



### Séance 3 : Simulation du fonctionnement d'une serrure connectée

La modélisation en 3D d'une maquette de serrure associée et la simulation des mouvements permettent de mieux comprendre le fonctionnement de cette partie de la chaîne de puissance. De même, la modélisation de la partie électrique et sa simulation permettent la mise au point du programme. L'ensemble des modèles peut permettre la détermination de certains paramètres qui pourrait être vérifiés ensuite (mesure des écarts) sur la maquette réelle.

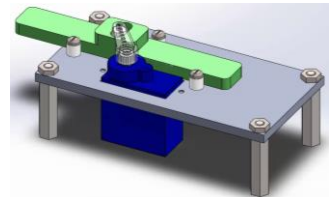
Plan de la séance 3	Objectifs d'apprentissages
<ul style="list-style-type: none"> <li>Analyser la vidéo d'animation d'une maquette de serrure – identifier les fonctions des éléments (chaîne de puissance) et les types de mouvements mis en jeu.</li> <li>Analyser un modèle simplifié de la maquette de la serrure et identifier les fonctions des éléments (chaîne d'information)</li> <li>Identifier et corriger une erreur dans la partie initialisation du programme grâce à la simulation.</li> <li>Compléter et mettre au point et vérifier le comportement de la simulation de la serrure.</li> </ul>	<ul style="list-style-type: none"> <li>Notion de diagramme d'états</li> <li>Notion de chaîne de puissance et d'information</li> <li>Notion de modélisation et de simulation</li> <li>Notion de programmation événementielle</li> <li>Identification de paramètres sur une modélisation 3D</li> </ul>

#### Partie 1 : Analyse de ma maquette

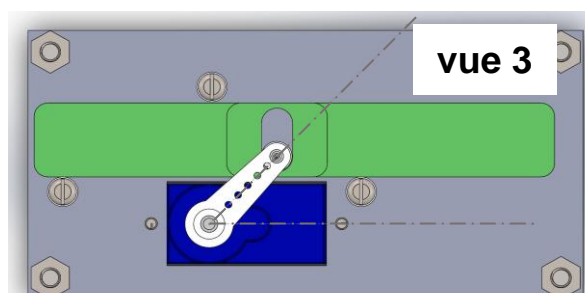
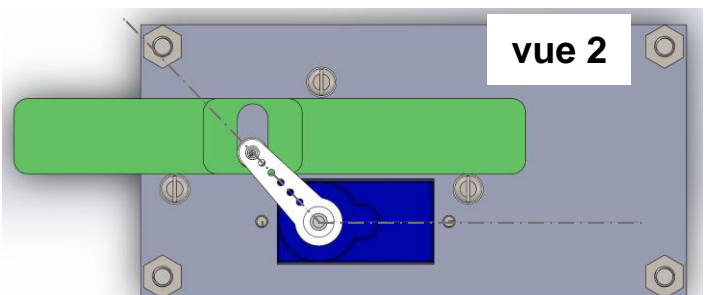
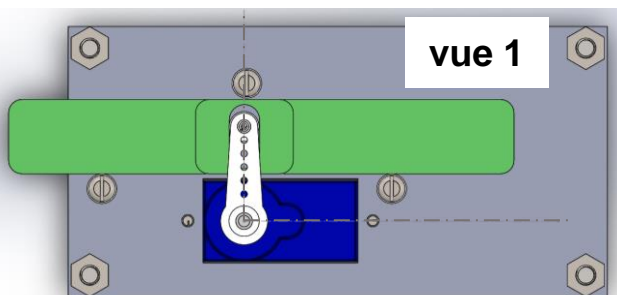


1. Ouvrir la vidéo *Animation serrure connectée.mp4* qui montre le fonctionnement d'une maquette représentant partiellement et de façon simplifiée le fonctionnement possible d'une serrure connectée puis répondre aux questions suivantes :

- Quel élément possède un mouvement de rotation ?
- Quel élément de la serrure possède un mouvement de translation ?
- Quel élément appartient au bloc fonctionnel « convertir » de la chaîne de puissance ?
- Quel(s) élément(s) apparten(en)t au bloc fonctionnel « transmettre » ?
- En utilisant le vocabulaire de la vidéo, explique le fonctionnement de la serrure.



2. Parmi les vues ci-dessous, laquelle correspond à la position porte déverrouillée ? à la porte verrouillée ? à la serrure en cours de fonctionnement ?





### Partie 2 : Programmation et simulation de la maquette

- Se connecter à  **votre compte classe** Tinkercad grâce au lien ([ou au code](#)) donné par le professeur

puis .

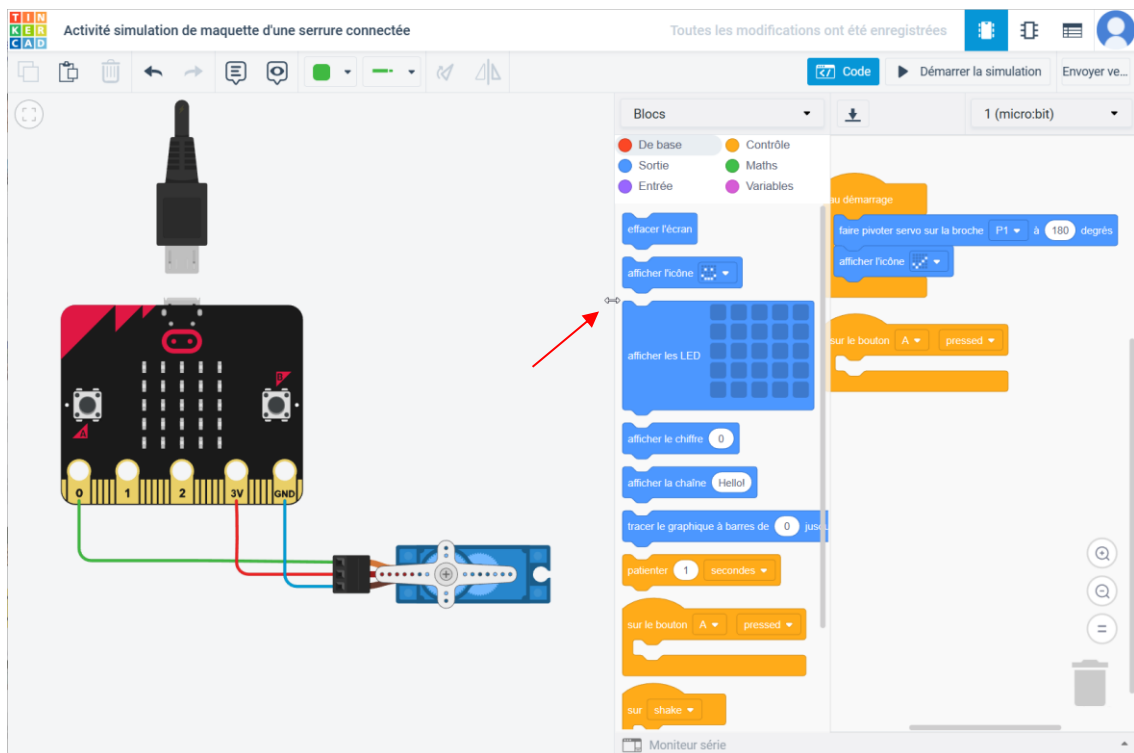
- S'identifier en cliquant sur puis saisir votre code personnel donné

par le professeur



- Ouvrir l'activité de programmation et de simulation [en lien ici](#) puis cliquer sur « **Copier et Editer** ».

L'environnement de programmation et de simulation ci-dessous doit s'ouvrir. Accéder à la zone codage en cliquant sur en haut. L'élargissement de cette zone est aussi possible avec la souris en déplaçant la bordure (voir flèche rouge).



La partie électrique de la maquette de serrure connectée est modélisée de façon simplifiée. Le servomoteur est connecté directement à une carte Micro:bit. En réalité, l'alimentation du servomoteur n'est pas réalisée par le 3,3 V insuffisant de la carte Micro:bit mais c'est une simplification du modèle sans conséquence ici. Les boutons A et B permettront de commander la serrure.

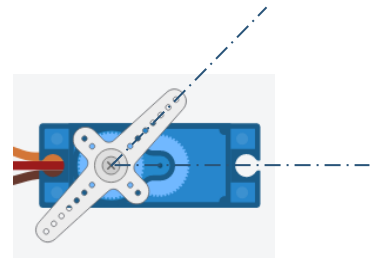
#### Répondre à la question suivante :

- Quels sont les blocs fonctionnels de la chaîne d'information assurés ici par la carte Micro:bit ?



2. La simulation de la partie initialisation du programme proposée est testable en cliquant sur « Démarrer la simulation » en haut.

- **Tester ce début de programme.** L'affichage fonctionne bien mais le servomoteur ne tourne pas. En observant attentivement le câblage proposé, **trouver l'erreur dans le programme, la corriger et tester de nouveau.**
- Après avoir modifié et testé plusieurs valeurs d'angle de rotation dans le programme entre 0° et 180°, **représenter** par un arc de cercle sur la vue ci-contre **l'angle correspondant à celui programmé.** L'angle sera nommé  $\alpha$ .
- A l'aide des vues de la question 2 et d'un rapporteur, **déterminer la valeur de l'angle correspondant à la serrure en position verrouillée.**



- **Déterminer la valeur de l'angle correspondant à la serrure en position déverrouillée.**

- **Expliquer** pourquoi il ne faut pas que le programme demande un angle plus grand que celui déterminé ci-dessus au servomoteur pour la porte verrouillée ? S'aider de la vidéo de l'animation si nécessaire.

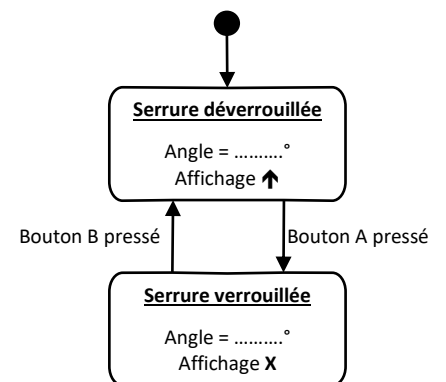


Diagramme d'état de la maquette

- **Reporter les angles dans le diagramme d'états** de la serrure ci-contre.
- D'après le diagramme, dans quel état doit être la serrure au démarrage du programme ?

- Le programme décrit par ce diagramme d'états a-t-il une fin ?

- **Corriger et compléter le programme de la serrure** en respectant l'algorithme décrit par ce diagramme d'états.

Le programme sera réussi :

- si la serrure se met dans le bon état au démarrage ;
- si elle réagit bien aux appuis sur les boutons (angles et affichages corrects).



Les blocs manquants se trouvent dans le menu « de base ». Il est également possible de dupliquer un bloc par un clic droit.

- D'après le type de blocs orange ajoutés dans le programme constitué, s'agit-il d'une programmation de type :  événementielle  séquentielle
- Insérer ci-dessous une copie d'écran de ton programme.
- Bilan de séance à recopier :



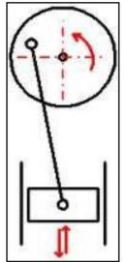
### Ce que je dois retenir

#### Notion de type de mouvements (rotation - translation)

Dans un mécanisme, certains organes (éléments) peuvent être en mouvement. Il existe **deux types de mouvement de base** : la translation et la rotation.

- Dans une **translation**, l'organe suit une **trajectoire rectiligne** qui sera symbolisée par une flèche droite.
- Dans une **rotation**, l'organe suit une **trajectoire circulaire** qui sera symbolisée par une flèche en arc de cercle.

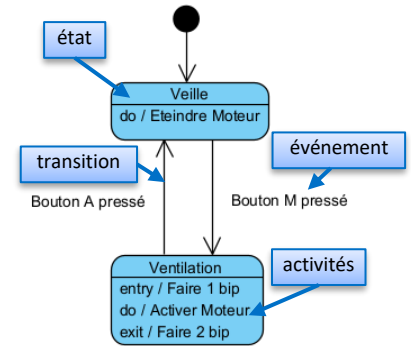
Tout autre mouvement résulte de la **combinaison** de ces deux mouvements.



#### Notion de diagramme d'états

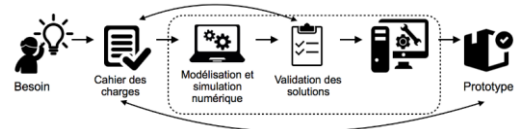
Il permet **modéliser le comportement** d'un système en décrivant **les différents états** du système (ou d'un bloc) en fonction des **événements** qui lui arrivent.

- Les différents états du système sont représentés par des rectangles au bord arrondi.
- Le **cercle noir** indique l'état initial. ●
- Les flèches représentent les **transitions** possibles entre les états.
- Pour passer à l'état suivant, **l'état précédent doit être actif** et **l'événement déclencheur** de la transition survient (ex : *bouton A pressé*).
- **L'activité** pendant l'état peut être décrite (entry (à l'entrée dans l'état) / do (activité durable) / exit (à la sortie))



#### Notion de modélisation et de simulation

La **modélisation** et la **simulation** numérique interviennent au moment de la conception et de la validation de solution. Elles permettent de **comprendre le fonctionnement** d'un système, de formaliser, présenter et partager des recherches mais aussi d'investiguer, **concevoir et trouver des solutions** pour les fabriquer et construire.



#### Notion de programmation événementielle

Un **événement** permet de déclencher un ensemble d'opérations qui correspondent à un algorithme.

Un **événement est une modification de l'état du système** (mouvement de souris, touche de clavier, message envoyé par un autre script, ...). On parle alors de programmation événementielle.

*Exemple ci-contre : Impossible de prévoir le résultat (jusqu'ou et si le lutin va s'arrêter...) car il dépend d'un événement extérieur.*

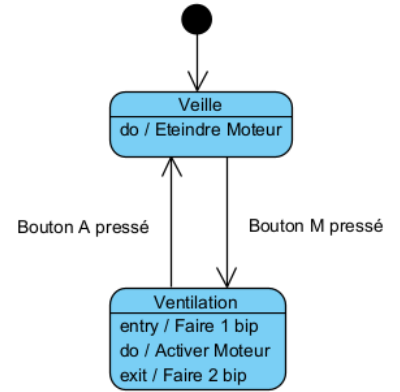




### Exercices

1- Pour le **diagramme d'état** ci-contre, cocher les propositions vraies :

- Le système comporte 2 états
- Au démarrage, le système active sa ventilation
- Le système n'a pas d'état final
- Pour passer de l'état Ventilation à l'état Veille, il faut appuyer sur le bouton A
- On entend 1 bip après appui sur le bouton M si on était en veille.
- On entend 2 bips après appui sur le bouton M si on était en veille.
- On entend 2 bips après appui sur le bouton A si on était en ventilation.



2- Laquelle des 2 programmations ci-dessous est une programmation événementielle ? Laquelle est une programmation séquentielle ?



3- A l'aide des blocs ci-dessous, programmer le diagramme d'état de la question 1. Ecrire sur papier le programme en pseudo-code.

