



الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي والبحث العلمي
Ministère de l'Enseignement Supérieure et de la Recherche Scientifique

جامعة وهران 2 محمد بن أحمد
Université d'Oran 2 Mohamed Ben Ahmed

معهد الصيانة و الأمن الصناعي
Institut de Maintenance et de Sécurité Industrielle

Département de Maintenance en Instrumentation

MÉMOIRE

Pour l'obtention du diplôme de Master

Filière : Génie Industriel
Spécialité : Génie Industriel

Thème

Conception et réalisation d'un pilulier automatique basé sur une carte Arduino

Présenté et soutenu publiquement par :

Nom : AITEMRAR

Nom : HIRECHE

Prénom : Abdelkader

et

Prénom : Abderrahmane

Devant le jury composé de :

Nom et Prénom	Grade	Etablissement	Qualité
Tari noria	MCB	IMSI-Univ. D'Oran2	Présidente
Hachemi Khalid	Pr	IMSI-Univ. D'Oran2	Encadreur
Titah Mawloud	MAA	IMSI-Univ. D'Oran2	Examineur

Année 2019/2020

Remerciements

Nous tenons à remercier premièrement Dieu tous puissant pour la volonté, la santé et la patience, qu'il nous a donné durant toutes ces longues années.

Nous tenons également à exprimer notre vif remerciement à notre encadreur Monsieur Hachemi Khalid, pour avoir d'abord proposé ce thème et suivi le déroulement de ce mémoire et pour la confiance et l'intérêt qu'ils nous ont accordé tout au long de la réalisation de ce travail.

Nous tenons à citer dans ces remerciements les membres du jury qui ont bien voulu examiner et juger notre travail.

Nous remercions tous nos professeurs de l'IMSI qui ont contribué à notre formation.

Merci enfin à tous ceux qui, de près ou de loin, nous ont aidé et donc ont contribué au succès de ce travail.

Dédicace

Je dédie ce travail:

À ALLAH le tout-puissant à qui je dois tout.

À mes chers parents, pour tout leurs sacrifices, leur amour, leur tendresse, leur soutien et leurs prières tout au long de mes études.

À mes chères sœurs Hadia et Fatima pour leur encouragements permanents et leur soutien moral.

À toute ma famille et mes amis pour leur soutien tout au long de mon parcours universitaire.

À tous mes professeurs: Leur générosité et leur soutien m'oblige de leurs témoigner mon profond respect et ma loyale considération.

À mon binôme: Abderrahmane

Abdelkader

Dédicace

Je dédie ce travail:

À ALLAH le tout puissant à qui je dois tout.

À mes chers parents, pour tout leurs sacrifices, leur amour, leur tendresse, leur soutien et leurs prières tout au long de mes études.

À mes chère frères Omar, ayoub et Yasser, et mes chères sœurs Hafsa et Marwa pour leur encouragements permanents et leur soutien moral.

À toute ma famille et mes amis pour leur soutien tout au long de mon parcours universitaire.

*À tous mes professeurs: Leur générosité et leur soutien m'oblige de leurs
témoigner mon profond respect et ma loyale considération.*

À mon binôme: Abdelkader

Abderahmane

Introduction générale	01
Chapitre I	Les piluliers automatiques
I.1.Introduction	02
I.2 Contexte du projet	02
I.3 Inportance de pilulier	02
I.4 Le pilulier connecté : une nouvelle tendance	02
I.5 Les avantages de ce dispositif	03
I.6 Les types de piluliers	03
I.7 Conseils pour le choix des piluliers	03
I.7.1 Recommandations	04
I.8 Cas particulier du patient autonome qui gère ses prises	04
I.9 Hygiène	04
I.10 Transport des piluliers	04
I.11 Stockage de piluliers	04
I.12 Des exemples simples sur les piluliers automatique	05
I.12.1 Version française	05
I.12.2 BOX Portable	06
I.12.3 Pilbox classic	07
I.12.4 SUKUOS Pilulier Semainier	08
I.12.5 SAKURAM Pilulier	09
I.13 Conclusion	10
Chapitre II	Systèmes de commande Arduino
II.1.Introduction	11
II.2 Présentation d'Arduino	11
II.2.1Définition	11
II.2.2 Bonnes raisons de choisir Arduino	11
II.2.3 Outils Arduino	12
II.2.4 Les gammes de la carte Arduino	12
II.2.5 Les types de carte Arduino	13
II.2.6 La carte UNO	13
II.3 Présentation d'Arduino UNO	14
II.3.1 Comment utiliser l'Arduino UNO	14
II.3.2 Caractéristiques de l'Arduino UNO	14
II.3.3 Synthèse des caractéristiques	16
II.3.4 Alimentation	16
II.3.5 Mémoire	18
II.3.6 Principaux fonctionnements	19
II.4 Étude de la partie logicielle	19
II.4.1 Plateforme de programmation Arduino	19
II.4.2 Proteus Professional	22
II.5 Le moteur Pas à Pas	22
II.5.1 Différents types des moteurs	22
II.5.2 Caractéristiques générales des moteurs pas à pas	23

Table des matières

II.5.3 Moteur pas-à-pas - 5V, 32 pas, réduction 1/16	25
II.5.4 Synthèse des caractéristiques	25
II.6 Conclusion	26
Chapitre III	Conception d'un pilulier automatique : hardware et software
III.1 Introduction	27
III.2 Les étapes de la réalisation	27
III.3 Description du système	27
III.4 Dispositifs matériels et technologie	27
III.5 Pilule.....	28
III.6 Choix des composants et justification	29
III.7 Simulation et test avec Arduino	35
III.8 Programmation avec Arduino	39
III.9 Description de fonctionnalité du pilulier réalisé	42
III.9.1 Comment le verrouiller ?	42
III.9.2 Comment les pilules sont-elles distribuées?.....	42
III.9.3 quelles sont les alertes ?	42
III.9.4 Affichage	42
III.9.5 Fonctions prises en charge	42
III.9.6 Les améliorations	43
III.9.7 Test	43
III.10 Conclusion	44
Conclusion générale.....	45

Chapitre I

Figure I.1 : *pilulier automatique version française*

Figure I.2 : *pilulier automatique Box portable*

Figure I.3 : *pilulier automatique Pilbox classic .*

Figure I.4 : *pilulier automatique SUKUOS Pilulier Semainier*

Figure I.5 : *pilulier automatique SAKURAM Pilulier*

Chapitre II

Figure II.1 : *Arduino UNO*

Figure II.2 : *le brochage de la carte UNO.*

Figure II.3 *schéma explicatif d'Arduino*

Figure II.4 : *Logiciel de programmation de la carte Arduino*

Figure II.5 : *Arduino 1.8.1*

Figure II.6 : *Barre d'actions*

Figure II.7 : *Moteur pas-à-pas*

Chapitre III

Figure III.1 : *La maquette en 3D*

Figure III.2 : *la partie statique.*

Figure III.3 : *la partie mobile*

Figure III.4 : *La maquette complète*

Figure III.5 : *La carte Arduino UNO*

Figure III.6 : *les caractéristiques de carte Arduino UNO*

Figure III.7 *Moteur pas à pas*

Figure III.8: *Driver moteur STP01(UNL2003)*

Figure III.9 : *Adapatateur 9v-1v*

Figure III.10 : *Afficheur LCD*

Figure III.11 : *Alimentation pour Arduino MB102*

Figure III.12 : *Les câbles*

Figure III.13 : *Le buzzer*

Figure III.14 : *branchement Arduino avec le moteur pas à pas et UNL2003*

Figure III.15 : *branchement réel Arduino avec le moteur pas à pas et UNL2003*

Figure III.16 : *branchement Arduino avec LCD*

Figure III.17 : *branchement réel Arduino avec LCD*

Figure III.18 : *branchement Arduino avec buzzer*

Figure III.19 : *branchement réel Arduino avec buzzer*

Chapitre II

Tableau II-1 : chronologie des sorties de carte Arduino

Tableau II-2 : les caractéristiques de la carte Arduino UNO

Chapitre III

Tableau III -1 : *Tableau de branchement de moteur pas à pas et UNL2003 avec Arduino*

Tableau III -2 *Tableau de branchement d'afficheur LCD avec Arduino*

Abréviation

PIC : Programmable Interface Controllers

SRAM : Static Random Access Memory

EEPROM : electrically erasable programmable read only memory

PWM : (Pulse Width Modulation) ,Modulation de largeur d'impulsion.

AC/DC : alternatingcurrent/direct current (« courant alternatif/courant continu »).

GND : Ground = Mise à la masse, à la terre

IDE : Integrated DevelopmentEnvironment.

LCD : Liquid Crystal Display, soit « affichage à cristaux liquides »

ICSP: In Circuit Serial Programming

TTL : Transistor-Transistor Logic

PC : partie commande

PO : partie opérative

PR : partie relation

DJIN : la Dispensation Journalière Individuelle et Nominative

ISO : Organisation internationale de normalisation

GRAFCET : Graphe de Commande Étape et Transaction

GEMMA : Guide d'Étude des Modes de Marche set Arrêts

SAP : Système Automatisé de Production

CBT : Cognitive behavioural therapy

IGAS : Inspection générale interministérielle du secteur social

Introduction générale

L'observance thérapeutique constitue un enjeu majeur dans les soins délivrés aux patients. Malheureusement pour certains patients, comme les sujets âgés avec plusieurs pathologies, l'observance devient une tâche très difficile à réaliser sans intervention d'une tierce personne. Dans ce contexte, le recours aux distributeurs automatiques, en particulier, les piluliers électroniques et automatiques constitue une solution salvatrice.

En effet, le pilulier automatique pourrait faciliter la tâche de la personne qui s'occupe d'un parent plus âgé. Il pourrait également être une assistance pour les personnes qui prennent plusieurs pilules et ont du mal à suivre l'ordre des prises.

La fonction d'un tel appareil consiste à délivrer automatiquement les médicaments, à des horaires préprogrammés de prises, dans un compartiment qui s'ouvre facilement et émet un signal sonore indiquant qu'il est temps de prendre le médicament tels systèmes consiste.

L'objectif de ce projet est de concevoir et de réaliser un prototype de pilulier automatique commandé par une carte Arduino Uno, permettant de distribuer cinq types de médicaments à différents moments de la journée.

Ce mémoire est composé de trois chapitres :

Le premier chapitre introduit les types de piluliers, leurs avantages, leurs importances et comment les choisir.

Le deuxième chapitre porte sur le système de commande Arduino en se focalisant sur l'Arduino UNO utilisé pour commander notre système.

Le troisième chapitre est consacré aux différentes parties de notre système (mécanique, électronique, électrique), les étapes de notre réalisation ainsi que les fonctionnalités du prototype.

Enfin, on termine par une conclusion générale avec des recommandations.

Chapitre 1

Les piluliers automatiques

I.1 Introduction

Les piluliers automatiques de pilules sont des boîtes de différentes formes, tailles et couleurs, et ils diffèrent même par le matériau de fabrication, y compris le plastique, le bois ou similaire, mais son objectif en est un, et il est de protéger le consommateur (la personne souffrant de maladies chroniques) de tomber dans l'oubli du moment de prendre. Médicament ou mélange accidentel entre les médicaments car il est programmé pour alerter le patient du moment et de la quantité de médicament à prendre à l'heure spécifiée, garantissant ainsi la sécurité du consommateur.

I.2 Contexte du projet

D'après des statistiques établis en 2014, un meilleur suivi des traitements médicaux et des médicaments dans le monde permettrait de réaliser 8% d'économies sur le coût total des dépenses de santé.

De plus, on estime en Europe que la mauvaise prise de médicaments est à l'origine de plusieurs dizaines de milliers de décès chaque année.

Selon le rapport IGAS2, en France, un mauvais suivi des heures de prises et de dosage de médicament aurait causé entre 8 000 à 12 000 décès par an et coûté près de 2 milliards € à l'Assurance Maladie.

13 millions de patients résidents en France seraient concernées par la poly médication, et parmi eux de nombreuses personnes âgées de plus de 60 ans.

La santé connectée est donc devenue un marché porteur.

C'est ainsi que de nombreuses solutions se développent chaque jour qui a pour but de venir en aide aux personnes âgées dans les horaires de leurs prises de médicaments et éviter les surdosages ou les erreurs de traitement.

I.3 Importance du pilulier

Le pilulier ou boîte à pilules est devenu un accessoire d'une grande importance surtout pour les personnes sous traitement médicaux, pour les seniors également. La fonction principale de cette boîte si spéciale, c'est d'organiser au mieux la prise de médicaments. Quand vous n'avez qu'une seule pilule à prendre deux ou trois fois par jour, l'usage d'un pilulier n'est pas très utile. Par contre, pour les personnes qui suivent des traitements médicaux, pour les personnes âgées qui ont tendance à oublier la prise de leurs médicaments, cet accessoire peut être salvateur. Un pilulier améliore considérablement l'observance thérapeutique. Et il devient, d'une grande importance que ce soit pour les patients mais aussi pour les médecins [1].

I.4 Le pilulier connecté : une nouvelle tendance

Pour les seniors connectés et amateurs de nouvelles technologies, ou les patients plus jeunes, les grandes marques proposent actuellement le pilulier connecté. Ce dernier se présente comme un genre de calculatrice qui garde notamment vos médicaments à l'abri. Ça peut être des interactions extérieures, des pertes en tout genre, de l'humidité, etc.

Le pilulier connecté est un allié pendant les déplacements. Et pour cause, il facilite l'accès aux traitements. Il est de petite taille, et ne prend donc pas de place, Il suffit d'appuyer sur le bouton correspondant pour accéder aux pilules dont on a besoin [2].

En outre, ce genre d'accessoire permet de suivre avec minutie les traitements. Certains appareils peuvent émettre des alertes et/ou des notifications quand il est l'heure de prendre les pilules. Ainsi, il réduit les risques à cause d'un oubli malencontreux ou d'un retard de prise. Ce qui peut être d'une importance capitale surtout pour les diabétiques ou ceux avec un problème cardiaque.

I.5 Les avantages de ce dispositif:

Le pilulier, qu'il soit électronique, simple, journalier ou encore semainier présente de nombreux avantages. Il propose une assistance aux personnes seules, sans famille et ayant régulièrement des trous de mémoire. Cela leur évite aussi d'avoir à payer un garde-malade qui peut s'avérer bien plus cher.

D'un autre point de vue, le pilulier peut être d'une aide capitale pour les parents trop occupés pour donner eux-mêmes les médicaments à leurs enfants. Il en est de même pour les enfants qui doivent prendre soin de leurs parents du troisième âge [3].

I.6 Les types de piluliers :

On trouve des piluliers semainiers et journaliers. Le semainier comporte des compartiments plus grands qui permettent de ranger des médicaments pour une semaine entière. La consistance dépend de la dose à prendre par le patient. En ce qui concerne le journalier, il doit être rempli chaque jour. Il aide à ne pas se tromper surtout pour les personnes âgées qui vivent en solitaire.

On trouve également sur le marché des piluliers simples et électroniques. Les modèles simples ne sont pas connectés à internet. En cas d'inobservance des doses prescrites, une tierce personne ne pourrait pas en avoir connaissance. En ce qui concerne le modèle électronique, le contrôle est plus poussé. Le médecin peut suivre la prise de chaque dose de façon journalière. Le pilulier électronique aide beaucoup plus à une prise régulière, car à l'heure de chaque prise, le compartiment concerné clignote. Certains piluliers électroniques sont programmés pour envoyer des SMS ou des mails afin de notifier qu'il est l'heure de la prise des médicaments [4].

I.7 Conseils pour le choix des piluliers

Il existe plusieurs types/modèles de piluliers sur le marché [5].

Les critères de choix du modèle doivent tenir compte :

- de la typologie des patients
- du type d'hospitalisation, de la durée moyenne de séjour
- des modalités de transport (robustesse, facilité d'entretien)
- de l'ergonomie : pilulier identifiable et modulable, présenté en 8 piluliers journaliers.

I.7.1 Recommandations

- Choisir un pilulier facilement nettoyable
- Choisir un pilulier avec un système de fermeture efficace et un seul sens d'ouverture possible, ce qui limite le risque de mélange des traitements lors du transport ou en cas de chute du pilulier

- Chaque pilulier doit être identifié avec le jour de la semaine écrit en toutes lettres (en plus de l'identification du patient). L'inscription peut être complétée par l'utilisation d'un code couleur différent pour chaque jour.
- Privilégier les piluliers journaliers présentant au moins 4 cases par jour. Chaque case ou séparation est identifiée : matin 8h, midi 12h, soir 18h, couché 22h.
- Privilégier les piluliers avec des cases modulaires ou amovibles, ce qui permet d'adapter leur capacité au traitement à stocker et aux horaires particuliers.

I.8 Cas particulier du patient autonome qui gère ses prises. [6]

- S'assurer de ses capacités à manipuler le système de fermeture/ouverture
- Lui remettre avec son pilulier son plan de prise / plan d'administration
- Lui expliquer l'importance du respect de la « règle des 5B »
- Donner la consigne de signaler toute chute du pilulier
- Lui demander de noter ses heures de prise sur le plan

I.9 Hygiène

L'hygiène des piluliers doit être définie avec une fréquence régulière, au sein d'une procédure. Cette procédure doit préciser [7] :

- L'hygiène des mains et générale.
- L'hygiène de la zone de travail.
- L'hygiène des piluliers.

I.10 Transport des piluliers

Le transport des piluliers doit être organisé et sécurisé [7].

I.11 Stockage de piluliers

Les piluliers doivent être stockés dans un endroit sécurisé fermé à clef [7].

I.12 Des exemples simples sur les piluliers automatique

I.12.1 Version française



Figure I.1 : pilulier automatique version française

Description du produit

➤ Le pilulier électronique automatique avec alarme Dose Control permet la distribution programmée de médicaments jusqu'à 6 fois par jour, en distribuant uniquement la dose prescrite pour chaque jour et heure (heure, minute), Le patient n'a aucun accès à d'autres doses

➤ Pour chaque jour, de 1 à 6 alarmes sonores (sélection de 3 tonalités) et visuelles (voyant rouge clignotant sur le couvercle) peuvent être réglées à l'heure et à la minute désirées pour vous rappeler le besoin de médicaments

➤ Le distributeur automatique de comprimés est doté d'une alarme sonore réglable en trois tonalités pouvant durer jusqu'à 30 minutes ; après cette période, l'alarme visuelle continuera à clignoter L'affichage a un changement de temps automatique.

➤ Dans le délai souhaité, l'alarme est déclenchée et le conteneur de comprimés en rotation déplace le médicament en position de l'ouverture de distribution ; dans ce moment-là le patient retire simplement les comprimés en basculant le distributeur ; Le renversement désactive immédiatement les alarmes sonores et lumineuses déclenchées.

➤ Le pilulier automatique Dose Control peut être verrouillé avec une clé métallique pour protéger le patient.

➤ Le distributeur de médicaments comporte 28 grands compartiments pour la conservation des médicaments (chaque plateau peut contenir jusqu'à 18 comprimés d'aspirine).

➤ L'ajout du médicament pour le dosage se réalise à l'aide de disques de dosage en français.

➤ Le pilulier automatique dispose d'un couvercle blanc opaque (le patient ne voit pas les comprimés) ou transparent (le patient peut voir les comprimés).

➤ L'alimentation électrique du distributeur automatique de comprimés Dose Control est basée sur 4 batteries AA, l'écran affiche le déchargement des batteries.

I.12.2 BOX Portable



Figure I.2 : *pillulier automatique Box portable*

Description du produit

➤ pillulier automatique intelligent pour les personnes âgées avec cinq alarmes de synchronisation de la voix.

➤ La fonction intelligente de rappel empêche l'oubli, les six petites grilles répondent à la dose hebdomadaire, et cinq ensembles d'horloges d'alarme peut être réglée, ce qui peut effectivement empêcher les gens oublieux de prendre trop ou oublier de prendre des médicaments. Chaque capsule peut contenir 4 ou 10 capsules. Les bouteilles et les canettes sont légères, faciles à transporter, en toute sécurité. Il est plus sûr pour les personnes, les hôpitaux, les personnes âgées, les patients et Voyage.

I.12.3 Pilbox classic



Figure I.3 : *pilulier automatique Pilbox classic*

Description du produit

➤ Grâce à son système breveté de bouton poussoir, votre traitement est délivré facilement d'une simple pression. Pratique et fonctionnel, il se remplit une seule fois pour chaque semaine de traitement.

I.12.4 SUKUOS Pilulier Semainier



Figure I.4 : *pilulier automatique* SUKUOS Pilulier Semainier

Description du produit

- est un pilulier hebdomadaire qui peut stocker pendant 7 jours.
- Boîtier de haute qualité.
- très facile à transporter, une excellente aide dans la vie quotidienne et les voyages.

Spécification :

- Matériau : PP de qualité alimentaire
- taille du produit : 17.5 *10.6 *2.6 cm
- Chaque pilules Taille : 10 * 2 *2 cm
- Dimensions du compartiment : 2.5*2*2 cm

I.12.5 SAKURAM Pilulier



Figure I.4 : *pilulier automatique SAKURAM Pilulier*

Description du produit

C'est une minuterie de boîte à pilules à deux compartiments qui est pratique pour stocker deux types de pilules ou deux boîtes. Il fournit un excellent sceau pour protéger les comprimés et permettra de prendre des médicaments en temps opportun, même si on oublie.

Traits :

- Couleur : bleu et blanc.
- Matériel : PP et composant électronique.
- Taille : env. 9,5 x 5,5 x 2 cm / 3,74 x 2,2 x 0,8 pouces
- Un compte à rebours, une alarme sonore pour assurer de toujours savoir quand les tablettes doivent arriver.
- Fabriqué à partir d'un matériau de qualité supérieure, sûr, sain et durable, il offre une grande protection pour les pilules.
- Avec 2 compartiments pour 2 types de pilules différents ou 2 boîtes.
- Transport facile.
- Convient à tous les âges et à tous les utilisateurs, facile à installer. Parfait pour la maison, le bureau, l'extérieur, etc.

I.13 Conclusion

Les piluliers automatiques existent sous plusieurs types et formes selon les besoins du patient car chaque pilulier a ses avantages, son importance, son mode de portage ou de transport et son mode de propreté et chaque pilulier a un certain nombre de chambres donc dans cette partie il a quelques conseils pour choisir le bon pilulier.

Chapitre 2

Systeme de commande Arduino

II.1 Introduction

Depuis plusieurs années, Arduino a popularisé l'utilisation des circuits programmables dans diverses applications. Nous allons, dans un premier temps, voir ce qu'est le système Arduino et la programmation. Puis nous enchaînerons sur la prise en main de certains composants électroniques compatibles avec l'Arduino.

II.2 Présentation d'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. Sans tout ne connaître ni tout comprendre de l'électronique, 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 [8].

Pont tendu entre le monde réel et le monde numérique, Arduino permet d'étendre les capacités de relations humain/machine ou environnement/machine. Arduino est un projet en source ouverte : la communauté importante d'utilisateurs et de concepteurs permet à chacun de trouver les réponses à ses questions.

II.2.2 Bonnes raisons de choisir Arduino :

Il existe pourtant dans le commerce, une multitude de plateformes qui permettent de faire la même chose. Notamment les microcontrôleurs « PIC » du fabricant Micro chip. Nous allons voir pourquoi choisir l'Arduino :

a. Le prix :

En vue des performances qu'elles offrent, les cartes Arduino sont relativement peu coûteuses, ce qui est un critère majeur pour le débutant.

b. Les ressources libres :

Elle constitue en elle-même deux choses : Le logiciel : gratuit et open source, développé en Java, dont la simplicité d'utilisation relève du savoir cliquer sur la souris. Le matériel : cartes électroniques dont les schémas sont en libre circulation sur internet. Cette liberté a une condition : le nom « Arduino » ne doit être employé que pour les cartes « officielles ». En somme, vous ne pouvez pas fabriquer votre propre carte sur le modèle Arduino et lui assigner le nom « Arduino ». Les cartes non officielles, on peut les trouver et les acheter sur Internet et sont pour la quasi-totalité compatibles avec les cartes officielles Arduino.

c. La compatibilité :

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

d. La communauté :

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

II.2.3 Outils Arduino :

Il est composé de deux choses principales, qui sont : le matériel et le logiciel. Ces deux outils réunis, il nous sera possible de faire n'importe quelle réalisation.

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

Le logiciel : Le logiciel va nous permettre de programmer la carte Arduino. Il nous offre une multitude de fonctionnalités.

II.2.4 Les gammes de la carte Arduino :

Actuellement, il existe plus de 20 versions de module Arduino, nous citons quelques un afin d'éclaircir l'évaluation de ce produit scientifique et académique.[9]

- Le NG d'Arduino, avec une interface 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 programme à l'aide d'une porte USB cette version utilisant un microcontrôleur ATmega168 (ATmega328 pour une plus nouvelle version).
- Le Lily Pad Arduino, une conception de minimaliste pour l'application wear able en utilisant un microcontrôleur ATmega168.
- Le NG d'Arduino plus, avec une interface 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 Décimal, avec une interface USB et utilise un microcontrôleur ATmega168.
- L'Arduino Duemilanove ("2009"), en utilisant un microcontrôleur l'ATmega168 (ATmega328 pour une plus nouvelle version) et actionné par l'intermédiaire de la puissance d'USB/DC.
- L'Arduino Méga, en utilisant un microcontrôleur ATmega1280 pour I/O additionnel et mémoire.
- L'Arduino UNO, utilise un microcontrôleur ATmega328
- L'Arduino Mega2560, utilise un microcontrôleur ATmega2560, et possède toute la mémoire à 256 KBS. Elle incorpore également le nouvel ATmega8U2 (ATmega16U2 dans le jeu de puces d'USB de révision 3). La table II.1 donne les différents types de cartes Arduino.

II.2.5 Les types de carte Arduino :

Chronologie des sorties de cartes Arduino

Version de carte	Année de sortie	Micro-contrôleur
Diecimila	2007	ATmega168V
LilyPad	2007	ATmega168V/ATmega328V
Nano	2008	ATmega328/ATmega168
Mini	2008	ATmega168
Mini Pro	2008	ATmega328
Duemilanove	2008	ATmega168/ATmega328
Mega	2009	ATmega1280
Fio	2010	ATmega328P
Mega 2560	2010	ATmega2560
Uno	2010	ATmega328P
Ethernet	2011	ATmega328
Mega ADK	2011	ATmega2560
Leonardo	2012	ATmega32U4
Esplora	2012	ATmega32U4
Micro	2012	ATmega32U4
Yún	2013	ATmega32U4 +Linino

Tableau II-1 : chronologie des sorties de carte Arduino

II.2.6 La carte UNO :

L'Arduino Uno est un microcontrôleur programmable qui permet, comme son nom l'indique, de contrôler des éléments mécaniques : systèmes, lumières, moteurs, etc. Cette carte électronique permet donc à son utilisateur de programmer facilement des choses et de créer des mécanismes automatisés, sans avoir de connaissances particulières en programmation. Il est un outil pensé et destiné aux inventeurs, artistes ou amateurs qui souhaitent créer leur propre système automatique en le codant de toute pièce[12].



FigureII.1 : Arduino UNO

II.3 Présentation d'Arduino UNO :

Il existe plusieurs modèles de cartes Arduino mais le modèle Uno est le plus répandu et permet déjà un large éventail de possibilités.

L'Arduino Uno dispose de caractéristiques techniques suffisantes pour bien commencer dans la programmation : 14 entrées/sorties numériques, 6 entrées analogiques, une mémoire flash de 32 KB et de SRAM 2 KB, un EEPROM de 1 KB

La carte se connecte très facilement à un ordinateur via un câble USB fourni.

II.3.1 Comment utiliser l'Arduino Uno :

Pour commencer à utiliser l'Arduino Uno, on doit dans un premier temps devoir télécharger la dernière version du logiciel Arduino. Il est compatible Windows, Mac ou Linux.

Une fois cela fait, on lance le logiciel et on pourra constater la simplicité de l'interface du logiciel qui présente très sobrement une page vierge destinée au code, ainsi qu'une barre de boutons et un menu.

Cette console servira de relais exclusif à la communication avec la carte Arduino, afin qu'on puisse compiler des programmes et les transférer sur la carte.

On doit ensuite connecter la carte Arduino Uno sur l'ordinateur grâce à sa connexion USB et on commence pas à pas l'initiation en suivant les explications et autres vidéos de programmation amateurs disponibles sur Internet.

Pour finir, le code qu'on rentre sur l'éditeur de texte vierge du logiciel Arduino (en langage C ou C++), sera la manière de faire comprendre au microcontrôleur ce qu'on souhaite instaurer comme programme.

Les boutons présents dans la barre du logiciel serviront à compiler et envoyer les informations à la carte et également à tester la configuration du programme, afin de vérifier sa viabilité.

II.3.2 Caractéristiques de l'Arduino UNO

Cette carte dispose :

- ✓ 14 broches numériques d'entrées/sorties (dont 6 peuvent être utilisées en sorties PWM (largeur d'impulsion modulée)).
- ✓ 6 entrées analogiques (qui peuvent également être utilisées en broches entrées/sorties numériques).
- ✓ Un quartz 16Mhz.
- ✓ Une connexion USB.
- ✓ Un bouton de réinitialisation (reset)

Elle contient tout ce qui est nécessaire pour le fonctionnement du microcontrôleur ; pour pouvoir l'utiliser et se lancer, il suffit simplement de la connecter à un ordinateur à l'aide d'un câble USB (ou de l'alimenter avec un adaptateur secteur ou une pile, mais ceci n'est pas indispensable, l'alimentation étant fournie par le port USB). La figure suivante présente le brochage de la carte UNO. [13]

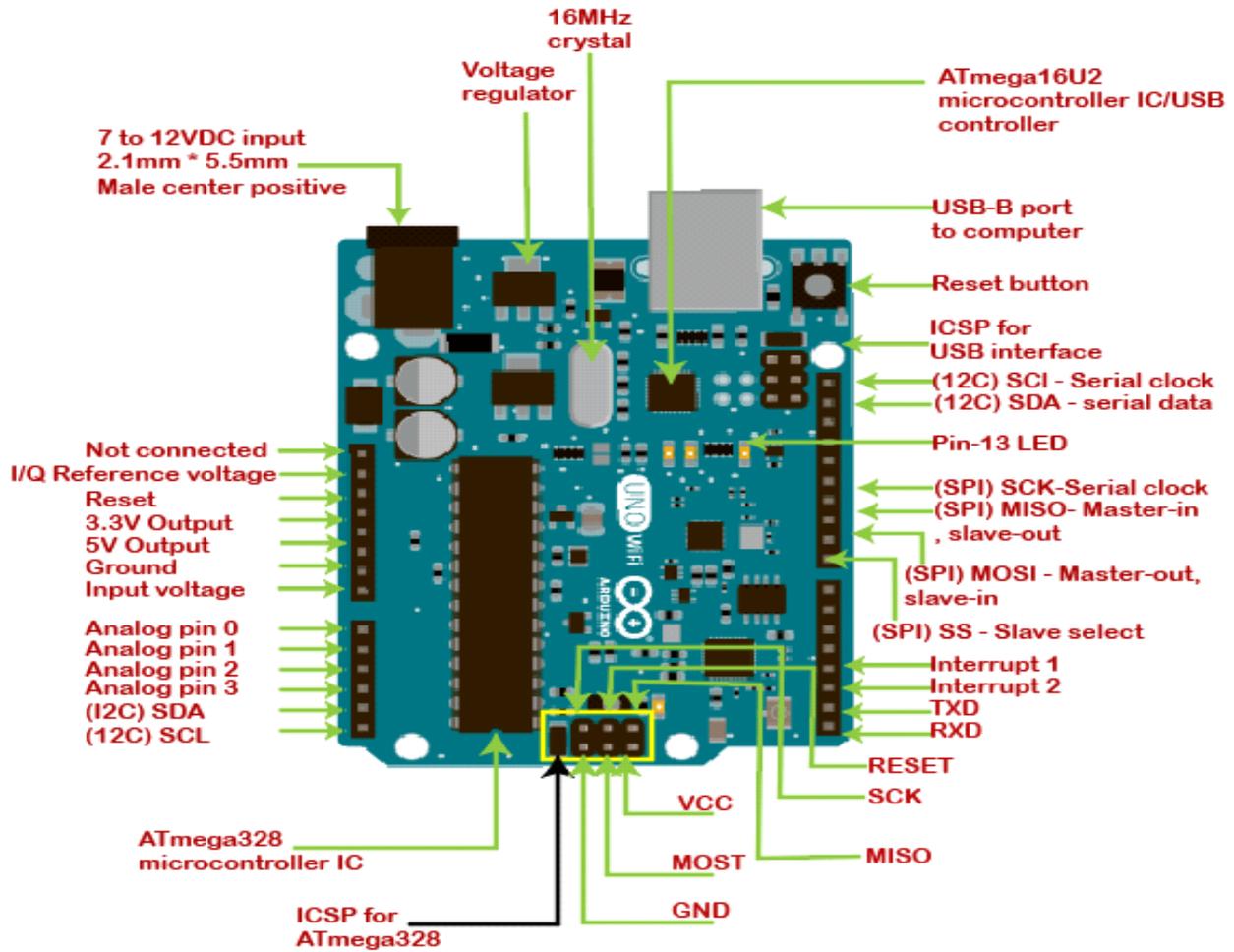


Figure II.2 : le brochage de la carte UNO. [13]

II.3.3 Synthèse des caractéristiques :

Le Tableau suivant présente les caractéristiques de la carte Arduino UNO[14]:

Mikrocontroller:	ATmega328
Operating Voltage:	5 V
Input Voltage (recommended):	7...12 V
Input Voltage (limit):	6...20 V
Digitale I/O-Pins:	14 (6 used for PWM-Output)
PWM-Pins:	6
Analog Input Pins:	6
DC Current per I/O Pin:	40 mA (max.)
DC Current for 3.3 V Pin:	50 mA (max.)
Flash Memory:	32 KB (0,5 KB used for Bootloader)
SRAM:	2 KB (ATmega328)
EEPROM:	1 KB (ATmega328)
Clock Speed:	16 MHz

Tableau II-2 : les caractéristiques de la carte Arduino UNO [14]

II.3.4 Alimentation :

- a. Dès lors que l'on teste un programme sous Arduino, la carte est reliée via le port USB à l'ordinateur. Le câble USB sert à deux choses : alimenter la carte mais aussi visualiser le programme que l'on teste.
- b. Quand le programme est testé et approuvé, la carte Arduino peut alors être utilisée de manière autonome. Elle n'a plus besoin de l'ordinateur pour fonctionner, mais il faut tout de même de l'énergie pour faire tourner la carte et les éventuels composants additionnels qui y sont connectés.

- c. Nous allons voir plusieurs façons d'apporter l'alimentation nécessaire à l'Arduino pour fonctionner[17].

d. La tension

Il faut tout d'abord connaître les spécifications de la carte Arduino que l'on possède. Elles utilisent tout d'abord toutes des courants continus.

Si l'on regarde les différentes broches possibles, on va généralement trouver une broche étiquetée 5V, une broche étiquetée 3,3V et une broche étiquetée VIN ou RAW.

- VIN ou RAW correspondent à l'entrée du régulateur de tension présent sur la carte. Pour alimenter l'Arduino via cette broche, il faut une alimentation dont la tension est comprise entre 7 et 12V ;
- 5V n'a pas le même rôle selon que le microcontrôleur qui équipe l'Arduino est alimenté en 5V ou en 3,3V. Dans le premier cas, c'est directement la tension d'alimentation du microcontrôleur. Dans le second cas, le microcontrôleur est alimenté via un régulateur 3,3V dont l'entrée est connectée sur le 5V ;
- De la même manière, le 3,3V va être la tension d'alimentation du microcontrôleur pour ceux alimentés en 3,3V et une tension de service pour connecter des dispositifs externes nécessitant du 3,3V si le microcontrôleur nécessite du 5V.

e. Alimentation par USB mais sans ordinateur

Cette méthode est la plus facile pour peu que l'on possède une alimentation régulée pour Smartphone et que la carte Arduino possède un port micro-USB.

Comme le port USB de l'ordinateur, le chargeur du Smartphone délivre du 5V continu. On vérifiera d'une part que l'alimentation régulée du Smartphone est suffisante pour la consommation de la carte, c'est généralement le cas, ces alimentations délivrent au moins 1A. D'autre part, les cartes Arduino officielles sont équipées d'un fusible électronique à réarmement automatique qui coupe l'alimentation si la limite de 500mA est dépassée. Son rôle est de protéger le port USB de votre ordinateur.

Cette méthode limite donc la consommation totale de votre montage à 500mA.

f. Alimentation par connecteur Jack

Il existe un connecteur Jack femelle. Il faut donc un connecteur jack mâle de 2.1mm de diamètre interne. Ce type de connecteur possède 2 fils dont le fil + est à l'intérieur et le fil - à l'extérieur.

Cela se schématise de cette façon :



Sur cette entrée, une diode de protection, évite les mauvais branchements.

Comme source, il nous faut donc une tension entre 7 et 12V continu (tension optimale). Nous pouvons utiliser plusieurs sources :

- Un bloc d'alimentation AC/DC, 9V est une tension fréquente et idéale ;
- Une pile 9V

➤ Un ensemble de 7 piles 1.5V...

g. Alimentation par 2 broches de l'Arduino : VIN et GND :

Dans l'hypothèse où notre transformateur ne possède pas le connecteur jack adapté, il existe cette solution.

Après avoir repéré les fils du transformateur à savoir + et -, il faut les relier à l'Arduino par les deux broches nommées, à savoir :

- Le + (habituellement le fil rouge) est relié à la broche VIN
- Le - (habituellement le fil noir) est relié à GND, soit Ground (masse en français)

Tout comme la façon 2, la tension optimale de ce courant continu se situe entre 7 et 12V.

Dans ce contexte et afin de protéger l'Arduino d'un mauvais branchement le + sur le - et le - sur le +, nous pouvons intercaler une diode qui ne laissera passer le courant que s'il est dans le bon sens. Attention à bien mettre la diode qui est un composant polarisé dans le bon sens.

La référence de diode est de type 1N400x.

Voici un petit schéma explicatif(fig.II.3) :

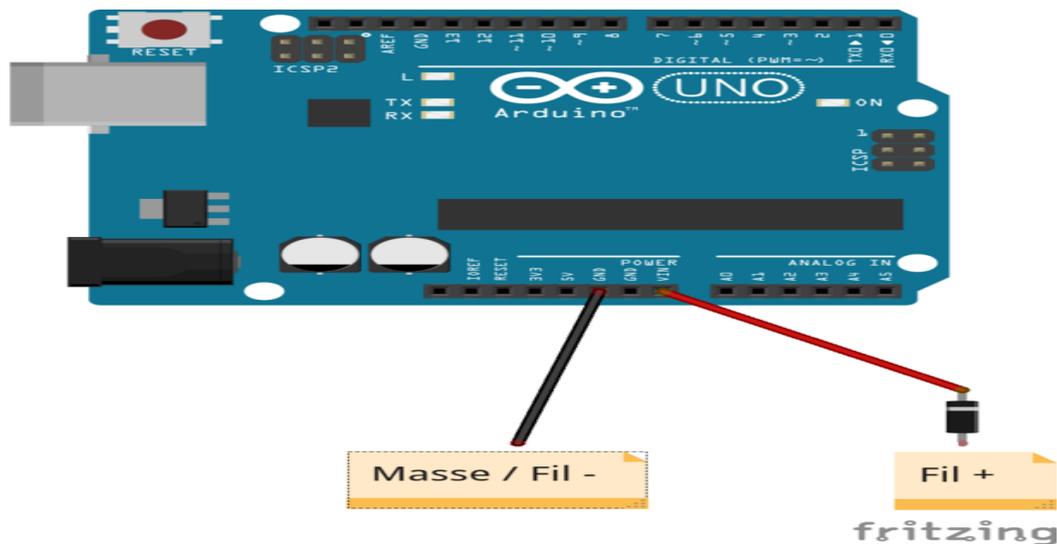


Figure II.3 : schéma explicatif d'Arduino

II.3.5 Mémoire :

La mémoire est l'un des principaux composants d'un ordinateur ou d'un microcontrôleur comme l'Atmega, cœur de l'Arduino.

Nous allons examiner les 3 types de mémoires disponibles au sein d'un Arduino uno, leurs capacités et leurs rôles afin d'en comprendre les nuances et usages.

Cet article a pour objet la présentation très synthétique de ce vaste sujet, nous évoquerons aussi quelques pistes pour résoudre les problèmes de manque de mémoire[16].

➤ La mémoire FLASH : Peu coûteuse cette mémoire sert à stocker les programmes à exécuter, c'est une mémoire qui perdure après arrêt de l'alimentation. Rapide, elle est donnée pour une capacité de 1 000 000 de cycles. L'atmega 328 en est doté de 32 kilobits (d'où la référence 32...8)

➤ La mémoire SRAM (Static Read Access Memory) : Coûteuse mais rapide, cette mémoire sert à stocker des données temporaires (les variables de votre programme par exemple). C'est une mémoire volatile. Arduino uno en possède seulement 2kilobits, ce qui peut s'avérer peu.

➤ La mémoire EEPROM : (Electrically-Erasable Programmable Read-Only Memory ou mémoire morte effaçable électriquement et programmable). Elle permet le stockage par le programme de données persistantes, peut être coûteuse, mais lente, l'Atmega 328 en possède 1 kilobit. Cette mémoire s'utilise plus rapidement, sa capacité est de 100 000 cycles d'écriture. Cette mémoire permettra aux programmes de stocker des données de log ou des états devant être conservées de manière pérenne (après arrêt de l'alimentation)

Arduino	Processor	Flash	SRAM	EEPROM
UNO, Uno Ethernet, Menta, Boarduino	Atmega328	32K	2K	1K

II.3.6 Principaux fonctionnements :

Les principales fonctionnalités de l'interface de l'application Arduino permet d'écrire et éditer un programme (appelé sketch) qui sera compilé puis téléversé sur la carte Arduino. Ainsi, lorsque l'on apporte des changements sur le code, ces changements ne seront effectifs qu'une fois le programme téléversé sur la carte. Il est à noter que ce manuel fait référence à la version en anglais de ce logiciel puisqu'elle comporte habituellement des mises à jour plus récentes que la version en français.

II.4 Étude de la partie logicielle

L'environnement de programmation open-source pour Arduino peut être téléchargé gratuitement (pour Mac OS X, Windows, et Linux) (fig.II.4).

II.4.1 Plateforme de programmation Arduino :

Le logiciel de programmation de la carte ARDUINO sert d'éditeur de code (Langage proche du C). Une fois, le programme tapé ou modifié au clavier, il sera transféré et mémorisé dans la carte à travers de la liaison USB. Le câble USB alimente à la fois en énergie la carte et transporte aussi l'information ce programme appelé IDE ARDUINO[10].



Figure II.4 : Logiciel de programmation de la carte Arduino

La fenêtre de l'application Arduino comporte les éléments suivants (fig.III.5)[11] :



Figure II.5 : Arduino 1.8.1

1. Un menu.
2. Une barre d'actions.
3. Un ou plusieurs onglets correspondant aux sketches.
4. Une fenêtre de programmation.
5. Une console qui affiche les informations et erreurs de compilation et de téléversement du programme.

✚ Eléments du menu :

Les différents éléments du menu permettent de créer de nouveaux sketches, de les sauvegarder, de gérer les préférences du logiciel et les paramètres de communication avec votre la carte Arduino.

✚ Dossier de travail :

Dans les préférences (File > Préférences), il est possible de spécifier le dossier de travail. Il s'agit du dossier où seront sauvegardés par défaut les programmes et les bibliothèques qui pourront y être associées. Lorsqu'un programme est sauvegardé, un dossier portant le nom du programme est créé. Celui-ci contient le fichier du programme portant le nom qu'on lui aura donné suivi de l'extension. On peut ainsi créer un dossier intitulé applet qui contient les différents éléments créés et nécessaires lors du processus de compilation du programme et de téléchargement vers la carte.

✚ Outils de configuration (Tools) :

Dans le menu Tools, il est possible et essentiel de spécifier le type de carte Arduino qu'on utilise. Sous Tools > Board, il faut spécifier pour quel type de carte on compile et téléverse le programme. Le type de carte est généralement inscrit sur la carte elle-même. Il est également nécessaire lorsqu'on branche une nouvelle carte Arduino ou que l'on change de carte de spécifier le port sériel virtuel qui sera utilisé pour la communication et le téléchargement du programme. Pour ce faire, il faut aller sous Tools > Serial Port et choisir le port approprié. Sous Windows, il s'agit la plupart du temps du port ayant un numéro supérieur à 3. Sous Mac OS X, il s'agit habituellement du premier élément de la liste. Une bonne technique pour déterminer quel port correspond à la carte Arduino consiste à débrancher celui-ci, attendre un peu et de prendre note des ports déjà présents.

✚ Barre d'actions (fig.II.6) :

	Bouton « Verify » (Vérifier) ; il permet de compiler votre programme et de vérifier si des erreurs s'y trouvent. Cette procédure prend un certain temps d'exécution et lorsque est terminée, elle affiche un message de type « Binary sketch size : ... » indiquant la taille du sketch téléversé.
	Pour transmettre le sketch compilé avec succès sur la carte Arduino dans le microcontrôleur.
	Bouton « New » (Nouveau) ; ce bouton permet de créer un nouveau sketch.
	Bouton « Open » (Ouvrir) ; il fait apparaître un menu qui permet d'ouvrir un sketch qui figure dans votre dossier de travail ou des exemples de sketches intégrés au logiciel.
	Bouton « Save » (Sauvegarder) ; il permet de sauvegarder votre sketch.

Figure II.6 : Barre d'actions

II.4.2 Proteus Professional :

C'est une suite logicielle destinée à l'électronique. Développé par la société LabcenterElectronics, les logiciels inclus dans Proteus Professional permettent la CAO (Construction Assistée par Ordinateur) dans le domaine électronique. Deux logiciels principaux composent cette suite logicielle : (ISIS, ARES, PROSPICE) et VSM. Cette suite logicielle est très connue dans le domaine de l'électronique. De nombreuses entreprises et organismes de formation (incluant lycée et université) utilisent cette suite logicielle. Outre la popularité de l'outil, Proteus Professional possède d'autres avantages :

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

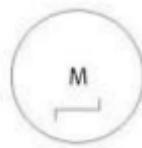
II.5 Le moteur Pas à Pas

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°).

Le nombre du pas par tour est déterminé par la relation suivante : $Np = 2\pi/\emptyset p$

On peut dire que 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[17].



II.5.1 Différents types des moteurs

Il existe deux types des moteurs pas à pas : il y a les moteurs à aimants permanents et les moteurs à réluctance variables. Les moteurs à aimants permanents se subdivisent en deux types :

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.

II.5.2 Caractéristiques générales des moteurs pas à pas

Les caractéristiques sont divisées en deux, statique et dynamique [17]:

a. Caractéristiques statiques

Ces caractéristiques concernent le moteur au repos. Elles sont essentiellement conditionnées par la construction mécanique du moteur.

- Angle de pas ϕ_p

C'est l'angle élémentaire effectué par le moteur lorsqu'un ou un ensemble d'enroulements sont alimentés. Cet angle est défini par les constructeurs et donné par :

$\phi_p = (360/m) \times N_r$; pour les moteurs à réluctance variable m : nombre de phase

- La précision du pas

Ecart relative de la position réelle de l'élément mobile par rapport à une position théorique d'équilibre mesuré à vide. Cette valeur est exprimée en pourcentage du pas :

Précision = Erreur max. d'angle x 100/pas

- Courant par phase

C'est l'intensité par phase à fréquence nulle (moteur à l'arrêt) qui provoque l'échauffement maximum admissible par le moteur en fonctionnement.

- Angle de puissance

C'est l'angle de retard de rotor par rapport au champ magnétique qu'est provoqué par une charge extérieure.

- La résolution

C'est le déplacement minimal demandé ou l'angle du pas en degré.

b. Caractéristiques dynamiques

Le démarrage, la rotation et le freinage du moteur dépend de la fréquence du commutateur des phases et du circuit d'alimentation pour pouvoir démarrer et l'entraîner sans perdre de pas. Donc, il est nécessaire de connaître ces caractéristiques.

- Le temps de montée (T_m)

C'est le temps nécessaire pour passer d'une position d'équilibre à une autre. Il est repéré par T_m

- Le temps d'établissement (T_e)

C'est le temps nécessaire à établir la position du rotor autour d'une position d'équilibre avec une bande d'erreur de position donnée

- Fréquence maximale réversible (réponse maximale)

Fréquence maximale permettant au moteur de changer le sens de rotation en restant en synchronisation avec les impulsions d'excitation des phases.

- Fréquence maximale de démarrage

C'est la fréquence maximale jusqu'à laquelle le moteur peut suivre la commande au moment de démarrage ; cette fréquence est légèrement plus haute que la fréquence maximale réversible.

- Couple maximale dynamique

C'est le couple maximal disponible sur l'arbre lorsque les phases sont alimentées successivement.

- Couple de démarrage et d'entraînement

C'est le couple maximal disponible sur l'axe de moteur lors de démarrage (entraînement) à une fréquence donnée sans perdre de pas.

II.5.3 Moteur pas-à-pas - 5V, 32 pas, réduction 1/16

Ce petit moteur pas à pas dispose de 32 pas, mais également d'un réducteur 1/16, cela permet ainsi d'obtenir 513 pas sur une révolution complète de l'arbre de sortie[18].

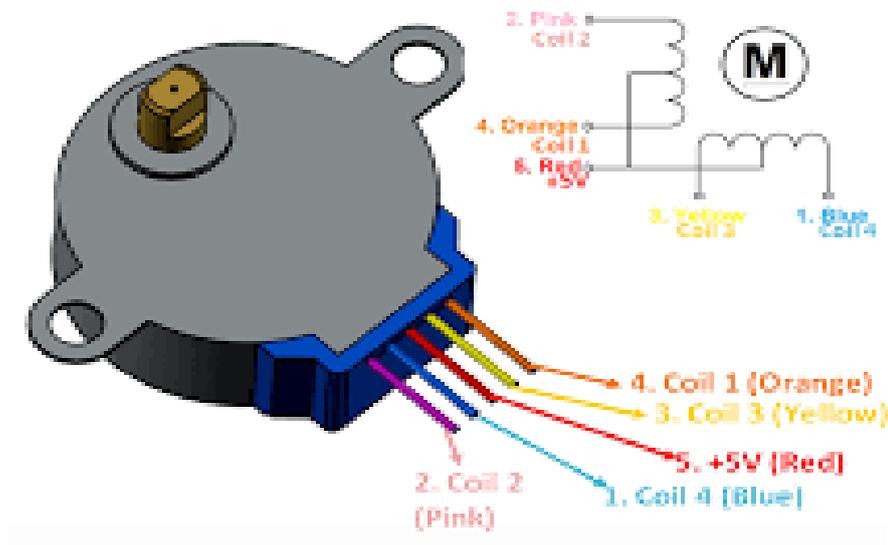


Figure II.7 : Moteur pas-à-pas

II.5.4 Synthèse des caractéristiques :

- Pas à pas unipolaire avec connecteur de câble 0.1 à broches 5 espacées
- Étapes 513 par tour
- Réduction réduite 1 / 16.032
- Opération suggérée par 5V DC
- Poids : 37 g.
- Dimensions : diamètre 28mm, hauteur 20mm, y compris arbre 9mm avec diamètre 5mm
- 9 " / 23 cm long câble
- Couple de maintien : 150 gram-force * cm, 15 N * mm / 2 oz-force * in
- Arbre : Diamètre 5mm aplati
- Impédance approximative 42 ohm cc pour enroulement

II.6 Conclusion

Dans ce chapitre, nous avons vu la carte Arduino, puis nous avons cité les différents types de cette dernière. Ensuite, nous avons expliqué les deux parties essentielles de l'Arduino ; plus précisément (la partie matérielle et la partie de programmation). Nous avons également expliqué le principe de fonctionnement de la carte Arduino sans oublier ses caractéristiques.

Enfin, quelques descriptions sur le moteur pas à pas en général et sur le moteur pas-à-pas - 5V, 32 pas, réduction 1/16 ont été présentées.

Chapitre 3

Conception d'un pilulier automatique : hardware et software

III.1 Introduction

Dans ce chapitre, on va procéder à l'application expérimentale, « la conception et la réalisation d'un pilulier automatique basé sur une carte Arduino ». Pour cette raison, on présentera les différents éléments constitutifs afin de réaliser une telle combinaison avec le module Arduino, les Outils de programmation (Arduino) ainsi que les différents tests et le montage final.

III.2 Les étapes de la réalisation :

La réalisation de notre dispositif a connu trois étapes :

La première étape : cette étape comporte la conception de ce système, les composants utilisés dans cette réalisation et le montage de ce projet.

La deuxième étape : Simulation du système avec programmation d'Arduino.

La troisième étape : test du montage complet, puis la réalisation du pilulier automatique.

III.3 Description du système :

Notre système assure la distribution au patient de 5 types de médicaments selon l'ordonnance de son traitement. La partie opérative se compose d'un plateau fixe et d'un plateau mobile par rotation. La partie mobile de forme circulaire est subdivisée en 6 chambres, contenant chacune un type de médicaments. La délivrance d'un médicament se fait par rotation de la partie mobile jusqu'à l'ouverture prévue dans le couvercle du boîtier. La partie commande est assurée par une carte Arduino Uno. La commande consiste à envoyer les données via le port série à l'Arduino sous forme d'une trame contenant les informations nécessaires, comme le numéro de chambre, le nom de médicaments à distribuer et le temps de distribution. A partir de ces données l'Arduino commande le moteur pas à pas pour tourner avec des rotations réglées (1^{er} chambre → 60°/ 2^{ème} chambre → 120°/ 3^{ème} chambre → 180°/ 4^{ème} chambre → 240°/ 5^{ème} chambre → 300°). Quand le malade prend le médicament, il appuie sur le bouton poussoir pour retourner le plateau tournant à sa position initiale.

III.4 Dispositifs matériels et technologie :

Pour réaliser notre système, nous avons utilisé les équipements suivants :

- Pilulier de 5 chambres.
- Circuit électronique (Arduino Uno).
- Alimentation : AC/DC adaptateur 100-240V~50/60Hz 0.35A (9V_1A).
- Moteur pas-à-pas - 5V, 32 pas, réduction 1/16 + driver STP01 (UNL2003).
- Câbles (PSG-JMP150FF – PSG-JMP150MM- PSG-JMP150MF).
- Bouton poussoir
- Afficheur LCD 2x16 I2C DFR0063
- MB102 platine de prototypage Module d'alimentation 3.3V 5V pour Arduino.
- Le buzzer

Technologies :

IDE Arduino

III.5 Pilulier

a. Description :

Le boîtier pilulier est fabriqué en plastique et forex avec une géométrie cylindrique d'un diamètre de 20 cm et d'une hauteur de 11 cm, divisé en 6 chambres pour chaque chambre 60 degrés. Il est composé de deux étages (un étage pour les chambres et un autre pour les composants) avec l'ajout de roulement dans le centre pour faciliter la rotation.

Représentation la maquette en 3D qu'on a réalisé avec le logiciel AutoCAD dans la figure III.1 :

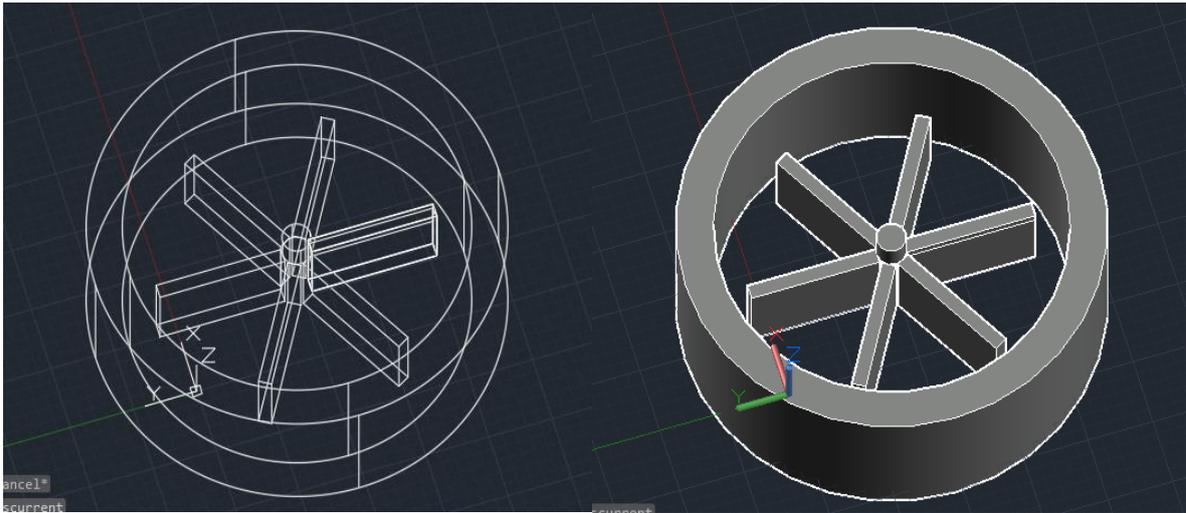


Figure III.1 : La maquette en 3D

b. Montage :

On a deux parties dans le boîtier pilulier :

La partie statique, qui est un plateau cylindrique (fig. III.2) :



Figure III.2 : la partie statique

La partie mobile se compose de tige centrée, roulement et 6 chambres(fig.III.3) :

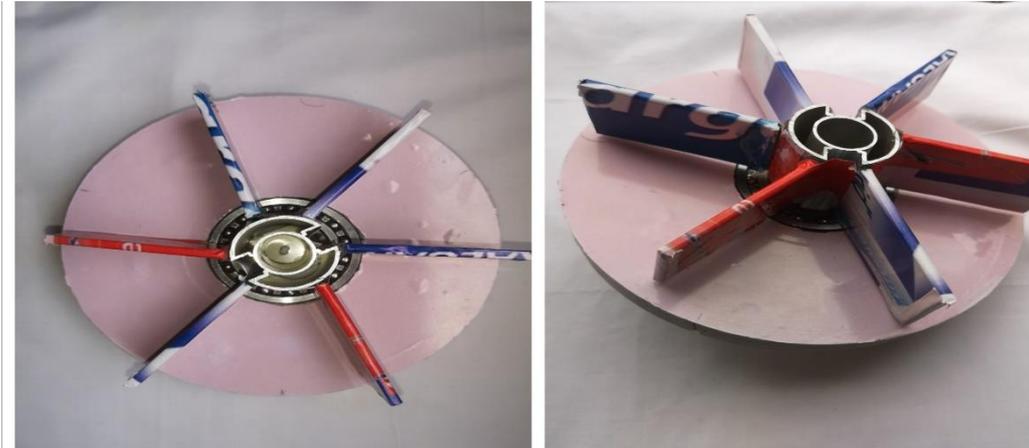


Figure III.3 : *la partie mobile*

Le système est muni d'un moteur pas à pas qui fait tourner la partie mobile avec le degré programmé correspondant au numéro de chambre concerné (fig.III.4)



Figure III.4 : *La maquette complète*

III.6 Choix des composants et justification :

a. Arduino Uno :

La carte Arduino Uno est basée sur le microcontrôleur ATmega328. Elle dispose de 14 entrées/sorties numériques (dont 6 peuvent être utilisées comme sorties PWM), 6 entrées analogiques, d'un oscillateur à quartz de 16 MHz, d'une connexion USB, d'une prise d'alimentation, un connecteur ICSP (In Circuit Serial Programming), et un bouton de reset. Elle contient tout le nécessaire pour piloter le microcontrôleur, il suffit simplement de le connecter

Un ordinateur avec un câble USB pour l'utiliser simplement(fig.III.5) :[21].



FigureIII.5 : La carte Arduino UNO

Caractéristiques :

Toutes les caractéristiques de la carte Arduino UNO sont exposées dans la figure suivante (fig.III.6) [22]

Microcontrôleur	ATmega328
Tension de fonctionnement	5V
Tension d'alimentation (recommandée)	7-12V
Tension d'alimentation (limites)	6-20V
Broches E/S numériques	14 (dont 6 disposent d'une sortie PWM)
Broches d'entrées analogiques	6 (utilisables en broches E/S numériques)
Intensité maxi disponible par broche E/S (5V)	40 mA (ATTENTION : 200mA cumulé pour l'ensemble des broches E/S)
Intensité maxi disponible pour la sortie 3.3V	50 mA
Intensité maxi disponible pour la sortie 5V	Fonction de l'alimentation utilisée - 500 mA max si port USB utilisé seul
Mémoire Programme Flash	32 KB (ATmega328) dont 0.5 KB sont utilisés par le bootloader
Mémoire SRAM (mémoire volatile)	2 KB (ATmega328)
Mémoire EEPROM (mémoire non volatile)	1 KB (ATmega328)
Vitesse d'horloge	16 MHz

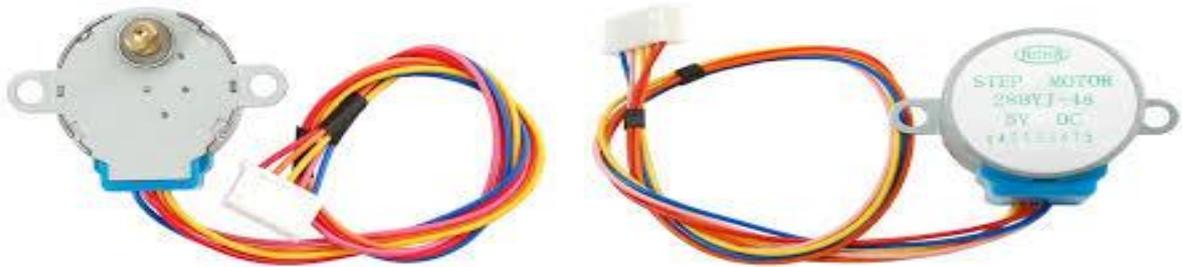
Figure III.6 : les caractéristiques de carte Arduino UNO

b. Moteur pas à pas

C'est un excellent moteur pour de petits projets. Ce moteur unipolaire dispose de deux emplacements/trous de montage solidaires du moteur. Le moteur ne dispose pas seulement de 32 pas (11.25 degrés/pas) par révolution mais il intègre un engrenage réducteur ayant un 1/16 (1/16.032 pour être précis).

Cela signifie que ce moteur dispose vraiment de 32×16.032 pas par révolution = 513 pas !

L'axe est aplati, il est par conséquent facile d'y attacher du matériel à l'aide d'un peu de visserie. C'est un parfait moteur pas-à-pas pour une première expérience, la réalisation de petite plateforme robotique et il



fonctionne bien avec le Shield moteur (motorshield) pour Arduino(fig.III.7) [23].

Figure III.7 : Moteur pas à pas

Caractéristiques

- Moteur pas à pas unipolaire avec connecteur 5 pins à 2,54mm
- 32 pas par révolution du moteur (sans réduction)
- Réduction à 1/16.025
- Alimentation 5V
- Poids : 37g
- Dimensions : Diamètre 28mm Hauteur 20mm et arbre de 9mm avec diamètre 5mm
- Force à 12V : 15 N*mm

c. Driver moteur STP01(UNL2003)

Un Driver pilotant un moteur pas à pas peut se diviser en 2 fonctions :

L'alimentation du moteur avec ses contraintes de tensions, courants et puissances à dissiper ainsi que la protection contre les surtensions et les surintensités.

Le séquenceur qui gère la chronologie des impulsions :

- Du sens de rotation désiré du moteur.
- De la Vitesse du moteur
- De mise en sécurité du driver



Figure III.8: Driver moteur STP01 (UNL2003)

Caractéristiques

- Il est équipé d'une interface standard pour les moteurs pas à pas et peut être inséré directement pendant l'utilisation.
- Taille : 42mmx30mm
- Utilise la puce de pilote ULN2003, 500mA
- Pas de trou : standard 2,54 mm
- A.B. C. D LED indiquant l'état de fonctionnement du moteur pas à pas à quatre phases
- La prise blanche est la prise standard du moteur pas à pas à quatre phases
- Les broches d'alimentation sont séparées

d. AC/DC adaptateur 100-240V~50/60Hz 0.35A (9V_1A).

Adaptateur est alimenté par une source de tension externe, pour cela on a utilisé une alimentation 'AC 100-240v 50/60 Hz' de 9v/1A. (Fig. III.9)



Figure III.9: Adaptateur 9v-1v

Caractéristiques

- Diamètre extérieur : environ 5.5mm.
- Diamètre intérieur : 2.1-2.5mm (réglable).
- Type d'adaptateur : Adaptateur secteur (norme UE).
- Tension d'entrée : AC 100 240V, 50 / 60Hz.
- Tension de sortie : DC 9V.
- Courant de sortie : 1A.
- Léger et portable, facile à utiliser.

e. Afficheur LCD 2x16 I2C DFR0063 :

Afficheur LCD 2 x 16 caractères rétro-éclairé se raccordant via le bus I2C sur un microcontrôleur compatible Arduino®. L'adresse peut être configurée de 0x20 à 0x27 via 3 jumpers.

Le module se raccorde sur une carte compatible Arduino par l'intermédiaire de la carte multi-interfaces DFR0074 via un câble 4 contacts non inclus. Il peut aussi se raccorder directement sur une carte Arduino® via des câbles de liaison non inclus (Fig.III.10)[24].



Figure III.10 : *Afficheur LCD*

Caractéristiques

- Alimentation : 5 Vcc
- Interface I2C (adresses 0x27)
- Caractères de couleur blanche sur fond bleu
- Contraste ajustable via potentiomètre
- Dimensions : 80 x 38 x 18 mm

f. MB102 platine de prototypage Module d'alimentation 3.3V 5V pour Arduino :

Il s'agit évidemment d'une planche double face qui est assemblée à la machine selon un bon standard de qualité. Il y a deux jeux de broches d'en-tête de montage, et ceux à l'avant sont si on veut utiliser les trous sur le bord extérieur de la platine de prototypage, utilisant ainsi un moins grand nombre de points d'attache.



Figure III.11 : *Alimentation pour Arduino MB102*

Caractéristiques

- Module d'alimentation de platine de prototypage, compatible avec 5 V, 3.3 V Appliquer à MB102 platine de prototypage.
- Tension d'entrée : 6.5-12 V (cc) ou alimentation USB.
- Tension de sortie : 3.3 V/5 V peut basculer.
- Courant de sortie maximal : <700 ma.
- Deux groupes embarqués de 3.3 V, broche de sortie cc 5 V, utilisation de plomb externe pratique.
- Taille : 5.3 cm x 3.5 cm.

g. Câbles (PSG-JMP150FF – PSG-JMP150MM- PSG-JMP150MF).

Pour établir des connexions entre les éléments du système les broches d'en-tête d'Arduino, on a relié les câbles par des jumpers. (**Fig.III.12**)



Figure III.12 : Les câbles**h. Le buzzer :**

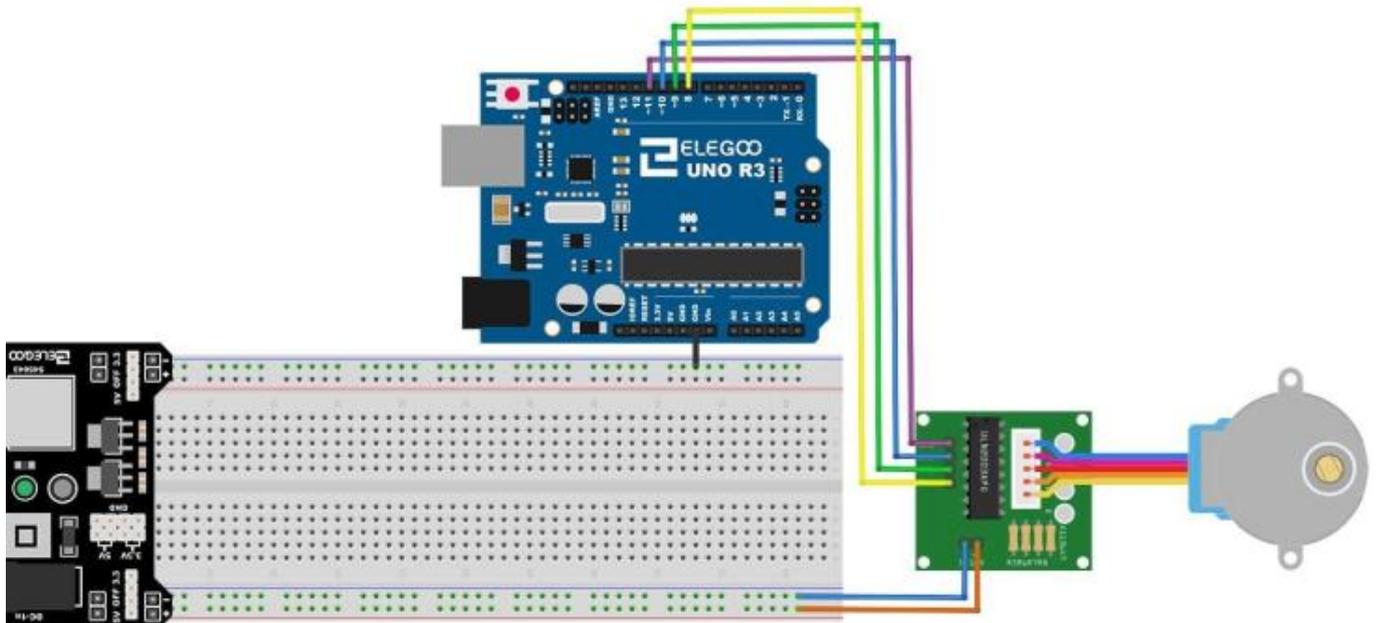
. Quand il est l'heure pour le patient de prendre ses médicaments, le buzzer sonne de façon discontinue. Or, quand il se trompe de case, le buzzer l'alerte de manière continue, moins agréable à l'oreille, ce qui permet de faire réagir le patient rapidement. (Fig.III.13)

**Figure III.13 : Le buzzer****Caractéristiques**

- Tension de fonctionnement : 3.3 - 5 V
- Signal I/O : Numérique TTL (0 ou 1)
- Dimensions : 33 x 13 mm

III.7 Simulation et test avec Arduino :**a. La carte Arduino avec le moteur pas à pas et UNL2003 :**

Pour notre projet on a utilisé le moteur pas à pas pour tourner le pilulier (fig.III.14)

**Figure III.14 : branchement Arduino avec le moteur pas à pas et UNL2003**

Le branchement réel de Arduino avec le moteur pas à pas et UNL2003(fig.III.15)

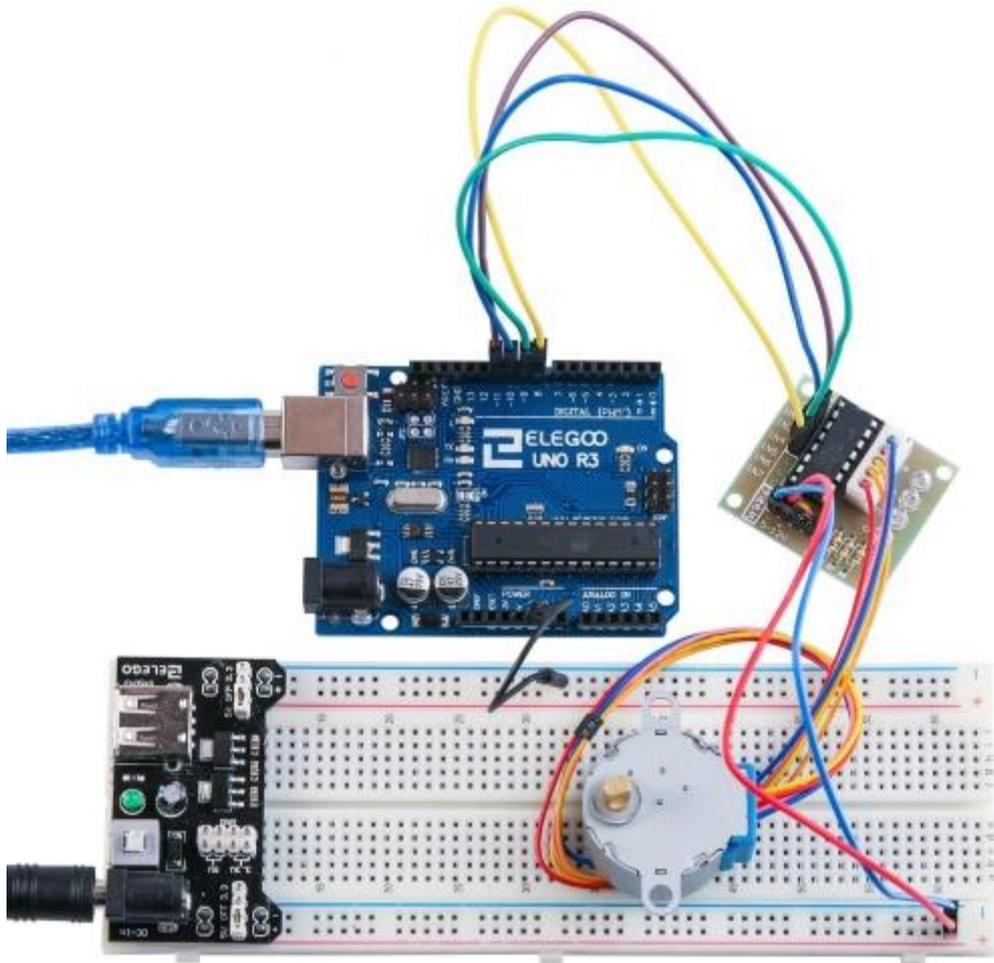


Figure III.15 : *branchement réel Arduino avec le moteur pas à pas et UNL2003*

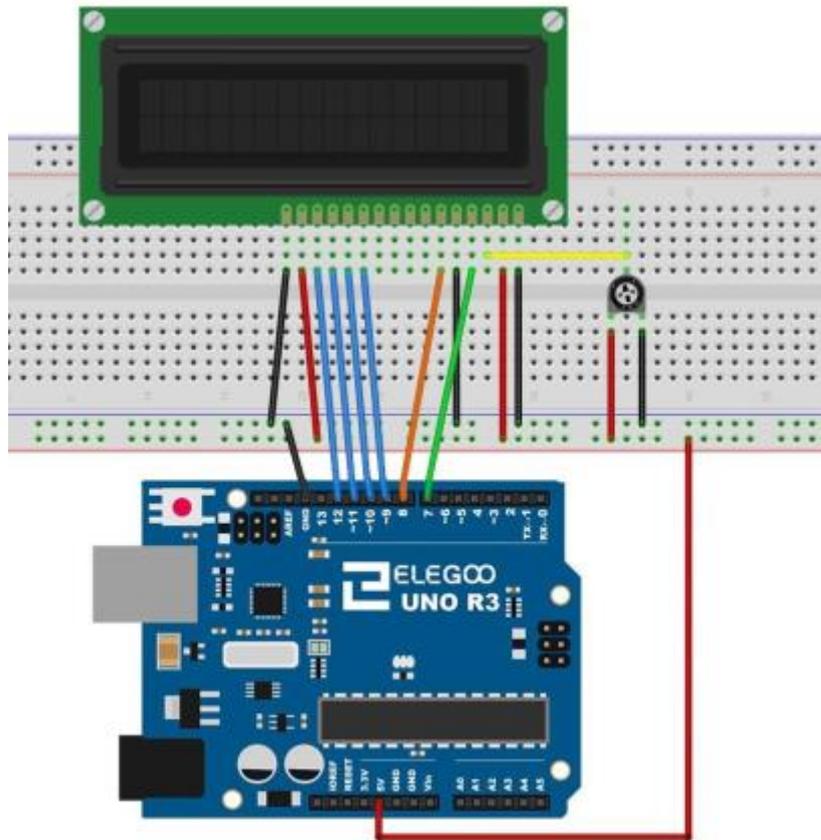
Le branchement de moteur pas à pas et UNL2003 avec Arduino :

Arduino	Module driver moteur
+5V	+
GND	-
8	IN1
9	IN2
10	IN3
11	IN4

Tableau III-1 : *Le branchement de moteur pas à pas et UNL2003 avec Arduino :*

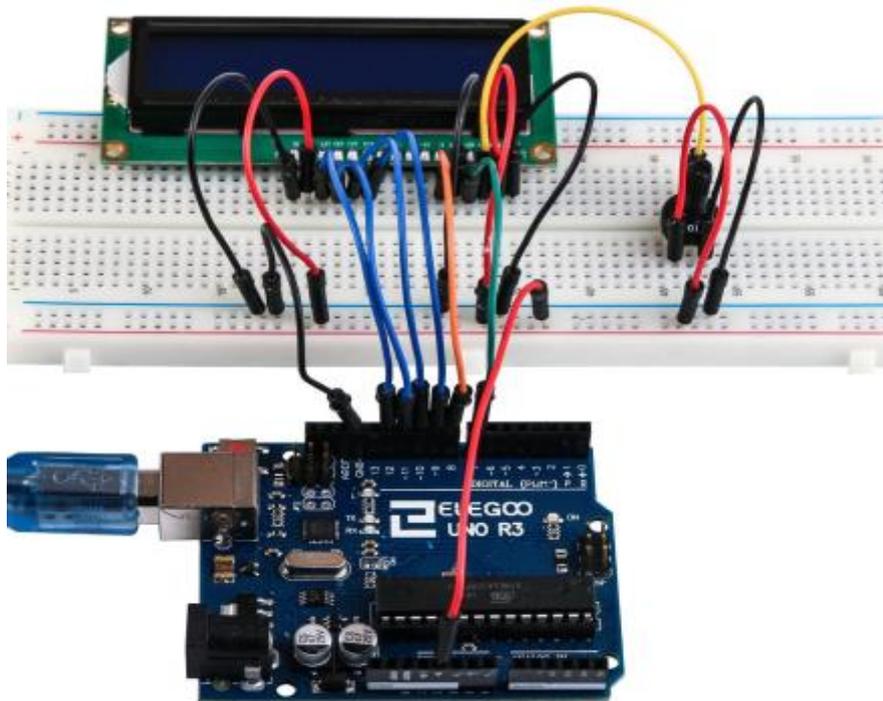
b. La carte Arduino avec afficheur LCD

Le branchement d'afficheur LCD avec Arduino est comme la figure suivante (Fig.III.16) :



FigureIII.16 : *branchement Arduino avec LCD*

Le branchement réel d'afficheur LCD avec Arduino est comme (fig.III.17) :



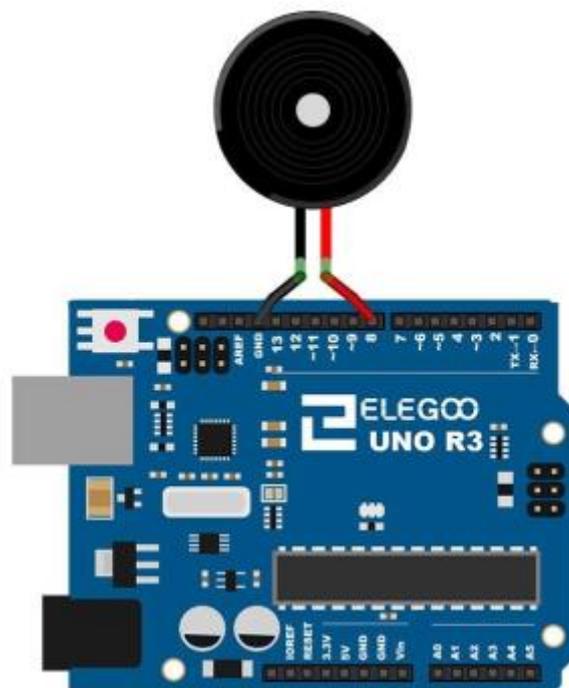
FigureIII.17: *branchement réel Arduino avec LCD*

Tableau de branchement d'afficheur LCD avec Arduino :

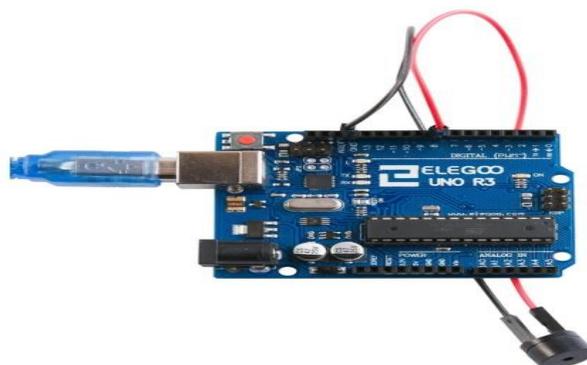
Afficheur LCD I2C	Arduino
GND	GND
VCC	+5V
SDA	A4
SCL	A5

Tableau III-2 : *Tableau de branchement d'afficheur LCD avec Arduino***c. La carte Arduino avec buzzer :**

Le branchement de buzzer avec Arduino est donné par la figure suivante (fig.III.18) :

**Figure III.18 :** *branchement Arduino avec buzzer*

Le branchement réel de buzzer avec Arduino est comme la figure suivante (fig.III.19) :

**Figure III.19 :** *branchement réel Arduino avec buzzer*

III.8 Programmation avec Arduino :

Pour programmer notre projet il faut utiliser les bibliothèques suivantes :

```
Stepper.h:#include<Stepper.h>
LiquidCrystal.h:#include<LiquidCrystal.h>
```

Pour utiliser la bibliothèque **Stepper.h** faut d'abord déclarer :

✓ La révolution de notre moteur :

```
const int stepsPerRevolution = 1500; // modifiez-le pour qu'il corresponde au nombre de pas par tour
```

Après les essais on a trouvé `stepsPerRevolution = 2040`

Donc pour programmer chaque chambre avec son angle dans notre programme nous prenons les valeurs suivantes

```
60° → 340
120° → 680
180° → 1020
240° → 1360
300° → 1700
360° → 2040
```

Aussi la déclaration des variables :

```
int val;
int t;
int button = 2;
int buzzer = 3;
```

Aussi nous avons déclaré le bouton poussoir comme une entrée **INPUT** et le buzzer comme une sortie **OUTPUT**

```
pinMode (button, INPUT_PULLUP);
pinMode (buzzer, OUTPUT);
```

Enfin on a fait le programme pour le premier médicament qui se trouve dans la première chambre, et le même programme est répété pour tous les quatre autres chambres on a changé juste angle de rotation.

Et voilà le programme final

```
#include <Stepper.h>
#include <LiquidCrystal.h>
const int stepsPerRevolution = 1500; // change this to fit the number of steps per revolution
int val;
int t;
int button = 2;
int buzzer = 3;
// initialize the stepper library on pins 8 through 11:
Stepper myStepper(stepsPerRevolution, 4, 6, 5, 7);

LiquidCrystal lcd(8, 9, 10, 11, 12, 13);
void setup() {

  lcd.begin(16,2);
  lcd.print("Auto Pill");
  // set the speed at 20 rpm:
  myStepper.setSpeed(20);
  // initialize the serial port:
  Serial.begin(9600);
  pinMode(button, INPUT_PULLUP);
  pinMode(buzzer, OUTPUT);
}

}

void loop() {

  if(digitalRead(button)==LOW){
    box1();
  }
  if (Serial.available()>0)
  {
    val = Serial.parseInt();
    myStepper.step(val);
    Serial.println(val); //for debugging
  }

}

void box1() {
  delay(5000);
  myStepper.step(340);
  lcd.setCursor(0,1);
  lcd.print("medicament 1");
  ...
}
```

```
do{
  digitalWrite (buzzer, HIGH);
  delay(4);
  digitalWrite (buzzer, LOW);
  delay(4);
}while (digitalRead (button)==HIGH);
myStepper.step(-340);
box2();
}
void box2 () {
  delay(5000);
  myStepper.step(680);
  lcd.setCursor(0,1);
  lcd.print("medicament 2");
  do{
    digitalWrite (buzzer, HIGH);
    delay(1);
    digitalWrite (buzzer, LOW);
    delay(1);
  }while (digitalRead (button)==HIGH);
  myStepper.step(-680);
  box3();
}
void box3 () {
  delay(5000);
  myStepper.step(1020);
  lcd.setCursor(0,1);
  lcd.print("medicament 3");
  do{
    digitalWrite (buzzer, HIGH);
    delay(1);
    digitalWrite (buzzer, LOW);
    delay(1);
  }while (digitalRead (button)==HIGH);
  myStepper.step(-1020);
  box4();
}
void box4 () {
  delay(5000);
  myStepper.step(1360);
  lcd.setCursor(0,1);
  lcd.print("medicament 4");
  do{
    digitalWrite (buzzer, HIGH);
    delay(1);
    digitalWrite (buzzer, LOW);
    delay(1);
  }while (digitalRead (button)==HIGH);
  myStepper.step(-1360);
  box5();
}
void box5 () {
  delay(5000);
  myStepper.step(1700);
  lcd.setCursor(0,1);
  lcd.print("medicament 5");
  .
```

```
do{
  digitalWrite(buzzer,HIGH);
  delay(1);
  digitalWrite(buzzer,LOW);
  delay(1);
}while(digitalRead(button)==HIGH);
myStepper.step(-1700);
}
```

III.9 Description de fonctionnalité du pilulier réalisé :

III.9.1 Comment le verrouiller ?

Quant à la façon de le verrouiller nous avons mis un petit cadenas entre le couvercle et la boîte afin qu'elle puisse être fermée manuellement par la personne qui s'occupe du patient, ce dernier ne peut pas accéder à la clé et donc peut obtenir uniquement les médicaments qui lui sont fournis.

III.9.2 Comment les pilules sont-elles distribuées ?

- Les heures de prise de médicaments ont été saisies au préalable.
- Les médicaments sont fournis à partir de l'emplacement désigné, par rotation du plateau mobile
- Le plateau mobile revient à sa position initiale après chaque distribution pour s'assurer qu'aucun médicament n'est pris accidentellement.

III.9.3 quelles sont les alertes ?

- Le patient est alerté par une alarme sonore et visuelle lorsqu'un médicament doit être pris.

III.9.4 Affichage :

Nous avons utilisé afficheur LCD ce qui nous permet d'afficher le temps jusqu'à l'heure de prise du médicament par le patient, puis afficher le nom du médicament et la quantité allouée pour ce temps

III.9.5 Fonctions prises en charge :

Le bon fonctionnement du système exige le respect des conditions suivantes :

❖ Le pilulier doit rendre la prise du médicament plus simple puisque le malade n'a plus besoin de se rappeler de les prendre, c'est le pilulier qui s'encharge. Il doit pouvoir prendre les médicaments facilement, et ce malgré son handicap moteur. Le pilulier ne doit pas compliquer la vie des aides-soignants, dans sa programmation comme dans son chargement hebdomadaire.

❖ Le pilulier présente une sécurité, il doit empêcher le malade d'avoir accès à l'intégralité des médicaments en dehors des heures prévues.

❖ La fonction secondaire est de vérifier que les médicaments sont bien pris et d'alerter si besoin.

III.9.6 Les améliorations :

Notre pilulier n'en étant qu'à son premier prototype, de nombreux points restent à modifier et à améliorer en voici une liste non exhaustive :

1. Réduire au maximum la boîte en termes de taille et de poids.
2. Un changement du matériau du pvc et bois en plastique souple.
3. Intégrer l'alerte sur l'expiration du médicament
4. Développer un panneau de contrôle pour faciliter la programmation par la personne concernée.
5. Envoyer un email à la personne concernée lorsque les médicaments sont épuisés.
6. Développement du processus de déverrouillage (déverrouillage avec une empreinte digitale uniquement pour la personne concernée).
7. Faire un pilulier automatique spécial pour 3 ou 4 personnes dans la même chambre à l'hôpital (par exemple surtout pour un homme âgé et sa femme).
8. Changer le buzzer par un haut-parleur décrivant le type et la quantité de médicament.
9. Ajouter un support pour faciliter le processus de transport.
10. Le plus gros point faible du pilulier réside dans la partie mécanique, bien que le schéma de principe semble fonctionnel, celui-ci reste superficiel et des choses comme les liaisons entre les pièces, les sécurités pour résister au forçage ou aux chutes sont à repenser.
11. Il est très important d'ajouter des sécurités logicielles qui permettent d'éviter les bugs ou redémarrer le pilulier. Une autre sécurité serait celle face au piratage. En se connectant en ligne on s'expose à ce que quelqu'un de mal intentionné prenne contrôle du pilulier et mette en danger le patient.

III.9.7 Test :

Nous avons effectué un test sur 2 jours de notre pilulier automatique.

Après 2 jours de test voici les résultats et nos ressentis :

- Le moteur "pas à pas" ne supporte pas le poids du plateau comme il faut mais répond tout de même bien à la consigne de l'Arduino.
- Le moteur tourne bien d'un angle de 60°.
- Les alertes sonores fonctionnent bien. L'alerte sonore est suffisante, juste assez pour prévenir un patient.
- Il semble que le bouton poussoir fonctionne correctement ce qui signifie que lorsqu'il est enfoncé il revient à l'état initial.
- Alimentation électrique fonctionne bien.
- Le couvercle n'obstrue pas le moteur et les chambre tournent facilement
- Bilan du test : tout semble fonctionnel.

Quelques changements à faire tout de même au niveau moteur, et bien sûr finaliser et mettre au point le pilulier pour permettre de le tester en conditions réelles devant un patient.

III.10 Conclusion

Dans ce chapitre, nous avons présenté en détail les étapes de réalisation de notre système. Nous avons commencé par la description de notre pilulier et la justification du choix des composants avec la détermination des caractéristiques de chaque composant.

Ensuite, Nous avons décrit et réalisé la phase de simulation et de programmation avec Arduino qui est validé par l'étape suivante 'réalisation finale'.

Enfin, nous avons présenté le fonctionnement général de notre projet et les tests que nous avons fait sur le prototype.

Conclusion générale :

Dans ce mémoire nous avons présenté une étude pratique qui explique la conception et la réalisation d'un pilulier automatique, l'objectif de ce système est de faciliter le processus de la prise de médicaments pour les patients et de les aider à une meilleure observance thérapeutique.

Dans ce projet, nous avons fabriqué une maquette d'un pilulier automatique qui contient 6 chambres avec différents médicaments.

Ce projet de réalisation nous a été une expérience très fructueuse, bénéfique et enrichissante, il nous a permis d'exploiter les différentes notions de programmation par l'Arduino. Dans une Première étape nous avons conçu la partie opérative, puis nous avons développé un programme de commande par ordinateur permettant de commander le moteur pas à pas qui fait tourner les 6 chambres contenant les différents médicaments. Ensuite nous avons installé et programmé un afficheur LCD pour afficher le nom de médicament puis nous avons installé et programmé un buzzer qui sonne quand il est temps de prendre des médicaments.

Nous avons cherché à rendre le coût de réalisation du projet le plus économique possible. Le coût de ce projet est estimé à 22700 DA. L'intérêt de ce projet a été double pour nous. D'une part il servira comme un prototype réaliste permettant de faire des démonstrations pour les professionnels qui voudraient acquérir ce système, et d'autre part constituera une maquette didactique pour les étudiants en travaux pratiques d'automatisation et de conception d'Interfaces Homme-Machine.

Ce travail reste perfectible, nous recommandons d'en améliorer la conception et la réalisation et pour cela nous proposons ci-dessous quelques améliorations pour les futurs développements :

- envisager d'envoyer un email à la personne concernée lorsque le médicament est épuisé.
- développement du processus de déverrouillage (déverrouillage avec une empreinte digitale uniquement pour la personne concerné).

Bibliographies

[1] : Site consulté en ligne :

www.illicopharma.com/blog/bien-suivre-traitement-en-deplacement-les-avantages-dopter-pilulier/

consulté le 10-08-2020

[2] : Site consulté en ligne :

www.illicopharma.com/blog/bien-suivre-traitement-en-deplacement-les-avantages-dopter-pilulier/Le pilulier connecté : une nouvelle tendance, illico pharma ,consulté le 30-10-2020

[3] : Site consulté en ligne

WWW.PHARMIDEA.FR/LE-PILULIER-A-QUOI-CA-SERT-EXACTEMENT/Les avantages de ce dispositif, PHARMIDEA, consulté le 31-10-2020

[4] : WWW.PHARMIDEA.FR/LE-PILULIER-A-QUOI-CA-SERT-EXACTEMENT/Les types de piluliers, PHARMIDEA, consulté le 02-09-2020

[5] : Conseils pour le choix des piluliers, "Les Bonnes Pratiques de réalisation des piluliers" – version 18 mars 2015, OMÉDIT Centre Val de Loire

[6] : Cas particulier du patient autonome qui gère ses prises, "Les Bonnes Pratiques de réalisation des piluliers" – version 18 mars 2015, OMÉDIT Centre Val de Loire

[7] : Hygiène, "Les Bonnes Pratiques de réalisation des piluliers" – version 18 mars 2015, OMÉDIT Centre Val de Loire

[8] : Site consulté en ligne :

<https://wiki.md129.net/.../fetch.php...Arduino-pour-bien-commencer-en-électronique.pdf>, consulté le 11-09-2020

[9] : Site consulté en ligne :

Jean-Noël Montagné, « Livret Arduino en français », Centre de Ressources Art Sensitif sous licence CC, novembre 2006. consulté le 04-10-2020

[10] : Site consulté en ligne :

https://arduino.technologiescollege.fr/IMG/pdf/cahier_0_initialisation. consulté le 12-09-2020

[11] : Site consulté en ligne :

Logiciel Arduino 1.8.1, outil capteur. PC, 2019. consulté le 16-09-2020

[12] : Site consulté en ligne :

Bibliographie

<https://www.arduino-france.com/review/arduino-uno/Qu'est-ce-que-l-Arduino-Uno-?>, consulté le 20-09-2020

[13] : Site consulté en ligne :

<https://www.javatpoint.com/arduino-uno-pinout>, consulté le 22-09-2020

[14] : Site consulté en ligne :

<http://schemobotics.com/product/arduino-uno-rev3/Properties>, consulté le 27-09-2020

[15] : Site consulté en ligne :

<https://www.locoduino.org/spip.php?article16/comment-alimenter-l-Arduino-sans-ordinateur-?>, consulté le 29-09-2020

[16] : Site consulté en ligne :

<https://quai-lab.com/arduino-ses-memoires/> consulté le 06-10-2020

[17] : Site consulté en ligne :

<http://dspace.univ-tlemcen.dz/bitstream/112/11743/1/Ms.ELN.Chigango%2BMapuranga.pdf>, consulté le 19-10-2020

[18] : Site consulté en ligne :

<https://boutique.semageek.com/fr/354-petit-moteur-pas-a-pas-5v-avec-reducteur.html>, consulté le 20-10-2020

[19] : Site consulté en ligne :

<http://www.arobose.com/shop/microcontrolleurs-robot/53-carte-arduino-uno-rev-3.html>, consulté le 29-10-2020

[20] : Site consulté en ligne :

http://www.mon-club-elec.fr/pmwiki_reference_arduino/pmwiki.php?n=Main.MaterielUno, consulté le 07-09-2020

/ synthèse des caractéristiques

[21] : Site consulté en ligne :

<https://shop.mchobby.be/fr/moteurs-pas-a-pas/767-moteur-pas-a-pas-5v-32-pas-reduction-1-16-3232100007673.html>, consulté le 09-10-2020

[22] : Site consulté en ligne :

<https://www.smart-cube.biz/produit/afficheur-lcd-i2c-16x02/> description , consulté le 13-10-2020

Résumé :

Ce travail a porté sur la réalisation d'un pilulier automatique à base d'une carte Arduino et un moteur pas à pas. Ce système permet d'aider les patients à organiser leurs prises de médicaments.

Nous avons commencé, tout d'abord, par la construction d'une maquette composée de six chambres qui contiennent les médicaments, après nous avons intégré les composants nécessaires pour la réalisation de projet.

Dans une autre étape, nous avons commencé la programmation sur la carte Arduino avec le logiciel Arduino IDE, et nous avons fait beaucoup des tests pour trouver le meilleur programme compatible avec notre projet.

Enfin, nous avons réussi à concevoir une maquette prototype commandée par la carte Arduino qui est synchronisée selon les besoins des patients.

Mots-clés : Arduino, moteur pas à pas, pilulier automatique, synchronisation, Arduino IDE, médicaments, patients, programmation.

Abstract :

This work resulted in an automatic pill dispenser based on an Arduino board and a stepper motor that helps patients organize their medications.

We started, first, by building a mockup consisting of six chambers that contained the drug, afterwards we obtained the necessary materials for the realization of the project.

Then, we started programming on the Arduino board with the Arduino IDE software, and we did a lot of testing to find the best program compatible with our project.

Finally, we succeeded in designing a clean model controlled by the Arduino board which is synchronized according to the needs of the patients.

Keywords: Arduino, stepper motor, automatic pillbox, synchronization, Arduino IDE, drugs, patients, programming.