



Les 1: Blink
Les 2: Blink Blink Blink
Les 3: Knop if else
Les 4: Knop flip

Figure 1: Boek 1: LED en knop

Contents

Voorwoord	1
Les 1: Blink	2
Les 2: Blink Blink Blink	11
3. Knop if ... else	17
2. Knop if ... else	24

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. 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 1: Blink

In deze les gaan we de allergemakkelijkste schakeling maken die er is: Blink!



We beginnen makkelijk!

1.1 Blink: een Arduino aansluiten

Sluit een Arduino zo aan:

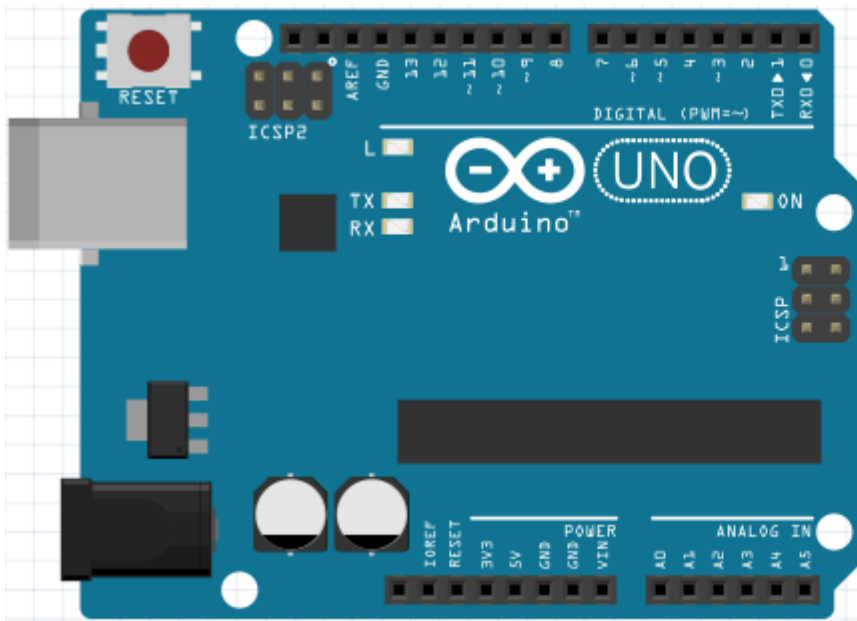


Figure 2: Blink zonder led

Doe een USB kabel in je Arduino en in je laptop.

1.2 Blink: opdracht 1

Sluit de Arduino aan.



De Arduino heeft zelf al een lampje dat je kunt programmeren.

1.3 Blink: de Arduino IDE opstarten



We programmeren de Arduino met de Arduino IDE



‘IDE’ spreek je uit als ‘ie-dee-ee’

Start de Arduino IDE door:

- Klik op de snelkoppeling op het Bureaublad
- Druk op de Windows toets (linksonder, tussen **Ctrl** en **Alt**). Type dan **arduino** (kleine letters) en dan Enter

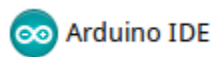


Figure 3: Logo van de Arduino IDE



Figure 4: Win toets

Je ziet nu je de Arduino IDE:

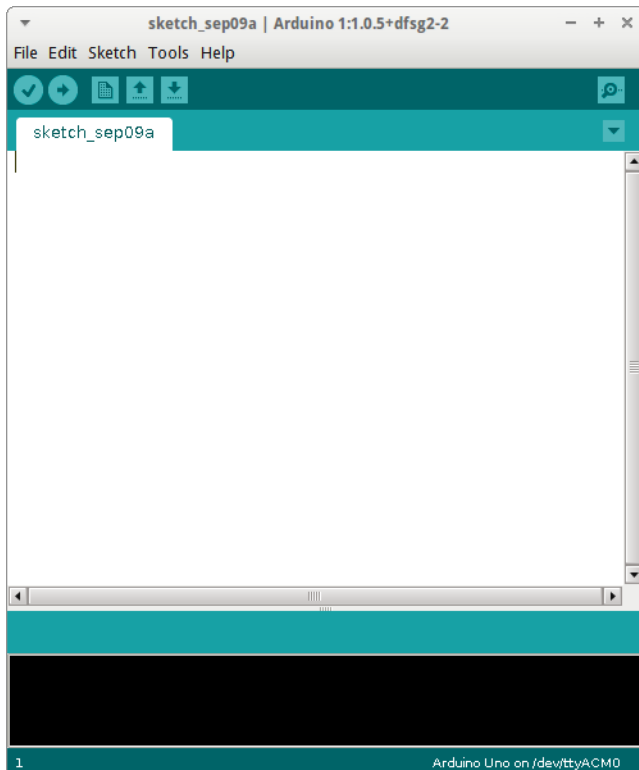


Figure 5: De Arduino IDE



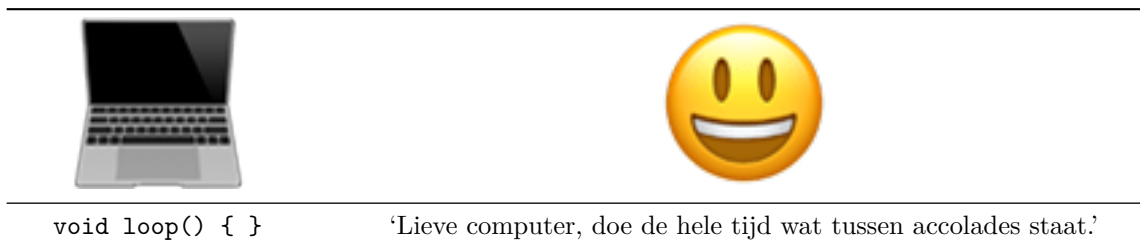
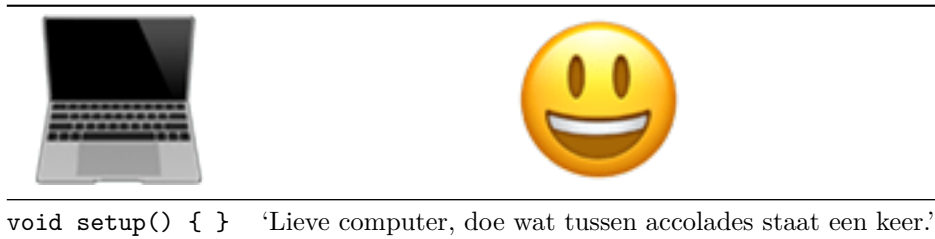
IDE betekent ‘Integrated Development Environment’, het programma waarmee je programmeert.

1.4 Blink: ons eerste programma

Dit is (een versie van) de code van Blink:

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

void loop()
{
  digitalWrite(13, HIGH);
  delay(1000);
  digitalWrite(13, LOW);
  delay(1000);
}
```



Type de code over in de Arduino IDE en klik op Upload.

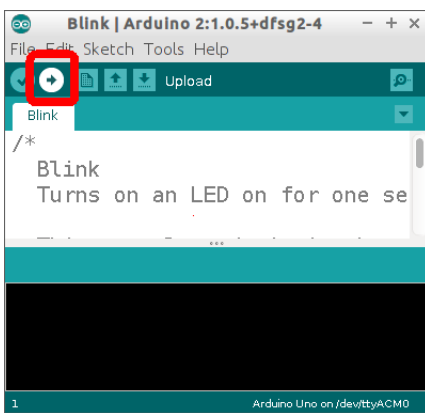


Figure 6: Hier kun je op Upload klikken

1.5 Blink: opdracht 2

Laat het LEDje 10x zo snel knipperen. Hint: het LEDje is nu 1000 milliseconden aan en 1000 milliseconden uit.

1.6 Blink: oplossing 2

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

void loop()
{
  digitalWrite(13, HIGH);
  delay(100);
  digitalWrite(13, LOW);
  delay(100);
}
```



`pinMode(13, OUTPUT);`

‘Lieve computer, zorg dat er stroom uit pin 13 kan komen.’



`digitalWrite(13, HIGH);`

‘Lieve computer, zet spanning op pin 13.’



`delay(1000);`

‘Lieve computer, doe nu 1000 milliseconden niks.’



`digitalWrite(13, LOW);`

‘Lieve computer, zet geen spanning op pin 13.’

1.7 Blink: opdracht 3

Laat het LEDje zo snel als kan knipperen. Wat zie je?

1.8 Blink: oplossing 3

Er zijn meer oplossingen, dit is er een:

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

void loop()
{
  digitalWrite(13, HIGH);
  delay(1);
  digitalWrite(13, LOW);
  delay(1);
}
```



Ook goed: `delay(0);` of de regels met `delay` helemaal weghalen.

Je ziet het lampje niet meer knipperen.



Het lampje knippert wel, maar onze ogen kunnen niet snel genoeg kijken om het te zien.

1.9 Blink: LED aansluiten

Nu is het tijd Blink aan te sluiten:

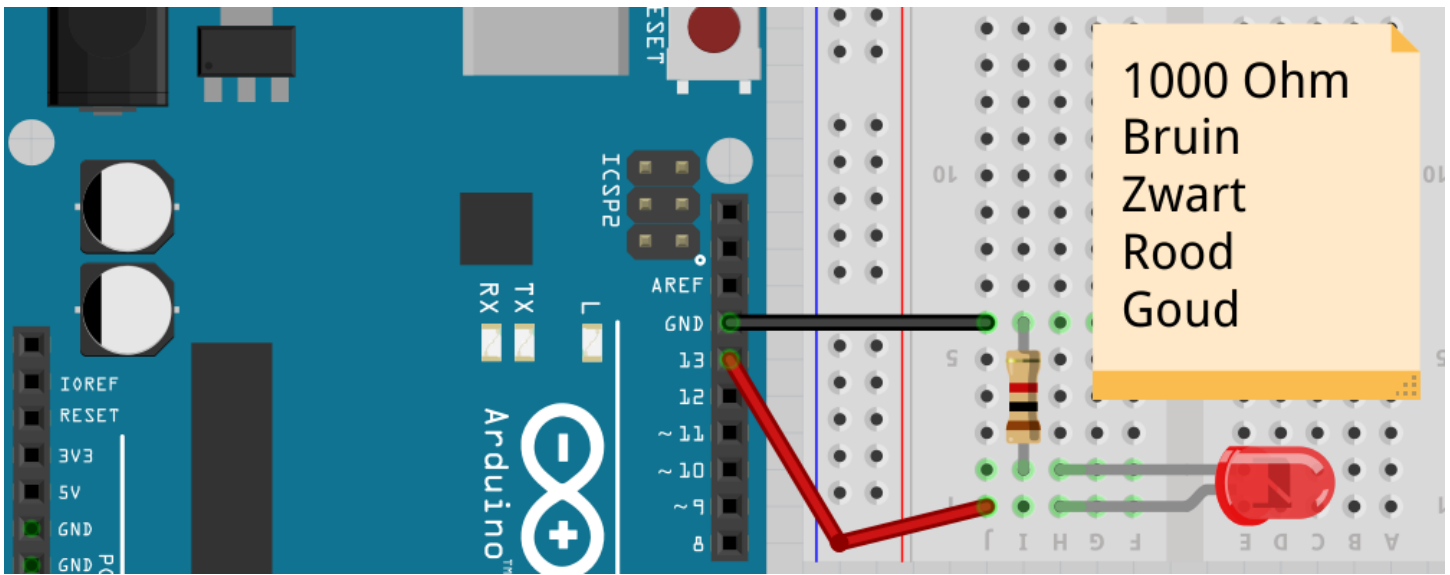


Figure 7: Blink

- Haal het USB snoer uit de computer, zodat de Arduino geen spanning meer heeft
- Sluit Blink aan zoals op de tekening
- Zet de spanning weer op de Arduino
- Als het goed is, knippert het rode LEDje nu mee met het LEDje op de Arduino.



Als het LEDje niet brandt, draai deze dan om.

1.10 Blink: opdracht 4

Sluit je LEDje nu aan op pin 12.



Het is cool om na pin 13 pin 12 te pakken



Dit is ook omdat pinnen 0 en 1 speciaal zijn

1.11 Blink: oplossing 4

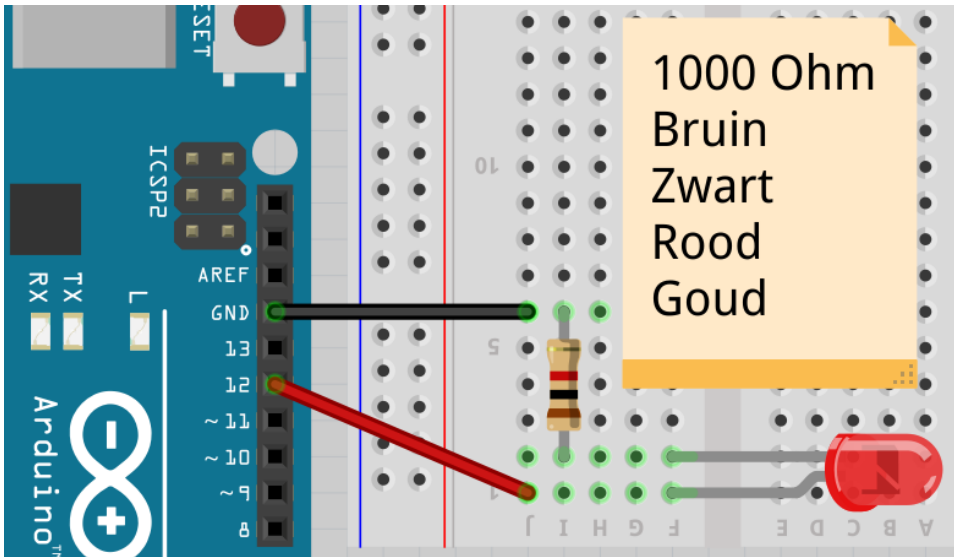


Figure 8: Blink met LEDje op pin 12

1.12 Blink: opdracht 5

Zorg nu dat het LEDje op pin 12 gaat knipperen op de seconde.



LED is een afkorting voor 'Light Emitting Diode'



Goed gezegd! Beter te laat dan nooit!



LED is in het Nederlands: 'Diode waar licht uit komt'

1.13 Blink: oplossing 5

```
void setup()
{
  pinMode(12, OUTPUT);
}

void loop()
{
  digitalWrite(12, HIGH);
  delay(100);
  digitalWrite(12, LOW);
  delay(100);
}
```

1.14 Blink: eindopdracht

Sluit de LED aan op pin 11 en laat deze 2 keer per seconde knipperen.



Gelukt? Laat dit zien aan een volwassene voor een paraaf!

Les 2: Blink Blink Blink

Deze les heet 'Blink Blink Blink', omdat het de les Blink is, maar dan met drie lampjes.

2.1: Blink Blink Blink: Blink

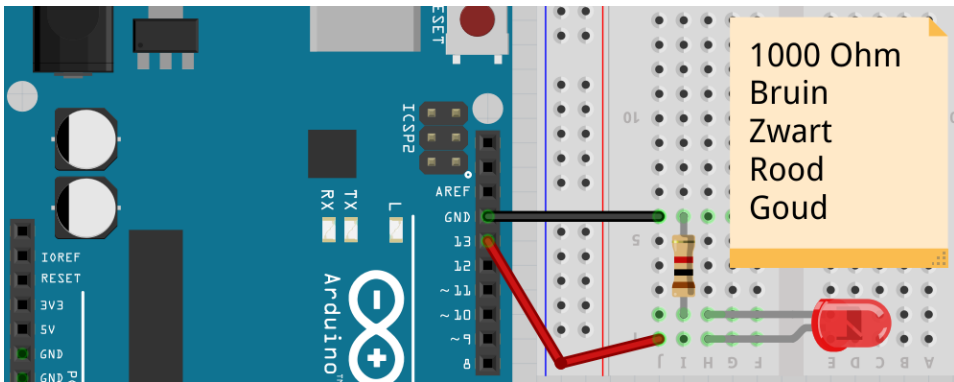


Figure 9: Blink

Dit is de code van Blink op een andere manier:

```
const int pin_led = 13;

void setup()
{
  pinMode(pin_led, OUTPUT);
}

void loop()
{
  digitalWrite(pin_led, HIGH);
  delay(1000);
  digitalWrite(pin_led, LOW);
  delay(1000);
}
```

pin_led wordt een variabele genoemd: een stukje computergeheugen met een naam.



```
const int pin_led = 13;
```



‘Lieve computer, onthoud een heel getal met de naam `pin_led` met beginwaarde 13.’

2.2: Blink Blink Blink: Opdracht 1

Sluit de LED aan op pin 12 en verander de code zodat deze gaat knipperen.

2.3: Blink Blink Blink: Oplossing 1

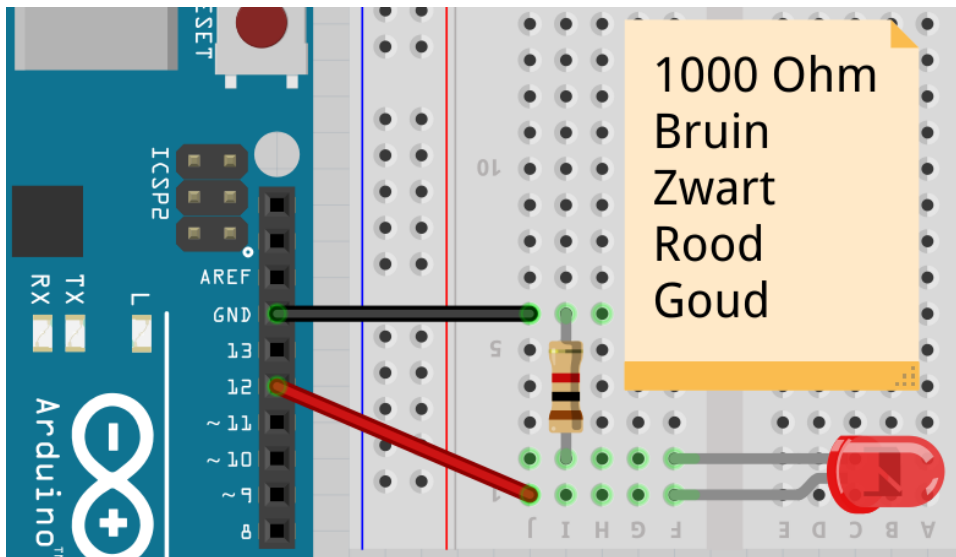


Figure 10: Blink met LEDje op 12

```
const int pin_led = 12;

void setup()
{
  pinMode(pin_led, OUTPUT);
}

void loop()
{
  digitalWrite(pin_led, HIGH);
  delay(1000);
  digitalWrite(pin_led, LOW);
  delay(1000);
}
```

2.4: Blink Blink Blink: Opdracht 2

Maak zelf een nieuwe variabele met de naam `wachttijd`. `wachttijd` is een heel getal met beginwaarde 1000. Gebruik `wachttijd` in de regels met `delay`.



Slim! Als je `wachttijd` leest, weet je waar het voor is. Bij 1000 weet je dat niet

2.5: Blink Blink Blink: Oplossing 2

```
const int pin_led = 12;
const int wachttijd = 1000;

void setup()
{
  pinMode(pin_led, OUTPUT);
}

void loop()
{
  digitalWrite(pin_led, HIGH);
  delay(wachttijd);
  digitalWrite(pin_led, LOW);
  delay(wachttijd);
}
```



Programmeurs gebruiken veel variabelen, omdat de code dan beter te snappen is.

2.6: Blink Blink Blink: Blink Blink Blink aansluiten

Nu is het tijd 'Blink Blink Blink' aan te sluiten:

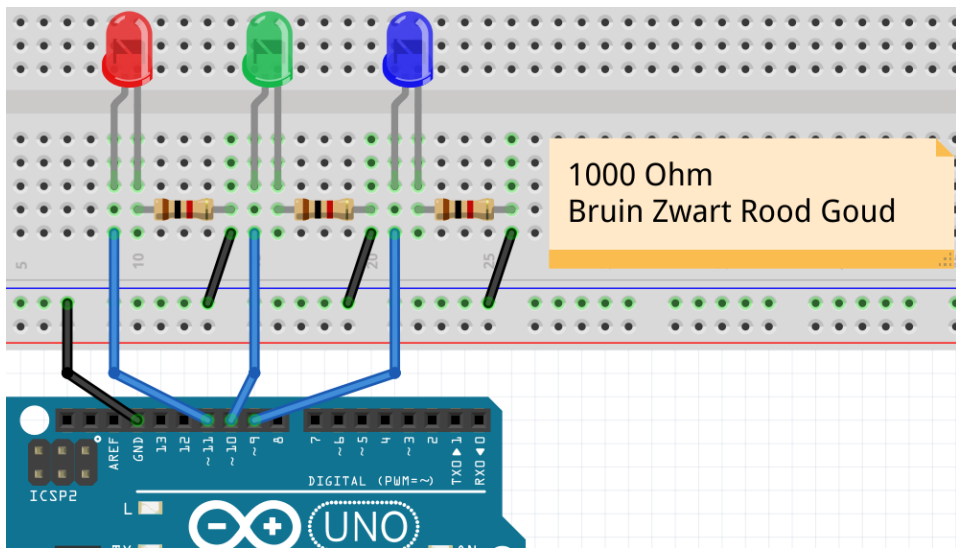


Figure 11: BlinkBlinkBlink

- Haal het USB snoer uit de computer, zodat de Arduino geen spanning meer heeft
- Sluit de onderdelen aan zoals op de tekening

2.7: Blink Blink Blink: Opdracht 3

Sluit 'Blink Blink Blink' aan. Hernoem de variable `pin_led` naar `pin_led_1` en zorg dat 'ie de juiste beginwaarde heeft.

Oplossing 3

```
const int pin_led_1 = 11;
const int wachttijd = 1000;

void setup()
{
  pinMode(pin_led_1, OUTPUT);
}

void loop()
{
  digitalWrite(pin_led_1, HIGH);
  delay(wachttijd);
  digitalWrite(pin_led_1, LOW);
  delay(wachttijd);
}
```

2.8: Blink Blink Blink: Opdracht 4

Maak een nieuwe variable `pin_led_2`. Laat dan eerst het eerste LEDje aan en uit gaan, laat dan het tweede LEDje aan en uit gaan.



De schrijfwijze `pin_led_2` wordt 'snake case' genoemd: 'snake' is Engels voor slang



Zou je schrijven `pinLed2`, dan wordt dat 'camel case' genoemd: 'camel' is Engels voor kameel



Mij maakt het niet uit welke je kiest

2.9: Blink Blink Blink: Oplossing 4

```
const int pin_led_1 = 11;
const int pin_led_2 = 10;
const int wachttijd = 1000;

void setup()
{
  pinMode(pin_led_1, OUTPUT);
  pinMode(pin_led_2, OUTPUT);
}

void loop()
{
  digitalWrite(pin_led_1, HIGH);
  delay(wachttijd);
  digitalWrite(pin_led_1, LOW);
  delay(wachttijd);
  digitalWrite(pin_led_2, HIGH);
  delay(wachttijd);
  digitalWrite(pin_led_2, LOW);
  delay(wachttijd);
}
```

2.10: Blink Blink Blink: Opdracht 5

Maak een derde variabele `pin_led_3`. Laat nu steeds alle lampjes tegelijk knipperen: allemaal aan, dan allemaal uit.



Dit is best veel typewerk! Later leer je hoe dit slimmer kan



Klopt! Met arrays kun je meerdere getallen in een variabele opslaan

2.11: Blink Blink Blink: Oplossing 5

```
const int pin_led_1 = 11;
const int pin_led_2 = 10;
const int pin_led_3 = 9;
const int wachttijd = 1000;

void setup()
{
  pinMode(pin_led_1, OUTPUT);
  pinMode(pin_led_2, OUTPUT);
}

void loop()
{
  digitalWrite(pin_led_1, HIGH);
  digitalWrite(pin_led_2, HIGH);
  digitalWrite(pin_led_3, HIGH);
  delay(wachttijd);
  digitalWrite(pin_led_1, LOW);
  digitalWrite(pin_led_2, LOW);
  digitalWrite(pin_led_3, LOW);
  delay(wachttijd);
}
```

2.12: Blink Blink Blink: Eindopdracht

Laat de lampjes nu in een ‘Knight Rider patroon’ gaan: 1-2-3-2. Er moet altijd precies een lampje branden.



Knight Rider was een TV serie met een pratende auto.

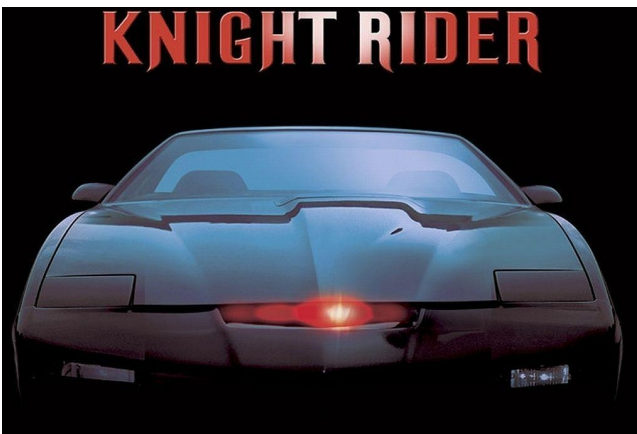


Figure 12: Knight Rider

3. Knop if ... else

In deze les laten we de Arduino vragen aan zichzelf stellen.



De Arduino wordt dus een soort filosoof!

3.1. Knop if ... else: Aansluiten

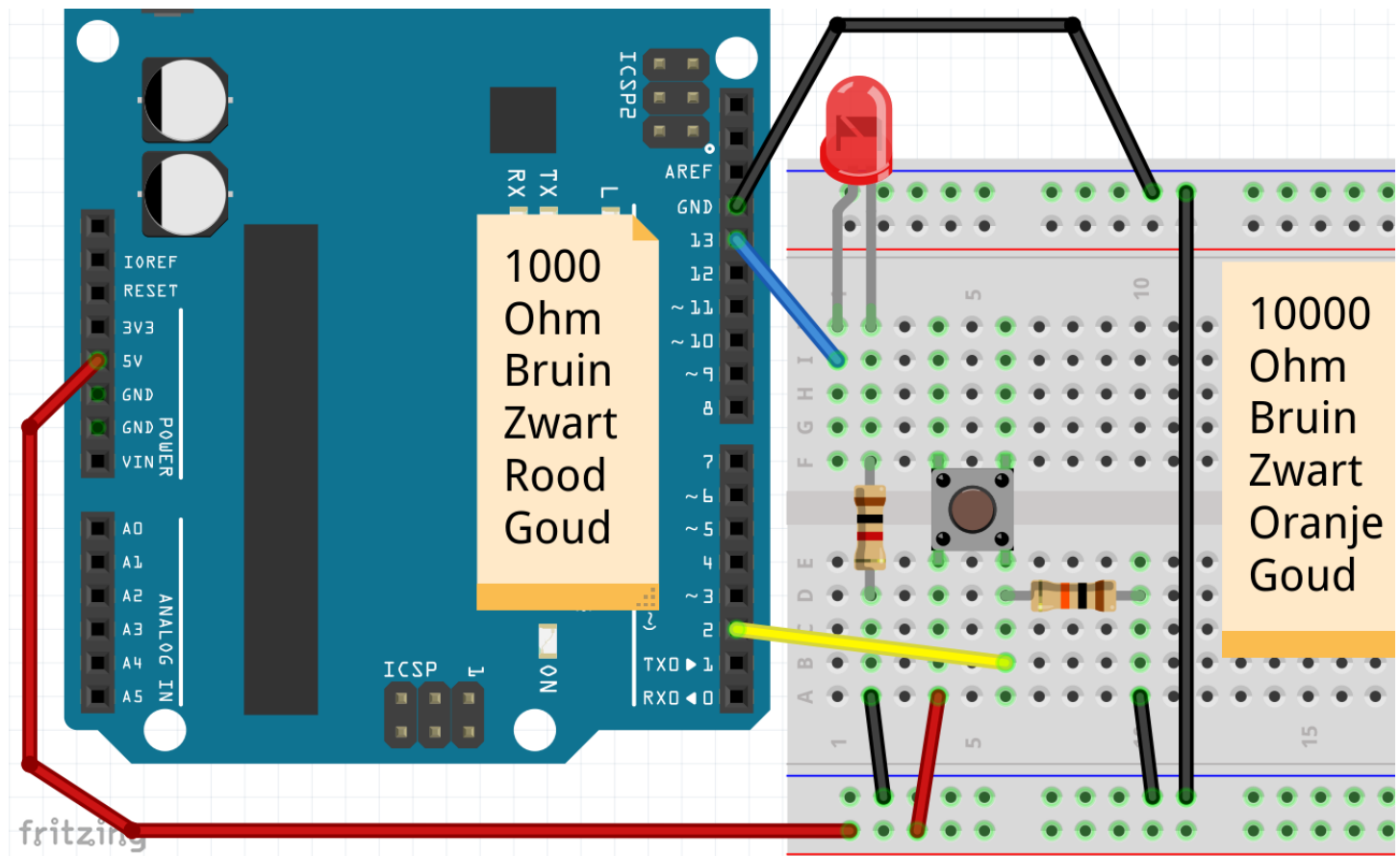


Figure 13: Stroonschema



De weerstand aan de knop wordt een 'Pull Down' weerstand genoemd

3.2. Knop if ... else: Code

Hier is code om het lampje aan te laten gaan, als de knop wordt ingedrukt:

```
void setup()
{
  pinMode(13, OUTPUT);
  pinMode( 2, INPUT );
}

void loop()
{
  digitalWrite(13, digitalRead(2));
}
```



`digitalRead(2)`

‘Lieve computer, geef mij HIGH als er op pin 2 spanning staat.
Geef anders LOW’

`digitalWrite(13, digitalRead(2))`

‘Zet op pin 13 spanning als er op pin 2 spanning staat’



De ‘Pull Down’ weerstand zorgt dat pin 2 verbonden is met GND als de knop niet ingedrukt is

3.3. Knop if ... else: Vragen

- 1. Als je de knop indrukt, gaat de lamp dan uit of aan?
- 2. Er is een regel `pinMode(2, INPUT)`. Waarom staat er een spatie voor de 2? Mag die spatie weg?

3.4. Knop if ... else: Antwoorden

- 1. De knop gaat dan aan
- 2. Er staat een spatie, omdat dit mooier eruit ziet met de regel erboven. De spatie mag weg.



Goede programmeurs werken netjes

3.5. Knop if ... else: if

Met if kun je de Arduino iets laten doen, als iets zo is:

```
if (digitalRead(4) == HIGH)
{
    digitalWrite(5, HIGH);
}
else
{
    digitalWrite(6, LOW);
}
```

In deze code wordt gekeken of op pin 4 spanning staat. Zo ja, dan zet de Arduino spanning op pin 5. Anders (else) haalt de Arduino de spanning van pin 6 af.



if (digitalRead(4) == HIGH) {}



‘Lieve computer, als er spanning op pin 4 staat, doe dan hetgeen tussen accolades’



De = kun je uitspreken als ‘zet op’. De == kun je uitspreken als ‘is gelijk aan’

3.6. Knop if ... else: Opdracht 1

```
void setup()
{
  pinMode(4, INPUT );
  pinMode(5, OUTPUT);
  pinMode(6, OUTPUT);
}

void loop()
{
  if (digitalRead(4) == HIGH)
  {
    digitalWrite(5, HIGH);
  }
  else
  {
    digitalWrite(6, LOW);
  }
}
```

Maak de code zo, dat:

- als je op de knop drukt, dat het lampje aan gaat
- als je op de knop niet indrukt, dat het lampje uit gaat



Na de ronde haken van `if` komt geen puntkomma

3.7. Knop if ... else: Oplossing 1

```
void setup()
{
  pinMode(13, OUTPUT);
  pinMode( 2, INPUT );
}

void loop()
{
  if (digitalRead(2) == HIGH)
  {
    digitalWrite(13, HIGH);
  }
  else
  {
    digitalWrite(13, LOW);
  }
}
```

3.8. Knop if ... else: Opdracht 2

Sluit een tweede LEDje aan. Maak de code zo, dat:

- als je op de knop drukt, dat het eerste lampje aan gaat en het tweede LEDje uit
- als je op de knop niet indrukt, dat het eerste lampje uit gaat en het tweede LEDje uit



Binnen de accolades van een **if** kun je meerdere regels typen. Net als tussen de accolades van **setup** en **loop**!



Vergeet de **else** ('doe anders') niet!

3.9. Knop if ... else: Oplossing 2

Figuur Oplossing van 'Knop met twee LEDjes' laat zien hoe je dit aan moet sluiten.

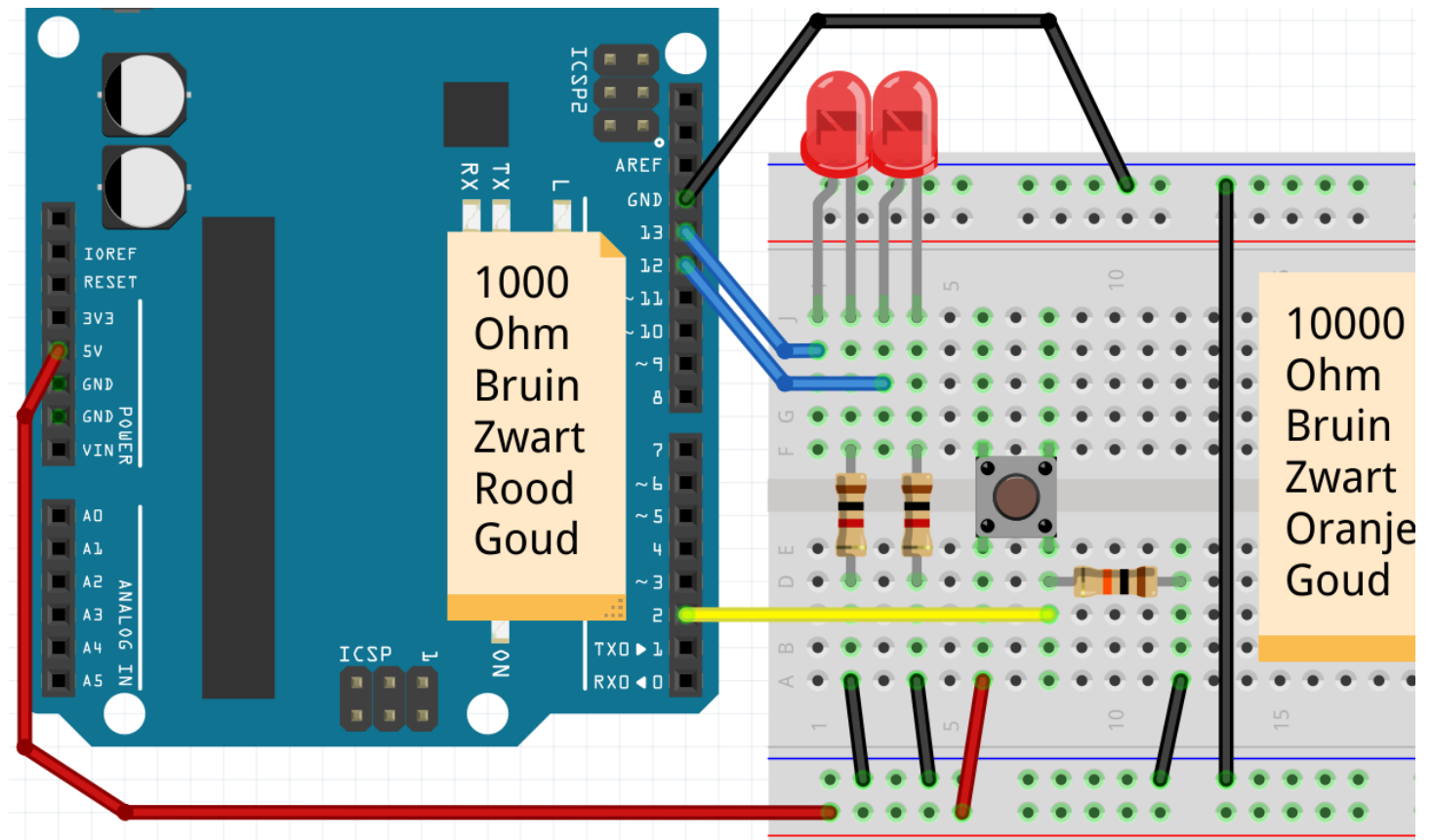


Figure 14: Oplossing van 'Knop met twee LEDjes'

Dit is de code:

```
void setup()
{
  pinMode(13, OUTPUT);
  pinMode(12, OUTPUT);
  pinMode( 2, INPUT );
}

void loop()
{
  if (digitalRead(2) == HIGH)
  {
    digitalWrite(12, LOW );
    digitalWrite(13, HIGH);
  }
  else
  {
    digitalWrite(12, HIGH);
    digitalWrite(13, LOW );
  }
  delay(10);
}
```

3.10. Knop if ... else: Eindopdracht

Sluit een tweede knop aan. Maak de code zo, dat

- als je op de eerste knop drukt, dat het eerste lampje aan gaat
- als je de eerste knop niet indrukt, dat het eerste lampje uit gaat
- als je op de tweede knop drukt, dat het tweede lampje uit gaat
- als je de tweede knop niet indrukt, dat het tweede lampje aan gaat



Je kunt vaker `if` na elkaar zetten



Voor een tweede knop heb je een tweede weerstand van tienduizend Ohm nodig

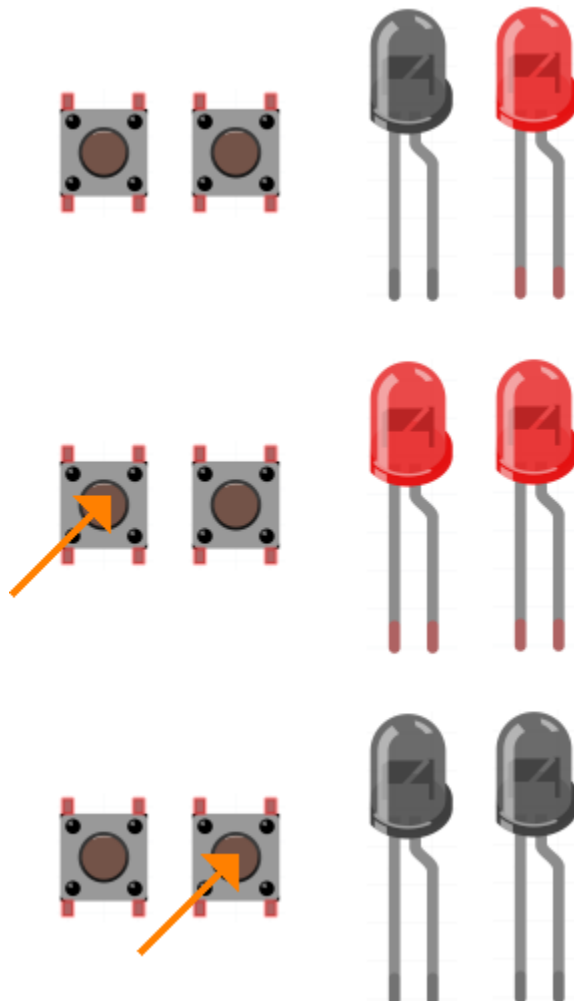


Figure 15: Eindopdracht

2. Knop if ... else

Aansluiten

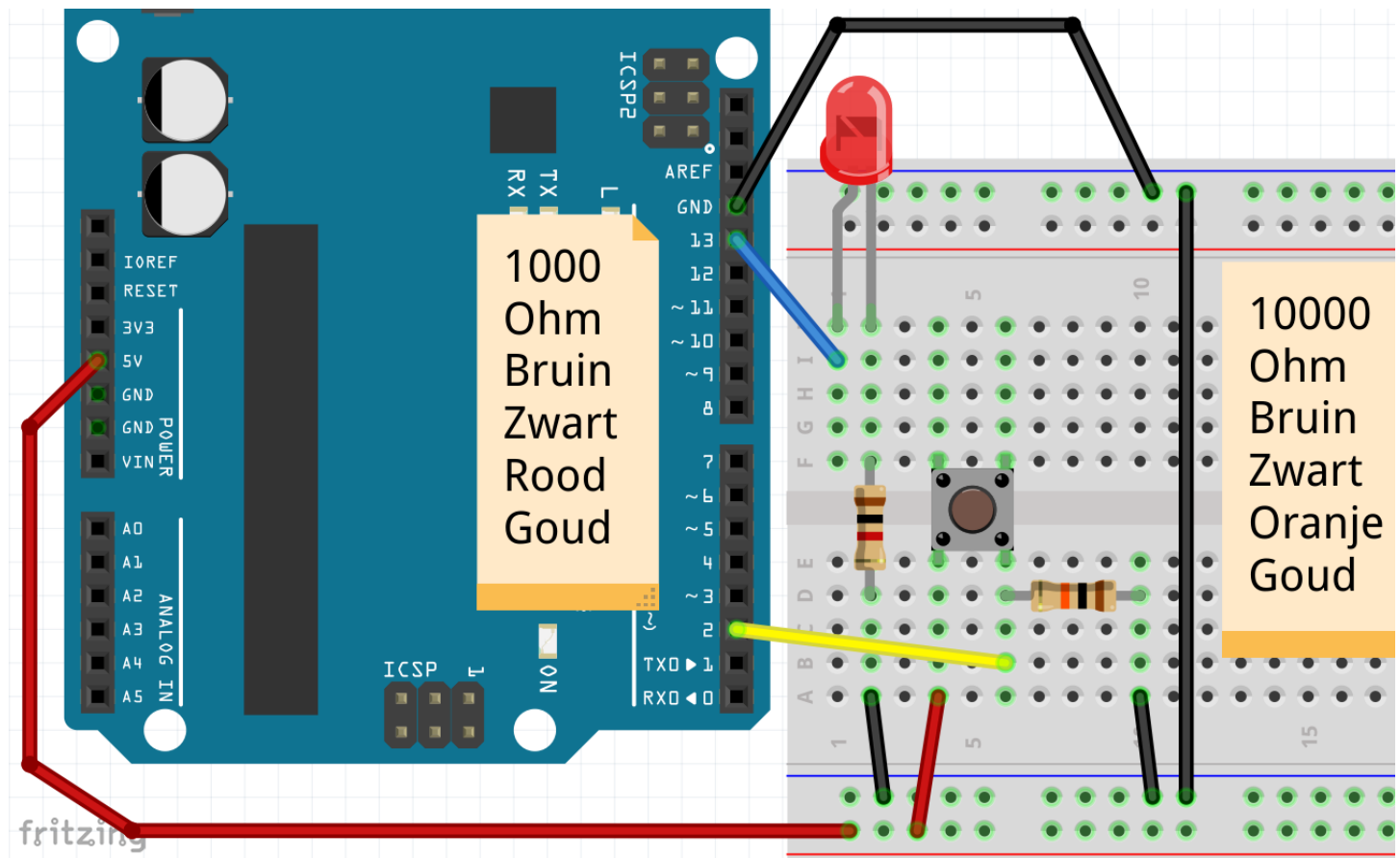


Figure 16: Stroomschema



De weerstand aan de knop wordt een 'Pull Down' weerstand genoemd



De 'Pull Down' weerstand zorgt dat pin 2 verbonden is met GND als de knop niet ingedrukt is

Code

Hier is code om het lampje aan te laten gaan, als de knop wordt ingedrukt:

```
void setup()
{
  pinMode(13, OUTPUT);
  pinMode( 2, INPUT );
}

void loop()
{
  digitalWrite(13, digitalRead(2));
}
```



`digitalRead(2)`

‘Lieve computer, geef mij HIGH als er op pin 2 spanning staat.
Geef anders LOW’

`digitalWrite(13, digitalRead(2))`

‘Zet op pin 13 spanning als er op pin 2 spanning staat’

Vragen

- 1. Als je de knop indrukt, gaat de lamp dan uit of aan?
- 2. Er is een regel `pinMode(2, INPUT)`. Waarom staat er een spatie voor de 2? Mag die spatie weg?

Antwoorden

- 1. De knop gaat dan aan
- 2. Er staat een spatie, omdat dit mooier eruit ziet met de regel erboven. De spatie mag weg.



Goede programmeurs werken netjes

if

Met if kun je de Arduino iets laten doen, als iets zo is:

```
if (digitalRead(4) == HIGH)
{
    digitalWrite(5, HIGH);
}
else
{
    digitalWrite(6, LOW);
}
```

In deze code wordt gekeken of op pin 4 spanning staat. Zo ja, dan zet de Arduino spanning op pin 5. Anders (else) haalt de Arduino de spanning van pin 6 af.



```
if (digitalRead(4) == HIGH) {}
```

‘Lieve computer, als er spanning op pin 4 staat, doe dan hetgeen tussen accolades’



De = kun je uitspreken als ‘zet op’. De == kun je uitspreken als ‘is gelijk aan’

Opdracht 1

```
void setup()
{
  pinMode(4, INPUT );
  pinMode(5, OUTPUT);
  pinMode(6, OUTPUT);
}

void loop()
{
  if (digitalRead(4) == HIGH)
  {
    digitalWrite(5, HIGH);
  }
  else
  {
    digitalWrite(6, LOW);
  }
}
```

Maak de code zo, dat:

- als je op de knop drukt, dat het lampje aan gaat
- als je op de knop niet indrukt, dat het lampje uit gaat



Na de ronde haken van `if` komt geen puntkomma

Oplossing 1

```
void setup()
{
  pinMode(13, OUTPUT);
  pinMode( 2, INPUT );
}

void loop()
{
  if (digitalRead(2) == HIGH)
  {
    digitalWrite(13, HIGH);
  }
  else
  {
    digitalWrite(13, LOW);
  }
}
```

Opdracht 2

Sluit een tweede LEDje aan. Maak de code zo, dat:

- als je op de knop drukt, dat het eerste lampje aan gaat en het tweede LEDje uit
- als je op de knop niet indrukt, dat het eerste lampje uit gaat en het tweede LEDje uit



Binnen de accolades van een **if** kun je meerdere regels typen. Net als tussen de accolades van **setup** en **loop**!



Vergeet de **else** ('doe anders') niet!

Oplossing 2

Figuur Oplossing van 'Knop met twee LEDjes' laat zien hoe je dit aan moet sluiten.

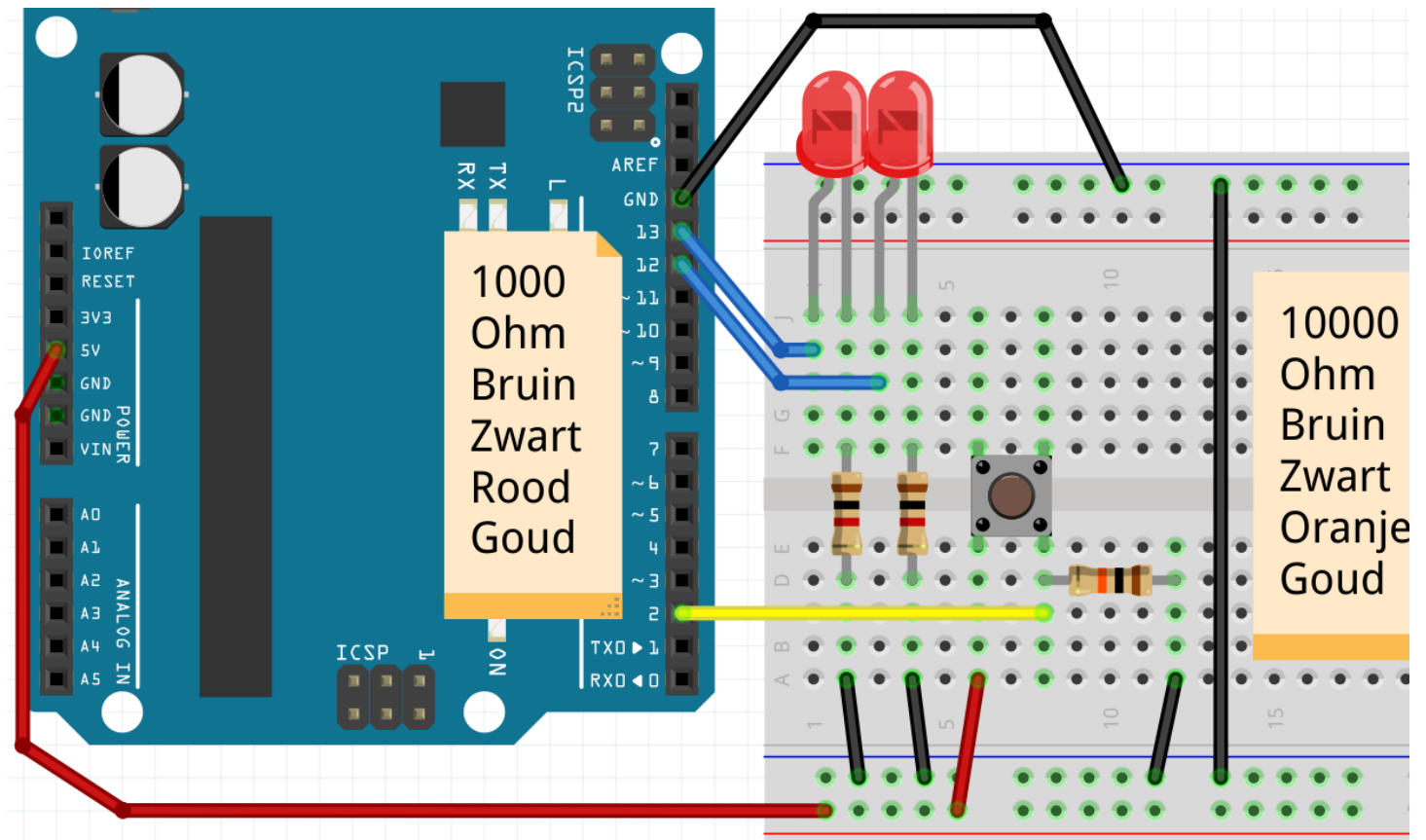


Figure 17: Oplossing van 'Knop met twee LEDjes'

Dit is de code:

```
void setup()
{
  pinMode(13, OUTPUT);
  pinMode(12, OUTPUT);
  pinMode( 2, INPUT );
}

void loop()
{
  if (digitalRead(2) == HIGH)
  {
    digitalWrite(12, LOW );
    digitalWrite(13, HIGH);
  }
  else
  {
    digitalWrite(12, HIGH);
    digitalWrite(13, LOW );
  }
  delay(10);
}
```

Eindopdracht

Sluit een tweede knop aan. Maak de code zo, dat

- als je op de eerste knop drukt, dat het eerste lampje aan gaat
- als je de eerste knop niet indrukt, dat het eerste lampje uit gaat
- als je op de tweede knop drukt, dat het tweede lampje uit gaat
- als je de tweede knop niet indrukt, dat het tweede lampje aan gaat



Je kunt vaker `if` na elkaar zetten



Voor een tweede knop heb je een tweede weerstand van tienduizend Ohm nodig

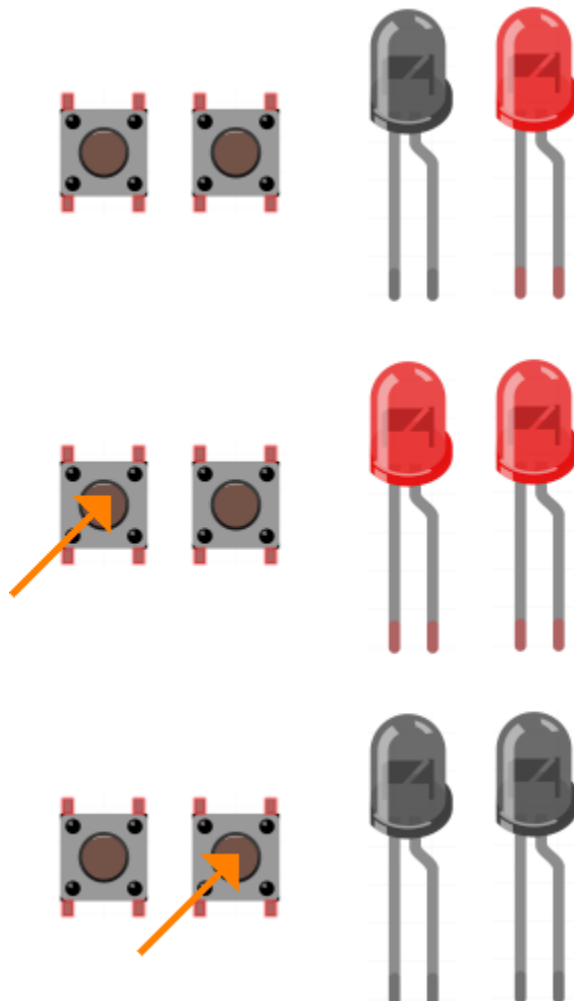


Figure 18: Eindopdracht