



## *Dédicace*

*Je dédie ce modeste travail à, mon symbole de la tendresse, qui s'est sacrifiée pour mon bonheur et ma réussite à mère.*

*A mon père l'école de mon enfance, qui a tété mon ombre durant toutes les années d'études, et qui a viellé tout au long de ma vie à m'encourage à m'aider et me protéger.*

*A mes adorables frères et ma chère sœur : Pour toute la complicité et l'entente qui nous unissent, je ne pourrais jamais exprimer le respect et l'amour que j'ai pour vous. Vos prières, vos encouragements et votre soutien m'ont toujours été d'un grand secours.*

*À mes chère amis : Autant de phrases et d'expressions aussi éloquentes soit-elles ne sauraient exprimer ma gratitude et ma reconnaissance. Merci.*





## *Remerciements*

*Nous tient à remercier tout premièrement Allah le tout puissant pour la volonté, la santé et la patience, qu'il nous a donné durant toutes ces longues années. Nous tient également à exprimer nos vifs remerciements à notre encadreur Mr. DOUGHMANE HAKIM pour avoir d'abord proposé ce sujet et de son suivi continu tout le long de la réalisation de ce mémoire et qui n'a pas cessé de nous donné ses conseils.*

*Enfin, nous tient à remercier vivement tous les enseignants du département d'électronique et de télécommunications.*



# Sommaire

<b>Introduction générale</b> .....	1
<b>Chapitre 1 généralité sur les ascenseurs</b>	
1.1 Introduction.....	2
1.2 Définition.....	2
1.3 Les différents types d'ascenseur.....	2
1.3.1 Les ascenseurs hydrauliques .....	2
1.3.1.1 Description.....	2
1.3.1.2 Les différentes parties d'un ascenseur hydraulique.....	3
1.3.1.3 Les principaux avantages et inconvénients.....	4
1.3.2 Ascenseurs à traction Câbles.....	5
1.3.2.1 Description.....	5
1.3.2.2 Les différents avantages et inconvénients.....	5
1.4. Fonctionnement.....	6
1.4.1 Constitution d'un ascenseur.....	6
1.4.1.1 La cabine.....	7
1.4.1.2 Gaine.....	8
1.4.1.3 Le contre poids.....	9
1.4.1.4 Les guides.....	10
1.4.1.5 Le moteur.....	10
1.4.1.6 Un système de traction.....	11
1.5. Domaine d'utilisation d'ascenseur.....	11
1.5.1 Domaine urbaine .....	11
1.5.2 Domaine médicale.....	12
1.6. Exemple d'ascenseur a traction.....	12
1.6.1 Ascenseur MRL (Machine Room-less).....	12
1.7. Quelque technologie sur ascenseur.....	13
1.7.1 La charge nominale.....	13
1.7.2 Le frein parachute.....	14
7.3 Le contrôleur de manœuvre ou armoire de commande.....	14
7.4 Boitier d'inspection.....	15
1.8 Conclusion.....	16

## Chapitre 2 arduino

2.1 Introduction .....	17
2.2 Microcontrôleur.....	17
2.2.1 Les types de microcontrôleur les plus connue courant .....	17
2.3 Définition d'un Arduino .....	19
2.3.1 Carte arduino Mega 2560.....	19
2.3.1.1 Le but et l'utilité .....	20
2.3.1.2 Caractéristiques .....	21
2.3.1.3 Entrées et sorties numériques.....	22
2.3.1.4 Broches analogiques.....	23
2.3.1.5 Autres broches.....	23
2.3.1.6 Communications.....	23
2.3.1.7 Applications .....	25
2.3.1.8 Alimentation .....	25
2.3.1.8 Avantages et Inconvénients .....	26
2.4 Les différents accessoires utilisés.....	26
2.4.1 Moteur pas à pas .....	26
2.4.1.1 Différents types des moteurs.....	27
2.4.1.2 Principe de fonctionnement .....	27
2.4.2 Capteur de mouvement infrarouge.....	28
2.4.2.1 Définition.....	29
2.4.2.2 Quelque avantage et inconvénient .....	29
2.4.3 Les afficheurs à cristaux liquides .....	30
2.4.3.1 Principe des cristaux liquides.....	31
2.4.3.2 Brochage.....	31
2.4.3.3 Les modes de fonctionnement de l'afficheur .....	32
2.4.3.3.1 Mode de 8 bits .....	32
2.4.3.3.2 Mode de 4 bits .....	33
2.4.4 Le logiciel PROTEUS.....	33
2.5 Conclusion. ....	35

## Chapitre 3 ascenseurs à 4 étages

3.1 Introduction.....	36
3.2 Partie 1: Cotés matérielles.....	37
3.2.1 Structure et principe de fonctionnement.....	37
3.2.2 Structure des composants de système.....	37
3.2.3 Cahier de charge.....	37
3.2.4 Description des composants du circuit.....	38
3.2.4.1 Moteur pas à pas.....	38
3.2.4.1.1 Principe de fonctionnement.....	38
3.2.4.1.2 Domaine d'application.....	39
3.2.4.2 Capteur mouvement infrarouge.....	39
3.2.4.2.1 Domaines d'application.....	40
3.2.4.3Ecran LCD.....	40
3.2.4.3.1 Rôle d'un afficheur LCD.....	40
3.2.4.4 Description d'un clavier 4x4.....	41
3.3 Partie 2: logicielles.....	42
3.3.1 Structure d'un programme arduino.....	43
3.3.2 Logiciel Isis PROTEUS.....	45
3.3.3 Schéma électrique.....	45
3.3.4 Les organigrammes.....	46
3.3.5 Organigramme principal de fonctionnement.....	46
3.5 Conclusion.....	48
<b>Conclusion générale.....</b>	<b>49</b>
Bibliographie.....	50
Annexe.....	52

## Liste des figures

Figure (1.1) : schéma cylindrique.....	3
Figure (1.2): fonctionnement d'un ascenseur hydraulique.....	3
Figure (1.3): type d'ascenseur hydraulique.....	4
Figure (1.4): type d'ascenseur a traction câble.....	5
Figure (1.5) : Schéma explicatif.....	7
Figure (1.6) : Cabine.....	8
Figure (1.7) : Gaine ou trémie.....	9
Figure (1.8) : Contre poids.....	9
Figure (1.9) : Guide d'ascenseur.....	10
Figure (1.10) : Moteur d'ascenseur.....	10
Figure (1.11):Machinerie.....	11
Figure (1.12) : Ascenseur dans un bâtiment.....	11
Figure (1.13) : Ascenseur pour handicapé.....	12
Figure. 1.14 : As censeur moteur électrique à attaque directe (type MRL).....	13
Figure (1.15) : plaque signalétique.....	13
Figure (1.16) : Schéma système de parachute.....	14
Figure (1.17) : armoire de commande.....	15
Figure (1.18) : Boitier d'inspection.....	15
Figure (2.1) : Schématique d'un arduino mega2560.....	18
Figure (2.2) : schémas électroniques et typons.....	19
Figure (2.3) :Arduino mega 2560.....	20
Figure (2.4) : les model arduino.....	21
Figure (2.5) : Schéma interne de l'ATmega 2560.....	24
Figure Organigramme (2.6) Structure générale d'un projet arduino.....	25
Figure (2.7) : phase du stator.....	28
Figure (2.8) : le spectre lumineux de l'ultraviolet à l'infrarouge.....	29
Figure (2.9): Afficheur LCD.....	30
Figure (2.10) : simulateur isis protues.....	34
Figure. (3.1) : maquette d'ascenseur a 4 étages.....	36
Figure (3. 2) : Schéma synoptique de l'ascenseur.....	37
Figure (3. 3) : Pilotage avec une carte Arduino d'un moteur pas à pas unipolaire.....	38
Figure (3. 4) : Moteur pas a pas 28bj.....	39
Figure (3. 5) : Capteur Capteur mouvement infrarouge .....	40
Figure (3. 6) : Schéma fonctionnel d'un LCD.....	41

Figure (3.7) : clavier 4x4.....	41
Figure (3. 8) : Compileur arduino.....	42
Figure (3. 9) structure programme.....	43
Figure (3. 10) : partie déclaration des variables.....	44
Figure (3. 11) : Carte de simulation de l'ascenseur.....	45
figure (3. 12) : de détection de la position.....	46
figure (3. 13) : principal de fonctionnement.....	48

## Liste des tableaux

Tableau (1.1) : Présentation des caractéristiques de la carte Arduino Méga 2560.....	21
Tableau (1.2): Différentes broches entrées/sorties numériques.....	21
Tableau (1.3): description de différentes broches de l'afficheur LCD.....	31

# Introduction générale

---

## Introduction générale

Les ascenseurs modernes sont le résultat de l'évolution technologique de trois domaines techniques l'électronique, la mécanique et l'informatique. Leur objectif est d'aider l'être humain afin de faciliter ses tâches quotidiennes:

- La partie électronique : les composants hardwares représentent Les actionneurs, les capteurs et la partie commande (microcontrôleur).
- La partie mécanique : représente la forme globale de l'ascenseur et les différents composants mécaniques (poulet, ...etc.).
- La partie informatique : réside dans tous les instructions permettant la gestion de l'ascenseur et les différents traitements d'informations.

Dans le cadre de ce projet de fin d'études, nous avons choisi de nous pencher sur le sujet de «commande numérique d'ascenseur par carte arduino». Ce projet consiste à étudier et élaborer une maquette d'un ascenseur à quatre étages.

Ce manuscrit est organisé en trois chapitres :

Le premier chapitre présente une vue générale sur l'ascenseur par la description de son différentes catégories et des ses modules qu'ils composent.

Le second chapitre aborde les notions fondamentales du microcontrôleur ainsi que la carte Arduino Mega 2560 qui sera présentée avec plus de détails.

Le troisième chapitre est consacré à la partie réalisation de ce projet.

Enfin, une conclusion générale clôture ce travail.

## **1.1 Introduction**

Pendant ces dernières années, la création des nouvelles villes modernes basées essentiellement sur la construction des bâtiments (appartements) avec des étages plus au moins élevés cela rend la présence des ascenseurs une chose indispensable afin de mettre l'habitat dans un confort.

Dans ce chapitre, on commence par présenter une bref définition sur l'utilisation de l'ascenseur et de ses étapes d'évolution. Puis, on décrit les différents types d'ascenseur que l'on peut trouver.

## **1.2 Définition**

D'après la rousse «Appareil élévateur permettant de transporter des personnes dans une cabine se déplaçant entre des guides verticaux, ou faiblement inclinés sur la verticale.» [1] [2].

Une autre définition proposée par futura tech «L'ascenseur est un dispositif mobile permettant le déplacement de personnes ou d'objets sur un axe prédéfini au sein d'une construction. L'ascenseur été inventé en 1853 par Elisha Graves Otis».

## **1.3 Les différents types d'ascenseurs**

Il existe des points communs entre tous les ascenseurs:

- L'étrier
- Les guides
- Les coulisseaux
- La cuvette
- La gain

On distingue essentiellement deux types de familles d'ascenseur :

- les ascenseurs à traction à câble,
- les ascenseurs hydrauliques.

En règle générale, ces deux types utilisent l'énergie électrique pour déplacer les cabines verticalement (moteur électrique continu ou alternatif) [3].

### **1.3.1 Les ascenseurs hydrauliques :**

#### **1.3.1.1 Description**

Il faut noter qu'il y a trois types d'ascenseurs hydrauliques :

- Ascenseur hydraulique à cylindre de surface.

- Ascenseur à cylindre enterré.
- Ascenseur à cylindre de surface télescopique [3].

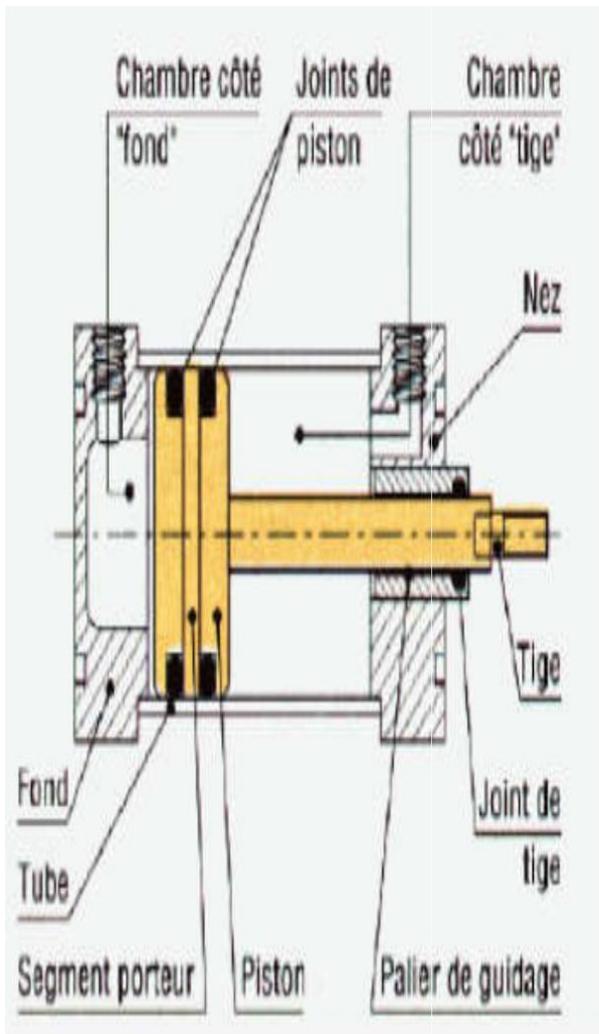


Figure (1.1) : schéma d'un cylindre

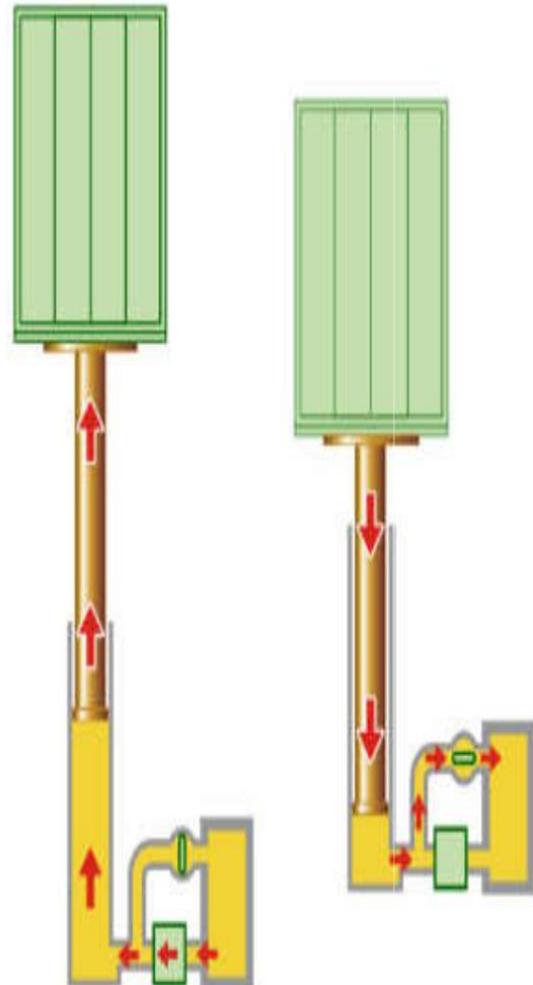


Figure (1.2): fonctionnement d'un ascenseur

### 1.3.1.2 Les différentes parties d'un ascenseur hydraulique

- Une cabine
- Guides, d'un ensemble pistons-cylindres hydrauliques placé sous la cabine de l'ascenseur
- Réservoir d'huile
- Moteur électrique accouplé à une pompe hydraulique
- Contrôleur [3].

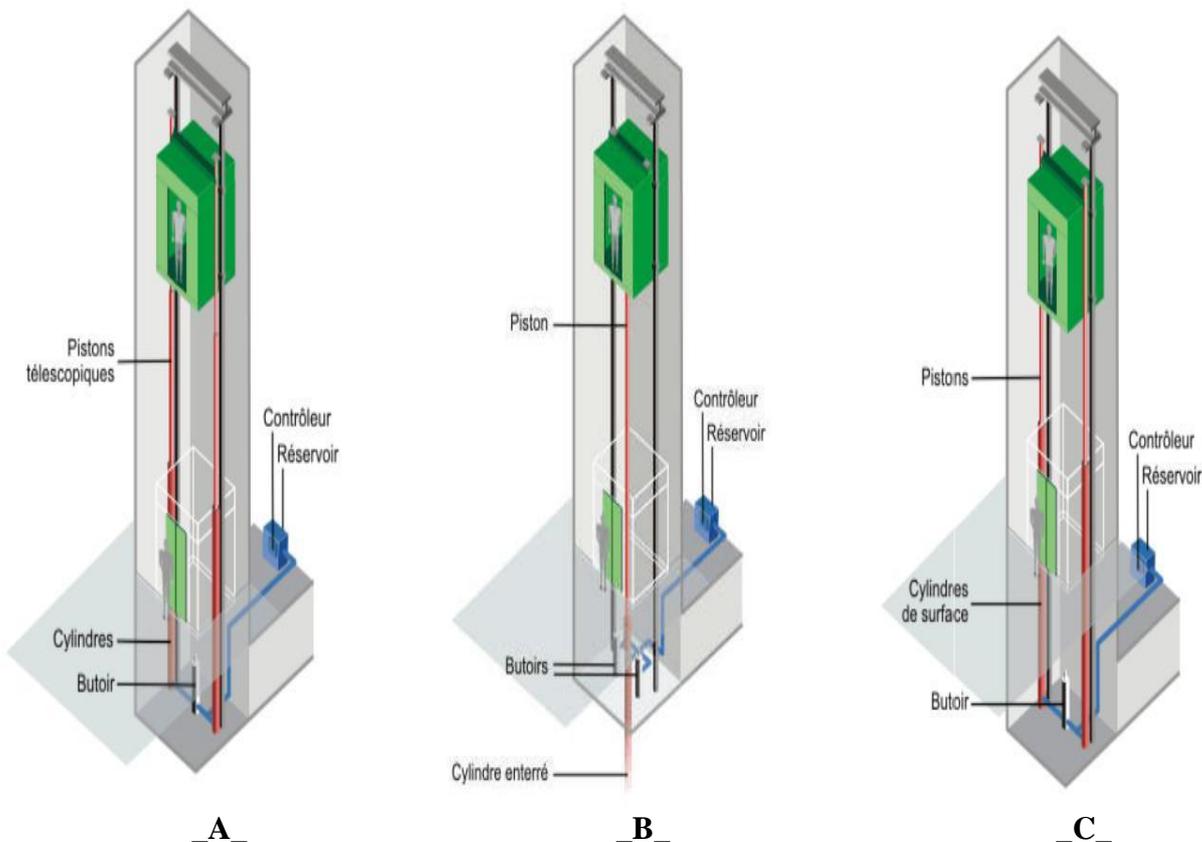


Figure (1.3) : -A- : Ascenseur à cylindre enterré. -B- : Ascenseur à cylindre de face télescopique. -C- : hydraulique à cylindre de surface

### 1.3.1.3 Les principaux avantages et inconvénients

#### Avantages

- Précision au niveau du déplacement (mise à niveau).
- Réglage facile de la vitesse de déplacement.
- Ne nécessite pas de cabine de machinerie.
- Implantation facile dans un immeuble existant

#### Inconvénients

- Course verticale limitée à une hauteur entre 15 et 18 m.
- Risque de pollution du sous-sol.
- Consommation énergétique importante.
- Nécessiter de renforcer la dalle de sol.

## 1.3.2 Ascenseurs à traction Câbles

### 1.3.2.1 Description

Les ascenseurs à traction à câbles sont les types d'ascenseurs que l'on rencontre le plus, notamment dans les bâtiments tertiaires.

Ils se différencient entre eux selon le type de motorisation :

- A moteur-treuil à vis sans fin
- A moteur-treuil planétaire
- à moteur à attaque directe (couramment appelé "Gearless" ou sans treuil) [3].

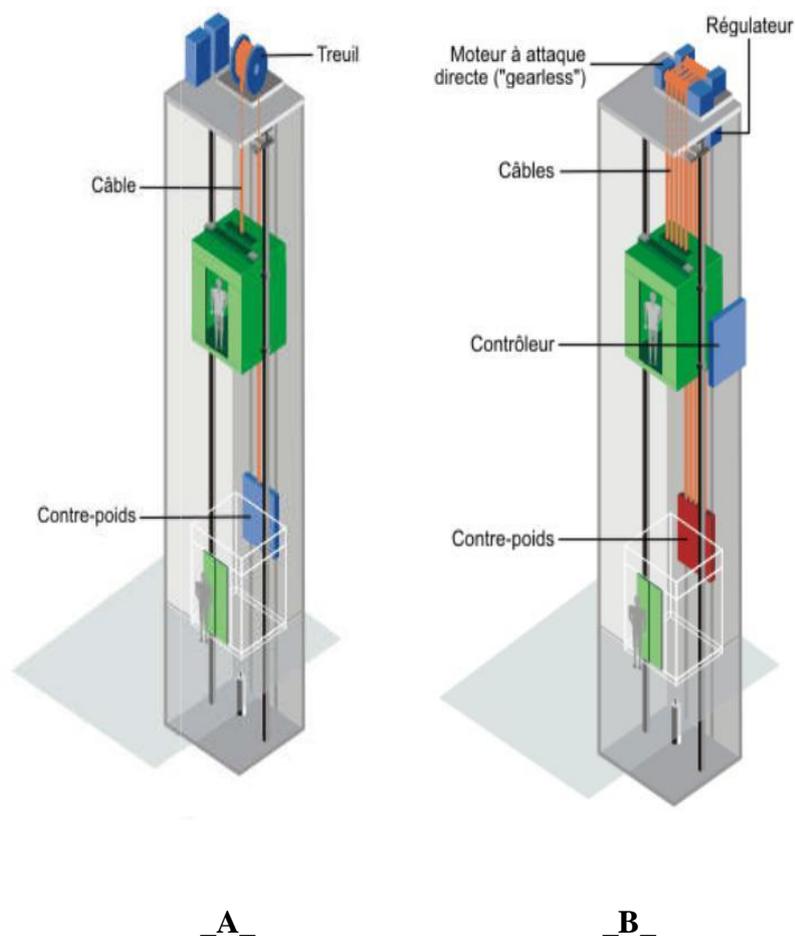


Figure (1.4) : A : types d'ascenseurs à traction câble - B : Ascenseur à moteur treuil planétaire.

### 1.3.2.2 Les différents avantages et inconvénients

#### Avantage :

- Course verticale pas vraiment limitée.
- Rapidité de déplacement.
- Efficacité énergétique importante.

- Pas de souci de pollution.

**Inconvénients :**

- En version standard, nécessite un cabanon technique en toiture.
- Exigence très importante sur l'entretien.

Néanmoins la traction par câble possède d'importants avantages par rapport à la traction hydraulique [3].

**1.4 Fonctionnement :**

Les explications suivantes se focalisent essentiellement sur le fonctionnement des ascenseurs à traction câble. La cabine est suspendue à un câble qui s'enroule sur une poulie à gorge et dont l'autre extrémité porte un contre poids. Le moteur entraîne la poulie par l'intermédiaire d'un réducteur de vitesse ; la poulie entraîne le câble par adhérence. La cabine est équilibrée à demi- charge par le contre poids ; il suffit donc d'un couple relativement faible pour mettre en mouvement la cabine ; la charge est positive pour la montée en charge et la descente à vide ; elle est négative dans le cas contraire. D'où un moteur qui doit être à deux sens de rotation, et de freinage dans les deux sens [20] [21].

**1.4.1 Constitution d'un ascenseur**

Quel que soit le type d'ascenseur à traction câble il comporte généralement :

- Une cabine.
- Une gaine.
- Un contrepoids, des câbles reliant la cabine au contrepoids.
- Des guides.
- Un système de traction au-dessus de la cage de l'ascenseur.

La Figure 1.5 présente un schéma explicatif sur le placement de chaque partie d'un ascenseur.

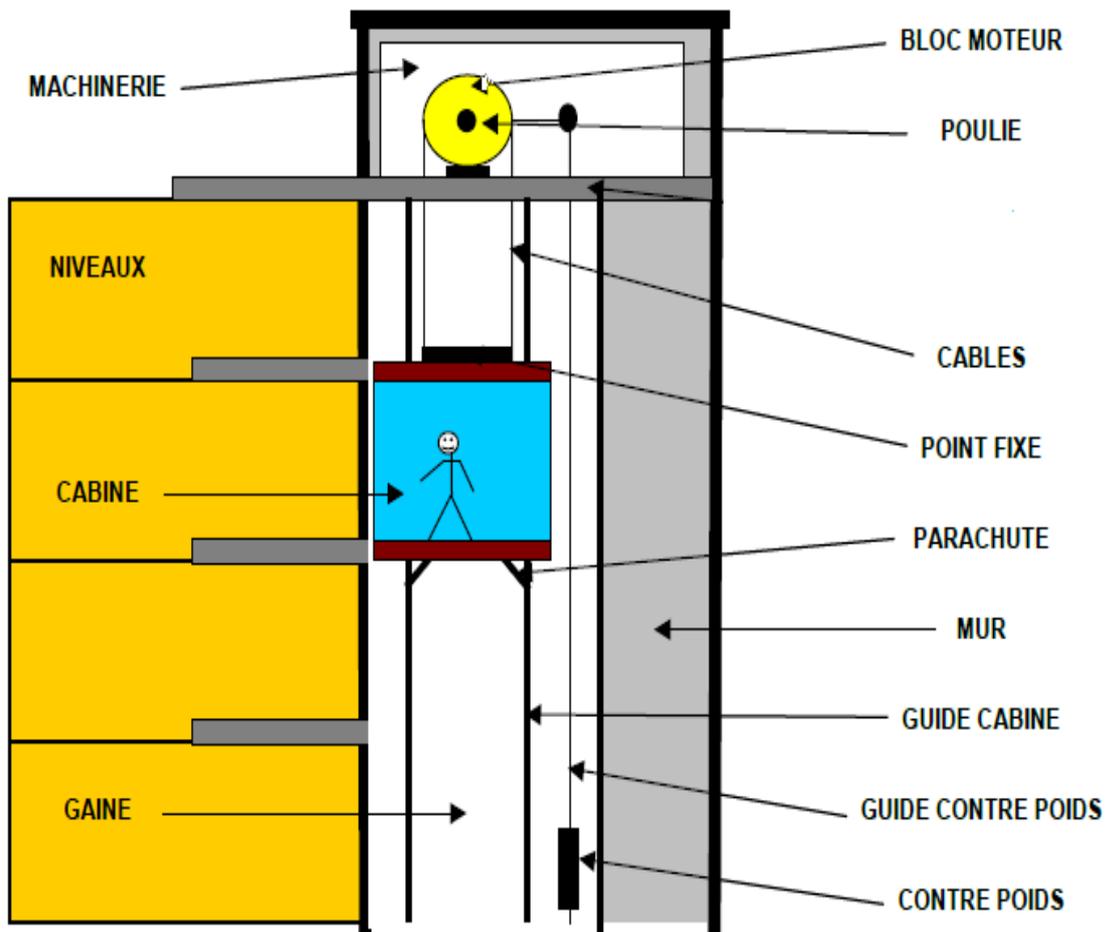


Figure (1.5) : Schéma explicatif de différentes parties d'un ascenseur.

#### 1.4.1.1 La cabine

La cabine est l'habitacle dans lequel les usagés voyagent. Si les premiers ascenseurs ne pouvaient contenir que cinq à six personnes, certains modèles actuels ont une capacité de trente personnes. Elle peut aussi être réalisée en verre afin d'offrir des vues panoramiques sur certains paysages. Elle possède des portes avec un système d'ouverture sécurisée, un système d'appel et d'affichage des paliers. Elle est disposée sur un piston et glisse sur des câbles créant des rails de circulation [20] [21].

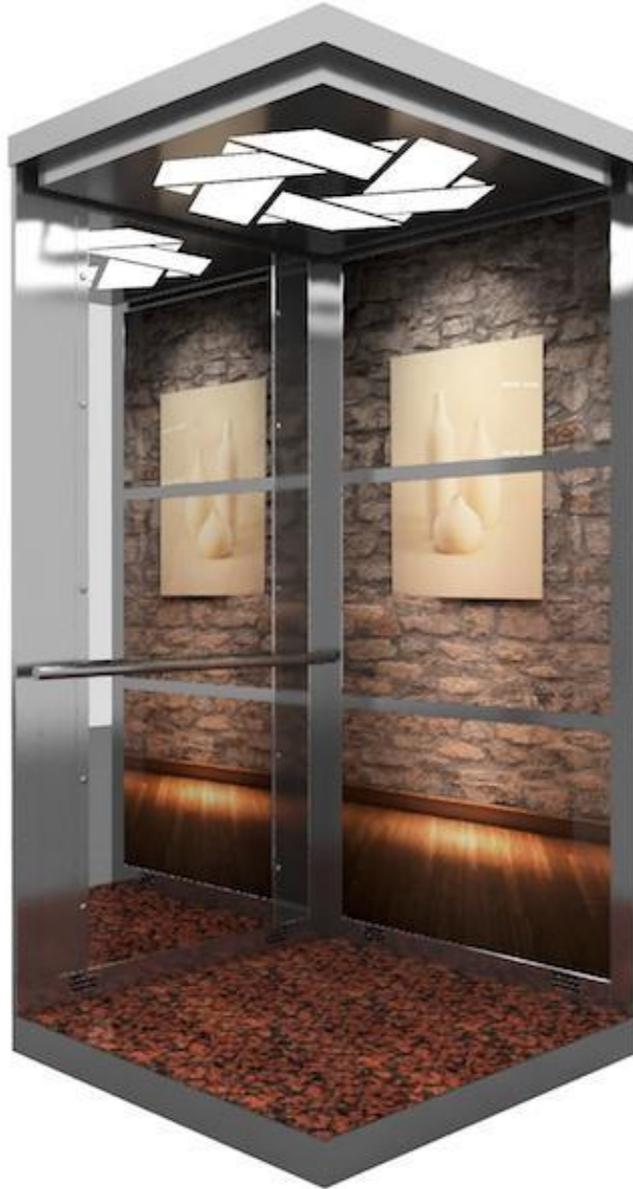


Figure (1.6) : Cabine

#### 1.4.1.2 Gaine

Contenant la cabine de l'ascenseur ainsi que les câbles et les mécanismes. Dans le cas des modèles modernes se passant de salle des machines, le moteur plus réduit est directement intégré dans la gaine [20][21].



Figure (1.7) : Gaine ou trémie

### 1.4.1.3 Le contre poids

Masse plus lourde que l'ascenseur, elle représente l'équivalent du poids de la cabine et la moitié de sa charge maximale.

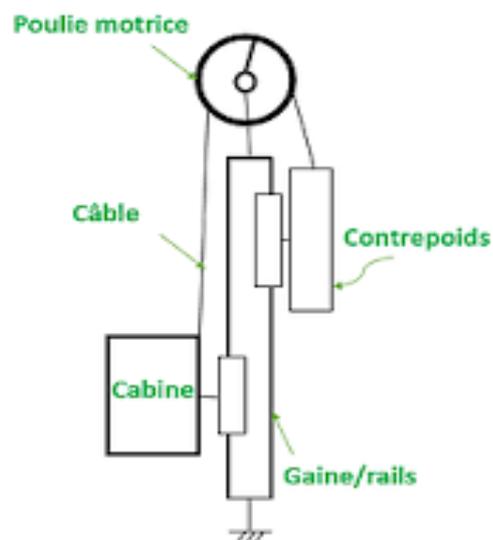


Figure (1.8) : Contre poids

**1.4.1.4 Les guides**

Sont les câbles sur lesquels est attachée la cabine. Ils servent de rails pour guider la cabine dans la gaine, sans qu'elle n'en touche les parois. Enroulés dans des gorges afin de ne pas glisser tout seul, ces câbles ou courroies sont eux-mêmes reliés à la cabine et au contrepoids.

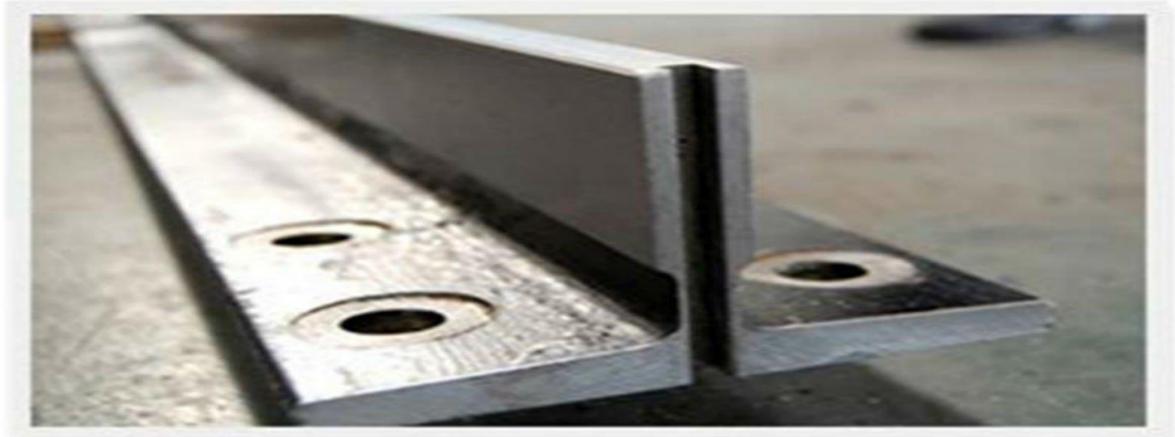


Figure (1.9) : Guide d'ascenseur

**1.4.1.5 Le moteur**

Il se situe dans une salle des machines tout en haut ou au sous-sol du bâtiment. De plus petits moteurs peuvent également être placés dans la gaine, sans salle des machines. Ils permettent de contrôler les mouvements de la cabine, en fonction des appels des utilisateurs. Les versions modernes intègrent aussi divers traitements de paramètres, afin de rentabiliser au mieux les déplacements des appareils [7] .



Figure (1.10) : Moteur d'ascenseur

### 1.4.1.6 Un système de traction

La figure ci-dessous représente un système de traction situé au-dessus de la cage de l'ascenseur (Machinerie). C'est le siège où se trouve le moteur et l'armoire de commande. Il peut être situé en partie haute, basse ou latéral [7].



Figure (1.11):Machinerie

## 1.5. Domaine d'utilisation d'ascenseur

### 1.5.1 Domaine urbaine

Un ascenseur est utilisé dans le domaine urbain pour faciliter le déplacement des gens et les faire gagner du temps et de force, dans les grands bâtiments qui ont de grande hauteur.

$N_i$  : NOMBRE DE PERSONNES POUVANT ÊTRE REÇUES SIMULTANÉMENT À L'ÉTAGE  $i$

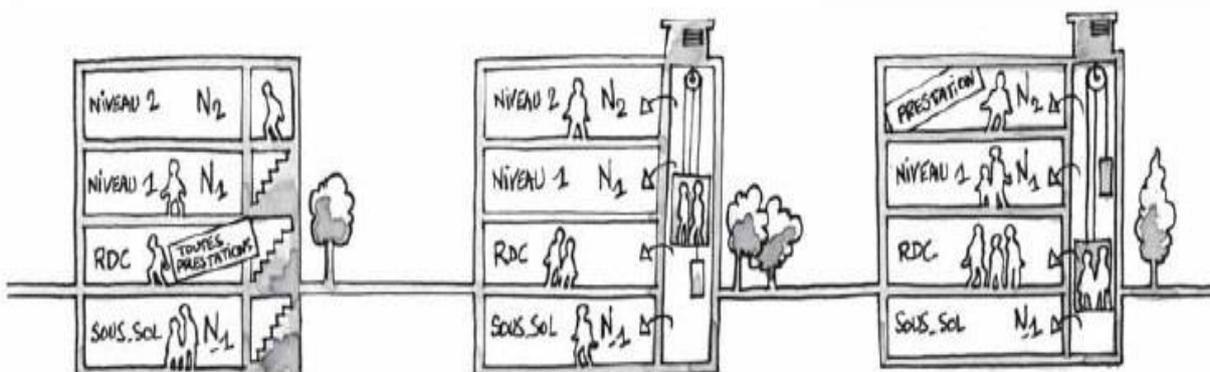


Figure (1.12) : Ascenseur dans un bâtiment.

### 1.5.2 Domaine médicale

On trouve ce domaine presque dans les hôpitaux et quelque bâtiment, qui en un service spécialisé il sert a transporté, les malades et les handicapés qui ne peuvent pas ce déplacer facilement.

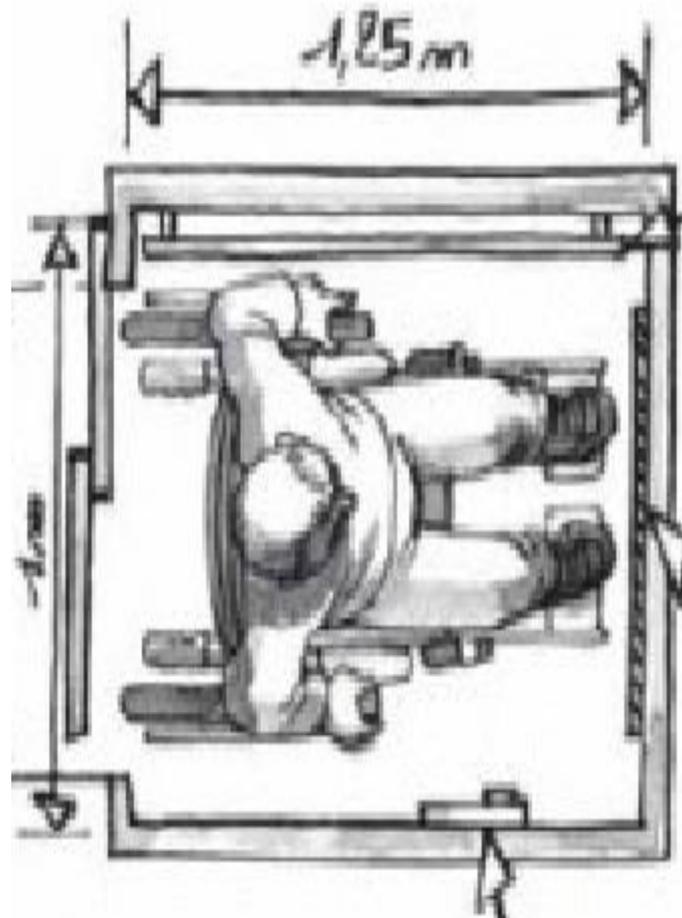


Figure (1.13) : Ascenseur pour handicapé

## 1.6 Exemple d'ascenseur a traction

### 1.6.1 Ascenseur MRL (Machine Room-less)

C'est un ascenseur à traction de câble mais avec une autre conception, les élévateurs MRL sont situés dans le puits, ce qui élimine le besoin d'une salle des machines traditionnelle, Ils sont conçus pour les bâtiments de faible à moyenne hauteur ayant jusqu'à 75m [6].



Figure. 1.14 : Ascenseur moteur électrique à attaque directe (type MRL)

## 1.7 Quelques technologies sur ascenseur

### 1.7.1 La charge nominale

C'est la charge pour laquelle l'appareil a été construit, elle varie en fonction de la surface de la cabine. Par exemple 225kg  $\rightarrow$  0.70m<sup>2</sup> [7].



Figure (1.15) : plaque signalétique

### 1.7.2 Le frein parachute

Le parachute est bien l'élément le plus important d'un ascenseur. C'est grâce à son inventeur, Elisha Otis, que quasiment tous les immeubles possèdent une ou plusieurs ascenseurs.

Quand la cabine descend trop vite, le limiteur de vitesse fait déclencher le bloc parachute. Les coins remontent alors violemment et se retrouvent coincés entre les parois du bloc parachute et le guide. Ce coincement provoque l'arrêt instantané de la cabine [7].

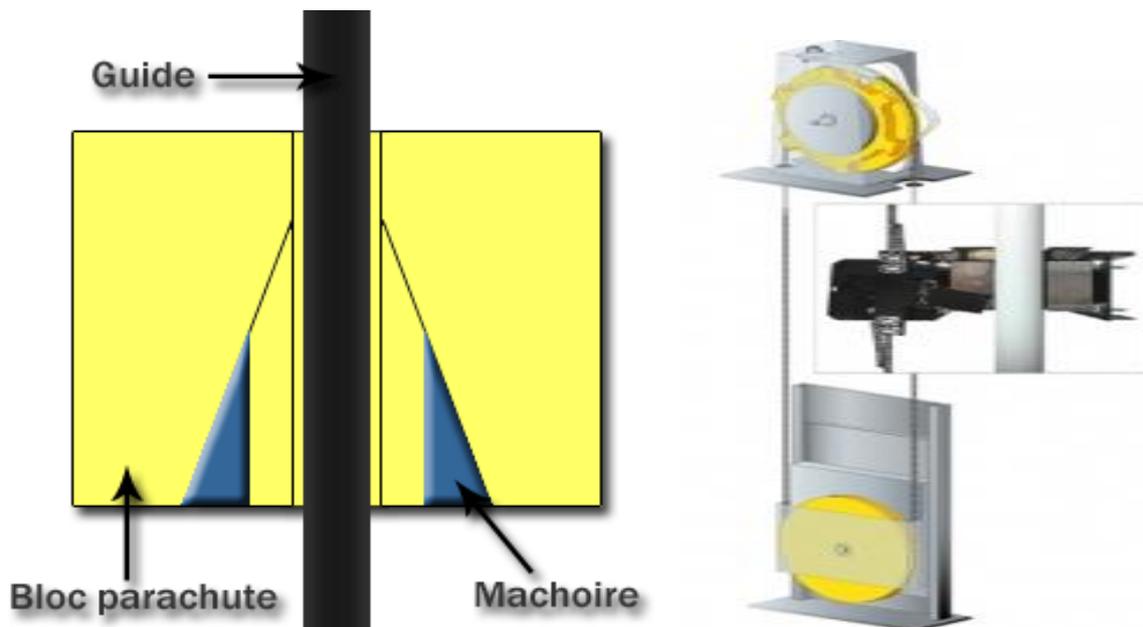


Figure (1.16) : Schéma système de parachute

### 1.7.3 Le contrôleur de manœuvre ou armoire de commande

Le contrôleur de manœuvre est un des composants les plus importants d'un ascenseur. C'est lui qui se charge de mémoriser les appels et envois la cabine, c'est lui qui commande le démarrage et l'arrêt du moteur, de l'ouverture de la porte cabine, etc. En fait, c'est le cerveau de l'ascenseur. Il est constitué de matériel électrique (des cartes électroniques et des relais) [7].



Figure (1.17) : armoire de commande

#### 1.7.4 Boîtier d'inspection:

Le boîtier d'inspection est un boîtier électrique situé sur le toit de la cabine. Son usage est réservé aux intervenants. Il permet de prendre le contrôle de l'appareil et de se déplacer (avec la cabine). Son bon fonctionnement doit être testé avant de monter sur la cabine, chose qui, si elle n'est pas faite, peut générer un accident pouvant aller jusqu'à la mort![7].



Figure (1.18) : Boîtier d'inspection

**1.8 Conclusion**

Les ascenseurs se développent de jour en jour d'une part pour améliorer l'expérience des utilisateurs et d'autre part la sortie de projets moins chers. Cet appareil occupe une place importante dans la catégorie des transports en commun, mais certains défis persistent tels que la gestion des flux de personnes dans les grands bâtiments ou la vitesse de déplacement dans les bâtiments longs car le déplacement à grande vitesse dans des pièces fermées impose des contrôles à respecter.

Dans ce premier chapitre, nous avons présenté le concept d'ascenseur électrique, puis nous avons discuté des types et des divers dispositifs inclus dans l'installation, en tenant compte de l'explication détaillée de chaque appareil, à travers tous les développe du système d'ascenseurs.

## 2.1 Introduction

Dans ce chapitre on s'intéresse à l'aspect programmation de l'ascenseur. et on va faire une petite vue sur les microcontrôleurs et ces différentes caractéristiques et aussi une définition de l'ensemble Arduino et exactement Arduino Atmega 2560.

Il y a de nombreux microcontrôleurs et de nombreuses plateformes basées sur des microcontrôleurs disponibles pour l'électronique programmée et beaucoup d'autres qui offrent des fonctionnalités comparables. Tous ces outils prennent en charge les détails compliqués de la programmation des microcontrôleurs et les intègrent dans une présentation facile à utiliser.

On trouve que les systèmes conçus à partir d'une carte Arduino simplifient la façon de travailler avec les microcontrôleurs.

## 2.2 Microcontrôleur

Un microcontrôleur est un système qui ressemble à un ordinateur : il a une mémoire, un processeur, des interfaces avec le monde extérieur.

Les microcontrôleurs ont des performances réduites, mais sont de faible taille et consomment peu d'énergie, les rendant indispensables dans toute solution d'électronique embarquée (voiture, porte de garage, robots, ...).

### 2.2.1 Les types de microcontrôleur les plus connus courants :

- Les microcontrôleurs Atmel AVR (Atmega, Attiny, Xmega, etc..).
- Les microcontrôleurs PIC de technologie micro puce.
- Le microcontrôleur basé sur technologie ARM.
- Les microcontrôleurs Philips : P89C51RD2BPN.
- Les microcontrôleurs Motorola famille 68HCXXX.

Un microcontrôleur, contrairement à un processeur d'ordinateur classique, est une sorte de mini-ordinateur qui embarque dans une puce électronique pratiquement tout ce qui est nécessaire pour son fonctionnement. Cette puce contient entre autres :

- Un processeur (CPU) : exécutant les instructions du programme.
- Une mémoire RAM : utilisée durant l'exécution du programme.
- Une mémoire morte ROM : utilisée pour stocker des données permanentes.
- Une mémoire flash : utilisée pour stocker les programmes.

- Des entrées-sorties : pour communiquer avec l'extérieur.

La Carte Arduino n'est pas le microcontrôleur le plus puissant, mais son architecture a été publiée en open-source, et toute sa philosophie s'appuie sur le monde du libre, au sens large [11].

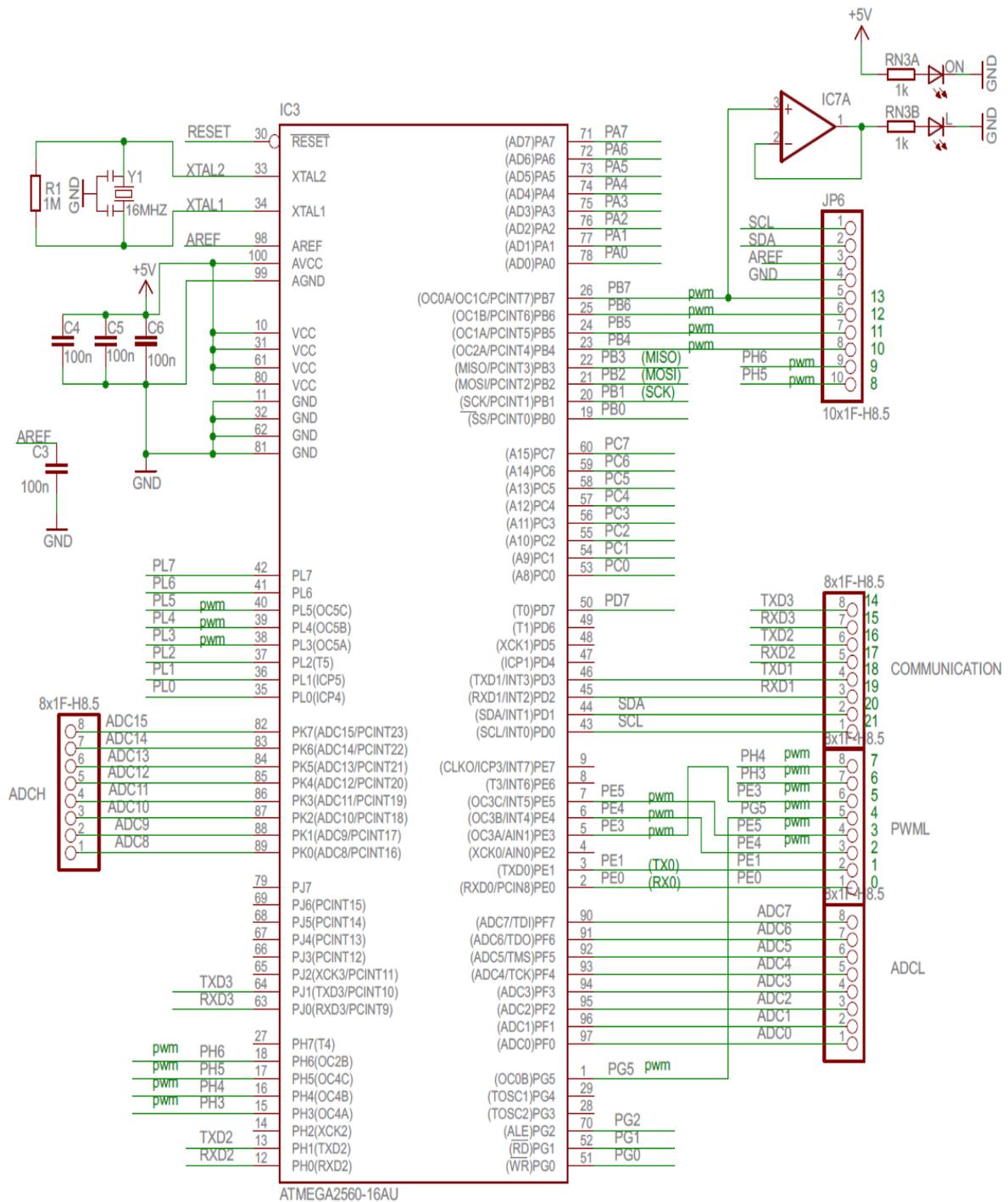


Figure (2.1) : Schématique d'un arduino Mega 2560.

### 2.3 Définition d'un Arduino

Arduino est un circuit imprimé en matériel libre sur lequel se trouve un microcontrôleur qui peut être programmé, analyser et produire des signaux électriques, de manière à effectuer des tâches très diverses comme des applications sur la domotique (le contrôle des appareils domestiques, éclairage, chauffage, ...etc.), le pilotage d'un robot, ...etc.

Arduino est un projet créé par une équipe de développeurs, composée de six individus comme: Massimo Banzi, et Gianluca Martino. Cette équipe a créé le "système Arduino". C'est un outil qui va permettre aux débutants, amateurs ou professionnels de créer des systèmes électroniques plus ou moins complexes.

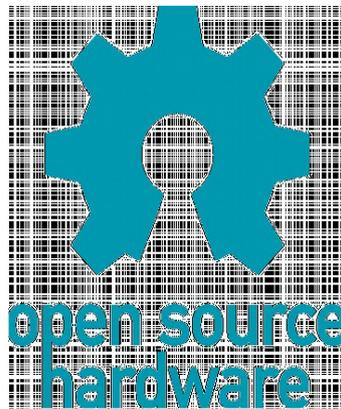


Figure (2.2) : Schémas électroniques et typons

Le code source de l'environnement de programmation et les bibliothèques embarquées sont disponibles sous licence LGPL [9].

#### 2.3.1 Carte arduino Mega 2560

La carte Arduino Mega 2560 est basée sur un ATmega2560 cadencé à 16 MHz. Elle dispose de 54 E/S dont 14 PWM, 16 analogiques et 4 UARTs. Elle est idéale pour des applications exigeant des caractéristiques plus complètes que l'uno. Des connecteurs situés sur les bords extérieurs du circuit imprimé permettent d'enficher une série de modules complémentaires. Elle peut se programmer avec le logiciel Arduino. Le contrôleur ATmega2560 contient un bootloader qui permet de modifier le programme sans passer par un programmeur. Le logiciel est téléchargeable gratuitement [10].



Figure (2.3) : Arduino Mega 2560

### 2.3.1.1 Le but et l'utilité

Le système Arduino, nous donne la possibilité d'allier les performances de la programmation à celles de l'électronique, plus précisément nous allons programmer des systèmes électroniques, Le gros avantage de l'électronique programmée c'est qu'elle simplifie grandement les schémas électroniques et par conséquent, le coût de la réalisation, mais une fois que vous serez rentré dans le monde de l'Arduino, vous serez fasciné par l'incroyable puissance dont il est question et des applications possibles, Une la carte Arduino est une petite carte électronique équipée d'un microcontrôleur.

Le microcontrôleur permet, à partir d'événements détecté par des capteurs, de programmer et commander des actionneurs, la carte Arduino est donc une interface programmable. Prenons exemple l'arduino méga comme il et l'un des Catre les plus utilisé.

La plateforme Arduino se présente sur plusieurs séries à savoir : Arduino UNO, Arduino Nano, Arduino Lilypad, Arduino DUE et Arduino Méga 2560 qui sera le choisi pour notre projet.

Nous utiliserons une carte Arduino MEGA 2560 disposant d'un microcontrôleur Atmel ATmega 2560[11].



Figure (2.4) : les modelés d'arduino

### 2.3.1.2 Caractéristiques

Parmi les caractéristiques de la carte Arduino Méga 2560, on trouve :

Microcontrôleur	ATmega2560
Tension de fonctionnement	5V
Tension d'entrée (recommandé)	7-12V
Tension d'entrée (limite)	6-20V
Digital I/O Pins	54 (dont 15 fournissent sortie PWM)
Broches d'entrée analogiques	16
DC Courant par I/O Pin	40 mA
Courant DC pour 3,3 Pin	50 mA
Mémoire Flash	256 Ko (ATmega2560) dont 8 Kb utilisé par Boot Loader
SRAM	8 Kb(ATmega2560)

EEPROM	4 Kb (ATmega2560)
Fréquence d'horloge	16 MHz

Tableau (1.1) : Présentation des caractéristiques de la carte Arduino Méga 2560

### 2.3.1.3 Entrées et sorties numériques

Chacune des 54 broches numériques de la carte Arduino Méga 2560 peut être utilisée soit comme une entrée ou une sortie numérique, par programmation de cette broche, en utilisant les instructions *pinMode()*, *digitalWrite()* et *digitalRead()* de la plateforme Arduino. Chaque broche peut fournir ou recevoir un courant maximum de 40mA et dispose d'une résistance interne de "rappel au plus" (pull-up) de 20-50 KΩ déconnectée par défaut. Cette résistance interne s'active sur une broche en entrée à l'aide de l'instruction `digitalWrite(nom_broche, 'HIGH')`. De plus, certaines broches possèdent des fonctions spéciales comme `serial.begin()`, `attachInterrupt()`, ...etc :[11]

Communication Série	<ul style="list-style-type: none"> <li>✚ Port Série Serial 0: broche 0 ←RX, broche 1 ←TX</li> <li>✚ Port Série Serial 1: broche 19 ←RX, broche 18 ←TX</li> <li>✚ Port Série Serial 2: broche 17 ←RX, broche 16 ←TX</li> <li>✚ Port Série Serial 3: broche 15 ←RX, broche 14 ←TX</li> </ul>
Interruptions Externes	<ul style="list-style-type: none"> <li>✚ Interruption N°0 correspond au Broche 2</li> <li>✚ Interruption N°1 correspond au Broche 3</li> <li>✚ Interruption N°2 correspond au Broche 21</li> <li>✚ Interruption N°3 correspond au Broche 20</li> <li>✚ Interruption N°4 correspond au Broche 19</li> <li>✚ Interruption N°5 correspond au Broche 18</li> </ul>
Impulsion PWM (Largeur d'impulsion modulée)	Broches 0 à 13 fournissent une impulsion PWM 8-bits à l'aide de l'instruction <code>AnalogWrite()</code>
SPI (Interface Série Périphérique)	<ul style="list-style-type: none"> <li>✚ Broche 50 (MISO) (Master In, Slave Out)</li> <li>✚ Broche 51 (MOSI) (Master Out, Slave In)</li> <li>✚ Broche 52 (SCK) (Serial CLock)</li> <li>✚ Broche 53 (SS) (Slave Select)</li> </ul>
I <sup>2</sup> C (Inter –Intergrated)	<ul style="list-style-type: none"> <li>✚ Broche 20 (SDA) (Serial Data)</li> <li>✚ Broche 21 (SCL) (Serial CLock)</li> </ul>
LED	LED ← Broche 13.

Tableau (1.2) : Différentes broches entrées/sorties numériques

### **2.3.1.4 Broches analogiques**

La carte Arduino Méga 2560 dispose de 16 entrées analogiques, chacune pouvant fournir une mesure d'une résolution de 10 bits (c.-à-d. sur 1024 niveaux soit de 0 à 1023) à l'aide de la 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.

Note: les broches analogiques peuvent être utilisées en tant que broches numériques [11].

### **2.3.1.5 Autres broches**

Il y a deux autres broches disponibles sur la carte :

AREF : Tension de référence pour les entrées analogiques (si différent du 5V). Utilisée avec la fonction prédéfinie de la plateforme arduino `analogReference ( )`.

RESET : Mettre cette broche au niveau BAS entraîne la réinitialisation (= le redémarrage) du microcontrôleur. Typiquement, cette broche est utilisée pour ajouter un bouton de réinitialisation sur le circuit qui bloque celui présent sur la carte [12].

### **2.3.1.6 Communications:**

La carte Arduino Méga 2560 peut facilement communiquer via une communication série avec un ordinateur, une autre carte Arduino, ou avec d'autres microcontrôleurs. Elle dispose de 4 UARTs (Universal asynchrone nous Receiver Transmitter ou émetteur-récepteur asynchrone universel en français) pour communication série de niveau TTL (5V). Le circuit intégré ATmega8U2 sur la carte Arduino assure la connexion entre cette communication série de l'un des ports série de l'ATmega 2560 vers le port USB de l'ordinateur qui apparaît comme un port COM virtuel pour les logiciels de l'ordinateur.

Le logiciel Arduino inclut une fenêtre terminal série (ou moniteur série) sur l'ordinateur et qui permet d'envoyer des textes simples depuis et vers la carte Arduino. Les deux LEDs de RX et de TX sur la carte clignote lorsque les données sont transmises via le circuit intégré ATmega8U2.

L'ATMega 2560 supporte également la communication par protocole I<sup>2</sup>C (ou interface TWI (Two Wire Interface - Interface "2 fils") et SPI [12].

Le boot loader (programme) est préprogrammé une fois pour toute dans l'ATMega et qui permet la communication entre l'ATMega et le logiciel Arduino via le port USB, notamment lors de chaque programmation de la carte arduino .

Les mémoires des cartes Arduino sont organisées suivant une architecture de type Harvard, c'est-à-dire que des bus séparés sont utilisés pour l'espace mémoire du programme (mémoire Flash) et l'espace mémoire associée aux données (mémoire SRAM) [11].

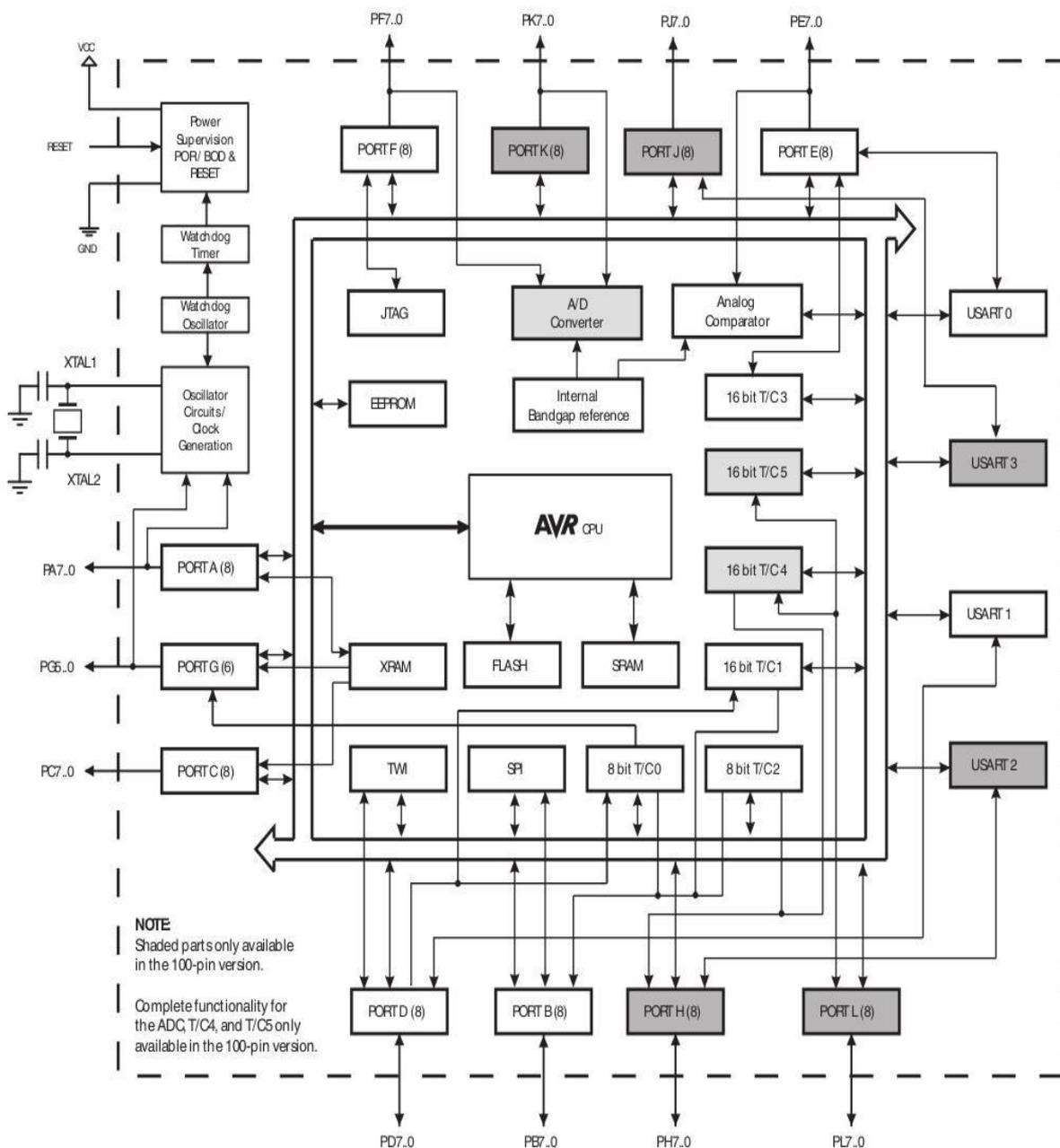


Figure (2.5) : Schéma interne de l'ATmega 2560

### 2.3.1.7 Applications

Le système Arduino nous permet de réaliser un grand nombre de choses, qui ont une application dans tous les domaines, l'étendue de l'utilisation de l'Arduino est gigantesque. On a quelques exemples:

- Contrôler les appareils domestiques.
- télécommander un appareil mobile (modélisme)

### 2.3.1.8 Alimentation

L'Arduino Méga 2560 peut être alimentée via la connexion USB ou avec une alimentation externe. Dans le premier cas, la source d'alimentation est automatiquement sélectionnée.

Externe (non-USB) alimentation peut provenir d'un adaptateur AC –DC ou batterie. L'adaptateur peut être connecté en branchant une prise de centre positif de 2,1 mm 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) du connecteur d'alimentation. [9].

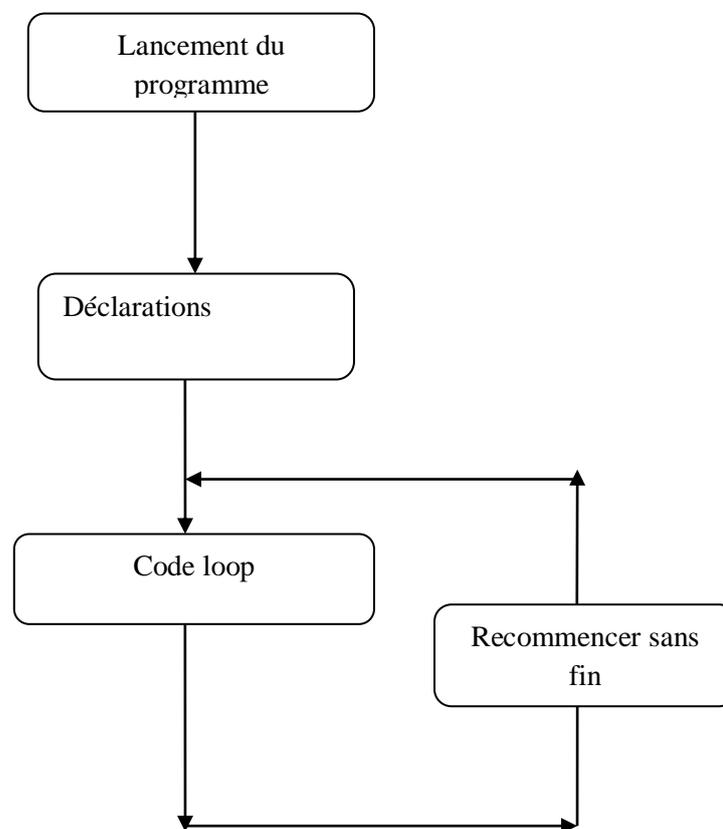


Figure (2.6) : Organigramme Structure générale d'un projet arduino

### 2.3.1.8 Avantages et Inconvénients

#### Avantage

- Multiplateforme : même IDE pour linux.
- Programmation claire et intuitive
- Entièrement open-source : hardware et software
- Possibilité d'utiliser l'Arduino comme programmeur d'AVR
- Communauté très importante : forum officiel, sites tiers, ...
- Librairies officielles ou non
- Le plus important : on branche.

#### Inconvénients

- Puissance du microcontrôleur bridée par les fonctions simples d'emploi.
- Programmation en C, et pas en assembleur (peut être gênant).
- Prix des shields assez élevé en général [13].

### 2.4 Les différents accessoires utilisés

Généralement, la carte Arduino est associée aux accessoires (Shields en anglais) qui simplifient les réalisations de notre projet.

#### 2.4.1 Moteur pas à pas

Un moteur pas à pas est une machine qui tourne en fonction des impulsions électriques reçues dans ses bobinages. L'angle de rotation minimal entre deux modifications des impulsions électriques s'appelle un pas ( $\emptyset p$ ). On caractérise un moteur par le nombre de pas par tour (c'est à dire pour  $360^\circ$ ).

Le nombre du pas par tour est déterminé par la relation suivante :

$$N_p = 2\pi / \emptyset p$$

On peut dire, le moteur pas à pas est un convertisseur électromécanique destiné à transformer le signal électrique (impulsion ou train d'impulsion) en déplacement (angulaire ou linéaire) mécanique.

Il existe plusieurs types des moteurs pas à pas selon leurs différents principes de réalisations. Néanmoins, on utilise dans les schémas électriques un seul symbole unique [15].

### **2.4.1.1 Différents types des moteurs**

Il y a les moteurs à aimants permanents et les moteurs à réluctance variables Il existe deux types des moteurs à aimants permanents :

- a) les moteurs unipolaires
- b) les moteurs bipolaires.

Sur un plan fondamental, ces deux types fonctionnent exactement de la même manière ;des électro-aimants sont activés de façon séquentielle, ce qui induit l'arbre moteur central à tourner.

La différence entre les deux types est les niveaux de tension. Un moteur pas à pas unipolaire fonctionne uniquement avec une tension positive, de sorte que les hautes et basses tensions appliquées aux bobines électromagnétiques seraient quelque chose comme 5V et 0 V. Un moteur pas à pas bipolaire a deux polarités, positives et négatives, de sorte que ses hautes et basses tensions seraient quelque chose comme 2,5V et -2,5 V.

### **2.4.1.2 Principe de fonctionnement**

Un moteur pas-à-pas (stepping Motors en anglais) est un moteur à Courant continu qui avance d'un seul pas lorsque le sens du courant dans une des bobines change de sens.

Les moteurs pas-à pas sont des micromoteurs à réluctance (dits également à réluctance variable), qui fonctionne grâce à l'attraction d'une masse Polaire (le rotor) par un champ magnétique.

Chaque phase (1-2-3) du stator reçoit tour à tour une impulsion électrique conventionnellement positive ou négative selon le sens du courant dans la bobine.

Les impulsions arrivent dans un ordre prédéterminé de distribution et suivant une fréquence déterminée et réglable.

Chacune de ces impulsions correspond à un déplacement angulaire appelé pas ou “ step ” Un "pas" est une unité de décalage angulaire, en général :  $1.8 \text{ degrés}$ . (le moteur aura alors 200 pas.)  $1,8 \times 200 = 360 \text{ °}$  [15].

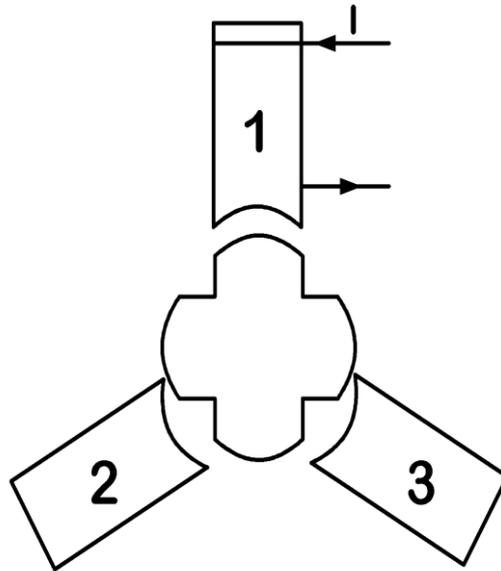


Figure (2.7) : phase du stator

### 2.4.2 Capteur de mouvement infrarouge

Après 1975, le rayonnement infrarouge a remplacé progressivement les ultrasons. L'utilisateur y gagne en portée et en sûreté de transmission, le fabricant en prix de revient. La vitesse de propagation élevée du rayonnement infrarouge exclut les perturbations par interférences, réflexion et effet Doppler. Les bruits de distorsion n'ont pas ici l'influence néfaste qui perturbait les vieux appareils à ultrasons.

#### 2.4.2.1 Définition :

Rayonnement électromagnétique dont les longueurs d'onde sont comprises entre 750 et 1500 nm. La figure ci-dessous représente le spectre lumineux de l'ultraviolet à l'infrarouge. Seule la partie comprise entre 390 et 760 nm est visible [13].

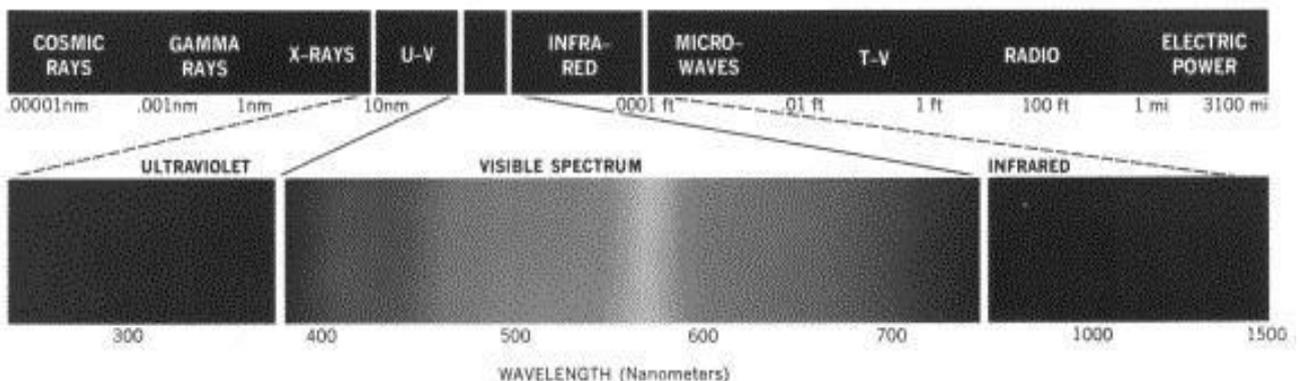


Figure (2.8) : le spectre lumineux de l'ultraviolet à l'infrarouge.

### **2.4.2.2 Quelque avantage et inconvénient :**

#### **Avantages**

- Construction simple.
- Bon marché.
- Utilisation non soumise à autorisation.
- Faible consommation.
- Fonctionnement sur piles possible si les émissions ne sont pas permanentes.
- Faible de tension de service.
- Utilisation simple.
- Encombrement réduit.

#### **Inconvénients**

- Portée réduite dans les milieux éclairés.
- Sensibilité aux parasites des lampes à faible consommation ou des tubes fluorescents.
- Trop grande variété du fait de l'absence de normes.
- Propagation limitée.
- Directivité.
- Faible de portée, que peut encore réduit l'encrassement faible cadence de modulation(ne concerne pas IrDA) [13].

### **2.4.3 Les afficheurs à cristaux liquides :**

Les afficheurs à cristaux liquides, autrement appelés afficheurs LCD (Liquid Crystal Display), sont des modules compacts intelligents et nécessitent peu de composants externes pour un bon fonctionnement. Ils consomment relativement peu (de 1 à 5 mA), sont relativement bons marchés et s'utilisent avec beaucoup de facilité.

Plusieurs afficheurs sont disponibles sur le marché et diffèrent les uns des autres, non seulement par leurs dimensions, (de 1 à 4 lignes de 6 à 80 caractères), mais aussi par leurs caractéristiques techniques et leur tension de service. Certains sont dotés d'un rétro éclairage de l'affichage. Cette fonction fait appel à des LED montées derrière l'écran du module, cependant, cet éclairage est gourmand en intensité (de 80 à 250 mA).



Figure (2.9): Afficheur LCD

Ils sont très utilisés dans les montages à microcontrôleur, et permettent une grande convivialité. Ils peuvent aussi être utilisés lors de la phase de développement d'un programme, car on peut facilement y afficher les valeurs de différentes variables [18].

#### 2.4.3.1 Principe des cristaux liquides

L'afficheur est constitué de deux lames de verre, distantes de 20  $\mu\text{m}$  environ, sur lesquelles sont dessinées les mantisses formant les caractères. L'espace entre elles est rempli de cristal liquide normalement réfléchissant (pour les modèles réfléchitifs). L'application entre les deux faces d'une tension alternative basse fréquence de quelques volts (3 à 5 V) le rend absorbant. Les caractères apparaissent sombres sur fond clair. N'émettant pas de lumière, un afficheur à cristaux liquides réfléchitif ne peut être utilisé qu'avec un bon éclairage ambiant. Sa lisibilité augmente avec l'éclairage. Les modèles transmissifs fonctionnent différemment. Ils sont normalement opaques au repos, le cristal liquide devient transparent lorsqu'il est excité; pour rendre un tel afficheur lisible, il est nécessaire de l'éclairer par l'arrière, comme c'est le cas pour les modèles rétro éclairés [18].

#### 2.4.3.2 Brochage

Un circuit intégré spécialisé est chargé de la gestion du module. Il remplit une double fonction: d'une part, il commande l'affichage et de l'autre se charge de la communication avec l'extérieur.

Les broches 15 et 16 ne sont présentes que sur les afficheurs LCD avec rétro éclairage. Les connexions à réaliser sont simples puisque l'afficheur LCD dispose de peu de broches.

Il faut évidemment, l'alimenter, le connecter à un bus de donnée (4 ou 8 bits), et connecter les broches E, R/W et RS [18].

Broche	Nom	Niveau	Fonction
1	Vss	-	Masse
2	Vdd	-	Alimentation positive +5V
3	Vo	0-5V	Cette tension permet, en la faisant varier entre 0 et +5V, le réglage du contraste de l'afficheur.
4	RS	TTL	Sélection du registre (Register Select) Grâce à cette broche, l'afficheur est capable de faire la différence entre une commande et une donnée. Un niveau bas indique une commande et un niveau haut indique une donnée.
5	R/W	TTL	Lecture ou écriture (Read/Write) L : Écriture H : Lecture
6	E	TTL	Entrée de validation (Enable) active sur front descendant. Le niveau haut doit être maintenue pendant au moins 450 ns à l'état haut.
7	D0	TTL	Bus de données bidirectionnel 3 états (haute impédance lorsque E=0)
8	D1	TTL	
9	D2	TTL	
10	D3	TTL	
11	D4	TTL	
12	D5	TTL	
13	D6	TTL	
14	D7	TTL	
15	A	-	Anode rétro éclairage (+5V)
16	K	-	Cathode rétro éclairage (masse)

Tableau (1.3): description de différentes broches de l'afficheur LCD

### 2.4.3.3 Les modes de fonctionnement de l'afficheur :

Il existe Deux modes de fonctionnement de l'afficheur sont disponibles, le mode 4 bits et le mode 8 bits.

#### 2.4.3.3.1 Mode de 8 bits :

Dans ce mode, les données sont envoyées à l'afficheur sur les broches D0 à D7. On place la ligne RS à 0 ou à 1 selon que l'on désire transmettre une commande ou une donnée. Il faut aussi placer la ligne R/W à 0 pour indiquer à l'afficheur que l'on désire effectuer une écriture. Il reste à envoyer une impulsion d'au moins 450 ns sur l'entrée E, pour indiquer que des données valides sont présentes sur les broches D0 à D7. Si on désire au contraire effectuer une lecture, la procédure est identique, mais on place cette fois la ligne R/W à 1 pour demander une lecture. Les données seront valides sur les lignes D0 à D7 lors de l'état haut de la ligne E [18].

#### 2.4.3.3.2 Mode de 4 bits :

Il peut être nécessaire dans certains cas de diminuer le nombre de fils utilisés pour commander l'afficheur comme, par exemple lorsqu'on dispose de très peu de broches d'entrées sorties disponibles sur un microcontrôleur. Dans ce cas, on peut utiliser le mode quatre bits de l'afficheur LCD. Dans ce mode, seuls les 4 bits de poids fort (D<sub>4</sub> à D<sub>7</sub>) de l'afficheur sont utilisées pour transmettre les données et les lire. Les 4 bits de poids faible (D<sub>0</sub> à D<sub>3</sub>) sont alors connectés à la masse. On a donc besoin, hors alimentation de sept fils pour commander l'afficheur. Les données sont alors écrites ou lues en envoyant séquentiellement les quatre bits de poids fort suivi des quatre bits de poids faible. Une impulsion positive de moins 450 ns doit être envoyée sur la ligne E pour valider chaque demi-octet.

Dans les deux modes, on peut, après chaque action sur l'afficheur, vérifier que celui-ci est en mesure de traiter l'information suivante. Pour cela, il faut demander une lecture en mode commande, et tester le flag Busy "BF" de contrôleur de l'afficheur LCD. Lorsque BF=0, l'afficheur est prêt à recevoir une nouvelle commande ou donnée.

Il se peut qu'on dispose encore de moins de broches disponibles dans l'application envisagée. Dans ce cas, on peut alors relier la ligne R/W à la masse de façon à forcer l'afficheur en écriture. On a alors besoin, de mettre seulement six fils hors alimentation en mode 4 bits, et dix fils en mode 8 bits, pour commander l'afficheur, mais on ne peut

alors plus relire l'afficheur. Ceci n'est pas gênant dans la mesure où on sait ce qu'on a écrit sur l'afficheur, mais on ne peut alors plus relire le flag Busy. Il faut alors utiliser des temporisations après chaque écriture sur l'afficheur. On perd alors un peu en temps d'affichage, mais on gagne une broche d'entrée sortie [18].

#### 2.4.4 Le logiciel PROTEUS:

Le logiciel PROTEUS est composé de trois modules :

- L'éditeur de schéma ISIS
- Le simulateur LISA
- L'outil de conception de circuit imprimé ARES

ISIS produit d'une part, une liste d'équipotentiels qui peut être utilisée par le simulateur LISA et l'outil de conception de circuit imprimé ARES, d'autre part, une liste de matériel et des rapports de contrôle des règles électriques.

LISA est un ensemble de modules de simulation lié à ISIS. Le noyau de simulation PROSPICE est basé sur la version 3F5 du moteur SPICE publié par l'université de Berkeley.

ARES est un module de conception de circuit imprimé compatible Windows 98, 2000 et XP. Il permet le placement des composants en mode automatique, manuel ou semi-automatique et le routage des liaisons sur plusieurs couches en mode automatique, manuel ou semi-automatique [23].

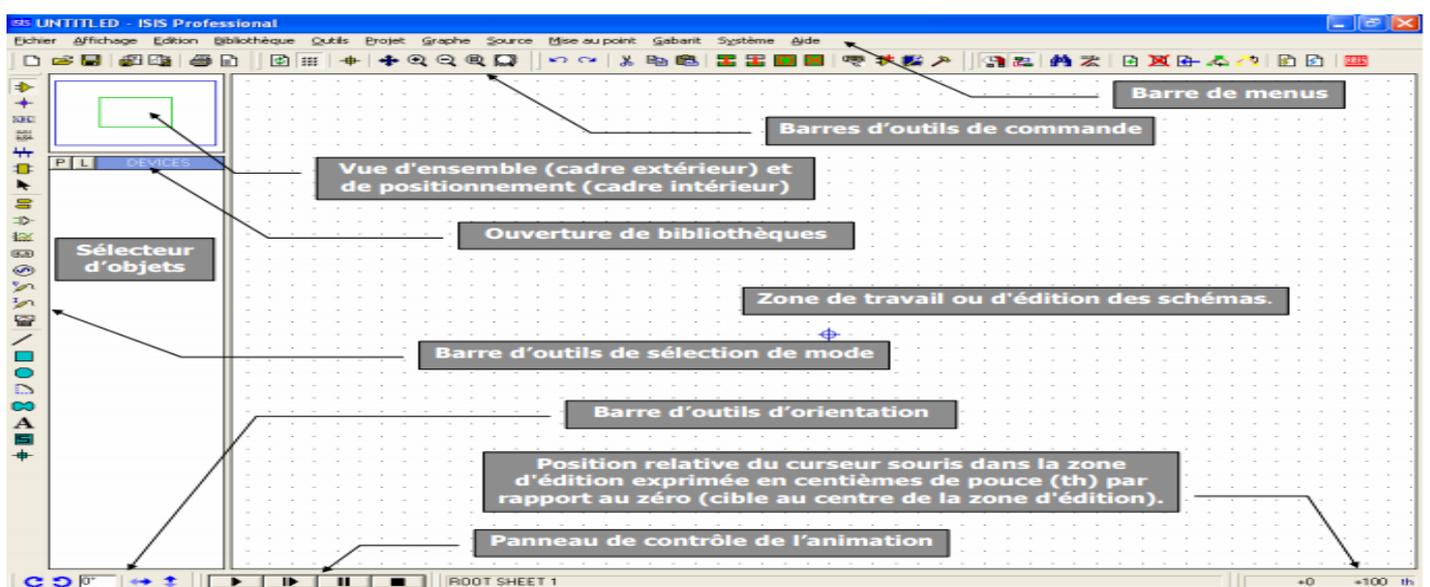


Figure (2.10) : L'interface du Simulateur ISIS Protues

## **2.5 Conclusion**

Dans ce chapitre nous avons illustré l'importance du microcontrôleur et le différent rôle qu'il peut occuper dans un système en présentant ces nombreuses caractéristiques, le bon fonctionnement de cette unité de contrôle nécessite qu'il soit programmé et adapté à un compilateur de programmation.

Dans notre projet de réalisation numérique d'une maquette d'ascenseur par Arduino nous avons utilisé l'ATmega 2560. Dans le chapitre suivant nous présenterons une maquette réduite d'un ascenseur.

### 3.1 Introduction

Après avoir donné une description théorique sur le module Arduino et son environnement de développement dans le chapitre précédent, on va maintenant, procéder à l'application expérimentale, notre projet est une maquette qui permet de simuler le fonctionnement réel d'un ascenseur à 4 niveaux (R+3), et travail à base de la carte Arduino Mega 2560.

Avant de passer à la réalisation pratique, nous avons utilisé un CAO (conception assisté par ordinateur): il s'agit d'ISIS proteus, c'est un CAO électronique conçu par "Labcenter Electroniques" qui permet de représenter des schémas électroniques, de les simuler et de réaliser le circuit imprimé correspondant. Le CAO électronique «PROTEUS » disponible et téléchargeable, se compose de nombreux outils regroupés en modules au sein d'une interface unique. Ce dernier, nous permet de schématiser notre maquette comme le montre les figures (3.1) et (3.2).

Notre projet qui est basé sur la réalisation d'une maquette, s'articule sur deux grandes parties essentielles:

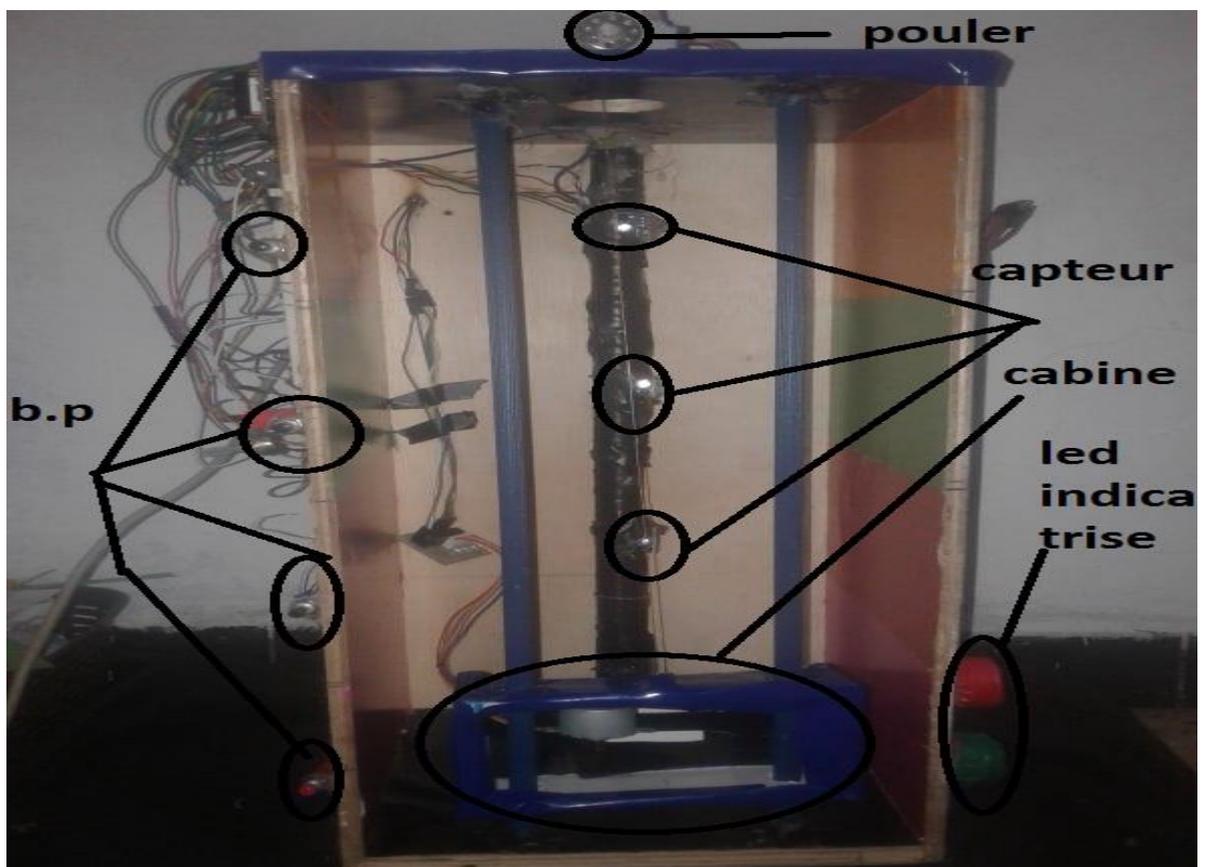


Figure (3.1) : maquette d'ascenseur à 4 étages

### 3.2 Partie 1: Cotés matérielles

#### 3.2.1 Structure et principe de fonctionnement

Le fonctionnement de notre ascenseur est basé sur le principe de commander un moteur pas à pas à partir d'une carte Arduino. avec le monde extérieur Le schéma de l'ascenseur que nous avons réalisés est indiqué dans la figure ( 3.2). Notre circuit est constitué de quatre boutons poussoirs qui permettent de commander le sens de déplacement (le sens de rotation du moteur) de la cabine en haut ou en bas selon notre l'ordre donner, un groupe de boutons représentant le pupitre de commande à l'intérieur de la cage d'ascenseur permet de :

- Sélectionner l'étage désiré.
- Déclencher l'alarme.
- Visualiser le niveau réel de l'ascenseur.

Un afficheur LCD est utilisé pour indiquer à quel étage se trouve la cabine de l'ascenseur. Comme la carte Arduino délivre des signaux (tensions) de faible courant qui ne sont pas capables de faire tourner le moteur, c'est pourquoi on utilise le module interface du moteur qui est basé sur le circuit ULN2003 qui est un amplificateur de courant de type d'Arlington.

#### 3.2.2 Structure des composants de système



Figure (3.2) : Schéma synoptique de l'ascenseur

#### 3.2.3 Cahier de charge

Généralement dans un ascenseur, on trouve les actions suivantes :

- Monté
- Descendre

- S'arrêté

### 3.2.4 Description des composants du circuit

Notre ascenseur est constitué essentiellement des modules suivants :

- D'une carte arduino mega 2560
- Capteur infrarouge émetteur récepteur
- Ecran LCD
- Moteur pas à pas
- Clavier
- Boutons poussoirs
- Driver uln2003

#### 3.2.4.1 Moteur pas à pas

Dans notre projet en va travailler avec un moteur pas a pas de type unipolaire, Le moteur pas-à-pas unipolaire à 4 enroulements avec réducteur et driver basé sur un ULN2003A permettant de contrôler directement le moteur avec un microcontrôleur (ex: Arduino) sans interface de puissance. Pour applications diverses telles que Equipement de bureau, robotique, automatique etc.

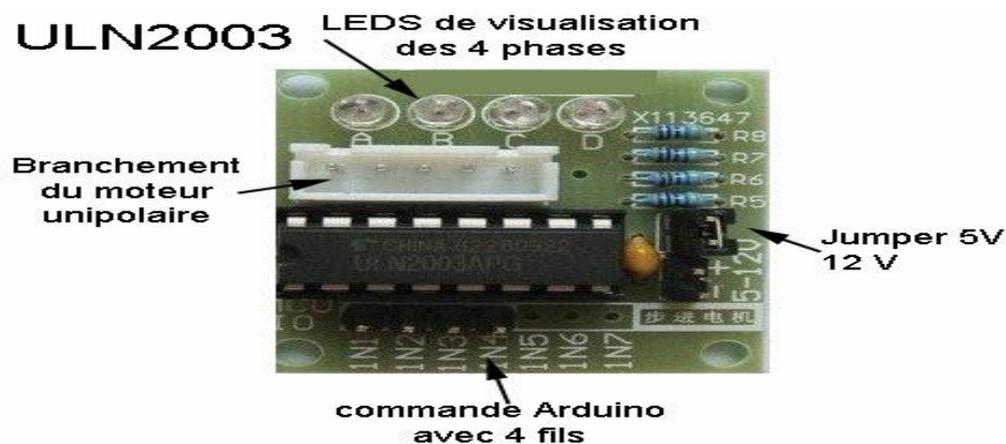


Figure (3.3) : Pilotage avec une carte Arduino d'un moteur pas à pas unipolaire avec un driver ULN2003

#### 3.2.4.1.1 Principe de fonctionnement:

Le moteur pas à pas à 4 phases réducté (1:64) peut être positionné sur une valeur angulaire précise. Il possède un couple important permettant de l'utiliser dans la

plupart des petits montages. Son driver ULN2003 (réseau de transistors Darlington) lui apporte la puissance que la carte Arduino ne peut pas fournir directement.



Figure (3.4) : Moteur pas à pas 28bj

#### 3.2.4.1.2 Domaine d'application :

Ce type de moteur peut être utilisé dans plusieurs Domaines tel que :

Petits robots, mécanismes divers, maquette d'ascenseur, modélisme ferroviaire, jouets, machines, ...etc.

#### 3.2.4.2 Capteur mouvement infrarouge

Description du capteur émetteur récepteur

Les capteurs infrarouges émetteurs récepteurs sont des capteurs réfléchissants qui comprennent un émetteur infrarouge et un phototransistor dans un emballage au plomb qui bloque la lumière visible.

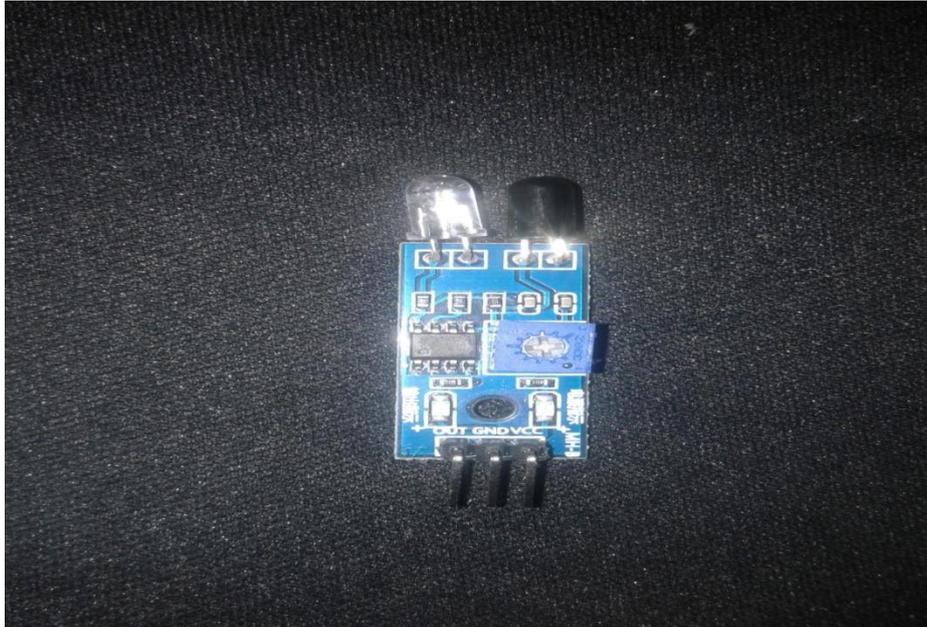


Figure (3.5) : Capteur mouvement infrarouge.

#### 3.2.4.2.1 Domaines d'application

Les domaines d'applications capteur mouvement infrarouge sont entre autre :

- Eclairage invisible pour camera.
- Système autofocus d'appareils photo et de caméras commutation optique.
- Commandes automatiques de portes dans l'industrie automobiles.
- Techniques de protection (alarmes, détecteurs de fumée, interrupteurs à infrarouge).
- Mesures de niveau
- Détecteurs de position des aiguilles des horloges radiocommandées
- Les systèmes de transmission à infrarouge ont beaucoup d'atouts mais également des inconvénients qu'il est bon de peser avant de s'engager.

#### 3.2.4.3Ecran LCD

##### 3.2.4.3.1 Rôle d'un afficheur LCD

Les afficheurs LCD sont devenus indispensables dans les systèmes électroniques qui nécessitent l'affichage des valeurs et des instructions pour le bon déroulement de leurs fonctionnements.

Grâce à la commande par un microcontrôleur ces afficheurs permettent de réaliser un affichage de messages aisés. Ils permettent également de créer ses propres caractères [18].

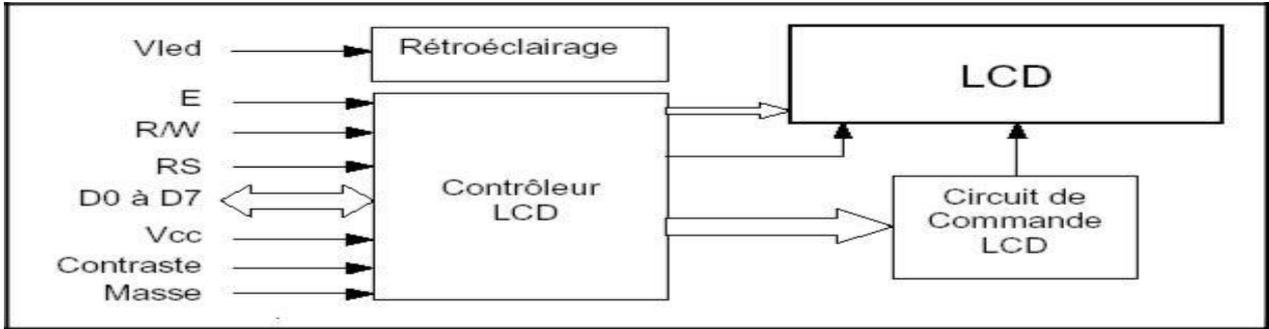


Figure (3. 6) : Schéma fonctionnel d'un LCD

### 3.2.4.4 Description d'un clavier 4x4

Les lignes sont des sorties. Les colonnes sont des entrées maintenues au niveau haut par une résistance interne à Arduino. Le système envoie par balayage un niveau bas sur chaque ligne (1 seule à la fois) et balaye les colonnes en lecture. Quand il lit un niveau bas, c'est que la colonne est reliée par une touche appuyée à la ligne qui est basse à ce moment. On n'a pas besoin de programmer ce balayage : c'est la bibliothèque qui le fait.

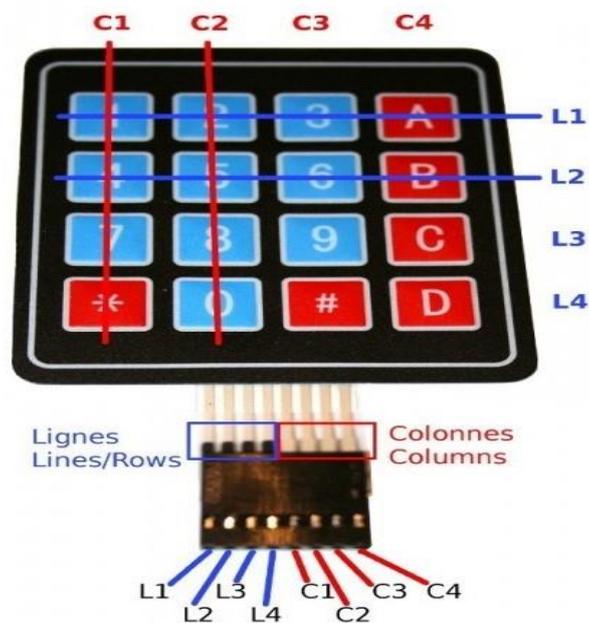


Figure (3.7) : Clavier 4x4

### 3.3 Partie 2: logicielles

Cette section s'intéresse à la partie programmation proprement dit. Elle sert à décrire les différents programmes adaptés et développés avec nous mêmes pour réaliser ce projet. Dans cette partie, nous avons utilisés l'environnement Arduino pour synthétiser des programmes, afin de bien gérer la communication entre la carte Arduino et les différents éléments électroniques qui composent notre maquette conçues.

Dans cette partie, nous allons présentés les différents programmes que nous avons réalisés, par arduino sous forme des organigrammes.

Il est possible de programmer le comportement de la carte Arduino de deux manières différentes : [9]

```
void setup()
{
  pinMode(1, INPUT);
  pinMode(12, OUTPUT);
}
void loop()
{
  if (digitalRead(1)) {
    digitalWrite(12, HIGH);
    delay(1000);
    digitalWrite(12, LOW);
    delay(1000);
  } else {
    digitalWrite(12, LOW);
  }
}
```

Programmation en langage C



Programmation par blocs

Figure (3.8) : Compilateur arduino

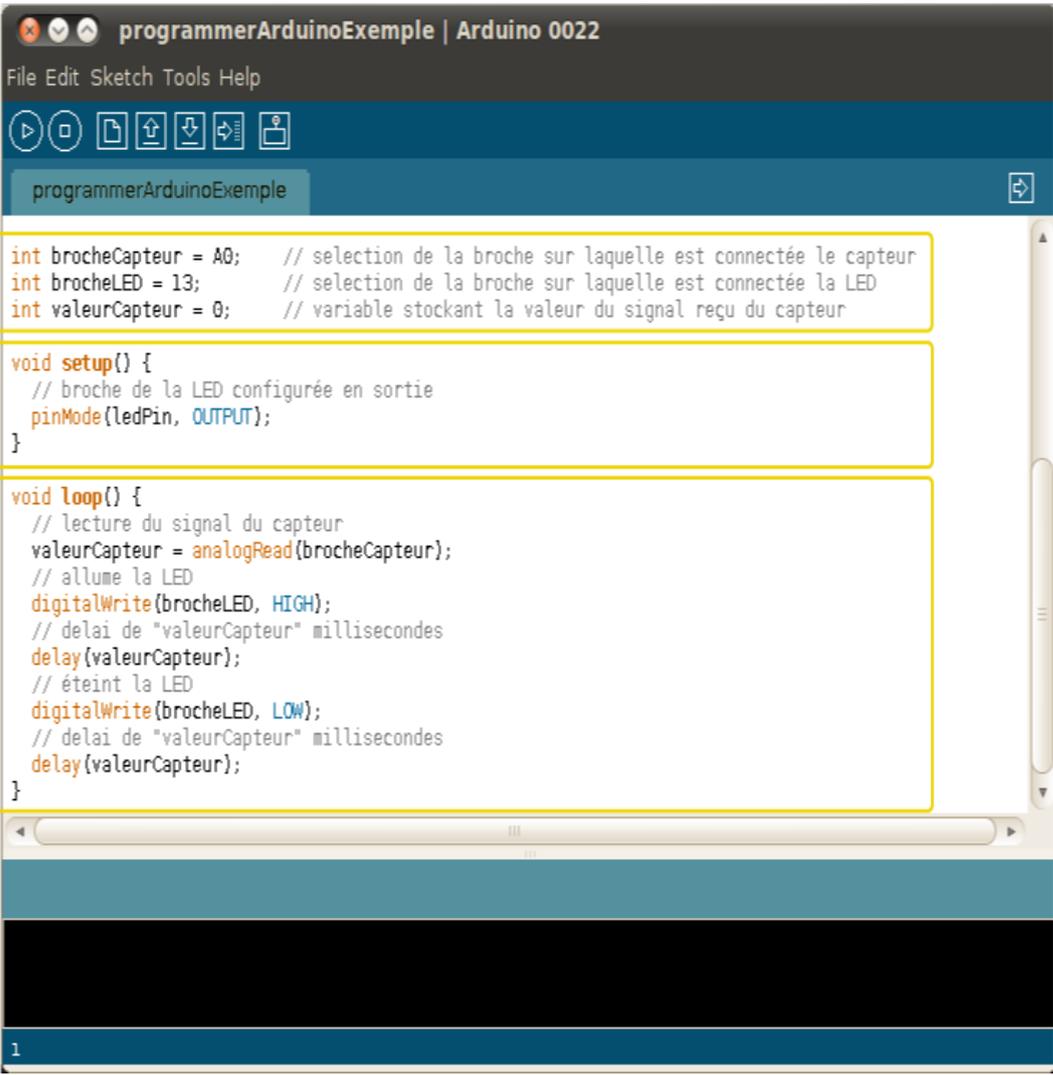
### 3.3.1 Structure d'un programme arduino

#### Un programme Arduino peut être structuré en 4 parties :

Avant cela il faut connaître et déclarer les bibliothèques nécessaires pour notre projet.

#### Partie 1 : commentaires :

Cette première partie définit le programme. Il doit donner un résumé du fonctionnement du programme. Tout commentaire doit être précédé et se terminer par des signes spéciaux.



```
int brocheCapteur = A0; // selection de la broche sur laquelle est connectée le capteur
int brocheLED = 13; // selection de la broche sur laquelle est connectée la LED
int valeurCapteur = 0; // variable stockant la valeur du signal reçu du capteur

void setup() {
  // broche de la LED configurée en sortie
  pinMode(ledPin, OUTPUT);
}

void loop() {
  // lecture du signal du capteur
  valeurCapteur = analogRead(brocheCapteur);
  // allume la LED
  digitalWrite(brocheLED, HIGH);
  // delai de "valeurCapteur" millisecondes
  delay(valeurCapteur);
  // éteint la LED
  digitalWrite(brocheLED, LOW);
  // delai de "valeurCapteur" millisecondes
  delay(valeurCapteur);
}
```

Figure (3.9) structure programme

#### Partie 2 : Définition des variables :

Dans cette partie, nous devons définir quelles variables nous allons utiliser dans notre programme, ainsi que leur nature (binaire, entier, chaîne de caractères, ...).

```
void setup() {  
  // put your setup code here, to run once:  
  Serial.begin(115200);  
}  
  
void loop() {  
  // put your main code here, to run repeatedly:  
  
  int data[50][6];  
  int data_max = 0;  
  int location;  
  pinMode(7, OUTPUT);  
  pinMode(2, OUTPUT);  
  pinMode(3, OUTPUT);  
  pinMode(4, OUTPUT);  
  pinMode(5, OUTPUT);  
  pinMode(6, OUTPUT);  
}
```



Figure (3.10) : Partie déclaration des variables

### Partie 3 : Configuration des entrées/sorties :

Dans cette partie nous définirons la nature des entrées/sorties matérielles utilisées (si telle ou telle broche est utilisé en entrée ou en sortie, de nature numérique (binaire : 0 ou 1) ou analogique. Cette phase doit être précédée par la commande "*void setup ()* " suivie d'une accolade "{" et se terminer d'une accolade "}".

### Partie 4 : Programmation des interactions :

Cette partie définit ce que doit faire votre programme, elle doit être précédée de la commande "*void loop()*", d'une accolade "{" et se terminer d'une accolade "}".

Un programme utilisateur Arduino est une suite d'instructions élémentaires sous forme textuelle, ligne par ligne.

La carte lit puis effectue les instructions les unes après les autres, dans l'ordre défini par les lignes de code.

Une fois la dernière ligne exécutée, la carte revient au début de la quatrième phase et recommence sa lecture et son exécution, Et ainsi de suite. Cette boucle se déroule plusieurs fois par seconde selon le nombre indiqué par le programme de fois et anime la carte [12].

3.3.2 Logiciel Isis PROTEUS

Isis Proteus est un logiciel de développement et de simulation d'application via un environnement graphique simple et interactif, dont les résultats doivent être comparés par la suite a ceux obtenus théoriquement et expérimentalement.

3.3.3 Schéma électrique

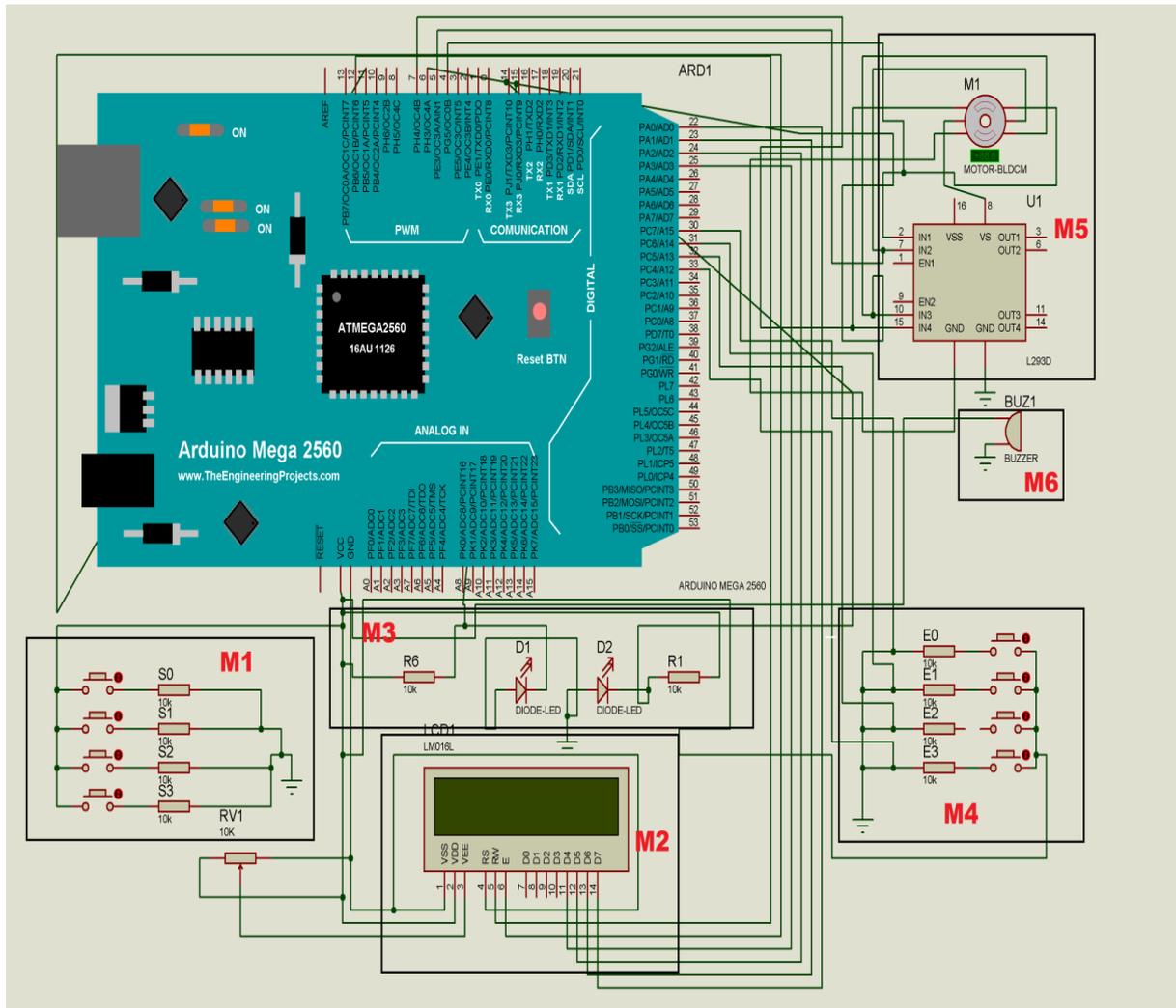


Figure (3. 11) : Carte de simulation de l’ascenseur

Module 1 :

Bouton poussoir situer a l’extérieure de la cabine.

Module 2 :

Ecran LCD qui nous indique le placement de la cabine d’ascenseur.

Module 3 :

Des LED indicateur

Module 4 :

Bouton poussoir situé à l'intérieur de la cabine

Module 5 :

Moteur pas à pas avec un driver pour commander l'action de la cabine (montée, descente)

Module 6 :

Indicateur sonore de l'ouverture et la fermeture de la porte

### 3.3.4 Les organigrammes

#### 3.3.4.1 Organigramme de détection de la position :

Pour détecter l'arrivée de l'ascenseur à l'étage nous proposons des détecteurs de mouvement. La présence de la cabine d'ascenseur est liée à l'activation et désactivation du capteur qui génère un niveau logique haut vers la carte de l'Arduino. Pour confirmer l'arrivée de l'ascenseur à l'étage il faut placer un capteur pour chaque étage.

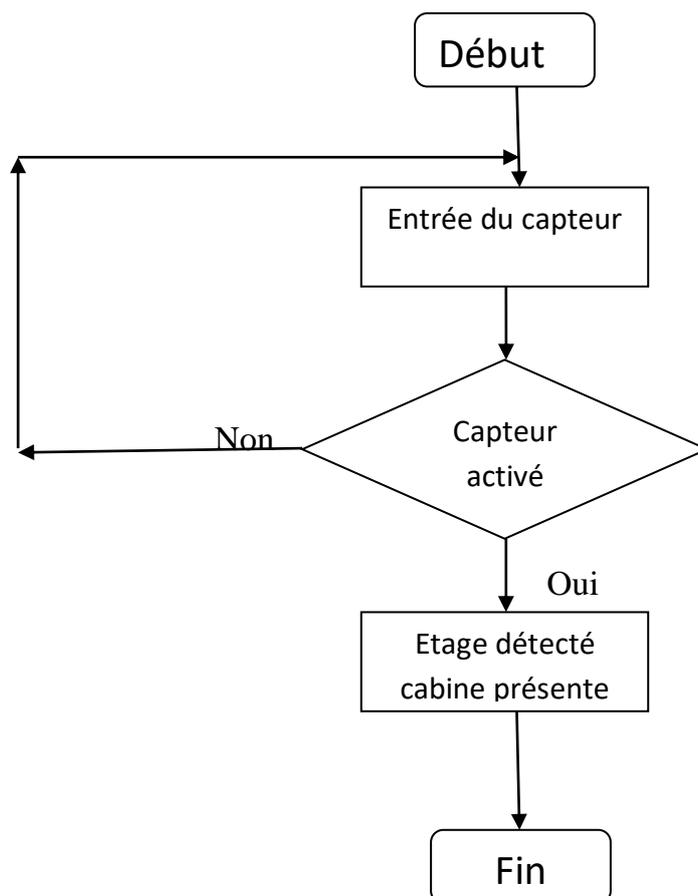
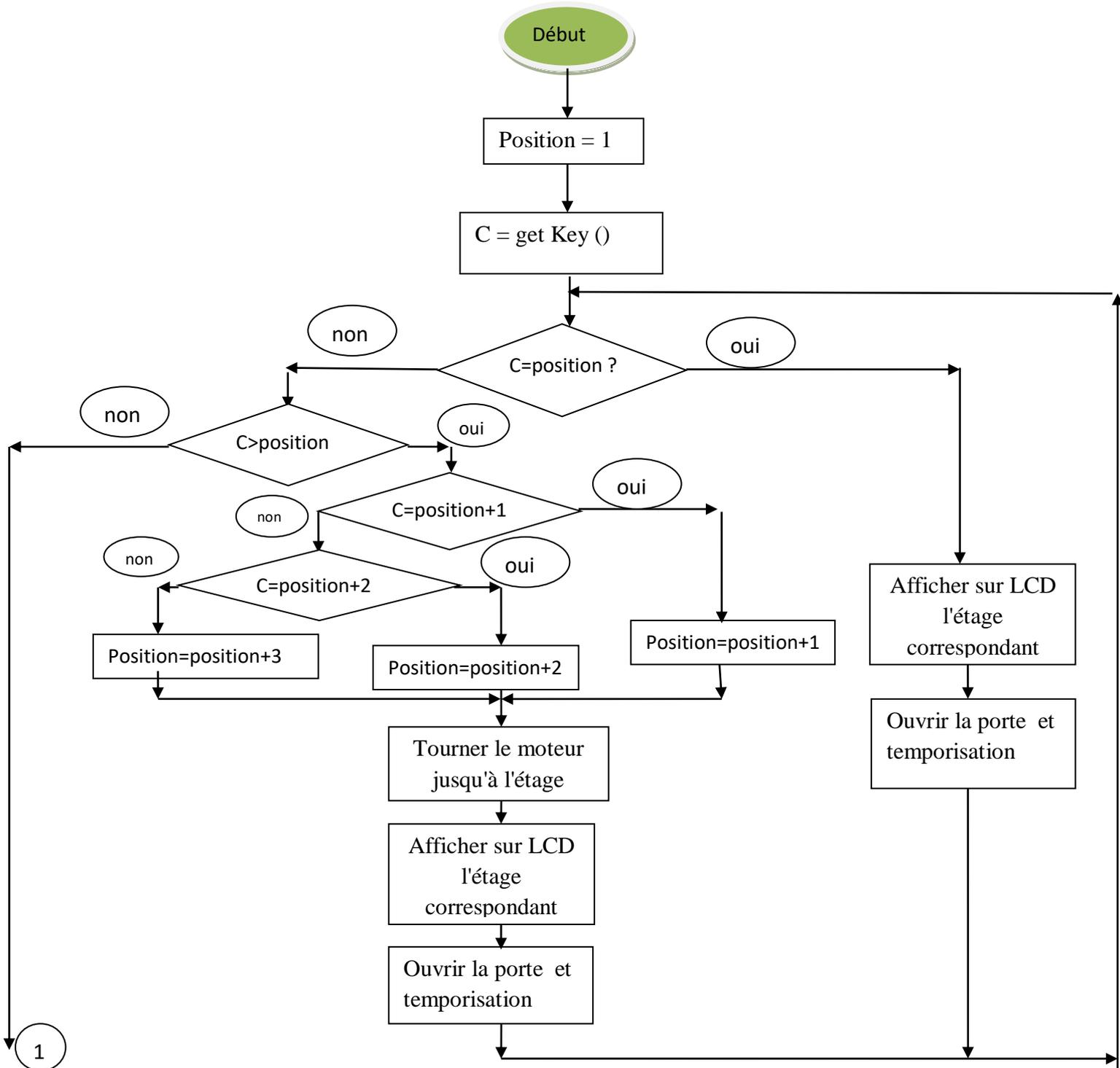


Figure (3.12) : Organigramme détection du position

3.3.4.2 Organigramme principal de fonctionnement

Après avoir introduit les différents mécanismes de travail dans le système, ainsi que les fonctions partielles du système, nous atteignons un niveau qui nous permet d'assembler l'ensemble des organigrammes dans une structure intégrée, afin de mieux représenter le mécanisme de fonctionnement du système (comme on a 4 étage donc  $c = 1, 2, 3, 4$  seulement).

L'organigramme décrivant le fonctionnement de notre ascenseur est le suivant:



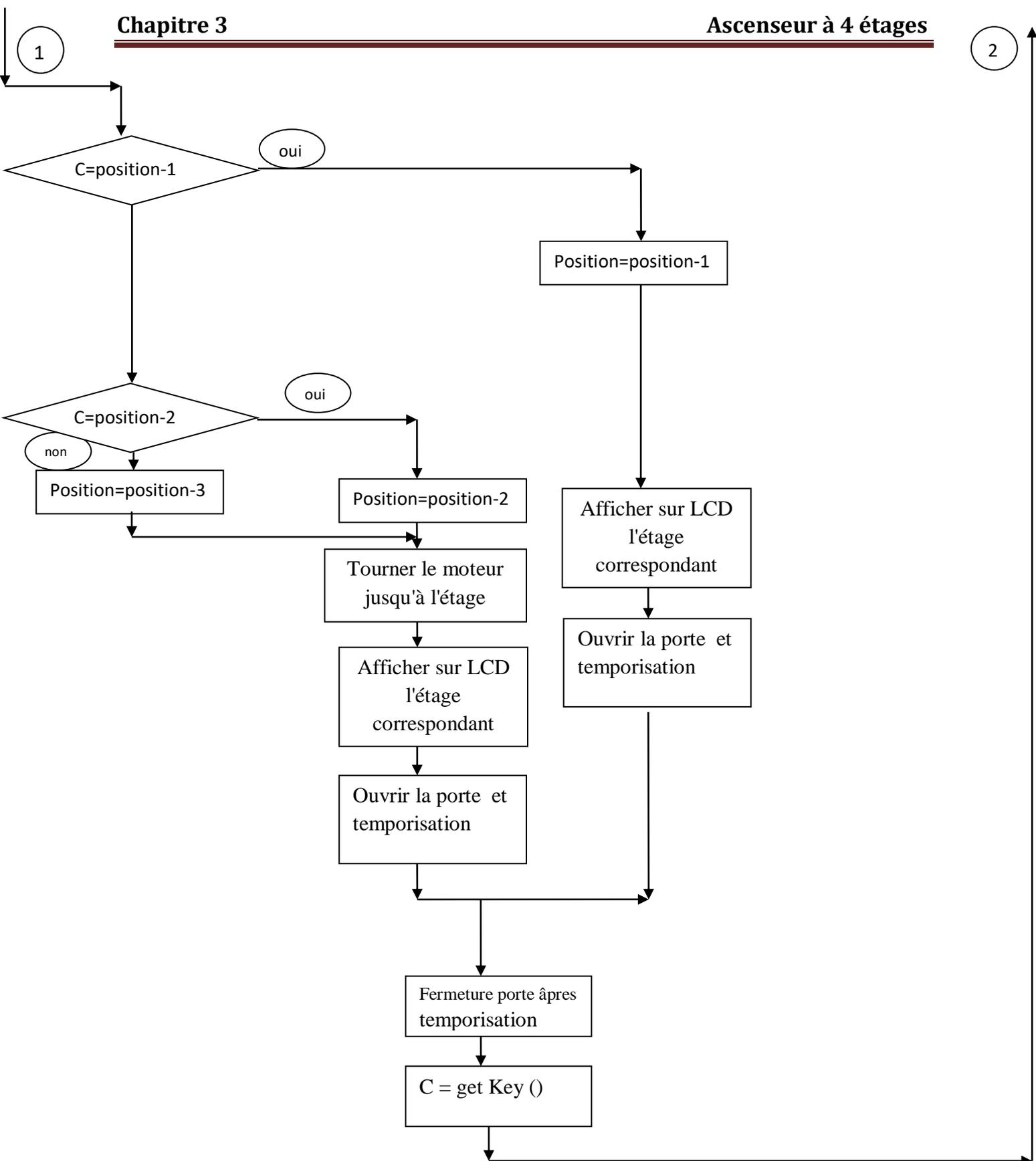


Figure (3.13) : Organigramme Principe de fonctionnement du projet réalisé.

**3.5 Conclusion**

- Dans ce projet nous avons pu faire une maquette d'ascenseur a 4 étage. Il est à noter que nous avons rencontré quelques problèmes, tels que :
  - Problème de matériel défectueux.
  - Conditionnement de batterie.
  - Problème de chute de tension produite par la carte de commande.

## Conclusion générale

---

### Conclusion générale

Le travail présenté dans ce manuscrit dont l'intitulé est "Etude et réalisation d'une maquette d'ascenseur par arduino", nous avons fait une étude sur les origines de l'ascenseur et les bases de cette science qui nous a permis de penser à l'architecture que doit prendre notre ascenseur et aux différents éléments indispensables à sa réalisation. Le fonctionnement de notre ascenseur est basé sur le principe de commander un moteur pas à pas à partir d'une carte Arduino.

Le monde Arduino est vraiment intéressant ; la facilité de son langage de programmation simplifié et la diversité des composants utilisés, nous donne un espace de création plus vaste avec une réalisation souple et confortable.

## Bibliographie

---

- [1] <https://www.larousse.com> consulté le 22/04/2019
- [2] futura tech <https://www.futura-sciences.com/tech/definitions/technologie-ascenseur-11102/> consulté le 22/04/2019
- [3] <https://www.energieplus-lesite.be/index.php?id=11532#c1506+c1507+c1508> consulté le 24/04/2019
- [4].extrait d'un livre, The Vertical transportation handbook par Georges R.Strakosch.2016 consulté le 22/04/2019
- [5][https://www.sdis14.fr/IMG/pdf/TOP\\_FEX\\_A3\\_Degagement\\_de\\_personne\\_bloquee\\_dans\\_un\\_ascenseur.pdf](https://www.sdis14.fr/IMG/pdf/TOP_FEX_A3_Degagement_de_personne_bloquee_dans_un_ascenseur.pdf) consulté le 25/04/2019
- [6]Jimmy Keyrouz. Ascenseur sans local de machinerie. Mécanique des structures [physics.class-ph]. 2011.<dumas-01081520>consulté le 30/04/2019
- [7] <http://ascenseurconseilplus.free.fr/> consulté le 01/05/2019
- [8]<http://www.sti2dsinhyrome.fr/cours%20et%20tp%20sin/fiche%20logiciel%20ARDUINO%20M%C3%A9ga.pdf>
- [9]/ <http://www.techmania.fr/> consulté le 05/05/2019
- [10]<http://tperobotiqueexosquelette.emonsite.com/pages/tpe/fonctionnement/micro-controleur-arduino.html>
- [11] <https://www.electronique-mixte.fr/kits-de-developpement-processeurs-et-microcontrolleurs/kit-arduino/>
- [12]<http://www.sti2dsinhyrome.fr/cours%20et%20tp%20sin/fiche%20logiciel%20ARDUINO%20M%C3%A9ga.pdf>
- [13] <http://site.uclouvain.be>
- [14] Patrick ABATI , Moteur pas à pas, 07/01/2000 , page 01 [14]
- [15]<http://dspace.univtlemcen.dz/bitstream/112/11743/1/Ms.ELN.Chigango%20BMapuranga.pdf>
- [16] <http://paristizi.free.fr/projet/rapport.pdf>
- [17]<https://disciplines.actoulouse.fr/sii/sites/sii/files/ressources/didacticiels/programmation/arduino/ardublock/didacticiel-evitement-obstacle-ir.pdf>G. Ruscassié - Ac. Toulouse – Juillet 2016

## Bibliographie

---

- [18]<http://www4.ac-nancy-metz.fr/lyc-vuillaume-mirecourt/pages/Pedagogie/DATA/Technique/Electronique/Afficheur%20LCD/Afficheur%20LCD.pdf>
- [19] <http://colmard.com/Arduino-lecon25.html> 28/05/2019
- [20][https://www.avh.asso.fr/sites/default/files/fiche\\_ascenseurs.pdf](https://www.avh.asso.fr/sites/default/files/fiche_ascenseurs.pdf) 28/05/2019
- [21]<http://www.ascenseurs.fr/wpcontent/uploads/2016/09/Descriptionetprincipesdefonctionnementcomplet.pdf> 28/05/2019
- [23] ISSAM KHELIL Commande d'un ascenseur par Arduino,mémoire fin d'étude master UNIVERSITE BADJI MOKHTAR ANNABA Année : 2018 consulté le 28/05/2019

## Annexe

---

### Caractéristiques techniques du Moteur pas à pas

#### Caractéristiques:

Alimentation : 5 V

Résistance: 50 ohms

Intensité: 25 mA

Réduction: 1/64

Nombre de pas: 64 (réduction de 4096 en sortie d'axe)

Couple: 340 gf.cm

Niveau sonore: 40 dB.

Entraxe de fixation: 35 mm

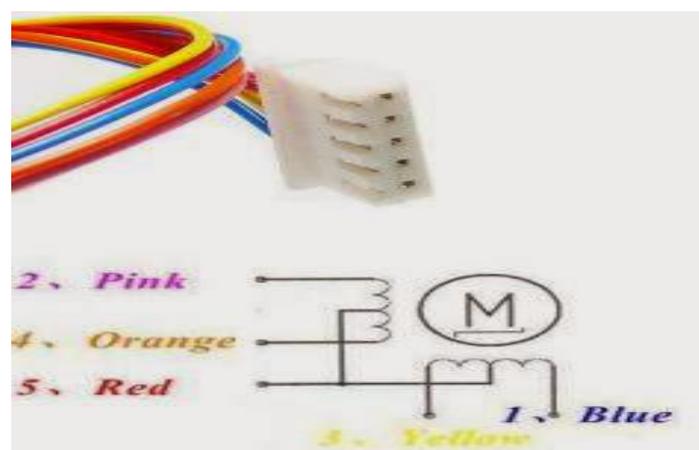
Axe: Ø5 mm avec double méplat (épaisseur 3 mm)

Longueur de l'axe: 8 mm

Le moteur se branche sur le driver ULN2003 avec son fil (fourni), muni d'un connecteur à 5 broches (avec détrompeur de sens de branchement).

#### Brochage

- Rouge = Commun
- Orange = Pin1
- Jaune = Pin2
- Pourpre = Pin3                      avec photo
- Bleu = Pin4



## Annexe

---

Le driver ULN2003 est relié suite d'application

à une alimentation de 5V DC pour le moteur (l'ULN supporte jusqu'à 30V mais pas les 4 LEDS de la carte ni le moteur DC) : +V et GND.

- à la carte Arduino sur 4 pins digitales (pin 4- pin 7), configurées en sorties, avec 4 fils Dupont mâle / femelle.

In1 -> pin arduino

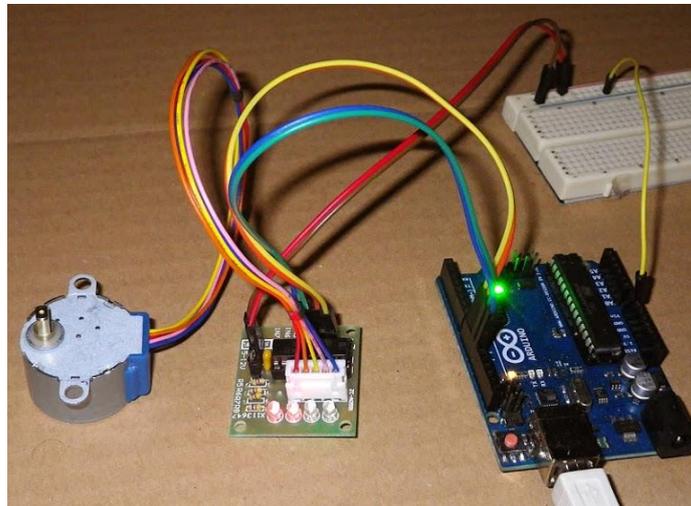
In2 -> pin arduino

In3 -> pin arduino

In4 -> pin arduino

- Relier les masses de l'Arduino et de celle de l'alimentation séparée du moteur.

La position du jumper dépend de la source de tension utilisée, en position 3-4 avec une alimentation externe.



### Utilisation du driver avec d'autres moteurs

Le driver peut s'utiliser avec d'autres moteurs unipolaires. Dans ce cas, repérer les 2 bobines avec un Ohmmètre en mesurant la résistance entre les 5 fils pris 2 à 2. Le fil qui avec les autres présente la résistance maximale est le commun, c'est la broche 1 du connecteur.