

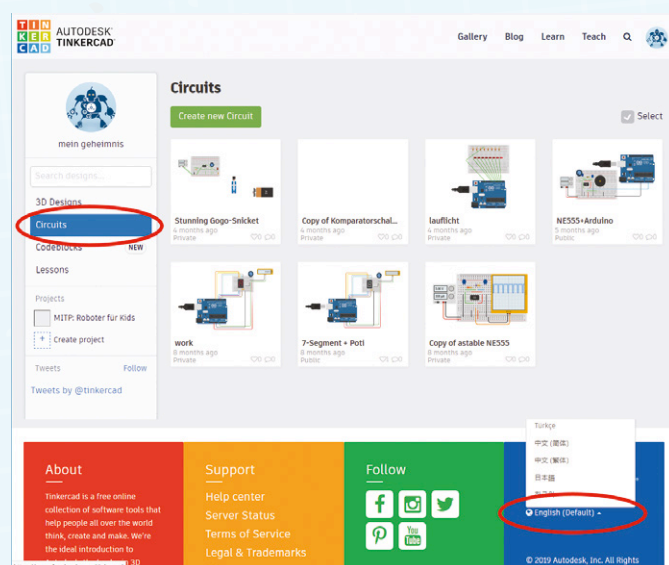
simule tes circuits en ligne

Florian Schäffer

Tu n'as (encore) aucun composant pour tester un circuit que tu as imaginé ou trouvé quelque part, et que tu voudrais maintenant tester. Si tu commençais par le simuler ? Essaie, c'est facile, tu peux même lui faire exécuter du code Arduino !



SPICE (*Simulation Program with Integrated Circuit Emphasis*) est probablement le plus connu des logiciels pour la simulation des circuits électriques analogiques et numériques. Cependant, son utilisation peut paraître lourde quand on débute, et le schéma doit se conformer à certaines exigences. Pour simuler un circuit comme sur une bonne vieille plaque d'essais, il existe heureusement une interface graphique en ligne (aucun logiciel à installer !). C'est *Tinkercad Circuits* qui propose même l'intégration des processeurs ATmega (Arduino Uno) et ATtiny, pour lesquels vous pouvez créer et exécuter des programmes en C/C++ et avec le langage graphique Scratch.



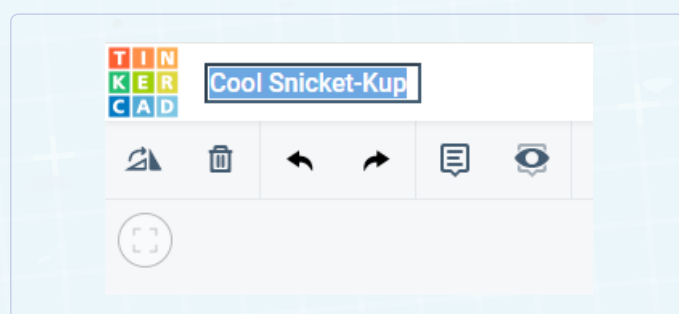
Dans le tableau de bord, choisis la section "Circuits". En bas, tu peux changer de langue si nécessaire (ou si tu as envie de rigoler).

Il faut te connecter sur le site www.tinkercad.com. Soit tu crées un nouveau compte, soit tu te connectes par un compte tiers tel que Google. Quand tu arrives comme nouvel utilisateur dans la section "Circuits" du tableau de bord, tu ne verras pas encore de modèles. En bas à droite, tu peux changer de langue si tu veux.

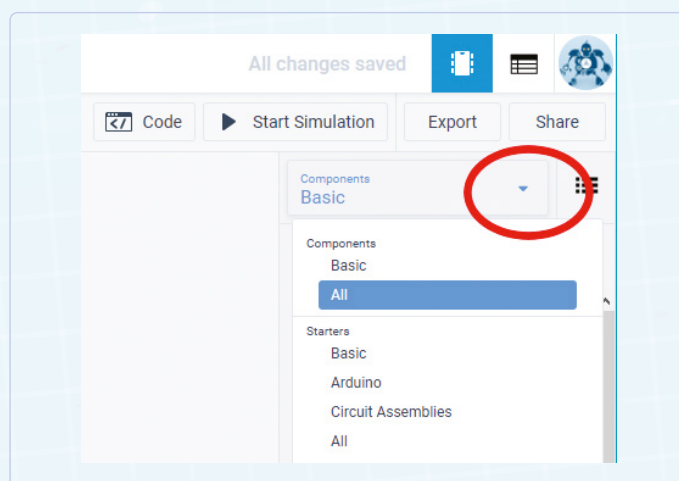
Le projet que je décris ici se trouve en ligne à l'adresse <https://www.tinkercad.com/things/az1gBrXUcIZ>.

1. Clique sur *Créer un nouveau circuit*. La vue de la plaque d'expérimentation ressemble à celle de *Fritzing* (voir l'article à ce sujet dans le précédent numéro d'Elektor).

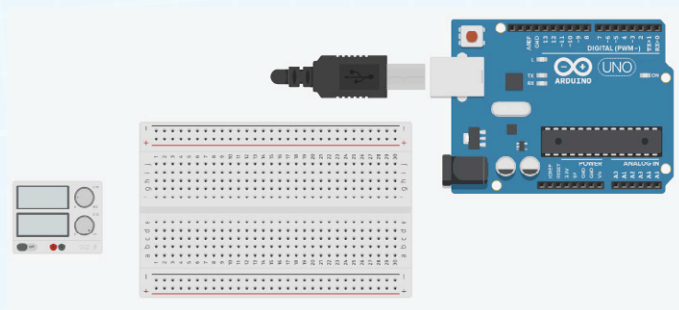
2. Ton dessin ou modèle sera stocké dans le nuage. Dans le coin en haut à gauche, à côté du logo *Tinkercad*, apparaît un nom du projet fantaisiste. Tu peux le garder ou le modifier, il suffit de cliquer sur le texte.



3. À droite de l'écran, tu peux choisir les composants ou des exemples de projets affichés. Dans la liste de sélection sous *Starters* (ou *Démarrateurs* en VF – bravo le traducteur !), tu trouveras quelques exemples pour te familiariser. Pour créer un nouveau projet, tu sélectionneras "All" pour voir tous les *composants*.



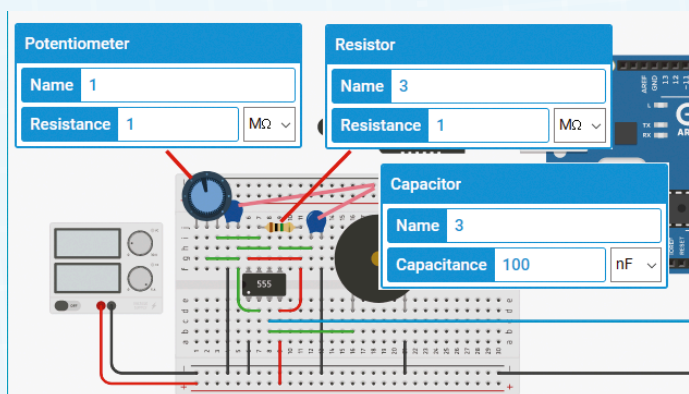
4. Dans la liste de composants, sous *Instruments* se trouve l'alim de labo. Au-dessus, sous *Microcontrôleurs* il y a l'Arduino Uno R3 et la petite carte enfichable. Clique sur le symbole correspondant et fais glisser le composant dans la zone de travail.



5. Pour beaucoup de composants, quand tu cliques sur le symbole dans l'espace de travail, tu peux définir des paramètres (nom, couleur, type, dimensions, etc.). Une boîte de dialogue propose les options de réglage. Clique sur l'alimentation et saisis "5" pour la tension. Le programme est américain, il utilise le point décimal, pas de virgules.

Power Supply	
Name	1
Voltage	5
Current	5

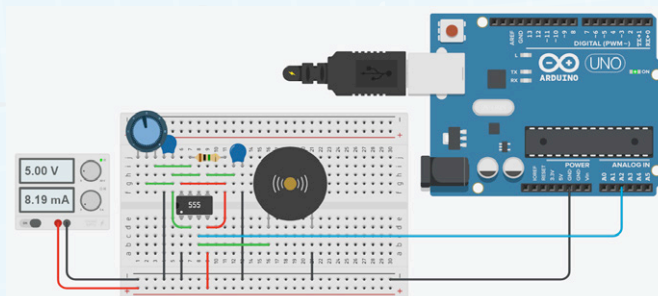
6. Place les composants et les câbles sur la plaque d'expérimentation, puis saisis dans la mesure du possible les numéros et les valeurs de tous les composants.



7. N'oublie pas la connexion de masse de la broche GND de l'Arduino (ligne noire à droite), ni la connexion bleue à la broche A2.

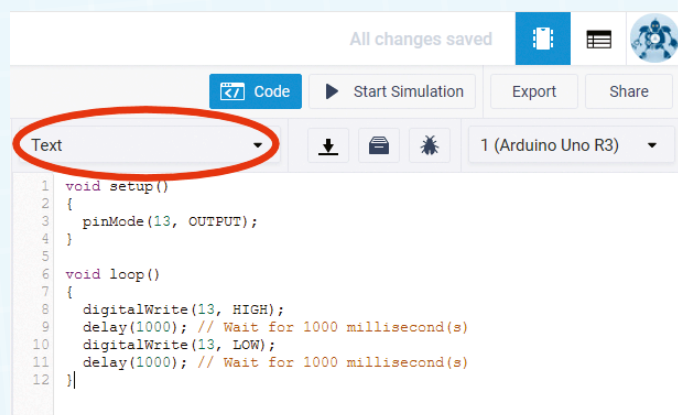
8. Une fois terminée la « construction » du circuit, tu peux tester et simuler le fonctionnement en cliquant sur **Start simulation** en haut à droite. Tu remarqueras la brève animation qui montre l'insertion de la prise USB dans l'Arduino : ce dernier est maintenant alimenté, la LED "ON" s'allume. Le cadran de l'alim de labo affiche la tension et le courant, mais pour l'instant, aucun programme n'est exécuté.

9. Tu peux faire tourner les boutons de l'alim en maintenant enfoncé le bouton de la souris. Le circuit tolère jusqu'à 18 V environ, mais pas l'Arduino, connecté par le fil bleu. Sur un vrai circuit, avec une tension aussi élevée, ton μC serait détruit. Si tu as un HP connecté à ton PC, tu entendas des impulsions sonores émises par l'élément piézo. Avec la souris, tu peux faire tourner le potentiomètre bleu en haut à gauche de la plaque d'expérimentation et changer le son.





10. Clique sur *Stop Simulation* puis sur *Code* pour ouvrir la fenêtre de saisie du code source. Il propose d'abord l'éditeur pour Scratch. Clique sur le triangle de sélection pour passer à l'éditeur de texte. Ne tiens pas compte de l'avertissement, continue.



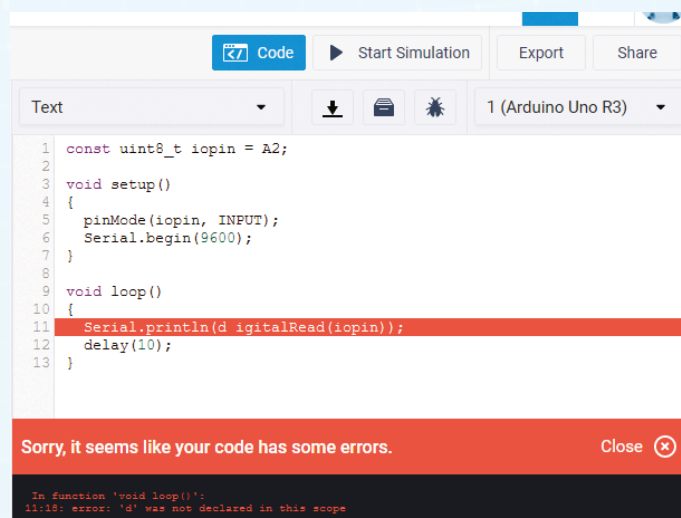
11. Dans la fenêtre de code, il y a déjà un petit programme. Efface-le et remplace-le par les commandes suivantes :

```
const uint8_t iopin = A2;

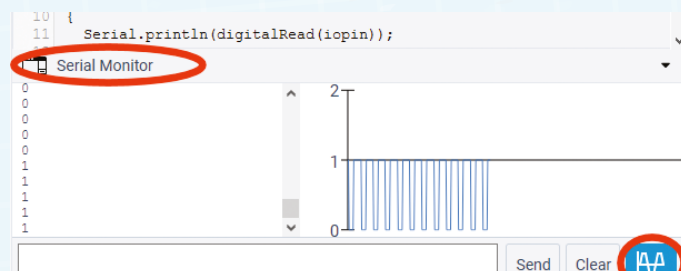
void setup()
{
  pinMode(iopin, INPUT);
  Serial.begin(9600);
}

void loop()
{
  Serial.println(digitalRead(iopin));
  delay(10);
}
```

12. Clique de nouveau sur *Démarrer la simulation* et le nouveau code sera exécuté. Si le programme comporte des erreurs, celles-ci seront signalées comme dans l'EDI Arduino. Avant de poursuivre, il faut les corriger.



13. Dans cet exemple, l'Arduino est censé fonctionner comme un oscilloscope rudimentaire et délivre sur sa broche d'entrée/sortie le niveau du signal qui pilote l'élément piézo. Tu verras la séquence de zéros et de uns si tu cliques sur *Serial monitor* au bas de la vue du code. En bas à droite, clique sur l'icône avec une trace d'oscilloscope et tu verras le rendu graphique.



14. Fais tourner le bouton du potentiomètre pour changer le son et tu verras changer la trace de l'oscilloscope.

Cet article est extrait d'un hors-série d'Elektor *Elektor Special: Introduction to Electronics with Arduino*
www.elektor.fr/elektor-special-introduction-to-electronics-with-arduino
disponible dans l'e-choppe d'Elektor. ◀

200225-C-02