



# Premier pas dans tinkercad

<b>I. PREMIERS PAS DANS TINKERCAD.....</b>	<b>2</b>
1. INSCRIPTION.....	2
2. PREMIER CIRCUIT VIRTUEL.....	3
3. AJOUT DE COMPOSANTS.....	5
4. LES FILS DE LIAISON « JUMPER » .....	5
5. QUELQUES « TRUCS » .....	6
a. Cacher les composants .....	6
b. Si vous n'avez l'accès à la programmation par texte : .....	7
c. Pour faire disparaître l'écran de code : .....	7
d. Pour zoomer sur tout le montage .....	7
6. FIN DE LA PREMIERE SIMULATION .....	8
a. Fin du câblage .....	8
b. Code Arduino.....	8
c. Simulation .....	8
7. PARTAGE DE VOTRE SIMULATION .....	8
<b>II. EXERCICES AVEC LED ET BP.....</b>	<b>10</b>
1. CLIGNOTER PENDANT QUE L'ON PRESSE .....	10
2. ALLUMER 2 SECONDES QUAND ON PRESSE .....	10
<b>III. UN EXEMPLE D'UTILISATION : CAPTEUR A ULTRASONS.....</b>	<b>11</b>
1. LA THEORIE .....	11
a. Caractéristique du capteur.....	11
b. Calcul de la distance de l'obstacle 1 <sup>ère</sup> approximation .....	12
c. Avec plus de précisions .....	12
2. SIMULATION DANS TINKERCAD.....	13
a. Montage à réaliser .....	13
b. Capteur ultrasons par défaut.....	13
c. Trouver le capteur à ultrasons HC-05.....	14
d. Utiliser le capteur .....	15
e. Utilisation du port série .....	16
f. Code Arduino.....	17

## Description

C'est T.D. rédigé en « urgence » pour les élèves de 1<sup>ère</sup> année de BTS SN-EC du lycée E. Branly d'Amiens et donné cette semaine, il reste donc sans doute des problèmes de formatage ou des fautes d'orthographe.

Cette séance introduit l'utilisation du site [www.tinkercad.com](http://www.tinkercad.com) permettant notamment de simuler une carte Arduino.

Certaines corrections sont données en texte masqué. Si vous ne les voyez pas, il faut permettre l'affichage de texte caché : *Fichier->Option* puis l'onglet *Affichage* et cocher *Texte masqué*

Merci pour vos retours : [tomczak.branly@outlook.fr](mailto:tomczak.branly@outlook.fr)



## I. PREMIERS PAS DANS TINKERCAD

### 1. inscription

Allez sur <https://www.tinkercad.com/>

Cliquez sur Join Now



Puis sur Create a personal account (si c'est en anglais)

### Start Tinkering

How will you use Tinkercad?

In school?

[Educators start here](#)

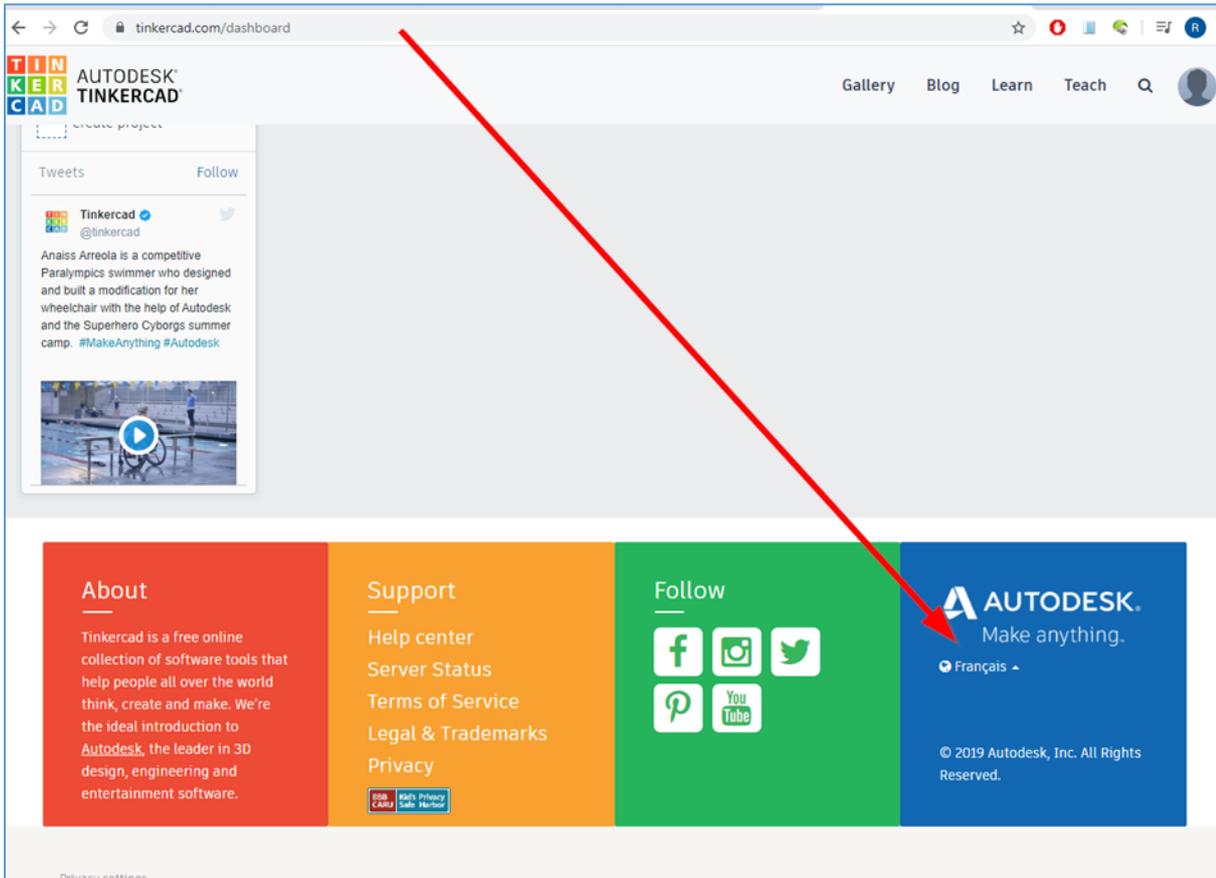
[Students, join a Class](#)

On your own

[Create a personal account](#)

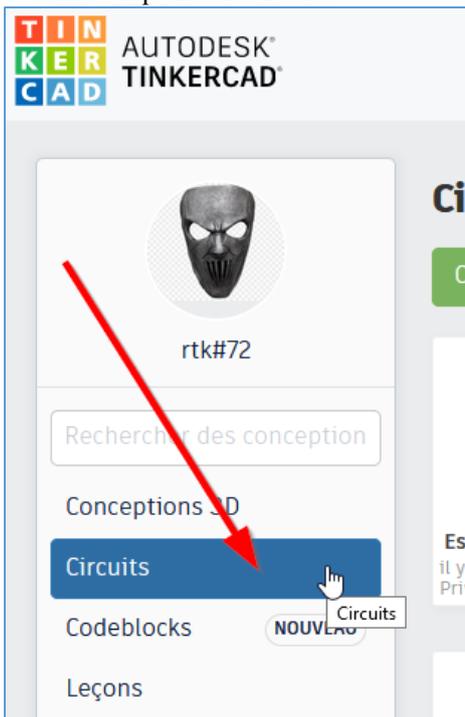
Already have an account?  
[Sign In](#)

Si ce n'est pas en français :



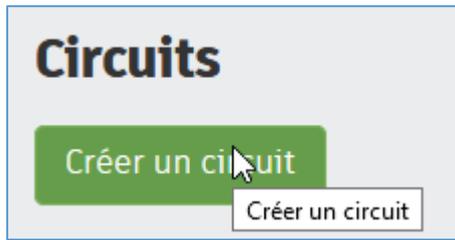
## 2. Premier circuit virtuel

 Cliquez sur Circuits

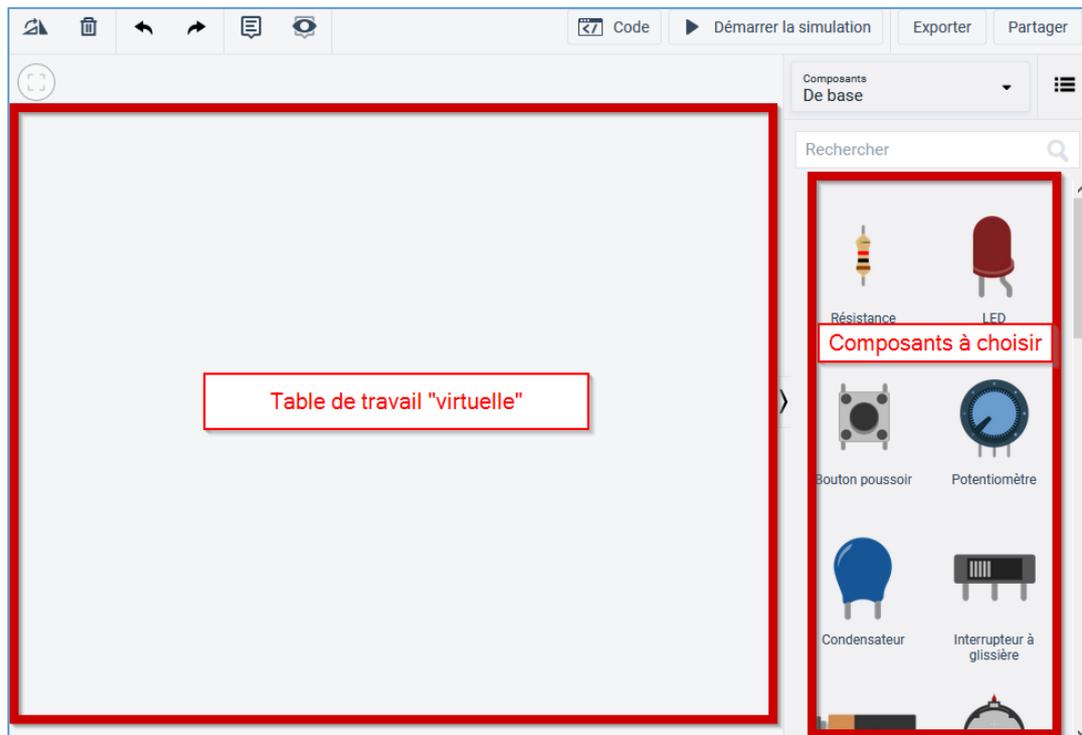




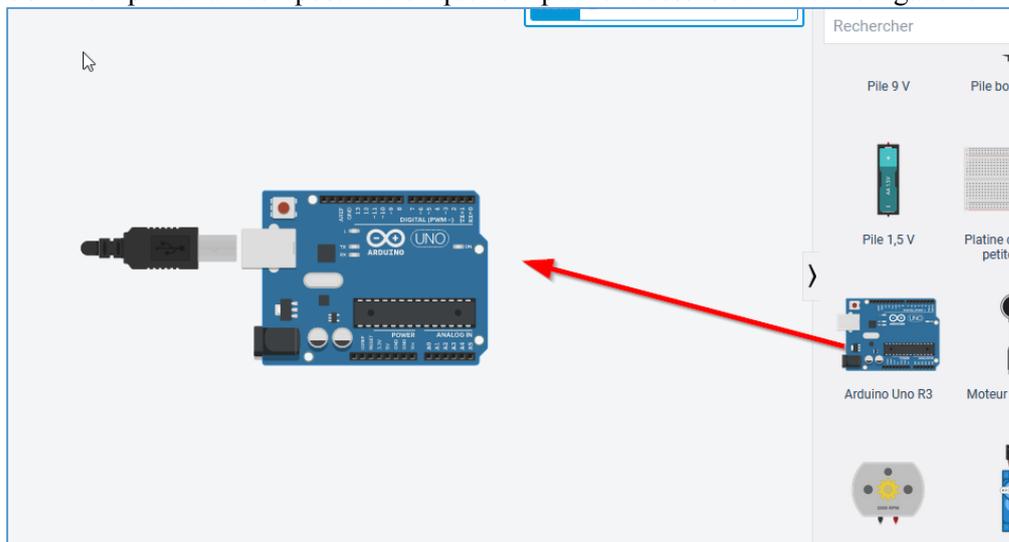
Puis Créer un circuit



L'écran est séparé en deux : une table virtuelle et des composants



Comment placer un composant ? Cliquez simplement dessus et mettez-le à gauche

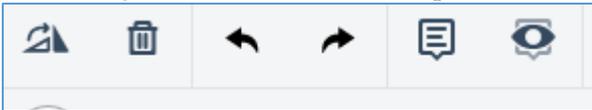


### 3. Ajout de composants

Ajoutez :

- Une platine
- Une led
- Une résistance

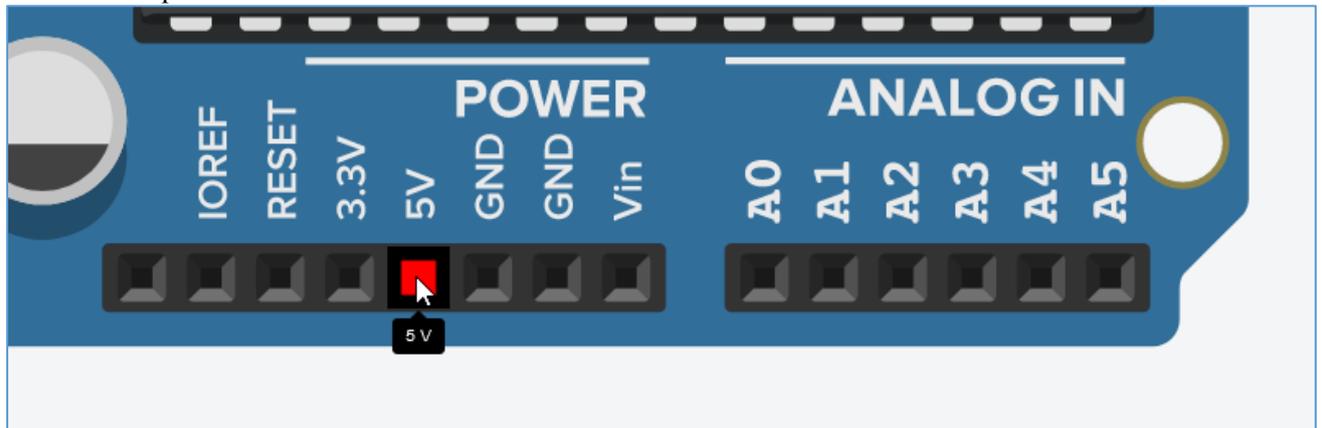
En haut à gauche se trouve un menu permettant de manipuler les composants



Zoomez avec la molette

### 4. Les fils de liaison « jumper »

Il suffit de déplacer sa souris sur une broche et une extrémité du câble s'affiche

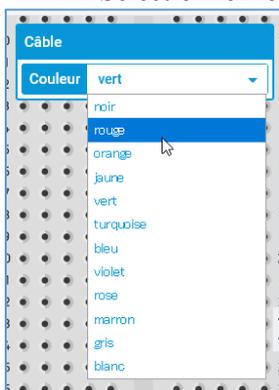


Cliquez un câble s'affiche

Refaites les mêmes manipulations pour l'autre extrémité

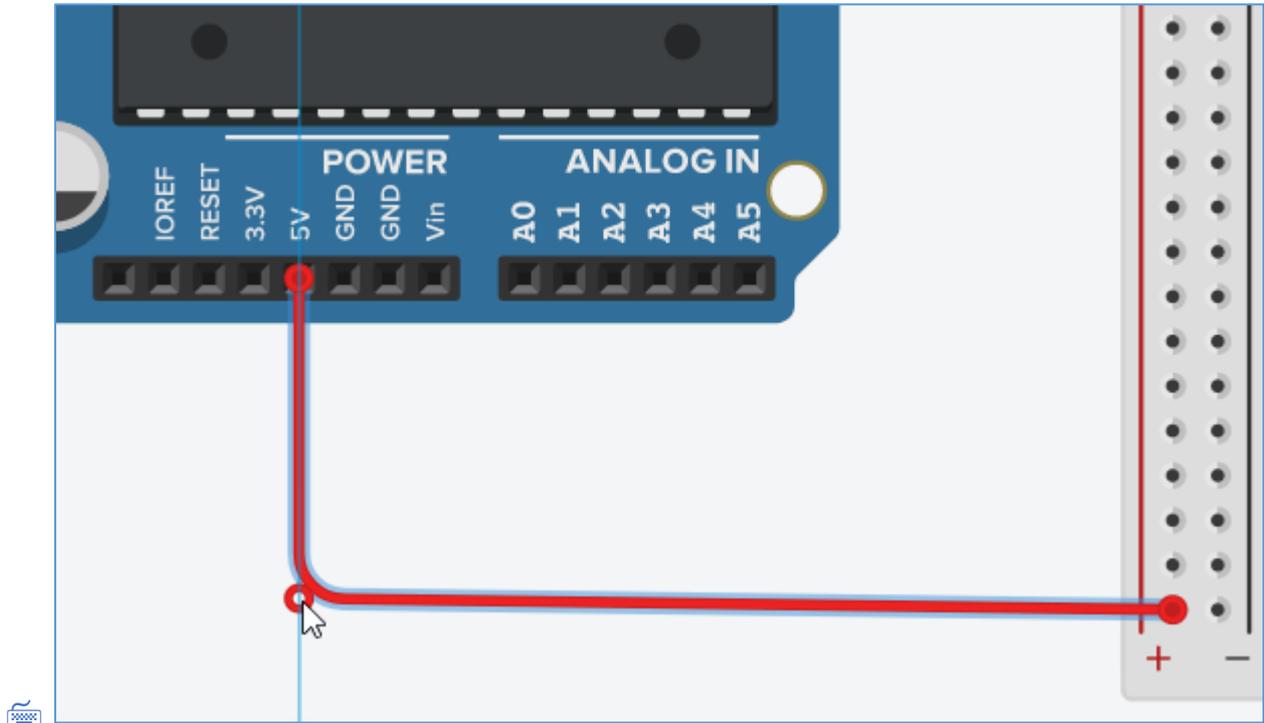
Pour changer la couleur :

Sélectionnez le câble puis la couleur



Pour ajouter des points de courbure :

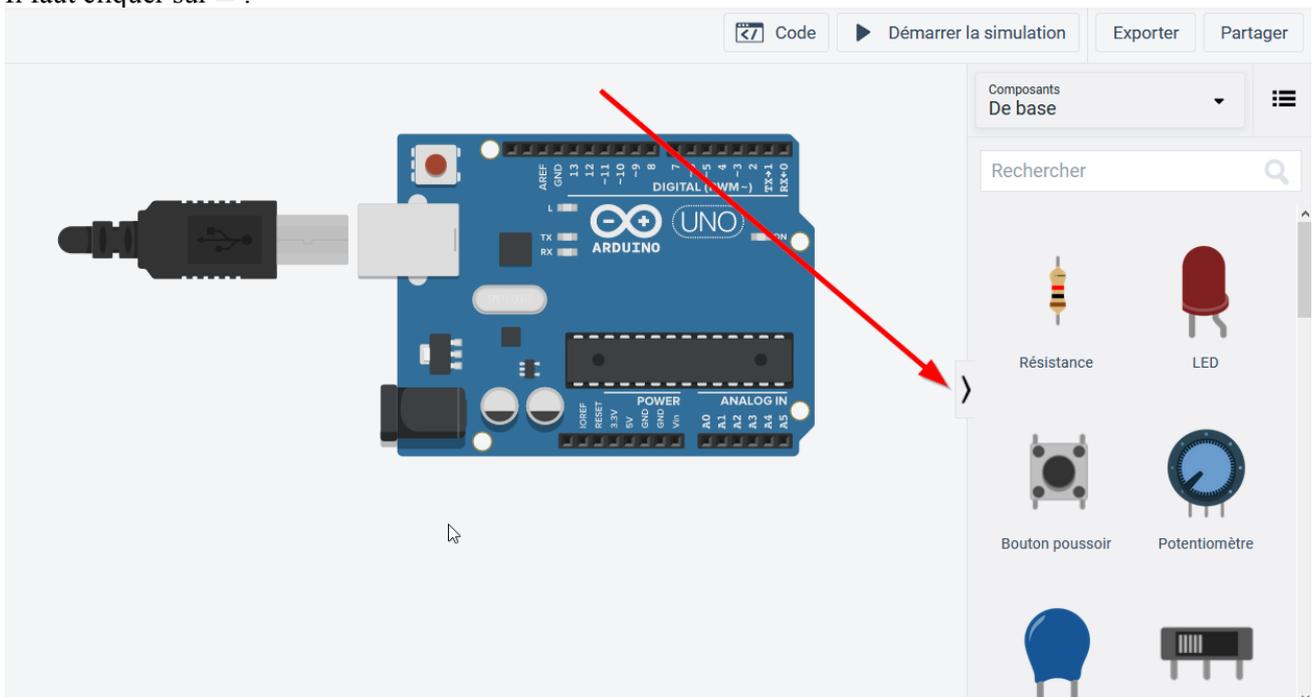
Cliquez deux fois sur le jumper



## 5. Quelques « trucs »

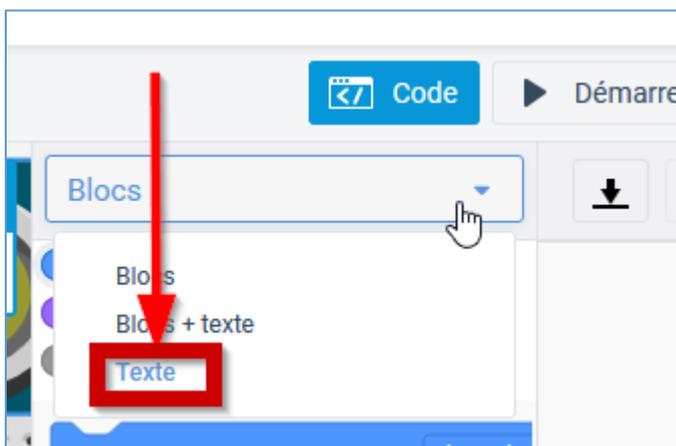
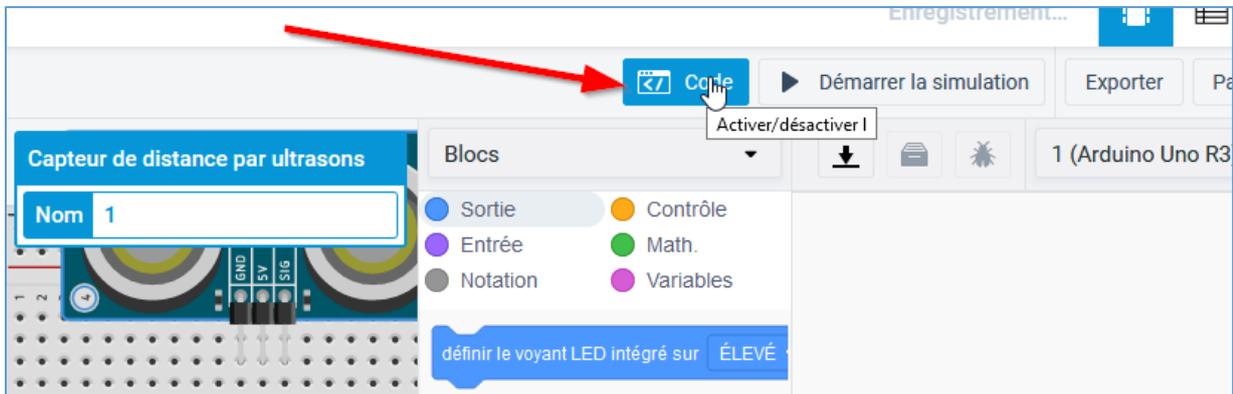
### a. Cacher les composants

Il faut cliquer sur > :



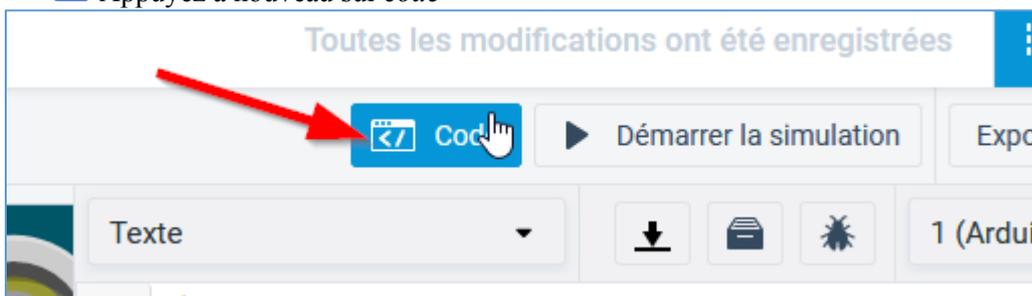


**b. Si vous n'avez l'accès à la programmation par texte :**

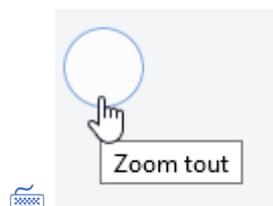


**c. Pour faire disparaître l'écran de code :**

Appuyez à nouveau sur *code*



**d. Pour zoomer sur tout le montage**





## 6. Fin de la première simulation

### a. Fin du câblage

🔧 Câblez correctement la led sur le port 8 (avec la bonne valeur de résistance et le bon sens de la led)

### b. Code Arduino

🔧 Ecrivez le code permettant de la faire clignoter

### c. Simulation

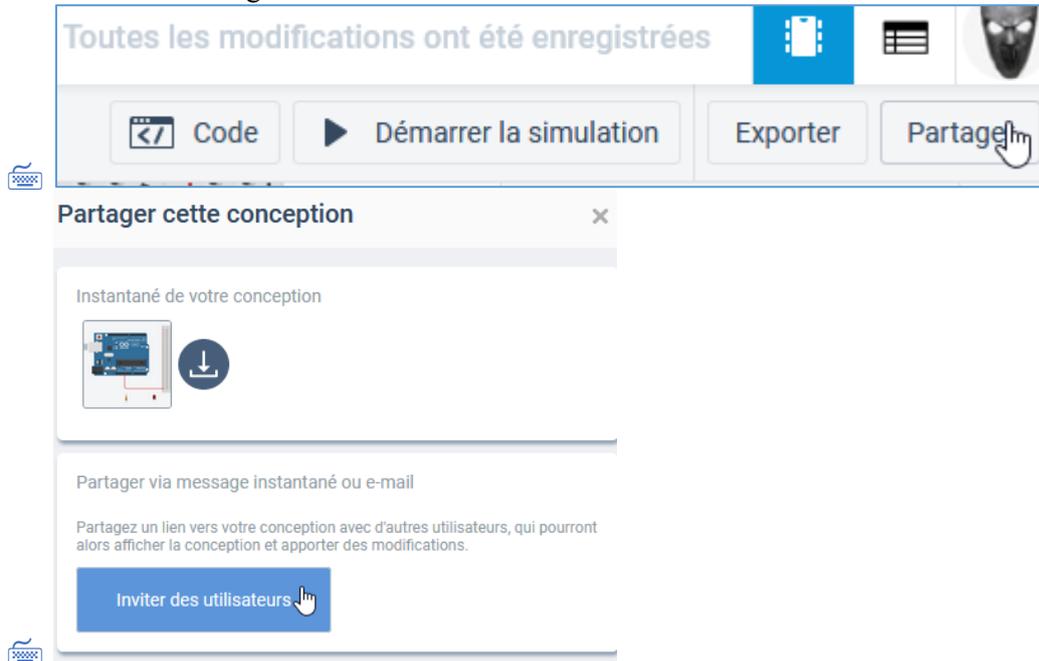
🔧 Démarrer la simulation :

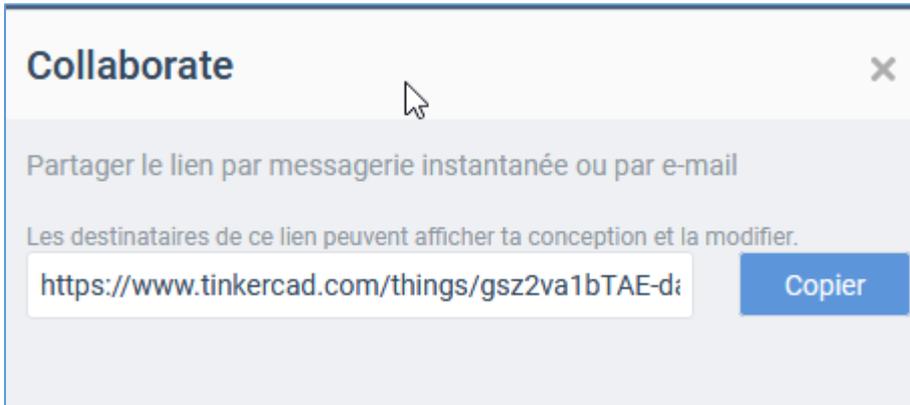
toutes les modifications ont été enregistrées



## 7. Partage de votre simulation

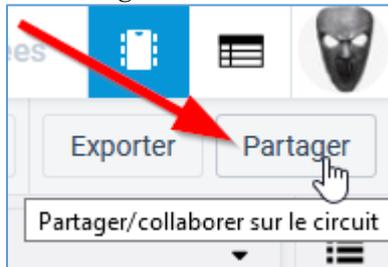
🔧 Choisissez Partager et fournissez le lien





En cas de problème de partage, vérifiez que votre projet soit bien en public :

Partager



Puis modifiez la confidentialité du partage :

Nom de la conception  
HC05

Description de la conception  
Donne à tes utilisateurs quelque chose à commenter. Ajoute une courte description de ta conception.

Balises (10 maximum)  
Entre la ou les étiquettes ici, en les séparant par des virgules. Appuie sur

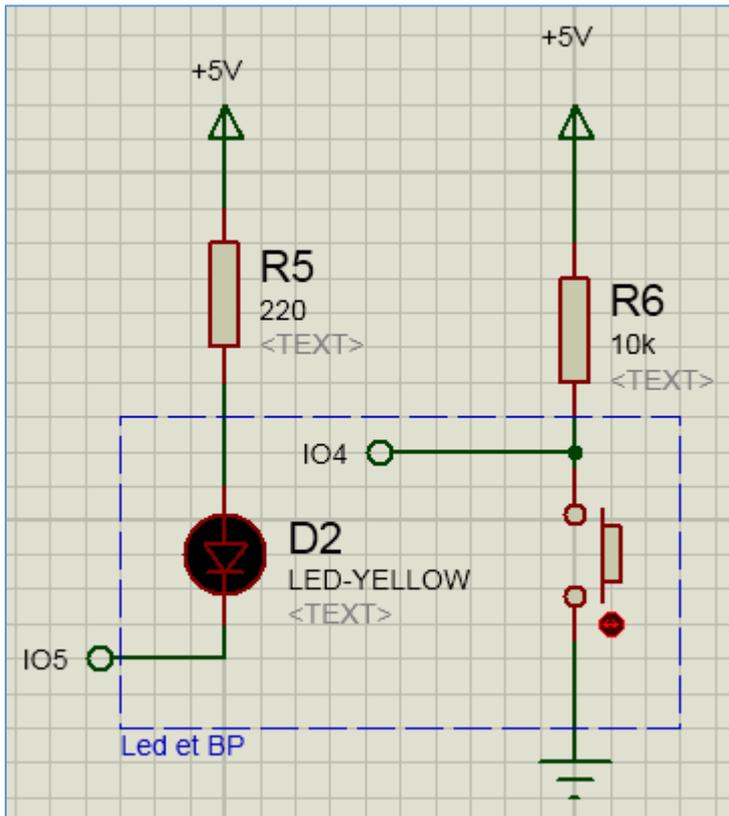
Confidentialité  
Privé Not publicly listed, visible only to you  
Public  
Public Domain

This license lets others remix, tweak, and build upon your work even for commercial purposes, for use with works that are already free of known licenced or copyright restrictions. Informations supplémentaires sur les licences Creative Commons



## II. EXERCICES AVEC LED ET BP

### 1. Clignoter pendant que l'on presse



📖 Consignes :

- Un poussoir actif a zéro est câblé sur la pin 4.
- Une Led sur la pin 5.
- Si on presse, la Led doit clignoter à 2 Hz (période 500 ms).

### 2. allumer 2 secondes quand on presse

📖 Consigne : si on appuie sur le bouton poussoir la led s'allume



## III. UN EXEMPLE D'UTILISATION : CAPTEUR A ULTRASONS

### 1. La théorie

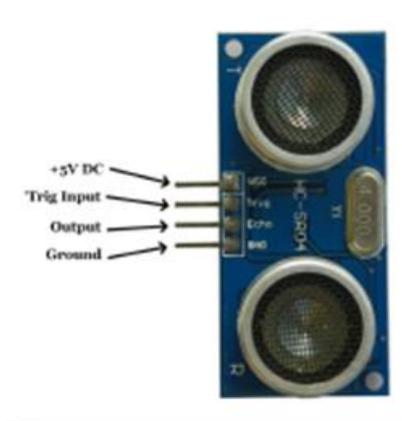
#### a. Caractéristique du capteur

(D'après : <http://itechnofrance.wordpress.com/2013/03/12/utilisation-du-module-ultrason-hc-sr04-avec-arduino/>)

Les caractéristiques techniques du module sont les suivantes :

- Alimentation : 5v.
- Consommation en utilisation : 15 mA.
- Gamme de distance : 2 cm à 5 m.
- Résolution : 0.3 cm.
- Angle de mesure : < 15°.

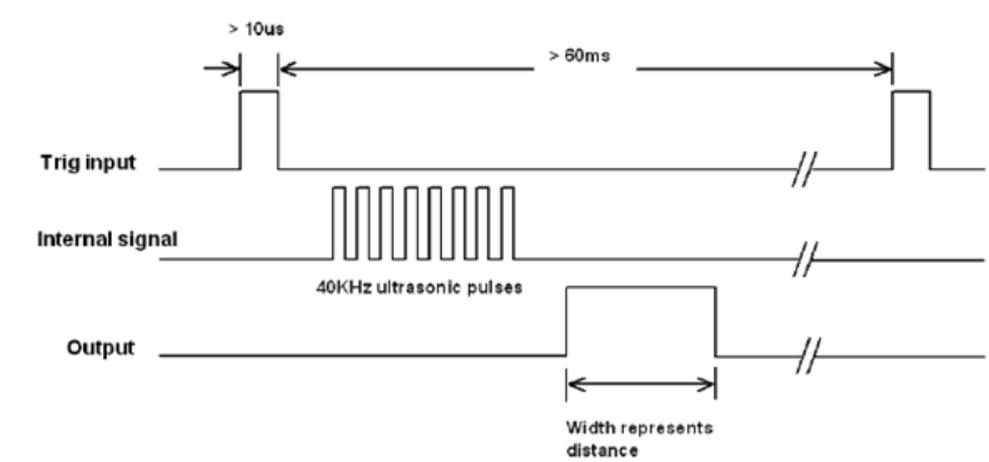
Le brochage du module est le suivant :

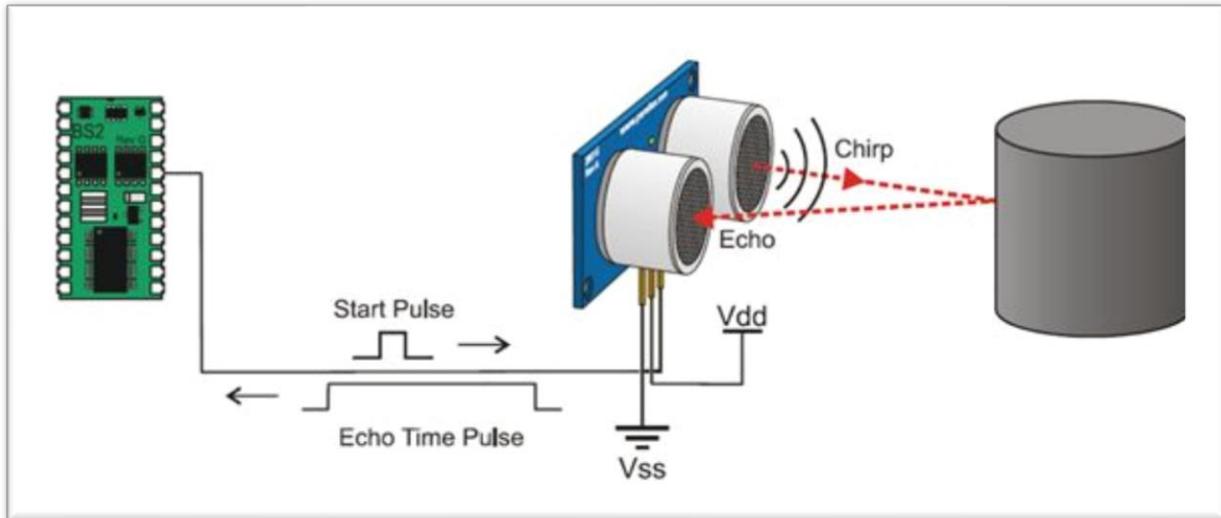


Le fonctionnement du module est le suivant :

Il faut envoyer une impulsion niveau haut (à + 5v) pendant au moins 10  $\mu$ s sur la broche 'Trig Input'; cela déclenche la mesure. En retour la sortie 'Output' ou 'Echo', va fournir une impulsion + 5v dont la durée est proportionnelle à la distance si le module détecte un objet.

Voici une représentation graphique de la séquence de fonctionnement du module :





<http://csharpcorner.mindcrackerinc.netdna-cdn.com/UploadFile/167ad2/how-to-use-ultrasonic-sensor-hc-sr04-in-arduino/Images/HC-SR04.jpg>

### b. Calcul de la distance de l'obstacle 1<sup>ère</sup> approximation

Soit  $T$  le temps que met le signal pour revenir au capteur : temps aller-retour de la sortie du capteur pour y revenir après avoir été réfléchi par l'obstacle.

Sachant que le son se propage à la vitesse  $C = 340\text{m/s}$ .

`pulseIn(echo, HIGH);` retourne le temps du retour en en microsecondes

### c. Avec plus de précisions

Pour plus de précision, modifiez la vitesse du son selon la température de l'air.

$C = (331,5 + 0,607 \cdot \theta)$  en m/s avec  $\theta$  en °C, l'hygrométrie influe peu.

Question N°3 : A l'aide du thermomètre (demander au professeur) calculer plus précisément la distance

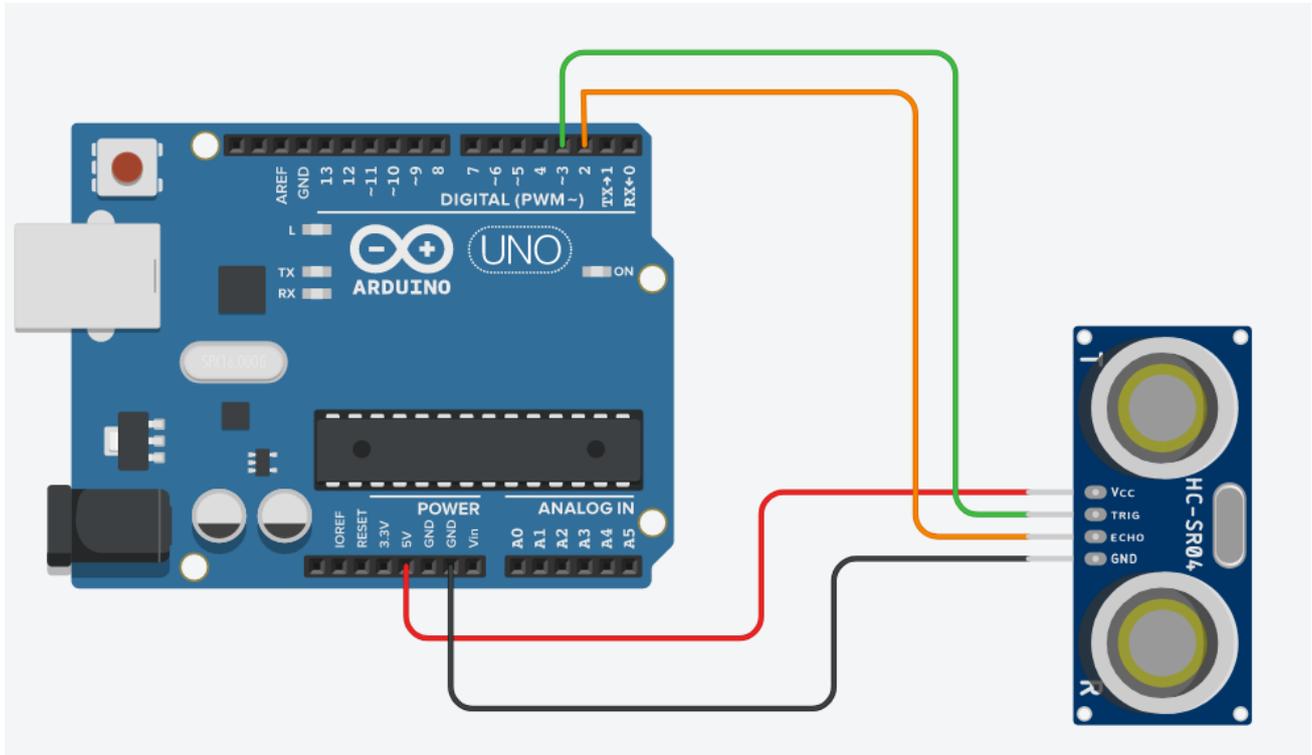
Distance (cm) = \_\_\_\_\_

Question N°4 : Vérifier cette formule en disposant un obstacle à un mètre et modifier  $\cdot\theta$  pour avoir exactement cette distance



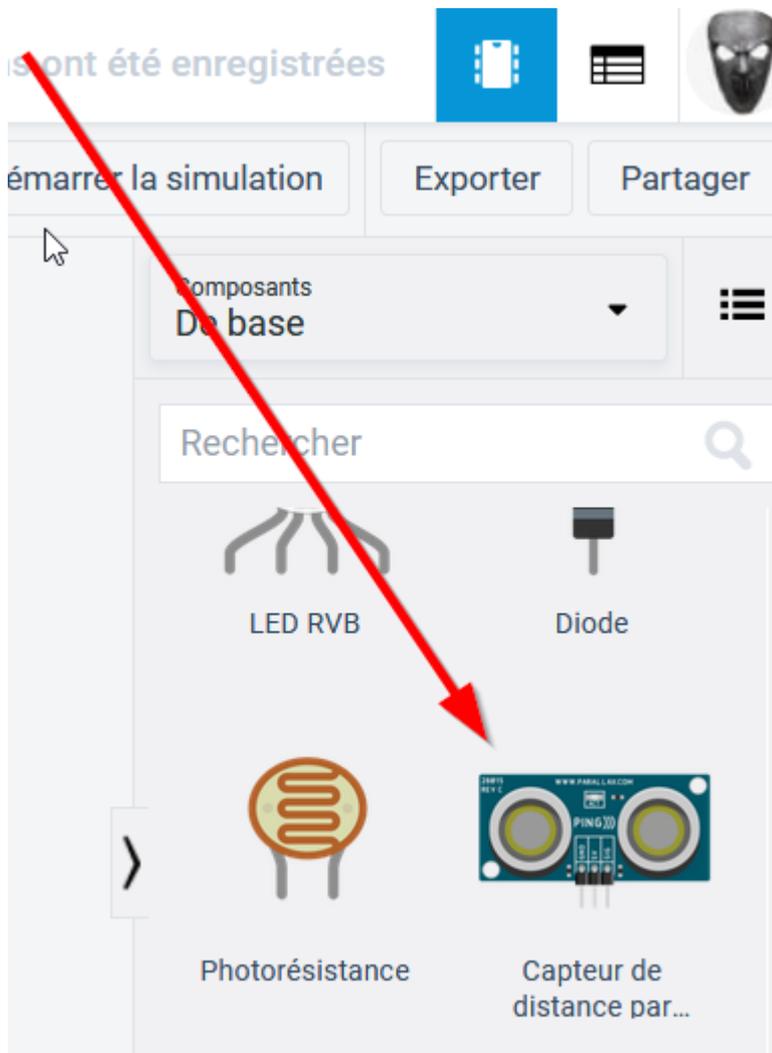
## 2. Simulation dans tinkercad

### a. Montage à réaliser



### b. Capteur ultrasons par défaut

Comment composant de base, le capteur de distance proposé est le « Ping » de marque Parallax. La différence avec le HC-05 qui possède 4 broches, le Parallax lui en possède 3 : 5V, masse et la dernière birectionnelle (echo et trigger en même temps ?)



### c. Trouver le capteur à ultrasons HC-05

Afin d'avoir un montage identique avec le matériel disponible au lycée, nous allons utiliser le hc-05.

 Pour le trouver :

1. Clic sur *Composants de base*
2. *Tout*

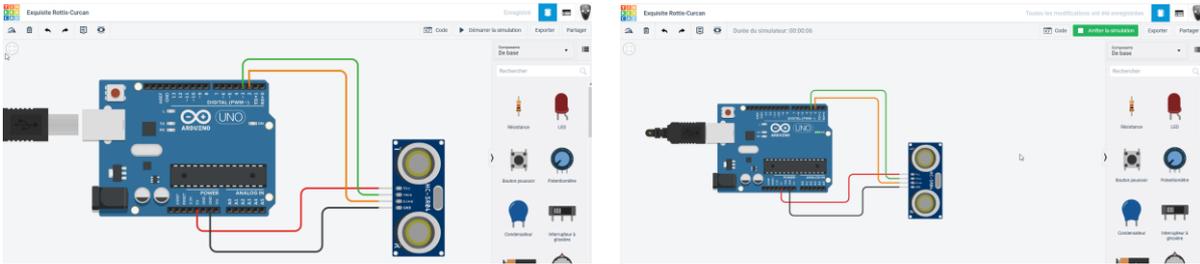


Descendez et choisissez le bon



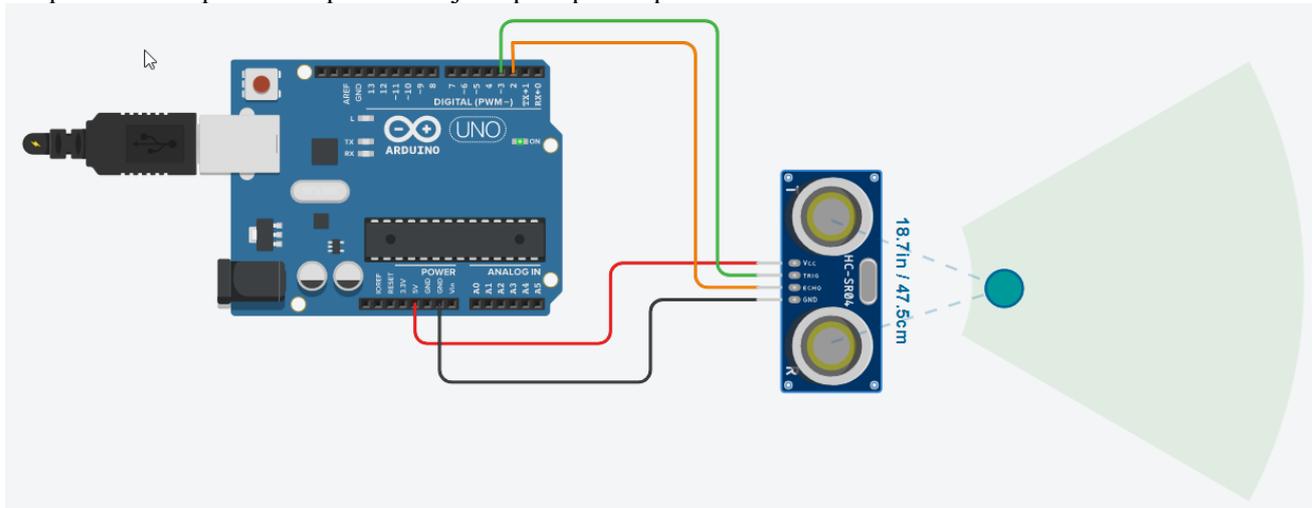
#### d. Utiliser le capteur

Dézoomez avec la molette et déplacer légèrement le montage vers la gauche



Démarrez la simulation

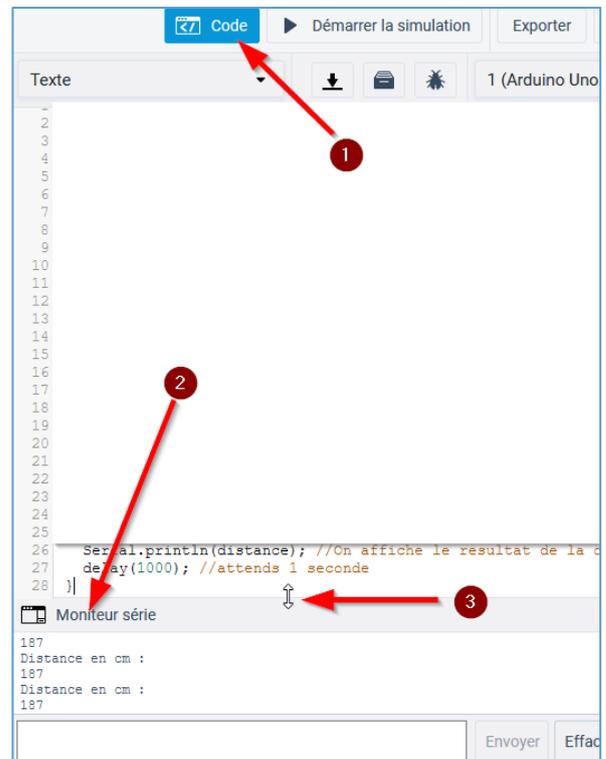
Cliquez sur le capteur et déplacez l'objet repéré par un point bleu avec la souris



Maintenant place au code.

### e. Utilisation du port série

1. Code
2. Moniteur série s'affiche
3. Si nécessaire agrandissez la zone d'affichage du port série





## f. Code Arduino

 Traduisez cet algorithme en langage Arduino

### Algorithme

Déclarer deux broches *Trigger et Echo*

Déclarer une variable qui va récupérer la durée de l'impulsion de la broche Echo

Initialiser les broches

Dans la boucle :

Envoyez un signal de déclenchement qui fournit un signal à l'état haut

Attendre 10us

puis le remettre à l'état bas

Récupérer la durée de l'impulsion avec la fonction [pulseIn](#) (c.f. Annexe) de la broche Echo en ms et l'afficher

Calculer la distance en cm et l'afficher

Attendre 1 seconde