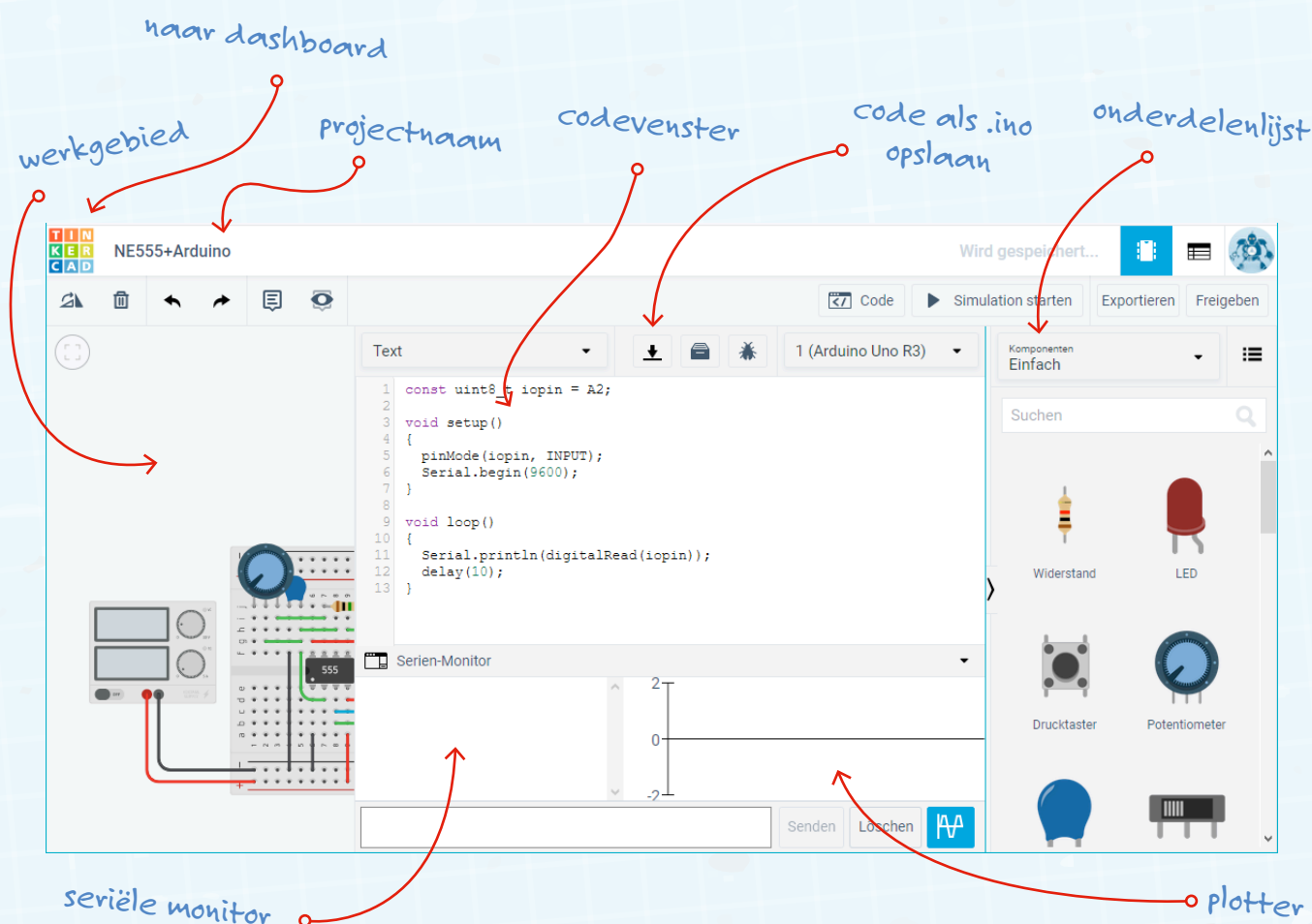




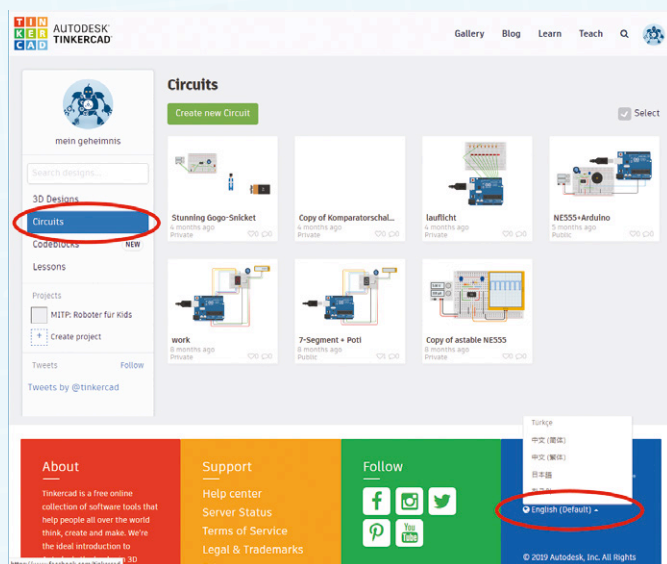
Schakelingen online simuleren

Florian Schäffer (Duitsland)

Hebt u (nog) geen componenten om uw schakelingen in de praktijk te testen, of maakt u liever niets kapot? Dan komt een simulatie die zelfs Arduino-code kan uitvoeren goed van pas!



SPICE (Simulation Program with Integrated Circuit Emphasis) is waarschijnlijk de bekendste software voor het simuleren van analoge en digitale elektrische schakelingen. Het gebruik ervan is echter vaak lastig en vereist een geschikt schema. Daarom is het goed dat er nu een grafische gebruikersinterface is die u online kunt gebruiken zonder software te moeten installeren, en waarin u de schakeling kunt opbouwen zoals op een breadboard. Tinkercad Circuits biedt zelfs integratie van ATmega- (Arduino Uno) en ATtiny-processoren, waarvoor u programma's in C/C++ en de grafische taal Scratch kunt maken en uitvoeren.



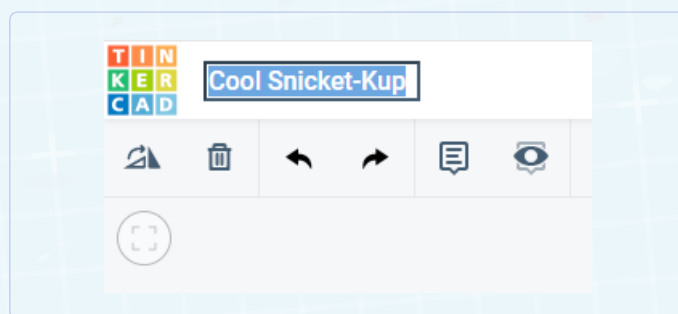
Ga in het dashboard naar het bereik Circuits. Indien nodig kunt u onderin de taal wijzigen.

Meld u aan op de pagina www.tinkercad.com. U kunt hiertoe een nieuw account aanmaken of inloggen via een externe provider zoals Google. Ga naar het bereik Circuits in het dashboard – als nieuwe gebruiker ziet u daar nog geen ontwerpen staan. Indien nodig kunt u de taal onderin wijzigen.

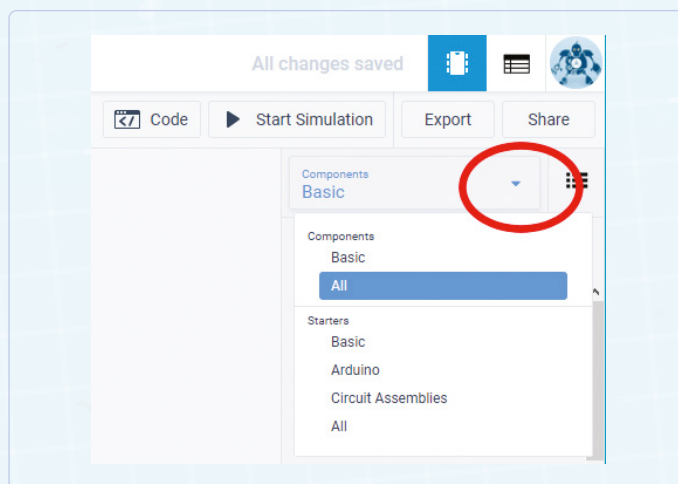
U kunt het hier besproken project ook online vinden op www.tinkercad.com/things/az1gBrXUclZ.

1. Klik op *Create new Circuit*. De breadboard-weergave is vergelijkbaar met die van Fritzing, dus het is goed wanneer u daar al iets van weet en het betreffende artikel erover hebt gelezen (Elektor november/december 2020).

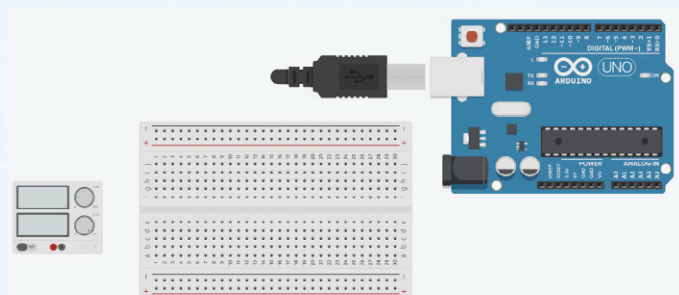
2. Uw ontwerp wordt permanent opgeslagen in de cloud. Linksboven naast het Tinkercad-logo staat daarvoor al een fantasierijke naam. U kunt op die naam klikken en uw eigen projectnaam invoeren.



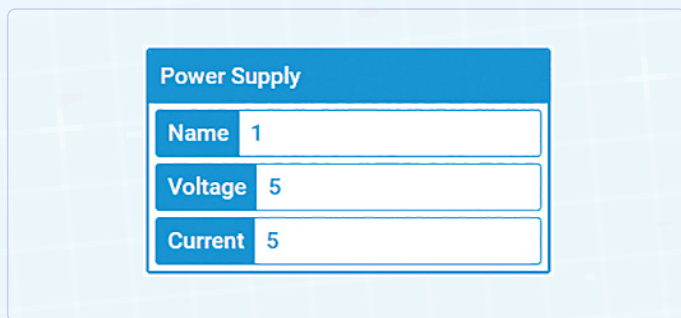
3. Rechts kunt u een preselectie maken van de componenten en voorbeeldprojecten die getoond moeten worden. In de keuzelijst onder *Starters* staan enkele voorbeelden die u kunt uitproberen. Om zelf een nieuw project te maken, selecteert u onder *Components* de optie *All*.



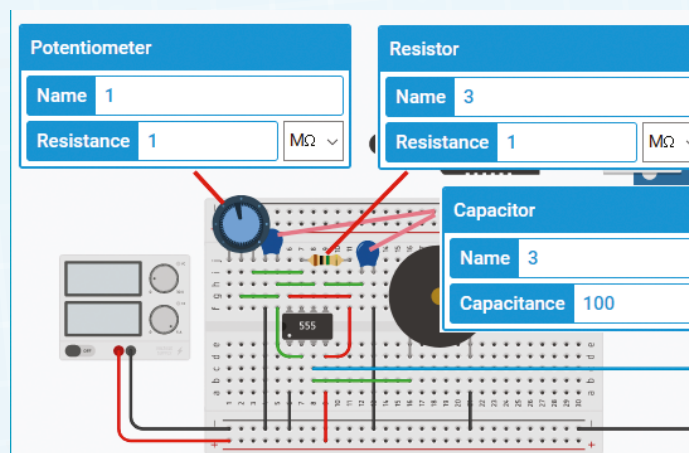
4. Zoek onder *Components* de *Power Supply* (labvoeding), de Arduino Uno R3 en het kleine breadboard. Klik op het betreffende symbool en plaats het onderdeel vervolgens in het werkgebied.



5. Bij veel componenten kunt u parameters zoals kleur, type of waarde definiëren door er in het werkgebied op te klikken. Er verschijnt dan een klein dialoogvenster met opties. Klik op de stroombron en voer "5" in bij *Voltage*. Houd er bij alle waarden rekening mee dat u de Amerikaanse notatie gebruikt, met een decimale punt in plaats van een komma.



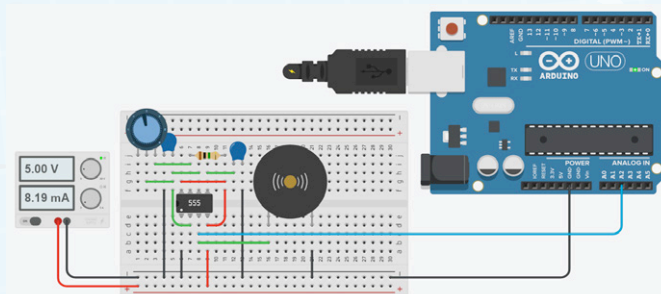
6. Plaats de componenten en draden op het breadboard. De onderstaande figuur toont welke waarden u voor de componenten moet opgeven (waar van toepassing).



7. Ook is een verbinding met een GND-pin op de Arduino nodig (de zwarte lijn naar rechts) en de blauwe verbinding met pin A2.

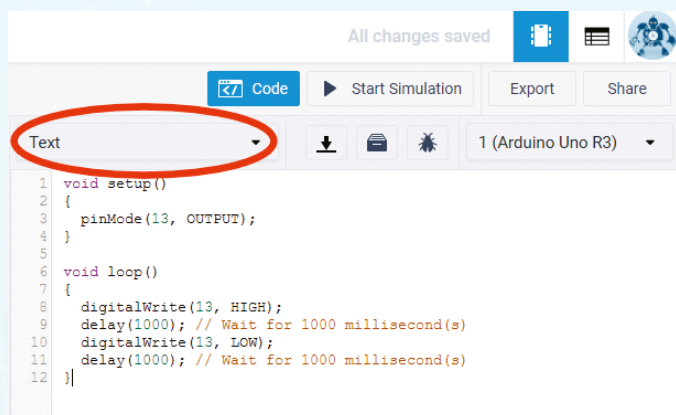
8. Nadat de schakeling is opgebouwd, kunt u de elektrische functie testen en simuleren door rechtsboven op Start Simulation te klikken. De USB-connector wordt in een korte animatie in de Arduino gestoken om te laten zien dat die nu van stroom wordt voorzien (de "ON"-LED licht op). Er is echter nog geen programma voor de Arduino. Op de labvoeding worden de spanning en de actueel opgenomen stroom weergegeven.

9. U kunt (met ingedrukte muisknop) de knoppen op de voedingseenheid verdraaien. De schakeling zelf verdraagt maximaal ongeveer 18 V – maar de Arduino, die is aangesloten via de blauwe draad, kan daar niet tegen. In de praktijk zou u zo de microcontroller kapot maken – hier kan het echter geen kwaad. De piezo-zoemer maakt een krakend geluid als u een luidspreker op uw PC hebt aangesloten. Met de muis kunt u de blauwe potmeter linksboven op het breadboard verdraaien en daarmee de toon veranderen.





10. Klik op *Stop Simulation* en vervolgens op *Code* (aan de linkerkant) om het invoervenster voor de broncode te openen. Het codevenster schuift vanaf de rechterkant in beeld. Standaard wordt de Scratch-editor getoond. Schakel bovenin het venster om naar Text. Er verschijnt een waarschuwing, die u met *Next* bevestigt.



11. Er staat al een klein programma in het codevenster. Wis alles wat daar staat en vervang deze door de volgende opdrachten:

```

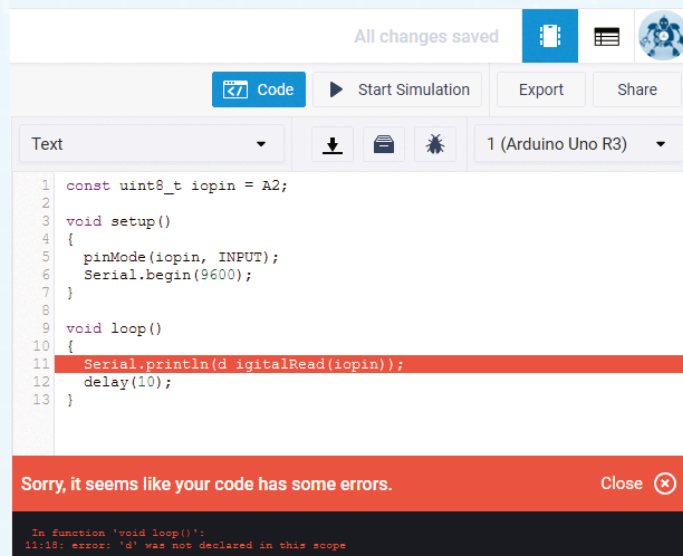
const uint8_t iopin = A2;

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

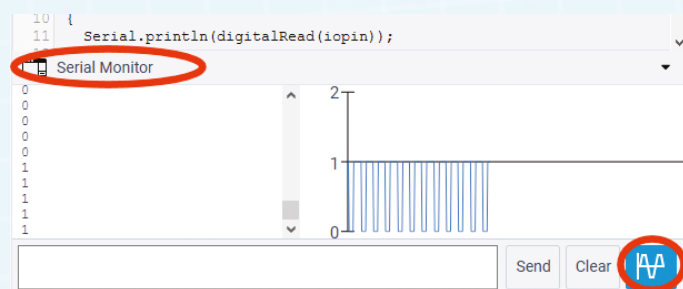
void loop()
{
  Serial.println(digitalRead(iopin));
  delay(10);
}

```

12. Als u nu weer op *Start Simulation* klikt, wordt de programmacode daadwerkelijk uitgevoerd. Als uw programma fouten bevat, worden deze gemarkeerd zoals in de Arduino IDE en moeten ze eerst worden gecorrigeerd.



13. In dit voorbeeld werkt de Arduino als een eenvoudige oscilloscoop die via zijn I/O-pin een signaalniveau uitvoert dat correspondeert met het signaal dat de piezo-zoemer aanstuurt. Hiertoe klikt u op *Serial Monitor* onder in het codevenster, zodat u een reeks nullen en enen ziet. Helemaal rechts onderaan staat een pictogram van een oscillogram: klik erop om de grafische uitvoer te openen.



14. Draai nogmaals aan de potmeter om de toon te veranderen; u zult zien hoe de weergegeven curve mee verandert.

Dit artikel is afkomstig uit de (Duitstalige) Elektor-special "Einstieg in die Elektronik mit Arduino", dat in de Elektor-shop kan worden besteld: www.elektor.de/elektor-special-einstieg-in-die-elektronik-mit-arduino. Vanaf oktober 2020 is de special ook in het Engels verkrijgbaar. ◀

200225-C-03