

# Opdracht Arduino

(versie 5V 20/21 corona-proof)

Deze periode maak je kennis met het Arduino-platform. Bij deze opdracht ga je zelf een eenvoudig apparaat maken dat gebruik maakt van een microcontroller.

## Projecten

Je kunt kiezen uit een van de volgende projecten:

### **Simon Says**

Simon Says is een geheugenspel, vergelijkbaar met 'ik ga op reis en neem mee...'. Het spel bestaat uit vier knoppen en vier LEDjes met andere kleuren. Het is de bedoeling dat je met de knoppen in precies dezelfde volgorde de LEDjes laat oplichten als het spel je voordoet. Het begint met één LED, maar de rij wordt, als je het goed nadoet, steeds langer, totdat je het niet meer kunt onthouden en een fout maakt.

#### *Uitbreidingen:*

- score op display
- niet alleen licht, maar bijvoorbeeld ook geluid
- maak bij het oplichten van een LED ook een specifiek geluidje
- ...

### **Stopwatch**

Maak een stopwatch. De stopwatch laat honderdsten van seconden, seconden, minuten en uren zien op een LCD display. Er zit een knop op waarmee je de stopwatch starten, stoppen en weer op nul zetten.

#### *Uitbreidingen:*

- Een extra knop die de tussenstand laat zien terwijl de tijd op de achtergrond doorloopt.
- Audio signalen als je een knop indrukt
- Een timer-functie die aftelt van een in te stellen aantal uren, minuten en seconden. Als de timer op 00:00:00 staat, dan klinkt er een alarm.
- ...

### **Tiksnelheid meter**

Maak een spel waarbij twee spelers gedurende 1 minuut lang zo snel vaak mogelijk op een knop moet drukken. Elke speler heeft zijn eigen knop. Er is ook een knop waarmee je het spel kunt starten. Bij het starten 'telt' het spel met LEDjes of geluid 3 seconden af voordat de minuut ingaat. Na de minuut merk je dat je moet stoppen. Het spel toont vervolgens op een LCD display de score.

#### *Uitbreidingen:*

- toon ook de hoogste score
- geef door middel van het LED-display direct feedback over de snelheid van de speler

- Maak een variant waarbij bij de start van het spel in het midden van het display een balletje wordt getoond. Het balletje beweegt naar links of rechts, afhankelijk van het aantal klikt dat beide spelers hebben gegeven. Het balletje beweegt naar de speler die het langzaamst klikt. Als het balletje aan de rand van het scherm is dan heeft de speler waar het balletje geëindigd is verloren.
- ...

### **Je eigen idee**

Je mag een apparaat of spel zelf bedenken. De docent met je idee goedkeuren.

### *Uitbreidingen:*

- ...

## **Inleveren**

Je levert in:

1. Een circuit/aansluitschema (.pdf of .jpg bestand) van je Arduino en alles wat je erop hebt aangesloten.
2. Testprogramma (.ino bestand) dat laat zien dat alles wat je het aangesloten op je Arduino werkt.
3. Een toestandsdiagram van je apparaat (.pdf of .jpg bestand).
4. De gebruikte code (.ino bestand), inclusief Nederlandstalig commentaar. Als je stukjes code hebt overgenomen, dan staat erbij waar je het vandaan hebt.
5. Een video (.mp4 of .mov bestand met video van maximaal 20MB of .txt bestand met link naar youtube, maximaal 2 minuten) waarin je je apparaat demonstreert.

Inleveren doe je op GitHub. Je zet alle bestanden in de map “inleveren” van de GitHub-repo die de docent voor je heeft aangemaakt.

Tip: Je kunt meerdere keren inleveren. Lever een dag voor de deadline alvast een versie in die (bijna) af is. Op die manier voorkom je dat je door onverwachte problemen niets kunt inleveren.

Je levert deze praktische opdracht uiterlijk in op **donderdag 22 april 23:59** in.

## **Beoordeling**

Becijfering gebeurt op de kwaliteit van de volgende onderdelen:

- Code
- Moeilijkheid en uitbreidingen
- Elektronica (circuit)
- Verzorgdheid, originaliteit en inzet  
(bewaar je tussenresultaat elke keer dat je eraan werkt in GitHub)

# Aanpak voor Opdracht Arduino

**Je bewaart je tussenresultaten op GitHub, telkens als je aan je project gewerkt hebt!**

Voor het uitvoeren van de opdracht volg je de volgende stappen.

## Stap 1: Starten

- Je hebt je GitHub-inlognaam aan de docent doorgegeven, zodat die een Github-repo kan aanmaken.
- Je hebt gekozen wat je gaat maken en dat aan de docent doorgegeven, bij een eigen idee moet de docent het goedkeuren.
- Je hebt je bakje met Arduino-onderdelen opgehaald op school.
- Je hebt thuis een computer of laptop (Arduino werkt niet op een iPad) waarop je de gratis Arduino software hebt geïnstalleerd, of je hebt afgesproken dat je een paar uur per week op school op een computer kan werken.

## Stap 2: Circuit maken

- Je weet welke sensoren en actuatoren je wilt gebruiken.
- Je hebt een schema getekend van je ontwerp, waaruit blijkt hoe alles aangesloten moet worden.
- Je hebt je circuit in elkaar gezet.

## Stap 3: Testcode maken

- Je hebt code gemaakt waarmee je kunt testen dat alle onderdelen op je circuit goed zijn aangesloten.

## Stap 4: Toestandsdiagram maken

- Je hebt een toestandsdiagram gemaakt.
- Je hebt het toestandsdiagram in code omgezet, zodat het werkt met jouw circuit. Daarbij heb je het door de docent aangeleverde code-template gebruikt.
- Nog niet alle onderdelen van je apparaat hoeven te werken, maar je apparaat moet wel de toestanden door kunnen lopen.

## Stap 5: Je basis-code afmaken

- Je hebt de code van je toestandsdiagram uitgebreid, zodat alle onderdelen van je apparaat werken.

## Stap 6: Uitbreidingen maken

- Je hebt je ontwerp uitgebreid met extra ideeën en deze heb je verwerkt in je aansluitschema (als je extra Arduino-onderdelen hebt toegevoegd) en in je code.

## Stap 7: Inleveren

- Je hebt een filmpje gemaakt van de laatst werkende versie van je project dat het doet.
- Je hebt alle documenten (zie onder “inleveren”) op GitHub gezet in de map “inleveren”.