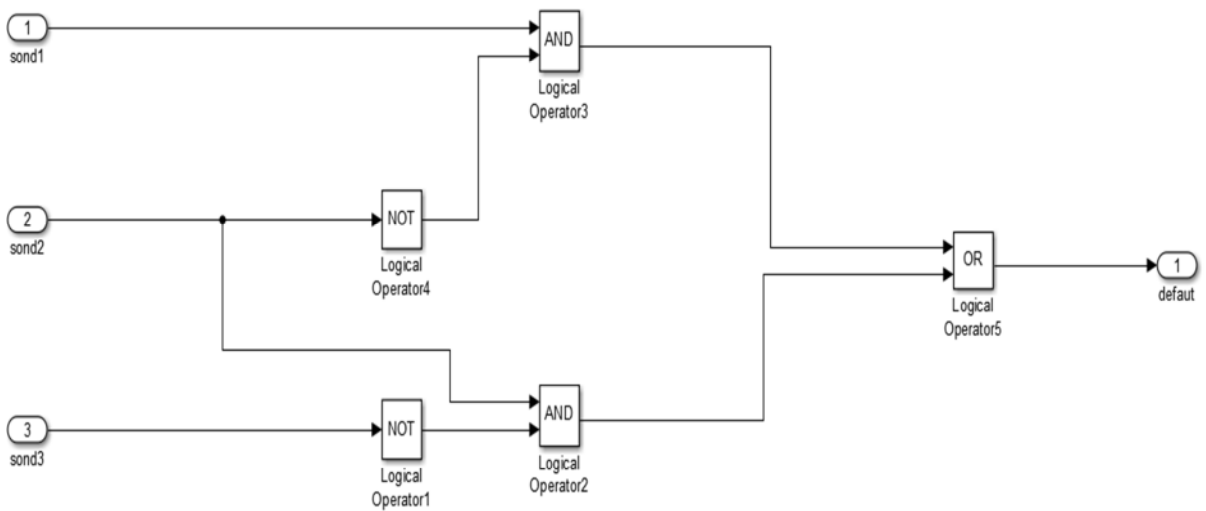


Figure (13) : Le temps de fonctionnement de chaque pompe.



Figure(14) : Démarrage étoile-triangle.



Figure(15) : L'logigramme de défauts capteur.

Matériels utilisés :

➤ Relais:

Un relais électromagnétique, dans son principe de commutation, s'apparente à un interrupteur mécanique dont la manœuvre serait non pas effectuée manuellement, mais en faisant circuler un courant dans le circuit d'excitation du relais. Ce circuit est constitué par une bobine appelée bobine d'excitation ou bobine de commande.

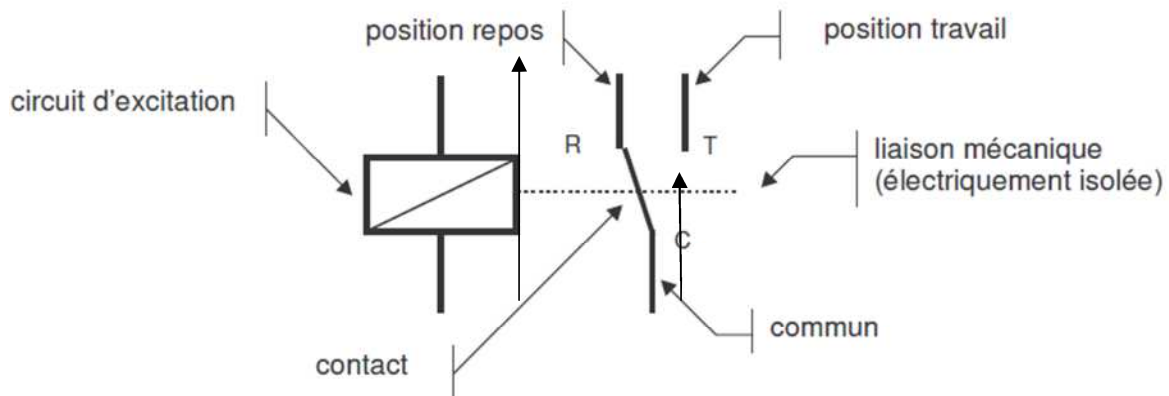


Figure : Constitution d'un relais électromagnétique.

La mise en action du relais par sa bobine de commande demande une puissance électrique faible. Elle provoque l'ouverture ou la fermeture d'un circuit dans lequel circule un courant élevé, et par là même elle peut commander une puissance beaucoup plus grande que la puissance d'excitation. Les contacts d'un relais est toujours représentée en position de repos. Lorsque le relais est commandé, le contact initialement en position repos passe en position travail et reste dans cette position tant qu'un courant circule dans le circuit d'excitation. Lorsque le courant dans le circuit d'excitation disparaît, le contact revient en position repos.

D005 : ^[34]



Figure : D005

Caractéristiques techniques :

Alimentation	5 V DC
Courant :	84mA.
Nombre de broches :	5
Charge nominale :	12A/400V
Résistance de bobine	60 Ω
Puissance de bobine	420mW

Annexe

➤ Tip 122 :

Un montage Darlington est constitué de deux transistors bipolaires. Ces derniers peuvent être intégrés dans un même boîtier

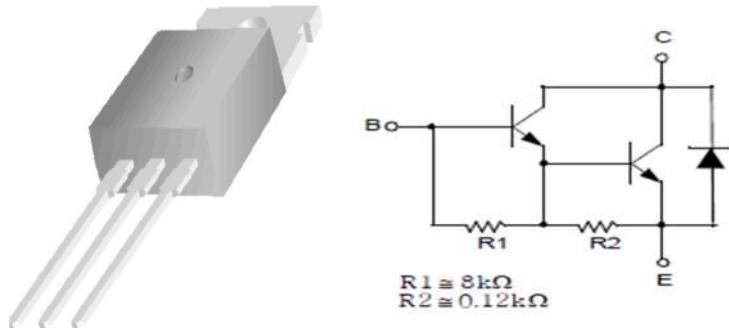


Figure : Tip 122

Tension Collecteur-Base V_{CBO} = 100V

Tension Collecteur-Emetteur V_{CEO} = 100V

Tension Emetteur-Base V_{EBO} = 5V

Courant Collecteur (DC) I_c = 5A

Courant Base (DC) I_b = 120mA

Gain h_{FE} = 1000

$V_{ce(sat)}$ Collecteur-Emetteur Saturation Voltage 2.0V

$V_{be(on)}$ * Base-Emetteur On Voltage 2.5 V

➤ Contacteur :



Figure : contacteur NC1-32

Le contacteur NC1 est utilisé pour démarrer et protéger des moteurs asynchrones triphasés en classe d'application AC-3, AC-4, démarrages fréquents. Le NC1, accouplé au relais thermique NR2, compose un démarreur électromagnétique, avec protection thermique du circuit contre les surcharges. Il est fiable, compact, léger, consomme peu et dure longtemps.

Caractéristiques techniques :

- Tension max : 690 V
- Courant max : 95 A

Annexe

- Tension nominale U_e : (Vac) 230 V ; 400 V ; 690 V
 - Fréquence nominale 50-60 Hz
 - Courant max. 95 A
 - Classe d'application AC-3.
 - Courant nominale (A) 32A
 - Puissance moteur triphasé contrôlé
- | | |
|--------------|---------|
| 220, 230 V : | 7.5Kw |
| 380, 400 V : | 15Kw |
| 660, 690 V : | 18.5 Kw |
- Tension de la bobine : 220V
 - Résistance de la bobine : 290Ohms
 - 1 contact auxiliaire normalement ouvert(NO).

Le pressostat :

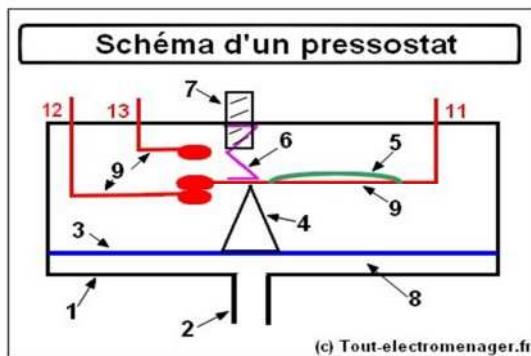


Figure : Pressostat

Constitution :

1-Corps (boîtier plastique). 2-Tétine de raccord à la chambre de compression. 3-Membrane caoutchouc flexible. 4-Pointeau. 5-Système de retour (contre force) du contact. 6-Ressort, 7-Vis de réglage : elle permet de faire varier la tension du ressort influent ainsi sur le déclenchement du contact entre 11-12 et 11-13, et donc par conséquent une modification du niveau d'eau dans l'appareil, 8-Chambre à air étanche, 9-Lamelles en cuivre. 11, 12, et 13-Bornes du pressostat.

Le branchement du capteur de niveau a la carte Arduino :

Le montage consiste à relier l'alimentation (5V et GND) et une lecture logique de l'Arduino au capteur.

5V (Arduino) : pin + (Capteur)

GND (Arduino) : pin – (Capteur)

Annexe

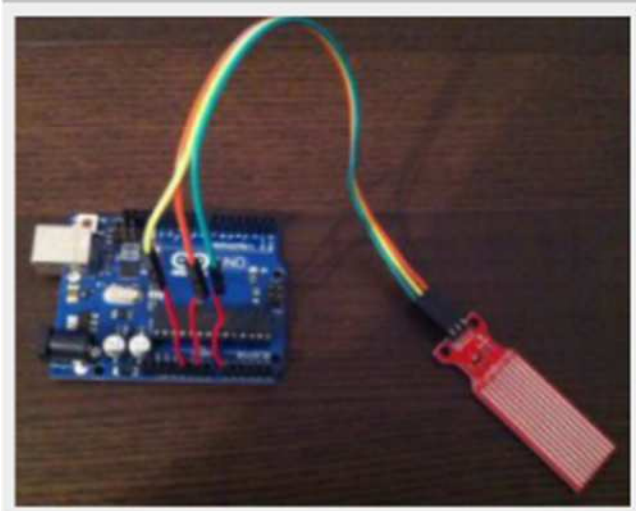


Figure : Branchement du capteur de niveau.

Figure : Schéma électrique de la carte Arduino Mega2560