



République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche
Scientifique
Université Akli Mohand Oulhadj – Bouira



Faculté des sciences et des sciences appliquées
Département : Génie électrique.

Mémoire de fin d'étude

Présenté par :

AKKOUCHE Fani

En vue de l'obtention du diplôme de Master en :

Filière : Electronique

Spécialité : Electronique des Systèmes Embarqués

Intitulé du sujet :

Réalisation d'un ascenseur à base d'Arduino

Date de soutenance : 06/11/2021

Devant le jury composé de

Présidente	Mm. AGGOUN Ghania	UAMOB
Examineur	Mr. GRICHE Issam	UAMOB
Encadreur	Mr. AIT ABBAS Hamou	UAMOB

Année Universitaire 2020-2021

Remerciement

Nous tenons à remercier notre Dieu, le tout puissant, de nous avoir donné la santé et la volonté pour compléter ce modeste travail.

Nos remerciements les plus chaleureux vont à nos **chers parents** pour leurs encouragements, leur patience, et leur grand soutien durant toutes ces années d'études.

Nos remerciements s'adressent également au docteur **AIT ABBAS.H**, pour son encadrement, sa motivation professionnelle, son orientation afin de donner notre mieux.

Nos profondes gratitudeux aux membres du jury qui ont l'honneur d'évaluer ce travail de fin d'études.

Tous nos vifs remerciements vont à tous les enseignants qui ont contribué à notre formation durant notre cursus universitaire en particulier, monsieur **ISSAOUNI. S, AIT ABBAS. H, SAOUD. B, MOUDACHE. S, MEDJEDOUB. S.**

Enfin, nous tenons à remercier également tous ceux qui nous ont aidé de près ou de loin pour l'élaboration de ce mémoire.

A tous ceux dont le soutien nous a été utile et nécessaire

En particulier monsieur **BOUACHERINE.Y.**

Dédicace

Ce mémoire représente bien plus que de simples travaux. Ce mémoire est la finalité de cinq longues années d'études. C'est pourquoi je tiens à exprimer ma reconnaissance et ma sympathie bien que je dédie ce travail :

A mes chers parents, mes piliers, que dieu me les gardes

Que ce mémoire soit le meilleur cadeau que je puisse vous offrir

A mes frères et mes sœurs que j'adore

KHALED, OUAZNA, SILIA, DIANA, LINA, RAYAN

A celles que je considère comme des sœurs

ASSIA, MARINA, SARA, NADIA, OUAZNA

Qui m'avais toujours soutenu

Et encouragé durant ces années d'études

Aux membres de ma grande famille

A tous ceux que j'aime et ceux qui m'aiment

...

FANI

Liste Des Figures

Figure I.1 : Structure d'un système automatisé.....	3
Figure I.2 : élévateur Otis avec moteur à vapeur sortie 1862.....	4
Figure I.3 : Ascenseur de la tour Eiffel.....	5
Figure I.4 : Schéma en bloc d'un ascenseur automatisé.....	6
Figure I.5 : Les familles d'ascenseurs.....	8
Figure I.6 : Schéma illustratif du principe de fonctionnement des ascenseurs hydrauliques.....	9
Figure I.7 : Les différents modèles de l'ascenseur hydraulique.....	9
Figure I.8 : Les types d'ascenseur à traction à câble.....	11
Figure I.9 : Les composants d'un ascenseur à traction électrique.....	12
Figure II.1 : Le microcontrôleur de la carte Arduino.....	19
Figure II.2 : Les 14 entrées sorties numériques.....	19
Figure II.3 : Les 6 entrées sorties analogiques.....	20
Figure II.4 : Le schéma électrique de l'alimentation de la carte Arduino.....	20
Figure II.5 : Les composants de la carte Arduino (Uno).....	21
Figure II.6 : L'interface de l'IDE Arduino.....	23
Figure II.7 : Le menu fichier.....	24
Figure II.8 : Le Menu outils.....	24
Figure II.9 : La barre d'outils.....	25
Figure II.10 : Exemple d'un programme Arduino.....	26
Figure III.1 : Carte Arduino Mega 2560.....	29
Figure.III.2 : Moteur à courant continu.....	29
Figure.III.3 : Stator d'un moteur à courant continu.....	30
Figure.III.4 : Rotor d'un moteur à courant continu.....	30
Figure III.5 : Le module L298N.....	31
Figure III.6 : Pont en H : les 2 configurations possibles.....	32
Figure III.7 : Afficheur 7 segments.....	33
Figure III.8 : Afficheur 7 segments identifier avec des lettres.....	33
Figure III.9 : Types d'afficheur 7 segments.....	34
Figure III.10 : La visualisation des chiffres.....	34

Figure III.11 : Types des capteurs de présence.....	35
Figure III.12 : -A- capteur de présence ; B- placement des capteurs.....	35
Figure III.13 : les ondes IR (infrarouge).....	36
Figure III.14 : Bouton-poussoir.....	37
Figure III.15 : Fenêtre du module de simulation Proteus ISIS.....	39
Figure IV.1 : Schéma synoptique de la carte de commande.....	40
Figure IV.2 : Schéma de conception de circuit d'affichage sur ISIS.....	41
Figure IV.3 : circuit d'affichage	41
Figure IV.4 : Schéma de conception de circuit de puissance sur ISIS.....	42
Figure IV.5 : Branchement de capteur infrarouge.....	42
Figure IV.6 : Branchement des boutons poussoirs et leds.....	43
Figure IV.7 : Schéma de la carte de commande.....	43
Figure IV.8 : Programme principale.....	44
Figure IV.9 : Sous-programme "vérification boutons "	45
Figure IV.10 : Sous-programme " déplacement moteur.....	46
Figure IV.11 : Sous-programme : détection de la position.....	47
Figure IV.12 : Réalisation final.....	48

SOMMAIRE

Introduction générale.....	01
----------------------------	----

CHAPITRE I : Systèmes Automatisés et Généralité sur les Ascenseurs.

I.1. Introduction.....	02
I.2. Définition du système.....	02
I.3. Les systèmes automatisés.....	02
I.3.1. Définition d'un système automatisé.....	02
I.3.2. Structure d'un système automatisé.....	02
I.3.3. Le fonctionnement.....	03
I.4. Historique de l'ascenseur.....	04
I.5. Les ascenseurs.....	06
I.5.1. Présentation de l'ascenseur.....	06
I.5.2. Approche Ascenseur-Système Automatisé.....	06
I.5.3. Différent types d'ascenseur.....	07
I.5.4. Les familles d'ascenseur.....	08
I.5.4.1. Les ascenseurs hydrauliques.....	09
I.5.4.2. Les ascenseurs à traction à câble.....	11
I.5.5. Différentes parties d'un ascenseur à traction.....	12
I.5.5.1. Système de motorisation d'un ascenseur à traction.....	15
I.6. Conclusion.....	16

CHAPITRE II : Etude de la carte et logiciel de programmation.

II.1. Introduction.....	17
II.2. La carte Arduino.....	17
II.2.1. Définition.....	17
II.2.2. Les gammes de la carte Arduino.....	17
II.2.3. Les composants de la carte.....	18
II.3. Le logiciel Arduino IDE.....	22
II.3.1. Définition de logiciel.....	22
II.3.2. Présentation du logiciel.....	22
II.3.3. Approche et utilisation de logiciel.....	23
II.3.4. Structure d'un programme.....	25

SOMMAIRE

II.4. Conclusion.....	27
-----------------------	----

CHAPITRE III : Etude et Conception d'un Ascenseur.

III.1. Introduction.....	28
III.2. Partie électronique.....	28
III.2.1. Description des composants.....	28
III.2.1.1. Le microcontrôleur (carte Arduino Mega 2560).....	28
III.2.1.2. Le moteur à courant continu (MCC).....	29
III.2.1.3. Le module L298N.....	31
III.2.1.4. Afficheur 7 segments.....	33
III.2.1.5. Les capteurs de position.....	35
III.2.1.6. Bouton poussoir.....	37
III.3. Partie mécanique.....	38
III.3.1. Présentation de prototype.....	38
III.4. Partie logicielle.....	38
III.4.1. Présentation de Proteus.....	38
III.4.1.1. ISIS.....	39
III.5. Conclusion.....	39

CHAPITRE IV : Réalisation et Validation.

IV.1. Introduction.....	40
IV.2. Principe de fonctionnement.....	40
IV.3. Conception de la carte.....	40
IV.3.1. Circuit d'affichage.....	41
IV.3.2. Circuit de puissance.....	42
IV.3.3. branchement de capteur infrarouge.....	42
IV.3.4. Branchement des boutons poussoirs leds.....	43
IV.3.5. Schéma de la carte de commande.....	43
IV.4. Programmation.....	44
IV.4.1. Réalisation du programme.....	44
IV.4.1.1. Organigramme de déroulement du programme.....	44
IV.5. Description de la maquette.....	47
IV.5.1. La partie externe.....	47
IV.5.2. La partie interne.....	47

SOMMAIRE

IV.6. Conclusion.....	48
Conclusion générale.....	49

Introduction

Générale

Introduction générale

La croissance démographique a imposé à l'urbanisme de développer les méthodes nouvelles pour adapter l'habitat urbain aux besoins de la population qui fait face aux problèmes de densité et donc du manque d'espace. Parmi ces méthodes, nous notons la plus imminente qui est le développement de la construction des immeubles de très grande hauteur à plusieurs niveaux.

Cependant, la réalisation de cette solution semble être la source de deux grands problèmes :

- ❖ La montée des charges (matériaux de construction) ;
- ❖ L'accès des usagers de toutes natures a des niveaux supérieurs.

Pour y remédier, les architectes ont mis sur pied des dispositifs appropriés permettant de minimiser les pertes d'énergie. Parmi ces dispositifs, on peut citer les monte-charges et les ascenseurs.

Pendant l'utilisation de ces dispositifs la sécurité des usagers doit toujours être assurée. Les constructeurs doivent assurer en même temps le confort et la sécurité des utilisateurs. La tâche la plus difficile dans la réalisation est la gestion des conflits et le contrôle en temps réel.

L'ascenseur est un moyen sur lequel se penchent beaucoup de spécialistes. Chacun d'eux étant soumis aux limites de sa spécialité. Le mécanicien construit la cabine, les guides, et toutes les pièces nécessaires, l'électricien calcule le moteur adapté au démarrage et freinages fréquents, et l'automaticien étudie les commandes.

Dans le cadre du projet de fin d'études en master nous avons concentré ces domaines d'intérêts pour en proposer une maquette d'ascenseur de trois niveaux (R+2) à base d'une carte de commande type Arduino Mega 2560.

Pour mener à terme notre travail, nous avons organisé le mémoire en quatre chapitres :

- Le premier chapitre donne une idée générale sur les ascenseurs et les Systèmes Automatisés.
- Le deuxième chapitre est consacré à l'étude de la carte et logiciel de programmation (IDE Arduino).
- Le troisième chapitre présente la description des composants utilisés et leurs caractéristiques ; ainsi que la présentation de logiciels de simulation Proteus ISIS.
- Le quatrième et dernier chapitre est consacré à la réalisation de la carte de commande d'ascenseur et les différentes étapes de programmation de cette maquette.

On clôture ce mémoire par une conclusion générale.

Chapitre 1

Systemes Automatisés et Généralité sur les Ascenseurs

I.1. Introduction :

Longtemps considéré comme un luxe, l'ascenseur est devenu de nos jours un élément indispensable dans la vie quotidienne.

Les ascenseurs constituent toujours un moyen de transport et d'aide pour l'homme lors de ces déplacements dans de hauts immeubles. Les usagers de ces dispositifs doivent se déplacer en sécurité et dans le confort.

Dans ce chapitre nous présenterons brièvement en premier lieu les différentes parties d'un système automatisé, la deuxième partie est consacrée pour l'historique de l'ascenseur et la définition des différents éléments constituant un ascenseur.

I.2. Définition du système :

Un système est un ensemble d'éléments organisés en interaction mutuel et avec l'environnement en vue d'atteindre un ou plusieurs objectifs définis [3].

I.3. Les systèmes automatisés :

L'automatisation d'un procédé (c'est-à-dire une machine, un ensemble de machines ou plus généralement un équipement industriel) consiste à en assurer la conduite par un dispositif technologique. Le système ainsi conçu peut prendre en compte les situations pour les quelles sa commande a été réalisée. L'intervention d'un opérateur est souvent nécessaire pour assurer un pilotage global du procédé, pour surveiller les installations et prendre en commande manuelle (non automatique) tout ou une partie du système [4].

I.3.1. Définition d'un système automatisé :

Un système automatisé est un moyen d'assurer à une entreprise la compétitivité de ses produits. Il permet d'ajouter une valeur aux produits entrants. Il est composé de plusieurs éléments conçus pour effectuer un ensemble de tâches programmés.

Un système automatisé simplifie, sécurise et rend moins pénibles les tâches de production, les systèmes automatisés peuvent être de nature technique très divers et différentes disciplines se rencontrent : l'automatique, l'informatique, la mécanique, l'électronique, l'électrotechnique. Plus le système est complexe, plus les disciplines concernées sont nombreuses [3], [4].

Exemples : porte d'un magasin ; ascenseur ; feux de croisement ; affichage lumineux...etc.

I.3.2. Structure d'un système automatisé :

L'analyse structurelle conduit à décomposer tout système automatisé comme suit [3]:

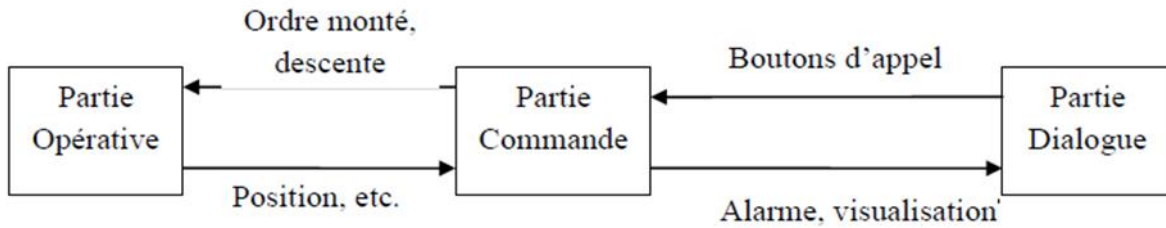


Figure I.1 : Structure d'un système automatisé

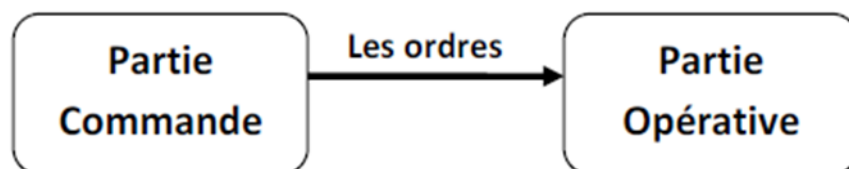
- **Partie Opérative (P.O) :** elle consomme de l'énergie électrique, pneumatique (air) ou hydraulique (eau ou huile) elle contient :
 - **Les actionneurs :** qui transforment l'énergie reçue en mouvement de rotation ou translation.
 - **Les capteurs :** qui transforment les variations de grandeurs physiques (température, luminosité, présence, position,...) en signaux électrique produisant une information qui permet au système automatisé de déclencher des actions.
- **Partie commande (P.C) :** elle transmet les ordres aux actionneurs à partir :
 - Du programme qu'elle contient.
 - Des informations reçues par les capteurs.
 - Des consignes données par l'utilisateur ou l'opérateur.
- **Partie Dialogue (P.D) :**

On l'appelle aussi la partie relation qui comporte le pupitre de dialogue homme-machine, équipé des organes de commande (boutons d'appel).

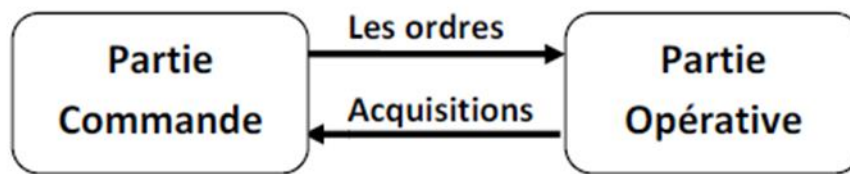
Dans un ascenseur, l'ensemble électromécanique (cabine, moteur, portes,...) constitue la partie opérative, les boutons d'appel, la logique (combinatoire, séquentielle, programmée) constituent la partie commande [21].

I.3.3. Le fonctionnement :

Boucle ouverte : La partie commande envoie des ordres à la partie opérative, mais elle ne vérifie pas s'ils ont bien été effectués (ex : feux de croisement).



Boucle fermée : Ce système est beaucoup plus fiable car le système vérifie que les ordres donnés ont bien été effectués (ex : Ascenseur).



I.4. Historique de l'ascenseur :

Les premiers ascenseurs firent leur apparition dans les années 1850 en Amérique. Leurs plateformes alors actionnées par la vapeur servaient au transport des marchandises ou à l'extraction du charbon dans les mines. Cependant l'idée remonte bien avant Jésus Christ. On peut penser qu'un treuil actionné par la force humaine aurait servi à la réalisation des grandes pyramides d'Égypte il y a plus de quatre millénaires. Ce treuil était aussi utilisé au moyen-âge pour permettre aux Hommes d'avoir accès aux endroits isolés, tels que les châteaux et les monastères.

Il faudra attendre 1852 pour qu'Elisha Otis permette une application universelle de cette invention. Il met en effet au point un ascenseur "parachute" pour répondre au problème de la chute lorsque le câble se cassait. Le "parachute" consistait en un vieux ressort de charrette muni de barre à cliquets fixé sur des rails qui empêchait la chute libre par le seul poids de la plateforme.

Le premier ascenseur affrété au transport de personnes est installé le 23 mars 1857 dans le magasin d'E. V. Haughwout & Company, à New York. Ce bâtiment de cinq étages était considéré comme le plus élevé pour l'époque. Otis imagina un ascenseur actionné par une série d'arbres et de courroies entraînés par une centrale à vapeur dans le bâtiment. Sa capacité était de 450 kilogrammes à une vitesse de 0,2 mètre par seconde. Quelques années plus tard il inventera le moteur à vapeur alternatif Otis. [5]

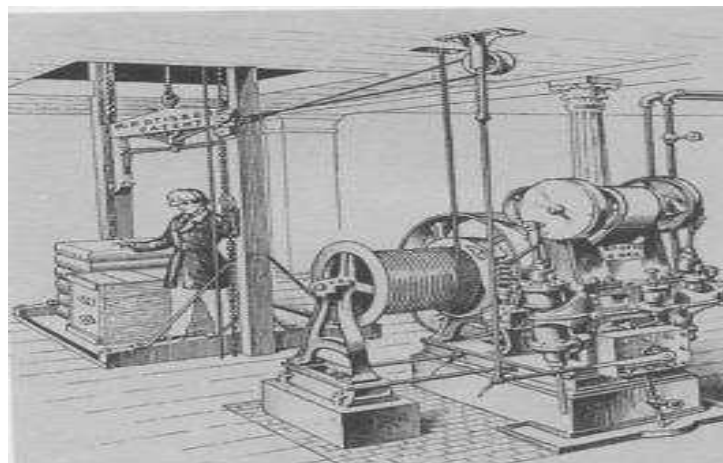


Figure I.2 : élévateur Otis avec moteur à vapeur sortie 1862.[5]

Depuis cette invention les techniques n'ont cessé d'évoluer pour transporter plus de personnes en moins de temps. Comme en 1878 lors de l'exposition universelle Léon Edoux présente un élévateur Hydraulique, cet ascenseur pouvait transporter 80 personnes à 60 mètres de hauteur, sa vitesse était de 1,10 mètre par seconde, un exploit pour l'époque. Le premier ascenseur électrique fut construit par la firme allemande SIEMENS, pour l'Exposition industrielle de Mannheim, en 1880.

Il montait à 22 m en 11 secondes. Il transporta 8 000 passagers en un mois au sommet d'une tour d'observation dominant l'exposition. Puis en 1889 dans le cadre de l'exposition universelle à Paris, la tour Eiffel est dotée de quatre ascenseurs hydrauliques, pour permettre aux visiteurs de l'exposition d'admirer Paris depuis la Tour Eiffel.



Figure I.3 : Ascenseur de la tour Eiffel.[5]

En 1903 une mise au point de l'ascenseur électrique ouvre la voie à la construction d'immeubles plus conséquents et plus haut. En 1925, la technologie de l'ascenseur devient fiable, plus besoin de liftier (personne manœuvrant un ascenseur).

En 1980 grâce à l'informatique et l'électronique les ascenseurs rejoignent les technologies de pointe. Des améliorations, en particulier au plan de la vitesse, de la précision d'arrêt, de la gestion du trafic, des systèmes de téléalarme ou télésurveillance, ont été effectuées.

Enfin en 1990 les dernières techniques améliorent encore la sécurité, favorisent l'accès des appareils aux personnes à mobilité réduite. [5]

I.5. Les ascenseurs :

I.5.1. Présentation de l'ascenseur :

C'est un dispositif utilisé pour transporter des personnes ou des charges entre différents étages d'un immeuble. Il comporte une cabine entraînée par un moteur électrique à l'aide d'un câble métallique.

La montée et la descente de la cabine se font par le changement du sens de rotation du moteur.

I.5.2. Approche Ascenseur-Système Automatisé :

Les ascenseurs modernes sont de nos jours des systèmes intelligents. Leur « cerveau » des ascenseurs a lui aussi su évoluer dans le temps. Il s'agissait au départ de simples API (Automate Programmable Industriel) qui n'exploitaient pas à son maximum le rendement possible d'un ascenseur, multipliaient les mouvements inutiles, gaspillaient de l'énergie, sans oublier l'usure prématurée de l'installation et le mécontentement des usagers voyant passer devant eux une cabine seulement partiellement occupée.

De plus de nouvelles normes par exemple en matière de précision à l'arrêt et pour les personnes âgées, l'exigence de confort des usagers et les besoins de rendement accrus pour limiter les pertes d'énergie, ont accéléré les processus de modernisation et ont poussé les concepteurs à intégrer des composants plus complexes tels que les microprocesseurs. Les ascenseurs modernes sont désormais plus ou moins construits suivant le modèle de la figure ci-dessous.

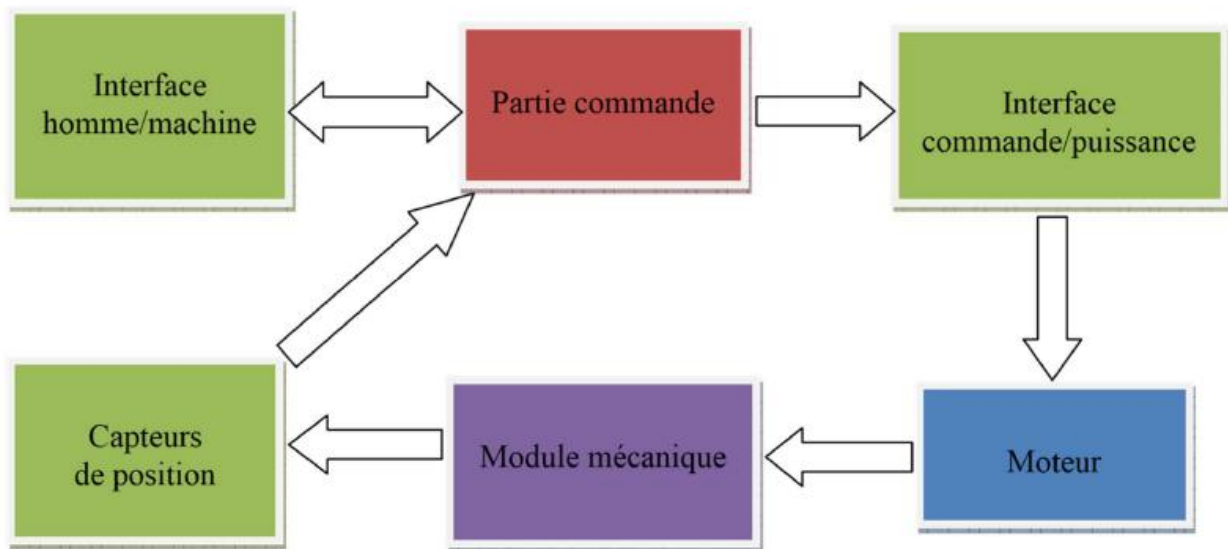


Figure I.4 : Schéma en bloc d'un ascenseur automatisé.

Description des blocs :

- **Partie commande (rouge) :** génère les consignes de déplacement de l'ascenseur, traite les données des capteurs de position et les demandes de l'utilisateur

- **Entrées/sorties (vert) :**
 - ✓ **Interface Commande/ Puissance :** assure l'alimentation électrique du moteur en fonction de la consigne donnée.
 - ✓ **Capteurs de position :** informe en permanence la partie commande sur la présence ou l'absence de la cabine à chaque étage.
 - ✓ **Interface Homme/Machine :** assure la communication avec l'utilisateur ; reçoit ses demandes via les boutons poussoir et informe sur l'état actuel de l'ascenseur.
- **Actionneur (bleu) :** Moteur : il assure le mouvement de l'ascenseur.
- **Module mécanique (violet) :** C'est la cabine ou habitacle de l'ascenseur

I.5.3. Différent types d'ascenseur :

- **Ascenseur :**

Appareil élévateur installé dans un immeuble à des endroits bien précis, afin de permettre aux usagers de se déplacer entre différents niveaux. Son déplacement se fait le long de guides verticaux dont l'inclinaison est inférieure à 15 degrés [4].

- **Ascenseur de charge :**

Est un ascenseur principalement destiné au transport de charges qui sont généralement accompagnés par des personnes. En particulier cet appareil élévateur sert uniquement au transport des objets, lorsque les dimensions et la constitution de leur cabine permet l'accès à des personnes et d'atteindre les boutons de commande, ces dispositifs doivent être classés dans la catégorie "Ascenseurs" et non "Monte-charge" [4].

- **Monte-charge :**

Le fonctionnement de ces appareils élévateurs reste le même avec les ascenseurs de charge, cependant la cabine n'est pas accessible à des personnes et les dimensions diffère, ces dernières sont données comme suit :

- ✓ surface = 1 m² ;
- ✓ profondeur = 1 m ;
- ✓ hauteur = 1,20 m ;

Toutefois la hauteur peut être supérieure à 1,20m si la cabine comporte plusieurs compartiments fixes répondant chacun aux dimensions ci- dessus [4].

- **Monte-charge industriel :**

Un appareil de levage installé à l'industrie, desservant des niveaux définis, qui comporte une cabine ou un plateau accessible aux personnes pour le chargement ou déchargement, qui se déplace le long d'un ou de plusieurs guides verticaux ou dont l'inclinaison est inférieure par rapport à la

verticale à 15 degrés. La commande de ces appareils ne peut se faire que de l'extérieur. Ces dispositifs sont interdits au transport de personnes [4].

- **Monte voitures :**

Ascenseur dont la cabine est dimensionnée pour le transport de véhicules automobiles de tourisme. Si les voitures sont accompagnées par des personnes, la réglementation régissant l'utilisation de ces appareils est identique à celle des ascenseurs [4].

- **Ascenseur pour le transport de personnes handicapées :**

Toute installation installée à demeure, construite et utilisée principalement pour le transport des personnes handicapées, debout ou en fauteuil roulant, avec ou sans accompagnateur [4].

- **Ascenseur sur plan incliné :**

Tout ascenseur sur plan incliné utilisé principalement au transport de personnes à mobilité réduite. Cet appareil peut être équipée d'un siège et/ou d'une plate-forme pour le transport d'une personne en position debout ou d'une plate-forme pouvant recevoir un fauteuil roulant.

Les ascenseurs sur plan incliné peuvent être installés dans des cages d'escalier droites ou présentant des virages. Ils peuvent également être installés à l'extérieur des bâtiments dans des rampes ou escalier d'accès [4].

I.5.4. Les familles d'ascenseur :

On distingue essentiellement deux types de familles d'ascenseur :

- Les ascenseurs à traction à câble ou électrique.
- Les ascenseurs hydrauliques.

En générale, ses deux types utilisent l'énergie électrique pour déplacer les cabines verticalement (moteur électrique à courant continu ou alternatif) [6].

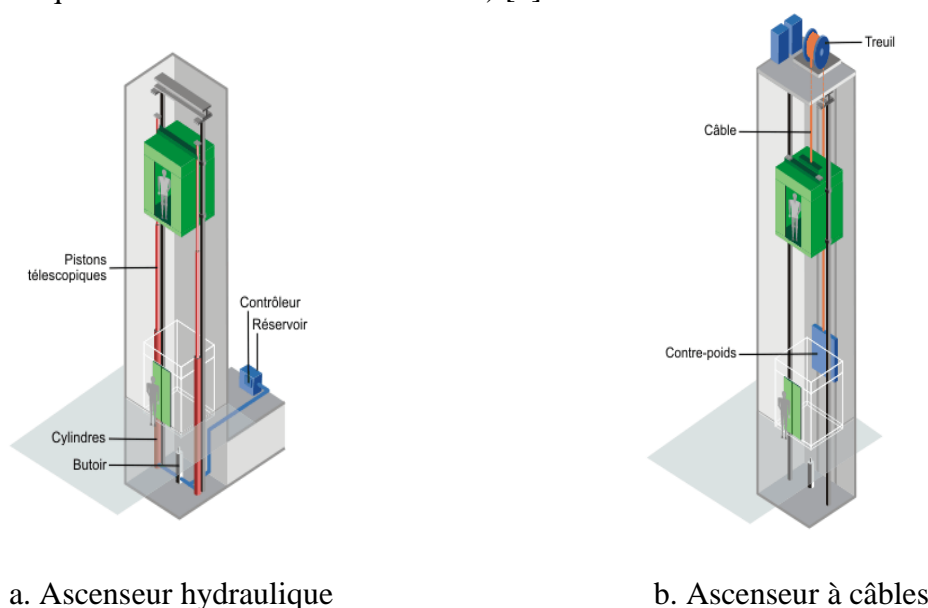


Figure I.5 : Les familles d'ascenseurs.

I.5.4.1. Les ascenseurs hydrauliques :

➤ **Principe de fonctionnement :**

Comme toute machine hydraulique la pompe met sous pression l'huile qui pousse le piston hors du cylindre vers le haut. Lorsque la commande de descente est programmée, la vanne de la pompe permet de laisser sortir l'huile du cylindre vers le réservoir [6].

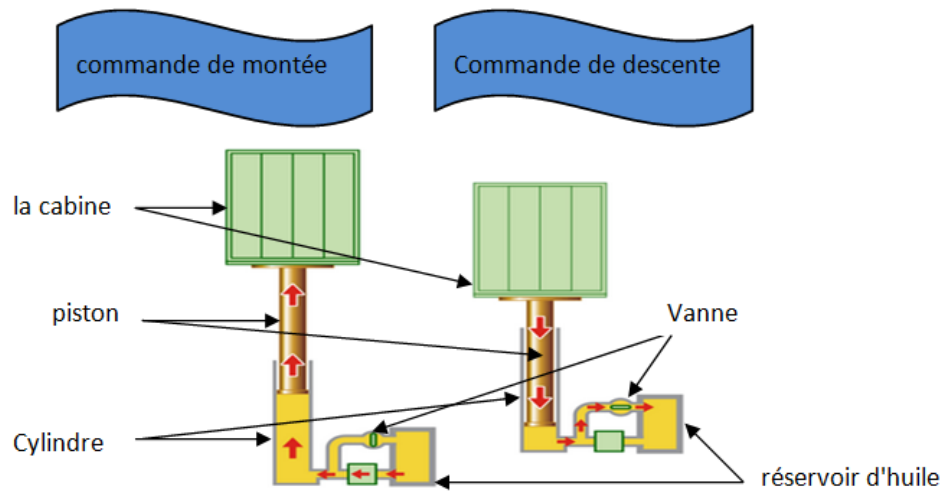
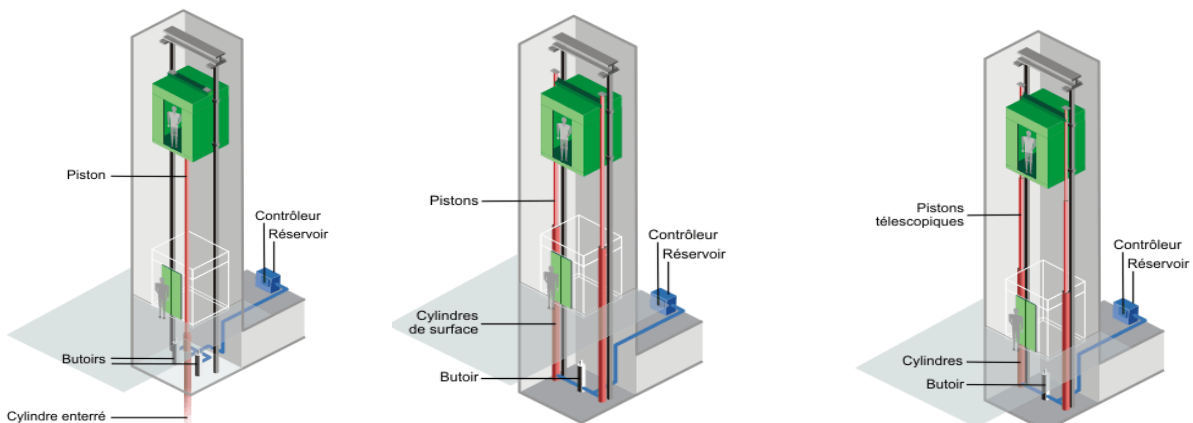


Figure I.6 : Schéma illustratif du principe de fonctionnement des ascenseurs hydrauliques.

➤ **Description :**

Les ascenseurs hydrauliques sont utilisés en générale pour satisfaire des déplacements relativement courts de l'ordre de 15 à 18 m maximums. Plusieurs modèles existent sur le marché on citera :

- Les ascenseurs hydrauliques à cylindre de surface ;
- Les ascenseurs hydrauliques à cylindre enterré ;
- Les ascenseurs hydrauliques télescopiques à cylindre de surface [6].



a. A cylindre enterré

b. A cylindre de surface

c. A cylindre de surface télescopique

Figure I.7 : Les différents modèles de l'ascenseur hydraulique.

Les différents modèles permettent de tenir compte des critères suivants :

- ✓ de place ;
- ✓ de hauteur d'immeuble à desservir ;
- ✓ de stabilité de sol et de sous-sol ;
- ✓ de risque de pollution par rapport au sol et plus spécifiquement aux nappes phréatiques ;
- ✓ d'esthétique ;
- ✓ Etc.

Les ascenseurs hydrauliques se composent principalement de :

- ✓ d'une cabine ;
- ✓ de guides ;
- ✓ d'un ensemble pistons-cylindres hydrauliques placé sous la cabine de l'ascenseur ;
- ✓ d'un réservoir d'huile ;
- ✓ d'un moteur électrique accouplé à une pompe hydraulique ;
- ✓ d'un contrôleur ;
- ✓ Etc [6].

➤ **Energie :**

Energétiquement parlant les ascenseurs hydrauliques posent un problème dans le sens où il n'y a pas de contrepoids qui équilibre la cabine comme dans les systèmes à traction à câble par exemple [6].

➤ **Avantages et inconvénients :**

Avantage

- ✓ Précision au niveau du déplacement ;
- ✓ Réglage facile de la vitesse de déplacement ;
- ✓ Ne nécessite pas de salle de machinerie ;
- ✓ Implantation facile dans un immeuble existant ;
- ✓ Etc.

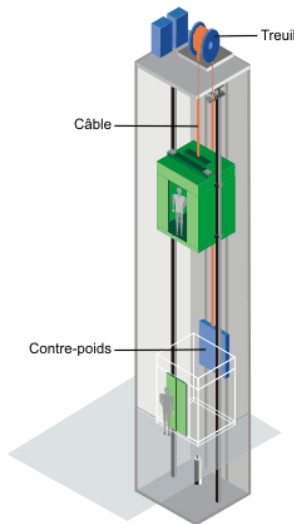
Inconvénients

- ✓ Course verticale limitée à une hauteur entre 15 et 18 m ;
- ✓ Risque de pollution des sous-sols ;
- ✓ Consommation énergétique ;
- ✓ nécessité de renforcer la dalle de sol ;
- ✓ Etc [6].

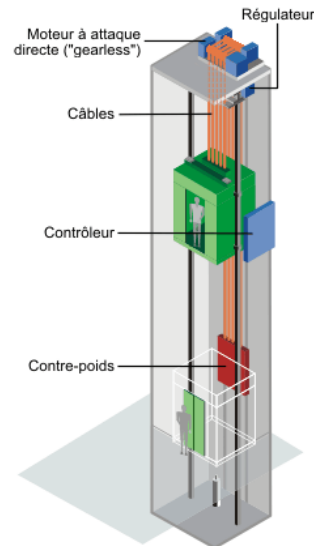
I.5.4.2. Les ascenseurs à traction à câble :

➤ Description :

Les ascenseurs à traction câble sont les types d'ascenseurs que l'on rencontre souvent. Ils se différencient entre eux selon le type de motorisation [6].



a. Ascenseur à moteur à treuil



b. Ascenseur à moteur à attaque directe

Figure I.8 : Les types d'ascenseur à traction à câble.

Quel que soit le type, les ascenseurs à traction à câble comprennent généralement :

- ✓ une cabine ;
- ✓ un contrepoids ;
- ✓ des câbles reliant la cabine au contrepoids ;
- ✓ des guides ;
- ✓ un système de traction au-dessus de la cage de l'ascenseur ;
- ✓ Etc [6].

➤ Energie :

Energétiquement parlant les ascenseurs à traction à câble sont plus intéressants que les ascenseurs hydrauliques dans le sens où le contrepoids réduit fortement la charge quel que soit le type de la motorisation [6].

➤ Avantages et inconvénients :

Avantages

- ✓ Course verticale pas vraiment limitée ;
- ✓ Suivant le type de motorisation précision au niveau de la vitesse et du déplacement ;
- ✓ Rapidité du déplacement ;
- ✓ Efficacité énergétique importante ;

- ✓ Pas de souci de pollution ;
- ✓ Etc.

Inconvénients

- ✓ En version standard, nécessite une salle de machinerie ;
- ✓ Exigence très importante sur l'entretien ;
- ✓ Etc [6].

I.5.5. Différentes parties d'un ascenseur à traction :

Les ascenseurs à traction sont en général constitués de ces parties (voir la figure I.9)

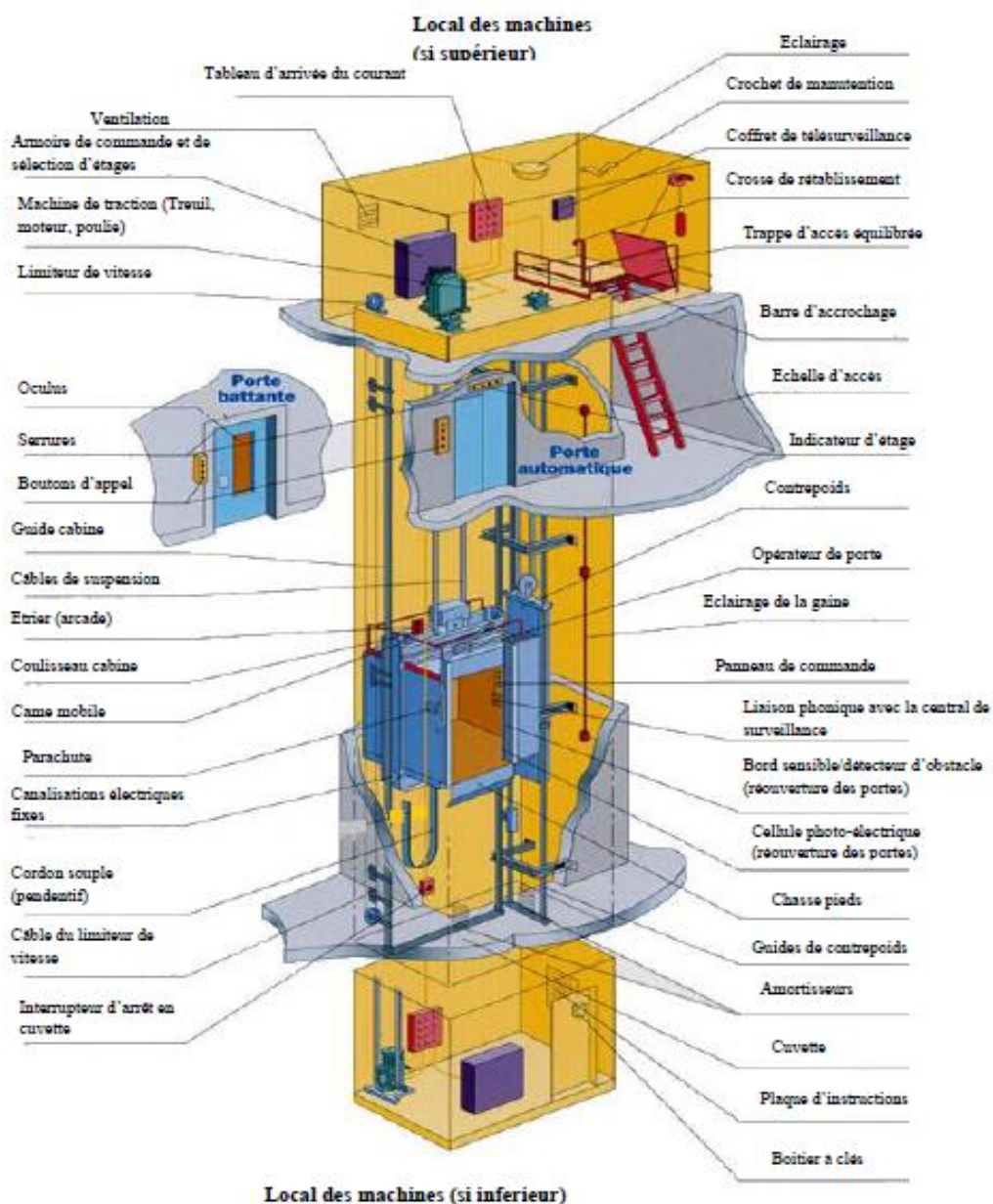


Figure I.9 : Les composants d'un ascenseur à traction électrique.

- **Cabine d'ascenseur** : Destinée à accueillir les personnes et les marchandises, elle est constituée de quatre parties principales qui - dans l'ordre de leur montage pendant l'installation sont :
- **L'étrier (suspension cabine)** : il est constitué de deux cadres en acier, réunis entre eux de telle sorte que l'ensemble est symétrique par rapport à l'axe transversal des guides. Chaque cadre comporte à la portée inférieure et supérieure deux traverses horizontales réunies par deux profilés verticaux parallèles aux guides. A la traverse supérieure sont attachés les câbles de suspension. Le traverse inférieure est constituée en générale d'une ou de plusieurs plaques soudées et les butées des amortisseurs. L'étrier est le premier élément mis en place pendant l'installation de la cabine.
 - **Le plancher** : Le socle (plate-forme) se monte directement sur les traverses de l'étrier auxquelles il est fixé par des vis. Il est formé également d'un cadre en acier. Sur le socle est disposé le plancher de la cabine. Le plancher est mise en place en fin de montage.
 - **Les parois** : Sont le plus souvent en tôles pliées ou en bois. Après assemblage des parois et de l'encadrement de la baie de la cabine, le toit est mis en place.
 - **Le toit** : est le plus souvent assemblé en atelier et monté d'un seul bloc. le toit de la cabine est formé d'un cadre en tôles pliées ou cintré [4], [7].
- **Porte de cabine** : Porte à fermeture généralement automatique destinée à confiner l'utilisateur dans la cabine pendant le déplacement de celle-ci, lui interdisant tous contacts avec les parties extérieures à la cabine [4].
- **Porte palières** : Une porte extérieure fixe est installée à chaque étage, isolant les personnes extérieures sur le palier du puits de gaine et de la cabine éventuellement en mouvement. Elles peuvent être battantes ou coulissantes et commandées manuellement ou automatiques. Elles doivent être équipées d'un dispositif empêchant leurs ouvertures si la cabine n'est pas sur le niveau et bloquant le départ pendant leur ouverture [4].
- **Serrures** : La serrure est l'élément qui sécurise un ascenseur pour le public. Son rôle est de verrouiller mécaniquement les portes palières. Elle possède également un contact électrique qui contrôle que le verrouillage mécanique est bien en place [7].
- **Came mobile** : Le rôle de cet organe est d'agir sur les galets de la serrure afin d'en effectuer le déverrouillage, mais seulement lorsque la cabine s'y arrête. Il est constitué essentiellement d'un électroaimant pouvant attirer une armature qui porte une came [4].
- **Boutons d'appels** : Ce sont les boutons qui commandent l'arrivée de la cabine, ils sont installés aux paliers [4].
- **Boutons d'envois** : Ce sont les boutons de précision d'étage ; ils sont installés dans la cabine [4].

- **Charge utile :** C'est la charge pour laquelle l'appareil a été construit, elle varie en fonction de la surface de la cabine. Au-delà de cette capacité, le système de traction n'est plus en mesure de contrôler le déplacement et l'arrêt correct de la cabine. dans certains cas de surcharge exagérée, des blocages intempestifs peuvent se produire [4].
- **Garde (chasse) pieds :** est une tôle fixe ou rétractable, destinée à protéger les chutes en gaines lorsque la cabine est immobilisée en dehors de la zone de déverrouillage [7].
- **Contrepoids :** Élément destiné à contre balancer le poids de la cabine (lorsque la cabine monte, le contre poids descend et vis-versa). Il est constitué d'un étrier (suspension) métallique et des masses de fonte appelées gueuses de contrepoids. La masse du contrepoids (P_{cp}) est plus lourde que l'ascenseur, elle représente l'équivalent du poids de la cabine (P_{ca}) et la moitié de sa charge utile (P_{ch}), ce calcul est donné par cette formule $P_{cp} = P_{ca} + P_{ch}/2$ [4].
- **Gaine ou trémie :** (Appelée aussi pylône) c'est une gaine verticale dans laquelle se déplace l'ascenseur et son contrepoids. Elle est équipée de rails de guidages (guides) des éléments mobiles (cabine et contrepoids). Elle doit être fermée sur toute sa hauteur [4], [7].
- **Guides :** Profilés en acier, situés de part et d'autre, le long de la course de la cabine, ils sont habituellement en forme de T. Ils sont destinés à guider la cabine et le contre poids dans la gaine [4].
- **Ancrage de guide :** Pièce métallique servant à fixer les guides aux murs de la gaine [4].
- **Coulisseau :** Ils sont situés à chaque coin de l'étrier, et sont en appui sur les guides. Durant le déplacement de la cabine, ceux-ci glisse sur les guides, huilés régulièrement pour limiter les frottements et les accrocs, et donc le bruit, et augmenter le confort. Dans certains cas ces coulisseaux peuvent être remplacés par des rollers (petite roue d'un diamètre de 80 mm à 200 mm) comme pour des cabines à grande vitesse ou charges lourdes [7].
- **Cuvette :** Partie de la gaine située en bas du niveau d'arrêt inférieur desservi par la cabine, contenant les poulies de renvoi et les amortisseurs [2].
- **Amortisseurs :** Les amortisseurs sont destinés à assurer le ralentissement et l'arrêt de la cabine en cas de dépassement des fins de course de sécurité. Ils sont placés généralement en font de cuvette à l'aplomb des traverses des étriers de cabine ou de contrepoids. On distingue deux types :
 - Amortisseurs à ressorts ;
 - Amortisseurs hydrauliques [7].
- **Poulie de renvoi :** Poulie tournante librement et destinée à guider les câbles entre la cabine et le contrepoids [4].
- **Poulie de mouflage :** Certains ascenseurs à grande capacité sont mouflés. C'est-à-dire qu'une démultiplication est installée à l'aide de poulies de mouflage. Lorsque la cabine parcourt un mètre,

les câbles au niveau des treuils en parcourent deux ou trois. Cette méthode permet d'installer des treuils moins puissants mais augmente la longueur des câbles et le coût de leur remplacement [4].

- **Fin de course :** Contact de sécurité placé généralement en gaine et destiné à stopper l'ascenseur en cas de dépassement de sa course normale [4].
- **Commande de révision (Boîtier d'inspection) :** La commande de révision est composée d'un boîtier placé sur le toit de la cabine de l'ascenseur. Ce boîtier, équipé de bouton de marche montée et descente ainsi que d'un bouton d'arrêt d'urgence, permet au préposé à l'entretien de manœuvrer, en toute sécurité et à faible allure, l'ascenseur pour inspecter et graisser les organes placés en gaine [4].
- **Parachute :** Organe mécanique placé sur la suspension de cabine est commandé par un câble de limiteur. En cas de rupture des câbles de traction ou de survitesse exagérée en descente, le mécanisme du parachute assure un blocage mécanique de la suspension dans les guides évitant la chute libre de la cabine. Ce dispositif peut, dans certains cas, équiper le contrepoids [4].
- **Machinerie (salle des machines) :** Local généralement placé au-dessus de la gaine et destiné à contenir l'appareillage et le système de traction [4].

I.5.5.1 Système de motorisation d'un ascenseur à traction :

I.5.5.1.1 Les moteurs-treuils à vis sans fin à une ou deux vitesses :

Dans ce type de motorisation, la vis sans fin entraîne beaucoup de pertes mécaniques et, par conséquent, des consommations électriques plus importantes. Au début de l'utilisation des vis sans fin, les rendements énergétiques de l'ensemble moteur treuil étaient de l'ordre de 20 %. Avec le perfectionnement des outils, des lubrifiants, les rendements se sont nettement améliorés pour atteindre les 45% et même récemment 60 à 65 %.

Les moteurs électriques couplés au treuil à vis sans fin étaient généralement des moteurs à courant continu à excitation indépendante ou shunt avec la faculté de pouvoir faire varier très facilement la vitesse.

Les moteurs électriques à courant alternatif utilisés avec ce type de réducteur sont en principe des moteurs à deux vitesses : au démarrage, la vitesse est plus lente (Petite vitesse), pour atteindre la vitesse de déplacement optimale, le moteur passe à la seconde vitesse en provoquant un léger choc d'accélération (passage de la petite vitesse à la grande vitesse).

I.5.5.1.2 Les moteurs-treuils planétaires :

Les appareils à treuil planétaire utilisent le système de réduction de vitesse par engrenages planétaires. Accouplés à un moteur électrique, ils permettent d'avoir un rapport de réduction appréciable pour obtenir une plage de vitesse compatible avec le confort et l'efficacité de déplacement souhaité.

Ce système à un rendement mécanique de l'ordre de 97 à 98 % permettant, pour autant que les moteurs d'entraînement soient performants, d'obtenir des rendements énergétiques globaux intéressants (de l'ordre de 80%).

Les réducteurs planétaires peuvent être accouplés à des moteurs électriques :

- A courant continu (grande plage de variation de vitesse).
- A courant alternatif asynchrone à deux vitesses.
- A courant alternatif asynchrone commandé par un variateur de fréquence.

I.5.5.1.3 Les moteurs à attaque directe « gearless» ou « sans treuil» :

Il s'agit d'un moteur sans réducteur, la poulie de traction est monté directement sur l'arbre de sortie du moteur et la régulation de vitesse est obtenue grâce à un variateur de fréquence.

Ce système est énergétiquement performant principalement de par la présence d'un variateur de fréquence qui optimise la consommation énergétique, aussi, les pertes mécaniques sont réduites vu l'absence des engrenages.

I.6. 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. Ces appareils occupent les premières positions 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 donné une idée générale sur les systèmes automatisés et leur structure. Puis on a passé à l'étude des différents types d'ascenseur dont on a cité les avantages et les inconvénients de chacun, leurs critères énergétiques et les critères de choix du type d'ascenseur. Nous nous sommes basé sur l'étude des ascenseurs à traction électrique puisqu'ils sont les plus utilisé de nos jours grâce à la nouvelle technologie.

Chapitre 2

Etude de la carte et logiciel de programmation

II.1. Introduction :

Le projet Arduino (2005) est issu d'une équipe d'enseignants et d'étudiants de l'école de Design d'Interaction d'Ivrea suite au problème rencontré à cette période-là : Les outils nécessaires à la création de projets d'interactivité étaient complexes et excessivement chers ce qui rendait difficile le développement des projets et ralentissait l'apprentissage des étudiants.

Pour cela ils se concentrèrent sur la réalisation d'un matériel moins cher et plus facile à utiliser. Ils souhaitaient créer un environnement proche de Processing (langage de programmation développé en 2001 par " Benjamin Fry" et " Casey Reas") et de la carte Wiring (un langage de programmation facile d'accès et adaptée aux développements de projets de designers créée par Hernando Barragan en 2001 pour sa thèse de fin d'étude) pour arriver à un dispositif simple à utiliser, dont les coûts seraient peu élevés, les codes et les plans libres (les sources peuvent être modifiés, améliorés ou distribués par les utilisateurs) et multi-plates-formes (indépendant du système d'exploitation utilisé) [8].

II.2. La carte Arduino :

II.2.1. Définition :

Arduino est une plate-forme de prototypage d'objets interactifs à usage créatif constituée d'une carte électronique et d'un environnement de programmation, cet environnement matériel et logiciel permet à l'utilisateur de formuler ses projets par l'expérimentation directe avec l'aide de nombreuses ressources disponibles en ligne.

Arduino est un projet open source (les plans, les schémas, etc. sont accessibles et libres de droits) la grande communauté d'utilisateurs et de concepteurs permet à chacun de trouver les réponses à ces questions et apporte un travail énorme de documentation au projet. Les projets Arduino peuvent être autonomes, comme ils peuvent communiquer avec d'autres logiciels installés sur l'ordinateur tel que Flash, Processing ou MaxMSP, Matlab). Ces cartes sont faites à base d'une interface entrée/sortie simple et d'un environnement de développement proche du langage [8], [9].

II.2.2. Les gammes de la carte Arduino :

Actuellement, il existe plus de 20 versions de module Arduino, nous citons quelques-unes afin d'éclaircir l'évaluation de ce produit scientifique et académique :

- Le NG d'Arduino, avec une interface d'USB pour programmer et usage d'un ATmega8.
- L'Arduino Mini, une version miniature de l'Arduino en utilisant un microcontrôleur ATmega168.
- L'Arduino Nano, une petite carte programmable à travers le port USB ; cette version utilise un microcontrôleur ATmega168 (ATmega328 pour une nouvelle version).

- Le LilyPad Arduino, une conception de minimaliste pour l'application wearable (que l'on peut porter) en utilisant un microcontrôleur ATmega168.
- Le NG d'Arduino plus, avec une interface d'USB pour programmer et usage d'un ATmega168.
- L'Arduino Bluetooth, avec une interface de Bluetooth pour programmer en utilisant un microcontrôleur ATmega168.
- L'Arduino Diecimila, avec une interface d'USB et utilise un microcontrôleur ATmega168.
- L'Arduino Duemilanove ("2009"), en utilisant un microcontrôleur l'ATmega168 (ATmega328 pour une nouvelle version) et actionné par l'intermédiaire de la puissance d'USB/DC.
- L'Arduino Mega, en utilisant un microcontrôleur ATmega1280 pour I/O additionnel et mémoire.
- L'Arduino UNO, utilisations microcontrôleur ATmega328.
- L'Arduino Mega2560, utilisations un microcontrôleur ATmega2560. Elle incorpore également le nouvel ATmega8U2.
- L'Arduino Leonardo, avec un morceau ATmega3U4 qui élimine le besoin de raccordement d'USB et peut être employé comme clavier.
- L'Arduino Esplora, ressemblant à un contrôleur visuel de jeu, avec un manche et des sondes intégrées pour le bruit, la lumière, la température, et l'accélération [10].

II.2.3. Les composants de la carte :

L'Arduino est composée de deux parties indissociables : La carte qui est la partie hardware avec laquelle on travaille en construisant chaque projet et la plateforme IDE ARDUINO qui est la partie logicielle fonctionnant sur le PC celle-ci permet de mettre au point et de transférer le programme qui sera par la suite exécutée par la carte ARDUINO [8], [11].

Dans ce qui suit nous allons présenter les composants de la carte Arduino UNO :

II.2.3.1. Le microcontrôleur :

C'est le processeur de la carte, il s'occupe de tout ce qui est calculs, exécution des instructions du programme et gestion des ports d'entrées/sorties. Il dispose de 1 Ko de mémoire vive, et 32 Ko de mémoire flash pour stocker les Programmes.

Pour les besoins de l'Arduino le microcontrôleur devait être petit et bon, les développeurs ont donc opté pour le circuit ATmega328 qui réunit ces deux conditions. Il est associé à des ports entrées/sorties qui permettent à l'utilisateur de communiquer avec l'environnement [8], [11].

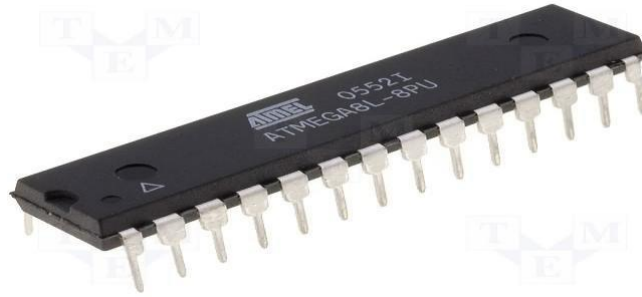


Figure II.1 : Le microcontrôleur de la carte Arduino.

II.2.3.2. Les 14 entrées sorties numériques (pattes 0_13) :

Elles peuvent être configurées en entrée ou en sortie en les indiquant dans le programme. Côté entrées, des capteurs qui collectent des informations sur l'environnement, le contact via un bouton-poussoir. Côté sorties, des actionneurs qui agissent sur le monde physique tel qu'une petite lampe qui produit de la lumière, un moteur qui actionne une cabine d'ascenseur...etc. Chaque broche peut fournir ou recevoir un maximum de 40mA d'intensité.



Figure II.2 : Les 14 entrées sorties numériques.

Certaines broches ont des fonctions spéciales :

- Les broches 0 et 1 : Utilisées pour recevoir (RX) et transmettre (TX) les données séries, Ces broches sont connectées aux broches correspondantes du circuit intégré ATmega programmé en convertisseur USB-vers-série de la carte (Le FTDI), qui assure l'interface entre le microcontrôleur et le port USB de l'ordinateur.
- Broches 2 et 3 : Ces broches peuvent être configurées pour déclencher une interruption sur une valeur basse, sur un front montant ou descendant ou sur un changement de valeur ; c'est l'instruction `attachInterrupt()` qui est adaptée à cette configuration.
- Broches 3, 5, 6, 9, 10, et 11 : Elles fournissent une impulsion PWM (Modulation à Largeur d'Impulsion) à l'aide de l'instruction `analogWrite()`.
- Broches 10, 11, 12, 13 : Ces broches supportent la communication SPI (Interface Série Périphérique) disponible avec la librairie pour communication SPI.
- Broches 4 et 5 : Supportent les communications de protocole I2C.

- Broche 13 : Il y a une LED incluse dans la carte connectée à la broche 13, lorsque la broche est au niveau HAUT, la LED est allumée, lorsque la broche est au niveau BAS, la LED est éteinte [8], [11].

II.2.3.3. Les 6 entrées analogiques (pattes A0_A5) :

Ces pattes sont réservées à la mesure de signaux analogiques et permettent de convertir une tension analogique (Ve) de 0 à 5 V en une valeur numérique [8], [11].



Figure II.3 : Les 6 entrées sorties analogiques.

II.2.3.4. Alimentation de la carte :

La carte Arduino peut-être alimentée via un port USB qui l'alimente par +5V. La norme USB limite à 500mA maximum l'alimentation de la carte. Cela suffit pour des leds mais est généralement insuffisante pour des moteurs ou des servomoteurs (avec cette alimentation seule, lors de la mise en route d'un moteur, cela peut entrainer une chute de tension).

La carte peut aussi fonctionner avec une alimentation externe de 6 à 20 volts. Cependant, si la carte est alimentée avec moins de 7V la broche de 5V pourrait fournir moins de 5V et la carte risque d'être instable. Si on utilise plus de 12V, le régulateur de tension de la carte pourrait surchauffer et endommager la carte. Aussi, la plage idéale recommandée pour alimenter la carte Arduino est entre 7V et 12V.

La carte peut être alimentée aussi par les broches Vin(+) GND(-) qui peuvent aussi supporter une tension comprise entre 7et 12volt (tension convertie aussi en 5v par l'Arduino) [8], [11].

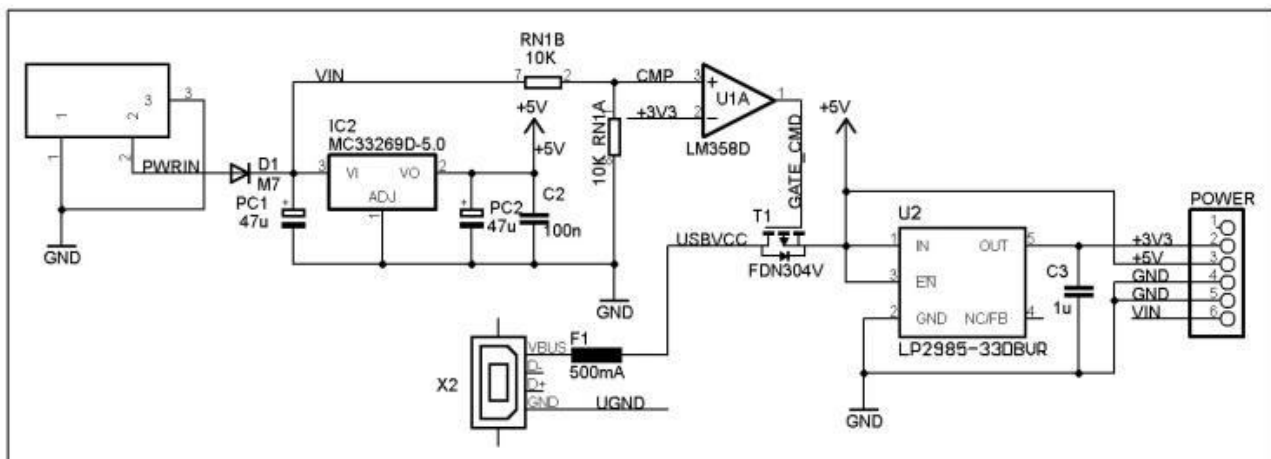


Figure II.4 : Le schéma électrique de l'alimentation de la carte Arduino.

II.2.3.5. Les LEDS :

- LED verte d'alimentation ;
- LED jaune ou verte active lors de téléchargement du programme (LED de transmission réception) ;
- LED jaune de test.

Reset : pour la réinitialisation du microcontrôleur.

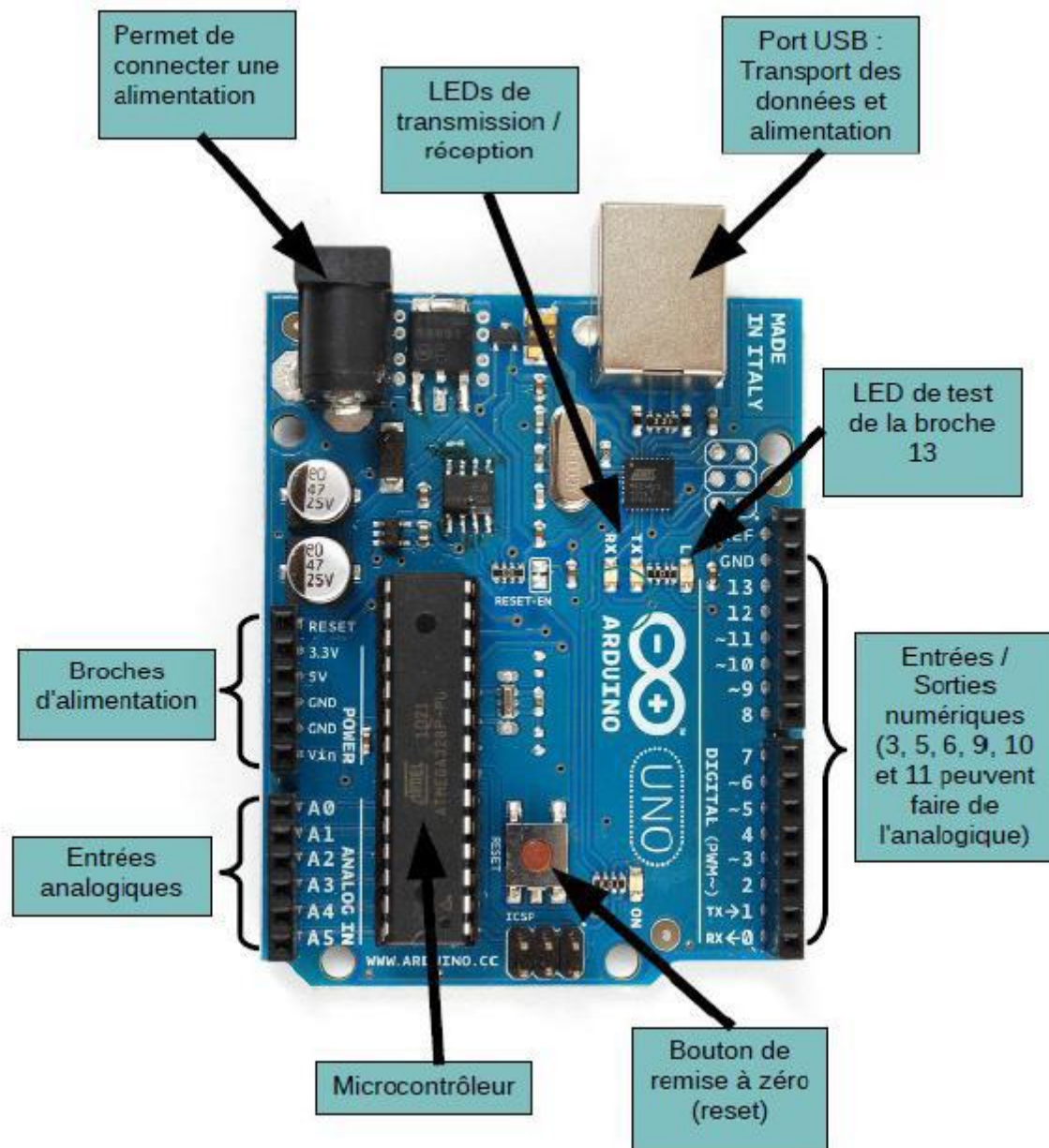


Figure II.5 : Les composants de la carte Arduino (Uno).

Nb : Contrairement au signal numérique qui ne peut prendre que deux états différents un signal analogique peut prendre une infinité de valeurs en tension qui peut varier entre 0V et 5V. La carte Arduino fonctionne en numérique, le microcontrôleur ne comprend que les « 0 » et les « 1 ». Les entrées A0 à A5 sont dotées de convertisseurs analogiques/numériques qui convertissent une tension en un entier variant de 0 à 1023 [9], [11].

II.3. Le logiciel Arduino IDE :

II.3.1. Définition de logiciel :

L'environnement de développement intégré (IDE) Arduino est une application multiplateforme (Pour Windows, MacOS, Linux) écrit en langage de programmation Java. Il est utilisé pour écrire et télécharger des programmes sur des cartes compatibles Arduino.

Le code source de l'IDE est publié sous licence GNU General Public, version 2. L'IDE Arduino supporte les langages C et C++ en utilisant des règles spéciales de structuration de code. L'IDE Arduino fournit une bibliothèque de logiciels du projet Wiring, qui fournit de nombreuses procédures d'entrée et de sortie communes. Le code écrit par l'utilisateur nécessite seulement deux fonctions de base, pour le démarrage de l'esquisse et de la boucle du programme principal, qui sont compilées et liées à un talon de programme main () dans un programme exécutable cyclique avec la chaîne d'utilité GNU, également incluse dans la distribution IDE.

L'IDE Arduino utilise le programme Avrdude pour convertir le code exécutable en un fichier texte au codage hexadécimal chargé dans la carte Arduino par un programme de chargement dans le microprogramme de la carte.

II.3.2. Présentation du logiciel :

1. options de configuration du logiciel
2. boutons pour la programmation des cartes
3. programme à créer
4. débogueur (affichage des erreurs de programmation)

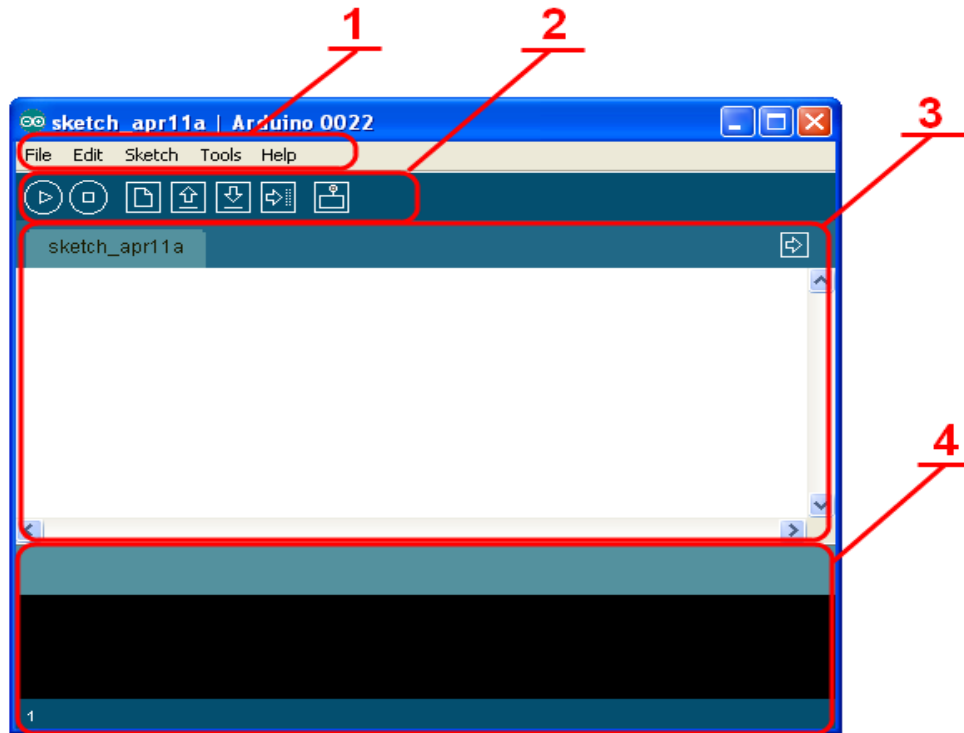


Figure II.6 : L'interface de l'IDE Arduino.

II.3.3. Approche et utilisation de logiciel :

II.3.3.1. Le menu file :

C'est principalement ce menu que l'on va utiliser le plus. Il dispose d'un certain nombre de choses qui nous vont nous être très utiles. Il a été traduit en français progressivement, nous allons donc voir les quelques options qui sortent de l'ordinaire :

- New (nouveau) : va permettre de créer un nouveau programme. Quand on appuie sur ce bouton, une nouvelle fenêtre, identique à celle-ci, s'affiche à l'écran.
- Open... (ouvrir) : avec cette commande, on peut ouvrir un programme existant.
- Save / Save as... (enregistrer / enregistrer sous...) : enregistre le document en cours / demande où enregistrer le document en cours.
- Exemples (exemples) : ceci est important, toute une liste se déroule pour afficher les noms d'exemples de programmes existants.
- Préférences : vous pourrez régler ici quelques paramètres du logiciel.

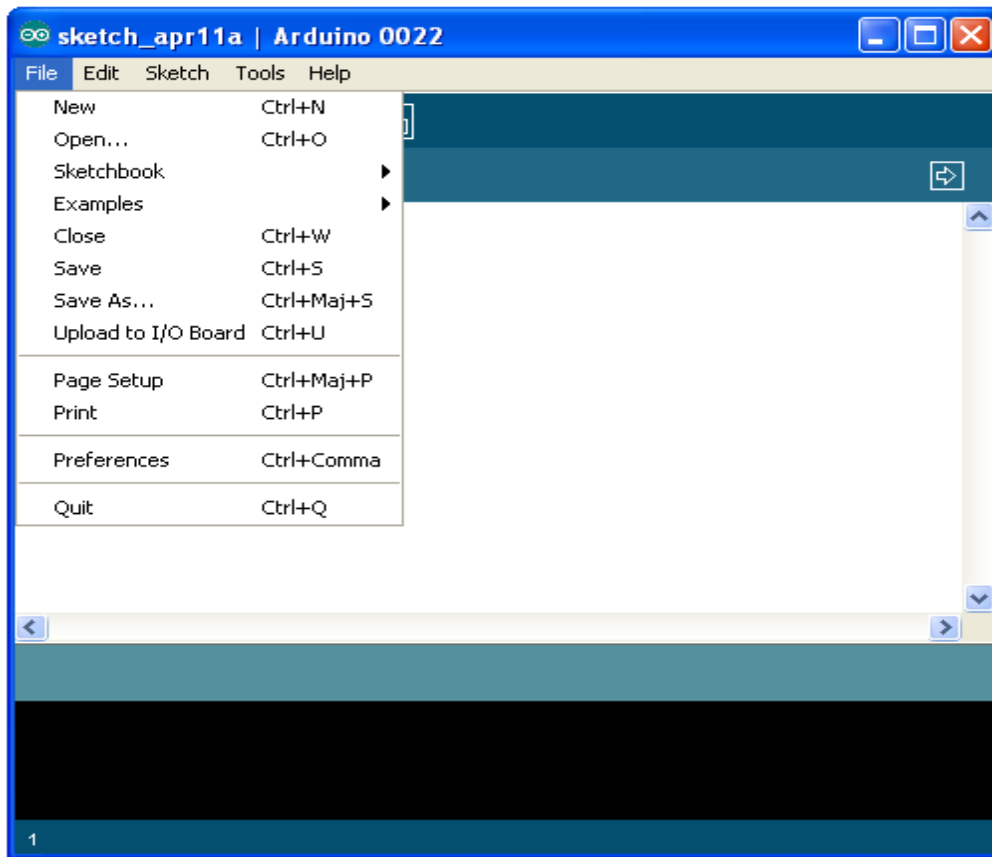


Figure II.7 : Le menu fichier

I.3.3.2. Le menu outils :

Une fois le code écrit (ou collé) dans la fenêtre de programmation, il faut l’envoyer sur l’Arduino. Pour cela, après avoir connecté l’Arduino à l’ordinateur, il faut sélectionner le port et le type de carte (Arduino Uno, dans notre cas).

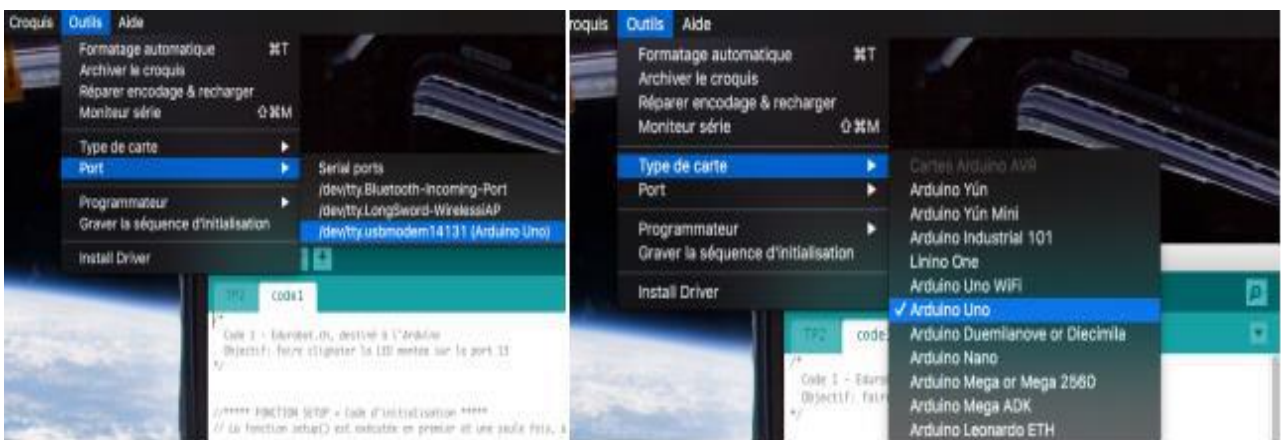


Figure II.8 : Le Menu outils.

II.3.3.3. Les boutons :

- Bouton 1 : permet de vérifier le programme, il actionne un module qui cherche les erreurs dans le programme.
- Bouton 2 : Créer un nouveau fichier.
- Bouton 3 : Sauvegarder le programme en cours.
- Bouton 4 : Liaison série.
- Bouton 5 : Stoppe la vérification.
- Bouton 6 : Charger un programme existant.
- Bouton 7 : Compiler et envoyer le programme vers la carte.

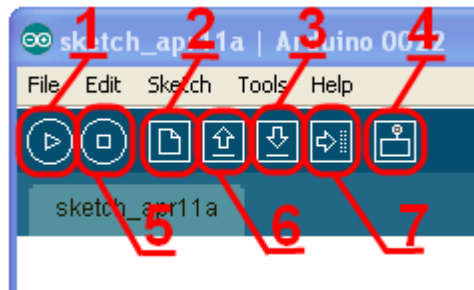


Figure II.9 : La barre d'outils

II.3.4. Structure d'un programme :

Un programme Arduino est une suite d'instructions élémentaires et séquentielles, dans tous les programmes Arduino il y a trois étapes. Pour mettre en évidence ces phases on va présenter la structure d'un exemple simple de programmation qui consiste à allumer une led pendant 1 seconde puis l'éteindre durant 3 seconde sur la broche N°13 ainsi de suite jusqu'à introduire un nouveau programme ou bien arrêter l'alimentations.

```

1  /* Ce programme fait clignoter une
   * led qui se situe dans la pin N°13
   */

2  int Led=13;           //Déclaration de la variable "Led" à utiliser

3  void setup ()
   {
     pinMode (Led,OUTPUT); // configure la broche N°13 comme une sortie
   }

4  void loop ()
   {
     digitalWrite(13,HIGH); // met la sortie N°13 à l'état haut (led allumée)
     delay(1000);           //attendre 1 seconde
     digitalWrite(13,LOW); // met la sortie N°13 à l'état bas (led éteinte)
     delay(3000);          // attendre 3 seconde
   }

```

Figure II.10 : Exemple d'un programme Arduino

1, 2, 3,4 représentent des segments expliqués ci-dessous

1 : Commentaires :

Pour écrire des commentaires sur le programme (cela aide à la relecture du programme et sa compréhension par une autre personne que celle qui l'a fait) on a 2 possibilités : Soit en multi ligne, en les mettant entre les signes `/** */`. Soit sur une ligne de code en les séparant du code avec les signes `//`.

2 : Définition et déclaration des variables :

Pour notre exemple, on va utiliser une sortie numérique de la carte qui est par exemple la 13ème sortie numérique. Cette variable doit être définie et nommée dans cette partie ; On lui donne un nom arbitraire Led.

3 : Configuration des entrées-sorties voidsetup () :

Les broches numériques de l'Arduino peuvent aussi bien être configurées en entrées numériques ou en sorties numériques. Ici on va configurer Broche LED en sortie

PinMode (nom, état) est une des fonctions relatives aux entrées-sorties numériques.

4 : Programmation des interactions voidloop () :

Dans cette boucle, on définit les opérations à effectuer, dans l'ordre :

- digitalWrite (nom, état) : Une des fonctions relatives aux entrées-sorties numériques.
- delay (temps en millisecondes) : La commande d'attente entre deux autres instructions.
- chaque ligne d'instruction est terminée par un point-virgule.
- les accolades encadrent la boucle [8], [9].

II.4. Conclusion :

Dans ce chapitre, notre étude s'est focalisée sur la présentation de la carte Arduino et son logiciel de programmation, on a commencé par quelques détails concernant la création et la définition de cette carte. Puis nous avons parlé de son côté matérielle dont nous avons cité les gammes et les composants, enfin on a donné une présentation de logiciel de programmation et la structure de programme.

La compréhension de données exposées dans ce chapitre nous offre des connaissances de bases pour l'usage de la carte de programmation.

Chapitre 3

Etude et Conception d'un Ascenseur

III.1. Introduction :

Après les généralités sur les ascenseurs et l'étude de la carte et logiciel de programmation, respectivement aux premier et deuxième chapitres, nous présenterons dans ce chapitre l'étude de la carte de commande de l'ascenseur.

Afin de réaliser la commande d'un ascenseur par Arduino, il nous faut une structure mécanique adéquate. Notre étude sera alors suffisamment généralisée pour tous les types d'ascenseurs, on a réalisé une maquette qui nous permet de mettre en œuvre la commande par Arduino. Cette maquette possède un moteur MCC fixé au sommet commandant une cabine à déplacement vertical.

III.2. Partie électronique :

III.2.1. Description des composants :

III.2.1.1. Le microcontrôleur (carte Arduino Mega 2560) :

La MEGA 2560 est conçu pour des projets plus complexes. Avec 54 broches E/S dont 14 PWM, 16 entrées analogiques et d'un plus grand espace pour les croquis. Cela donne à notre projet beaucoup d'espace et d'opportunités. En effet la carte Arduino Mega 2560 offre toutes les fonctionnalités de la carte UNO, mais avec des fonctionnalités supplémentaires. On retrouve notamment un nombre d'entrées et de sorties plus important ainsi que plusieurs liaisons séries. Elle est basée sur un ATmega2560 cadencé à 16 MHz [8],[13].

✓ Les caractéristiques de la Mega 2560 :

- Version : Rev.3 ;
- Alimentation : via port USB ou 7 à 12 V sur connecteur alimentation ;
- Microprocesseur : ATmega2560 ;
- Mémoire flash : 256 kB ;
- Mémoire SRAM : 8 kB ;
- Mémoire EEPROM : 4 kB ;
- 54 broches d'E/S dont 14 PWM ;
- 16 entrées analogiques 10 bits ;
- Intensité par E/S : 40 mA ;
- Cadencement : 16 MHz ;
- 3 ports séries ;
- Bus I2C et SPI

- Gestion des interruptions ;
- Fiche USB B ;
- Dimensions : 107 x 53 x15 mm [8],[13].

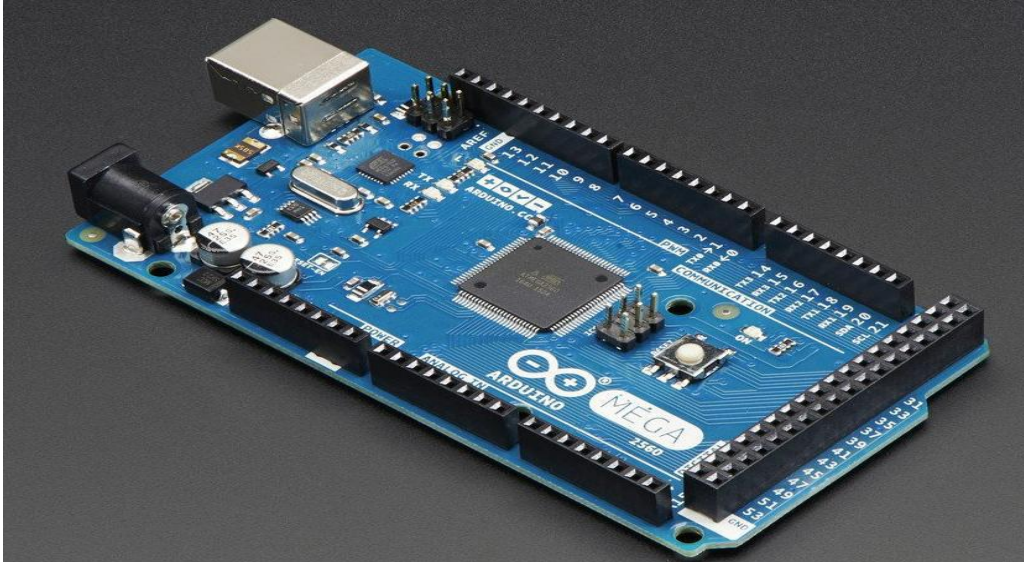


Figure III.1 : Carte Arduino Mega 2560.

III.2.1.2. Le moteur à courant continu (MCC) :

Description :

Un moteur à courant continu est constitué de deux parties électriques : le stator et le rotor. Lorsqu'on alimente le moteur, il se crée une interaction magnétique qui met le moteur en mouvement. Lorsqu'on inverse le sens de la tension qui alimente le moteur, il tourne en sens inverse.

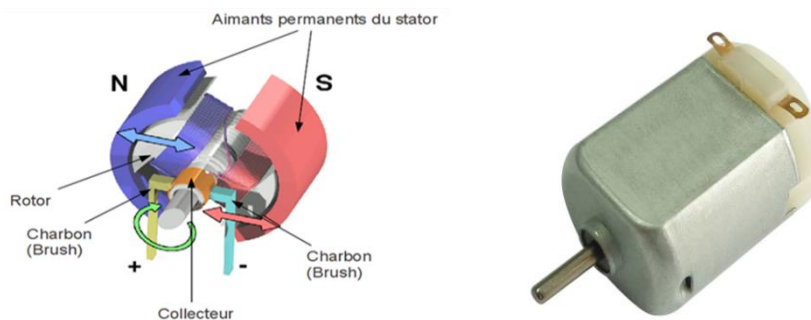


Figure.III.2 : Moteur à courant continu.

✓ Principe de fonctionnement :

Le stator : Le stator d'un moteur à courant continu est la partie fixe du moteur (statique). Le stator est aussi nommé l'inducteur ou l'excitation : on fait passer un courant dans le bobinage du stator et c'est

lui qui crée (qui induit) un champ magnétique. Le stator pose le décor pour le rotor qui se retrouve ainsi plongé dans ce champ magnétique. Le stator (inducteur) crée un champ magnétique B appelé champ inducteur, ou encore champ statorique.

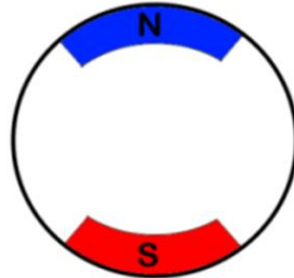


Figure.III.3 : Stator d'un moteur à courant continu.

Le rotor :

Le rotor est la partie en rotation de moteur, celui qui tourne. Il est constitué du bobinage induit. Il faut alimenter cette bobine pour la transformer en électroaimant qui entrera en interaction avec le stator. Si on n'alimente pas le rotor, il ne serait l'objet d'aucune force et ne tourne pas. Un système de frottement spécial permet d'alimenter le rotor : des balais ou charbons montés sur des ressorts frottent sur les contacts en rotation : le collecteur.

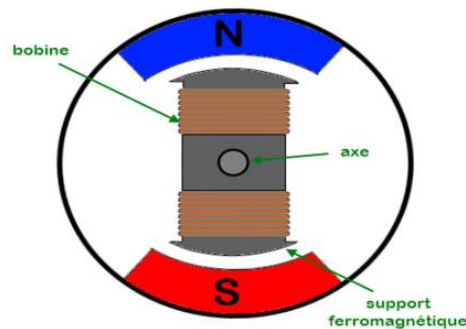


Figure.III.4 : Rotor d'un moteur à courant continu.

Le collecteur :

Le collecteur est un ensemble de plages métalliques qui font contact avec les charbons. Il appartient au rotor. Le frottement des charbons fait qu'ils s'usent : ils sont montés sur ressort pour garantir un contact même lorsqu'ils raccourcissent à cause de l'usure.

III.2.1.3. Le module L298N :

La puce L298N est un circuit intégré contient essentiellement :

- 2 ponts en H, permettant de piloter chacun 1 moteur électrique DC (dans un sens, ou dans l'autre)
- Et une logique de commande à « faible courant », pour piloter ces ponts à « fort courant »

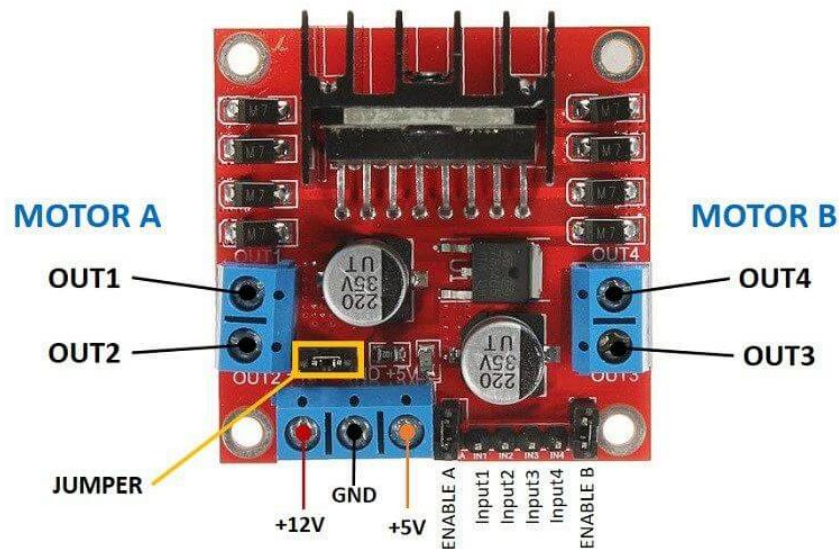


Figure III.5 : Le module L298N [22]

Le L298N requiert 2 alimentations distinctes pour fonctionner :

- Une tension pour la partie puissance (nommée V_s), qui servira à alimenter les moteurs, au travers de transistors de puissance.
- Une tension pour la partie commande (nommée V_{ss}), qui servira à alimenter toute la partie logique de commande, dont ces transistors de puissance.

Au niveau logique de pilotage, on distingue :

- Des broches d'activation des ponts (ENA et ENB) : qui permettent de lancer ou arrêter les moteurs. À noter qu'on peut alimenter ces entrées en tout ou rien (alors les moteurs « tourneront » à fond), ou en PWM, pour contrôler leur vitesse de rotation.
- Des broches de sélection des ponts (IN1, IN2, IN3, et IN4) : qui permettent de sélectionner les sens de rotation des moteurs (et comme il y a deux moteurs pilotables ici, on retrouve 4 entrées, correspondant aux 4 possibilités de sens de rotation, à raison de 2 par moteurs). [22]

✓ Principe de fonctionnement du L298N :

Les Ponts en H ne sont en fait qu'un nom donné à une configuration particulière, de composants électroniques. Classiquement, ceux-ci sont composés de 4 transistors, disposés d'une certaine manière, formant visuellement la lettre « H ».

Le principe d'un pont en H est super simple : en activant les transistors deux par deux (ceux de sens opposés), on peut contrôler le sens du passage du courant dans la charge, branchée « au milieu du H ». Et c'est ce changement de sens de courant, qui permet aux moteurs à courant continu de pouvoir tourner « à l'endroit », ou « à l'envers ». Visuellement, voici comment cela se passe :

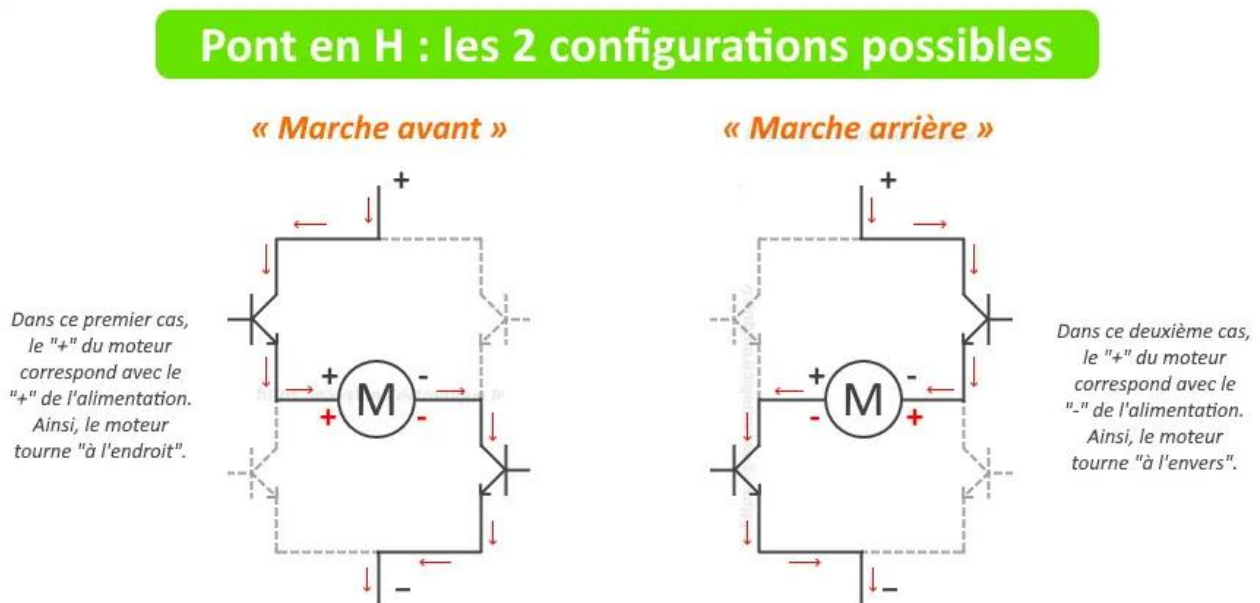


Figure III.6 : Pont en H : les 2 configurations possibles. [22]

- Dans le 1er cas : le « + » du moteur sur le « + » de l'alimentation, et le « - » du moteur sur le « - » de l'alimentation.
- Dans le 2ème cas : le « + » du moteur sur le « - » de l'alimentation, et le « - » du moteur sur le « + » de l'alimentation (faisant ainsi tourner le moteur en sens inverse).

✓ Ces caractéristiques :

- Tension logique 5V
- Tension d'entraînement 5V-35V
- Courant logique 0mA-36mA
- Courant d'entraînement 2A (MAX single bridge) [22].

III.2.1.4. Afficheur 7 segments :

Description

Comme son nom l'indique, l'afficheur 7 segments possède 7 segments. C'est un composant qui permet d'afficher les chiffres de 0 à 9. Ils sont encore utilisés aujourd'hui malgré l'importante présence du LCD sur le marché de l'affichage électronique. Ces afficheurs sont adaptés lorsqu'un large affichage numérique est requis comme pour les écrans dans les gares, les montres, calculatrices, ascenseurs ...etc. En plus de cela ils sont économiques, robustes et fiables comparés aux LCD [15], [16].

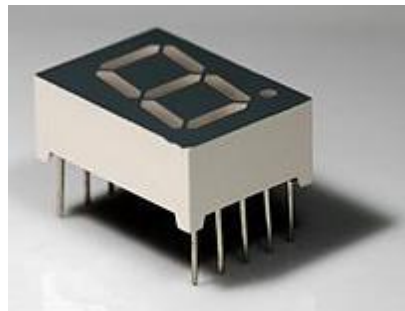


Figure III.7 : Afficheur 7 segments.

Fonctionnement :

Un segment est en fait une LED plate, et les 7 sont arrangées en forme de huit. Pour disposer les chiffres il faut allumer certains segments, et en laisser d'autres éteints. Ces segments pour être identifiés facilement sont associés à des lettres :

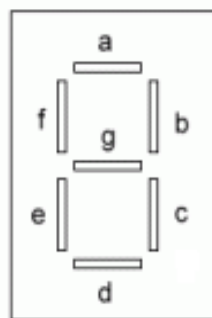


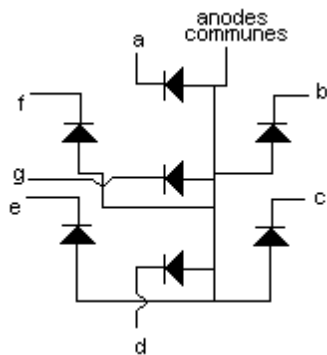
Figure III.8 : Afficheur 7 segments identifier avec des lettres.

Si l'on s'intéresse de plus près au brochage, on s'aperçoit que chaque LED possède une de ses extrémités connectée à une broche commune. Lorsque toutes les anodes des LED sont mises en commun on parle d'un 7 segment à anode commune (Common anode), et cathode commune

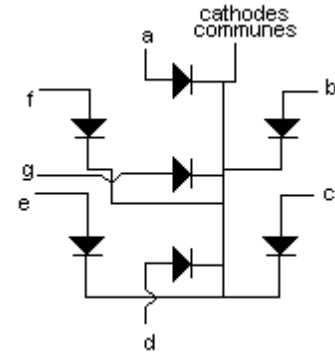
(Common cathode) lorsque les LED ont leur cathode en commun. Les boîtiers de ces équipements sont donc équipés de 10 broches (8 broches + 1 commune + alimentation + masse).

Afficheur à anode commune : toutes les anodes sont reliées et connectées au potentiel haut. La commande du segment se fait par sa cathode mise au potentiel bas.

Afficheur à cathode commune : toutes les cathodes sont reliées et connectées au potentiel bas. La commande du segment se fait par son anode mise au potentiel haut.



a. Schéma anode commune.



b. Schéma cathode commune.

Figure III.9 : Types d'afficheur 7 segments.

Par exemple, pour afficher le chiffre 1, il faut allumer les segments (LED) B et C, et pour afficher le chiffre 2, il faut allumer les segments A, B, G, E et D. Même principe pour les autres chiffres.

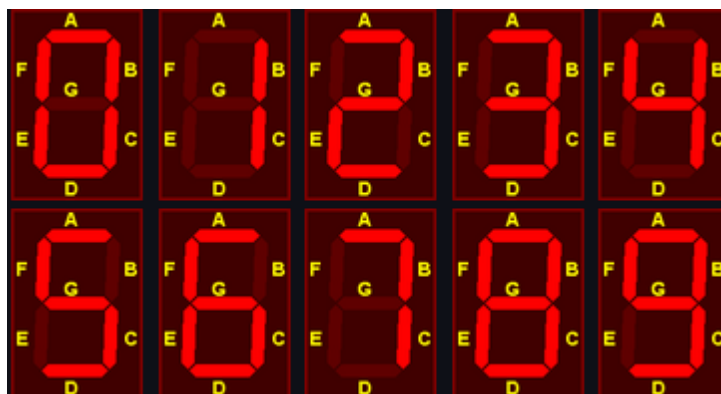


Figure III.10 : La visualisation des chiffres. [15]

Comme chaque segment est une LED, une résistance de limitation de courant doit être ajoutée en série avec chacun d'eux. Le calcul de la valeur de cette résistance de limitation est le même que celui à appliquer pour les LED "ordinaires" [15].

III.2.1.5. Les capteurs de position :

Dans les ascenseurs les capteurs de position détecte la position de la cabine [11].L'information est transmise à l'unité de traitement afin d'en tenir compte dans la gestion du mouvement de l'ascenseur, dans ce projet on utilise des capteurs de présence type infrarouge, nous avons monté un capteur par étage.



Figure III.11 : Types des capteurs de présence : A- fin de course ; B- capteur infrarouge ; C- capteurs magnétiques.

III.2.1.5.1. capteur infrarouge :

Est un capteur qui se présente comme un équipement électronique capable de réagir au rayonnement infrarouge renvoyé par les objets qui lui font face. Ces rayons se situent en dehors du spectre de la lumière visible.

Le capteur infrarouge ou le capteur IR a deux parties principales. Émetteur IR et récepteur IR. La responsabilité de l'émetteur IR ou de l'émetteur infrarouge est la transmission des ondes infrarouges tandis que le travail du récepteur IR est de recevoir ces ondes infrarouges. Le récepteur IR envoie constamment des données numériques sous la forme de 0 (LOW) ou 1 (HIGH) à la broche V out du capteur [18].

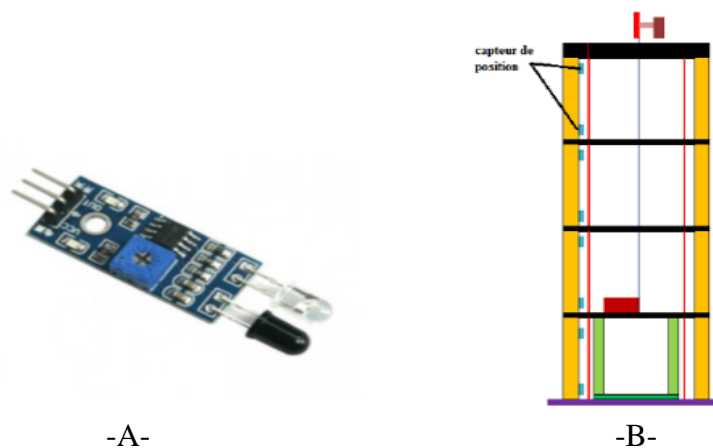


Figure III.12 : -A- capteur de présence ; B- placement des capteurs

✓ Principe de fonctionnement

L'émetteur infrarouge envoie un signal infrarouge qui, en cas de surface réfléchissante (par exemple de couleur blanche), rebondit dans certaines directions, y compris celle du récepteur infrarouge qui capte le signal détectant l'objet.

Lorsque la surface est absorbante (par exemple de couleur noire), le signal IR n'est pas réfléchi et l'objet ne peut pas être détecté par le capteur. Ce résultat se produirait même si l'objet est absent.

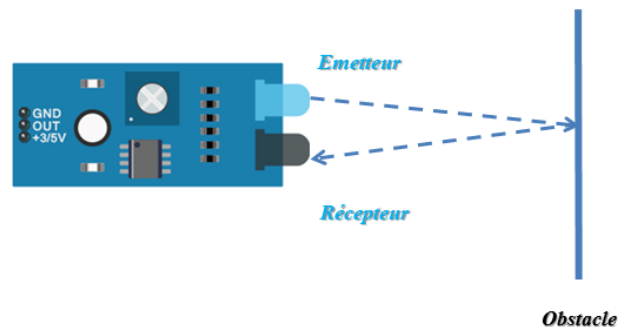


Figure III.13 : les ondes IR (infrarouge). [18]

✓ Caractéristiques Techniques

- Voltage de fonctionnement : DC 3,3 V-5 V
- Intensité du courant $\geq 20\text{mA}$
- Température de fonctionnement : $-10\text{ °C} \rightarrow +50\text{ °C}$
- Plage de distance : 2-40 cm
- Interface IO : interface 4 fils (GND / Vcc / OUT / EN)
- Signal de sortie : tension TTL
- Mode d'hébergement : régulation de résistance multi-cercle
- Angle effectif : 35°
- Taille : $41,7 * 16,7\text{ mm}$
- Poids : 5g

✓ les applications du capteur Infrarouge :

- Voiture intelligent.
- Eclairage automatique.
- Robinet intelligent.
- Porte intelligente [18].

III.2.1.6. Bouton poussoir :

✓ **Définition :**

Le bouton poussoir ou interrupteur poussoir, est la base de l'interactivité, entre l'homme et la machine. Il permet de commander un objet, une machine ou simplement une lumière.



Figure III.14 : Bouton-poussoir. [19]

Il en existe de deux types, les boutons poussoirs à fermeture et les boutons poussoirs à ouverture. Ils servent à ouvrir ou fermer un circuit électrique, dès qu'on les relâche ils reviennent dans leur position [19].

✓ **Symbolisation :**

Symbole bouton poussoir normalement fermé

Symbole bouton poussoir normalement ouvert

NC (normaly closed)

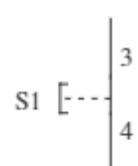
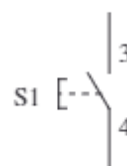
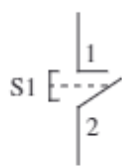
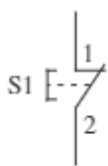
NO (normaly open)

Au repos

Actionné

Au repos

Actionné



Lorsque l'on appuie sur le bouton poussoir, celui-ci laisse passer le courant, et lorsqu'il revient à sa position initiale (simplement en lâchant la pression), il ne laissera plus passer le courant. C'est un bouton poussoir de type "NO".

Et inversement, lorsque l'on appuie sur le bouton, celui-ci ne laisse pas passer le courant, lorsque l'on relâche celui-ci laisse passer le courant. Il est de type "NC" [19].

III.3. Partie mécanique :

III.3.1. Présentation de prototype :

Le prototype est un système (ascenseur) à 3 niveaux (R+2), réduit d'environ 60 cm d'hauteur, 40 cm de large et 15 cm d'épaisseur. Il a été réalisé à partir d'une ossature en bois. Il est constitué de :

- une cabine d'environ 10cm de longueur, 10 cm de large.
- un fil pour la traction de la cabine.
- un afficheur 7 segments.
- Un capteur de présence type infrarouge à chaque niveau pour l'indication d'étage.
- un bouton poussoirs à chaque niveau.
- La montée et la descente de la cage d'ascenseur sont réalisées par la rotation d'un moteur MCC.

III.4. Partie logicielle :

III.4.1. Présentation de Proteus :

Cette suite logicielle est très connue dans le domaine de l'électronique. Elle est utilisée dans de nombreuses entreprises et organismes de formation. Outre la popularité de l'outil, Proteus possède d'autres avantages :

- Pack contenant des logiciels faciles et rapides à comprendre et à utiliser ;
- Support technique performant ;
- Outil de création de prototype virtuel permettant de réduire les coûts matériel et logiciel lors de la conception d'un projet [17].

III.4.1.1. ISIS :

Le module ISIS de Proteus est principalement utilisé pour éditer un schéma structurel d'un circuit électronique (assemblage de composants électroniques dont on fixe les valeurs et les références) reliés par des connexions électrique (fils). Par ailleurs, le logiciel permet également de simuler les différents types de montages ce qui permet de déceler certaines erreurs dès l'étape de conception. Indirectement, les circuits électriques conçus grâce à ce logiciel peuvent être utilisé dans des documentations car le logiciel permet de contrôler la majorité de l'aspect graphique des circuits. la figure ci-dessous montre la fenêtre du module de simulation Proteus ISIS [17].

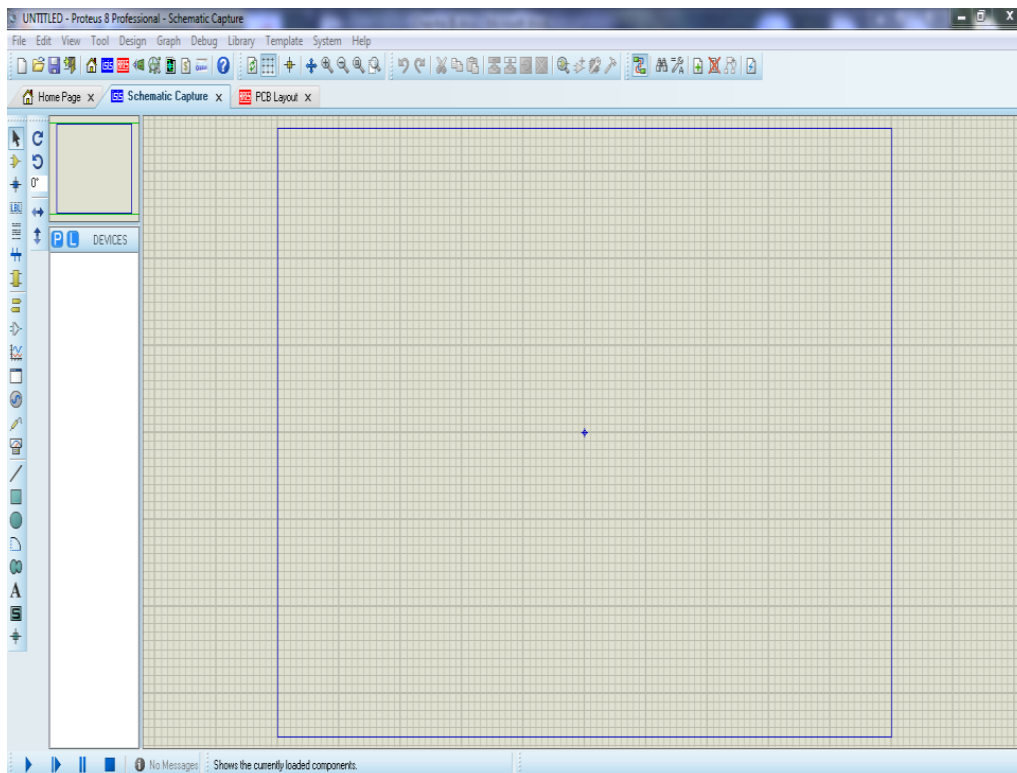


Figure III.15 : Fenêtre du module de simulation Proteus ISIS. [17]

III.5. Conclusion :

Dans ce chapitre nous avons donné la description des composants à utiliser dans la réalisation et leurs caractéristiques ; suivi d'une présentation logiciel de simulation Proteus ISIS.

Après avoir eu la liste de matériels et logiciels dont on a besoin pour le projet et comprendre leurs fonctionnements, il nous reste qu'assembler ces composants de façons raisonnable pour réaliser un prototype de commande d'un ascenseur, c'est ce qu'on va voir dans le chapitre suivant.

Chapitre 4

Réalisation et Validation

IV.1. Introduction :

Ce dernier chapitre présente la réalisation de la carte de commande d'ascenseur, et les différentes étapes de programmation de cette maquette ; ainsi que les différents organigrammes des différents programmes.

IV.2. Principe de fonctionnement :

Un usager présent devant l'ascenseur, fait appel à ce dernier si la cabine n'est pas à son niveau, la carte Arduino reçoit ces commandes d'appels, Le processeur traite toutes ses variables, et génère en conséquence les signaux de commandes pour le moteur à courant continu.

Le processeur détermine la position courante de la cabine grâce aux signaux générés par les capteurs de présence (capteurs IR) et les affiche à l'aide d'un afficheur 7 segment. L'arrêt ou non de la cabine à un étage dépend du calcul de priorité élaboré par l'algorithme.

IV.3. Conception de la carte :

Cette partie se base sur la conception de 3 (trois) circuit ; un circuit de puissance pour la commande du moteur MCC, un circuit d'affichage avec l'afficheur 7 segment et un circuit pour les capteurs IR et les boutons poussoirs, et pour l'alimentation j'ai utilisé 2 batteries rechargeables de 3.7V.

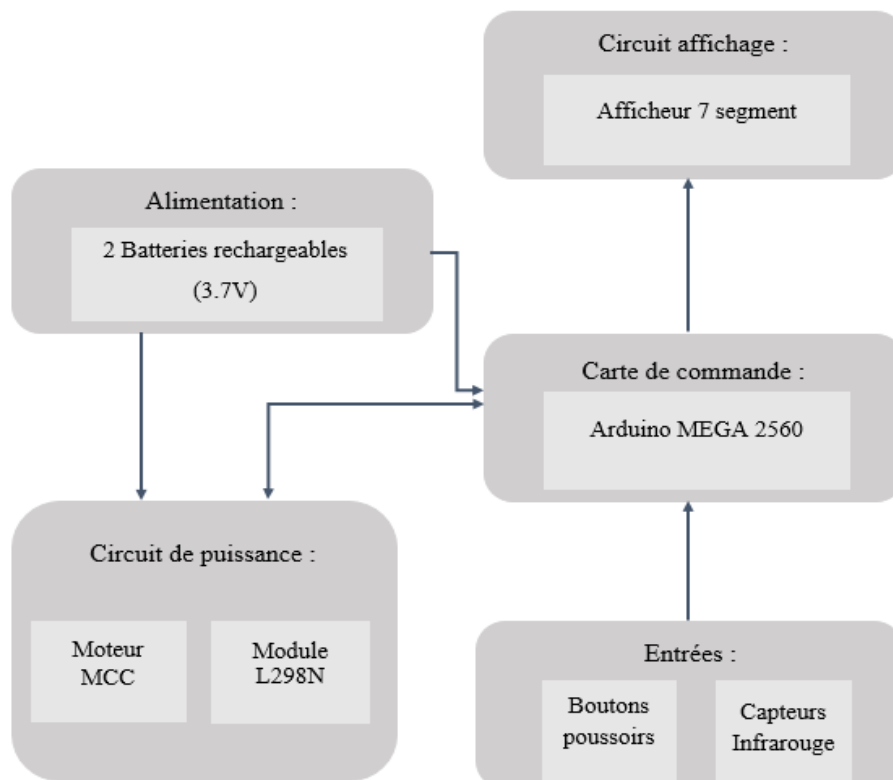


Figure IV.1 : Schéma synoptique de la carte de commande.

IV.3.1. Circuit d’affichage :

Cette figure représente le branchement avec la simulation de l’afficheur 7 segment anode commune avec un décodeur 74Is47 à la carte ARDUINO, pour cela on a utilisé les pines 3, 4,5 et 6.

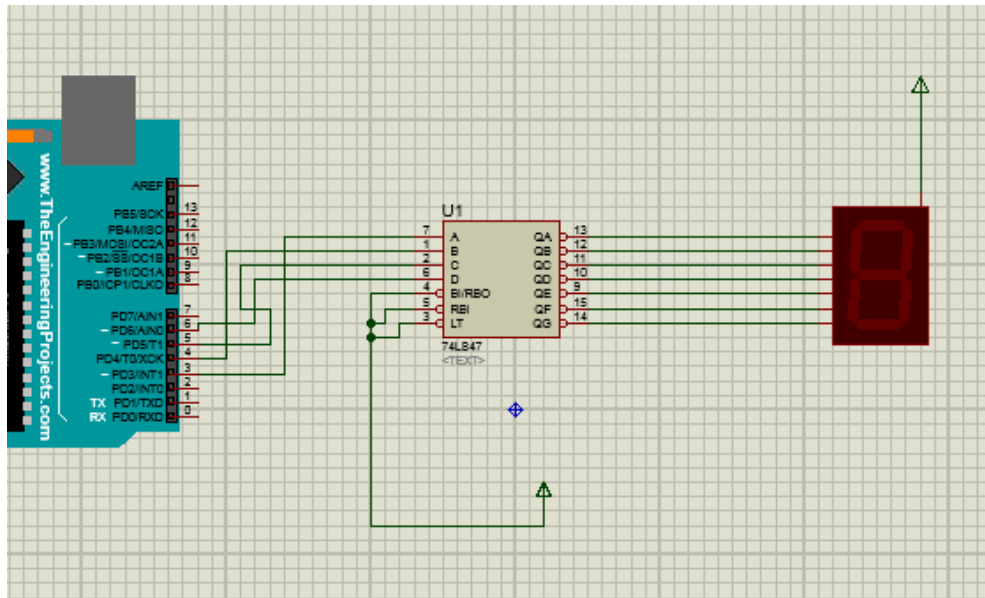


Figure IV.2 : Schéma de conception de circuit d’affichage sur ISIS.

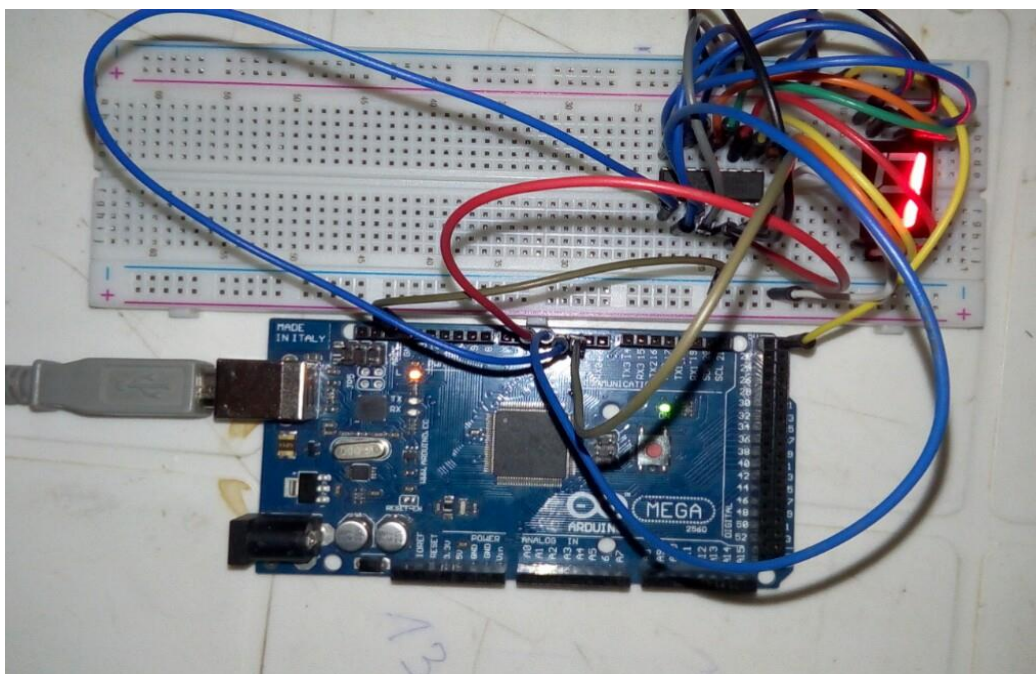


Figure IV.3 : circuit d’affichage.

IV.3.4. Branchement des boutons poussoirs et Leds :

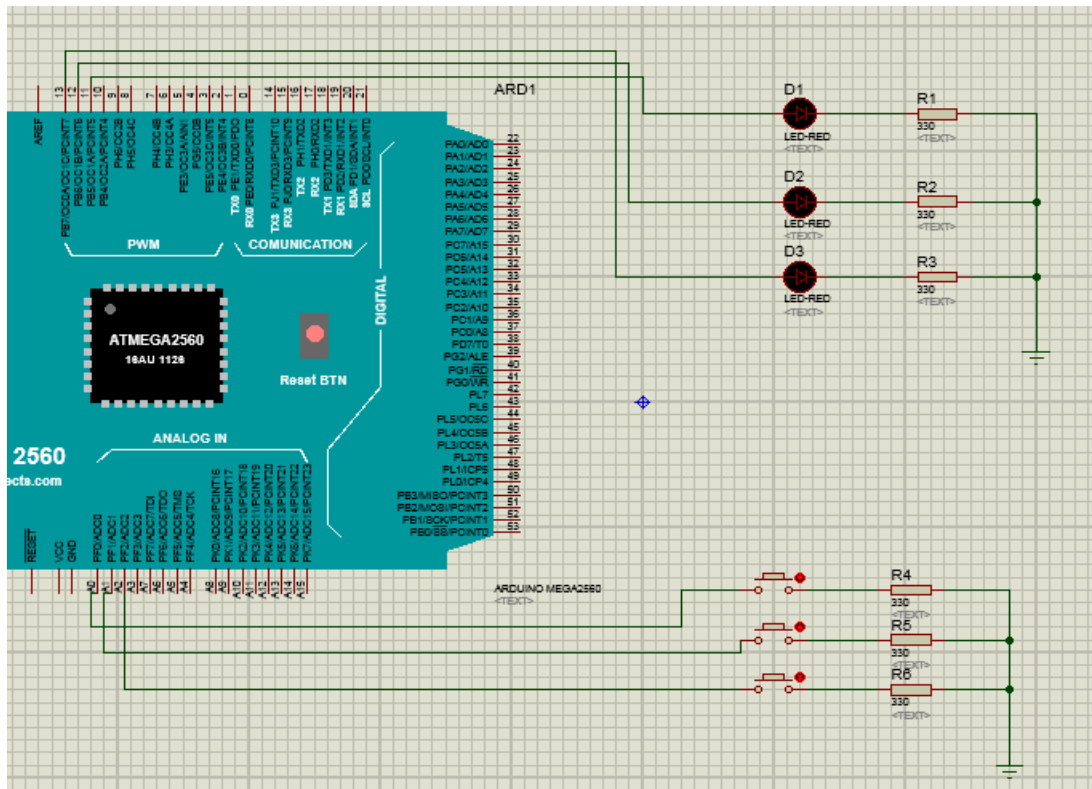


Figure IV.6 : Branchement des boutons poussoirs et leds.

L'ensemble de 3 boutons poussoirs : ils simulent les boutons d'appels de l'ascenseur au niveau des paliers. Lorsque l'un de ces 3 boutons est actionné, une led correspondante s'allume immédiatement pour indiquer à l'utilisateur que sa demande est prise en compte.

IV.3.5. Schéma de la carte de commande :

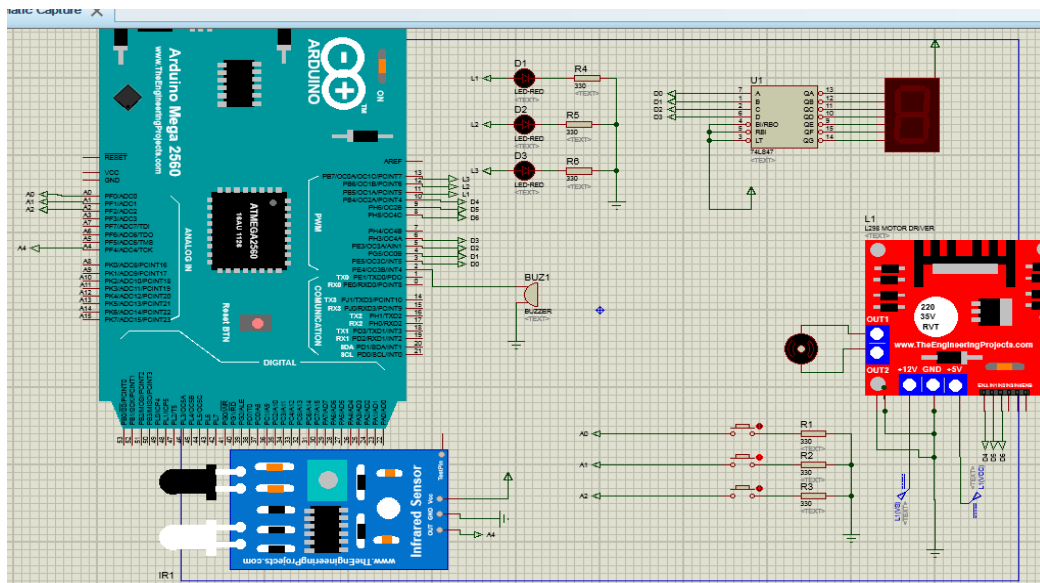


Figure IV.7 : Schéma globale de la carte de commande.

IV.4. Programmation :

Premièrement en alimente le système, Un appel émanant d'un étage quelconque se réalise comme suit :

- Si l'appel vient de l'étage où se trouvent l'ascenseur, un bip sonore et un affichage de 7 segments indique la présence de la cabine à l'étage demandé.
- S'il vient d'un autre étage, alors l'ascenseur monte ou descend vers l'étage demandé, l'afficheur indique le niveau d'étage au même temps.

IV.4.1. Réalisation du programme :

Vue la complexité de programme nous l'avions partagé en un ensemble de sous-programme, qu'on peut appeler dans le programme principale.

IV.4.1.1. Organigramme de déroulement du programme :

A. Programme principale

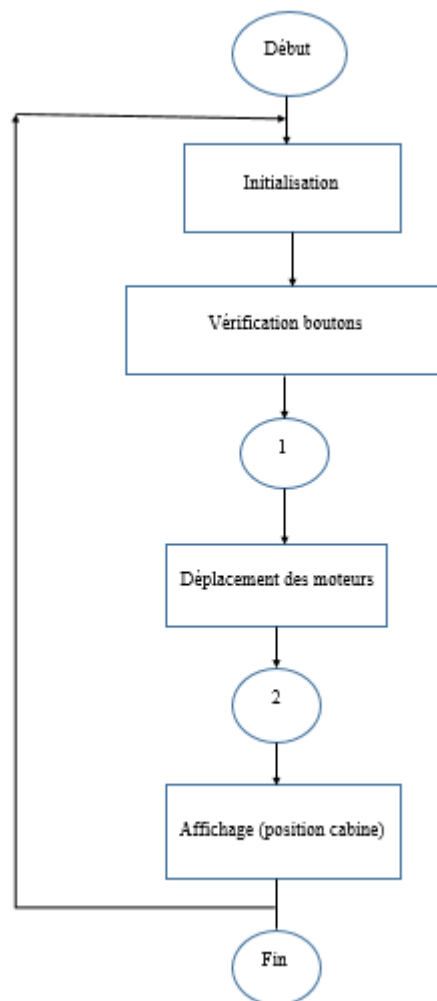


Figure IV.8 : Programme principale

B. Sous-programme "vérification boutons " :

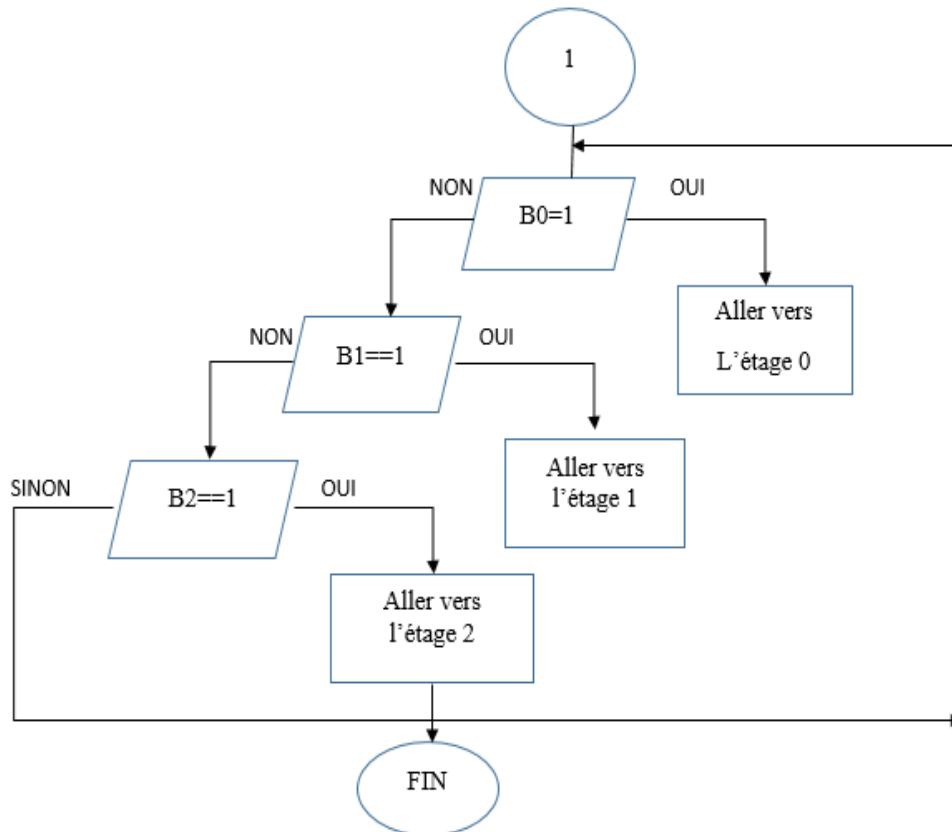


Figure IV.9 : Sous-programme "vérification boutons "

C. Sous-programme " déplacement moteur

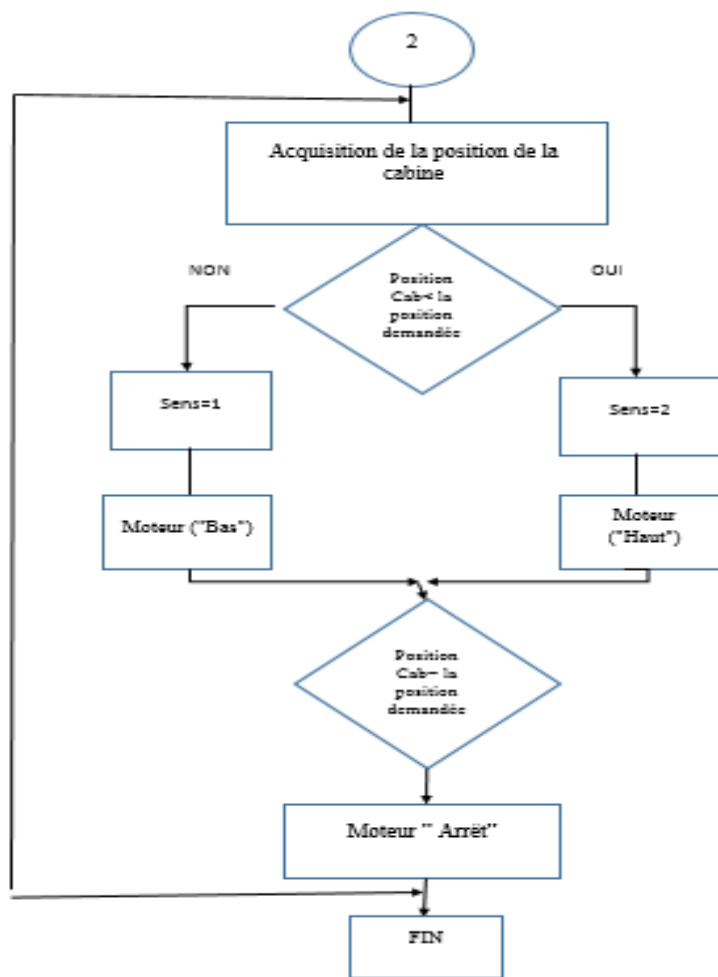


Figure IV.10 : Sous-programme " déplacement moteur

D. Sous-programme : détection de la position :

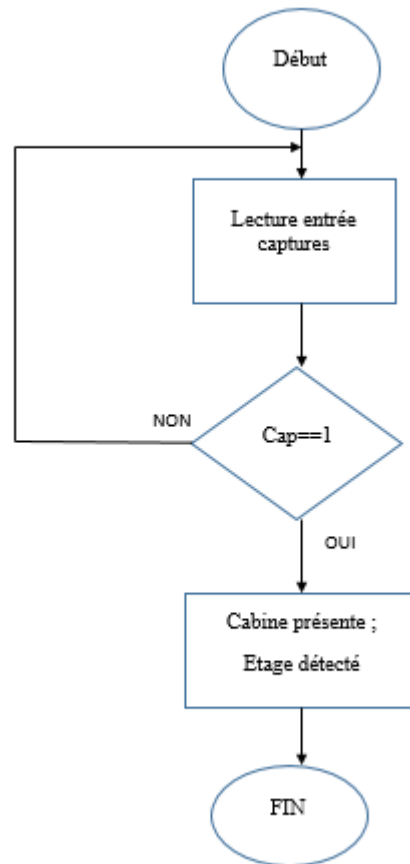


Figure IV.11 : Sous-programme : détection de la position

IV.5. Description de la maquette :

IV.5.1. La partie externe :

Elle est composée de bois, la cage d'ascenseur, dont nous pouvons distinguer 3 étages, nommés plus tard étages 0, 1 et 2 et d'un module mobile, qui peut monter et descendre, qui représente l'ascenseur. On trouve également un panneau de commande comportant 3 boutons poussoirs, permettant à l'utilisateur de donner ses instructions. Pour compléter la demande, on a ajouté par la suite un afficheur 7 segments, indiquant en permanence l'étage actif.

IV.5.2. La partie interne :

Pour faire fonctionner les tout, divers éléments opératifs sont installés en coulisses, d'abord un moteur, que l'on nous pouvons faire tourner dans le sens horaire ou antihoraire, enroulant ou déroulant un fil fixé à l'ascenseur, donnant ainsi un mouvement au module.

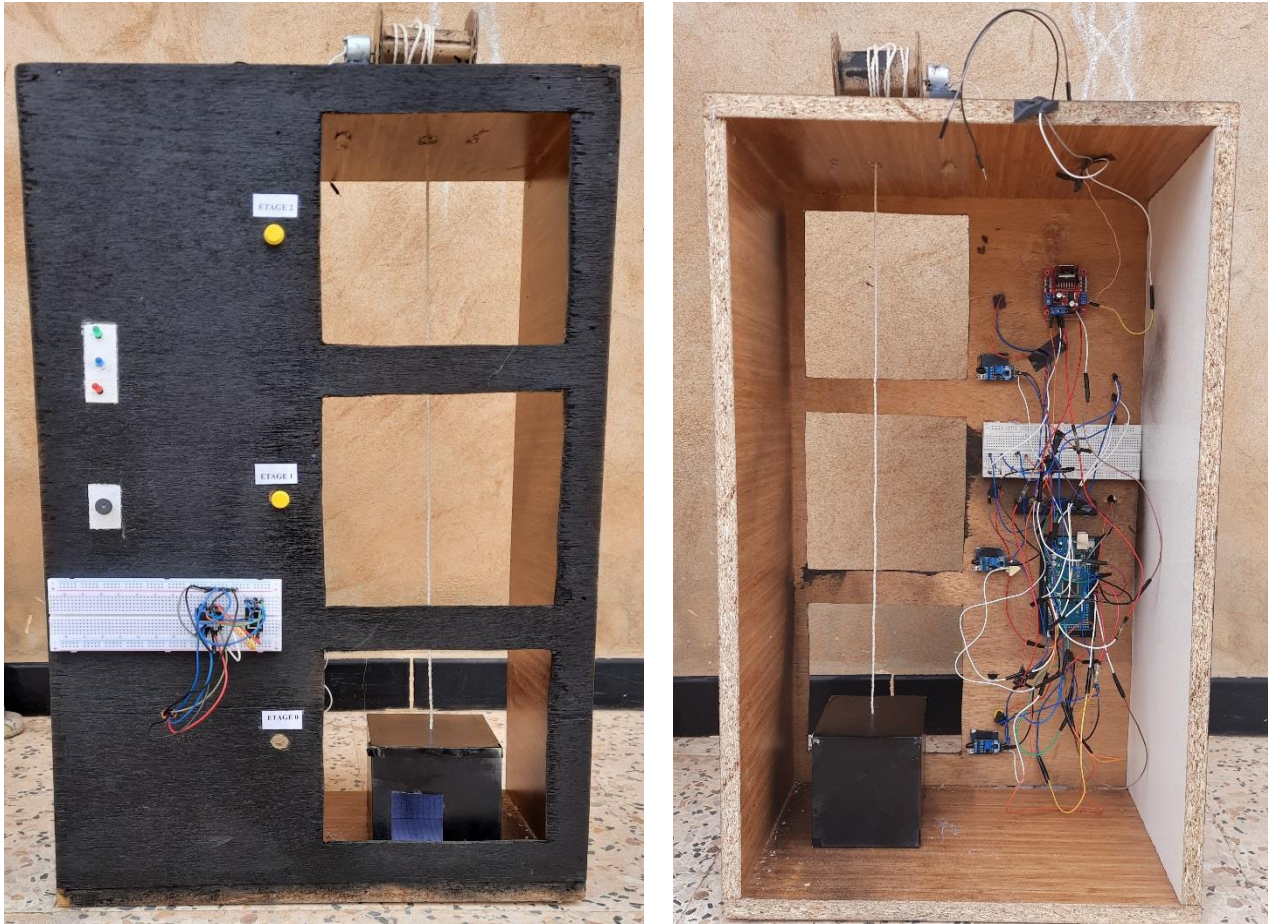


Figure IV.12 : Réalisation final.

IV.6. Conclusion :

Ce chapitre était le noyau de notre projet de fin d'étude. Dans ce dernier, nous avons illustré le fonctionnement de prototype ainsi la configuration de tout l'ensemble de matériels par des schémas de câblage électronique.

Conclusion

Générale

Conclusion générale

L'objectif initial de ce travail était la réalisation d'une carte de commande d'un ascenseur didactique à base d'une carte Arduino Mega 2560 et d'adopter une méthodologie permettant d'étudier les différents étapes de fonctionnement de ces dispositifs.

Ce projet nous a permis de développer nos connaissances et de mettre en application notre savoir-faire pour la concrétisation d'un projet de fin d'études. La recherche bibliographique a été riche en découverte sur les études et l'industrie des ascenseurs.

Malheureusement avec la complexité de nos programmes (en taille et en nombres de fonctions), il est devenu très difficile pour ajouter d'autres fonctions au programme tel que le système d'interruption ou le développement de la gestion des priorités des appels. En effet nous envisagerons d'élargir en termes d'application ce projet tout en ajoutant quelques options pour la maquette. On propose les améliorations suivantes :

- ✓ Ajouter des capteurs de sécurité en cas de dépassement de la course normale ;
- ✓ Automatiser l'ouverture et la fermeture des portes ;
- ✓ Interdire le déplacement de la cabine en cas où les portes sont ouvertes ;
- ✓ Ajouter des boutons de sélection d'étage (boutons d'envoi) ;

Bibliographie

- [1] : GPEM/ME (Groupe permanent d'étude des marchés de matériels mécaniques, électriques et électroniques) «Ascenseurs et montes charges électriques, fournitures et installation ». Journal officiel de la république Française N° 5653-1 édition, Juillet 1990.
- [2] : ZIDANE Taous, «Développement d'un outil numérique destiné à l'étude de performances dynamiques d'un entraînement électrique : Application aux ascenseurs» Projet de fin d'étude en électrotechnique, Université Mouloud MAMMERI, TIZI-OUZOU, 2010.
- [3] : Jean-Dominique Mosser, Yves Granjon, Jacques Tanoh «Sciences Industriel pour l'ingénieur» 1^{er} année MPSI-PCSI-PTSI Dunod, Paris, 2008, 296p, ISBN 978-2-10-053788-4.
- [4] : M. Lbouhmadi, J. Laayoun «Etude d'ascenseur commandé par automate programmable » Projet de fin d'étude, Université sidi Mohammed ben Abdallah, Fès, Maroc 2007.
- [5] : MOMO FRANCK HERMANN, « Etude Et Réalisation D'un Prototype De Commande D'un Ascenceur », 2015.
- [6] : <https://energieplus-lesite.be>, consulté le 22/08/2021.
- [7] : BOUGUETOF.H, BENTCHAKAL.D, BENSENOUCI.S, «Etude d'un ascenseur», Projet de fin d'étude en ETH, UMMTO, 2002.
- [8] : SIMON Landranlt, HIPPOLYTE Weisslinger, «Premier pas en informatique embarquée», Edition CC BY-NC-SA, 01 Juin 2014, 268p.
- [9] : Christian Tavenir, «Arduino, maîtrisez sa programmation et ces cartes d'interface (Schields)», Collection Dunod, Mars 2014, 256p.
- [10] : KRAMA Abdelbaset, GOUGUI Abdelmoumen, «Etude et réalisation d'une carte de contrôle par Arduino via le système Androïde », projet de fin d'étude Université KASDI Merbah, Ouargla, 2015.
- [11]: Massimo Banzi, Michael Shiloh «Getting started with Arduino », 3rd Edition, Décembre 2014, 245p.
- [12]: Brian W. Evans «Arduino Programming Notebook », First edition, August 2007, 38p.
- [13] : <https://www.arduino.cc>, consulté le 01/09/2021.
- [14] : ISSAM KHELIL, « Commande d'un ascenseur par Arduino », Projet de fin d'étude en Automatique et Système, UNIVERSITE BADJI MOKHTAR, ANNABA, 2018.
- [15] : Cours et montages d'électronique : <http://www.elektronique.fr>, consulté en Sep 2021.
- [16]: Electronic Components Datasheet Search: <https://www.alldatasheet.com>, consulté en Aout 2021.

- [17] : Guide d'utilisation de logiciel Proteus : <http://projet.eu.org>, consulté en Aout 2021.
- [18] : <https://zineddine-academie.blogspot.com>, consulté en Aout 2021.
- [19] : Les Boutons poussoirs : <https://docplayer.fr>, consulté en Sep 2021.
- [20] : Utiliser un afficheur 7 segments avec une carte Arduino / Genuino, 2016
<https://www.carnetdumaker.net>, consulté le 08/09/2021.
- [21] : ISSAOUNI .S, Cours « AUTOMATES PROGRAMMABLES INDUSTRIELS », M2 – ESE, Université AKLI MOHAND OULHAJ, Bouira, 2021.
- [22] : Module L298N : <https://passionelectronique.fr/tutoriel-l298n/>.

ANNEXE

Script Arduino:

```
int OUT_Motor_1=9;
// To IN1 Motor Controlled
int OUT_Motor_2=8;
// To IN2 Motor Controlled
-----

int floor_0=A0;
int floor_1=A1;
int floor_2=A2;
-----

int sensor_0=A4;
//IR Sensor 0
int sensor_1=A5;
//IR Sensor 1
int sensor_2=A6;
//IR Sensor 2
-----

int speed_ = 100 ;
-----

// 3 LED
int LEDrouge0=11;
int LEDbleu1=12;
int LEDvert2=13;
-----

void setup() {
  pinMode(LEDrouge0,OUTPUT);
  pinMode(LEDbleu1,OUTPUT);
  pinMode(LEDvert2,OUTPUT);
  pinMode(sensor_0,INPUT_PULLUP);
  pinMode(sensor_1,INPUT_PULLUP);
  pinMode(sensor_2,INPUT_PULLUP);
  pinMode(floor_0,INPUT_PULLUP);
```

```
  pinMode(floor_1,INPUT_PULLUP);
  pinMode(floor_2,INPUT_PULLUP);
  pinMode(OUT_Motor_1,OUTPUT);
  pinMode(OUT_Motor_2,OUTPUT);
  pinMode(3,OUTPUT);
  pinMode(4,OUTPUT);
  pinMode(5,OUTPUT);
  pinMode(6,OUTPUT);
-----

  analogWrite(OUT_Motor_2,HIGH);
  delay(500); }
-----

void loop() {
  if ( digitalRead(floor_0) ==0 ){
    digitalWrite(LEDrouge0,HIGH);
    digitalWrite(LEDbleu1,LOW);
    digitalWrite(LEDvert2,LOW);
    if ( digitalRead(sensor_0) ==1){
      analogWrite(OUT_Motor_1 ,speed_ );
      while(digitalRead(sensor_0)){ }
      digitalWrite(OUT_Motor_1 ,0); }
    digitalWrite(LEDrouge0,LOW);
    digitalWrite(LEDbleu1,LOW);
    digitalWrite(LEDvert2,LOW); }
  }

  ////////////
  if ( digitalRead(floor_2) ==0 ){
    digitalWrite(LEDrouge0,LOW);
    digitalWrite(LEDbleu1,LOW);
    digitalWrite(LEDvert2,HIGH);
    if ( digitalRead(sensor_2) ==1){
      analogWrite(OUT_Motor_2 ,speed_);
```

```

while(digitalRead(sensor_2)){
    delay(400);
    digitalWrite(OUT_Motor_2 ,0);    }
digitalWrite(LEDrouge0,LOW);
digitalWrite(LEDbleu1,LOW);
digitalWrite(LEDvert2,LOW); }
if ( digitalRead(floor_1)==0 ){
    digitalWrite(LEDrouge0,LOW);
    digitalWrite(LEDbleu1,HIGH);
    digitalWrite(LEDvert2,LOW);
    if ( digitalRead(sensor_1) ==1){
        //////////
        if ( digitalRead(sensor_2) ==0){
            analogWrite(OUT_Motor_1 ,speed_);
            while(digitalRead(sensor_1)){ }
            digitalWrite(OUT_Motor_1 ,0);    }
        //////////
        else if (digitalRead(sensor_0)==0 ){
            analogWrite(OUT_Motor_2 ,speed_);
            while(digitalRead(sensor_1)){ }
            delay(400);
            digitalWrite(OUT_Motor_2 ,0);    }
        else {
            digitalWrite(OUT_Motor_1,0);
digitalWrite(OUT_Motor_2 ,0);
        } }
digitalWrite(LEDrouge0,LOW);
digitalWrite(LEDbleu1,LOW);
digitalWrite(LEDvert2,LOW);
}
if(digitalRead(sensor_0) ==0){
    tone (2,440);
    delay(50); }
if( digitalRead(sensor_1) ==0){
    tone (2,440);

```

```

    delay(50); }
if ( digitalRead(sensor_2) ==0){
    tone (2,440);
    delay(50); }
else { noTone(2); }
    if ( digitalRead(sensor_0) ==0){
        zero();}
if ( digitalRead(sensor_1) ==0){
    un();}
if ( digitalRead(sensor_2) ==0){
    deux();} }
void zero(){
    digitalWrite(3,LOW);
    digitalWrite(4,LOW);
    digitalWrite(5,LOW);
    digitalWrite(6,LOW);
}
void un(){
    digitalWrite(3,HIGH);
    digitalWrite(4,LOW);
    digitalWrite(5,LOW);
    digitalWrite(6,LOW);
}
void deux(){
    digitalWrite(3,LOW);
    digitalWrite(4,HIGH);
    digitalWrite(5,LOW);
    digitalWrite(6,LOW);
}

```

المخلص

المصعد عبارة عن جهاز متحرك يسمح بحركة الأشخاص أو الأشياء في حجرة على محور عمودي محدد مسبقاً داخل مبنى متعدد الطوابق. الهدف من هذه الأطروحة هو دراسة وإنتاج نموذج أولي للتحكم في المصعد بواسطة أردو وينو ميجا 2560، خاصة وأن المصعد هو نظام آلي مثير للاهتمام وأن تحقيقه يتطلب العديد من المجالات التكنولوجية (آلي، كمبيوتر، ميكانيكي، إلخ)، علاوة على ذلك، فهي وسيلة نقل مستخدمة للغاية وأكثر انتشاراً.

تم تلخيص هذا العمل في أربعة فصول أساسية: يحتوي الفصل الأول على مراجعة ببليوغرافية للمصعد مثل: التاريخ التعريف، أنواع المصاعد، إلخ. والثاني مخصص لدراسة لوحة أردو وينو وبرامج البرمجة أردو وينو. يقدم الفصل الثالث وصفاً للمكونات المستخدمة وخصائصها؛ بالإضافة إلى عرض برنامج محاكاة. الفصل الرابع والأخير مكرس لتحقيق بطاقة التحكم في المصعد والمراحل المختلفة لبرمجة هذا النموذج.

Résumé

L'ascenseur est un dispositif mobile permettant le déplacement des personnes ou des objets dans une cabine sur un axe vertical prédéfini au sein d'une construction à plusieurs étages. Ce mémoire a pour objectif d'étudier et de réaliser un prototype de commande d'un ascenseur par Arduino MEGA 2560, surtout que l'ascenseur est un système automatisé intéressant et que sa réalisation fait appel à plusieurs domaines technologiques (automatique, informatique, mécanique, etc), de plus, c'est un moyen de déplacement très utilisé et de plus en plus répandu.

Ce travail se résume dans quatre chapitres essentiels : Le premier chapitre contient une revue bibliographique de l'ascenseur comme : historique, définition, type de l'ascenseur...etc. le deuxième est consacré à l'étude de la carte Arduino et logiciel de programmation (IDE Arduino). Le troisième chapitre présente la description des composants utilisés et leurs caractéristiques ; ainsi que la présentation de logiciels de simulation Proteus ISIS. Le quatrième et dernier chapitre est consacré à la réalisation de la carte de commande d'ascenseur et les différentes étapes de programmation de cette maquette.

Abstract

The elevator is a mobile device allowing the movement of people or objects in a car on a predefined vertical axis within a multi-storey construction. The objective of this thesis is to study and produce a prototype for controlling an elevator by Arduino MEGA 2560, especially since the elevator is an interesting automated system and that its realization calls for several technological fields (automatic, computer, mechanical, etc), moreover, it is a means of transport very used and more and more widespread.

This work is summarized in four essential chapters: The first chapter contains a bibliography review of the elevator such as: history, definition, type of elevator, etc. the second is devoted to the study of the Arduino board and programming software (Arduino IDE). The third chapter presents the description of the components used and their characteristics; as well as the presentation of Proteus ISIS simulation software. The fourth and final chapter is devoted to the realization of the elevator control card and the various stages of programming this model.