

PROJETS ARDUINO -- PEIP2

Année scolaire 2020-2021

“PHARMARDUINO”

Étudiants : Elisa ROUX et Yana RAGOZINA

Encadrants : Pascal MASSON

Rapport bibliographique

Sommaire

1. Descriptif du projet	p.3
2. Objectifs	p.3
3. Modélisation du projet	p.4
4. Solutions trouvées	p.5
a. Créer le corps de la machine	p.5
b. Établir la communication entre l'utilisateur et la machine	p.6
c. Créer l'élément moteur	p.7
d. Faire tomber les médicaments	p.12
e. Alimenter la machine	p.14
f. Contrôler le nombre de médicaments restants dans une case	p.17
g. Vérifier et alerter l'utilisateur lorsque le stock est vide	p.17
h. Assurer la sauvegarde en mémoire du stock	p.18
i. Encoder le nom du médicament	p.18
j. Transmettre l'ordonnance à la carte Arduino	p.20
5. Conclusion	p.24
6. Récapitulatif	p.24

- **Descriptif du projet**

Une machine qui, à partir d'une ordonnance sauvegardée sur un support particulier tel qu'un code, permet de délivrer le bon nombre de boîtes de médicaments automatiquement. Elle sera munie d'une application permettant de traiter et transmettre l'ordonnance ainsi que de gérer les stocks de médicaments.

- **Objectifs**

- ➔ **Créer le corps du distributeur**

- Trouver les matériaux pour le corps de la machine
 - Etablir la communication entre l'utilisateur et la machine

- ➔ **Créer les mécanismes mobiles**

- Créer l'élément moteur
 - Faire tomber les médicaments
 - Alimenter la machine

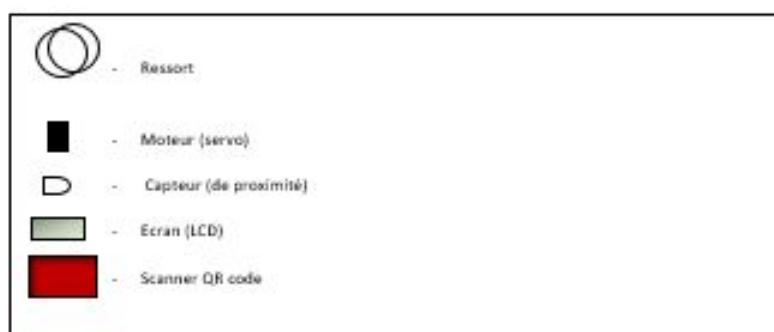
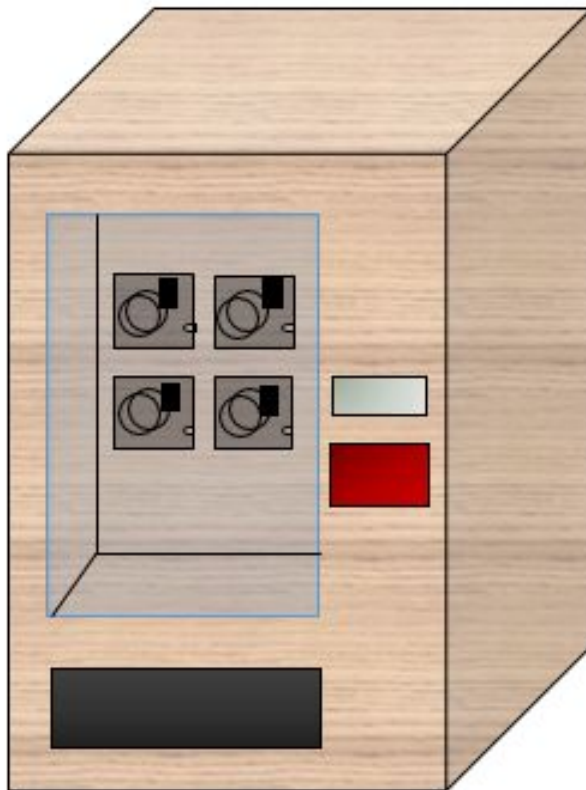
- ➔ **Gérer les stocks**

- Contrôler le nombre de médicaments restants dans une case
 - Vérifier et alerter l'utilisateur lorsque le stock est vide
 - Assurer le sauvegarde en mémoire du stock

- ➔ **Assurer la communication entre la carte Arduino et la machine**

- Encoder le nom du médicament
 - Transmettre l'ordonnance à la carte Arduino

- Modélisation du projet



Ressort

Moteur (servo)

Capteur (proximité)

Ecran (LCD)

Scanner QR code

- Solutions trouvées

→ Créer le corps du distributeur

Tout d'abord, nous cherchons à créer une machine qui serait non seulement fonctionnelle et pratique, mais qui assurerait aussi une bonne expérience pour l'utilisateur. Le distributeur doit également correspondre aux critères d'une machine moderne: éco-responsable, durable, facile à exploiter, intégrant des technologies innovantes et qui doit être esthétique.

➤ Trouver les matériaux pour le corps de la machine

Nous souhaitons créer un corps minimaliste à partir d'un matériel **solide, écologique et économique qui serait aussi facile à manipuler.**

Objectif: Créer le corps du distributeur		
Solutions existantes pour: Matériaux pour l'extérieur du distributeur		
Solution	+	-
Contreplaqué	écologique léger coût faible facile à manipuler	coupé avec des matériaux spécifiques pas souple
plastique	léger coût très faible découpage facile (selon les types de plastique) facile à manipuler	pas très écologique moins fiable pour une exploitation de longue durée
métal	coût faible résistant	plus lourd coupé avec des appareils spécifiques

	solide	nécessite des procédés particuliers pour coller les parties difficile à manipuler
carton	souple très léger écologique coût faible facile à manipuler	trop léger pour tenir les composants pas fiable pour une exploitation de longue durée
Solution retenue: Contre-plaqué		

Le contreplaqué est, en effet, le choix le plus optimal car il correspond exactement à nos recherches.

Afin de rendre le distributeur plus visuel, nous pourrions également mettre une tranche de verre ou de plastique transparent (qui serait l'option moins écologique mais plus légère et moins fragile).

➤ Etablir la communication avec l'utilisateur

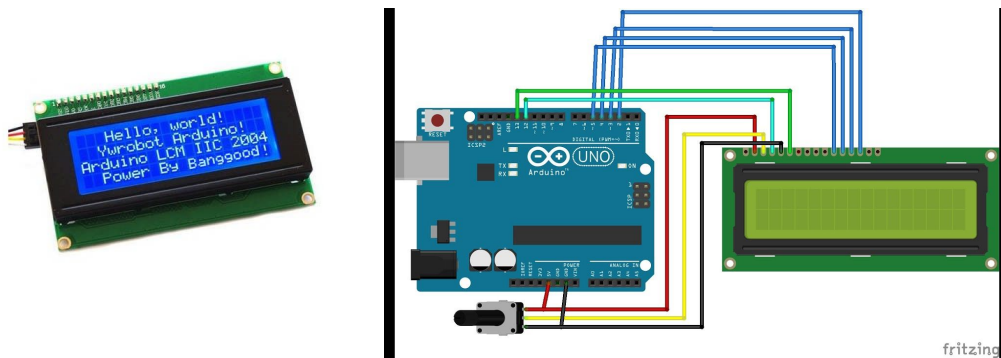
Afin de le rendre attractif, nous devons programmer un distributeur facile à utiliser. Pour cela, nous devons établir un retour des actions du distributeur pour l'utilisateur. Nous avons étudié les éléments de communication visuels et sonores.

Objectif : Créer le corps du distributeur		
Solutions existantes pour : Communiquer avec l'utilisateur		
Solutions	+	-
Afficheur LCD	<ul style="list-style-type: none"> - Affichage clair - Programmation simple - Permet un retour précis 	<ul style="list-style-type: none"> - Petit écran donc pas très visuel
Code lumineux avec des LED	<ul style="list-style-type: none"> - Programmation simple - Attractif visuellement 	<ul style="list-style-type: none"> - Possibilité d'afficher moins d'informations qu'avec l'écran LCD - Retour moins précis

Buzzer	<ul style="list-style-type: none">- Alerte effective	<ul style="list-style-type: none">- Retour difficile à rendre précis- Nécessité de créer différents codes sonores ce qui rajoute de la complexité
Solution retenue : Afficheur LCD		

Nous avons besoin que l'utilisateur sache précisément ce que distribue la machine.

Malgré sa petite taille, l'écran LCD est le choix le plus optimisé. En effet, le distributeur n'est pas très grand à notre échelle. Ainsi, nous n'avons pas besoin de que l'affichage soit visuellement attractif, mais clair et concis.



Ecran LCD et son branchement

[Liens utilisés:](#)

Image du branchement:

<https://i.ytimg.com/vi/pmq9dOTIaAM/maxresdefault.jpg>

Buzzer:

<https://www.ardumotive.com/how-to-use-a-buzzer-en.html>

→ Créer les mécanismes mobiles

Afin de garantir un bon fonctionnement du distributeur, nous devons trouver une solution pour mettre en mouvement les objets qui se trouvent dans les cases.

➤ Créer l'élément moteur

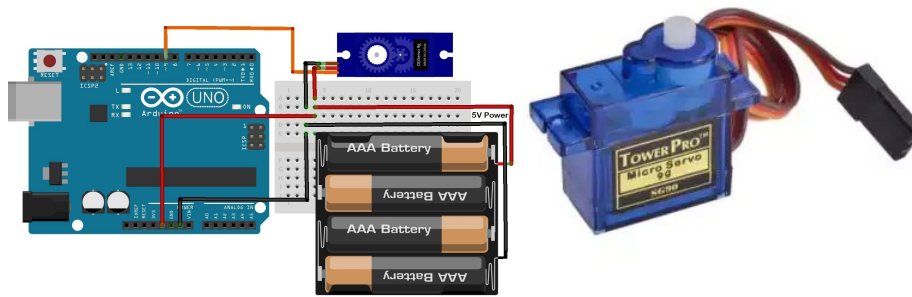
Pour cela, nous avons pensé à fixer un élément moteur alimenté de l'extérieur qui, en effectuant un mouvement, pousserait les objets au fur et à mesure jusqu'à leur chute dans le bac. Ce moteur doit ainsi avoir une **taille assez petite, être facilement programmable et connectable au reste du circuit, peu coûteux et économique.**

Objectif: créer les mécanismes du corps		
<i>Solutions existantes pour: bouger les cases du distributeur</i>		
Solutions	+	-
Servo moteur	<p>Capable de prendre en entrée des positions angulaires</p> <p>Ne nécessite pas de composants auxiliaires</p> <p>circuit compacte</p> <p>Branchement direct sur la sortie 5V</p> <p>Moteur le plus utilisé dans les projets similaires</p> <p>Grand diapason de contrôle des vitesses</p> <p>Capable d'augmenter la puissance</p> <p>Garantit une communication dans les deux sens</p> <p>Multitude des modèles en fonction de l'alimentation, vitesse, couple et la qualité de la mécanique</p> <p>Petite taille</p> <p>Consommation de l'énergie proportionnelle au travail</p>	<p>Il est parfois recommandé d'utiliser un driver ou un condensateur pour protéger la carte des sauts de tension causés par les moteurs</p> <p>Coût un peu plus élevé</p> <p>Possède un système de connexion à la carte plus complexe</p> <p>N'est pas adapté à l'alimentation de la carte via la sortie VIN (risque de surchauffe)</p> <p>Trop grande puissance dans le cas où on connectera plusieurs moteurs ⇒ le branchement directe sur la sortie 5V est insuffisant</p> <p>Nécessite un transistor pour les moteurs 6V</p> <p>Il est recommandé d'utiliser un</p>

	<p>S'éteint de façon autonome lorsque la position angulaire souhaitée est atteinte</p> <p>Contrôle et vérification des position autonome grâce à la communication dans les deux sens</p>	<p>bloc porteur afin d'éviter la torsion de l'arbre</p>
DC moteur	<p>Contrôle de vitesse (pas utile)</p> <p>Contrôle de la direction</p> <p>Moteur puissant</p> <p>Programmation compacte</p> <p>Coût faible</p>	<p>Alimentation 6V pour les versions plus avancées</p> <p>Nécessite une mise en place des encoders pour connaître sa position actuelle</p> <p>Circuit plus lourd pour les moteurs 6V</p>
Stepper moteur	<p>Contrôle de la vitesse et de la direction des "pas"</p> <p>Contrôle précis de la position angulaire</p> <p>Coût faible</p> <p>Fonction similaire au servo moteur</p> <p>Fonctionne pour des rotations extrêmement faibles</p> <p>Capacité de maintenir une position fixée sans consommer d'énergie</p> <p>Multitude de modèles en fonction de la polarité, alimentation et de la réluctance</p> <p>Utilisé dans les mécanismes qui nécessitent une rotation précise (ce qui peut nous servir dans le projet)</p>	<p>Circuit plus ou moins lourd</p> <p>Nécessite l'utilisation de drivers</p> <p>Code plus ou moins complexe</p> <p>Consommation invariante d' énergie</p> <p>Ne possède pas la communication dans les deux sens donc peut causer des pertes de "pas"</p> <p>Puissance faible comparée à celle du servo moteur</p> <p>Vitesse plus faible comparée à celle du servo moteur</p> <p>Nécessité de mise en place de méthodes spécifiques pour garantir l'homogénéité du mouvement</p>

Il est recommandé de choisir une alimentation externe

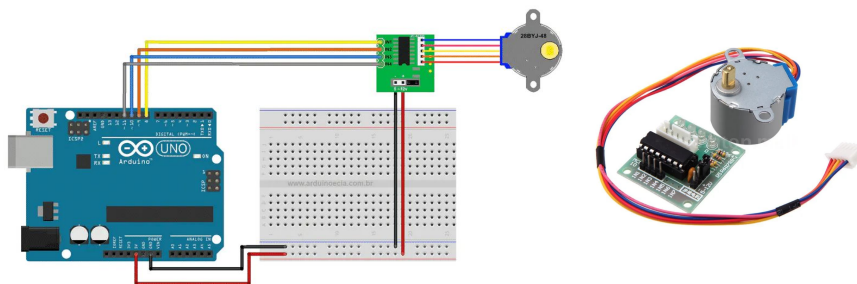
Solution retenue: Servomoteur



Exemple du branchement et un modèle du moteur servo



DC moteur



Exemple du branchement et le stepper moteur

Le moteur DC est le moteur le plus adapté aux machines mobiles dans l'espace. Pour l'adapter à notre projet, il faudrait donc penser à mettre une roue attachée au moteur, elle-même attachée à un ressort. De plus, ce moteur possède des

qualités qui ne nous sont pas utiles dans le projet. Ce n'est pas donc le choix le plus pratique.

Les moteurs Servo et Stepper se ressemblent dans leur fonctionnement : une rotation continue afin de rendre le mouvement du ressort homogène. Toutefois, ils possèdent chacun des caractéristiques différentes.

Le moteur Servo est le moteur le plus utilisé dans les projets type "distributeur" car son fonctionnement est le plus compatible avec le concept d'un ressort mis en mouvement. Il est capable de prendre des positions angulaires exactes tout en optimisant la consommation. En effet, le moteur servo et le ressort ne font qu'un seul mouvement synchronisé qui correspond à un tour. Après avoir pris la nouvelle position angulaire, le moteur s'éteint ce qui lui permet d'économiser une quantité d'énergie relativement importante. Il est aussi très compact avec un circuit simple ne nécessitant pas de composants auxiliaires

En revanche, le stepper moteur, malgré sa capacité à prendre des positions plus précises, est le choix le moins économique en termes d'énergie consommée car sa consommation est constante peu importe l'état actuel du moteur. De plus, comme le moteur ne possède pas la communication dans les deux sens, des erreurs peuvent se produire et on peut perdre certains "pas" sans qu'ils soient corrigés automatiquement, contrairement au moteur servo.

Finalement, pour un fonctionnement similaire, même si le servomoteur est l'option la plus longue à programmer et la plus chère à l'achat, elle est finalement la solution la plus économique, rentable et pratique de par son circuit très compact. En effet, si le stepper moteur et ses composants auxiliaires ont un coût d'achat plus faible, cette économie est contre-balancée par un coût énergétique plus important. Afin de savoir si le moteur aura effectué un tour complet, nous lui ajouterons un capteur de ligne qui détectera si la position angulaire souhaitée est atteinte.

Parmi les différents types de servomoteurs, nous opterons pour le modèle qui sera assez performant sous 9V.

Liens utilisés:

Explication du fonctionnement des moteurs:

Servomoteur:

https://www.tutorialspoint.com/arduino/arduino_servo_motor.htm#:~:text=AdVERTISEMENT,s.angular%20position%20of%20the%20shaft.

<https://www.instructables.com/Arduino-Servo-Motors/>

<https://arduinomaster.ru/motor-dvigatel-privod/servoprivody-arduino-sg90-mg995-shema-podklyucheniye-upravleniye/>

<http://arduino-diy.com/arduino-servomotor>

https://www.tutorialspoint.com/arduino/arduino_servo_motor.htm

https://www.circuitbasics.com/wp-content/uploads/2020/05/Simple-Servo_bb-2.jpg

<https://www.youtube.com/watch?v=J8atdmEqZsc>

DC moteur:

https://www.tutorialspoint.com/arduino/arduino_dc_motor.htm
<http://developer.alexanderklimov.ru/arduino/dc-motor.php>
<https://xn--18-6kcdusowqbt1a4b.xn--p1ai/%D0%BF%D0%BE%D0%B4%D0%BA%D0%BB%D1%8E%D1%87%D0%B5%D0%BD%D0%B8%D0%B5-%D0%BC%D0%BE%D1%82%D0%BE%D1%80%D0%B0-%D0%B0%D1%80%D0%B4%D1%83%D0%B8%D0%BD%D0%BE/>

Stepper moteur:

<https://www.arduino.cc/en/Tutorial/LibraryExamples/StepperSpeedControl>
<https://3d-diy.ru/wiki/arduino-mechanics/stepper-motor-28BYJ-48/>
https://www.youtube.com/watch?v=dZdLurV_JDA&v=ru
<https://learn.adafruit.com/adafruit-arduino-lesson-16-stepper-motors/stepper-motors>
https://www.tutorialspoint.com/arduino/arduino_stepper_motor.htm#:~:text=Advertisements,rotates%20in%20discrete%20step%20angles.
https://hackster.imgix.net/uploads/attachments/753249/stepper-motor-wiring_VwhmFhVIZG.png?auto=compress%2Cformat&w=1280&h=960&fit=max

Comparaison du servomoteur avec le stepper moteur (en russe):

<https://tehprivod.su/poleznaya-informatsiya/servoprivod-ili-shagovyy-dvigatel.html>

Comparaison des servomoteurs:

<https://www.robotshop.com/eu/en/compare-rc-servo-motors.html>

Autres liens:

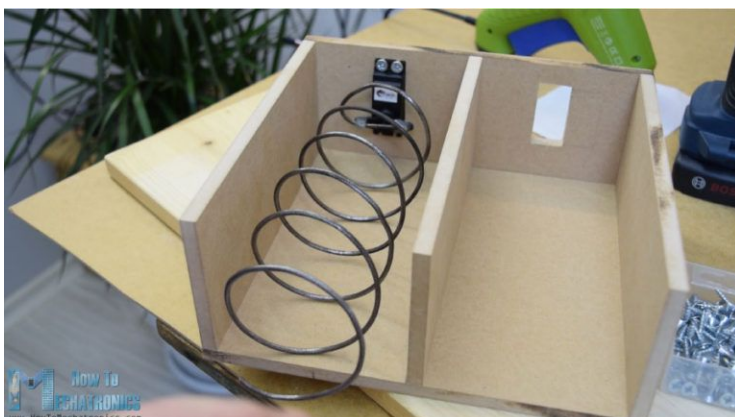
[Mas'19] P. Masson: *Éléments de robotique avec arduino : Moteurs (V12)*
<https://recitmst.qc.ca/arduino/detecteur-de-ligne/> -- capteur de ligne et sa mise en place

➤ Faire tomber les médicaments

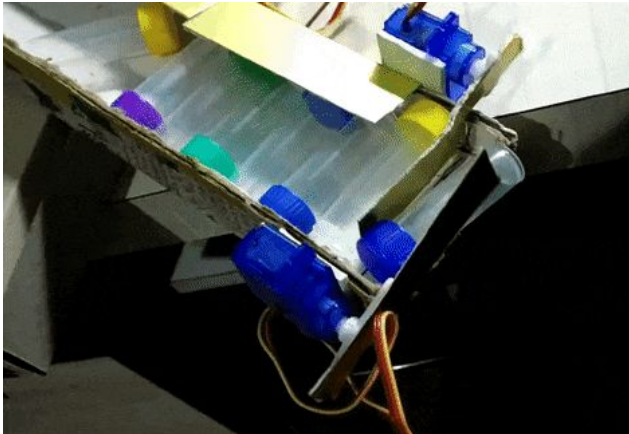
Pour fixer les objets dans les cases et pour les rendre dynamiques, nous avons besoin d'un support qui, une fois mis en mouvement par le moteur, causera la chute d'un seul objet par mouvement programmé. Nous devons donc trouver un élément qui sera **assez solide pour tenir l'ensemble des médicaments et qui garantira la chute d'une seule boîte.**

Objectif: créer les mécanismes du corps
Solutions existantes pour: Faire tomber les médicaments

Solution	+	-
Ressort	<p>Bonne fixation des médicaments</p> <p>Bonne séparation de chaque médicaments</p> <p>Facilement fabricable</p> <p>Pas déformable</p> <p>Garantit la chute d'un seul médicament à la fois</p>	<p>Peut provoquer des erreurs avec les médicaments qui ne sont pas tombés</p>
Trappe	<p>Garantit la chute des médicaments</p> <p>Compacte</p>	<p>Ne garantit pas la chute d'un seul médicament à la fois</p> <p>Peut se déformer à cause du poids du stock selon le matériel choisi</p> <p>Nécessite de la mise en place d'un algorithme spécial ou d'un capteur pour garantir la chute d'un seul médicament</p>
2 barrières	<p>Permet la séparation des médicaments</p> <p>Garantit la chute d'un seul médicament</p>	<p>Nécessite 2 moteurs pour chaque case</p> <p>Peut se déformer à cause du poids du stock selon le matériel choisi</p>
Solution retenue: Ressort		



Ressort+servomoteur



Système des 2 barrières

Les solutions de la barrière ou d'une trappe ne sont visiblement pas optimales pour notre projet car elles ne permettent pas de changer la taille des boîtes et pourraient entraîner des complications dans le programme. Il faudrait aussi mettre en place un système pour éventuellement débloquer les médicaments. Et ces derniers seraient difficiles à combiner avec les matériaux constituant notre distributeur.

C'est pourquoi nous décidons de retenir la solution du ressort qui est la plus économique et pratique. Un calcul d'angle nous permettra de distribuer avec précision une seule boîte à la fois et couplée à un capteur de mouvement, d'assurer une distribution correcte.

Liens utilisés:

Exemples des distributeurs déjà faits:

<https://howtomechatronics.com/projects/diy-vending-machine-arduino-based-mechatronics-project>

https://create.arduino.cc/projecthub/mihir-khara/medicine-reminder-and-vending-machine-18f063?ref=similar&ref_id=348703&offset=2

<https://create.arduino.cc/projecthub/hadamiera/candy-dispenser-a61a60>

<https://www.hackster.io/qatoninja236/raspberry-pi-powered-candy-dispenser-fd018f>

➤ Alimenter la machine

Afin de rendre notre distributeur autonome et mobile nous devons trouver un moyen d'alimenter notre carte arduino sur laquelle tous les composants seront branchés. Ce sera donc une alimentation externe et d'après les recommandations

du constructeur elle devra avoir une tension comprise entre 7V et 12V et un courant de 500mA pour assurer un fonctionnement correct. Notre choix se basera sur des **critères écologiques et économiques**.

Objectif: créer les mécanismes du corps		
<i>Solutions existantes pour: alimentation externe</i>		
Solution	+	-
Adaptateur secteur 200 V -> 5V	<ul style="list-style-type: none"> - Permet une alimentation continue -Ne nécessite pas de rechargement -Compacte -Multitude de modèles avec différentes caractéristiques 	<ul style="list-style-type: none"> -Rend la machine dépendante de la prise -Coût moyennement élevé
Piles (tension comprise en 7V et 12V et un courant de 500mA)	<ul style="list-style-type: none"> -Grande variété -Permet la portabilité et indépendance de la machine -Coût faible -Adaptabilité aux changements dans le circuit (on peut rajouter/enlever des piles si besoin) -Installation facile -Assez durable 	<ul style="list-style-type: none"> -De taille plus ou moins importante -Nécessite un changement des piles en cas de décharge -Solution pas très écologique
Batterie externe (powerbank)	<ul style="list-style-type: none"> -Permet la portabilité et indépendance de la machine 	<ul style="list-style-type: none"> -Nécessite une recharge fréquente -Option assez coûteuse -Solution pas compacte -S'éteint lorsque le courant <200mA pour la

		plupart des powerbanks
Batterie lithium	-Économise de l'énergie	-Fonctionnement en dessous de 3.5-4V n'est pas garanti -Donne un diapason de 3.5-4.2V sur la sortie 5V
Solution retenue: Pile		

La meilleure solution est la pile car, par comparaison, c'est la solution la plus pratique et la plus facile à exploiter. En effet, en cas de dysfonctionnement, nous pouvons facilement remplacer la pile en question en cas de panne ou en rajouter/enlever car c'est aussi un élément toujours disponible en vente à portée de main. De plus, malgré le fait que les piles ne sont pas écologiques, elles restent très durables et économiques. Cette solution rend également le distributeur indépendant des sources de courant externes, ce qui permet de rendre le distributeur mobile.

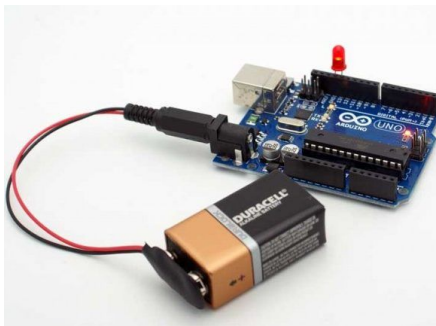


Image montrant le branchement de la pile à la carte Arduino



Exemple de l'alimentation depuis une prise avec un adaptateur

Nous tenons à préciser également que nous allons utiliser la carte Arduino UNO, présentée ci-dessus ou son équivalent avec un port alimentation externe afin de minimiser le risque de son endommagement par connexion manuelle.

Liens utilisés:

Alimentation externe de la carte Arduino:

<https://www.locoduino.org/spip.php?article16>
<https://zestedesavoir.com/tutoriels/364/alimenter-une-arduino-sans-usb/>
<https://letmeknow.fr/shop/fr/blog/33-tuto-rendre-le-arduino-autonome>
<https://navlab.fr/projets/outils/arduino/alimenter-votre-carte-arduino/>
<https://alexgyver.ru/lessons/arduino-power/>
<https://hobbytech.com.ua/%D1%8D%D0%BB%D0%B5%D0%BA%D1%82%D1%80%D0%BE%D0%BF%D0%B8%D1%82%D0%B0%D0%BD%D0%B8%D0%B5-arduino/>

Alimentation des servomoteurs:

<https://learn.adafruit.com/adafruit-16-channel-pwm-slash-servo-shield/shield-connections>

➔ Gérer les stocks

Afin de rendre la distribution des médicaments rigoureuse, il faut mettre en place un système de gestion des stocks. La machine doit, à chaque nouvelle distribution d'un paquet, mettre à jour les données en mémoire concernant le nombre de paquets restants dans chaque case. Et, si une case est vide, alerter l'utilisateur.

➤ Contrôler le nombre de médicaments restants dans une case

Pour contrôler le nombre des médicaments en stock, nous avons choisi d'optimiser le processus de gestion en positionnant le **compteur** dans le programme de la carte car c'est la solution la plus simple et qui ne nécessite aucun matériel.

➤ Vérifier et alerter l'utilisateur lorsque le stock est vide

Afin de garantir que les calculs de stock soient justes, nous mettons en place un **capteur de proximité** qui vérifiera en calculant la distance capteur-obstacle si le dernier paquet restant dans le stock est aussi le dernier élément du compteur programmé.



Capteur de proximité IR

Nous avons également pensé à utiliser un capteur laser, un capteur ultrason ou même une balance. Toutefois, il se trouve que les 2 premières options sont très volumineuses et, pour une case du distributeur portable, cela causerait un problème de taille et cela donnerait une “surcharge” d’une case en composants. De plus, puisque l’on a choisi d’utiliser un ressort, il serait impossible d’installer une balance. Finalement, le capteur de proximité est l’option la plus compacte, efficace et pratique.

Liens utilisés:

Fonctionnement du capteur:

<https://www.mataucarre.fr/index.php/2017/05/24/capteur-de-proximite-infra-rouge-fc-51-arduino/>

- Assurer la sauvegarde du stock

Enfin, pour assurer que lors de l’extinction et puis de l’allumage de la machine les données ne soient pas perdues, nous allons sauver les informations du stock sur la **mémoire ROM** de la carte.

- ➔ Assurer la communication entre l’utilisateur et la machine

L’objectif est que la machine distribue automatiquement des médicaments inscrits sur une ordonnance. Pour ce faire, nous avons étudié comment notre application pourrait récupérer les médicaments à fournir et la manière dont elle pourrait transmettre les informations à la carte Arduino pilotant la distribution.

- Encoder le nom des médicaments

Objectif : Trouver un moyen pour que l’application reconnaisse les médicaments à fournir

Solutions existantes pour : Lire l’ordonnance
--

Solution	+	-
Code à barres	<ul style="list-style-type: none"> - Coût faible - Facile d'utilisation - Facile à gérer - Lecture rapide et fiable - Correction des erreurs de saisie - Peut être imprimé sur différents supports 	<ul style="list-style-type: none"> - Durée de vie courte - Ne peut-être lu qu'au contact rapproché d'un lecteur - Stockage d'informations restreint
La radio identification (RFID passive ou active)	<ul style="list-style-type: none"> - Pas besoin d'un contact visuel - Possibilité de lire plusieurs étiquettes en même temps - Stockage d'informations bien plus important que pour le code à barres - Longue durée de vie de la puce électronique 	<ul style="list-style-type: none"> - Coût plus élevé - Difficile à générer - Technologies plus complexe à mettre en place - Besoin d'un lecteur physique particulier - Désavantages liés au champ électromagnétique
QR Code	<ul style="list-style-type: none"> - Coût faible - Facile d'utilisation - Facile à générer - Lecture rapide et fiable - Possibilité de stocker un grand nombre d'informations 	<ul style="list-style-type: none"> - Nécessite de passer par une application extérieur pour accéder au contenu (ex API) , il redirige - Nécessite une application pour la lecture - Ne peut être utilisé que sur support mobile
Solution retenue : QR Code Nous avons besoin d'un code facile à générer et à traiter		

Après comparaison, la meilleure solution pour identifier les médicaments à distribuer est le QR code. C'est un code facile à générer et les données seront aisément traitable depuis un lecteur intégré à notre application. Afin de simplifier le traitement nous identifierons les médicaments par des caractères (des chiffres ou des lettres) sous forme textuelle car il est pratique de gérer des chaînes de caractère avec une carte Arduino.



Liens utilisés:

Lecteur code à barres:

<https://www.gotronic.fr/art-lecteur-de-code-barres-dfr0314-22399.htm>

<https://arduino103.blogspot.com/2016/04/scanner-barcode-usb-abordable-truc-et.html>

RFID:

<https://www.seeedstudio.com/blog/2019/11/04/rfid-how-does-it-work-and-how-to-use-it-with-the-arduino/>

QR code:

<https://circuitdigest.com/microcontroller-projects/arduino-qr-code-generator>

➤ Transmettre l'ordonnance à la carte Arduino

Objectif : Faire en sorte que la carte sache quels médicaments faire tomber		
Solutions existantes pour : Transmettre l'ordonnance		
Solution	+	-
Module Bluetooth HC-05 (et non le HC-06 car il ne peut qu'être esclave)	<ul style="list-style-type: none">- Coût faible- Connexion sans fil- Mise en oeuvre facile- Peut être émetteur et récepteur (donc communication entre carte et application et entre application et carte)- Transmission qui n'est pas déviée	<ul style="list-style-type: none">- Communication qui peut être polluée par la présence d'autres appareils connectés en Bluetooth- Moyen de communication énergivore qui déchargera rapidement la batterie du

	<ul style="list-style-type: none"> - par les obstacles - Portée d'environ 10 m - Possibilité d'envoyer un nombre important de données 	smartphone
Module radio fréquence (par exemple Xbee)	<ul style="list-style-type: none"> - Coût faible - Connexion sans fil - Mise en oeuvre facile - Peut être émetteur et récepteur - Portée de 30m - Fiabilité - Ondes qui ne sont pas déviées par les obstacles - Pas très énergivore - Existence de bibliothèques Arduino pour les piloter (ex SoftwareSerial) - Possibilité d'envoyer un grand nombre d'informations 	<ul style="list-style-type: none"> - Difficile à combiner avec une application - Besoin d'une deuxième carte
Module WIFI	<ul style="list-style-type: none"> - Coût faible - Connexion sans fil - Mise en oeuvre facile - Peut être émetteur et récepteur - Possibilité d'envoyer un nombre important de données - Utilisation pratique avec un 	<ul style="list-style-type: none"> - Signal pouvant être atténué par la distance ou les obstacles - Nécessite une connexion WIFI - Performances qui peuvent être compromises par la qualité du réseau WIFI - Mauvaise sécurisation des données

	smartphone	transmises
Capteur lumineux (Infrarouge)	<ul style="list-style-type: none">- Coût faible- Mise en oeuvre facile- Connexion sans fil	<ul style="list-style-type: none">- Faisceau facilement dévié- Portée faible- Difficultés pour la récupération de l'information émise- Communication qui ne fonctionne que dans le sens de la carte vers l'application- Quantité de données pouvant être envoyées faible-
Solution retenue : Module Bluetooth HC-05 Car un seul module sera suffisant pour la communication, la liaison Bluetooth est pratique puisqu'un des supports de communication sera un smartphone		

Le moyen de transmission que nous avons décidé de retenir est le module bluetooth HC-05. Aisément connectable à distance à l'application, il permettra une transmission fluide d'un grand nombre de données (donc éventuellement toute une ordonnance) même s'il faut prendre en compte le fait que cette dernière pourrait être compromise par la présence d'autres objets connectés autour.



Module Bluetooth HC-05

Liens utilisés:

Module Bluetooth:

<https://components101.com/wireless/hc-05-bluetooth-module> --module HC05

<https://www.gotronic.fr/pj2-guide-de-mise-en-marche-du-module-bluetooth-hc-1546.pdf>

<https://www.gme.cz/data/attachments/dsh.772-148.1.pdf>
<https://create.arduino.cc/projecthub/electropeak/getting-started-with-hc-05-bluetooth-module-arduino-e0ca81>
<https://maker.pro/arduino/tutorial/bluetooth-basics-how-to-control-led-using-smartphone-arduino> -- contrôler la carte avec un smartphone
<https://pecquery.wixsite.com/arduino-passion/copie-de-le-detecteur-a-ultrasons-h-1>
--module HC06

RF module::

<https://electropeak.com/learn/wireless-communication-w-arduino-and-nrf24l01/>
https://en.wikipedia.org/wiki/RF_module
<https://iotdesignpro.com/projects/rf-communication-between-arduino-using-rf-modules>
<https://www.arduino.cc/en/Guide/ArduinoWirelessShieldS2> -- XBee

Module WiFi:

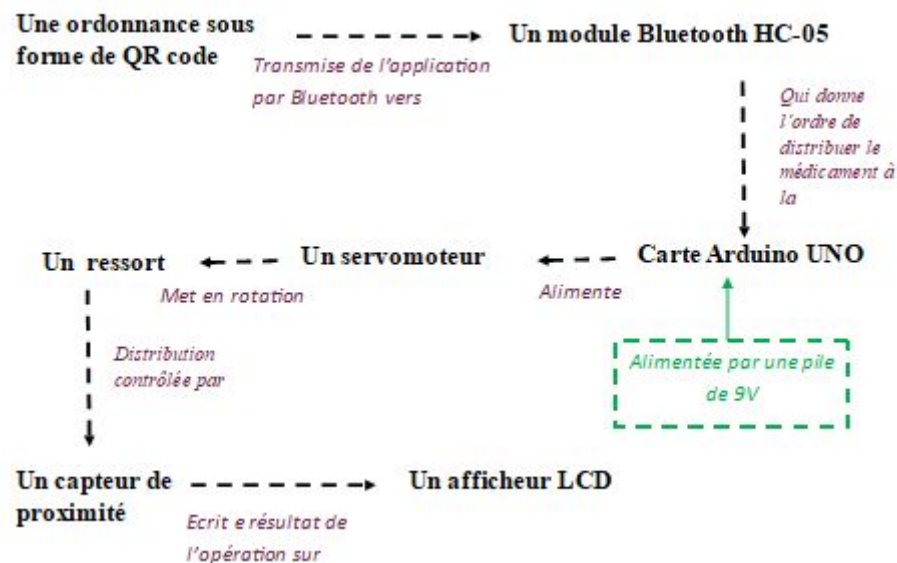
<https://www.arduino.cc/en/Guide/ArduinoUnoWiFi>
<https://create.arduino.cc/projecthub/imjeffparedes/add-wifi-to-arduino-uno-663b9e>
<https://www.instructables.com/Add-WiFi-to-Arduino-UNO/>

Capteur IR:

<https://www.instructables.com/Using-Infrared-Sensor-With-Arduino/>

● Conclusion :

Après avoir comparé les différentes solutions qui s'offraient à nous, nous sommes parvenus à la composition suivante pour notre système



● Récapitulatif

Matériel nécessaire au projet:

- Carte arduino uno munie d'un port pour brancher l'alimentation
- Module Bluetooth HC-05
- 4 ressorts ou un fil de fer
- 4 servomoteurs à rotation continue
- 1 capteur de proximité
- 1 capteur de ligne
- 1 pile de 9V et 500 mA
- plaques de contreplaqué
- support de piles
- afficheur LCD
- potentiomètre 5kOhm