

Figure 1: Boek 3: Actuatoren met hoog vermogen

#	Beschrijving
9	RGB LED
10	Potmeter RGB
11	FSR RGB
12	LDR RGB

# Contents

Voorwoord	1
Les 1c: Meer Kleur Fade Met For	2
Potmeter	11
Aansluiten potmeter met LED	13
4. FSR	16
9. LDR	31

## Voorwoord

Dit is een boek over Arduino, geschreven voor jonge tieners. Een Arduino is een machine die je kunt programmeren. Dit boek leert je hoe je elektronica op de Arduino aansluit, en hoe je deze programmeert.

## Over dit boek

Dit boek heeft een CC-BY-NC-SA licentie.



Figure 1: De licentie van dit boek

(C) Richèl Bilderbeek en alle docenten en alle leerlingen

Met dit boekje mag je alles doen wat je wilt, als je maar verwijst naar de oorspronkelijke versie op deze website: [https://github.com/richelbilderbeek/arduino\\_voor\\_jonge\\_tieners](https://github.com/richelbilderbeek/arduino_voor_jonge_tieners). Dit boekje zal altijd gratis, vrij en open blijven.

Het is nog een beetje een slordig boek. Er zitten tiepvauten in en de opmaak is **niet** altijd *even mooi*. Omdat dit boek op een website staat, kan iedereen die dit boek te slordig vindt minder slordig maken.

# Les 1c: Meer Kleur Fade Met For

Soms denk je ‘Kan dit slimmer?’. Een for-loop is een manier om iets slimmer te doen!

## Regenboog

In deze les gaan we een regenboog programmeren!

Hier zie je hoe een regenboog werkt:

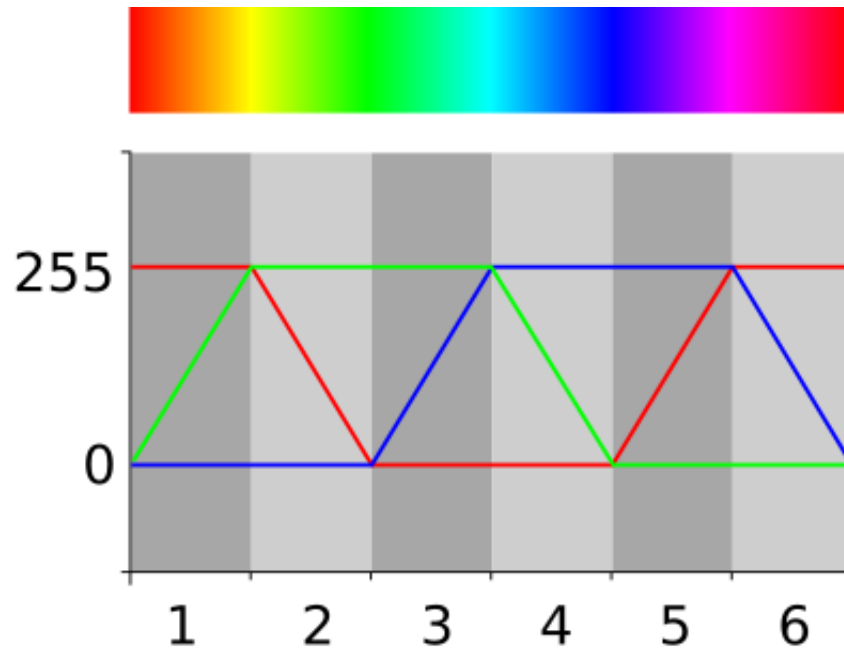


Figure 2: Regenboog



Geel licht is rood en groen licht samen.

---

# Opdracht 1

Sluit onderstaand schema aan:

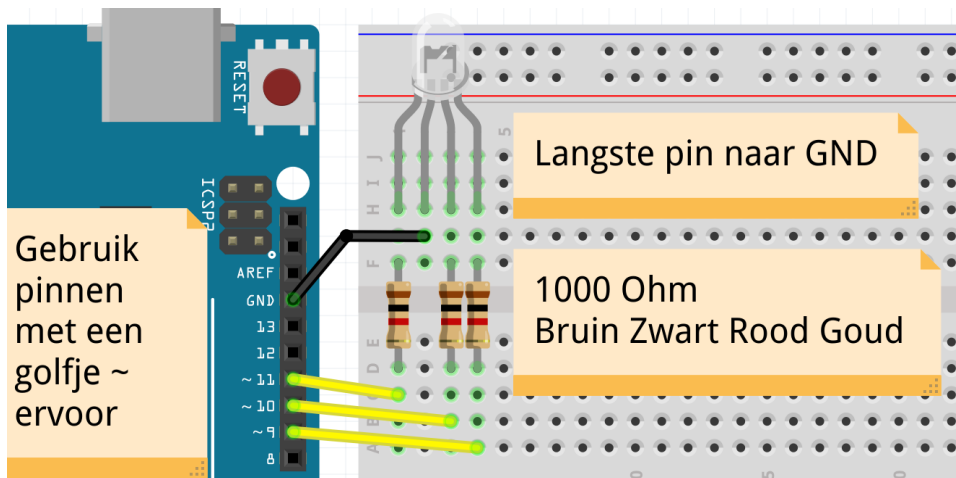


Figure 3: Schema

Gebruik deze code:

```
const int pin_rood = 9;
const int pin_groen = 10;
const int pin_blaauw = 11;

void setup()
{
  pinMode(pin_rood, OUTPUT);
  pinMode(pin_groen, OUTPUT);
  pinMode(pin_blaauw, OUTPUT);
}

void loop()
{
  digitalWrite(pin_rood, HIGH);
  digitalWrite(pin_groen, HIGH);
  digitalWrite(pin_blaauw, HIGH);
}
```

De getallen achter `pin_rood`, `pin_groen` en `pin_blaauw` zijn fout. Zorg dat dit de goede getallen worden. Hoe? Trek twee draden los en kijk welke kleur het LEDje krijgt.

## Oplossing 1

Ha, dit mag je zelf uitvinden :-).

# For loop

We beginnen met deze code:

```
const int pin_rood  = 9; // Fout getal, zie vorige opdracht
const int pin_groen = 10; // Fout getal, zie vorige opdracht
const int pin_blaauw = 11; // Fout getal, zie vorige opdracht

void setup()
{
  pinMode(pin_rood , OUTPUT);
  pinMode(pin_groen, OUTPUT);
  pinMode(pin_blaauw, OUTPUT);
}

void loop()
{
  for (int i=0; i<256; ++i)
  {
    analogWrite(pin_rood, i);
    analogWrite(pin_groen, 0);
    analogWrite(pin_blaauw, 255);
    delay(30);
  }
}
```



---

for (int i=0; i<256; ++i) { }	‘Lieve computer, laat een getal i lopen van 0 tot 256 in stapjes van 1.’
analogWrite(9, 255)	‘Lieve computer, zet pin 9 vol aan.’
analogWrite(9, 127)	‘Lieve computer, zet pin 9 half aan.’
analogWrite(9, 0)	‘Lieve computer, zet pin 9 uit.’
analogWrite(9, i)	‘Lieve computer, laat pin 9 langzaam aan gaan (moet binnen de for-loop hierboven).’

---



Tellen tot drie: '1, 2'. Tellen tot en met drie: '1, 2, 3'

---

## Opdracht 2

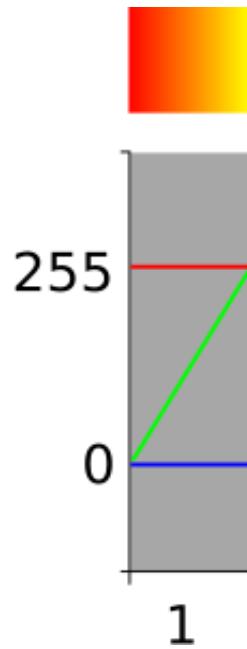


Figure 4: Eerste gedeelte van de regenboog

Kijk naar het eerste gedeelte van de regenboog, figuur **Eerste gedeelte van de regenboog**. De rode lijn staat daar op 255, de groene lijn gaat langzaam van 0 naar 255 en de blauwe lijn staat bij nul.

Pas de code zo aan dat het lampje van rood naar geel (rood en groen) gaat.



---

<pre>for (int i=0; i&lt;256; ++i) { } analogWrite(9, 255) analogWrite(9, 0) analogWrite(9, i)</pre>	<pre>‘Lieve computer, laat een getal i lopen van 0 tot 256 in stapjes van 1.’ ‘Lieve computer, zet pin 9 vol aan.’  ‘Lieve computer, zet pin 9 uit.’ ‘Lieve computer, laat pin 9 langzaam aan gaan (moet binnen de for-loop hierboven).’</pre>
---	--

---

## Oplossing 2

```
const int pin_rood   = 9; // Fout getal, zie vorige opdracht
const int pin_groen  = 10; // Fout getal, zie vorige opdracht
const int pin_blaauw = 11; // Fout getal, zie vorige opdracht

void setup()
{
  pinMode(pin_rood , OUTPUT);
  pinMode(pin_groen, OUTPUT);
  pinMode(pin_blaauw, OUTPUT);
}

void loop()
{
  for (int i=0; i<256; ++i)
  {
    analogWrite(pin_rood, 255);
    analogWrite(pin_groen, i);
    analogWrite(pin_blaauw, 0);
    delay(30);
  }
}
```



## Opdracht 3

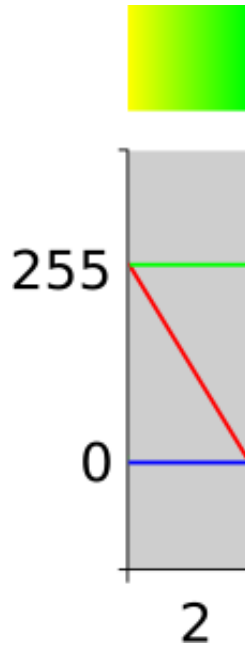


Figure 5: Tweede gedeelte van de regenboog

Kijk naar het tweede gedeelte van de regenboog, figuur **Tweede gedeelte van de regenboog**. De rode lijn gaat van 255 naar 0, de groene lijn staat op 255 en de blauwe lijn staat bij nul.

Pas de code zo aan dat het lampje van rood naar geel (rood en groen) gaat.

Tip: voeg deze code toe:

```
for (int i=0; i<256; ++i)
{
  analogWrite(pin_rood, 255 - i);
  analogWrite(pin_groen, 255);
  analogWrite(pin_blaauw, 0);
  delay(30);
}
```



---

```
for (int i=0;
i<256; ++i) { }
  analogWrite(9,
    255)
analogWrite(9, 0)
```

‘Lieve computer, laat een getal i lopen van 0 tot 256 in  
stapjes van 1.’

‘Lieve computer, zet pin 9 vol aan.’

‘Lieve computer, zet pin 9 uit.’



---

```
analogWrite(9, 255 - i)  'Lieve computer, laat pin 9 langzaam uit gaan (moet binnen  
                           de for-loop hierboven).'
```

---

## Oplossing 3

```
const int pin_rood   = 9; // Fout getal, zie vorige opdracht
const int pin_groen  = 10; // Fout getal, zie vorige opdracht
const int pin_blauw  = 11; // Fout getal, zie vorige opdracht

void setup()
{
  pinMode(pin_rood , OUTPUT);
  pinMode(pin_groen, OUTPUT);
  pinMode(pin_blauw, OUTPUT);
}

void loop()
{
  for (int i=0; i<256; ++i)
  {
    analogWrite(pin_rood, 255);
    analogWrite(pin_groen, i);
    analogWrite(pin_blauw, 0);
    delay(30);
  }
  for (int i=0; i<256; ++i)
  {
    analogWrite(pin_rood, 255 - i);
    analogWrite(pin_groen, 255);
    analogWrite(pin_blauw, 0);
    delay(30);
  }
}
```

## Eindopdracht

Maak nu de regenboog af.

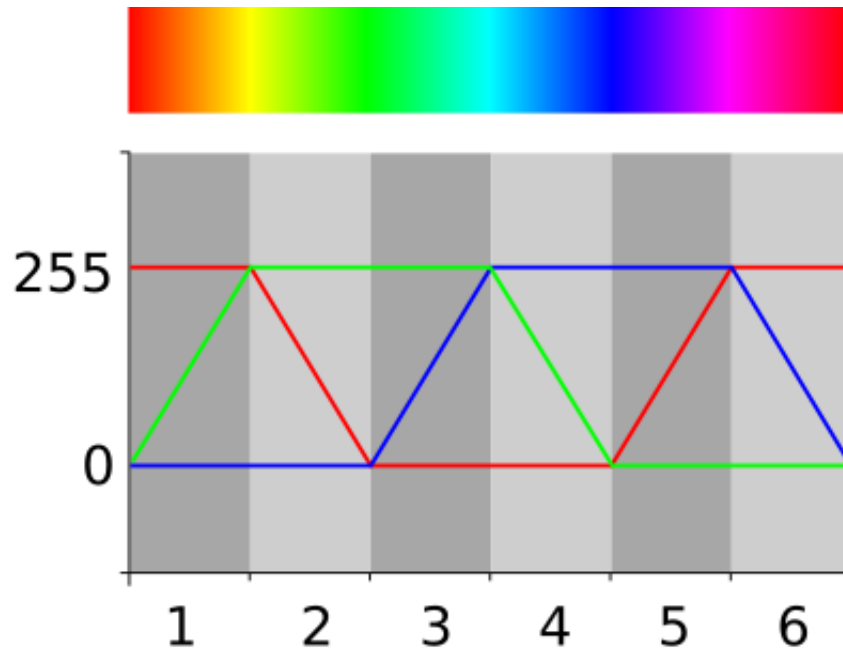


Figure 6: Regenboog

## Potmeter

Met een potmeter kun je een spanning naar de Arduino sturen. Deze kun je gebruiken als bijvoorbeeld een volumeknop.



Figure 7: Een potmeter

In deze les leer je:

- Wat een potmeter is
- Hoe je een potmeter gebruikt

## Aansluiten

Eerst sluiten we alleen een Arduino aan:

Ik denk dat dit wel moet lukken :-)

## Code: lezen potmeter met seriele monitor

Met deze code meten we de stand van de potmeter:

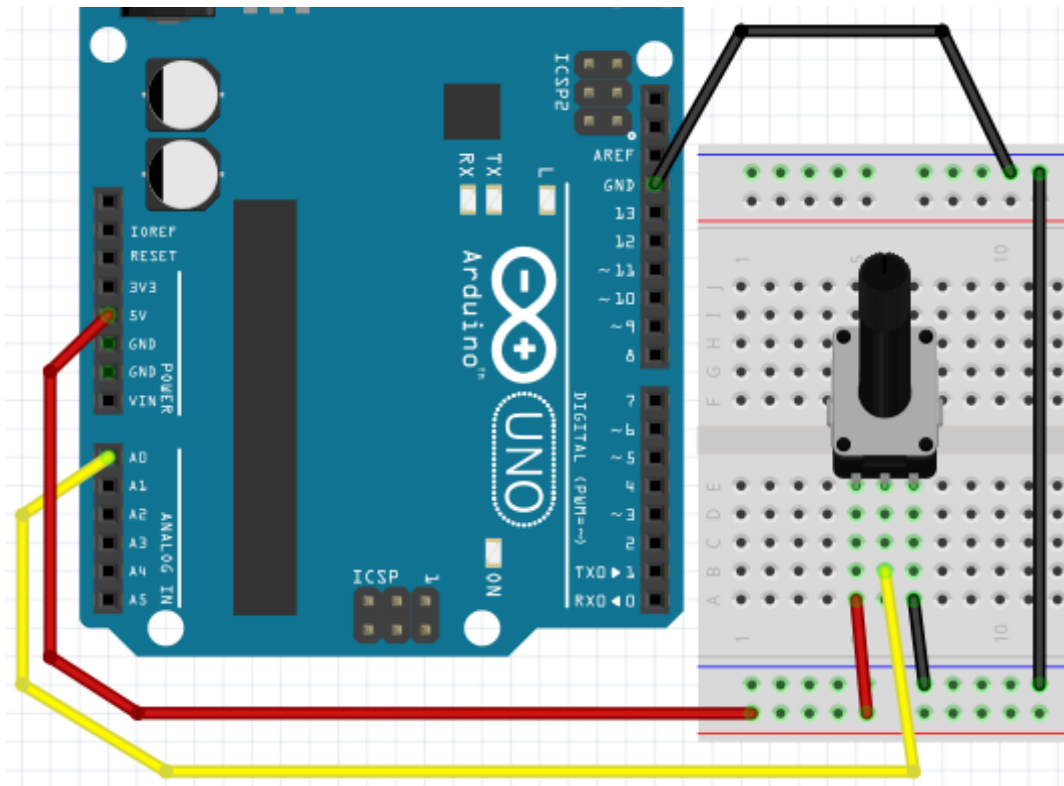


Figure 8: Potmeter aansluiten

```
void setup()
{
  pinMode(A0, INPUT);
  Serial.begin(9600);
}

void loop()
{
  Serial.println(analogRead(A0));
  delay(100);
}
```

Dit doet de code:

- In de `setup` functie gebeuren twee dingen
  - `pinMode(A0, INPUT)`: de pin A0 is een pin die leest, een input
  - `Serial.begin(9600)`: de seriele monitor stuurt 9600 bits ('nullen en enen') per seconde
- In de `loop` functie gebeuren twee dingen
  - `Serial.println(analogRead(A0))`: lees de pin A0 uit en schrijf deze naar de seriele monitor
  - `delay(100)`: wacht honderd milliseconden

## Opdrachten

1. Upload het programma. In de Arduino IDE, klik rechtsboven op 'Seriele Monitor'. Wat zie je?
2. Draai de potmeter helemaal naar links (tegen de klok in). Welke waarde zie je op de seriele monitor?
3. Draai de potmeter helemaal naar rechts (met de klok mee). Welke waarde zie je op de seriele monitor?

## Oplossingen

1. Je ziet een getal van nul tot 1024, afhankelijk van de stand van de potmeter
2. Je ziet het getal nul
3. Je ziet het getal 1023

## Aansluiten potmeter met LED

Nu sluiten we ook een LED aan:

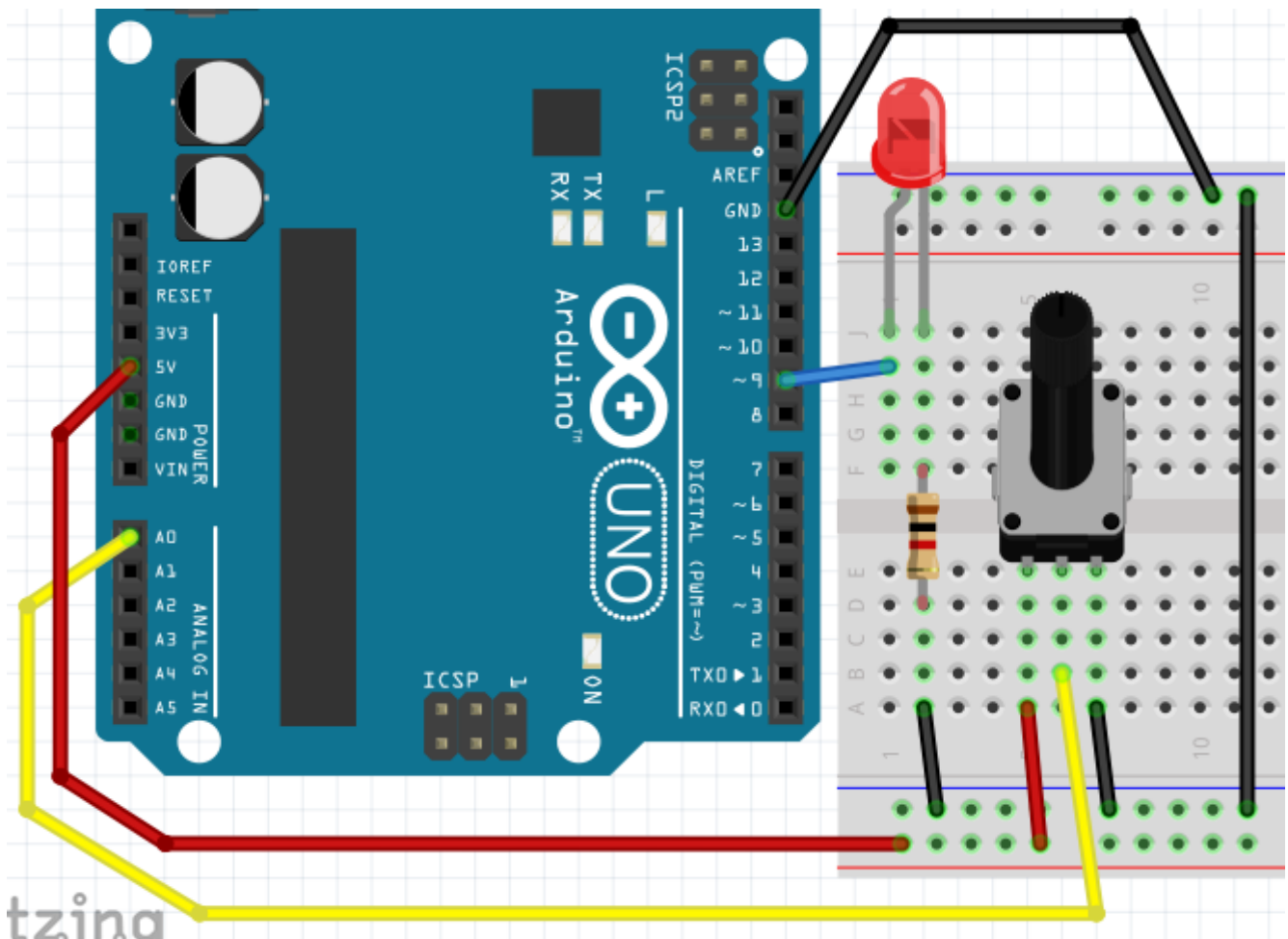


Figure 9: Potmeter met LED

Let op:

- het weerstandje aan de LED is duizend Ohm (bruin-zwart-rood-goud)
- het LEDje moet op een pin met PWM ('een golfje')

## Reageren op potmeter

Nu gaan we het LEDje laten reageren op de potmeter:

```
void setup()
{
  pinMode(A0, INPUT);
  pinMode(9, OUTPUT);
}

void loop()
{
  analogWrite(9, analogRead(A0) / 8);
  delay(100);
}
```

Dit doet de code

- In de `setup` functie gebeuren drie dingen:
  - `pinMode(A0, INPUT)`: de pin A0 is een pin die leest, een input
  - `pinMode(9, OUTPUT)`: pin 9 is een pin waar stroom uitkomt, een output
- In de `loop` functie gebeuren twee dingen:
  - `analogWrite(9, analogRead(A0) / 8)`: zet op pin 9 een spanning van `analogRead(A0) / 8`. `analogRead(A0) / 8` betekent: de waarde gelezen van A0 gedeeld door acht (de / is een deelstreep).
  - `delay(100)`: wacht honderd milliseconden

## Opdracht

- Wat gebeurt er als je het getal van de deling verandert?
- Wat is de beste waarde van het getal voor de deling/

## Oplossingen

1. Bij een hoger getal, gaat het lampje steeds minder hard in volle stand. Bij een lager getal dan vier gaat het lampje raar reageren: het lampje gaat eerst harder, dan uit en weer harder, etc
2. Het beste getal is vier. Want uit `analogRead` komt hoogstens 1023, terwijl je met `analogWrite` hoogstens 255 kunt schrijven. 1023 gedeeld door 4 is 255 rest 3. De

Ardiono maakt daar 255 van

## Eindopdracht

- Sluit vier LEDjes aan: een rode, gele, groene en blauwe
- Als de potmeter helemaal naar links is, moet er geen LEDje branden.
- Als de potmeter meer naar rechts gedraaid wordt, gaat het groene LEDje branden
- Als de potmeter nog meer naar rechts gedraaid wordt, gaat het gele LEDje branden
- Als de potmeter nog meer naar rechts gedraaid wordt, gaat het rode LEDje branden
- De blauwe LED gaat langzaam aan als er meer naar rechts gedraaid wordt



## 4. FSR

### Seriele monitor

---



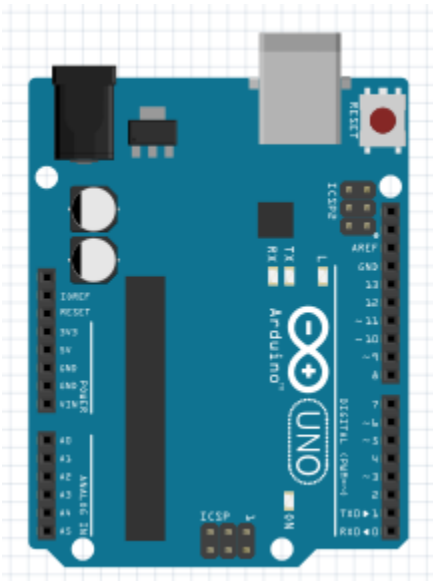
Seriele monitor: de plek waar je de Arduino kunt laten praten via een seriele poort

---

Met de seriele monitor kunnen we de Arduino laten praten. Of precies: dat deze tekst naar de seriele monitor stuurt. De seriele monitor laat deze tekst op je computer zien.

### Alleen Arduino aansluiten

Eerst sluiten we alleen een Arduino aan:



Ik denk dat dit wel moet lukken :-)

---



De seriele monitor gaat via het USB snoer tussen Arduino naar computer

---

## Code: seriele monitor

```
void setup()
{
  Serial.begin(9600);
}

void loop()
{
  Serial.println("Hallo");
  delay(1000);
}
```



---

`Serial.begin(9600);` 'Lieve computer, laat de Arduino praten met 9600 bits per seconde'

`Serial.print("Hallo");` 'Lieve computer, laat de Arduino het woord Hallo zeggen'

`Serial.println("Hallo");` 'Lieve computer, laat de Arduino het woord Hallo zeggen en een nieuwe regel beginnen'

---

## Opdrachten 1

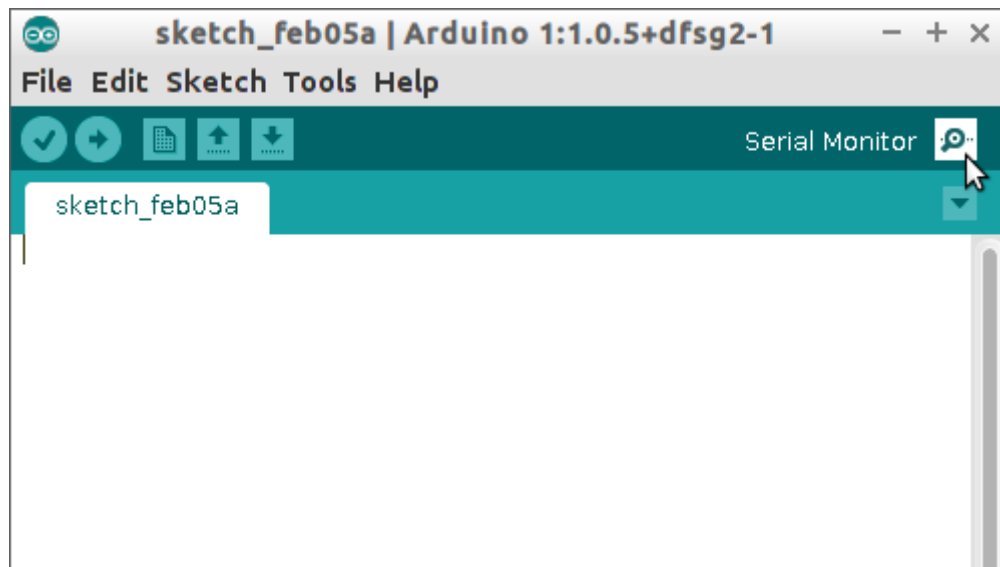


Figure 10: De seriele monitor zit hier

1. Upload het programma. In de Arduino IDE, klik rechtsboven op 'Seriele Monitor'. Wat zie je?

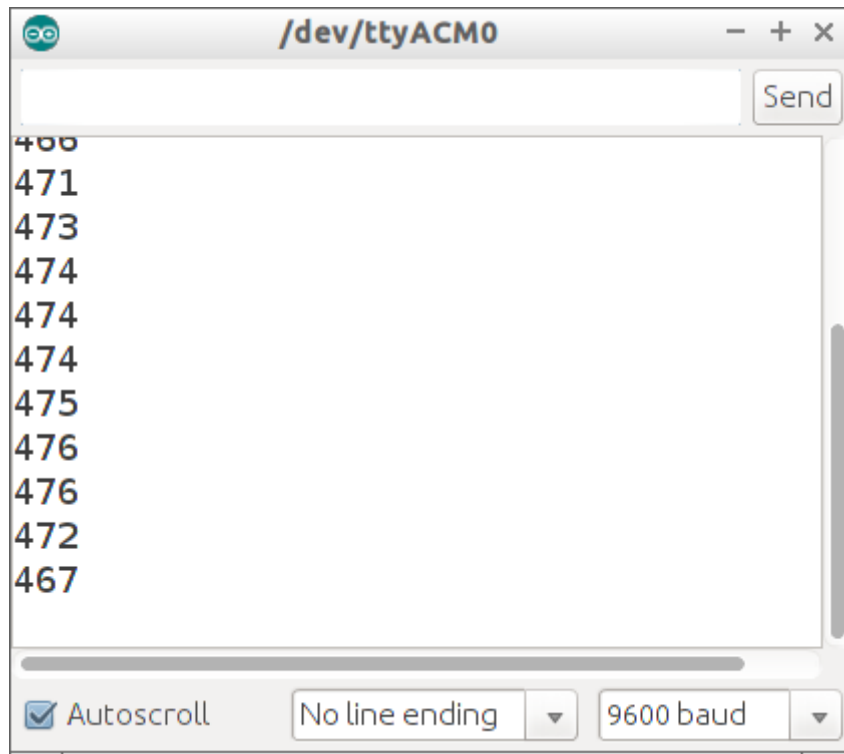


Figure 11: De seriele monitor met getallen

2. Kun je de tekst veranderen naar 'Hallo Richel' (of je eigen naam?)
3. Verander `Serial.println` naar `Serial.print`. Wat zie je?
4. Verander de tekst `Serial.begin(9600)` naar `Serial.begin(4800)`. Wat zie je? Waarom?

## Antwoorden 1

1. De seriele monitor laat elke second een extra regel zien, met de tekst 'Hallo'
2. Verander de regel `Serial.println("Hallo");` naar `Serial.println("Hallo Richel");`
3. De woorden komen na elkaar, in plaats van onder elkaar
4. Nu laat de seriele monitor onleesbare tekst zien. Dit komt omdat de Arduino langzamer tekst stuur naar je computer (4800), dan je computer de tekst leest (9600)

## Aansluiten FSR zonder LED

---



FSR betekent 'Force Sensitive Resistance'

---

Eerst sluiten we alleen een FSR aan:

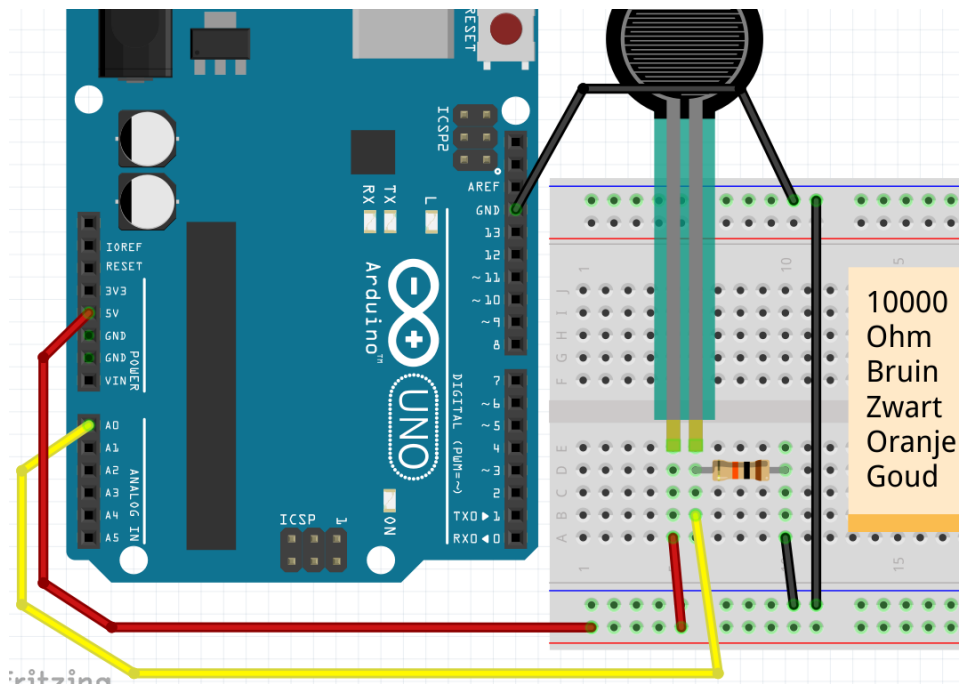


Figure 12: Stroomschema



Is er geen FSR, gebruik dan een LDR

---

## Code: lezen FSR met seriele monitor

Met deze code meten we de waarde van de FSR:

```
void setup()
{
  pinMode(A0, INPUT);
  Serial.begin(9600);
}

void loop()
{
  Serial.println(analogRead(A0));
  delay(100);
}
```



---

`Serial.println(analogRead(A0))` ‘Lieve computer, laat de waarde van pin A0 op de seriele monitor zien’

---

## Opdrachten 2

1. Upload het programma. In de Arduino IDE, klik rechtsboven op ‘Seriele Monitor’. Wat zie je?
2. Druk de FSR in met je vingers (of, met een LDR: houd je vinger boven de LDR) terwijl je de seriele monitor bekijkt. Wat zie je?
3. Verander `Serial.println` naar `Serial.print`. Wat zie je?
4. Verander de tekst `Serial.begin(9600)` naar `Serial.begin(4800)`. Wat zie je? Waarom?
5. Haal de draad naar A0 weg. Ja, haal de draad tussen A0 en de LDR weg. Kijk op de seriele monitor. Wat zie je?



De weerstand tussen A0 en LDR een ‘Pull Down’ weerstand genoemd

---

## Oplossingen 2

1. Je ziet een getal van nul tot 1024, afhankelijk van de waarde van de FSR
2. Je zit de getallen veranderen
3. Alle getallen komen na elkaar
4. Nu laat de seriële monitor onleesbare tekst zien. Dit komt omdat de Arduino langzamer tekst stuurt naar je computer (4800), dan je computer de tekst leest (9600)
5. Nu zie je het getal willekeurig veranderen. Dit wordt een zwevende input genoemd



Een 'Pull Down' weerstand voorkomt een zwevende input

---

## Aansluiten FSR met LED, aan/uit



'Force Sensitive Resistance' betekent 'Kracht afhankelijke weerstand'

---

Nu sluiten we ook een LED aan:

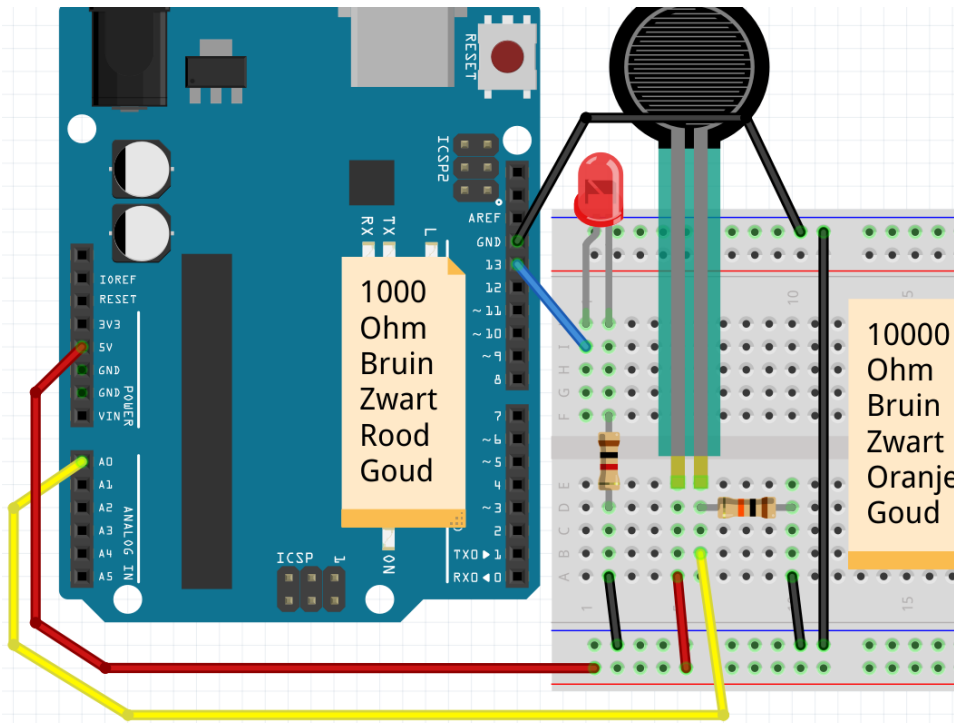


Figure 13: Stroomschema

## Reageren op FSR, aan/uit

Nu gaan we het LEDje laten reageren op de LED:

```
void setup()
{
  pinMode(A0, INPUT);
  pinMode(13, OUTPUT);
}

void loop()
{
  if (analogRead(A0) < 512)
  {
    digitalWrite(13, HIGH);
  }
  else
  {
    digitalWrite(13, LOW);
  }
  delay(100);
}
```





---

```
if (analogRead(A0) < 512) {  
    'Lieve computer, als op A0 minder dan 2,5 Volt staat, doe  
    dan datgeen tussen accolades'  
}
```

---

### Opdrachten 3

1. Wat gebeurt er als je 512 hoger zet? Wat gebeurt er als je 512 lager zet?
2. Zorg dat de seriele monitor ook A0 meet en laat zien. Welk getal meet de FSR in rust?
3. Zorg dat de seriele monitor het woord AAN laat zien als de LED aan gaat, en het woord UIT als de LED uit wordt gezet

## Oplossingen 3

1. Als 512 wordt veranderd naar een te hoog getal, is het lampje altijd aan, hoe hard/zacht je ook drukt. Als 512 wordt veranderd naar een te laag getal, is het lampje altijd uit, hoe hard/zacht je ook drukt
2. Hiervoor gebruik je de code van de vorige opdracht: voeg in de `setup` function toe `Serial.begin(9600);`, in de `loop` functie voeg je `Serial.println(analogRead(A0));` toe. De waarde die je gaat zien is afhankelijk van de weerstand, FSR en situatie
3. Dit kan door `Serial.println("AAN");` in het eerste gedeelte van het `if` statement te zetten. Zet `Serial.println("UIT");` in het tweede gedeelte van het `if` statement.

```
void setup()
{
  pinMode(A0, INPUT);
  pinMode(13, OUTPUT);
  Serial.begin(9600);
}

void loop()
{
  Serial.println(analogRead(A0));
  if (analogRead(A0) < 512)
  {
    digitalWrite(13, HIGH);
    Serial.println("AAN");
  }
  else
  {
    digitalWrite(13, LOW);
    Serial.println("UIT");
  }
  delay(100);
}
```

## Reageren op FSR, dimmen

Nu gaan we het LEDje laten reageren op de LED. Dit keer *dimt* het LEDje.

```
void setup()
{
  pinMode(A0, INPUT);
  pinMode( 9, OUTPUT);
  Serial.begin(9600);
}

void loop()
{
  const int fsr_waarde = analogRead(A0);
  Serial.print("FSR: ");
  Serial.println(fsr_waarde);
  const int led_waarde = map(fsr_waarde, 0, 1023, 0, 255);
  Serial.print("LED: ");
  Serial.println(led_waarde);
  analogWrite(led_waarde, 9);
  delay(100);
}
```



---

```
analogWrite( 0,
             9)
```

‘Lieve computer, zet pin 9 uit’

```
analogWrite(128,
             9)
```

‘Lieve computer, zet pin 9 halfvol aan’

```
analogWrite(255,
             9)
```

‘Lieve computer, zet pin 9 vol aan’

`map(analogRead(A0), 0, 1023, 0, 255)`  
Lieve computer, lees de spanning van A0. Dit is een waarde van 0 tot en met 1023. Bouw de gelezen waarde om tussen 0 en 255.’

---

## Opdrachten 4

1. Het LEDje zit op een andere pin. Kijk in de code en sluit de LED aan op de juiste pin
2. Welke pinnen kunnen we gebruiken om een LEDje te dimmen?

## Oplossingen 4

1. Het LEDje moet nu op pin 9 aangesloten worden
2. Alle pinnen met een golfje (~) voor het getal. Dit zijn 3, 5, 6, 9, 10, 11.

## Opdracht 5

Sluit twee LEDjes aan op pinnen 12 en 13. Als de FSR in rust is, moet er geen LEDje branden. Als je de FSR zacht indrukt, gaat er een LEDje branden. Als je de FSR hard indrukt twee.



Tip: gebruik twee `if` statements

---

## Oplossing 5

De getallen in de if statement moeten goed ingesteld worden.

```
void setup()
{
  pinMode(A0, INPUT);
  pinMode(12, OUTPUT);
  pinMode(13, OUTPUT);
  Serial.begin(9600);
}

void loop()
{
  Serial.println(analogRead(A0));
  if (analogRead(A0) < 256)
  {
    digitalWrite(13, HIGH);
  }
  if (analogRead(A0) < 512)
  {
    digitalWrite(12, HIGH);
  }
  delay(100);
}
```

## Opdracht 6

Je kunt een LEDje ook laten reageren op een FSR door deze te faden/dimmer

1. Met welk commando deed je dat ook alweer?
2. Kan dat met elke pin? Zo nee, met welke wel/niet?
3. Wat is de hoogste waarde waarmee je een LEDje kunt laten branden?
4. Wat is de hoogste waarde die de FSR kan meten?
5. Stel je wil een LED laten branden afhankelijk van een FSR waarde. Hoe zou je dit kunnen doen?
6. Hoe laat je code een deling doen?
7. Laat de LED branden afhankelijk van de FSR waarde

## Oplossingen 6

1. Een LEDje kun je laten faden met `analogWrite`, bijvoorbeeld `analogWrite(11, 255);`
2. Je kunt een LEDje alleen laten dimmen met PWM pinnen. Dit zijn de pinnen met een golfje (~) naast hun getal. Op de Arduino Uno zijn dit de pinnen 3, 5, 6, 9, 10 en 11
3. Met `analogWrite` kun je maximaal 255 geven, bijvoorbeeld `analogWrite(11, 255);`
4. Met `analogRead` kun je maximaal 1023 meten
5. Je leest een waarde, deelt deze door vier (1024 gedeeld door 256 is vier) en laat de LED zo hard branden
6. Met de deelstreep, /.
7. Zie hieronder. Vergeet niet een LEDje op pin 11 te zetten

```
void setup()
{
  pinMode(A0, INPUT);
  pinMode(11, OUTPUT);
  Serial.begin(9600);
}

void loop()
{
  analogWrite(11, analogRead(A0) / 4);
  delay(100);
}
```



---

```
analogWrite(11,
analogRead(A0) /
4)
```

‘Lieve computer, dim pin 11 op de waarde van pin A0 (dit moet je delen door vier)’

---

## Eindopdracht

- Sluit vier LEDjes aan: een witte, een rode, een gele en een groene
- Als de FSR in rust is, moet er geen LEDje branden.
- Als je de FSR zacht indrukt gaat het groene LEDje branden
- Als je de FSR harder indrukt gaan de groene en gele LEDjes branden
- Als je de FSR hard indrukt gaan de groene, gele en rode LEDjes branden
- Het witte LEDje gaat harder en zachter branden afhankelijk van de FSR

Als je geen wit LEDje hebt, gebruik dan een andere kleur.

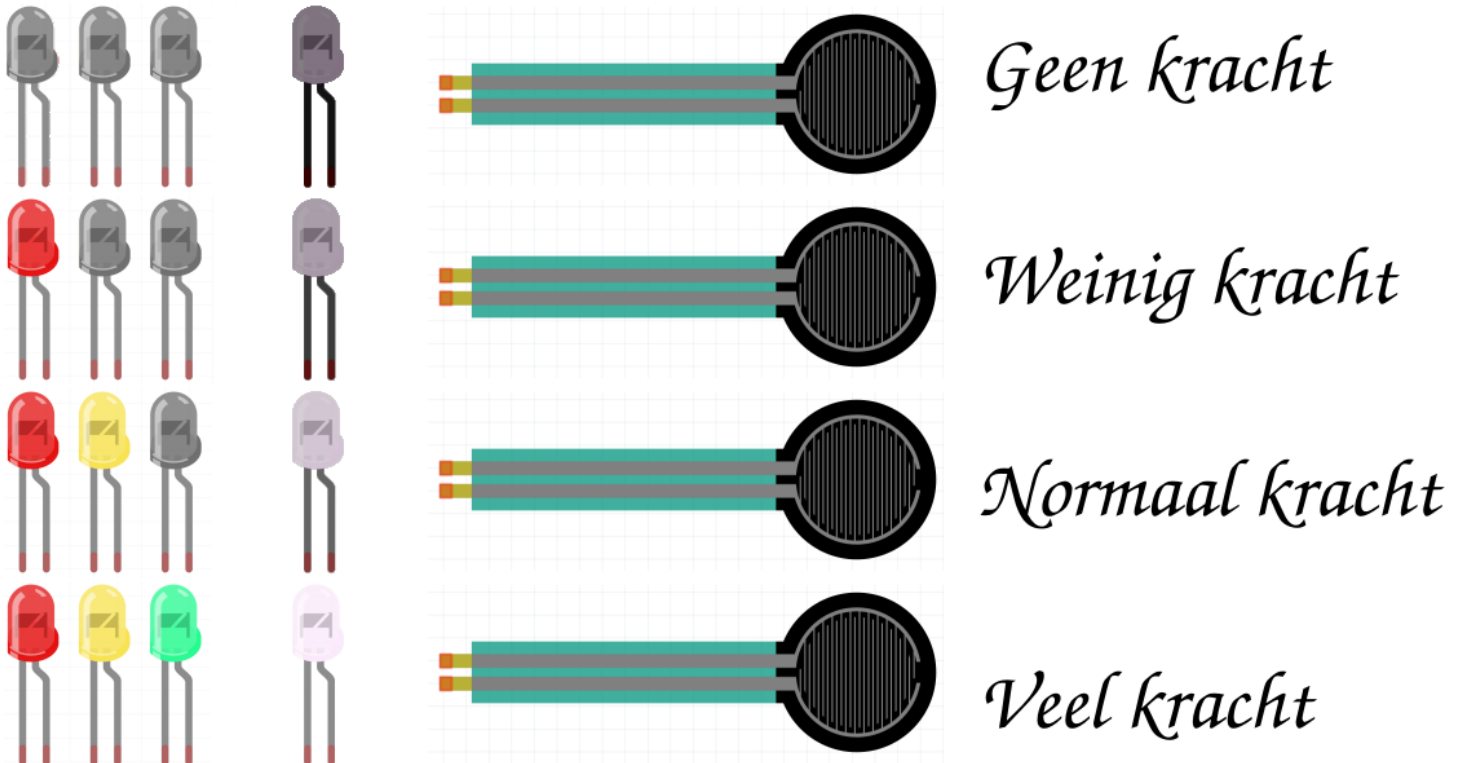
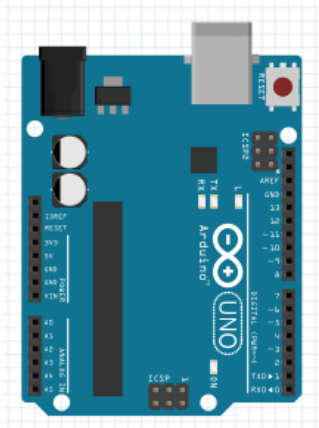


Figure 14: FSR eindopdracht

## 9. LDR

### Alleen Arduino aansluiten

Eerst sluiten we alleen een Arduino aan:



Ik denk dat dit wel moet lukken :-)

### Code: seriele monitor

```
void setup()
{
  Serial.begin(9600);
}

void loop()
{
  Serial.println("Hallo");
  delay(1000);
}
```



Vergeten wat dit is? Ga naar les 4 'FSR'

---

### Opdrachten 1

1. Upload het programma. In de Arduino IDE, klik rechtsboven op 'Seriele Monitor'. Wat zie je?
2. Kun je de tekst veranderen naar 'Hallo Richel' (of je eigen naam?)
3. Verander `Serial.println` naar `Serial.print`. Wat zie je?



4. Verander de tekst `Serial.begin(9600)` naar `Serial.begin(4800)`. Wat zie je? Waarom?

## Oplossingen 1

1. De seriele monitor laat elke second een extra regel zien, met de tekst 'Hallo'
2. Verander de regel `Serial.println("Hallo");` naar `Serial.println("Hallo Richel");`
3. De woorden komen na elkaar, in plaats van onder elkaar
4. Nu laat de seriele monitor onleesbare tekst zien. Dit komt omdat de Arduino langzamer tekst stuur naar je computer (4800), dan je computer de tekst leest (9600)

## Aansluiten LDR zonder LED



LDR betekent 'Light Dependent Resistance'

Eerst sluiten we alleen een LDR aan:

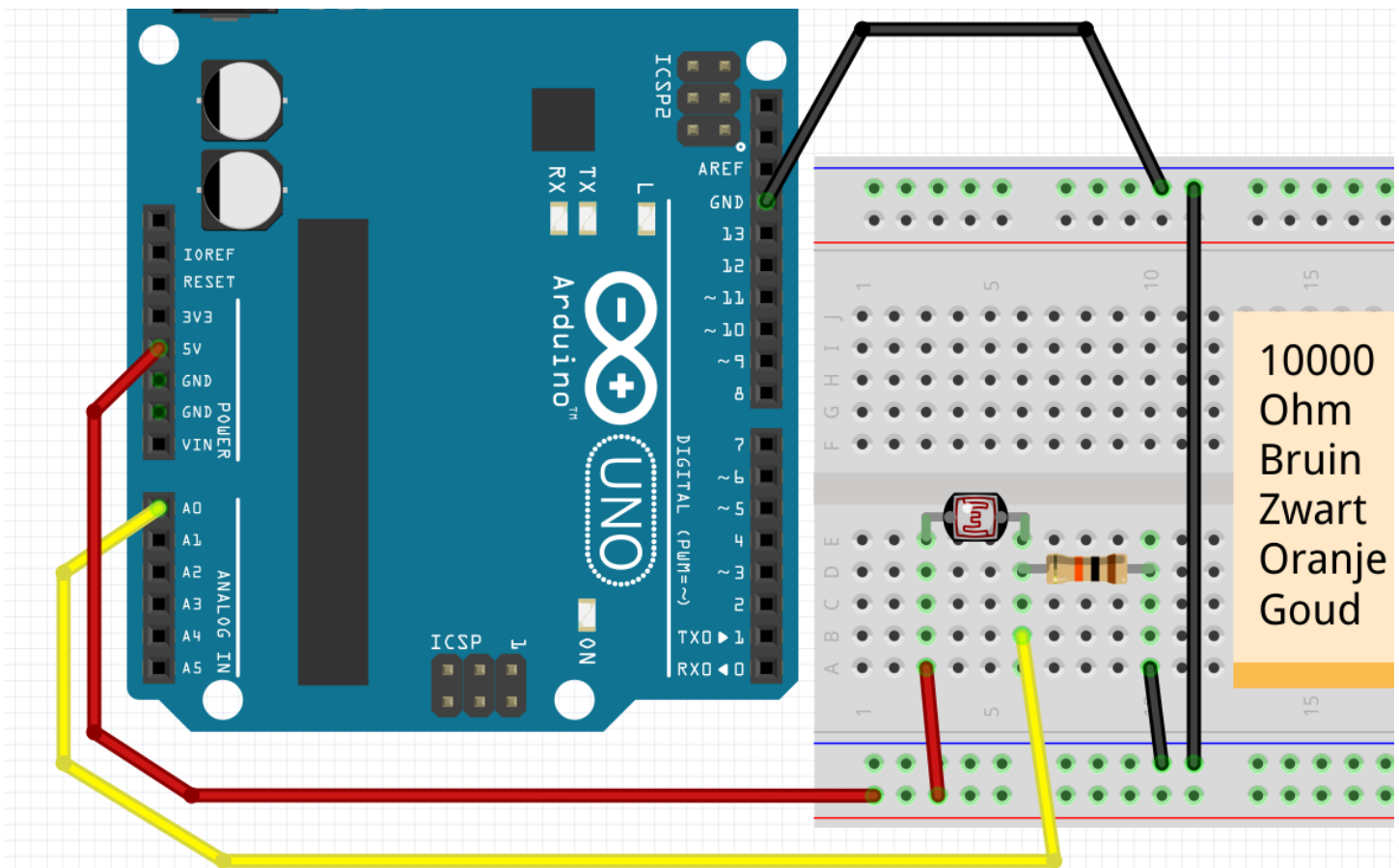


Figure 15: Stroomschema

## Code: lezen FSR met seriele monitor

Met deze code meten we de waarde van de LDR:

```
void setup()
{
  pinMode(A0, INPUT);
  Serial.begin(9600);
}

void loop()
{
  Serial.println(analogRead(A0));
  delay(100);
}
```



Vergeten wat dit is? Ga naar les 4 'FSR'

---

## Opdrachten 2

1. Upload het programma. In de Arduino IDE, klik rechtsboven op 'Seriele Monitor'. Wat zie je?
2. Houd je vinger boven de LDR terwijl je de seriele monitor bekijkt. Wat zie je?
3. Verander `Serial.println` naar `Serial.print`. Wat zie je?
4. Verander de tekst `Serial.begin(9600)` naar `Serial.begin(4800)`. Wat zie je? Waarom?
5. Haal de draad naar A0 weg. Ja, haal de draad tussen A0 en de LDR weg. Kijk op de seriele monitor. Wat zie je?

## Oplossingen 2

1. Je ziet een getal van nul tot 1024, afhankelijk van de waarde van de LDR
2. Je zit de getallen veranderen
3. Alle getallen komen na elkaar
4. Nu laat de seriële monitor onleesbare tekst zien. Dit komt omdat de Arduino langzamer tekst stuurt naar je computer (4800), dan je computer de tekst leest (9600)
5. Nu zie je het getal willekeurig veranderen. Dit wordt een zwevende input genoemd



Heb je een zwevende input? Dan heb je vaak iets fout aangesloten

---

## Aansluiten LDR met LED

---



‘Light Dependent Resistance’ betekent ‘lichtafhankelijke weerstand’

---

Nu sluiten we ook een LED aan:

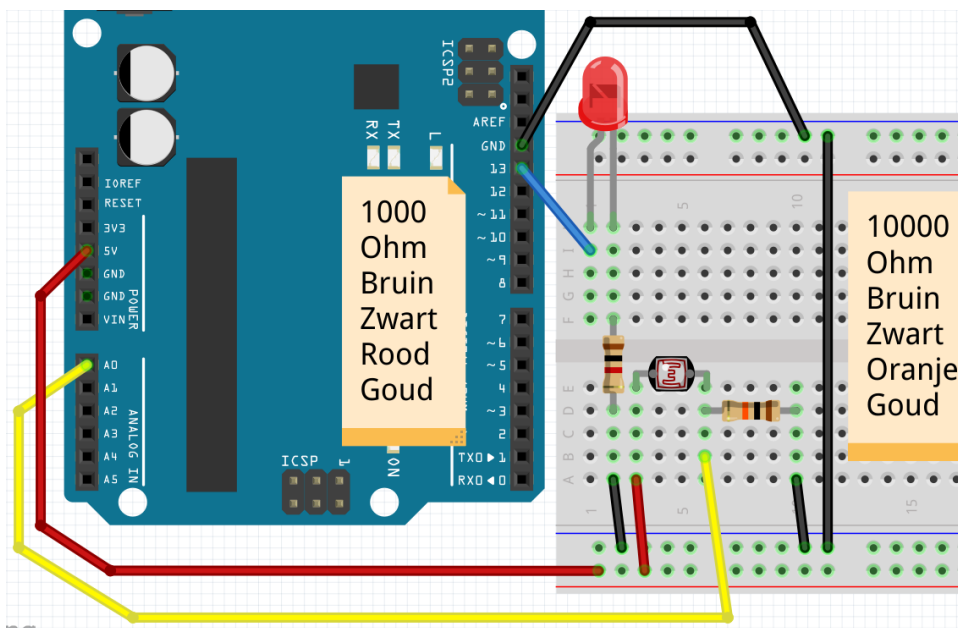


Figure 16: Stroomschema

## Reageren op LDR

Nu gaan we het LEDje laten reageren op de LDR:

```
void setup()
{
  pinMode(A0, INPUT);
  pinMode(13, OUTPUT);
}

void loop()
{
  if (analogRead(A0) < 512)
  {
    digitalWrite(13, HIGH);
  }
  else
  {
    digitalWrite(13, LOW);
  }
  delay(100);
}
```



Vergeten wat dit is? Ga naar les 4 'FSR'

---

## Opdracht 3

1. Wat gebeurt er als je 512 hoger zet? Wat gebeurt er als je 512 lager zet?
2. Zorg dat de seriele monitor ook A0 meet en laat zien. Welk getal meet de FSR in rust?
3. Zorg dat de seriele monitor het woord AAN laat zien als de LED aan gaat, en het woord UIT als de LED uit wordt gezet

## Oplossingen 3

1. Als 512 wordt veranderd naar een te hoog getal, is het lampje altijd aan, hoe hard/zacht je ook drukt. Als 512 wordt veranderd naar een te laag getal, is het lampje altijd uit, hoe hard/zacht je ook drukt
2. Hiervoor gebruik je de code van de vorige opdracht: voeg in de `setup` function toe `Serial.begin(9600);`, in de `loop` functie voeg je `Serial.println(analogRead(A0));` toe. De waarde die je gaat zien is afhankelijk van de weerstand, LDR en hoeveelheid licht
3. Dit kan door `Serial.println("AAN");` in het eerste gedeelte van het `if` statement te zetten. Zet `Serial.println("UIT");` in het tweede gedeelte van het `if` statement.

```
void setup()
{
  pinMode(A0, INPUT);
  pinMode(13, OUTPUT);
  Serial.begin(9600);
}

void loop()
{
  Serial.println(analogRead(A0));
  if (analogRead(A0) < 512)
  {
    digitalWrite(13, HIGH);
    Serial.println("AAN");
  }
  else
  {
    digitalWrite(13, LOW);
    Serial.println("UIT");
  }
  delay(100);
}
```

## Opdracht 4

Sluit een extra LEDje aan. Als de LDR in normaal licht is, moet er geen LEDje branden. Als je de LDR een beetje donkerder maakt met je hand, gaat er een LEDje branden. Als je de LDR helemaal donker maakt twee.



Tip: gebruik twee `if` statements

---

## Oplossing 4

De getallen in de if statement moeten goed ingesteld worden.

```
void setup()
{
  pinMode(A0, INPUT);
  pinMode(12, OUTPUT);
  pinMode(13, OUTPUT);
  Serial.begin(9600);
}

void loop()
{
  Serial.println(analogRead(A0));
  if (analogRead(A0) < 256)
  {
    digitalWrite(13, HIGH);
  }
  if (analogRead(A0) < 512)
  {
    digitalWrite(12, HIGH);
  }
  delay(100);
}
```



## Eindopdracht

- Sluit drie LEDjes aan: een rode, gele en groene
- Als de LDR in het licht is, moet er geen LEDje branden.
- Als je de LDR een beetje verduisterd wordt, gaat het groene LEDje branden
- Als je de LDR meer verduisterd wordt, gaan de groene en gele LEDjes branden
- Als je de LDR helemaal verduisterd wordt, gaan alle LEDjes branden



Figure 17: LDR eindopdracht



Tip: gebruik drie of vier keer `if`

---