

Préambule : « Le projet de spécialité SI (12h) / Thématiques, Problématiques & Compétences »



- Vérifier le cahier des charges du produit afin de savoir s'il est adapté aux besoins exprimés.
- Modéliser le produit, le simuler, l'expérimenter afin de le faire évoluer vers un produit adapté.



**+1** milliard  
de personnes  
handicapées  
dans le monde

**6%**

de la population mondiale  
touchée par une forme de  
handicap auditif

**1%**

de la population mondiale  
a besoin  
d'un fauteuil  
roulant

**2,6%** de la population mondiale  
touchée par un handicap  
intellectuel

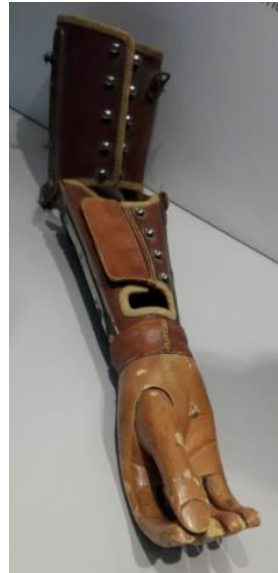
**17%**

de la population mondiale  
touchée par une forme de  
déficience visuelle



## 1/. Thématique d'étude : « L'homme assisté, réparé, augmenté ».

Depuis de nombreuses années, l'homme a essayé de réparer ses membres avec des prothèses : on date ainsi de l'Égypte ancienne, entre 710 à 950 avant JC, les premières prothèses de doigts de pied.

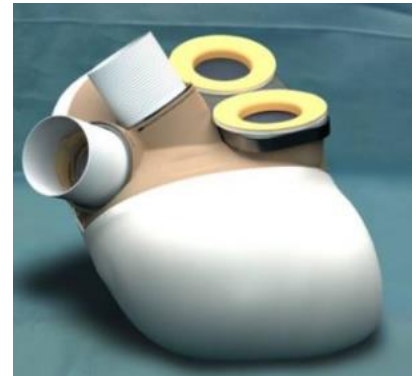


1. Ci-contre, prothèses de main, d'avant-bras (avec pouce articulé) et de jambe datées de 1910.

Les techniques ont ensuite fait un bond en avant lors du conflit de la première guerre mondiale<sup>1</sup>, jusqu'à aujourd'hui, où des **exosquelettes**<sup>3</sup> couplés à des impulsions neuronales permettent à des invalides de retrouver le sens de la marche, mais à des coûts encore très élevés !

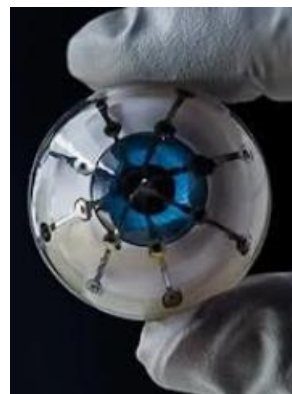
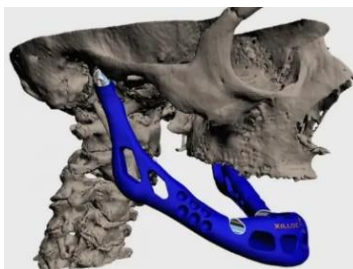
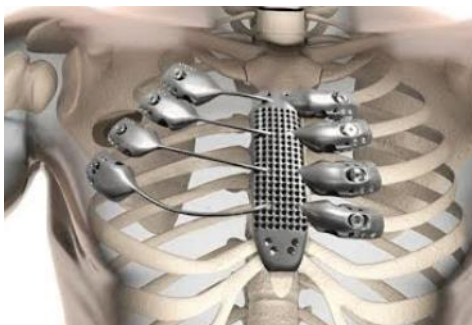


2. Ci-contre en 2020, le cœur artificiel « Carmat »



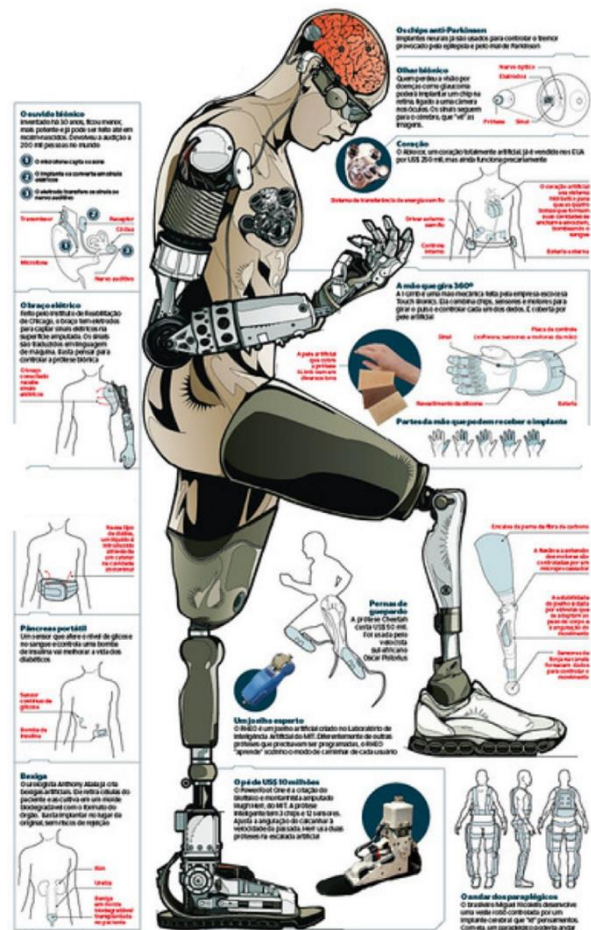
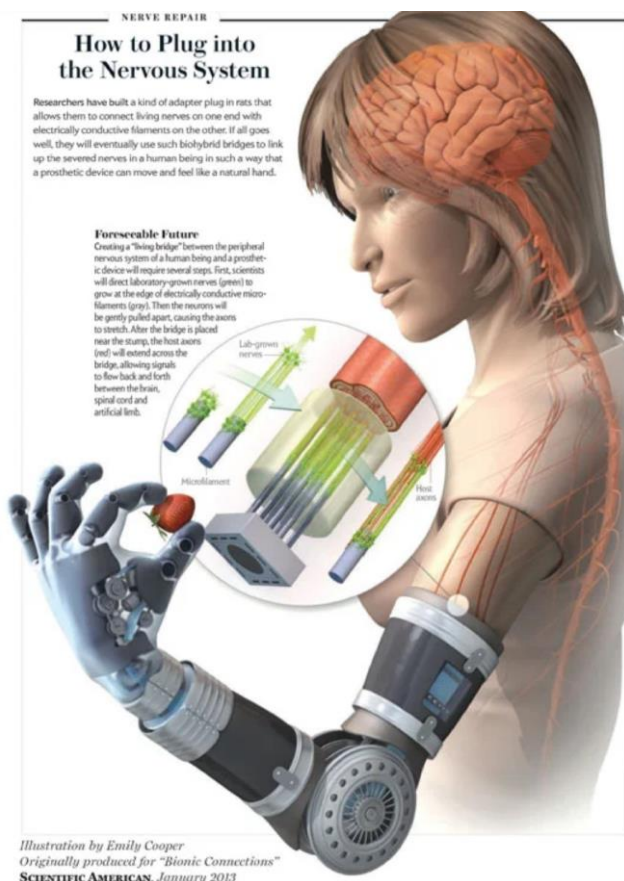
3. En 1960, « Hardiman » fut le premier prototype d'exosquelette.

Réparer notre corps comme nous réparons nos voitures, en allant changer des pièces, pourrait-il devenir une réalité accessible au plus grand nombre ?



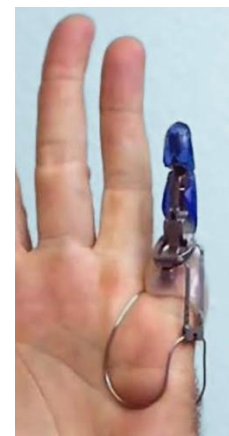
Oui, répondent certains : grâce à **l'impression 3D** qui ne se limite désormais plus à des exosquelettes, mais qui permet également de reconstituer des os, des vaisseaux sanguins, des valves cardiaques, des oreilles, de la peau synthétique, et peut-être bientôt des organes complets comme les poumons ou le cœur<sup>2</sup> ...





En revanche, parmi les prothèses purement mécaniques, on trouve les X fingers<sup>4</sup> (de Didrick Medical), dépourvus de tout système électrique, ils permettent de retrouver une mobilité fonctionnelle et des actions de saisies élémentaires.

Le système est à la fois ingénieux et simple, le mécanisme utilise les restes amputés des doigts et à l'aide de combinaisons de ressorts et de tenseurs, il reproduit les articulations d'un doigt « organique ».



La reproduction est si parfaite qu'un patient portant ces doigts artificiels peut saisir une pièce, faire ces lacets, porter une charge ou même jouer du piano.

Les patients peuvent porter un ou plusieurs X fingers à la fois. Leur légèreté est aussi remarquable, conçu en acier chirurgical, un X finger pèse moins de 10 grammes, et environ 30 grammes si le patient souhaite recouvrir sa prothèse de doigt d'une peau artificielle en silicone faite sur mesure.

#### 4. X fingers de « Didrick Medical » (prothèse mécanique, non électrique)

## 2/. Challenge de spécialité SI (12h) : Commande « robotisée » de la gestuelle des mouvements des doigts (phalanges) d'une main.

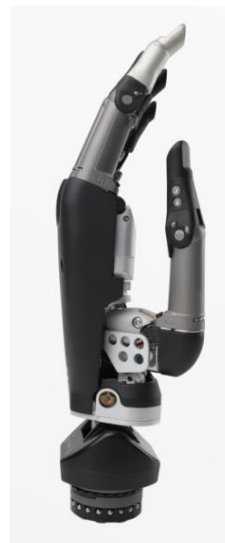
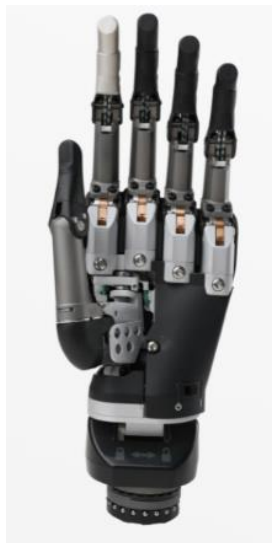
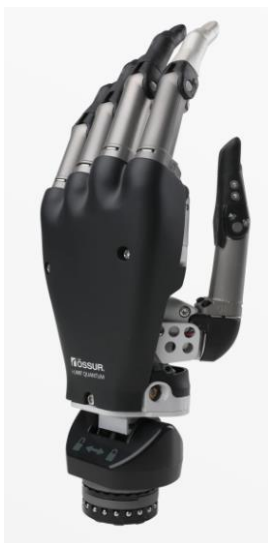
**D**e tout temps, parmi les différents organes ayant subi le plus de handicaps et de mutilations, la main prédomine.



Sans doute par sa fonction essentielle de « **préhension** » (action, faculté de prendre, de saisir) et de l'importante utilité qu'elle revêt dans tous les actes du quotidien, l'homme a donc cherché à lui substituer son équivalent « mécanique » dans un premier temps (Ambroise Paré 1510-1590), puis « biométrique robotisé » aujourd'hui.



La prothèse de main bionique i-Limb Quantum (voir ci-dessous), main myoélectrique poly-digitale d'Össur, permet aux utilisateurs de modifier leurs prises d'un geste, avec gestion de la puissance et de la rapidité de mouvement. Elle est par ailleurs équipée de série avec des doigts en titane.



<https://www.ossur.com/fr-fr/protheses/bras/i-limb-quantum>

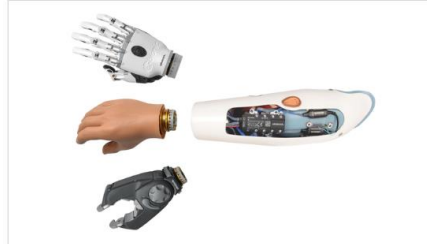


# Main bebionic EQD – esthétique, naturelle, adaptable



## Main multi-articulée très fonctionnelle

Un moteur placé dans chaque doigt offre une saisie naturelle et adaptative d'objets.



## Compatible avec la reconnaissance de formes Myo Plus

Adaptée à la reconnaissance de formes Myo Plus grâce à une interface numérique.



## Doigts mobiles passifs

En cas de contact avec des personnes ou de choc involontaire contre des objets, les doigts se plient sans provoquer d'inconfort pour protéger les composants mécaniques.



## Module Bluetooth pour les deux tailles de main

Connexion facile au logiciel de réglage bebalance+ avec un seul adaptateur Bluetooth.



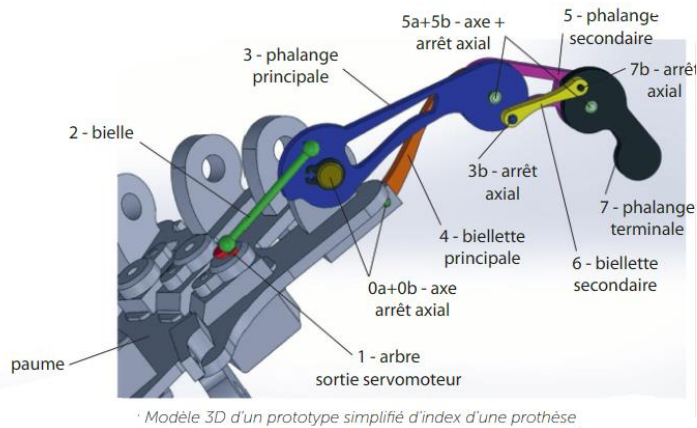
## Trois variantes de poignet

Différentes hauteurs d'alignement permettent une utilisation variée et adaptée au niveau d'amputation de l'utilisateur.

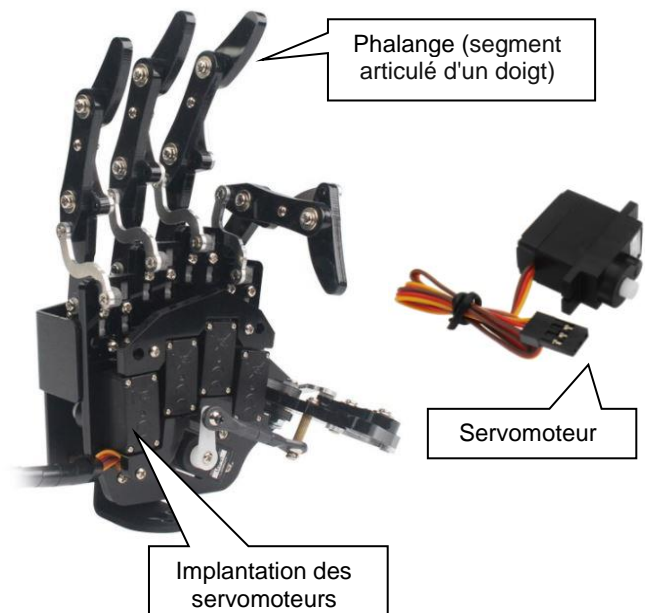


## Compatible avec la DynamicArm sans AnalogAdapter

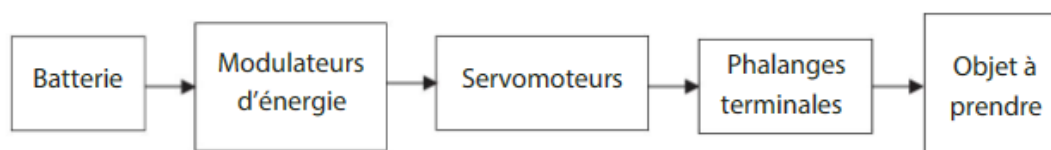
<https://www.ottobock.com/fr-fr/product/8E70>



Le système objet de ce défi est une main « robotisée » (voir ci-contre) déjà assemblée, équipée de **5 servomoteurs** (logés au niveau de la paume, dans la face interne de la main) et assurant la commande des mouvements des ensembles de phalanges de chaque doigt.



## Chaîne de puissance instantanée classique :



Les phalanges constituant chaque doigt sont liées entre elles et pilotées par **un servomoteur** (associé à un « effecteur », voir ci-dessus) dédié à chaque doigt. Le schéma cinématique de la figure 1 fait apparaître le type de liaisons utilisées pour la commande d'un doigt.

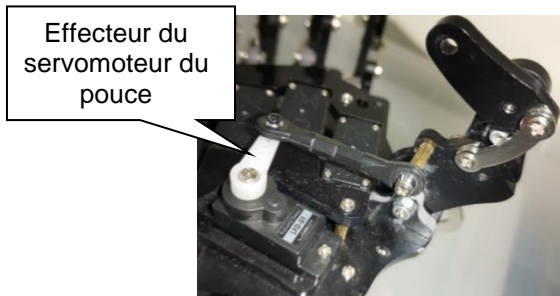
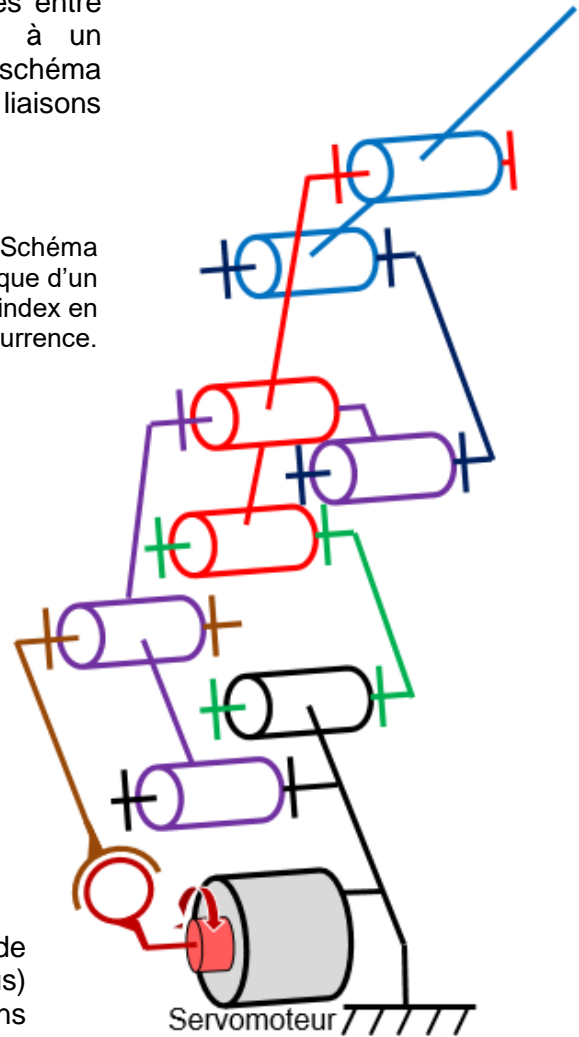


fig.1 : Schéma cinématique d'un doigt, l'index en l'occurrence.

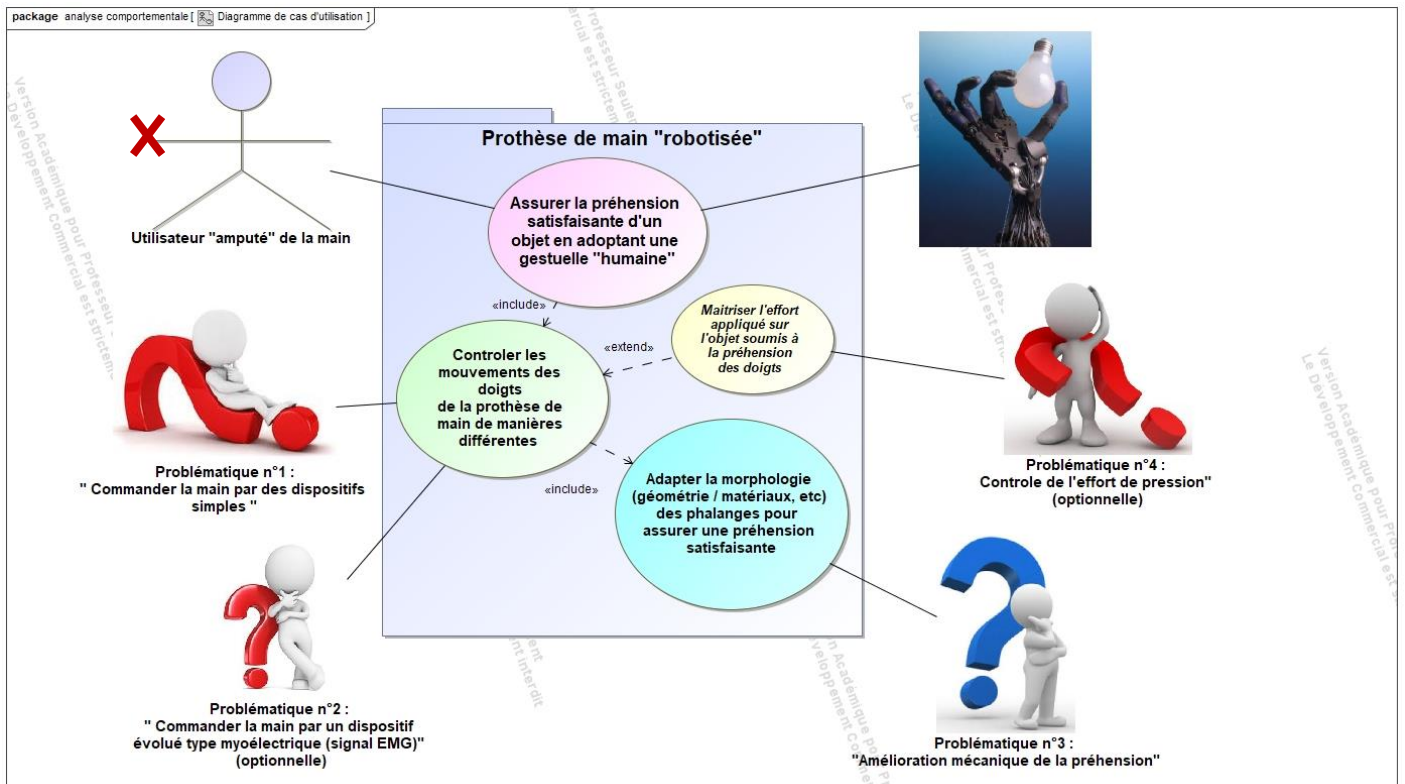


Cette main est en outre parfaitement opérationnelle, si tant est qu'on lui rajoute une partie traitement de l'information (capteurs, interface logicielle et matériel, etc.) lui permettant d'adapter et de synchroniser les gestes qu'elle doit réaliser !

Pour être complet, le poids de cette main est d'environ **185 grammes**. Les doigts sont réalisés en **plexiglass PMMA** coulé noir opaque/brillant.



Le diagramme SysML comportemental de « cas d'utilisation » Use Case (ci-dessous) montre les services et interactions fonctionnelles attendues de ce système et les différentes problématiques associées.





## SysML ⇨ « Langage de Modélisation Système »

« SysML ou Systems Modeling Language » est un langage de modélisation adapté pour l'Ingénierie Système. L'Ingénierie Système ou Systems Engineering, est une démarche méthodologique pour répondre à des problèmes complexes par la réalisation de solutions logicielles et matérielles. L'Ingénierie Système s'adresse à de nombreux secteurs tels que l'automobile, la sécurité numérique, le ferroviaire, l'aéronautique, l'espace, le militaire, les télécoms, le médical, la production d'énergie, etc.

Les méthodes de l'Ingénierie Système (IS) reposent sur des approches de modélisation et de simulation pour valider les exigences ou pour évaluer le système.

SysML est apparu à l'issue d'initiatives pour une évolution vers une approche orientée « modèles ». Cette modélisation permet de maîtriser la complexité du système, car chaque modèle donne accès à une représentation abstraite mais structurée de différents aspects du système.



A noter, que seules les études sur les problématiques 1 et 3 vont nous intéresser dans un premier temps ! En effet, les problématiques 2 et 4 de la commande « myoélectrique » et de la maîtrise de l'effort, sont plus délicates à appréhender en première et seront davantage à envisager dans le cadre du projet de spécialité de terminale !

### 3/. Consignes de travail

L'organisation du travail pour ces études s'effectuera par groupe de trois étudiants sous la forme d'un travail collaboratif complet et achevé, inventif et non exhaustif permettant, au travers des réponses aux différentes questions posées dans ce document, de mettre en œuvre et de faire fonctionner correctement cette main.

*Vous êtes en revanche libre de votre organisation interne au sein du groupe !*

Une phase finale de restitution sera à produire par groupe sous la forme :

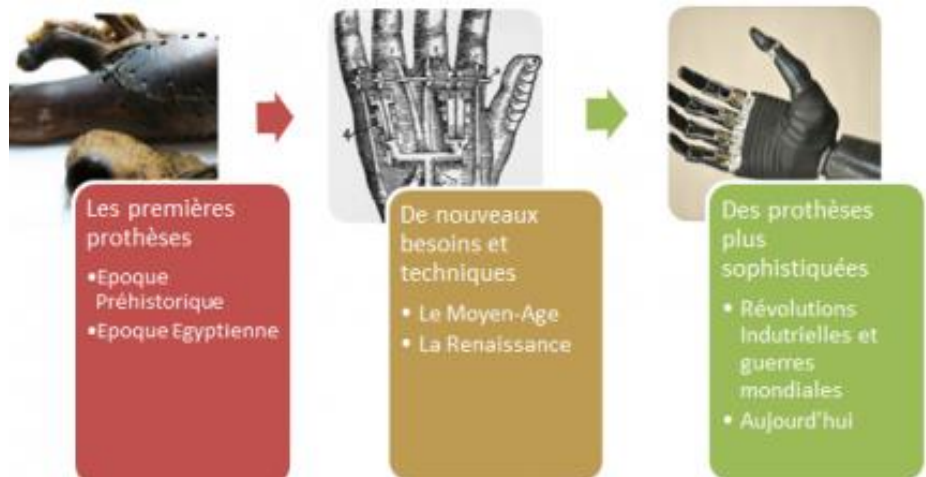
- d'un dossier de synthèse au format (.pdf),
- et d'un document multimédia (Powerpoint, Prezi, Slides, etc.) pour la présentation orale de ce projet.

Chaque groupe tiendra à jour également sa progression, ainsi que l'évolution de chaque élève du groupe, et ce sous la forme d'un « carnet de bord » recensant l'ensemble des travaux effectués au fil des 6 / 7 séances de projet.

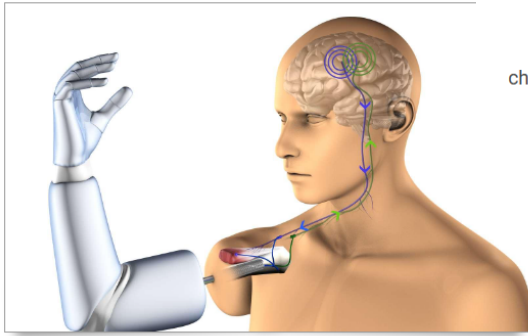
Ces carnets de bord faciliteront la restitution finale et pourront être demandés pour évaluer le rythme de chaque « équipe » sur ce projet.

### 4/. Questions à répondre « en équipe »

4.1/. Elaborer une frise chronologique montrant les évolutions des prothèses et tout particulièrement de celle de la main durant les trois « époques » suivantes en développant plus particulièrement les évolutions et projections futuristes de ces mains.







Frise chronologique

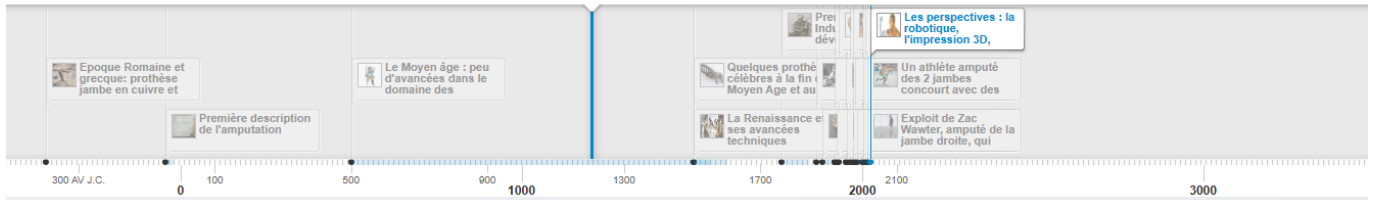
2021

## Les perspectives : la robotique, l'impression 3D, introduire des cellules vivantes au sein des prothèses

Il existe plusieurs axes d'études et de développement concernant les prothèses.

Une des principales perspectives en matière de prothèses repose sur les progrès de la robotique, ou encore sur l'impression 3D, qui permet de réduire le coût de fabrication.

Les chercheurs cherchent également à se rapprocher de la greffe, en introduisant des cellules vivantes au sein des prothèses



4.2/. Identifier les éléments particuliers de la main robotisée présentée avec son schéma cinématique (voir page 6) en correspondance avec les os d'une main humaine réelle décrite ci-contre ? Quelle différence principale constatez-vous ?

4.3/. Comment définiriez-vous le type de liaisons mécaniques entre phalanges, entre phalanges et os métacarpiens et entre os métacarpiens et carpiens pour une main réelle ? Argumenter.

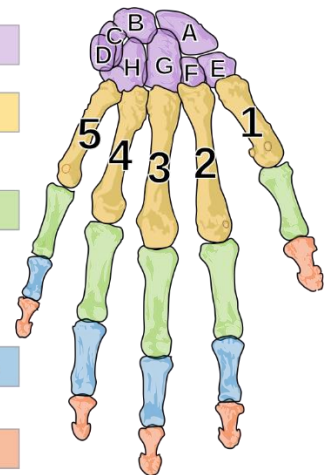
Os carpiens

Os métacarpiens

Phalanges proximales

Phalanges intermédiaires

Phalanges distales



Potentiomètre rotatif

## 5/. Problématique n°1 & 2 : « Contrôler & mettre en mouvements les phalanges des doigts »

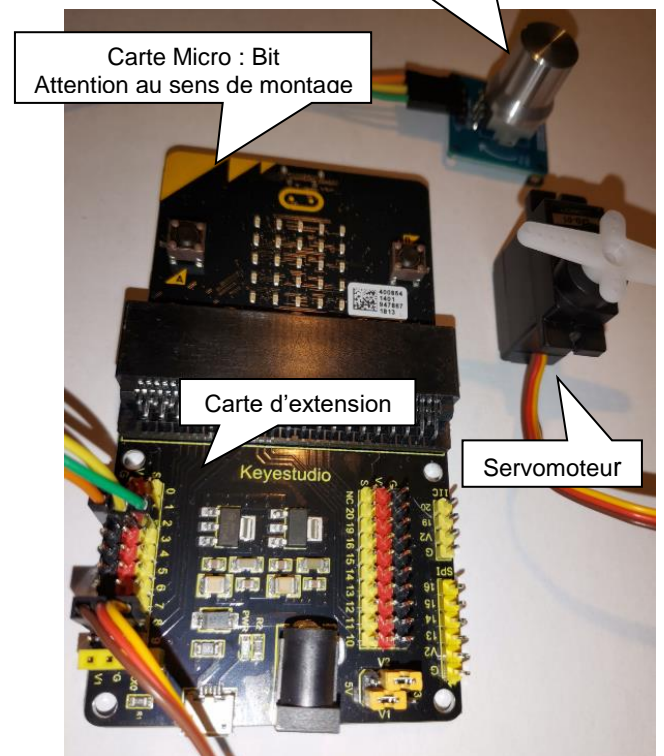
Dans un premier temps, nous allons chercher à piloter un doigt (le pouce en l'occurrence) en position « repliée », puis en « extension ».

Pour cela nous disposons d'un servomoteur (voir explication du fonctionnement page suivante), identique à celui implanté dans la main, pour agir sur le pouce et d'un potentiomètre rotatif permettant de contrôler sa position angulaire. La photo ci-contre permet d'appréhender le montage de la carte Micro:Bit dans sa carte d'extension.

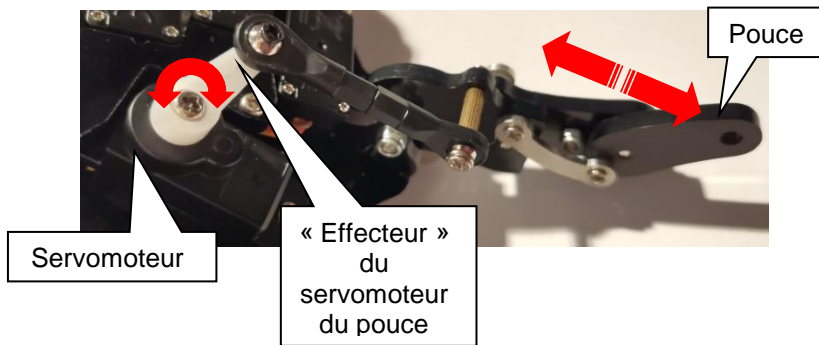
Cette carte d'extension « Keystudio » permet de connecter la résistance variable (ou potentiomètre) à la carte Micro:Bit, ainsi que le (ou les) servomoteur(s) aux différentes sorties « PWM » (anglicisme de Pulse Width Modulation [Signal modulé en largeur d'impulsion]).



Attention, ne vous fiez pas aux couleurs des différents câbles de connexion qui peuvent changer !  
En cas de doute, appelez le professeur pour vérification !







Ci-contre, servomoteur de commande du pouce de la main robotisée. La rotation de l'axe va déplacer son effecteur (pièce blanche) provoquant ainsi **la rétraction ou le déploiement** du pouce !



Attention à respecter scrupuleusement les raccordements, connexions et polarités indiqué(e)s dans chaque étape sous peine de **destruction « irréversible »** du matériel.

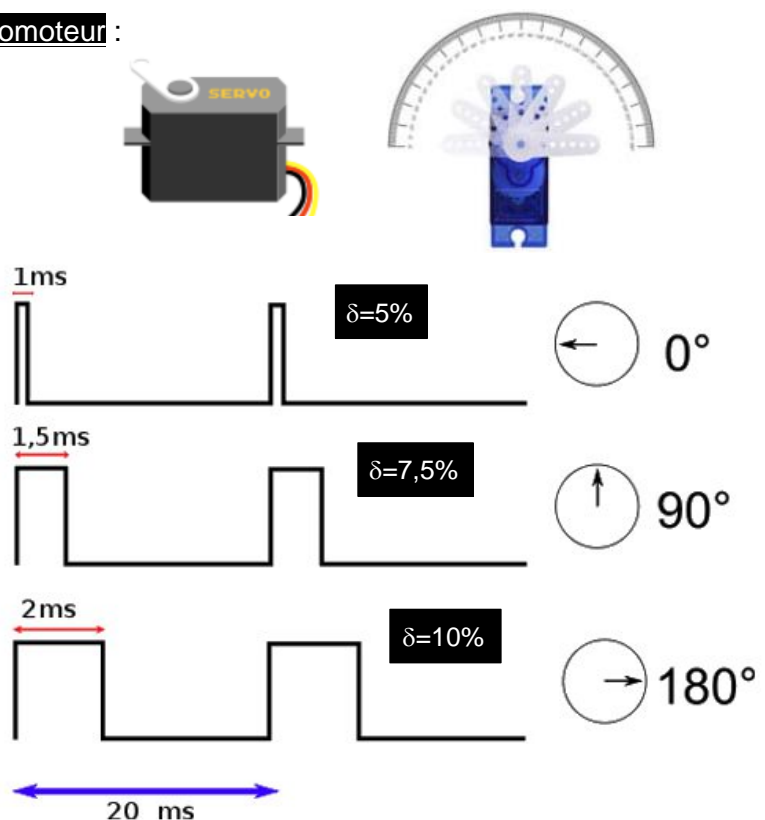
En cas de doute, ne pas mettre sous tension (... ni avec le port USB, ni avec le bloc secteur !) et ne pas hésiter à appeler le professeur.


Penser également à réintégrer **l'intégralité du matériel en fin de séance** !

### ➤ Principe de commande d'un servomoteur :

Il faut demander à la carte Micro:Bit de générer un signal de type « **MLI** » ( **Modulation de Largeur d'Impulsion**) aussi appelé **PWM** (pour « Pulse Width Modulation ») de **période  $T=20\text{ms}$**  correspondant à une fréquence de **50Hz** ! (voir ci-contre)

Ainsi, la position de l'axe du servomoteur se déplacera proportionnellement à cette commande. Les changements  $0^\circ$ ,  $90^\circ$  et  $180^\circ$  de positionnement de l'axe seront obtenus en fonction des trois valeurs imposées du rapport cyclique (voir ci-contre signaux de commande initiaux).

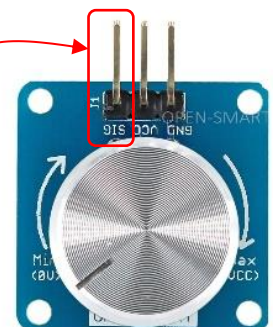


 Attention, certaines valeurs peuvent parfois **varier** en fonction du servomoteur utilisé !

### ➤ Potentiomètre (résistance variable) et signal analogique $V_{SIG}$ produit :

La tension analogique  $V_{SIG}$  (broche SIG sur le connecteur 3 broches) issue du potentiomètre est convertie en une valeur numérique par un « **CAN** » (Convertisseur Analogique / Numérique) interne à la carte Micro:Bit sur **un format de 10 bits**.

Cela signifie que la tension d'entrée de  $V_{SIG}$  comprise initialement entre **0 et 5V** sera convertie en une valeur numérique entière comprise entre **0 et 1023**, soit 1024 échantillons numériques différents !

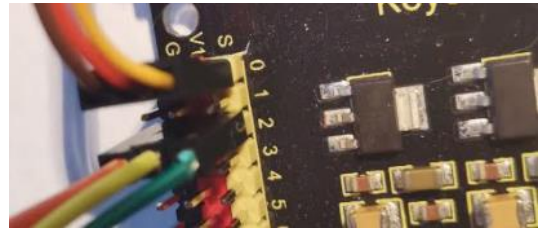




Pour la programmation sous MicroPython, nous utiliserons l'environnement habituel « Mu » associé à une carte Micro:Bit. Cet environnement est simple, libre de droit et gratuit (<https://codewith.mu/en/download>).

Le programme partiel « One\_finger » de commande du servomoteur (connecté au slot / broche 0) associé au pouce et commandé par un potentiomètre (connecté au slot / broche 2) est donné à la page suivante.

Les connexions à réaliser sont celles ci-contre !



### 5.1/. Compléter le programme partiel nommé « One\_finger » du pouce en page suivante, en détaillant chaque commentaire incomplet ?

Vous remarquerez la présence d'une « fonction » `def servo(angle)` {sorte de sous-programme} améliorant la lecture du programme que vous analyserez !

```
1 from microbit import *
2 pin0.set_analog_period(20)
3
4
5 def servo(angle):
6     pin0.write_analog(angle)
7
8
9
10 while True:
11     pot = pin2.read_analog()
12     print('pot = ', pot,)
13     print((pot,))
14     angle = (pot / 1023) * 180
15     print('angle = ', angle, 'Deg')
16     servo(angle)
17     sleep(20)
```

Grapher BBC micro:bit

BBC micro:bit REPL

```
(413,)
angle = 73.0205 Deg
pot = 414
(414,)
angle = 72.8445 Deg
pot = 414
(414,)
```



Appelez le professeur pour validation et procédure de câblage et tests sur carte.

### 5.2/. Implémenter le programme précédent dans la carte Micro:Bit. Analyser le comportement réel du servomoteur lors de la rotation du potentiomètre, y compris en suivant les paramètres du « Grapheur BBC Micro:Bit » et de l'outil « REPL » (*Read Evaluate Print Loop*) permettant à la carte Micro:Bit de lire et d'évaluer le code en temps réel au fur et à mesure de l'exécution de son programme. Est-ce conforme à l'attendu d'une variation approximative d'environ 180° de l'axe du servo ? Justifier, argumenter !



Attention, la manipulation (rotation) du potentiomètre de commande ne doit pas être trop rapide !



5.3/. Créer un nouveau programme nommé « Two\_fingers » permettant de contrôler simultanément le pouce (connecté au slot / broche 0) et l'index (connecté au slot / broche 1) toujours à l'aide du potentiomètre (connecté au slot / broche 2). Connecter, pour l'index, un autre servomoteur à la broche n°1. Implémenter votre programme dans la carte Micro:Bit, puis le tester et conclure !



Appelez le professeur pour validation et tests (mesures) à l'oscilloscope.

5.4/. Pour ces mesures, retirer le servomoteur d'index, mettre en place la sonde de l'oscilloscope, puis relever le signal  $V_{INDEX}$ , pour les deux positions extrêmes du potentiomètre. Capturer les écrans des différents signaux à l'aide de l'interface USB et conclure sur leur exactitude en regard de ceux précédemment présentés ?

Mesurer les différentes valeurs instantanées (période, fréquence, amplitude, rapport cyclique) sur les relevés (voir exemple de signaux « PROF » à la page suivante).

Sont-elles conformes avec celles attendues (en page 8 et en page 10 pour les versions « PROF ») pour la commande d'un tel servomoteur ? Noter les éventuelles différences !

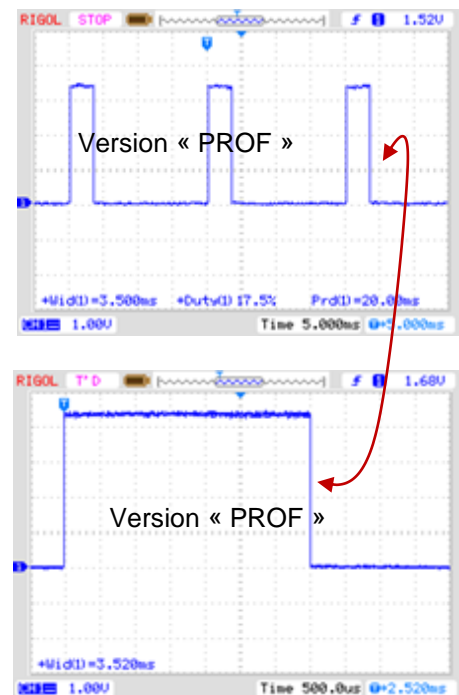
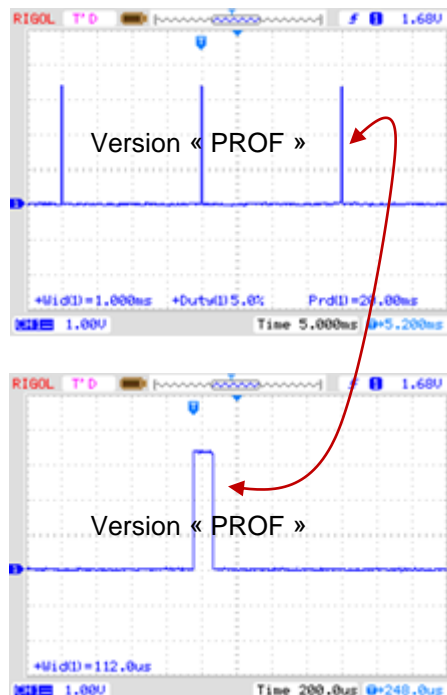


Position « 0° » [Index en « extension »]

Position « 180° » [Index « replié »]

⚠ Vos relevés, du signal  $V_{INDEX}$ , (... quelque peu différents de ceux « PROF » ci-contre !) seront commentés et analysés dans votre document de synthèse de fin de projet.

L'intégralité des mesures faites à l'oscilloscope (en capture d'écran) doivent également y figurer !



5.5/. Positionner le servomoteur à  $\approx 90^\circ$ , capturer le signal correspondant observé à l'oscilloscope, puis le justifier !



Appelez le professeur pour validation et explication de l'activité suivante.

On souhaite désormais commander le(s) servomoteur(s) par les boutons A et B disponibles sur la face avant de la carte Micro:Bit.

Le fonctionnement souhaité est décrit ci-dessous :

- ✓ Tant que le bouton A est actionné, le servomoteur devra tourner dans le sens « horaire », ou bien tant que le bouton B est actionné, le servomoteur devra tourner dans le sens « anti-horaire » !

5.6/. Proposer un programme nommé « AB\_finger » contrôlant le mouvement d'une phalange (1 servo) dont la structure reprendra celle du programme incomplet ci-contre. L'implémenter dans la carte Micro:Bit, puis le tester en l'améliorant afin qu'il réponde parfaitement au cahier des charges précédent.

```
Mu 1.0.3 - Servo_BP.py
Mode Nouveau Charger Enregistrer Flasher Fichiers REPL
Alum_led.py 1_servo.py Robot_Maqueen_Evit_Obst.py Two_finger.py Servo_E
1 from microbit import *
2 .....
3 .....
4 def servo(.....):
5     .....
6
7 while True:
8     servo(.....)
9     while button_a.is_pressed():
10        .....
11        .....
12        .....
13    while button_b.is_pressed():
14        .....
15        .....
16        .....
```



Appelez le professeur pour validation et explication de l'activité suivante.

5.7/. Revenir au programme « Two\_fingers » et remplacer les 2 servomoteurs par ceux de la main réelle, pour le pilotage simultané du pouce et de l'index. Observer l'amplitude des mouvements réalisés (index / pouce en phase « d'extension », puis « repliés »).

La commande vous paraît-elle satisfaisante en regard des mouvements réalisés ? A priori, un dysfonctionnement devrait apparaître dans la « cohérence » du mouvement ? Le décrire et modifier le programme en conséquence pour obtenir un fonctionnement satisfaisant !

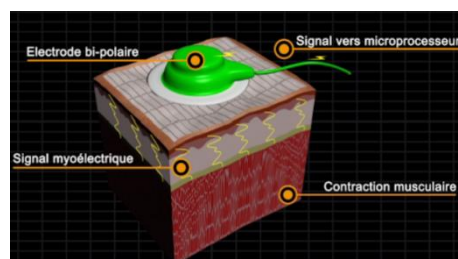


Appelez le professeur pour validation et explication de l'activité suivante.

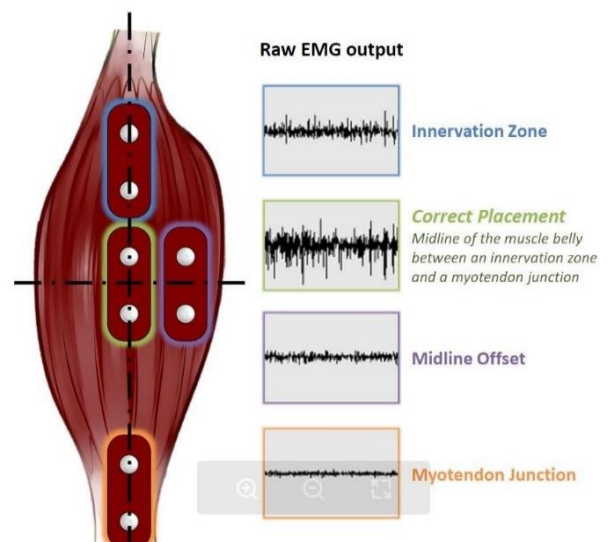
## 6/. Commande des actions simples de la main

Les recherches actuelles tendent à développer des interfaces de contrôles de mieux en mieux adaptées entre l'homme et le contrôle de sa prothèse.

Ci-contre, exemple de l'utilisation du signal électromyographique musculaire EMG permettant, au travers d'un capteur spécifique, de détecter l'activité électrique des nerfs et des muscles du bras



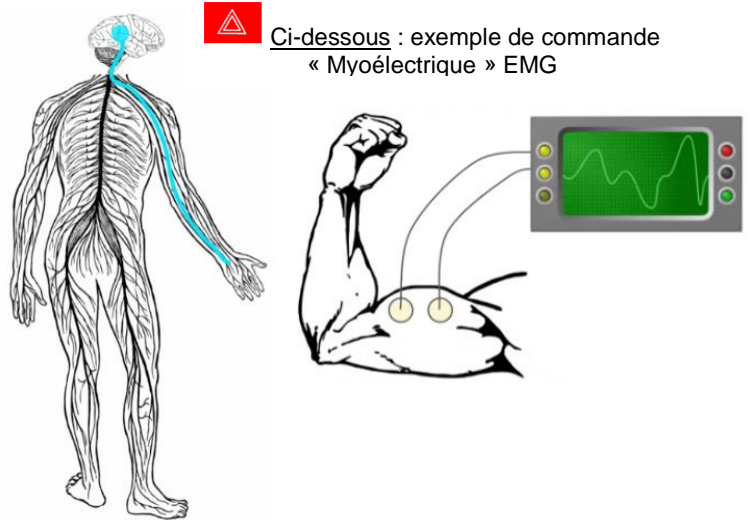
afin d'agir directement sur la commande de la prothèse de main robotisée. Les mains « myoélectriques » ont la capacité de se fermer, de s'ouvrir et d'inclure parfois une rotation du poignet !



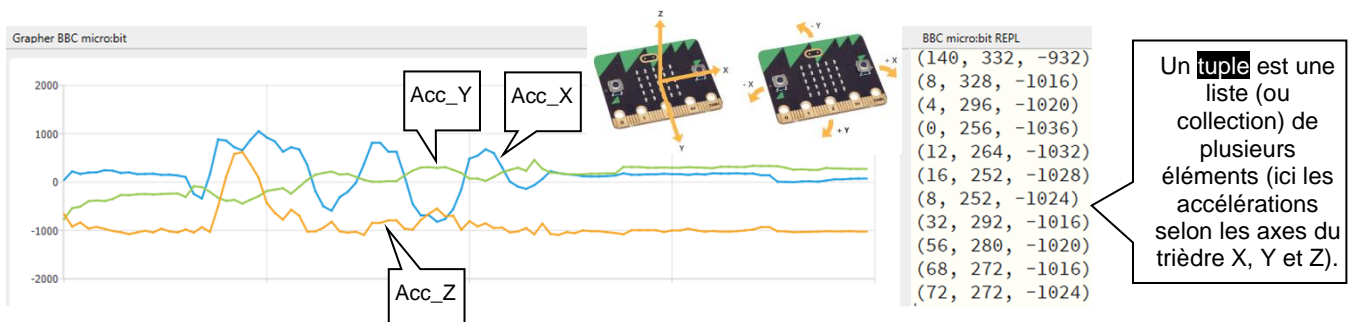


Une, des nombreuses solutions de commande pouvant être envisagées, serait la commande à distance de la main à partir d'un **accéléromètre** porté, comme une montre, sur le membre valide et envoyant, par ondes radios, les ordres de mouvement à la prothèse équipée de sa carte de commande.

A l'évidence, il ne pourra s'agir que de mouvements simples de la main, mais extrêmement usuels et pratiques au quotidien ! A vous de proposer une solution fonctionnelle en utilisant de préférence l'accéléromètre de la carte Micro:bit. A noter que le magnétomètre/compas (boussole), également présent sur cette carte, peut également être utilisé (attention à son étalonnage préalable !)



Ci-dessous : exemple de commande « Myoélectrique » EMG

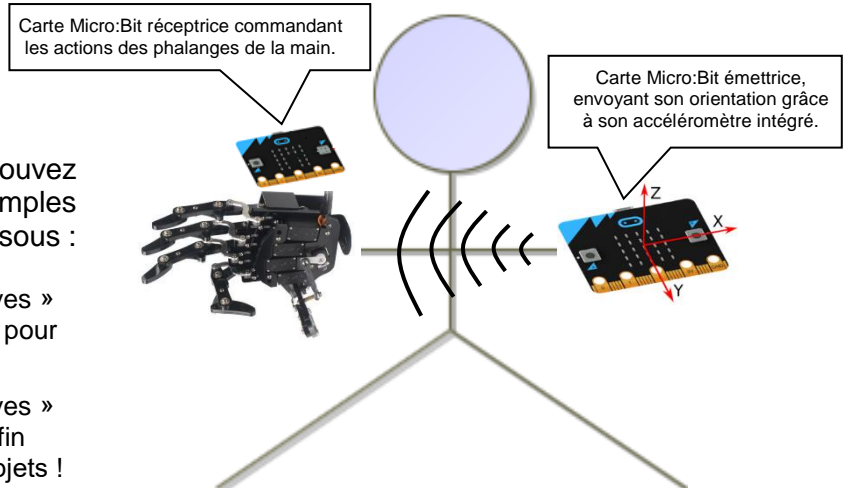


**Accéléromètre** — Tutoriel micro:bit UCL BBC ([microbit-challenges.readthedocs.io](https://microbit-challenges.readthedocs.io))  
**Boussole** — Tutoriel micro:bit UCL BBC ([microbit-challenges.readthedocs.io](https://microbit-challenges.readthedocs.io))

Le principe de fonctionnement de la solution envisagée est représenté sur le schéma ci-contre.

Dans un premier temps, vous pouvez vous limiter à des commandes simples comme celles proposées ci-dessous :

- ☒ Ouverture & fermeture « progressives » de toutes les phalanges de la main pour une saisie « grossière » d'objets,
- ☒ Ouverture & fermeture « progressives » de l'index et du pouce seulement afin d'assurer la préhension de petits objets !



Pour davantage de facilité, vous adopterez les emplacements (slot/broches) ci-contre pour le positionnement des servomoteurs de commande des différents doigts.

```

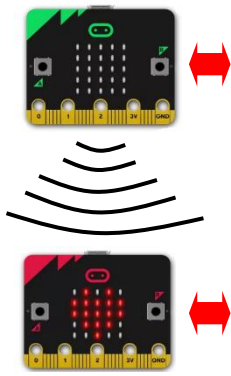
Mode Nouveau Charger Enregistrer Flasher Fichiers REPL Graphique Zoomer Dé-zoomer Thème Vérifier Aide Quitter
Two_finger.py | Alum_led.py | One_finger.py | Commande_Main.py | US_Evite_obstacle.py | 1_servo.py | Robot_Maqueen_Evit_Obst.py | Servo_BP.py |

1 from microbit import *
2 pin0.set_analog_period(20) # Pouce en slot/broche 0
3 pin1.set_analog_period(20) # Index en slot/broche 1
4 pin8.set_analog_period(20) # Majeur en slot/broche 8
5 pin12.set_analog_period(20) # Annulaire en slot/broche 12
6 pin16.set_analog_period(20) # Auriculaire en slot/broche 16

```

**Radio** — Tutoriel micro:bit UCL BBC ([microbit-challenges.readthedocs.io](https://microbit-challenges.readthedocs.io))

Ci-contre, exemple de création d'une communication entre 2 cartes Micro:Bit !  
Inspirez-vous en pour élaborer votre solution ...



```
# Programme à implanter dans la carte micro:bit n°1.
from microbit import *
import radio # Importation des fonctions liées à la communication "radiofréquence".
radio.config(group=1, power=7) # Attribution du module radio pour le groupe (ou réseau) n°1,
                                # Puissance d'émission maximale (1 mini - 7 maxi).
radio.on() # Activation du module radio.
while True:
    if button_a.was_pressed():
        radio.send("A") # Méthode pour l'envoi d'une donnée, ici la chaîne de caractère A.
    if button_b.was_pressed():
        radio.send("B") # Méthode pour l'envoi d'une donnée, ici la chaîne de caractère B.
```

```
# Programme à implanter dans la carte micro:bit n°2.
from microbit import *
import radio # Importation des fonctions liées à la communication "radiofréquence".
radio.config(group=1, power=7) # Attribution du module radio pour le groupe (ou réseau) n°1,
                                # Puissance d'émission maximale (1 mini - 7 maxi).
radio.on() # Activation du module radio.
while True :
    message = radio.receive() # Méthode pour la réception d'une donnée, ici message.
    if message == "A":
        display.show(Image.HAPPY)
    if message == "B":
        display.show(Image.SAD)
```

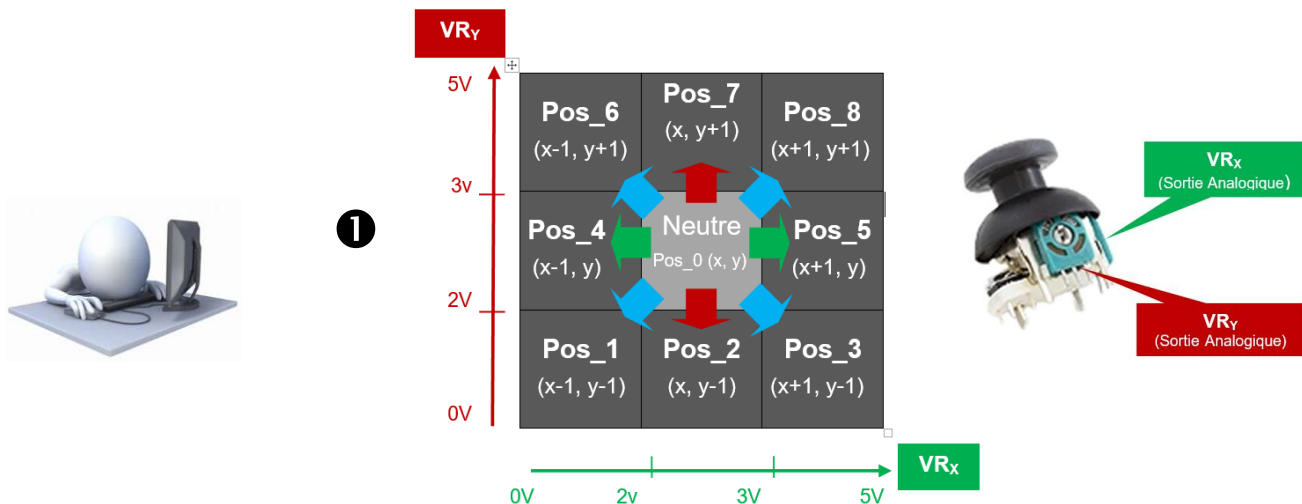
6.1/. Donner libre cours à votre imagination en élaborant **deux programmes** (pour chaque carte Micro:Bit), permettant de respecter le cahier des charges évoqué précédemment, pour piloter à cette main !

A noter que **deux algorithmes simples** doivent être préalablement produits pour mettre en évidence le rôle de chaque programme.



**BONUS/. « Autres solutions de commande possible de la main ... »**  
(pour les étudiants et groupes en avance)

Utilisation d'un joystick :



Utilisation d'un « smartphone » sous android avec l'application App Inventor :





## 7/. Problématique n°3 : « Assurer la préhension avec une gestuelle humaine »

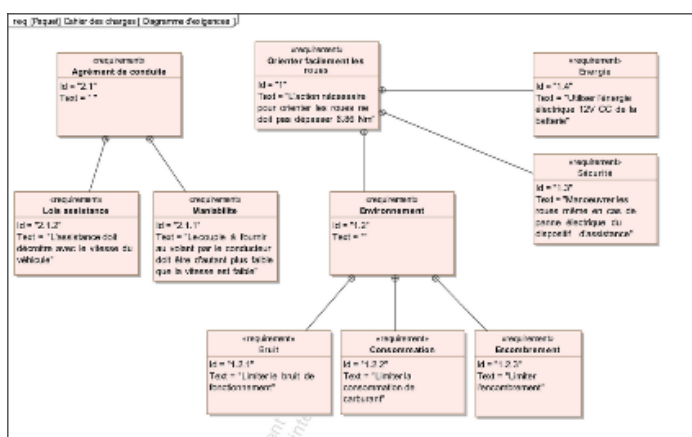
### 7.1/. Quels gestes souhaitez-vous reproduire ?


Pour cela réfléchissez par exemple aux gestes essentiels de votre vie, chez vous, au lycée, ... Sélectionnez les gestes les plus importants, ceux dont vous ne pourriez-vous passer. Puis à l'aide des documents ressources et d'une recherche bibliographique, précisez quels gestes de la vie courante nécessiteraient une amélioration prioritaire.



{Séance 1 : 1h00}

### 7.2/. Développer les améliorations précédemment proposées en définissant pour chacune d'elles deux à quatre exigences que vous représenterez sur un diagramme SysML.



 Afin de représenter votre diagramme vous pouvez utiliser le site [draw.io](https://draw.io). Après avoir choisi un répertoire d'enregistrement de vos fichiers allez dans UML puis uml/sysml.xml.

### 7.3/. Vérifier que les différents gestes et améliorations proposés ci-dessus sont réalisables par la main « robotisée ». Dans un tableau, indiquer les gestes réalisables par la main et ceux qui ne le sont pas !

Ce tableau pourra éventuellement comprendre un deuxième exemple de main « robotisée » documenté lors de vos recherches bibliographiques.

Exemple de tableau possible :

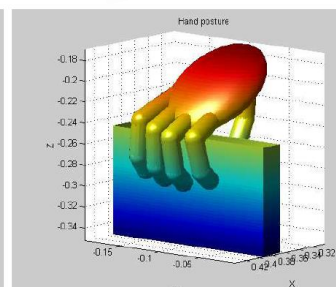
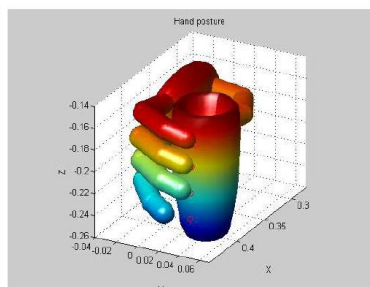
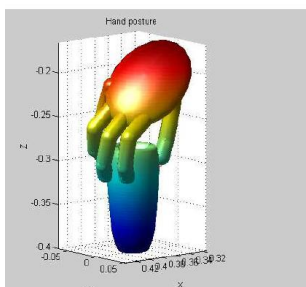
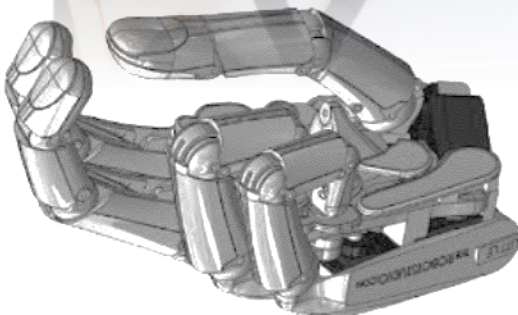
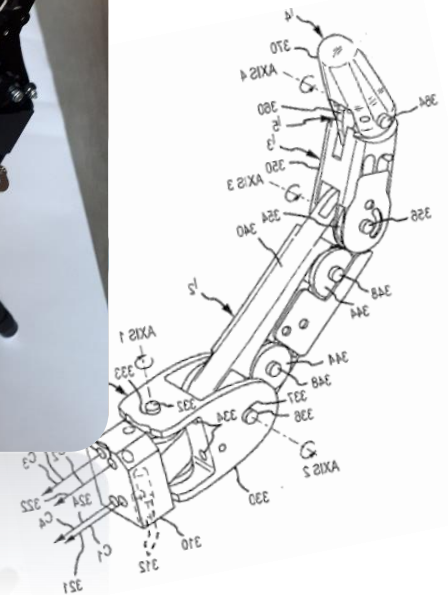
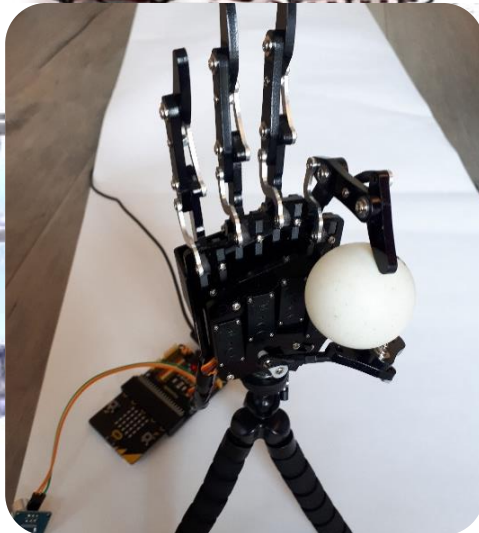
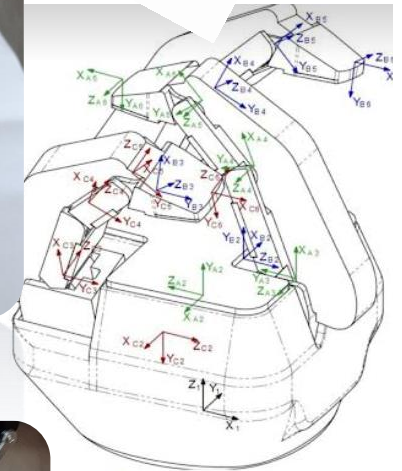
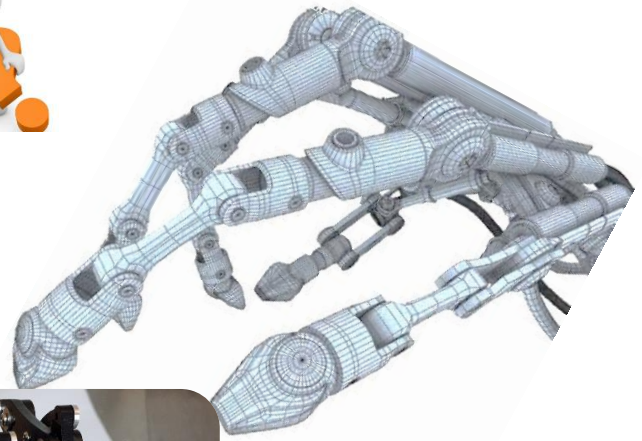
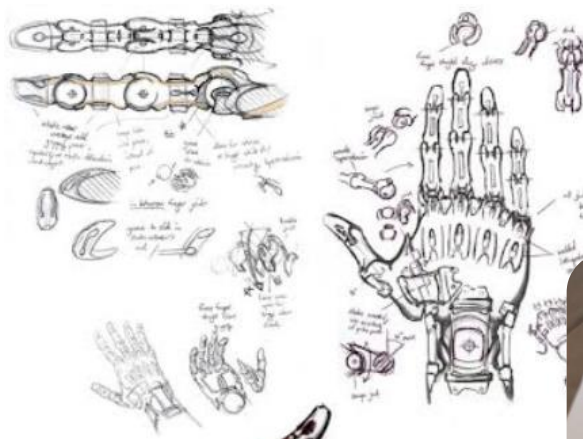
	Gestes réalisables	Gestes non réalisables	Propositions d'améliorations
Main « robotisée » réelle			
Autre « main robotisée »			

{Séance 2 : durée estimée : 1h00 + travail personnel}



Exemples de gestes « de préhension »:

Soyez imaginatifs !





**7.4/.** Etude de conception : cette étude vise à **améliorer la main « robotisée »** en lui permettant de réaliser un ou plusieurs gestes non réalisables en atteignant les exigences que vous avez définies.

Les tâches suivantes seront réparties entre les membres du groupe :

1. **Définir** des liaisons cinématiques,
2. **Mesurer** la vitesse de préhension,
3. **Choisir** les matériaux,
4. **Définir** les dimensions des composants du projet,
5. Et enfin, **modéliser** le projet sur Solidworks.



Les éléments à produire seront :

- Une **maquette numérique** comprenant vos améliorations,
- Un **document** expliquant votre démarche et décrivant votre réalisation.

{Séance 2 : 1h + Séance 3 : 2h00 + travail personnel}

**7.5/.** « Phase de Prototypage » : **Réalisation d'une maquette réelle** intégrant les améliorations mises en œuvre sur la main « robotisée ».

A noter que les modélisations issues de la conception doivent vous permettre de réaliser un prototype de votre projet et donc de :

1. **Fabriquer** des composants,
2. **Assembler** les composants entre eux et sur la main robotisée,
3. **Tester et ajuster** le prototype : liaisons, capacité et vitesse de préhension,
4. Et enfin, **valider**.

Les éléments à produire seront :

- La réalisation réelle des composants et leur assemblage,
- Ainsi qu'une vidéo des « nouveaux » gestes permis par votre reconception de la main « robotisée ».

{Séance 4 : 2h00 + travail personnel}

Matériel à disposition :

- Maquette numérique SW de la main « robotisée »,
- Pied à coulisse, comparateur, ...
- Machine de découpe laser : découpe des panneaux en bois ou PMMA jusqu'à 10mm d'ép.
- Imprimante 3D,
- Dynamomètre,
- Quincaillerie : petit matériel, vis, écrous, ...
- Tout matériel disponible en salle de spé SI.

