

Annexes

Annexe 1 - Vidéo de promotion de l'atelier

Annexe 2 - Flyer de promotion de l'atelier

Annexe 3 - Manuel d'utilisation

Annexe 4 - photos des tests et des résultats finaux avec l'imprimante 3D

Annexe 5 - Attestation de réussite MOOC

Annexe 6 - Vidéo d'application du code de test d'un servomoteur

Annexe 7 - Vidéo d'application du code de fonctionnement de la main

Annexe 1

Ci-dessous, le lien vers la vidéo de promotion de l'atelier *Montage et programmation d'une main robotisée* :

<https://youtu.be/RVmfrDZ8-VA>

Annexe 2

≥ DU 22 AU

25 FÉVRIER;



Coordon- nées :



■ A propos de nous

association ROBOTS!

contact@association-robots.com

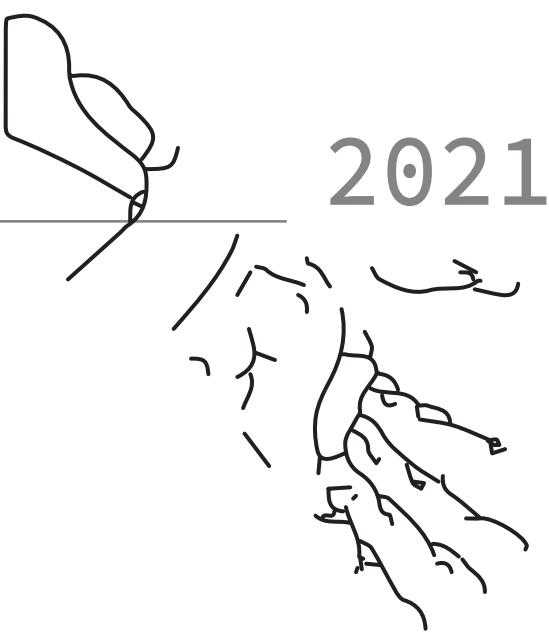
07 67 78 76 85

place du 51e régiment d'artillerie
44000, FRANCE

■ PERFOR- MANCE ROBOT- IQUES

L'association Robots! est déclarée d'intérêt général, elle a été fondée en 2014. Elle diffuse un savoir sur les robots et sur leur utilisation auprès du grand public. Elle mène aussi une réflexion sociétale sur comment, dès aujourd'hui, les robots peuvent améliorer notre quotidien.

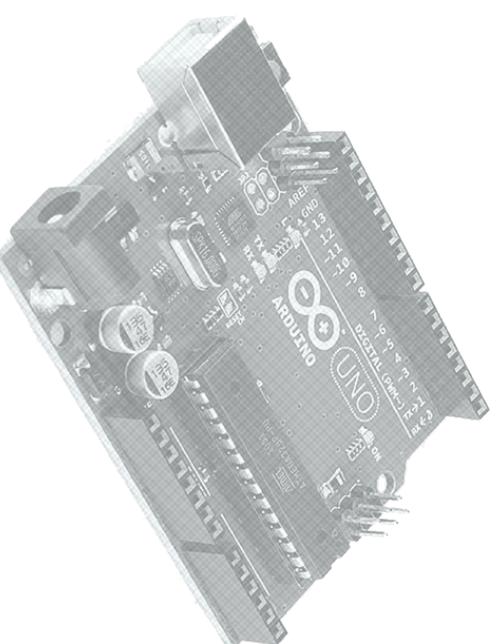
Nous mettons à la disposition des populations un savoir, des compétences et du matériel robotique. Nous désirons répondre aux nombreuses questions que génère la mutation technologique actuelle, accompagner, apaiser, informer sur les solutions robotiques existantes.



formations

Chaque vacances, nous vous proposons de venir apprendre à utiliser des robots. Qu'ils aient la forme d'humain, de chien ou de voiture, les robots s'immiscent peu à peu dans notre quotidien. C'est pourquoi il est important d'apprendre à les utiliser.

Nous vous donnerons les outils pour réaliser des performances à travers des robots. Aucun prérequis en informatique n'est nécessaire et les ateliers sont ouverts à tous à partir de 11 ans.



4 JOURS
16 HEURES

Inscription :

De 9H à 13H, venez découvrir comment fabriquer et utiliser une main robotisée.

Joris, votre animateur vous fera parcourir tous les outils pour y parvenir :

- Découverte de l'électronique (plateforme Arduino)
- Découverte de l'impression 3D
- Assemblage, câblage et utilisation de la main robotisée.

OUVERT À TOUTS LES PUBLICS À PARTIR DE 11 ANS.

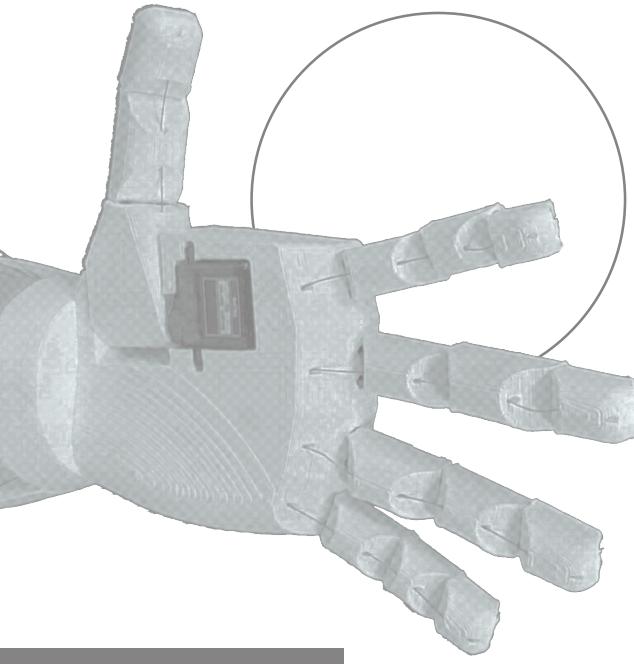
RENDEZ-VOUS SUR NOTRE SITE POUR LES INSCRIPTIONS.

<http://www.association-robots.com>



REPARTEZ AVEC VOTRE MAIN PRÊTE À L'EMPLOI ET LES COMPÉTENCES POUR PARTAGER L'EXPÉRIENCE

LES PLACES SONT LIMITÉES ALORS N'ATTENDEZ PAS !



Annexe 3

Manuel d'Utilisation : Montage et programmation d'une main robotisée

Albaladejo Joris

Association 1901 *Robots !*

Janvier 2021

-
Février 2021



Table des matières

1 Introduction à la CAO & à l'impression 3D	3
1.1 Qu'est-ce que la CAO ?	3
1.2 Qu'est-ce que l'impression 3D ?	3
1.3 Où trouver les ressources pour faire de l'impression 3D ?	3
1.3.1 Quels sites et/ou logiciels utiliser ?	3
1.3.2 Ce que l'on utilise pour cet atelier	4
1.4 Où réaliser ses propres impressions 3D ?	6
1.5 Utilisation de l'imprimante 3D	7
1.5.1 Notions de sécurité	7
1.5.2 Prise en main du logiciel <i>Ultimaker Cura</i>	8
2 Introduction à l'électronique	13
2.1 Définition de quelques termes	13
2.2 Ce que l'on utilise pour cet atelier	13
2.3 Fonctionnement des composants	14
2.4 Où acheter son matériel ?	17
3 Réalisation de la main	18
3.1 Montage de la main	18
3.2 Programmation du code	23
3.2.1 Quels outils utiliser ?	23
3.2.2 Test des servomoteurs	26
3.2.3 La main en elle même	33
4 Performance	34
4.1 Quelle utilité pour cette main robotique ?	34
4.1.1 Exprimer son humeur et/ou un mot	34
4.1.2 Dire un mot ou une phrase en langue des signes	34
4.1.3 Jouer à " <i>Pierre, Papier, Ciseaux</i> "	34
5 Pour aller plus loin	35
5.1 Comment créer son propre modèle en CAO ?	35
5.2 Comment choisir ses composants électroniques ?	35
5.3 Perspectives d'amélioration	35

Table des figures

1.1	Sites de modèles 3D	3
1.2	Projet de TechMartian	5
1.3	Notre main	5
1.4	Carte des FabLabs	6
1.5	Installation Cura - Étape 1	8
1.6	Étape 2	8
1.7	Étape 3	9
1.8	Configuration Cura - Étape 1	9
1.9	Étape 2	10
1.10	Étape 3	10
1.11	Étape 4	11
1.12	Étape 5	11
1.13	Étape 6	11
1.14	Étape 7	12
2.1	Matériel électronique	13
2.2	Platine d'essai	14
2.3	Carte Arduino	14
2.4	Potentiomètre	16
2.5	Servomoteur SG-90	16
2.6	Pile	17
3.1	Liste du matériel nécessaire au montage	18
3.2	Montage de la main - Étape 1	18
3.3	Étape 2	19
3.4	Étape 3	20
3.5	Étape 4	20
3.6	Étape 5	21
3.7	Étape 6	22
3.8	Configuration Tinkercad - Étape 1	23
3.9	Étape 2	23
3.10	Étape 3	24
3.11	Étape 4	24
3.12	Étape 5	24
3.13	Étape 6	25
3.14	Installation Arduino IDE - Étape 1	25
3.15	Étape 2	26
3.16	Test servomoteur - Étape 1	26
3.17	Étape 2	27
3.18	Étape 3	27
3.19	Étape 4	28
3.20	Étape 5	28
3.21	Étape 6	29
3.22	Étape 7	29
3.23	Étape 8	29
3.24	Étape 9	30
3.25	Étape 10	30
3.26	Étape 11	31
3.27	Étape 12	32
3.28	Programmation de la main - Étape 1	33

1 Introduction à la CAO & à l'impression 3D

1.1 Qu'est-ce que la CAO ?

La CAO, ou Conception Assistée par Ordinateur (Computer Aided Design en anglais) est l'ensemble des logiciels et des techniques de modélisation géométriques permettant de simuler et de visualiser la structure et le fonctionnement d'un objet en prévision de sa fabrication. De nos jours elle est couramment utilisée dans l'industrie.

Il existe de nombreux logiciels de CAO, certains adaptés à l'électronique, d'autres à la mécanique, à l'industrie automobile, etc..

Ici, un lien regroupant une liste de logiciels de CAO gratuits.



Astuce : Certains logiciels de CAO payants proposent une licence gratuite ou à bas coût pour les étudiants. (Par exemple SolidWorks développé par la société Dassault ou encore la suite de logiciels Autodesk)

1.2 Qu'est-ce que l'impression 3D ?

L'impression 3D est une technique permettant de fabriquer de manière automatisée et reproductive des objets en trois dimensions.

Elle permet de réaliser une pièce par superpositions successives de fines couches de matière (ALM : Additive Layer Manufacturing en anglais).

Il existe différents types d'imprimantes, utilisant des techniques de dépôt de matière spécifiques. De plus, il y a également plusieurs types de matériaux pour faire de l'impression 3D.



Attention : Il faut se renseigner sur le (ou les) matériau(x) supporté(s) par l'imprimante que l'on va utiliser. En effet chaque imprimante a ses spécificités.

1.3 Où trouver les ressources pour faire de l'impression 3D ?

1.3.1 Quels sites et/ou logiciels utiliser ?

Il existe de nombreux sites internet où l'on peut trouver des fichiers 3D en open source déposés et conçus par d'autres utilisateurs. Ci-dessous, quelques exemples et les liens vers ces sites.

GRAB**CAD**

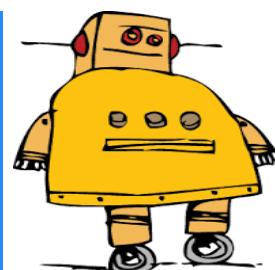


FIGURE 1.1 – Logos de quelques sites de modèles 3D

Site	Lien
Grabcad	https://grabcad.com/library
Cults3D	https://grabcad.com/library
Thingiverse	https://www.thingiverse.com/
Instructables	https://www.instructables.com/workshop/3d-printing/projects/

Ici, le lien d'un article répertoriant plus de plateformes pour trouver des fichiers 3D.



Attention : Certains de ces sites nécessitent la création d'un compte pour pouvoir télécharger les modèles, et certains modèles peuvent être payants.

Trouver son modèle c'est bien... L'imprimer c'est mieux ! **Mais comment faire pour imprimer son fichier 3D ?**

Pour pouvoir effectuer une impression 3D il faut passer via un logiciel dit "Trancheur 3D" (ou Slicer 3D en anglais).

Mais qu'est-ce qu'un trancheur 3D ?

Un trancheur permet de calculer les couches qui seront imprimées ce qui revient à découper l'objet en intervalles de hauteur réguliers. De nombreux paramètres peuvent être ajustés afin d'adapter l'impression à l'imprimante 3D et à la qualité souhaitée de l'objet. Ils sont généralement identiques d'un logiciel à l'autre, nous les aborderons donc lors de la prise en main d'*Ultimaker Cura*.

Ici, le lien d'un article regroupant les tranneurs 3D utilisés le plus couramment.



Attention : Certains de ces logiciels possèdent différentes versions. Certaines gratuites, d'autres payantes.

1.3.2 Ce que l'on utilise pour cet atelier



Au cours de cet atelier, nous allons uniquement utiliser le logiciel *Ultimaker Cura* pour visualiser puis imprimer les différentes pièces de la main.

La source des fichiers utilisés pour imprimer la main :

Toutes les pièces 3D que nous utilisons viennent du projet de TechMartian (cliquer sur le pseudo pour accéder à la page de son projet et aux fichiers d'impression 3D) et sont également disponibles sur le site de *Robots!*. Ci-dessous, une photographie du modèle final qu'il a réalisé :

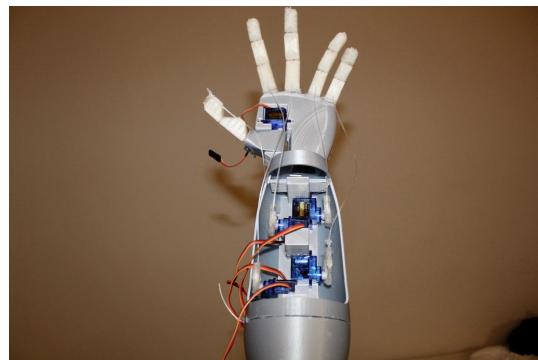


FIGURE 1.2 – Main robotique réalisée par TechMartian

Au cours de l'atelier nous allons pas à pas monter et programmer notre modèle. Ci-dessous, une photo du modèle auquel nous arriverons à la fin de l'atelier :



FIGURE 1.3 – Main robotique réalisée durant l'atelier

1.4 Où réaliser ses propres impressions 3D ?

Vous n'avez pas d'imprimante 3D à la maison ? Pas de problème !

À Nantes :

- Contacter la société *Dulse*, membre du collectif *Ici Nantes* pour réaliser vos impressions. Le collectif propose également des formations certifiantes en impression 3D. (cliquer sur Ici Nantes pour avoir accès à leur catalogue de formations)

Note : *Ces formations sont payantes mais sont éligibles au CPF (Compte Personnel de Formation).*

- Se rendre au FabLab de l'association *PiNG*.



Attention : Avant de pouvoir réaliser vos propres impressions sur les machines du FabLab, il vous faudra adhérer à PiNG. Renseignez vous sur leur site pour voir les offres proposées.

De manière un peu plus générale :

- Le pôle *EMC2* a développé le site *Meet3D*, qui recense les ressources en impression 3D.
- Il existe également une carte des FabLabs réalisée par le site *Makery.info*.

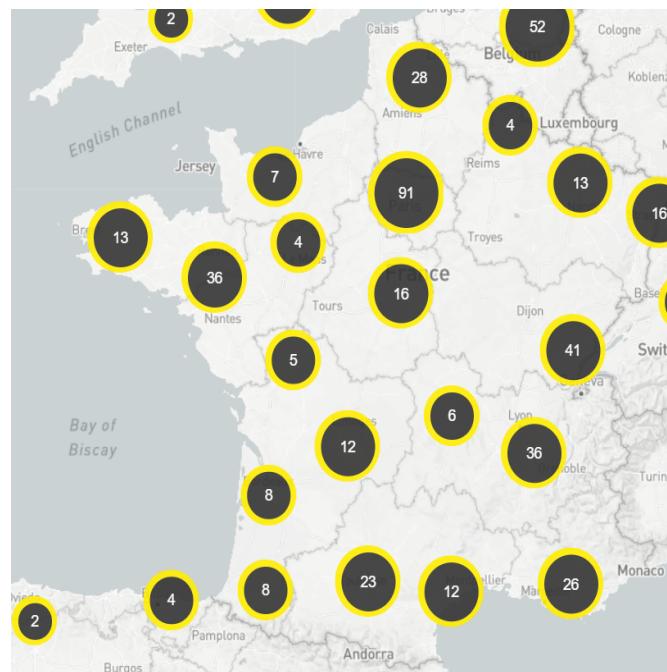


FIGURE 1.4 – Carte de France des FabLabs

La carte détaillée avec les adresses est disponible [ici](#).

1.5 Utilisation de l'imprimante 3D

1.5.1 Notions de sécurité



La buse et le plateau chauffent pendant l'utilisation. Attendre que l'imprimante ait refroidi avant d'essayer d'enlever les pièces ou les résidus de filaments.



L'imprimante présente des parties mobiles pouvant être dangereuses. Ne pas toucher la machine pendant son fonctionnement.



Ne pas laisser l'imprimante à la portée des enfants.



Il est recommandé d'utiliser des lunettes de protection lorsque l'on nettoie les pièces pour éviter que des projections entre en contact avec les yeux.



Ne pas rester trop proche de l'imprimante pendant l'impression, les vapeurs peuvent être dangereuses. Utiliser si possible l'imprimante dans une pièce aérée ou ventilée.



Ne pas mettre d'eau sur l'imprimante.

1.5.2 Prise en main du logiciel *Ultimaker Cura*

Installation du logiciel :

- Se rendre sur le site : <https://ultimaker.com/fr/software/ultimaker-cura>
- Aller dans la section *Logiciel* :

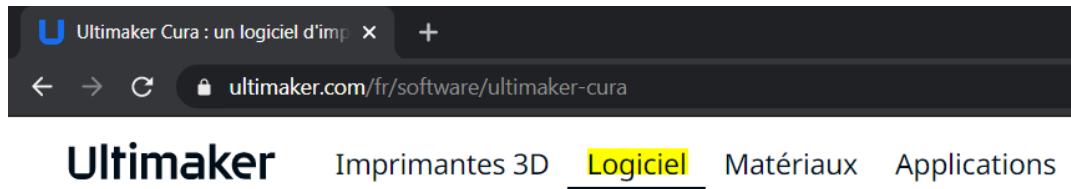


FIGURE 1.5 – Installation Cura - Étape 1

- Descendre jusqu'à *Ultimaker Cura* et cliquer sur *En savoir plus* :



FIGURE 1.6 – Installation Cura - Étape 2

- Cliquer sur *Télécharger gratuitement* :

Ultimaker Cura

Reconnu par des millions d'utilisateurs, Ultimaker Cura est le logiciel d'impression 3D le plus populaire au monde. Préparez des impressions en quelques clics, intégrez-les au logiciel de CAO pour simplifier votre flux de travail ou accédez aux paramètres personnalisés pour un contrôle avancé de l'impression et de la découpe.

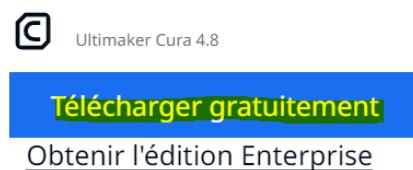


FIGURE 1.7 – Installation Cura - Étape 3

Configuration du logiciel :

- Cliquer sur la petite flèche :

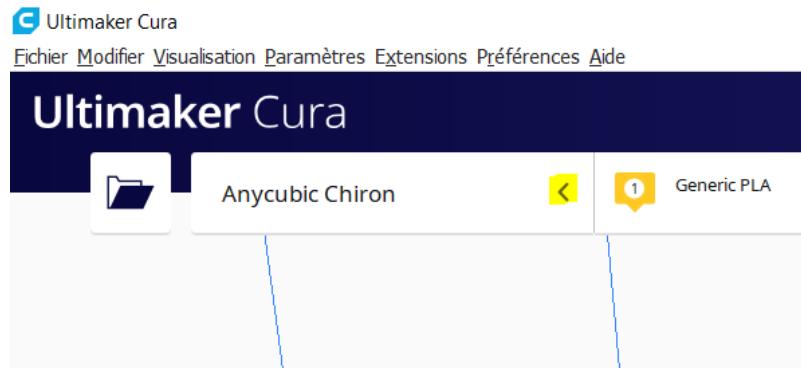


FIGURE 1.8 – Configuration Cura - Étape 1

- Cliquer sur *ajouter une imprimante* :

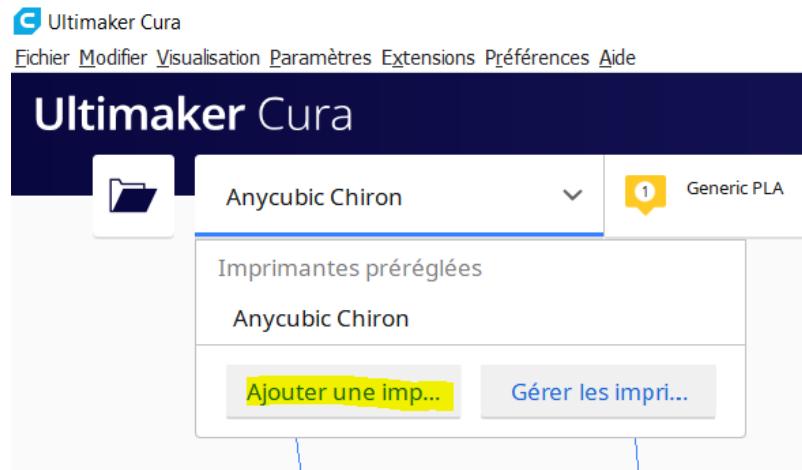


FIGURE 1.9 – Configuration Cura - Étape 2

- Choisir son imprimante en réseau si elle est disponible, ou la sélectionner manuellement en allant dans *hors réseau* sinon, puis l'ajouter :

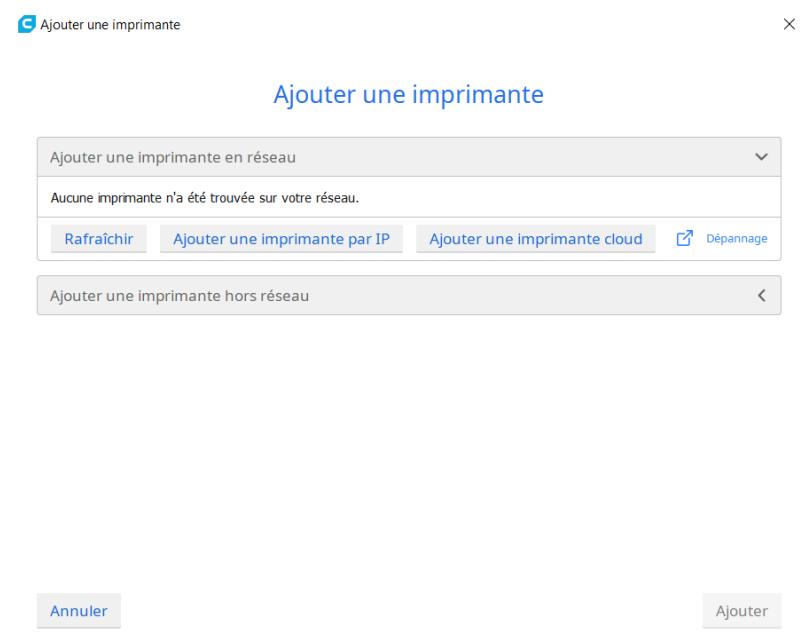


FIGURE 1.10 – Configuration Cura - Étape 3

- Cliquer sur la petite flèche pour choisir le type de filament que vous allez utiliser :



FIGURE 1.11 – Configuration Cura - Étape 4

- Cliquer sur le petit crayon pour régler les paramètres d'impression :

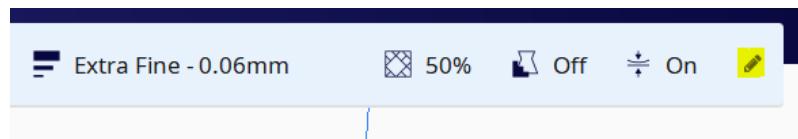


FIGURE 1.12 – Configuration Cura - Étape 5

- Modifier les paramètres d'impression comme suit :

Catégorie	Nom du paramètre	Valeur
Qualité	Hauteur de la couche	0.3 mm
Coque	Épaisseur de la paroi	0.5 mm
	Nombre de lignes de la paroi	1
	Épaisseur du dessus/dessous	0.8 mm
	Épaisseur du dessus	0.8 mm
	Couche supérieure	3
	Épaisseur du dessous	0.8 mm
	Couche inférieure	3
	Expansion horizontale	0 mm
Remplissage	Densité du remplissage	17.5%
	Motif de remplissage	Grille
Matériaux	Température d'impression	200°C
	Température du plateau	85°C
Vitesse	Vitesse d'impression	80mm/s
Déplacement	Activer la rétraction	Oui
Refroidissement	Activer le refroidissement de l'impression	Oui
	Vitesse du ventilateur	100%
Supports	Générer les supports	Non
Adhérence du plateau	Type d'adhérence du plateau	Bordure

FIGURE 1.13 – Configuration Cura - Étape 6



Attention : L'imprimante que nous avons utilisé pour le projet est une Chiron de chez Anycubic, et le type filament utilisé est le PLA. Les paramètres d'impression seront peut-être à régler différemment si vous utilisez une autre imprimante et un autre type de filament.

- Aller dans *fichier* puis *Ouvrir le(s) fichier(s)* et sélectionner celui que vous voulez imprimer :

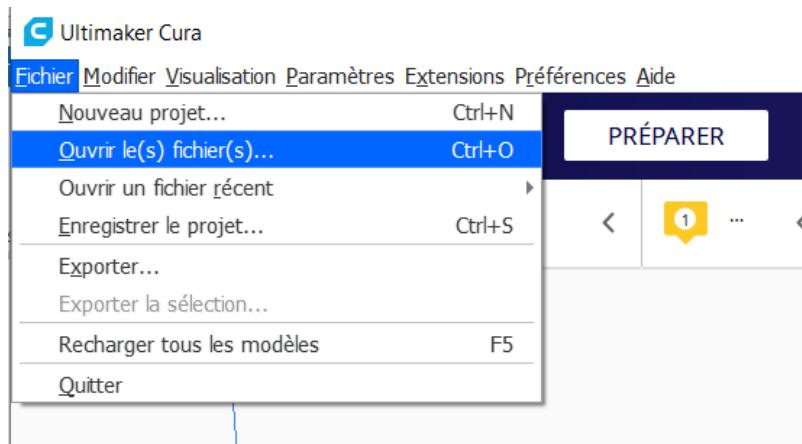


FIGURE 1.14 – Configuration Cura - Étape 7



Astuce : Si le plateau de votre imprimante est assez grand, il est possible d'imprimer plusieurs pièces simultanément. Pour cela il faut les positionner de part et d'autre du plateau virtuel sur Cura. Pour ce faire, cliquez sur la pièce que vous voulez déplacer et bouger la en restant appuyé sur la flèche correspondant à la direction souhaitée.



Astuce : La température de la pièce dans laquelle vous êtes influe également sur la qualité de votre impression.

2 Introduction à l'électronique

2.1 Définition de quelques termes

Avant toute chose, nous allons définir assez simplement quelques termes concernant le matériel que nous allons utiliser :

Servomoteur : Sa fonction principale est d'assurer la reproduction d'un mouvement en réponse à une commande externe. En l'occurrence la commande sera envoyée par nos potentiomètres linéaires.



Attention : *Les servomoteurs que nous utilisons ici sont des servomoteurs standards et ont une plage de mouvement entre 0° et 180°.*

Potentiomètre linéaire : Un potentiomètre linéaire est une résistance variable permettant d'ajuster la valeur de certains paramètres. En l'occurrence le paramètre ajusté sera la position angulaire du servomoteur correspondant.

Câbles Dupont mâle-mâle : Ces câbles permettent de créer une connexion sur une platine d'essai sans avoir à réaliser de soudure. Ils sont dits "mâles-mâles" car ils présentent de chaque côté une petite pointe venant s'insérer dans les ports de connexions.

Platine d'essai (Breadboard en anglais) : Dispositif permettant de réaliser le prototypage et le test d'un circuit électronique sans avoir à réaliser de soudure.

Entrée digitale : Également appelée entrée numérique. Ce sont des entrées ne pouvant prendre que 2 valeurs : 0 (le système est à l'arrêt) et 1 (le système est en fonctionnement).

Entrée analogique : Elle s'oppose à l'entrée digitale et permet de recueillir un signal électrique variable sur une plage de valeur définie au préalable.

2.2 Ce que l'on utilise pour cet atelier

Pour programmer le mouvement de cette main, nous allons utiliser le matériel suivant :

Type	Quantité/unité
Carte Arduino Uno	1
Câbles Dupont mâle-mâle	32
Potentiomètres linéaires 10kΩ	5
Servomoteurs 9g SG-90	5
Platine d'essai	1
Pile 9V	1
Support de pile (banchement sur port secteur)	1

FIGURE 2.1 – Liste du matériel électronique nécessaire

2.3 Fonctionnement des composants

La platine d'essai :

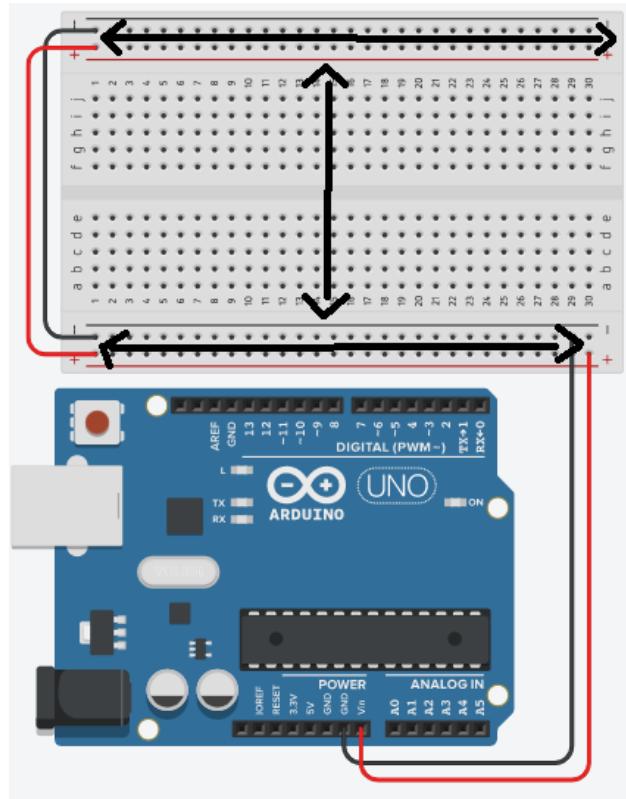


FIGURE 2.2 – Sens des connexions sur la platine d'essai

La carte Arduino :

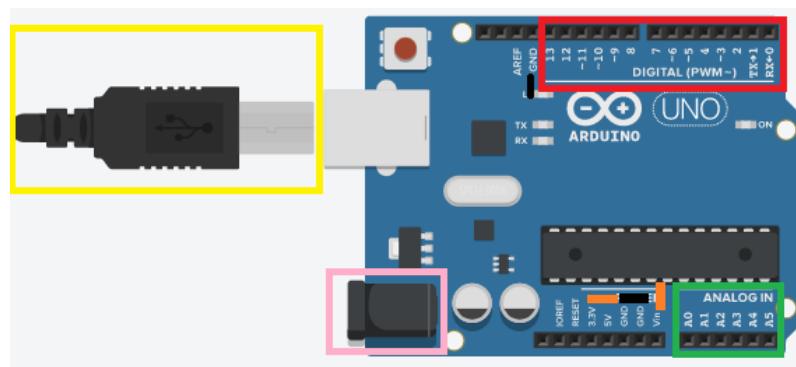


FIGURE 2.3 – Description des différents ports de la carte Arduino

Légende :

- **Traits orange** : Ce sont les ports de la carte permettant d'alimenter en tension la platine d'essai. Cette carte présente une alimentation en 3.3V, une en 5V et celle qui va nous intéresser étant donné que nous allons alimenter l'Arduino avec une pile 9V : V_{in} .



Attention : *La carte Arduino Uno peut supporter une tension d'alimentation externe entre 6V et 20V mais il est recommandé que cette tension soit comprise entre 7V et 12V.*

- Traits noir : Ce sont les ports de la carte permettant de mettre les différents composants branchés à la masse (= gnd sur la carte \Leftrightarrow ground en anglais). Cela correspond à une tension de 0V.



Astuce : *En pratique, lorsque l'on utilise une platine d'essai, on connecte un des ports de masse de l'Arduino à la ligne de masse de la platine d'essai avec un câble Dupont (Voir figure 15).*

- **Carré jaune** : C'est le port d'alimentation en USB de la carte Arduino. Il permet de téléverser le programme de l'ordinateur sur le microprocesseur de la carte, et également de l'alimenter si on le laisser branché une fois le programme téléversé.
- **Carré rose** : C'est le port d'alimentation sur secteur de l'Arduino. C'est celui qui va nous permettre d'alimenter la carte lors de notre projet.
- **Carré vert** : Ce sont les ports d'entrées analogiques de la carte. Ici ils vont nous servir à brancher les différents potentiomètres.
- **Carré rouge** : Ce sont les ports d'entrées digitales de la carte. Ici ils vont nous servir à brancher les différents servomoteurs.

Les potentiomètres :

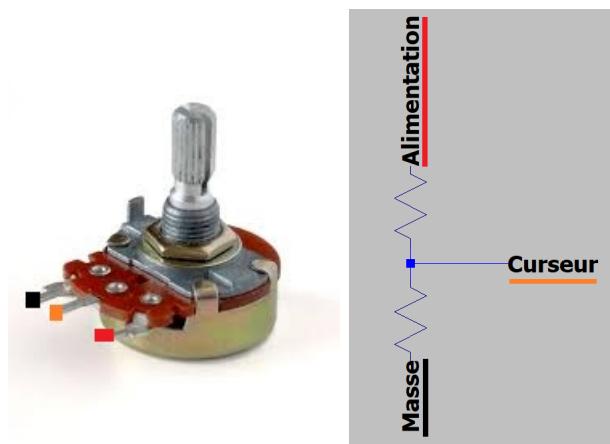


FIGURE 2.4 – Photo et schéma électronique d'un potentiomètre linéaire

Légende :

- Point et trait noir : Port relié à la masse de la platine d'essai.
- Point et trait rouge : Port relié à l'alimentation de la platine d'essai.
- Point et trait orange : Port relié au curseur du potentiomètre et à l'entrée analogique de l'Arduino, permettant de lire la valeur du potentiomètre.

Dans notre cas, faire bouger le curseur du potentiomètre va permettre d'envoyer une certaine valeur de tension au servomoteur en la convertissant en une position angulaire (entre 0° et 180°). Cette position sera alors lue par le servomoteur qui bougera jusqu'à atteindre la position demandée.

Le servomoteur SG-90 :



FIGURE 2.5 – Photo et schéma du servomoteur utilisé

Note : *PWM (= Pulse With Modulation en anglais) correspond à une entrée digitale sur l'Arduino.*

La pile :

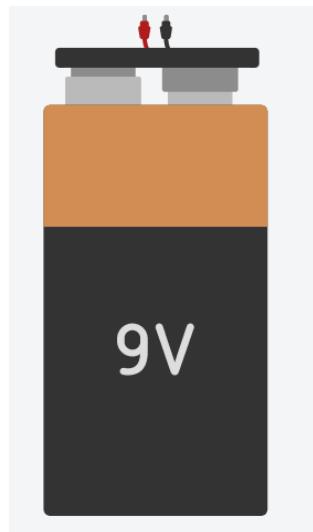


FIGURE 2.6 – Schéma de la pile 9V

Légende :

- Câble rouge : Câble d'alimentation 9V de la pile (borne +).
- Câble noir : Câble de masse de la pile (borne -).

2.4 Où acheter son matériel ?

Il est très facile de vous procurer le matériel. Il y a énormément de sites et/ou magasins spécialisés dans la vente de composants électroniques pour les particuliers. Un grand nombre de composants sont également disponibles sur des sites comme Amazon, Ebay, etc..

3 Réalisation de la main

3.1 Montage de la main

objet	quantité nécessaire
câble élastique pour les articulations	45cm/doigt
câble métal pour doigts	54cm/doigt
super glue ou pistolet à colle chaude	environ 1 tube
vis	14 (utiliser celles fournies avec les servomoteurs)

FIGURE 3.1 – Liste du matériel nécessaire au montage

- Positionner les différentes phalanges autour de la paume pour visualiser le résultat final :



FIGURE 3.2 – Montage de la main - Étape 1

- Passer le câble souple une 1ère fois dans les différentes articulations des doigts puis dans la paume, et faire un 2ème tour si nécessaire pour que le fil soit plus tendu :

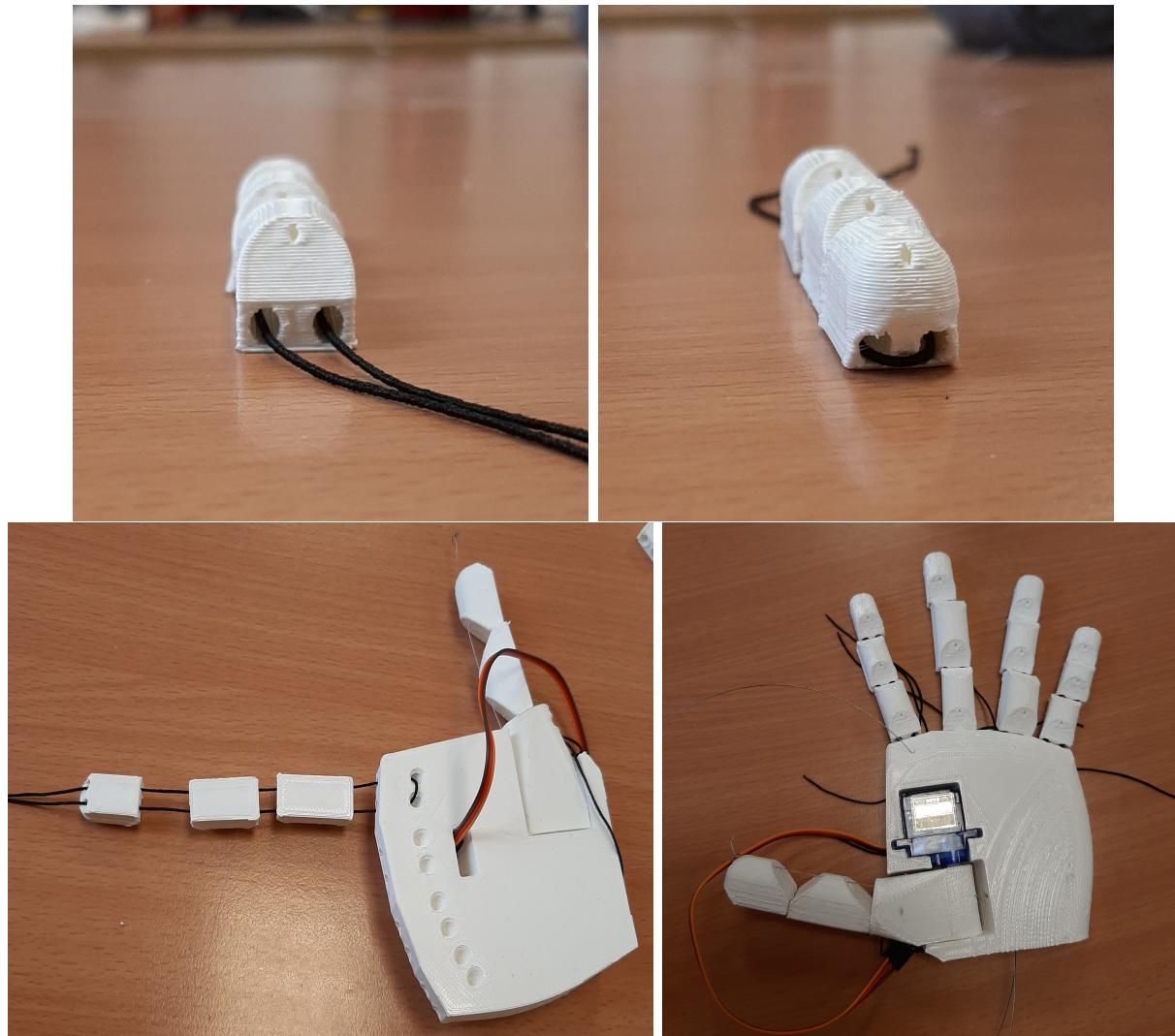


FIGURE 3.3 – Montage de la main - Étape 2

- Visser sur la poulie le petit palonnier en utilisant la petite vis à l'extremité et l'une des grandes au centre :



FIGURE 3.4 – Montage de la main - Étape 3



Attention : *Ne pas visser entièrement la grande vis.*

- Finir de fixer la grosse vis dans au centre de l'engrenage du servomoteur :



FIGURE 3.5 – Montage de la main - Étape 4



Attention : *Les servomoteurs que nous utilisons ont une course de 180°, il faut donc positionner correctement le palonnier avant de visser pour ne pas faire forcer le servomoteur lors du fonctionnement.*

- Fixer les servomoteurs sur le support. Pour cela utiliser la grande vis restante dans le sachet du servomoteur. Des trous sont présents sur le support :

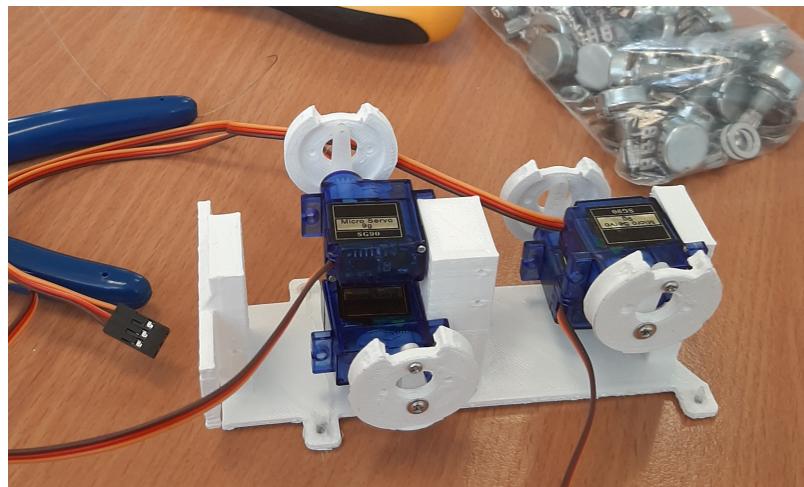


FIGURE 3.6 – Montage de la main - Étape 5

- Fixer le support à l'intérieur de l'avant bras, enrouler les câbles des doigts au niveau des poulies et câbler les potentiomètres et les servomoteurs sur votre breadboard et votre carte :

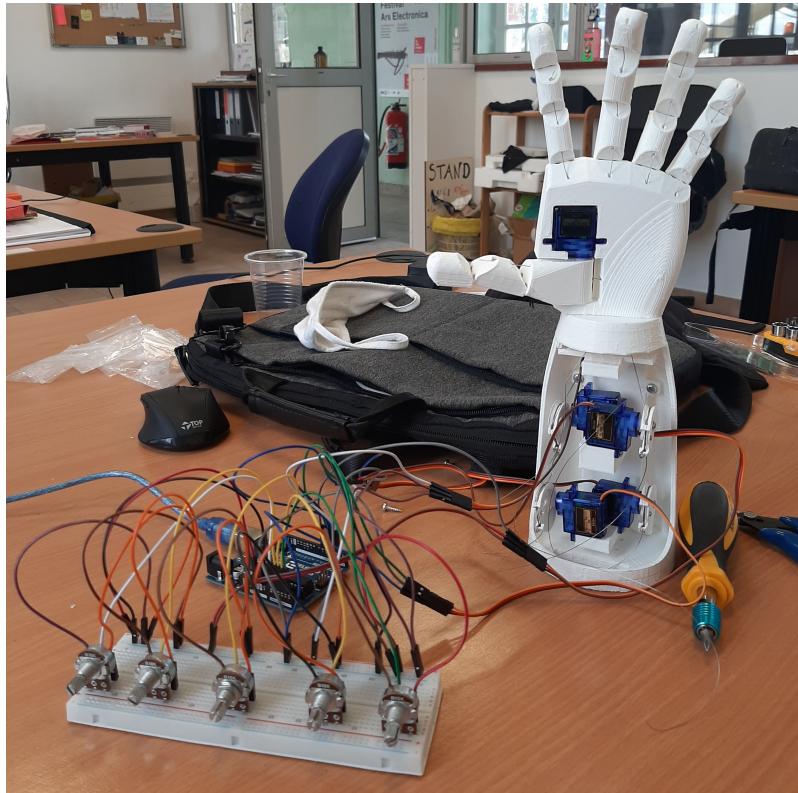


FIGURE 3.7 – Montage de la main - Étape 6

3.2 Programmation du code

3.2.1 Quels outils utiliser ?

Lorsque l'on ne souhaite pas s'initier à la programmation à proprement parler, il est possible d'utiliser Tinkercad pour créer ses programmes en passant par des algorithmes. Par ailleurs Tinkercad permet aussi de simuler vos montages pour être sûr qu'ils fonctionneront correctement lorsque vous les réaliserez avec votre carte.

- Se rendre sur le site : <https://www.tinkercad.com/> et cliquer sur *Rejoindre maintenant* :

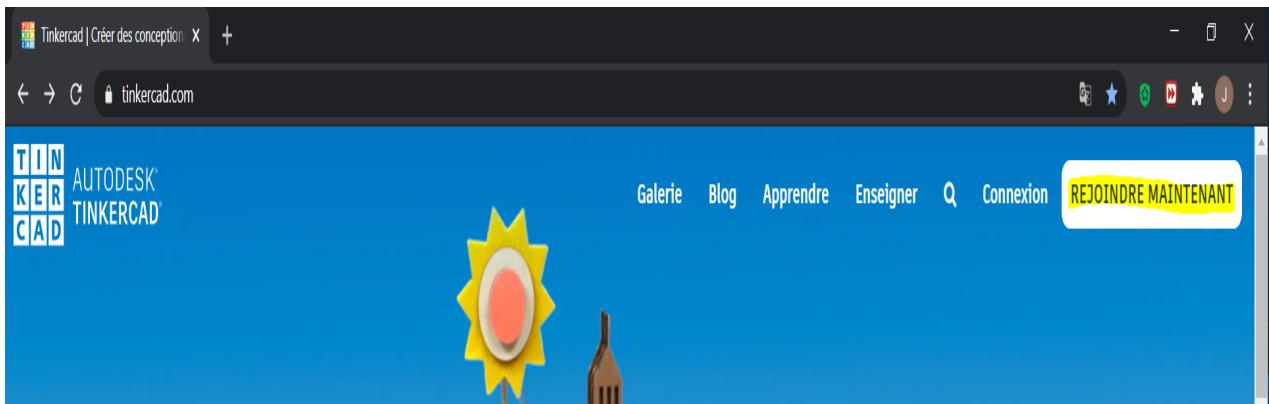


FIGURE 3.8 – Configuration Tinkercad - Étape 1

- Une fois votre compte créé, cliquer sur *Circuits* :

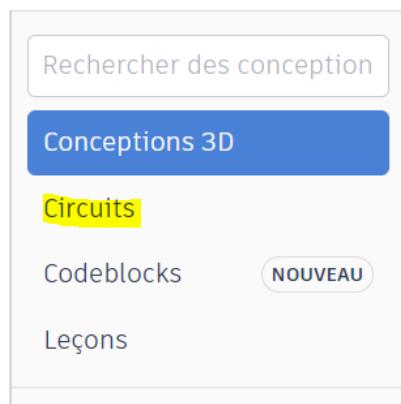


FIGURE 3.9 – Configuration Tinkercad - Étape 2

- Cliquer sur *Créer un circuit* :



FIGURE 3.10 – Configuration Tinkercad - Étape 3

- Cliquer sur la petite flèche :

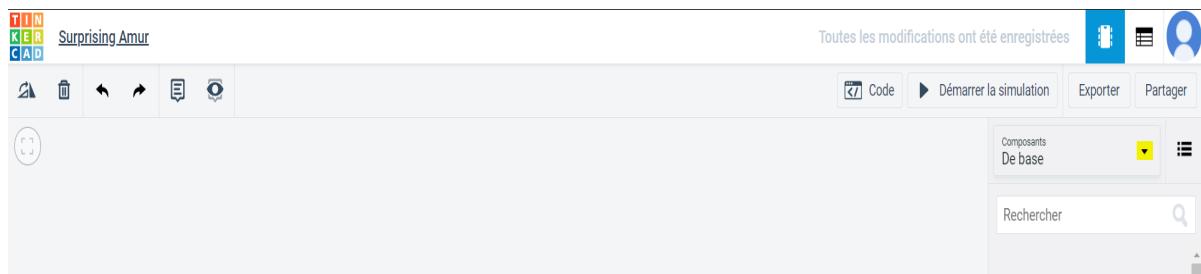


FIGURE 3.11 – Configuration Tinkercad - Étape 4

- Aller dans *Kits de démarrage*, sélectionner *Arduino* et choisir la *Platine d'essai* :

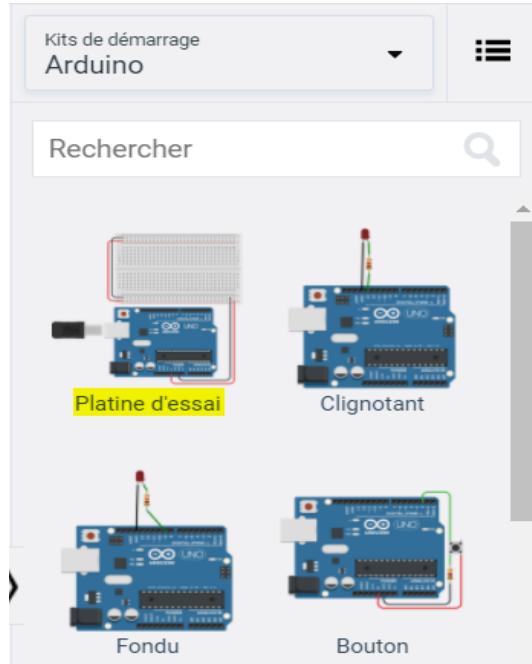


FIGURE 3.12 – Configuration Tinkercad - Étape 5

- Déposer la platine d'essai sur la surface de travail :

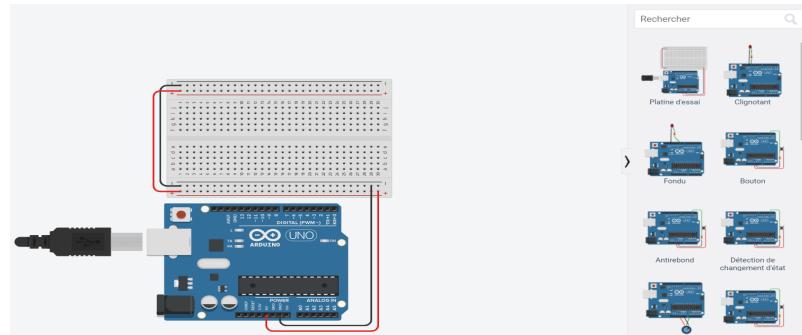


FIGURE 3.13 – Configuration Tinkercad - Étape 6

Pour programmer la main il est indispensable d'avoir le logiciel Arduino IDE installé sur son ordinateur. Pour ce faire :

- Se rendre sur le site : <https://www.arduino.cc/>
- Aller sur *SOFTWARE* et cliquer sur *DOWNLOADS*

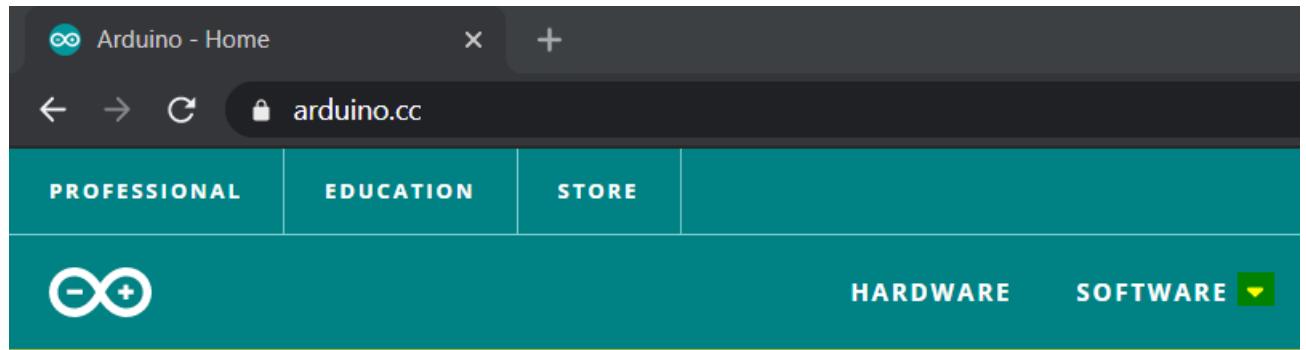


FIGURE 3.14 – Installation Arduino IDE - Étape 1

- Sélectionner la version correspondant à votre ordinateur :

Downloads

The screenshot shows the official Arduino website's download page. On the left, there's a large button for "Arduino IDE 1.8.13". Below it, a paragraph explains what the IDE is used for, followed by a link to the "Getting Started" page. A section for "SOURCE CODE" links to GitHub. On the right, a teal sidebar titled "DOWNLOAD OPTIONS" provides links for different operating systems:

- Windows** Win 7 and newer
- Windows** ZIP file
- Windows app** Win 8.1 or 10 [Get](#)
- Linux** 32 bits
- Linux** 64 bits
- Linux** ARM 32 bits
- Linux** ARM 64 bits
- Mac OS X** 10.10 or newer

At the bottom of the sidebar, there are links for "Release Notes" and "Checksums (sha512)".

FIGURE 3.15 – Installation Arduino IDE - Étape 2

3.2.2 Test des servomoteurs

- Retourner sur Tinkercad et ouvrir le fichier créé précédemment.
- Taper *servo* dans la barre de recherche et sélectionner celui ci-dessous :



FIGURE 3.16 – Test servomoteur - Étape 1

- Une fois le servomoteur déposé sur la surface de travail, cliquer sur la petit équerre jusqu'à ce qu'il ait effectué un demi tour sur lui même :

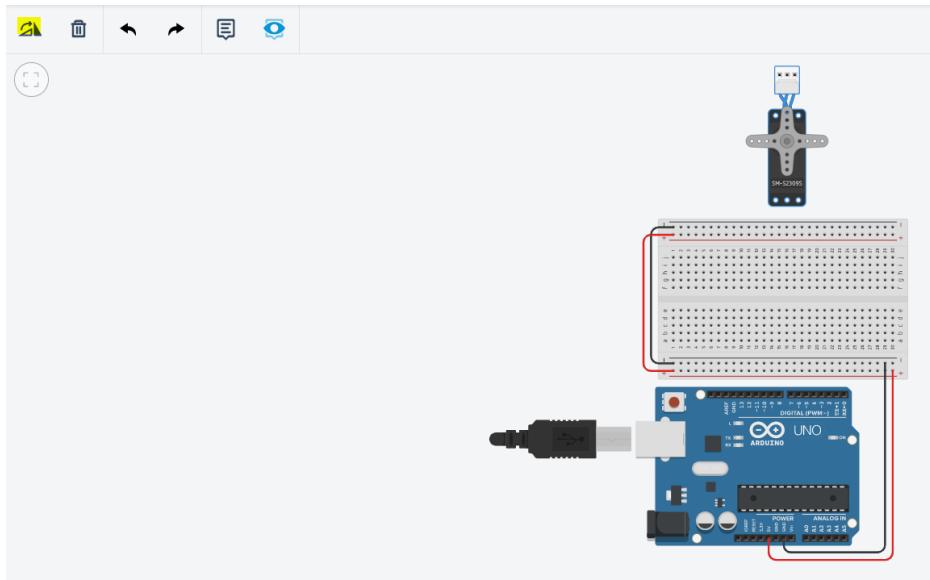


FIGURE 3.17 – Test servomoteur - Étape 2

- Ensuite, relier les différents ports du servomoteur comme ci-dessous :

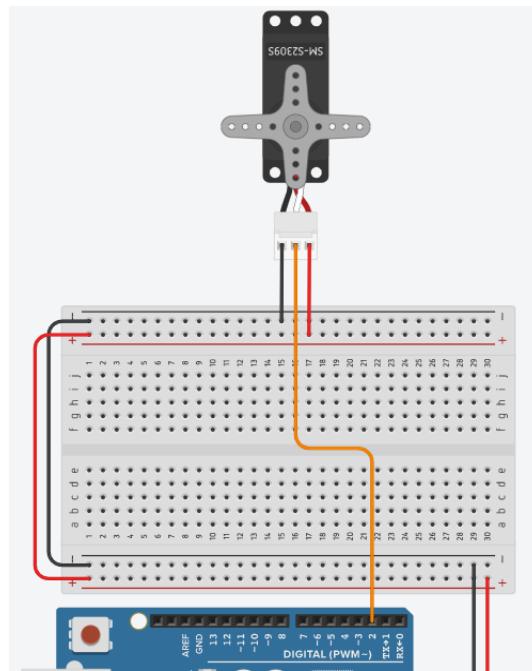


FIGURE 3.18 – Test servomoteur - Étape 3

- Cliquer sur *Code* :



FIGURE 3.19 – Test servomoteur - Étape 4

- Aller dans *Blocs + texte*, *Contrôle* cliquer sur *Totaliser..* et déposer le bloc dans l'espace de travail :

De nouvelles choses apparaissent sur la portion de l'écran comportant le code

```

int i = 0;
void setup()
{
}
void loop()
{
    for (i = 1; i <= 10; i += 1) {
    }
    delay(10); // Delay a little bit to improve simulation
}

```

FIGURE 3.20 – Test servomoteur - Étape 5

- Cliquer sur *i*, *Renommer la variable* et l'appeler "pos" :



FIGURE 3.21 – Test servomoteur - Étape 6

- Retourner dans *Sortie*, cliquer sur *Faire pivoter..* et déposer le bloc dans l'espace de travail :



FIGURE 3.22 – Test servomoteur - Étape 7

- Retourner dans *Contrôle*, cliquer sur *Patienter* et déposer le bloc dans l'espace de travail :

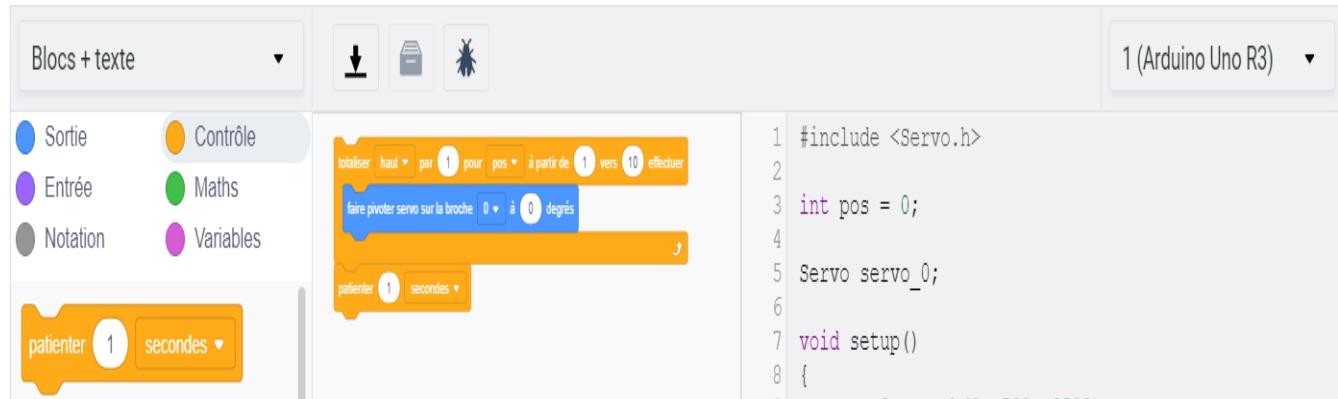


FIGURE 3.23 – Test servomoteur - Étape 8

- Faire un clic droit sur l'ensemble de blocs, cliquer sur *Duplicer* et venir positionner le nouveau bloc sous le précédent :

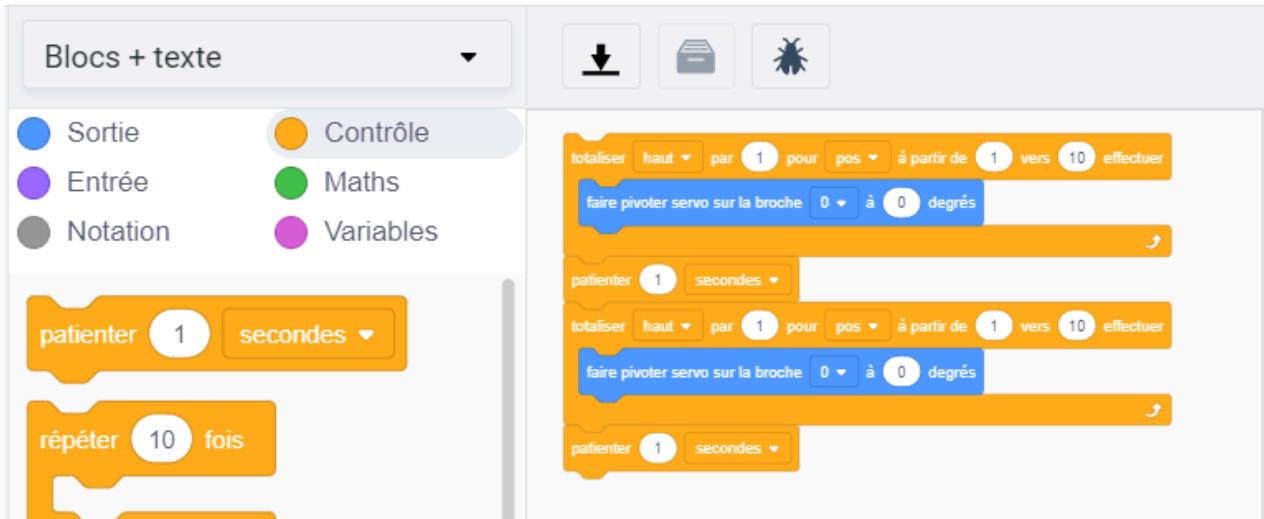


FIGURE 3.24 – Test servomoteur - Étape 9

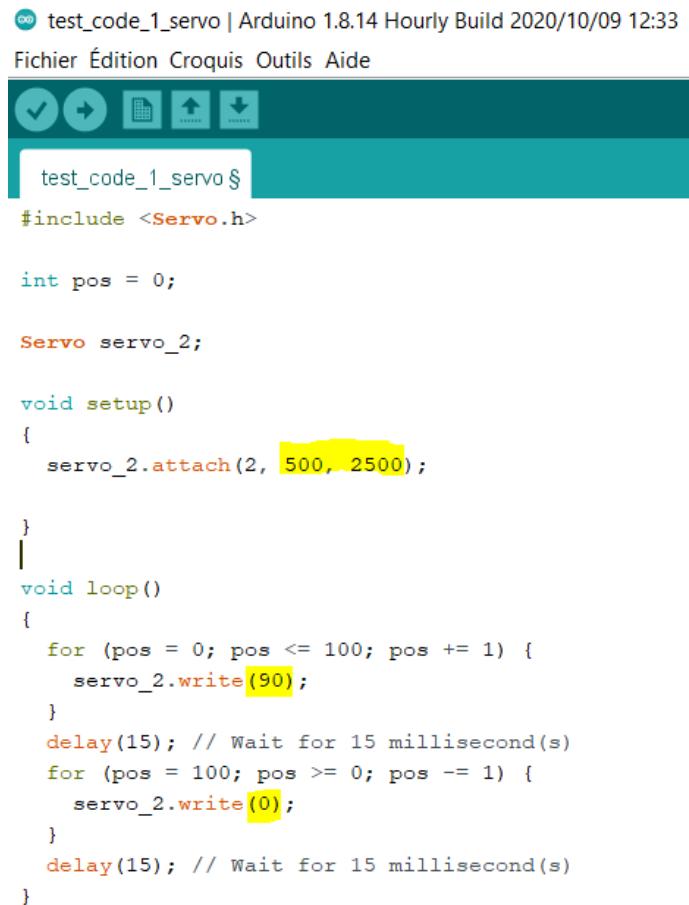
- Modifier les valeurs des blocs comme ci-dessous, voici le code final pour tester le fonctionnement du servomoteur :



FIGURE 3.25 – Test servomoteur - Étape 10

- Copier/coller votre code dans le logiciel Arduino IDE :

Il n'est pas possible de le faire sur Tinkercad lorsque l'on écrit un programme en blocs, il faut donc penser à remplacer les valeurs *0* et *90* par la variable "*pos*", ainsi qu'enlever la plage de valeur de déplacement du servomoteur qui est optionnelle et ne nous intéresse pas ici.



The screenshot shows the Arduino IDE interface with the following details:

- Title Bar:** test_code_1_servo | Arduino 1.8.14 Hourly Build 2020/10/09 12:33
- Menu Bar:** Fichier Édition Croquis Outils Aide
- Code Area:**

```
#include <Servo.h>

int pos = 0;

Servo servo_2;

void setup()
{
    servo_2.attach(2, 500, 2500);

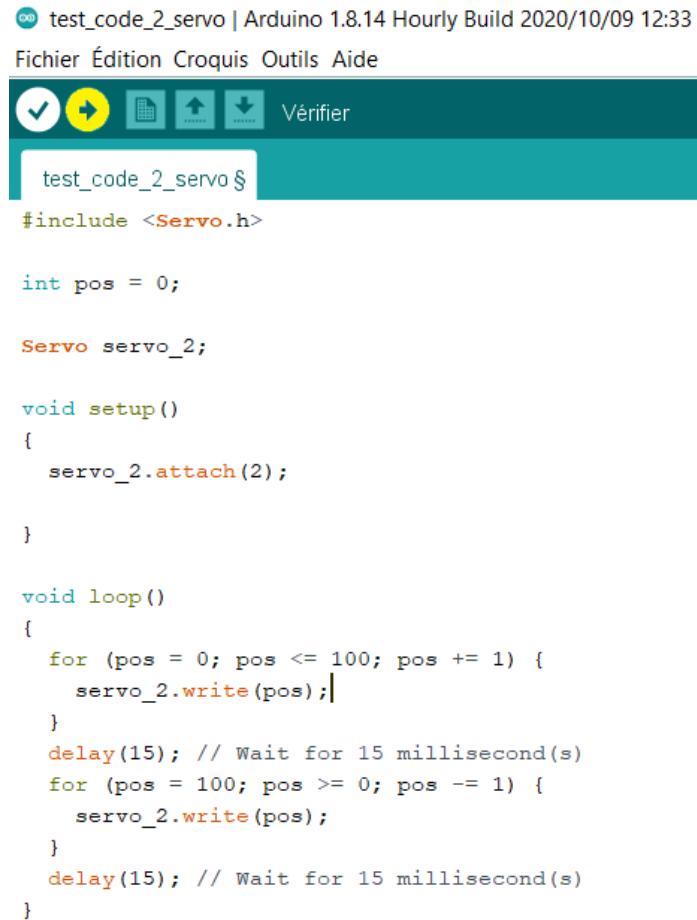
}

void loop()
{
    for (pos = 0; pos <= 100; pos += 1) {
        servo_2.write(90);
    }
    delay(15); // Wait for 15 millisecond(s)
    for (pos = 100; pos >= 0; pos -= 1) {
        servo_2.write(0);
    }
    delay(15); // Wait for 15 millisecond(s)
}
```

FIGURE 3.26 – Test servomoteur - Étape 11

- Effectuer les branchements du servomoteur sur votre carte Arduino (Se référer à la figure 3.18).

- Brancher la carte Arduino à l'ordinateur puis cliquer sur *Téléverser* :



```

    test_code_2_servo | Arduino 1.8.14 Hourly Build 2020/10/09 12:33
    Fichier Édition Croquis Outils Aide
    Vérifier
    test_code_2_servo §
    #include <Servo.h>

    int pos = 0;

    Servo servo_2;

    void setup()
    {
        servo_2.attach(2);
    }

    void loop()
    {
        for (pos = 0; pos <= 100; pos += 1) {
            servo_2.write(pos);
        }
        delay(15); // Wait for 15 millisecond(s)
        for (pos = 100; pos >= 0; pos -= 1) {
            servo_2.write(pos);
        }
        delay(15); // Wait for 15 millisecond(s)
    }
}

```

FIGURE 3.27 – Test servomoteur - Étape 12



Attention : *Il est possible que vos servomoteurs soient montés à l'envers. Si lorsque vous téléversez le programme sur la carte vous voyez que le servomoteur force, débranchez le, puis inversez le - et le + dans votre programme. Téléversez de nouveau, le servomoteur doit se comporter normalement.*

3.2.3 La main en elle même

Malheureusement, Tinkercad ne permet pas de réaliser un code correct pour gérer le mouvement de nos servomoteurs via des potentiomètres. Voici donc le code à réaliser :

```

bras | Arduino 1.8.14 Hourly Build 2020/10/09 12:33
Fichier Édition Croquis Outils Aide
bras §
#include <Servo.h> // On introduit la bibliothèque servo.h

Servo pouce; // On crée un objet servo correspondant au pouce
Servo index; // On crée un objet servo correspondant à l'index
Servo majeur; // On crée un objet servo correspondant au majeur
Servo annulaire; // On crée un objet servo correspondant à l'annulaire
Servo auriculaire; // On crée un objet servo correspondant à l'auriculaire

const int potpin = A0; // On définit une variable potpin sur le pin analogique A0
const int potpin1 = A1; // On définit une variable potpin1 sur le pin analogique A1
const int potpin2 = A2; // On définit une variable potpin2 sur le pin analogique A2
const int potpin3 = A3; // On définit une variable potpin3 sur le pin analogique A3
const int potpin4 = A4; // On définit une variable potpin4 sur le pin analogique A4

int val = 0; // On crée une variable permettant de lire la valeur analogique du potentiomètre associé au pouce
int val1 = 0; // On crée une variable permettant de lire la valeur analogique du potentiomètre associé à l'index
int val2 = 0; // On crée une variable permettant de lire la valeur analogique du potentiomètre associé au majeur
int val3 = 0; // On crée une variable permettant de lire la valeur analogique du potentiomètre associé à l'annulaire
int val4 = 0; // On crée une variable permettant de lire la valeur analogique du potentiomètre associé à l'auriculaire

void setup() {
    pouce.attach(2); // Le servo est relié au pin 2 de l'Arduino
    index.attach(3); // Le servo est relié au pin 3 de l'Arduino
    majeur.attach(4); // Le servo est relié au pin 4 de l'Arduino
    annulaire.attach(5); // Le servo est relié au pin 5 de l'Arduino
    auriculaire.attach(6); // Le servo est relié au pin 6 de l'Arduino

    pinMode(potpin, INPUT); //On définit potpin comme une entrée
    pinMode(potpin1, INPUT); //On définit potpin1 comme une entrée
    pinMode(potpin2, INPUT); //On définit potpin2 comme une entrée
    pinMode(potpin3, INPUT); //On définit potpin3 comme une entrée
    pinMode(potpin4, INPUT); //On définit potpin4 comme une entrée
}

bras | Arduino 1.8.14 Hourly Build 2020/10/09 12:33
Fichier Édition Croquis Outils Aide
bras §
void loop() { //Ce programme sera effectué en boucle

    val = analogRead(potpin); // On lit la valeur analogique du potentiomètre (entre 0 et 1023)
    val = map(val, 0, 1023, 0, 90); // On la met à l'échelle pour l'utiliser avec le servo (entre 0 et 90°)
    val1 = analogRead(potpin1); // On lit la valeur analogique du potentiomètre (entre 0 et 1023)
    val1 = map(val1, 0, 1023, 0, 90); // On la met à l'échelle pour l'utiliser avec le servo (entre 0 et 90°)
    val2 = analogRead(potpin2); // On lit la valeur analogique du potentiomètre (entre 0 et 1023)
    val2 = map(val2, 0, 1023, 0, 90); // On la met à l'échelle pour l'utiliser avec le servo (entre 0 et 90°)
    val3 = analogRead(potpin3); // On lit la valeur analogique du potentiomètre (entre 0 et 1023)
    val3 = map(val3, 0, 1023, 0, 90); // On la met à l'échelle pour l'utiliser avec le servo (entre 0 et 90°)
    val4 = analogRead(potpin4); // On lit la valeur analogique du potentiomètre (entre 0 et 1023)
    val4 = map(val4, 0, 1023, 0, 90); // On la met à l'échelle pour l'utiliser avec le servo (entre 0 et 90°)

    pouce.write(val); // Le servo se met à la position correspondante à la valeur sur le potentiomètre
    index.write(val1); // Le servo se met à la position correspondante à la valeur sur le potentiomètre
    majeur.write(val2); // Le servo se met à la position correspondante à la valeur sur le potentiomètre
    annulaire.write(val3); // Le servo se met à la position correspondante à la valeur sur le potentiomètre
    auriculaire.write(val4); // Le servo se met à la position correspondante à la valeur sur le potentiomètre

    delay(15); // On attend 15 millisecondes
}

```

FIGURE 3.28 – Programmation de la main - Étape 1

Pour le branchement des différents composants sur votre breadboard, référez vous aux figures 2.2, 2.4, 2.5 et 3.18.

4 Performance

4.1 Quelle utilité pour cette main robotique ?

4.1.1 Exprimer son humeur et/ou un mot

En effet, il est possible via l'utilisation des potentiomètres de régler les positions de chaque doigt de manière indépendante. Il est donc possible d'exprimer son humeur via des gestes simples du quotidien.

Quelques exemples :

Pour faire comme les plongeurs et signaler que tout va bien, il suffit de faire former au Pouce et à l'Index un rond. Par conséquent il suffit de bouger les potentiomètres respectifs de chacun de ces doigts pour réaliser cette forme avec votre main !

Pour signaler un énervement, on peut imaginer qu'un poing fermé serait la bonne traduction de cette humeur. Il suffit donc de tourner au maximum chaque potentiomètre afin que tous les doigts se referment sur la paume de la main.

Pour exprimer une réussite, on peut former le V de la victoire avec l'Index et le majeur. Il suffit donc de rabattre le Pouce, l'Annulaire et l'Auriculaire sur la paume de la main.

À vous de trouver d'autres expressions à faire dire à votre main !

4.1.2 Dire un mot ou une phrase en langue des signes

Munissez vous d'un dictionnaire de la langue des signes sur internet, et bougez les potentiomètres de sorte à réaliser une expression proche de celle du mot que vous voulez dire. La langue des signes ne se repose que sur les mouvements de la main mais aussi sur un mouvement de bras et une expression de visage ? Mouvez votre bras robotique avec votre propre main pour réaliser ces mouvements, ou bien trouvez votre propre manière de les réaliser !

4.1.3 Jouer à "Pierre, Papier, Ciseaux"

Si vous souhaitez construire ce bras pour jouer à ce jeu, utilisez trois boutons poussoirs plutôt que des potentiomètres, chaque bouton que vous connecterez à la main correspondra à une des trois positions ! Imaginez et rédigez votre propre programme pour commander les différents mouvements.

5 Pour aller plus loin

5.1 Comment créer son propre modèle en CAO ?

L'étape suivante va être de modéliser vous même les pièces que vous souhaitez imprimer. Pour cela il faudra vous munir d'un logiciel de CAO. Dans la partie d'introduction à la CAO vous trouverez un lien renvoyant vers une liste de logiciels de CAO, pour la plupart étant gratuits ou possédant une version gratuite.

Pour modéliser vos pièces il vous faut un logiciel, mais vous devez surtout définir les dimensions et la forme de la pièce que vous voulez concevoir. Pour cela il faut prendre en compte certaines contraintes comme l'utilité qu'aura la pièce (Un objet de décoration, une pièce mécanique, une pièce destinée à recevoir en son sein un (ou des) composant(s) électronique(s), etc...), la taille qu'elle devra avoir, etc...

Pour prendre en main les logiciels et de manière générale la CAO, il existe de nombreux tutoriels sur YouTube qui permettent de s'initier étape par étape.

5.2 Comment choisir ses composants électroniques ?

Pour choisir vos composants il est nécessaire de vous renseigner sur leurs dimensions (que votre modèle 3D soit adapté pour accueillir les composants choisis). Il faut également se renseigner sur les caractéristiques techniques (datasheet en anglais), afin de s'assurer qu'elles soient en accord avec l'utilisation souhaitée.

Tout cela se trouve généralement sur le site du fabricant ou le site d'achat du composant. Sinon tapez "datasheet" suivi du nom du composant sur internet.

5.3 Perspectives d'amélioration

Si vous souhaitez pousser encore plus loin votre expérience, vous pouvez envisager de rendre votre main un peu plus indépendante de votre Arduino en portant votre carte sur une breadboard ou, encore plus ambitieux, vous lancer dans la soudure et la porter sur une plaque de PCB.

Il est également possible d'essayer de vous rapprocher encore plus du projet de TechMartian en essayant de contrôler la main par Bluetooth.



Attention : *En plus des composants nécessaires mentionnés en figure 2.1, il vous faudra investir dans un module Bluetooth compatible avec la carte Arduino car nous n'utilisons pas la même que TechMartian, et il faudra également modifier le programme Arduino.*

Il est également envisageable de décider de modéliser son propre robot via un logiciel, pour concevoir un robot complet. De nouvelles contraintes se présenteront alors, différentes de celles que nous avons rencontré lors de ce projet.

Nous n'avons ici présenté que certains exemples d'améliorations à apporter, mais il en existe sans doute bien d'autres ! À vous de réfléchir à celles qui vous intéressent et à la manière dont vous souhaitez les mettre en place.

Annexe 4

Ci-dessous, quelques photos des tests réalisés avec l'imprimante, puis de la version finale des pièces :



Figure 1 – 1ère (à gauche) et 2nde (à droite) version des doigts lors des tests



Figure 2 – Version des doigts avec les paramètres finaux

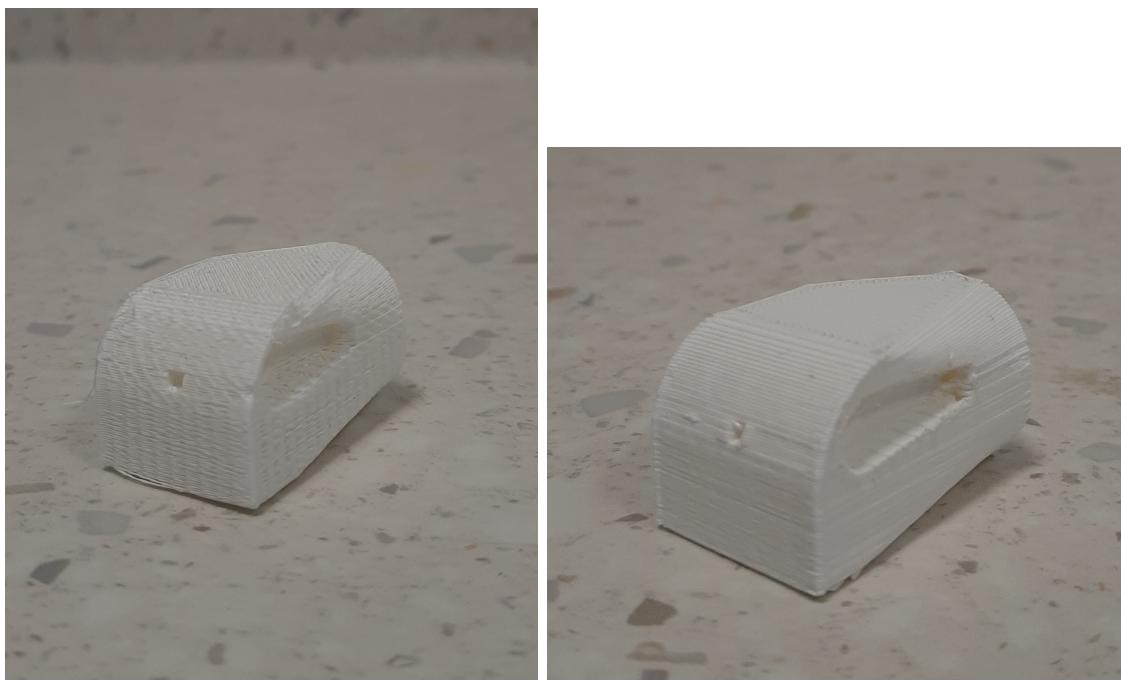


Figure 3 – Version de test de la partie amovible du pouce (à gauche) et version finale (à droite)

Annexe 5



ATTESTATION DE SUIVI AVEC SUCCÈS

Enseignants

Fabien Autrel

Ingénieur - IMT Atlantique

Baptiste Gaultier

Ingénieur - IMT Atlantique

Joris ALBALADEJO
a suivi avec succès le MOOC*

Imprimer en 3D
proposé par IMT
et diffusé sur la plate-forme FUN
Le 03/03/2021

<https://www.fun-mooc.fr>

* MOOC : cours en ligne

La présente attestation n'est pas un diplôme et ne confère pas de crédits (ECTS).
Elle n'atteste pas que le participant était inscrit à/au IMT.
L'identité du participant n'a pas été vérifiée.

Annexe 6

Ci-dessous, le lien vers la vidéo d'application du code de test d'un servomoteur :

https://youtu.be/Z_8ZKCjNHdY

Annexe 7

Ci-dessous, le lien vers la vidéo du code de fonctionnement de la main :

<https://youtu.be/NMrZOX8DzA8>