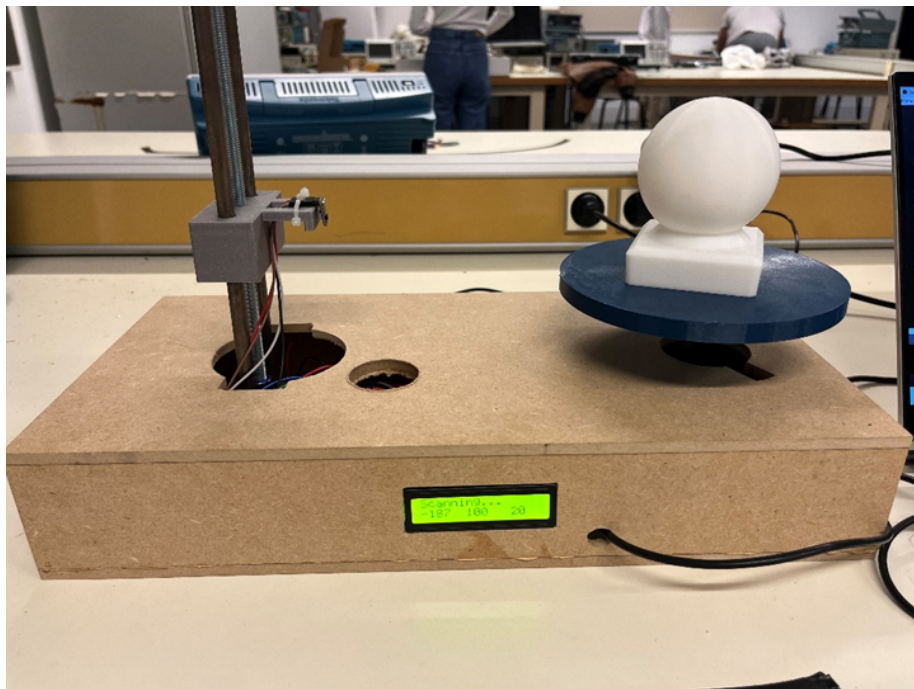


Rapport Projet Arduino

Scanner 3D

MEDINA Anne Marie G4

EL HASNAOUI Wiam G4



Sommaire

1. Introduction.....	
1.1. Objectifs.....	2
1.2. Fonctionnement général.....	3
2. Conception.....	
2.1. Modélisation.....	3
2.2. Fabrication et assemblage.....	4
3. Programmation et montage.....	
3.1. Matériel et montage.....	5
3.2. Programme Arduino.....	5
3.3. Programme Processing.....	6
4. Déroulement du projet.....	
4.1. Plannings.....	7
4.2. Problèmes rencontrés.....	7
5. Conclusion.....	
5.1. Résultat final.....	8
5.2. Améliorations possibles.....	8
5.3. Bibliographie.....	9

Introduction

Les scanners 3D sont utilisés dans de nombreux secteurs différents. Conception, contrôle, simulation... font partie des utilisations qui emploient cette technologie dans le secteur scientifique ou bien même créatif.

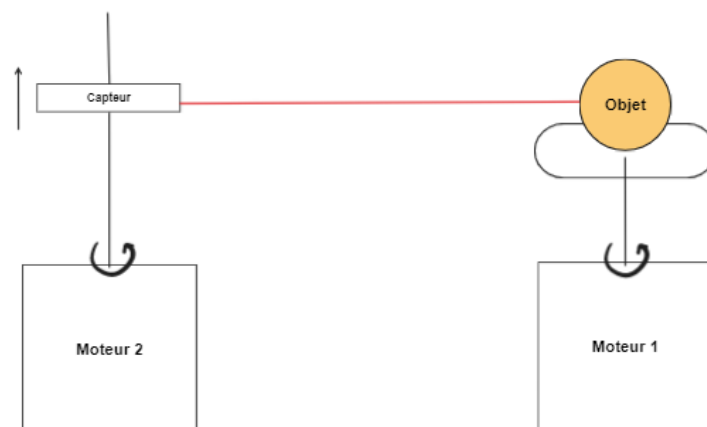
Nous souhaitons, lors de la mise en place de ce projet, nous renseigner et apprendre sur ce sujet pour finalement proposer un scanner 3D à la fois performant et à moindre coût.

1.1. Objectifs

Le scanner 3D réalisé devait remplir deux critères principaux: l'obtention d'un nuage de points représentant la modélisation de l'objet scanné et la modélisation affichée sur PC lors de la prise de mesures. De plus, les coordonnées des points sont générées sur un fichier de format 3D afin de pouvoir travailler dessus ultérieurement.

1.2. Fonctionnement général

L'objet scanné est situé sur une plaque qui effectue plusieurs rotations tandis que le capteur de distance prend la distance entre l'objet et lui-même. Le deuxième moteur lui permet donc de s'élever et ainsi de scanner sur toute la longueur de l'objet. Chaque mesure est associée à un point permettant d'obtenir un nuage de points pour la modélisation.

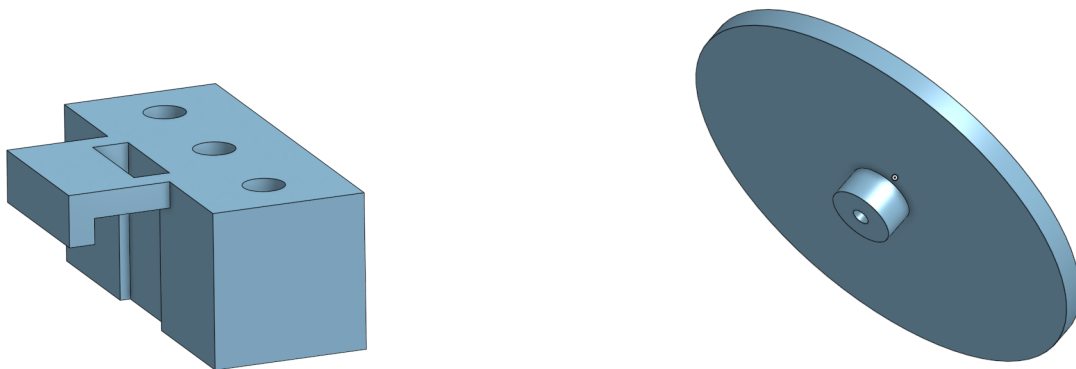


Les valeurs des coordonnées de ces points sont affichées sur un écran LCD nous permettant d'obtenir la valeur numérique et la représentation du point. Un bouton Start permet de commencer le scan de l'objet et un bouton Stop de l'arrêter. Et un appui sur la touche S du clavier permet de générer le fichier.

Conception

2.1. Modélisation

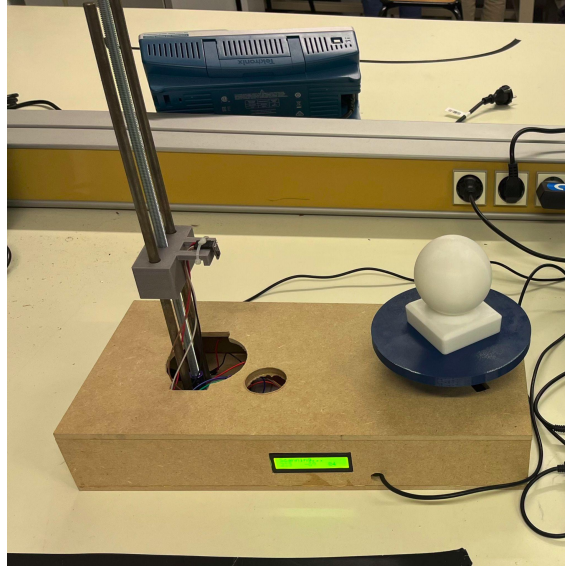
La structure de notre projet a été inspirée de modèles déjà existants. Nous avons adapté la forme des pièces en fonction des dimensions des composants que nous avons utilisés (moteurs, capteur de distance...). Après avoir schématisé le tout sur papier nous nous sommes servis du logiciel OnShape pour modéliser ces pièces. Voici quelques exemples:



Après cela, il a suffi de les imprimer à l'aide des imprimantes 3D mises à notre disposition.

2.2. Fabrication et assemblage

Concernant l'assemblage, il a fallu fixer les supports des moteurs sur une planche en bois sur laquelle on a percé les trous adéquats à l'aide du logiciel Inkscape. On a utilisé des tiges de 8mm de diamètre pour relier le tout ainsi qu'une vis sans fin pour permettre au support du capteur de distance de se déplacer de haut en bas. Après cela on a construit une boîte en bois permettant de contenir toute la structure afin de la rendre plus facilement transportable tout en protégeant les composants. Enfin, on a fixé les composants sur la planche reliant les supports des moteurs, l'écran LCD sur une face de la boîte et on a percé un trou facilitant l'accès aux boutons marche/arrêt.

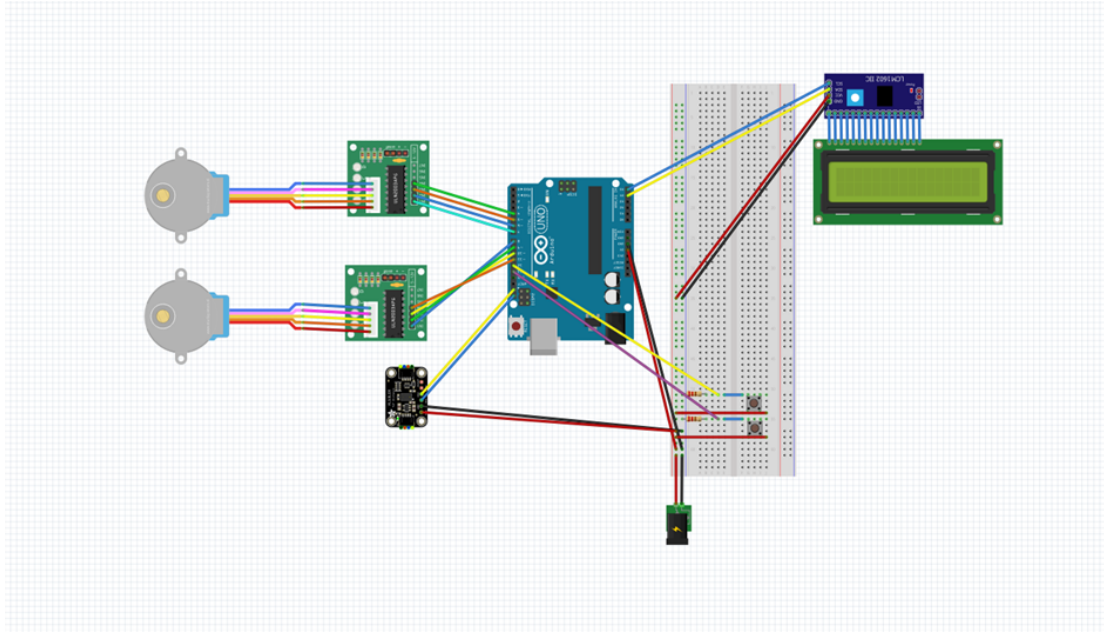


Programmation et montage

3.1 Matériel et montage

La réalisation de notre scanner 3D a principalement nécessité un capteur de distance haute précision et deux moteurs pas à pas.

Nous avons utilisé le capteur VL53L0X et deux moteurs 28-BYJ-48 accompagnés de modules UNL2003A. Ci-dessous le montage.



Un écran LCD I2C 16x2 et deux boutons ont été ajoutés.

3.2 Programme Arduino

La librairie Seeed nous permet d'utiliser le mode single et le high accuracy ranging et d'initialiser facilement le capteur VL53L0X. Afin d'éviter toutes mauvaises mesures, le capteur de distance affiche la valeur de la mesure prise seulement si elle est inférieure à 35 cm. Dans le cas contraire, elle prend cette valeur pour obtenir la distance du point telle que la distance $r = \text{distance entre le capteur et le centre de la plaque} - \text{distance mesurée par le capteur}$. A partir de cela, on obtient les coordonnées x , y et z :

$$x = r * \sin(\text{angle de rotation de la plaque})$$

$$y = r * \cos(\text{angle de rotation de la plaque})$$

$$z = z$$

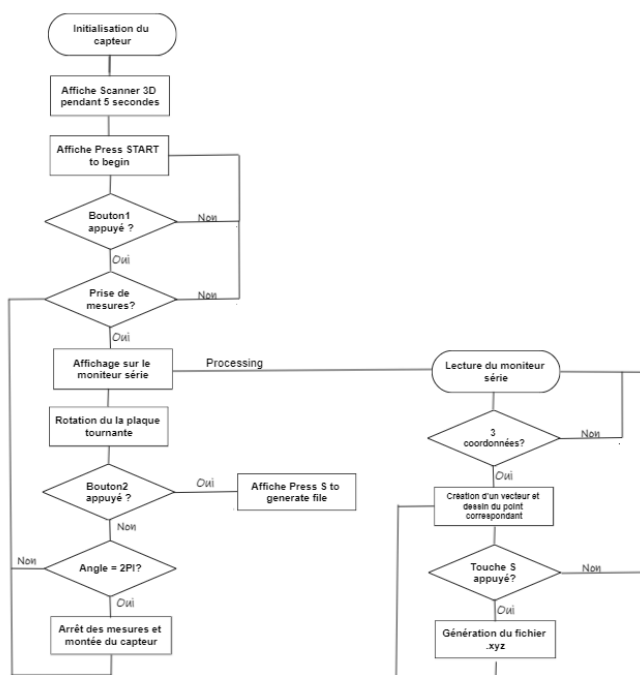
Ces trois valeurs sont affichées sur le moniteur série à chaque prise de mesure et le moteur tourne d'un angle de $\pi/64$ dès que le moniteur affiche une valeur afin d'éviter toute erreur si la distance mesurée n'est pas bonne. Lorsque le moteur a complété une rotation entière, les mesures s'arrêtent pour mettre en route le deuxième moteur permettant d'élever le capteur de distance.

L'écran LCD et les boutons s'ajoutent à ce programme. On définit deux variables permettant de savoir si le programme reste en marche pour les boutons marche/stop.

L'écran LCD affiche "Press START to begin" tant que la variable du bouton marche n'est pas vraie et il affiche « Scanning... x y z » ou « Out of range » si la valeur dépasse 35 cm (la valeur n'est pas utilisée dans ce cas). Toutes les demi-rotations de la plaque, l'écran affiche « Press STOP to end » et il affiche « Please wait ... » lorsque le deuxième moteur est en marche. Enfin, il affiche « Press S to generate file », ce qui appelle une méthode sur le programme Processing.

3.3 Programme Processing

Processing affiche les points selon les valeurs sur la fenêtre créée où sont dessinés trois axes pour une meilleure visualisation. Processing lit les valeurs des coordonnées à l'aide de la librairie Serial et dessine le point associé au cours de la lecture du moniteur série. Elle appelle la méthode saveToFile() dès que la touche S du clavier est appuyée.



Déroulement du projet

4.1. Planning

Au niveau du planning, la séparation des tâches s'est avérée être très différente de ce que l'on avait prévu au départ. On a sous-estimé la durée de certaines tâches donc il a fallu se réorganiser d'une

autre manière. Au final, les objectifs fixés ont tout de même été respectés et on a également pu ajouter des options qui n'étaient pas prévues au départ (boutons marche/arrêt, écran LCD).



4.2. Problèmes rencontrés

Les principaux problèmes rencontrés étaient liés aux programmes du capteur seul mais aussi du capteur synchronisé avec les deux moteurs. En revoyant minutieusement les programmes concernés on a pu régler ces problèmes.

Hormis cela, il y a également eu des soucis de mesure des pièces qu'il a fallu dessiner à nouveau ou encore bricoler, mais il n'y a pas eu de problèmes majeurs qui nous auraient empêchées de poursuivre notre projet.

Conclusion

5.1. Résultat final

Finalement nous avons réalisé un scanner 3D qui fonctionne tout en respectant notre cahier des charges. On obtient une modélisation utilisable après le scan complet. Cependant cette modélisation manque de précision, ce sont des points que nous aborderons dans la partie qui suit.

Pour ce qui est du coût matériel, nous avons calculé que le coût total, si l'on devait nous fournir entièrement sans accès aux ressources mises à disposition par l'école, serait situé entre 45 et 75 euros (Aliexpress et Bricoman), contre au minimum 400 euros sur le marché pour un petit scanner. Nous respectons donc bien notre objectif de fabriquer un scanner suffisamment performant et à moindre coût.

Quant au coût ingénieur, nous avons calculé que notre salaire combiné pour toutes les heures de travail passées sur ce projet reviendrait à 1140 euros brut.

5.2. Améliorations possibles

Un des premiers aspects que l'on peut aborder est l'esthétique du scanner. Les différentes contraintes telles que l'accès au montage principal et le câblage du capteur de distance à celui-ci n'améliorent pas l'esthétique.

Les principales améliorations qu'on pourrait apporter reposent sur la précision du scanner final. Tout d'abord la rotation de la plaque tournante peu stable peut être améliorée, et après cette amélioration, l'augmentation des points pour une rotation entière nous permet de gagner en précision.

De plus, tous les points de vue ne sont pas pris en compte lors du scan. En effet, le haut et le bas de l'objet sont négligés, donc un objet avec creux tel qu'un bol n'obtiendra pas une modélisation fidèle. Il faudrait alors modifier la structure globale pour permettre au capteur de prendre les mesures sous chaque angle.

Enfin, si ces différents cas sont pris en compte, on peut se concentrer sur la précision globale. En effet, on peut retrouver différentes manières d'approcher le scanner en dehors de l'approche directe que l'on a utilisée. Certains utilisent la photométrie afin d'augmenter la précision, cela peut représenter une piste dans la réalisation d'un autre scanner.

5.3. Bibliographie

- VL53L0X Datasheet, ST life.augmented, December 2022
- VL53L0X UM2039 User Manual, ST life.augmented, June 2016
- Arduino – Processing: serial data, video Programming for people, Youtube
- Projet Arduino Lidar Scanning & Java Rendering, Travis Ledo, Project Hub
- DIY Arduino based 3D scanning machine | IR 3D scanner, Mr Innovative, YouTube