

## Dag 1

(C) Richèl Bilderbeek 

August 25, 2014

## 0.2 Doel

1. Weten hoe de cursus werkt
2. Software installeren
3. Theorie: breadboard, kortsluiting en blink

## 0.3 Wie zijn wij?

- Peter Struwe
  - Weet veel van elektronica
  - Arduino winkel
- Richèl Bilderbeek
  - Weet veel van programmeren
  - Onderwijs

## 0.1 Vandaag

1. Introduction: introductie
2. WhatIsArduino: wat is een Arduino?
3. ConnectLed: een LEDje aansluiten
4. SwitchLed: een LEDje laten knipperen

## Arduino cursus introductie

(C) Richèl Bilderbeek 

August 25, 2014

## 0.5 Dagplanning

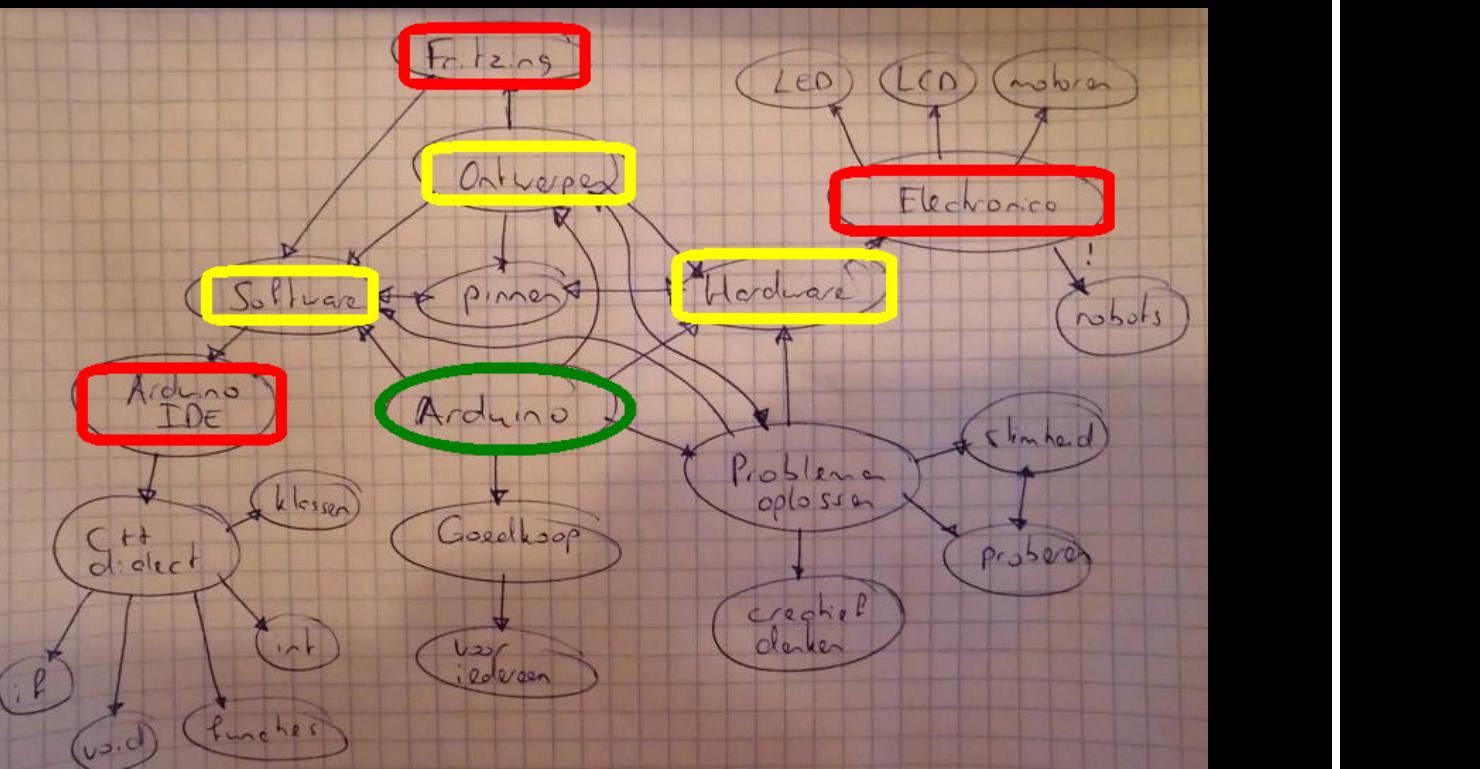
1. Soms begint Arduino met theorie/workshop, soms andersom

Tijd	A	M
19:00-19:15		Inloop
19:15-19:20	Introductie: wat gaan beide groepen doen?	
19:30-20:30	Theorie/Workshop	Projecten
20:30-21:00		Pauze
21:00-22:00	Projecten	Theorie/Workshop

## 0.1 Overzicht

1. Doel
2. Wie zijn wij?
3. Wie zijn jullie? Wat willen julie?
4. Wat willen we?
5. Hoe doen we dat?

## 0.6 Wat is Arduino?



6

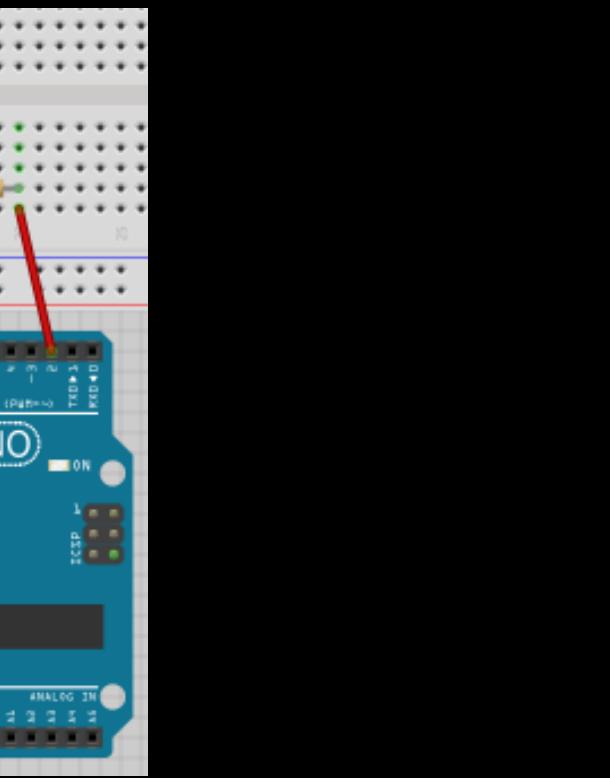
## 0.10 Software

Installeer de volgende programma's:

- Arduino IDE
- Fritzing

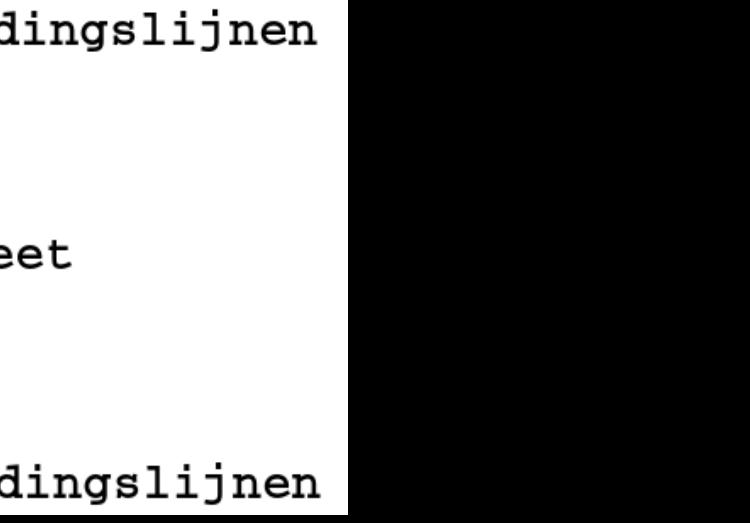
10

## 0.7 Elektronica



7

## 0.11 Breadboard



11

## 0.8 Software

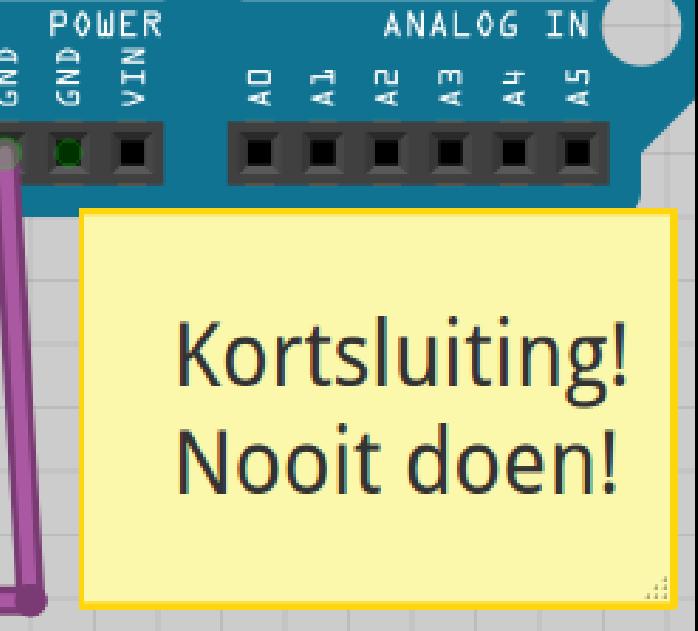
Screenshot of the Arduino IDE showing a sketch named 'Blink'. The code is as follows:

```
const int led = 13;  
  
void setup()  
{  
  pinMode(led, OUTPUT);  
}  
  
void loop()  
{  
  digitalWrite(led, HIGH);  
  delay(1000);  
  digitalWrite(led, LOW);  
  delay(1000);  
}
```

8

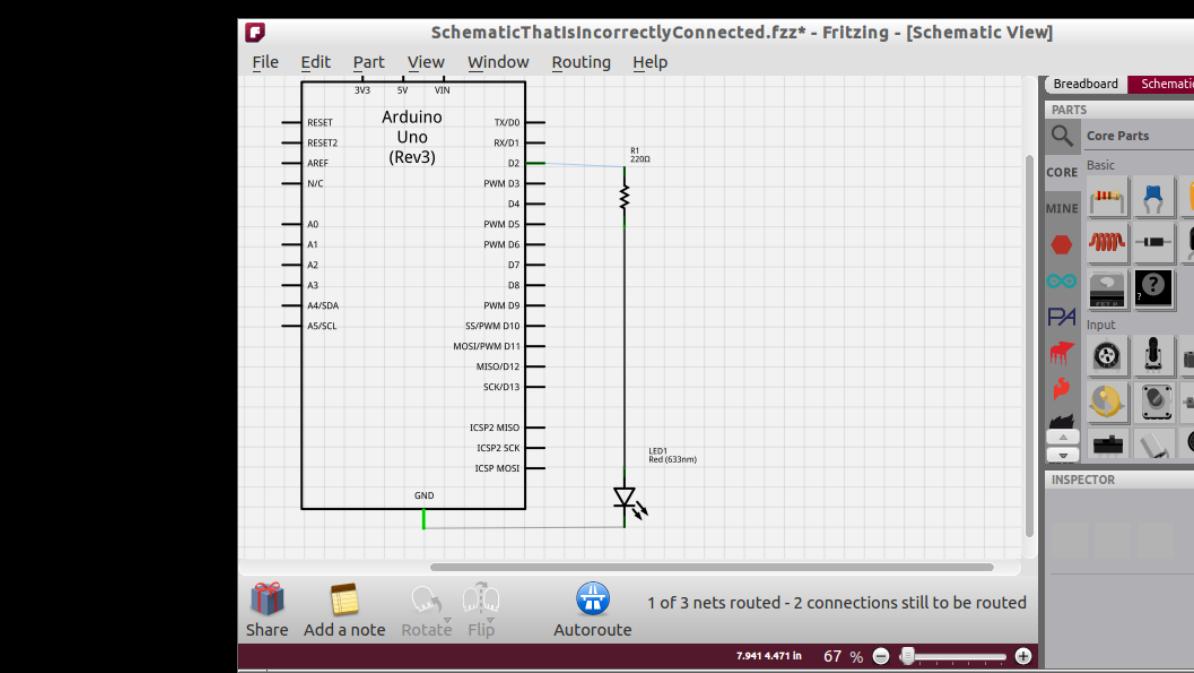
## 0.12 Wat is kortsluiting?

- Als de stroom zonder weerstand van 5V naar GND kan komen



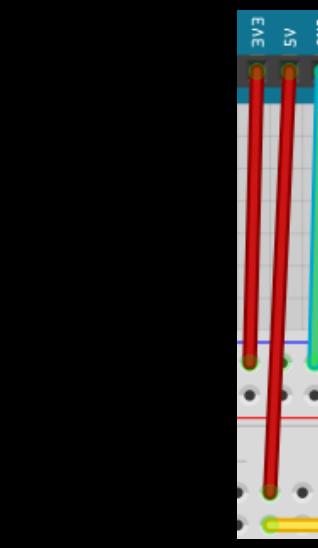
12

## 0.9 Ontwerpen



9

## 0.13 Kortsluiting



13

## 0.14 Hoe dit te voorkomen?

- Weerstand van minimaal 270 ohm
- Ander nuttig gebruik van de spanning

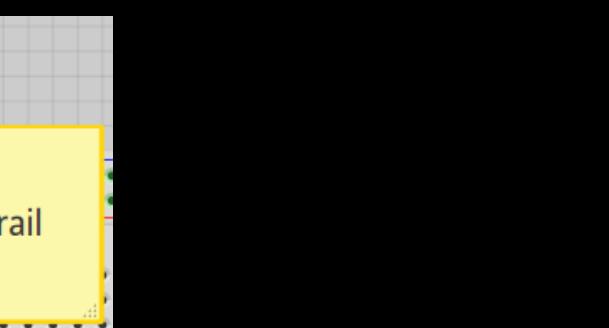
14

Wat is Arduino?

(C) Richèl Bilderbeek 

August 25, 2014

## 0.15 Geen kortsluiting

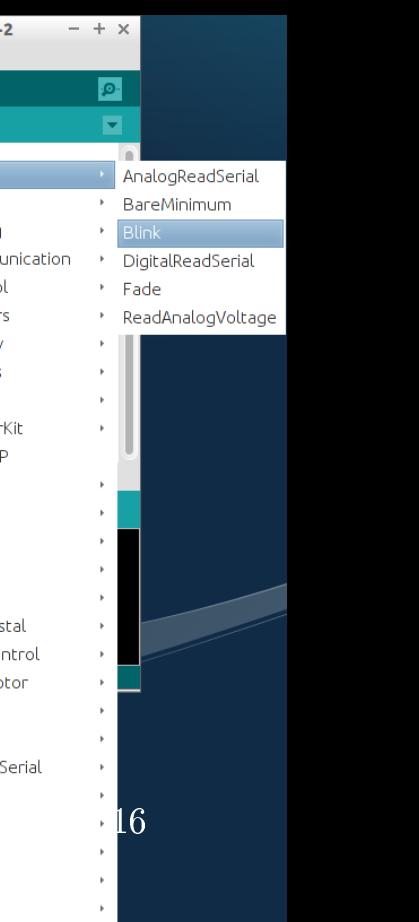


15

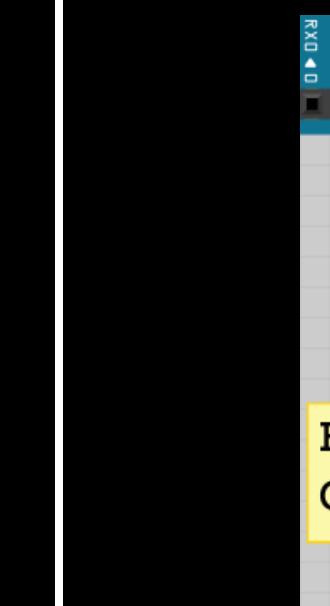
## 0.1 Overzicht

1. Wat weten jullie al?
2. Wat is een mindmap?
3. Maken mindmap
4. Wat is Arduino volgens Richèl
5. Conclusie

## 0.16 Blink code



## 0.17 Blink hardware

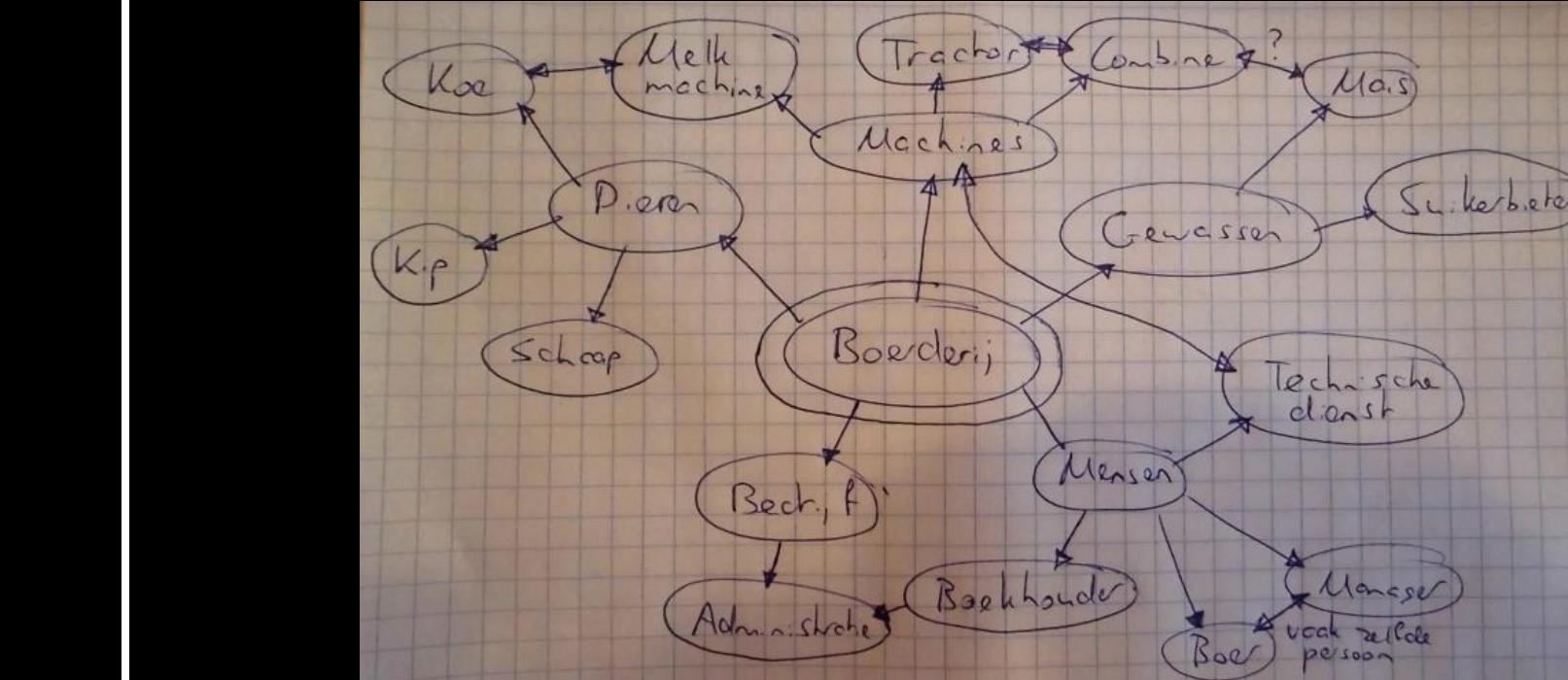


17

## 0.2 Wat weten jullie?

- Wij willen weten wat jullie al weten!
- Hiervoor hebben we een mindmap van jullie nodig

## 0.3 Wat is een mindmap?



## 0.4 Maken mindmap

- Opdracht: maak een mindmap met de tekst 'Arduino' in het midden
- Mijn voorbeeld is maar een voorbeeld: alles mag!

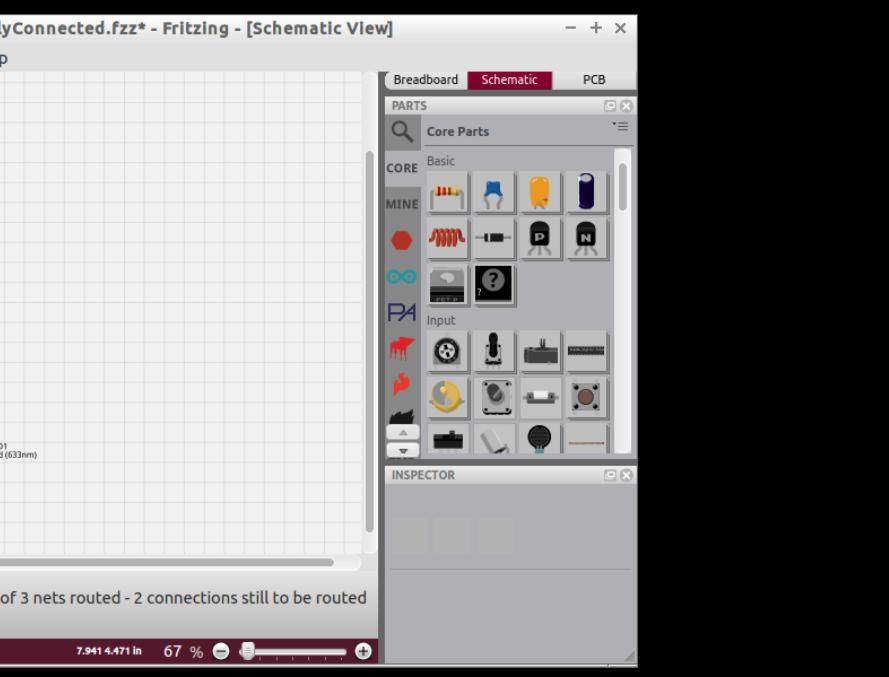
4

## 0.8 Software

```
Blink | Arduino 1:1.0.5+dfsg2-1
File Edit Sketch Tools Help
Blink $ 
const int led = 13;
void setup()
{
pinMode(led, OUTPUT);
}
void loop()
{
digitalWrite(led, HIGH);
delay(1000);
digitalWrite(led, LOW);
delay(1000);
}
```

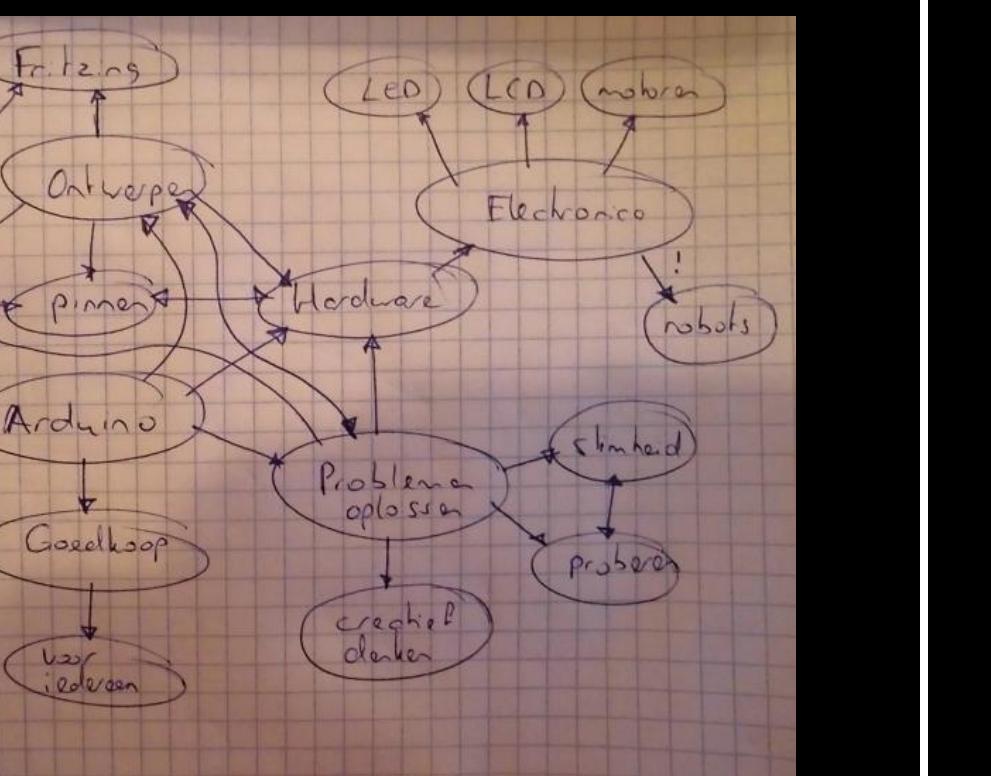
8

## 0.9 Ontwerpen



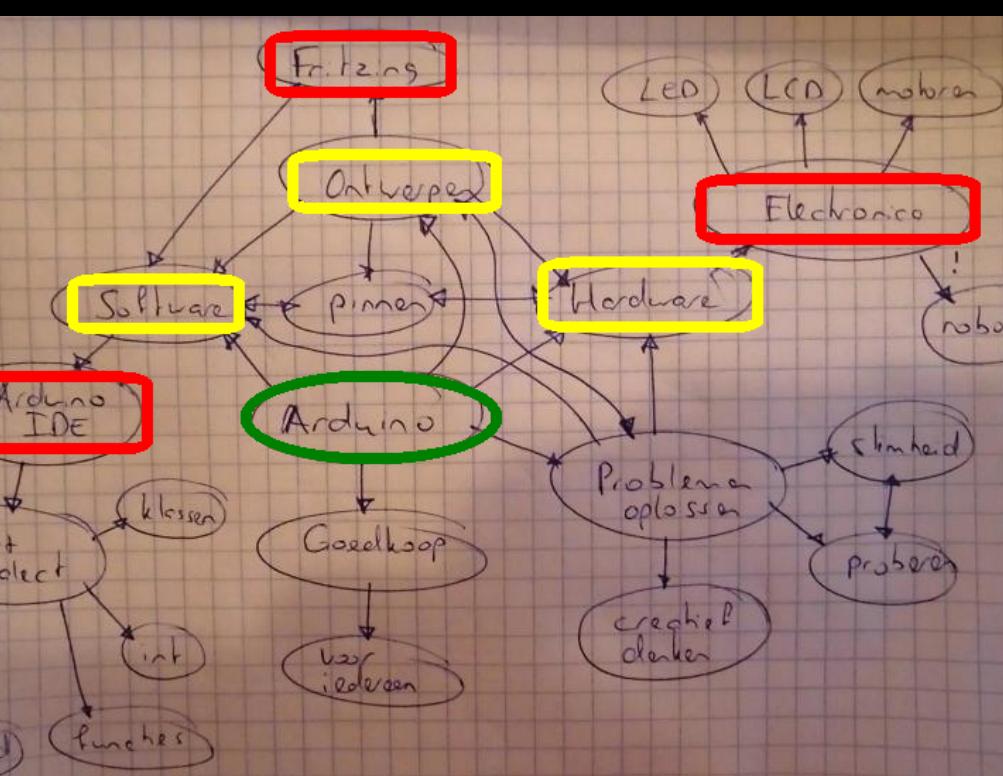
9

## 0.5 Wat is Arduino volgens Richèl?



5

## 0.6 Wat is Arduino volgens Richèl?



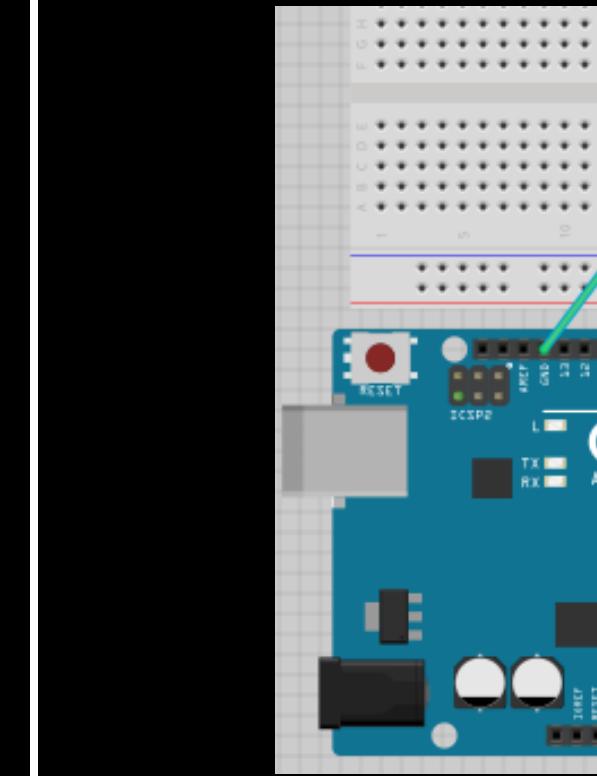
6

## 0.10 Conclusie

Wat we doen is divers:

- Ontwerpen: Fritzing
- Bouwen: Elektronica
- Programmeren: Arduino IDE

## 0.7 Elektronica



7

Arduino & werkwijze

(C) Richèl Bilderbeek

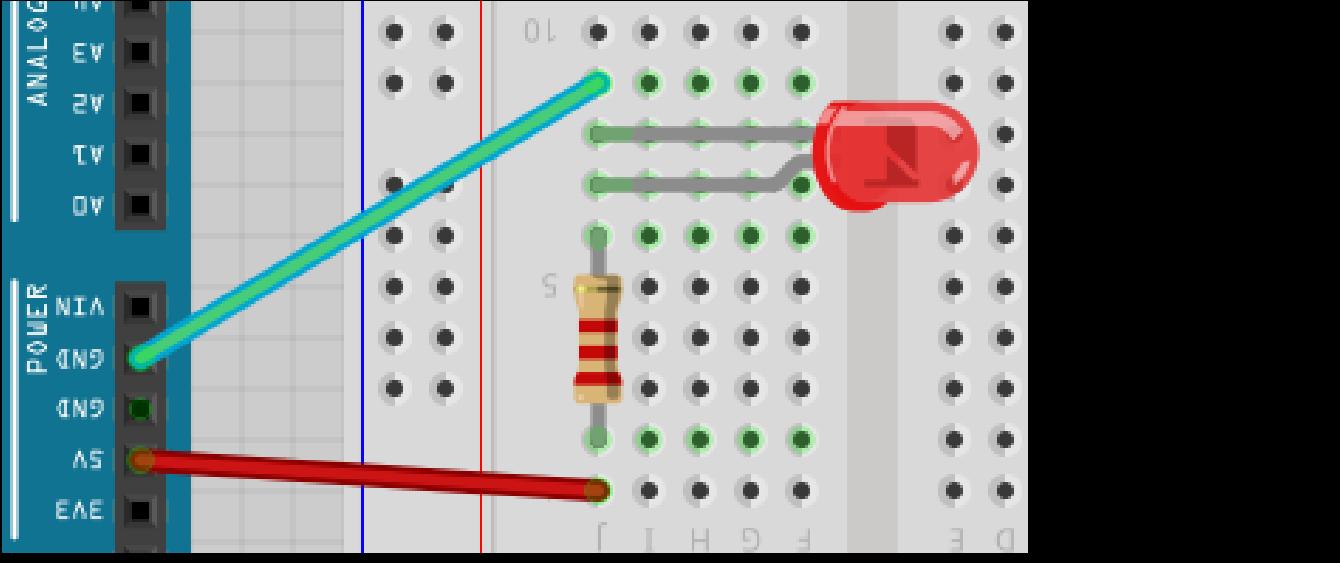
August 25, 2014

## 0.1 Overzicht

1. Waarom een werkwijze?
2. Welke werkwijze?

1

## 0.5 Stroomschema



5

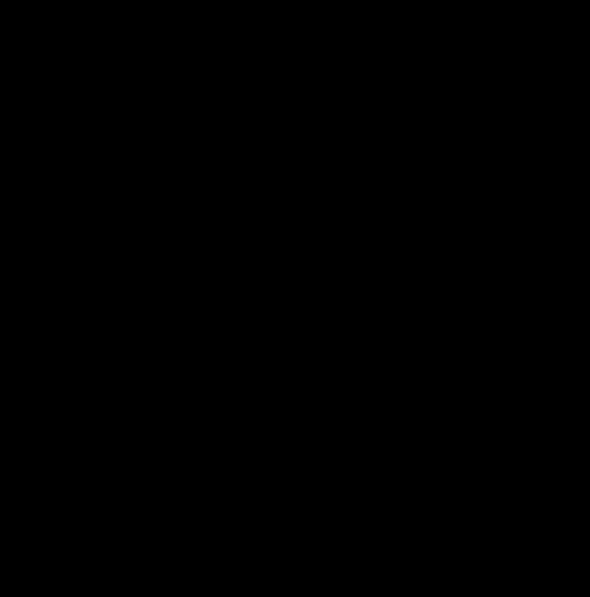
## 0.6 Wat is je programma?

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

void loop()
{
    int x = 0;
    if (x == 10)
    {
        digitalWrite(2,HIGH);
    }
    ++x;
}
```

6

## 0.2 Hoezo?



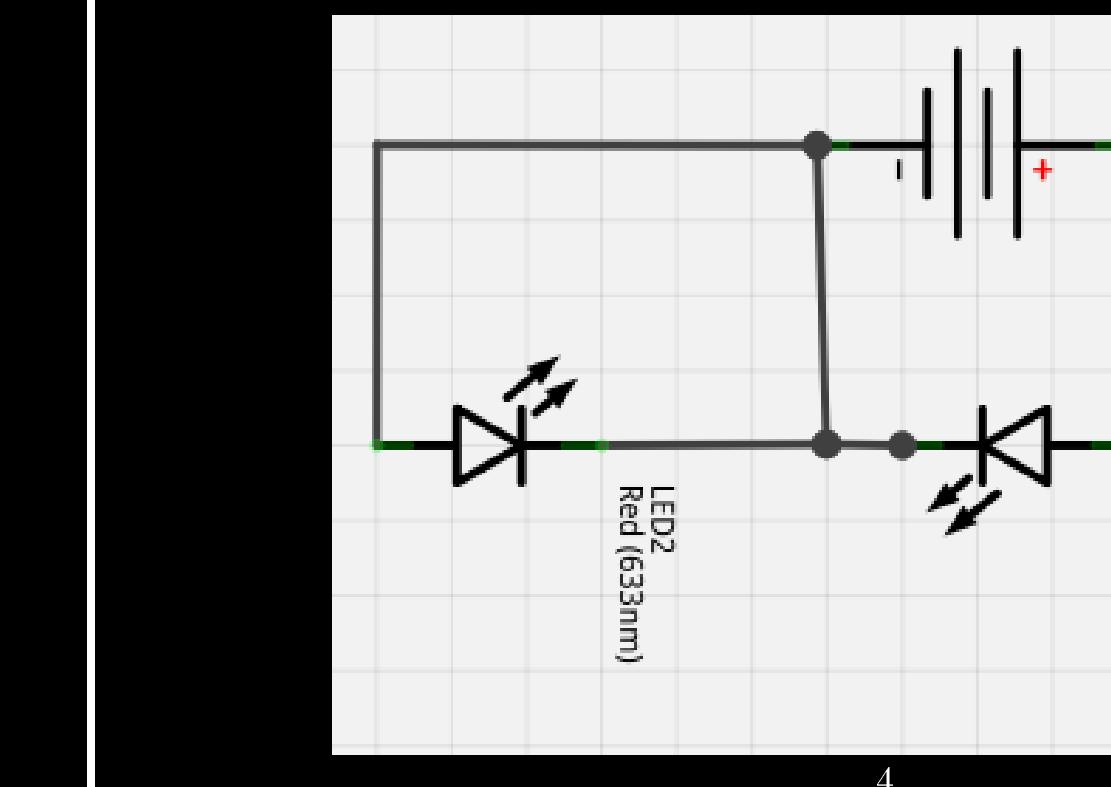
2

## 0.3 'Hij doet het niet'

- Wat is je stroomschema?
- Wat is je programma?
- Wat had je verwacht? Wat zie je gebeuren?

3

## 0.4 Stroomschema

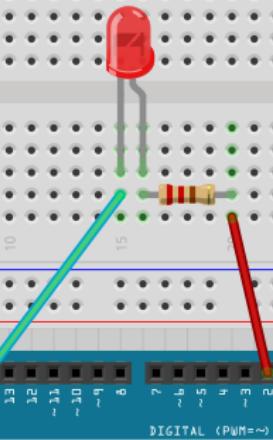


4

## 0.5 Stroomschema

## 0.7 Wat is je programma?

```
int led = 13;
void setup()
{
    pinMode(led, OUTPUT);
}
void loop()
{
    digitalWrite(led, HIGH);
    delay(1000);
    digitalWrite(led, LOW);
    delay(1000);
}
```



7

## 0.8 Wat had je verwacht? Wat zie je gebeuren?

- 'Dat er dingen gebeurden'
- 'Dat als ik op de knop druk, er een lampje elke seconde aan en uit gaat, de pieper gaat piepen met afwisselend elke twee seconde een hoge en een lage toon, het LCD scherm met een snelheid van een letter per seconde als een lichtkrantje het Wilhelmus toont ... camera ... robotarmen ... koude kernfusie'
- 'dat het LEDje om de seconde aan en uit gaat'
- [andere citaten van vage verwachtingen]

8

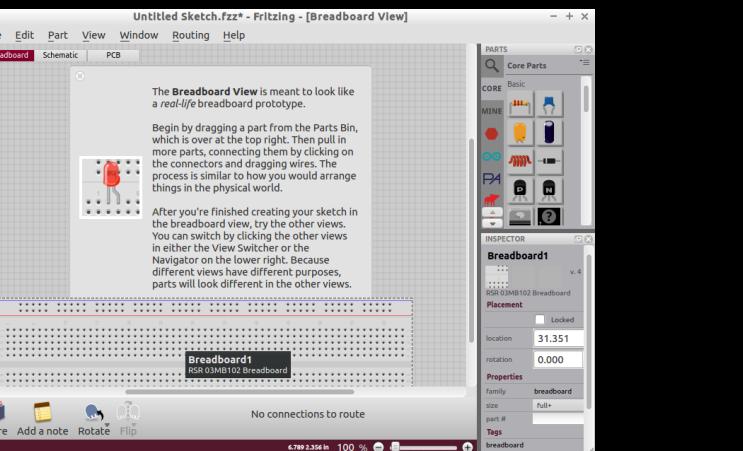
## 0.9 Wat had je verwacht? Wat zie je gebeuren?

- 'als ik de drukknop indruk, dat dit bij de Arduino binnenkomt'
- 'dat er elke seconde afwisselend wel en geen spanning op het LEDje staat'
- Dit zijn verwachtingen die wijzen richting de oplossing
- Elke verwachting omvat een aanname, die blijkbaar onjuist is gebleken

9

## 0.2 Fritzing

- Vrije software
- Linux, Mac OS X, Windows



2

## 0.3 Fritzing installatie

- sudo apt-get install fritzing
- yum install fritzing
- Download van [fritzing.org/download](http://fritzing.org/download)

3

## 0.10 Werkwijze

- Ontwerp precies
  - maak een stroomschema
- Werk precies:
  - sluit het stroomschema juist aan
  - laat het stroomschema met de software overeenkomen
- Denk precies
  - bedenk wat je verwacht
  - bedenk wat je aanneemt
- Als je dit doet, kun je alles

10

## Hoe sluit ik een LED aan?

(C) Richèl Bilderbeek CC BY NC SA

August 25, 2014

4

## 0.1 Overzicht

1. Fritzing
2. Er is een probleem?
3. Wat weten we?
4. Er is een probleem?
5. Hoe gaan we daar mee om?
6. Opdracht: vind dit uit! Noteer in je logboek

1

## 0.5 Er is een probleem?

- Ja: je kunt een LED niet direct op een Arduino aansluiten
- Hoe kun je dat merken?
  - Dat gaan we doen!
  - Waarom is dat zo?

5

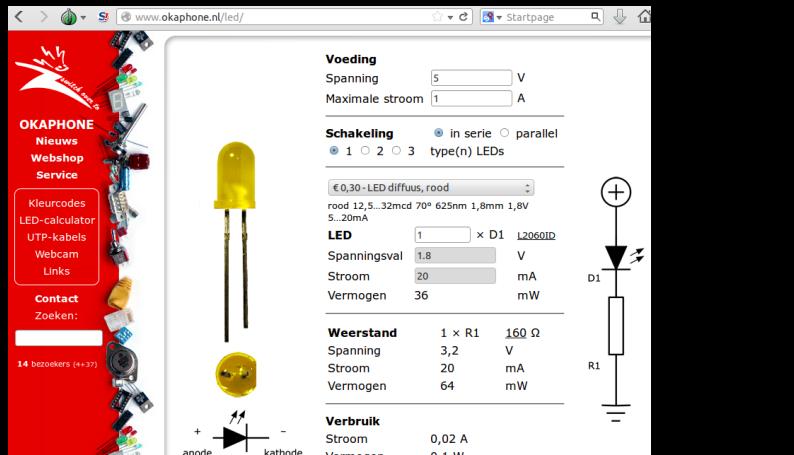
## 0.6 Wat weten we?

1. Wat is volt?
2. Wie weet hoeveel volt een LED gebruikt?
3. Wie weet hoeveel volt een Arduino levert?

6

## 0.10 Welke weerstand?

- Proberen: start met hoge weerstand en ga omlaag
- Berekenen



10

## 0.7 Antwoorden

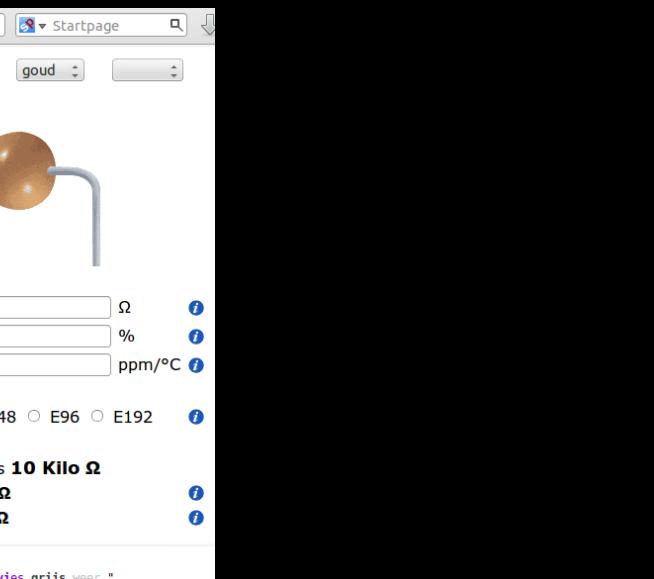
1. Volt = de hoeveelheid energie die stroom levert
2. Een LED (diffuus, zonder [...]) gebruikt 1,8-2,5 Volt (zie volgende slide)
3. Een Arduino levert 3,3 of 5,0 Volt

Wat kun hieraan doen?

7

## 0.11 Welke weerstand?

- Zij Bracht ROzen Op GErrits GRaf Bij Vies GRIJS Weer



11

## 0.8 Antwoorden



Wat kun hieraan doen?

8

## 0.9 Hoe hoge spanningen tegen te gaan?

- Met een weerstand!



9

## 0.12 Bouwen!

- Wat gebeurt er als je de LED omdraait?
- Welke problemen loop je tegen aan?

12

## 0.13 Geniet

- Wat zijn logische volgende stappen?
- Welke problemen loop je tegen aan?

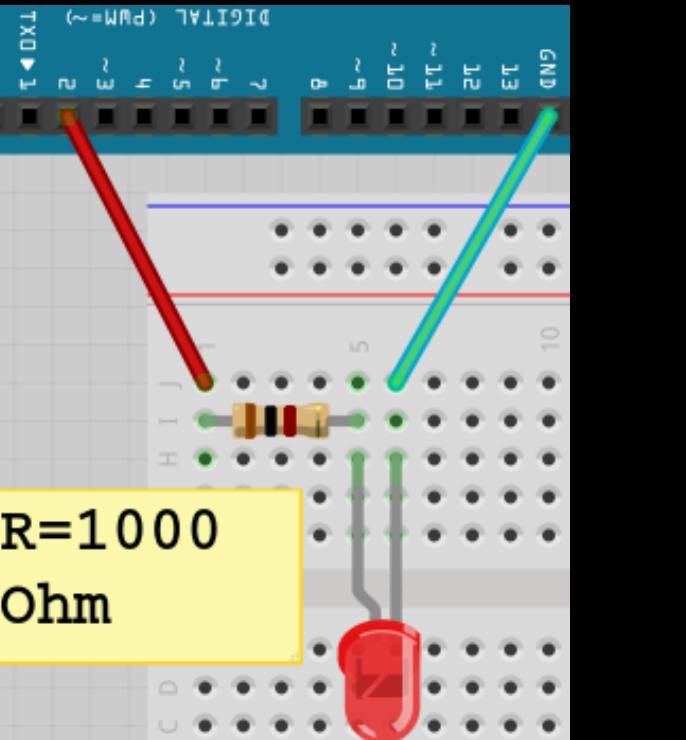
13

## Hoe schakel ik een LED?

(C) Richèl Bilderbeek 

August 25, 2014

## 0.4 Ontwerp



4

## 0.5 En nu?

- Programmeren!
- Met Arduino IDE ('Integrated Development Environment' ≈ 'Programmeeromgeving')

## 0.1 Overzicht

1. Doel
2. Hoe ontwerpen?
3. Arduino IDE
4. Bouwen!

1

## 0.6 Arduino IDE

2

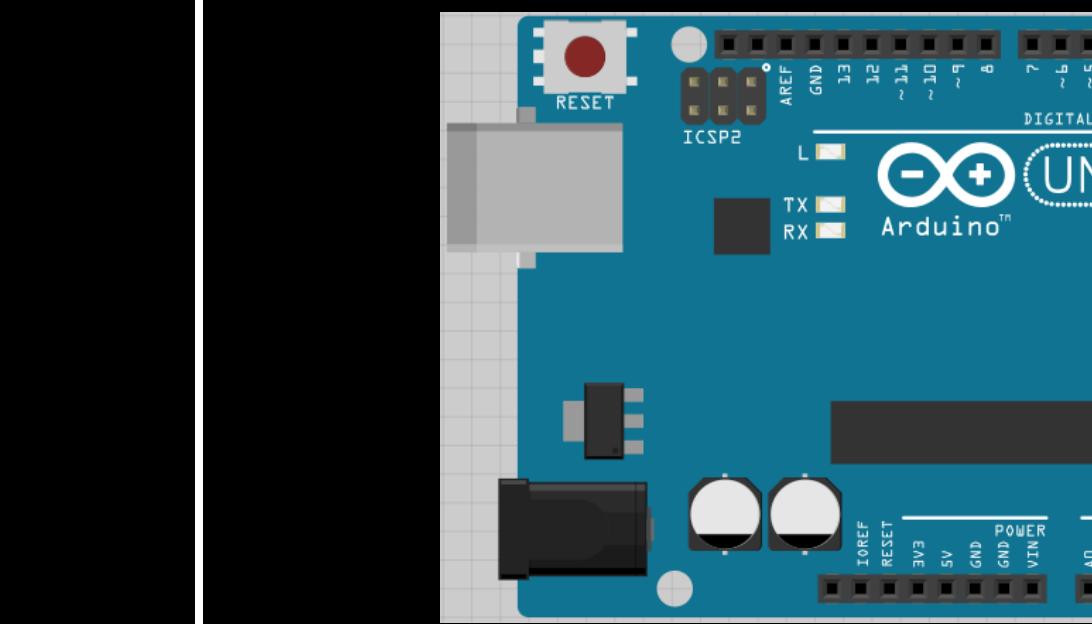
## 0.2 Doel

Een machine maken die een LED na een seconde aan doet, en deze een seconde later weer uit doet

3

## 0.3 Hoe ontwerpen?

- Waar denk je dat de Arduino dit mee kan?



7

## 0.7 Arduino IDE installatie

- sudo apt-get install arduino
- yum install arduino
- Download van <http://arduino.cc/en/main/software>

5

6

## 0.8 Opdracht

- Installeer Arduino IDE
- Bekijk wat de Arduino IDE kan: de voorbeelden, het uploaden1
- Ontwerp: stroomschema van een LED die knippert
- Programmeer: vind de code van een LED die knippert
- Bouw, test, noteer

8

## Dag 1 Oefeningen

(C) Richèl Bilderbeek 

August 25, 2014

## 0.3 Ontwerp in Fritzing

- Installeer Fritzing
- Bekijk wat Fritzing kan: 'Breadboard view', 'Schematic' en 'PCB'
- Ontwerp: stroomschema van een LED die altijd brandt
- Bouw, test, noteer

3

## 0.4 Arduino IDE installatie

- sudo apt-get install arduino
- yum install arduino
- Download van <http://arduino.cc/en/main/software>

4

## 0.1 Overzicht

1. Installeer Fritzing
2. Ontwerp in Fritzing
3. Installeer Arduino IDE
4. Programmeer een knipperende LED

1

## 0.5 Programmeer een knipperende LED

- Installeer Arduino IDE
- Bekijk wat de Arduino IDE kan: de voorbeelden, het uploaden1
- Ontwerp: stroomschema van een LED die knippert
- Programmeer: vind de code van een LED die knippert
- Bouw, test, noteer

5

## 0.2 Fritzing installatie

- sudo apt-get install fritzing
- yum install fritzing
- Download van [fritzing.org/download](http://fritzing.org/download)

2

Dag 2

(C) Richèl Bilderbeek 

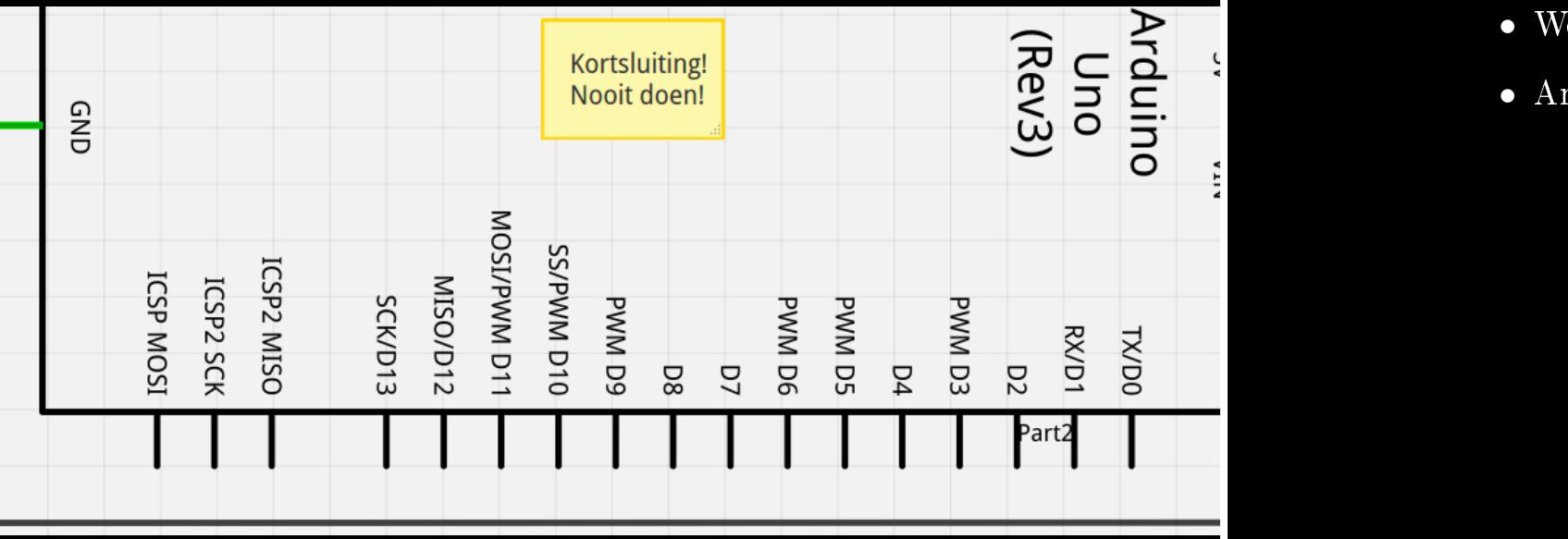
August 25, 2014

## 0.1 Vandaag

1. shortCircuit: kortsleuteling & breadboard
2. analogRead: belangrijk basisschakelingen met weerstandjes
3. readInput: nog een belangrijke basisschakeling met een drukknop
4. Per groepje: solderen bij Tonnie & bouwen met sensoren en motortje
5. Einddoel: een zo ingewikkeld mogelijke machine bouwen

1

## 0.3 Wat is kortsleuteling?



3

## 0.4 Hoe dit te voorkomen?

- Weerstand van minimaal 270 ohm
- Ander nuttig gebruik van de spanning

4

## Kortsleuteling & Breadboard

(C) Richel Bilderbeek

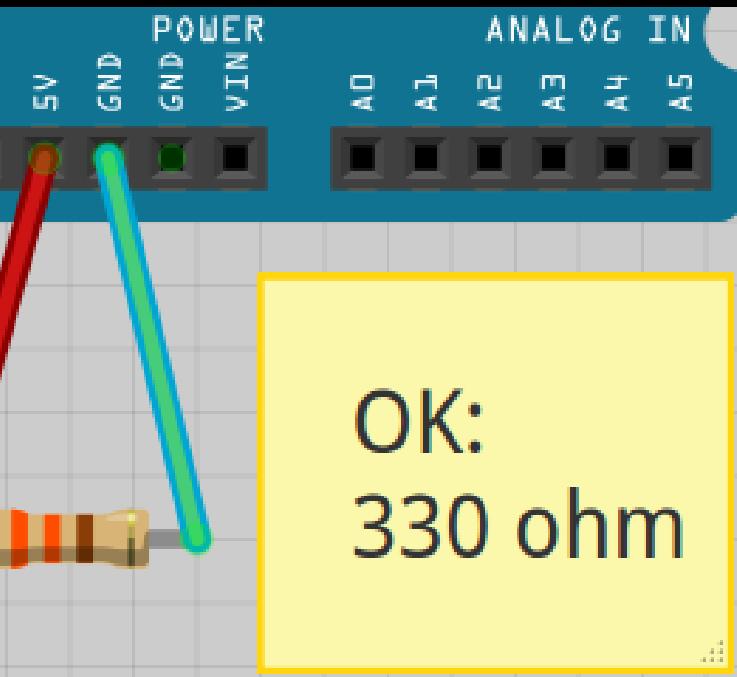
August 25, 2014

## 0.1 Overzicht

1. Wat is kortsleuteling?
2. Hoe dit te voorkomen?
3. Hoe werkt een breadboard?

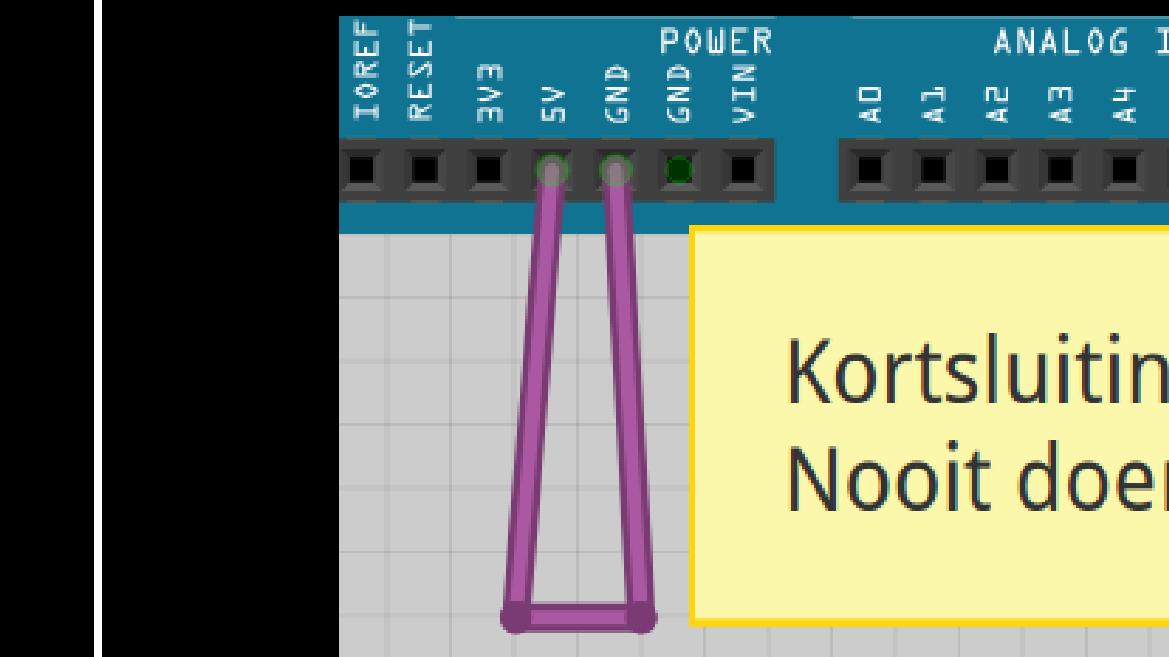
5

## 0.5 Wat is kortsleuteling?



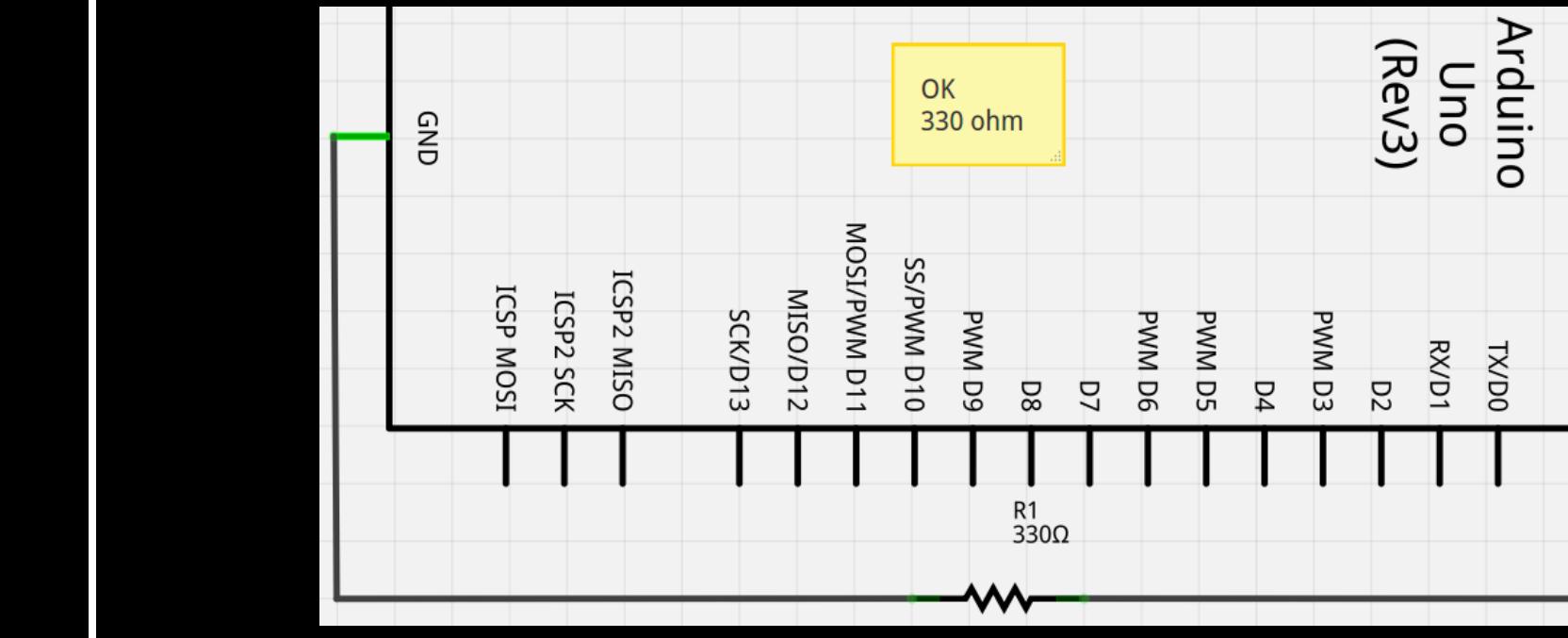
5

## 0.2 Wat is kortsleuteling?



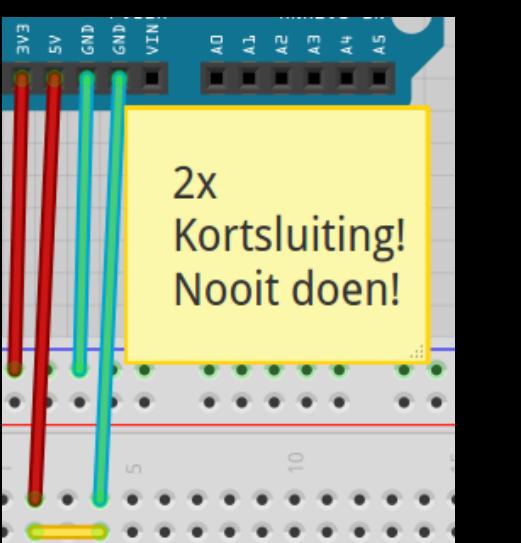
6

## 0.6 Wat is kortsleuteling?



6

## 0.7 Breadboard



7

## 0.2 Doel

- Spanning meten met Arduino: analogRead
- Basisschakelingen herkennen
- Bouwen van minstens vijf schakelingen
- Dit niet kennen = domme fouten gaan maken: drukknop, sensor, etcetera

2

3

## 0.8 Breadboard



4

5

## 0.3 Opdracht



6

7

## 0.4 Opdracht



8

9

## 0.5 Programma



10

11

## 0.1 Overzicht

1. Doel
2. Opdracht
3. Meten
4. Opschrijven

12

13

## 0.6 Meten

14

15

16

## 0.7 Opschrijven

17

18

19

## 0.8 Resultaat

20

21

22

## 0.9 Conclusie

23

24

25

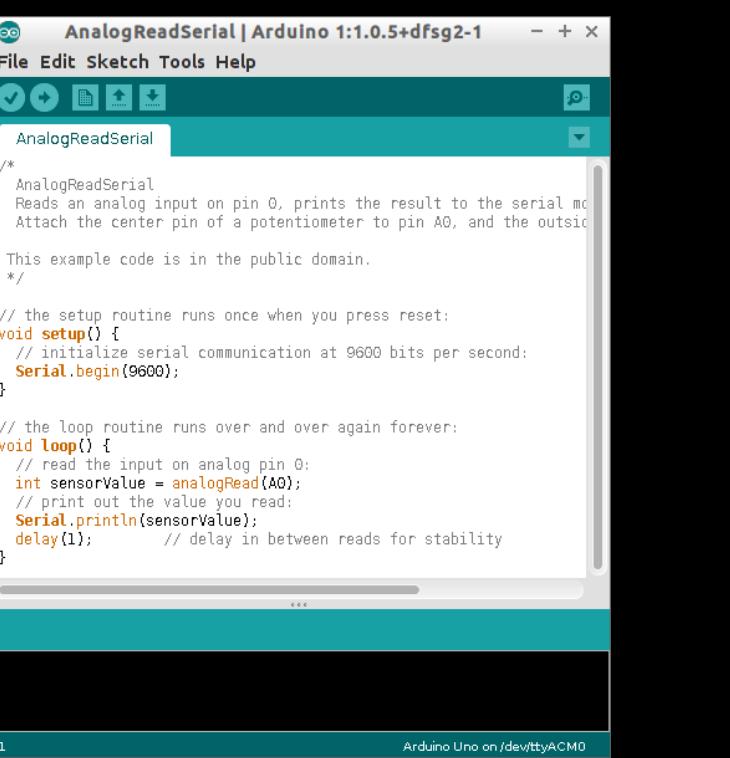
## 0.10 Toekomstige werkzaamheden

26

27

28

## 0.6 Programma



```
/* AnalogReadSerial | Arduino 1:1.0.5+dfsg2-1
File Edit Sketch Tools Help
AnalogReadSerial
AnalogReadSerial
Reads an analog input on pin 0, prints the result to the serial monitor.
Attach the center pin of a potentiometer to pin A0, and the outside pins to ground and +5V.
This example code is in the public domain.

/*
the setup routine runs once when you press reset:
void setup() {
  // initialize serial communication at 9600 bits per second:
  Serial.begin(9600);
}

// the loop routine runs over and over again forever:
void loop() {
  // read the input on analog pin 0:
  int sensorValue = analogRead(A0);
  // print out the value you read:
  Serial.println(sensorValue);
  delay(1); // delay in between reads for stability
}
```

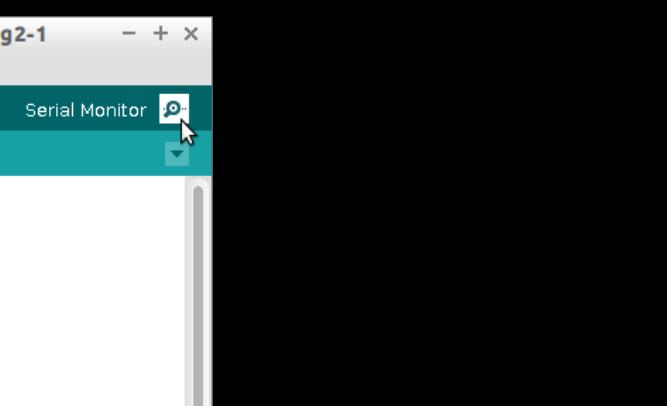
6

## 0.1 Overzicht

1. Doel
2. Vraag
3. Naief & experiment
4. Zo werkt het
5. Slim & experiment

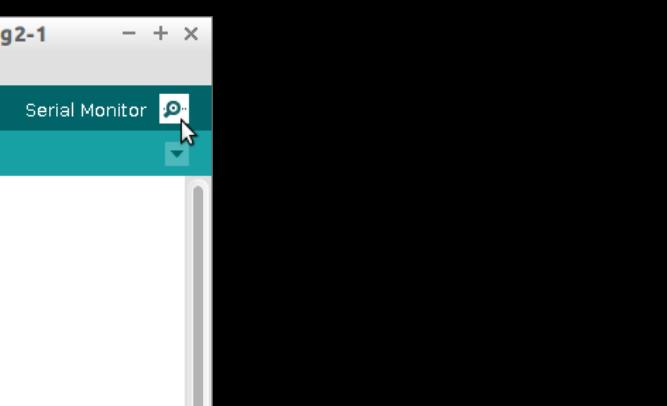
1

## 0.7 Bekijken



2

## 0.2 Doel



3

- Kunnen reageren op een drukknop
- Een nieuwe basisschakelingen leren
- Dit niet kennen = domme fouten gaan maken

4

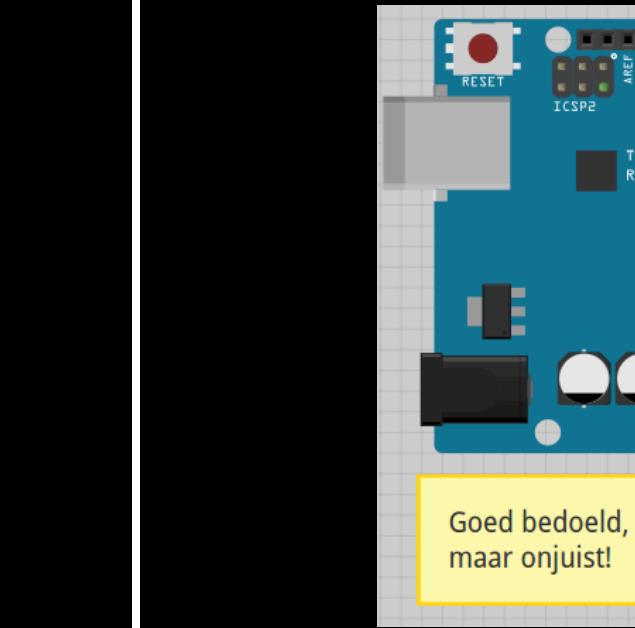
## 0.3 Vraag

### QUESTION

- Bouw een machine die kan reageren op een drukknop
- Gebruik je kennis van 'analogRead'
- ... dit gaat echter anders dan verwacht!

5

## 0.4 Ontwerp: naief



6

## 0.8 Opschrijven

### ANSWER

- Welke waarden meet je bij elk van de vijf schakelingen?
- Kun je voorspellingen maken? Maak gerust nieuwe schakelingen!
- Noteer! Deze schakelingen zul je vaak gaan bekijken!

Hoe lees ik input?

(C) Richel Bilderbeek 

August 25, 2014

7

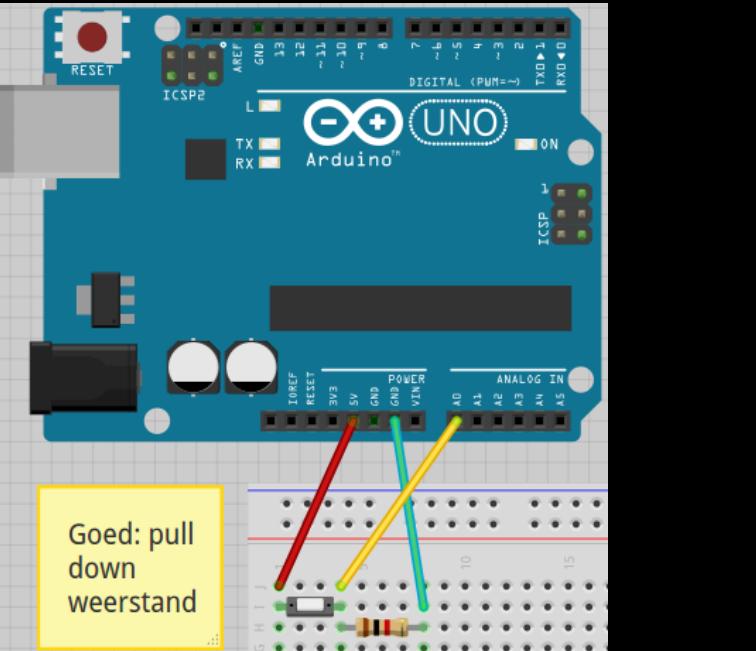
8

## 0.5 Opdracht

- Bouw dit!
- Wat voorspel je?
- Gebruik weer programma 'Examples | Basics | AnalogReadSerial'
- Wat meet je als de schakelaar wel/niet ingedrukt is?
- Noteer! Dit onverwachte gedrag zal je vaker tegenkomen!
- (tijd over: probeer de schakeling te laten werken)

5

## 0.9 Ontwerp: oplossing



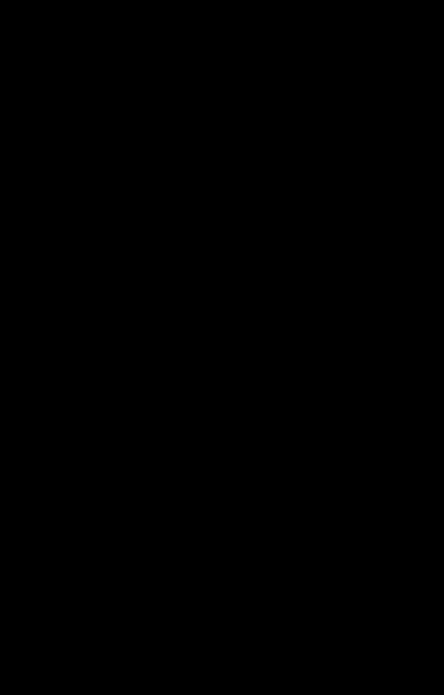
9

## 0.6 Doen...

- Volgende slide de oplossing...

6

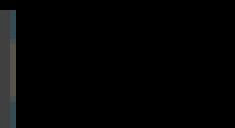
## 0.10 Ontwerp: oplossing



10

## 0.7 Waarom onjuist?

- Als de drukknop open is, is de spanning op de input onbepaald: dit kan elke waarde tussen nul en vijf Volt zijn!
- Dit wordt een zwevende input genoemd
- Hoe dit op te lossen?



7

## 0.11 Opdracht

- Bouw dit!
- Wat voorspel je?
- Gebruik weer programma 'Examples | Basics | AnalogReadSerial'
- Wat meet je als de schakelaar wel/niet ingedrukt is?
- (tijd over: gebruik analogWrite naar een LEDje)

## 0.8 Ontwerp: oplossing

- Verbind de input via een weerstand met de Aarde
- Hierdoor kan restspanning wegvalen tot nul Volt
- Dit wordt een pull-down weerstand genoemd

8

Dag 3

11

## 0.1 Vandaag

1. RgbLed: aansluiten RGB led
2. Cpp01: C++ basis
3. ReadSensor: lezen van een sensor

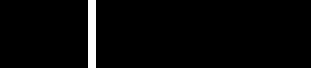
1

## 0.3 Arduino IDE

- Doel: gemakkelijk programmeren voor beginners (NB: voor gevorderden minder geschikt)
- taal: C++, volgens de C++03 standaard (niet C++0x, niet C++11)
- bibliotheken: C, Arduino
- voegt zelf nog dingen toe (zoals de main functie)

3

## Arduino C++ #1

(C) Richèl Bilderbeek 

August 25, 2014

4

## 0.1 Overzicht

1. Doel
2. Arduino IDE
3. 'setup' en 'loop'
4. Fouten
5. data typen
6. if
7. for

1

## 0.2 Doel

- Fouten leren lezen
- Internet zoektermen
- Basis van programmeren

2

## 0.4 Van code naar machinetaal

- Code is bedoeld voor mensen om te lezen en te schrijven
- Een chip begrijpt alleen maar machinetaal
- Een compiler doet de vertaling van code naar machinetaal (NB: er zijn meer programma's die helpen, die noem ik vandaag allemaal 'compiler')
- De compiler is erg precies, en dat is goed
- De compiler heeft altijd gelijk!

5

## 0.5 Minimum programma

```
void setup() {}  
void loop() {}
```

6

## 0.6 Fouten

De code:

```
//( alle code gewist )
```

Foutmelding:

```
core.a(main.cpp.o):  
In function ‘main’: / [...] / main.cpp:11:  
undefined reference to ‘setup’  
collect2: error: ld returned 1 exit status
```

Kern:

```
undefined reference to ‘setup’
```

Wat bedoelt de compiler:

De compiler heeft gehoord dat 'setup' moet bestaan, maar kan deze niet vinden

## 0.7 Opdracht

- Start 'Examples | Basics | Bare minimum'
- Probeer zoveel mogelijk verschillende fouten te maken, alleen in 'setup' ('loop' is immers precies hetzelfde)
- Noteer (1) de code (2) de kern van de foutmelding (3) wat je denkt dat de compiler bedoelt (4) laat ruimte over voor later; als je snapte wat de
- Ik vind er met gemak achtien!
- Deze fouten zul je nog vaak tegen komen!

7

## 0.11 if statement

- Als je iets soms wel en soms niet wil

```
if (analogRead(A0) < 512) { digitalWrite(2,HIGH); }
else { digitalWrite(2,LOW); }
```

Teken	Spreek uit
==	is gelijk aan
!=	is ongelijk aan
<	is kleiner dan
<=	is kleiner of gelijk dan
>	is groter dan
>=	is groter of gelijk dan

11

## 0.12 for loop

- Als je een teller wilt laten lopen

```
for (int teller=0; teller!=255; ++teller)
{
    analogWrite(2, teller);
```

- Lees dit als:

1. Zet een int genaamd 'teller' op de beginwaarde nul
2. Doe de code tussen accolades zo lang 'teller' niet gelijk is aan 255
3. Na de code tussen accolades: tel één op bij teller ('++teller') en ga naar stap 2

12

## 0.8 Fouten

//Een letter weg oid setup() {} vid setup() {} vod setup() {} voi setup() {} voidsetup() {} void etup() {} void stup() {} void seup() {} void setp() {} void setu() {}	void setup() {} void setup() {} void setup() {} void setup() {}  //Een woord of tekencombinatie weg setup() {} void () {} void setup {} void setup()
--	---

13

## 0.9 Variabelen en data typen

- Variabele: iets wat een computer moet onthouden, iets dat een mens kan lezen
- int: een geheel getal, bijvoorbeeld: een pin nummer, een aantal seconden
- double: een gebroken getal, bijvoorbeeld: een voltage
- bool: een ja/nee, bijvoorbeeld: is de drukknop ingedrukt?
- nog veel meer

9

## 0.10 Ombouwen van data type

- Soms gebeurt dit stiekum of per ongeluk
- Maar dit kan ook open en expres, met static\_cast

```
const int waarde
    = analogRead(A0);
const double voltage
    = static_cast<double>(waarde)
    / 1024.0;
```

10

## 0.13 Testen

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

void loop()
{
    const double x = 2.0 / 7.0; Serial.println(x);
    const int i = 123; Serial.println(i);
    const bool b = true; Serial.println(b);
    for (int i=0; i!=255; ++i) Serial.println(i);
}
```

## 0.14 Opdracht

- Zoek een toepassing voor een if statement
- Zoek een toepassing voor een for loop

14

## Hoe lees ik een sensor?

(C) Richèl Bilderbeek 

August 25, 2014

## 0.4 Opdracht

- Weet: een lichtsensor krijgt een andere weerstand bij meer/minder licht
- Weerstanden kennen we!
- Ontwerp een (of meer) schakeling(en) in Fritzing die bij meer/minder licht een andere waarde meet
- Bouw de schakeling, programmeer, meet de waarden, noteer!
- (tijd over: meet alleen de waarde als er op een drukknop wordt gedrukt, gebruik 'if' in je code)

## 0.5 Doen!

5

## 0.1 Overzicht

1. Doel
2. Vraag
3. Naief & experiment
4. Zo werkt het
5. Slim & experiment

6

## 0.2 Doel

- Zelf uitvinden hoe een lichtsensor werkt
- Experiment: ontwerpen, bouwen, meten, noteren

1

## 0.6 Conclusie

- Waar liep je tegen aan?
- Hoe reageert een lichtsensor?

2

Hoe stuur ik een zonnemotor?

(C) Richèl Bilderbeek 

August 25, 2014

## 0.3 Waar in Fritzing?



3

4

## 0.1 Overzicht

1. Doeel

1

## 0.5 Doen!

5

## 0.2 Doeel

- Zelf uitvinden hoe een zonnemotor werkt

2

6

## 0.6 Conclusie

- Waar liep je tegen aan?
- Hoe reageert een lichtsensor?

## 0.3 Waar in Fritzing?

- Niet

3

August 25, 2014

(C) Richèl Bilderbeek



## 0.4 Opdracht

- Weet: een zonnemotor werkt op maximaal 5 Volt
- Gebruik de code van 'Examples | Analog | Fading' met analogWrite erin
- Ontwerp een (of meer) schakeling(en) in Fritzing om de zonnemotor aan te sturen
- Bouw de schakeling, programmeer, bekijk, noteer

4

Potmeters

August 25, 2014

(C) Richèl Bilderbeek



## 0.1 Overzicht

1. Doel
2. Hoe ziet een potmeter eruit?
3. Wat en waarvoor is een potmeter?
4. Hoe meet je weerstanden?
5. Meten weerstanden potmeter
6. Aansluiten potmeter
7. Meten met Arduino

1

## 0.5 Hoe meet je weerstand?

- Met een multimeter
- Als er geen spanning staat op wat je wilt meten
- Zet de grote draaiknop op 'R' of 'Ohm'
- (demo met twee weerstandjes)

5

## 0.6 Meten weerstanden potmeter 1/2

- Noem de drie pinnen A,B en C



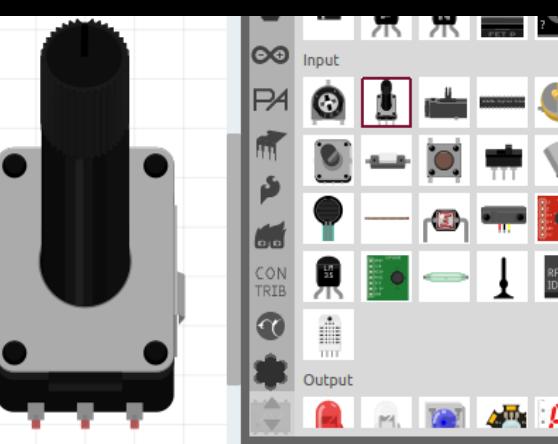
6

## 0.2 Doel

- Ontdekken hoe een potmeter werkt
- Weerstand meten met een multimeter
- Een potmeter juist aan kunnen sluiten

2

## 0.3 Hoe ziet een potmeter eruit?



3

## 0.4 Wat en waarvoor is een potmeter?

- Potmeter = potentiometer
- Potmeter = een stelsel van weerstanden die je kunt veranderen (dit gaan we zo meten!)
- Kan gebruikt worden als knop

4

## 0.7 Meten weerstanden potmeter 2/2

- Maak een tabel en meet de weerstanden van de pinnen
- Wat valt je op? Hebben we dit eerder gedaan? Hoe sluit je een potmeter aan? Waar moet je op letten bij het aansluiten: is kortsluiting mogelijk? Laat een Arduino reageren

Plek knop	Weerstand tussen A en B (Ohm)	A-C	B-C
Links	0	1000	1000
Tussen links en midden in	250	1000	750
Midden	500	1000	500
Tussen midden en rechts in	750	1000	250
Rechts	1000	1000	0

7

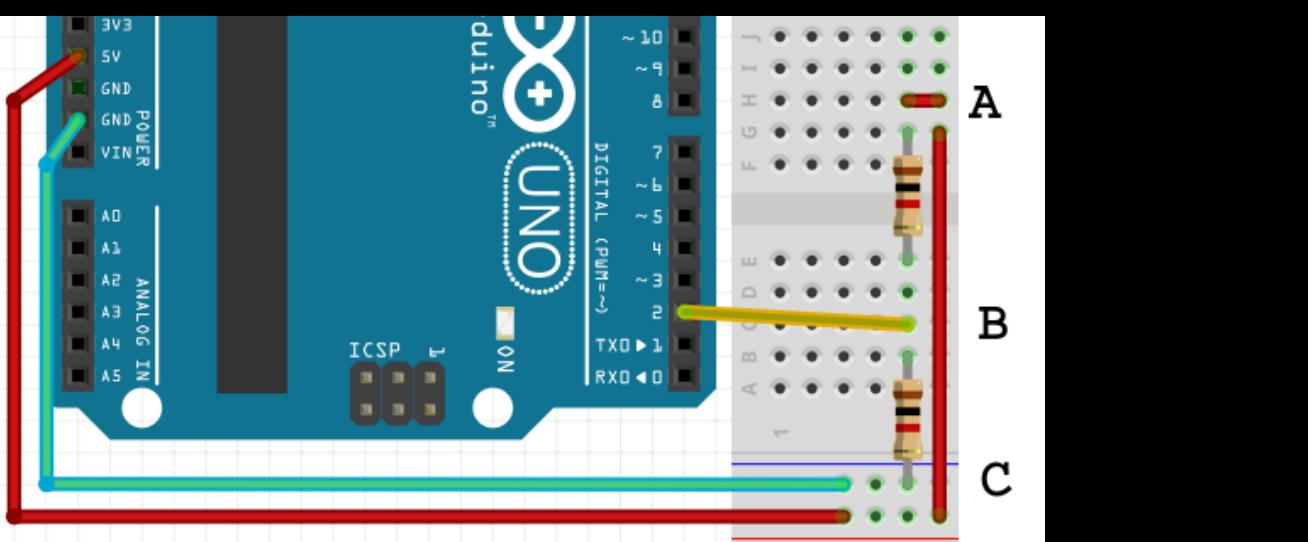
## 0.8 Voorbeeldresultaten

Plek knop	Weerstand tussen A en B (Ohm)	A-C	B-C
Links	0	1000	1000
Tussen links en midden in	250	1000	750
Midden	500	1000	500
Tussen midden en rechts in	750	1000	250
Rechts	1000	1000	0

8

## 0.9 Hoe zit dit in elkaar?

- Hoe kan ik dit vervangen door weerstanden?



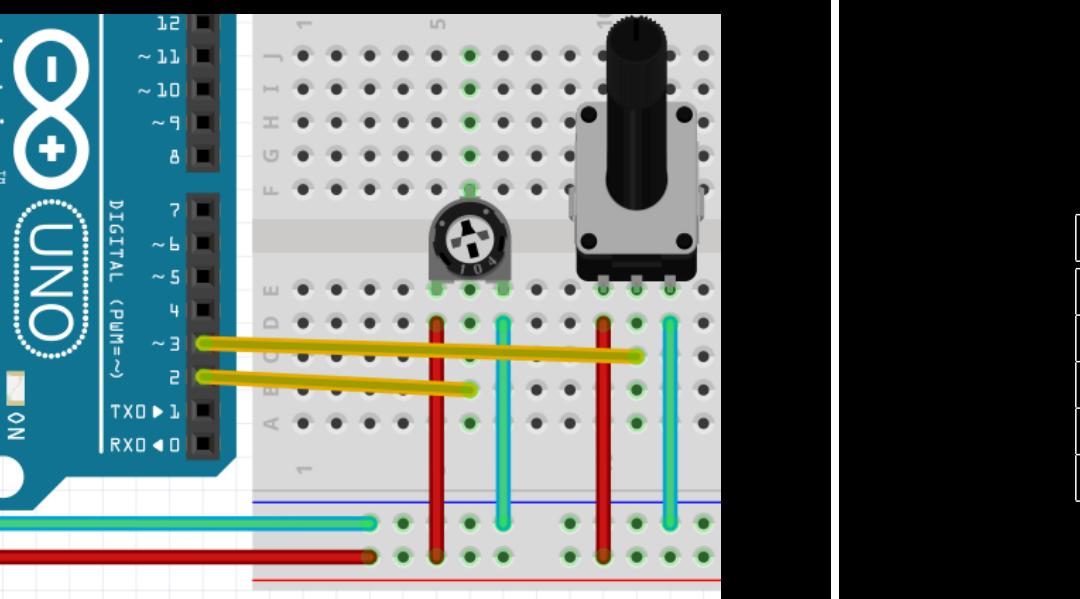
9

Hoe sluit ik een LCD schermpje aan?

(C) Richèl Bilderbeek

August 25, 2014

## 0.10 Aansluiten potmeter



10

## 0.1 Overzicht

- Doel
- Hardware
- Software

## 0.11 Meten met Arduino

- Maak een tabel en meet met de Arduino
- Wat valt je op? Hebben we dit eerder gedaan? Laat een Arduino reageren

Plek knop	Ik voorspel dat ik ga meten...	Ik heb gemeten...
Links		1023
Tussen links en midden in		768
Midden		512
Tussen midden en rechts in		256
Rechts		0

11

## 0.2 Doel

- Hoe sluit ik een LCD display aan?
- Hoe gebruik ik het LCD display in mijn code?

## 0.12 Voorbeeldresultaten

- Wat valt je op? Hebben we dit eerder gedaan? Laat een Arduino reageren

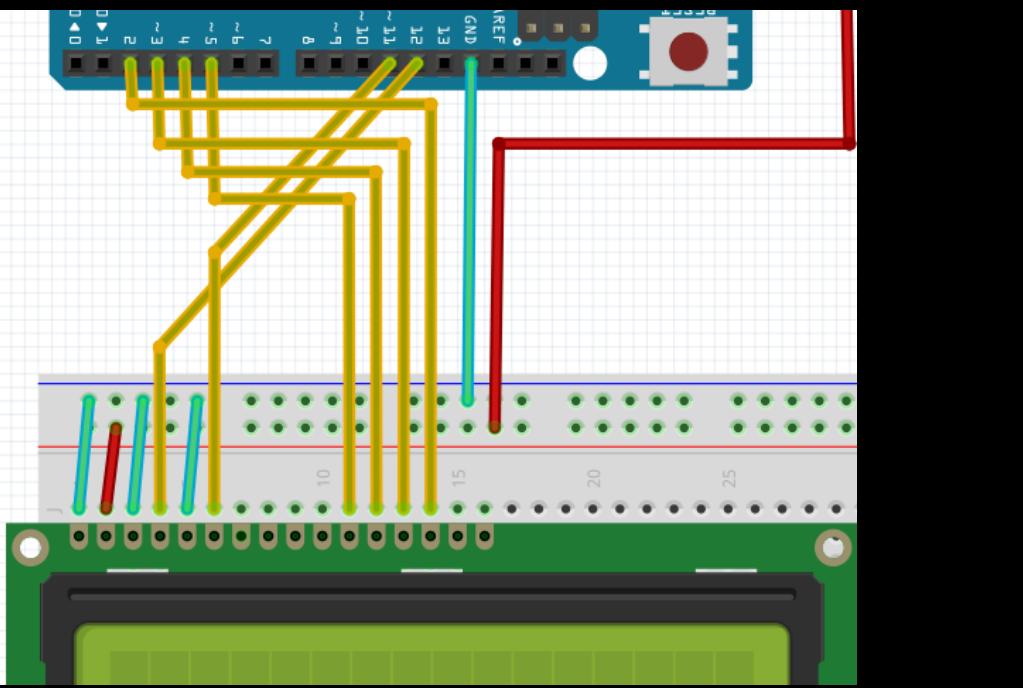
Plek knop	Ik voorspel dat ik ga meten...	Ik heb gemeten...
Links		1023
Tussen links en midden in		768
Midden		512
Tussen midden en rechts in		256
Rechts		0

12

## 0.3 Hardware

- Veel draden, maar met Fritzing en Arduino IDE kun je bedenken hoe je een LCD aan moet sluiten
- Twee keuzes: contrast en backlight
  - Contrast: pin 3, verbinden naar GND of met potmeter
  - Backlight: verbinden (samen met 220 Ohm) of niet

## 0.4 Hardware



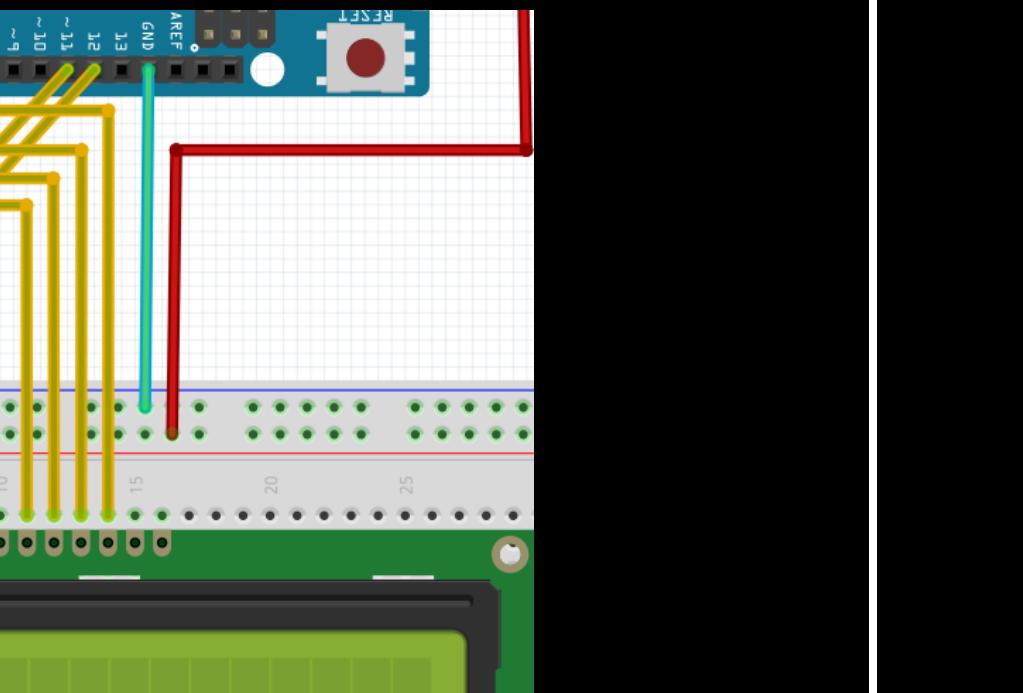
4

## 0.8 Software

- In Arduino IDE, 'Help | Reference', dan 'Libraries | LiquidCrystal':
  - begin(int letters, int regels): bepaal schermgrootte
  - clear(): maak scherm leeg
  - setCursor(int x, int y): bepaal positie cursor
  - print(...): zet tekst op het scherm, vanaf de cursor
  - createChar(int index, byte karakter[8]): sla een eigen 5x8 karakter op
  - write(int index): zet een karakter op het scherm, vanaf de cursor

8

## 0.5 Hardware



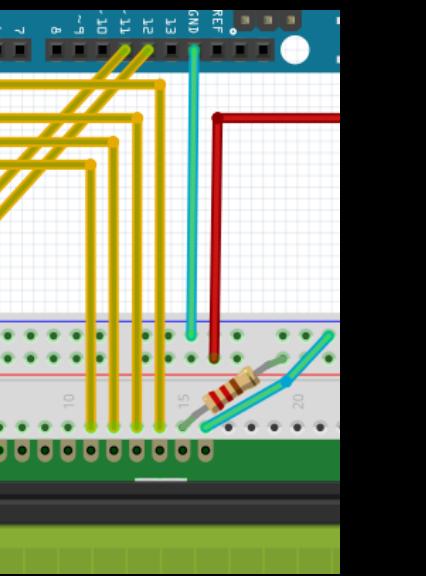
5

## 0.9 Software

- In Arduino IDE, 'Help | Reference', dan 'Libraries | LiquidCrystal':
  - home(): setCursor(0,0)
  - cursor(), noCursor(): maak cursor zichtbaar/onzichtbaar
  - blink(), noBlink(): laat cursor wel/niet knipperen
  - display(), noDisplay(): laat tekst wel/niet zien
  - scrollDisplayLeft(), scrollDisplayRight(): verplaats alle tekst met cursor naar links/rechts
  - autoscroll(), noAutoscroll(): zet autoscroll aan/uit
  - leftToRight(), rightToLeft(): richting tekst is links-naar-rechts/rechts-naar-links

9

## 0.6 Hardware



6

## 0.7 Software

- Voorbeeld in Arduino IDE: Examples | LiquidCrystal | Hello World

```
#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

void setup() {
  lcd.begin(16, 2);
  lcd.print("hello, world!");
}

void loop() {
  lcd.setCursor(0, 1);
  lcd.print(millis()/1000);
}
```

7

## 0.1 Overzicht

1. Doel
2. Er is een probleem? 40 mA per pin
3. Wat weten we?
4. Er is een probleem?
5. Hoe gaan we daar mee om?
6. Opdracht: vind dit uit! Noteer in je logboek

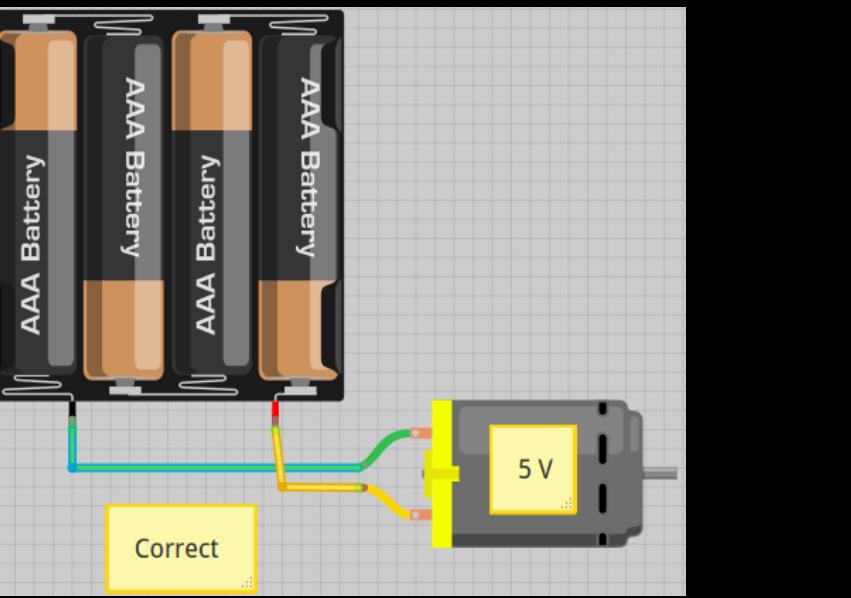
1

(C) Richèl Bilderbeek

August 25, 2014

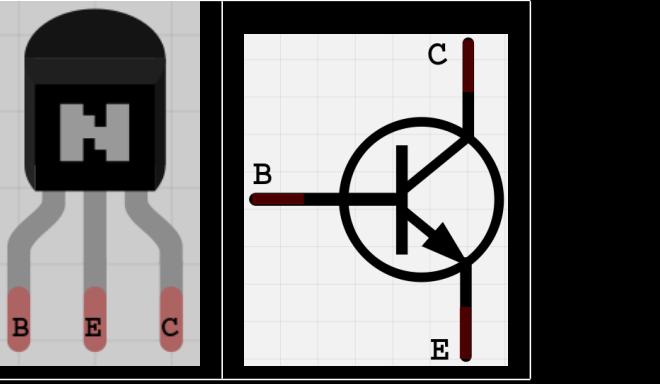
## 0.2 Doel

- Sluit een gelijkstroommotor aan op een Arduino



2

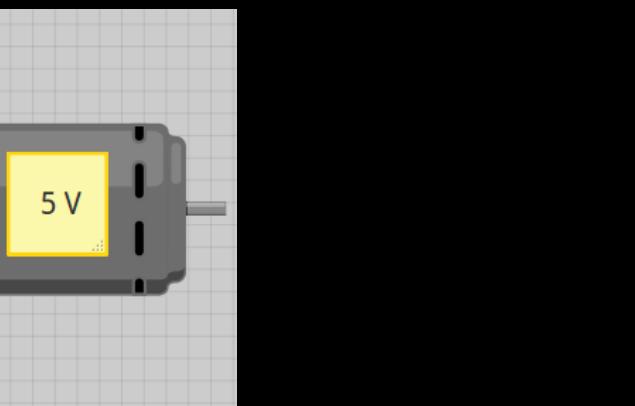
## 0.6 Transistor



- C: Collector ('kollektor'): naar vermogensbron
- B: Base ('bees'): aansturing door Arduino
- E: Emitter ('iemitter'): naar Aarde

6

## 0.3 Naieve oplossing



3

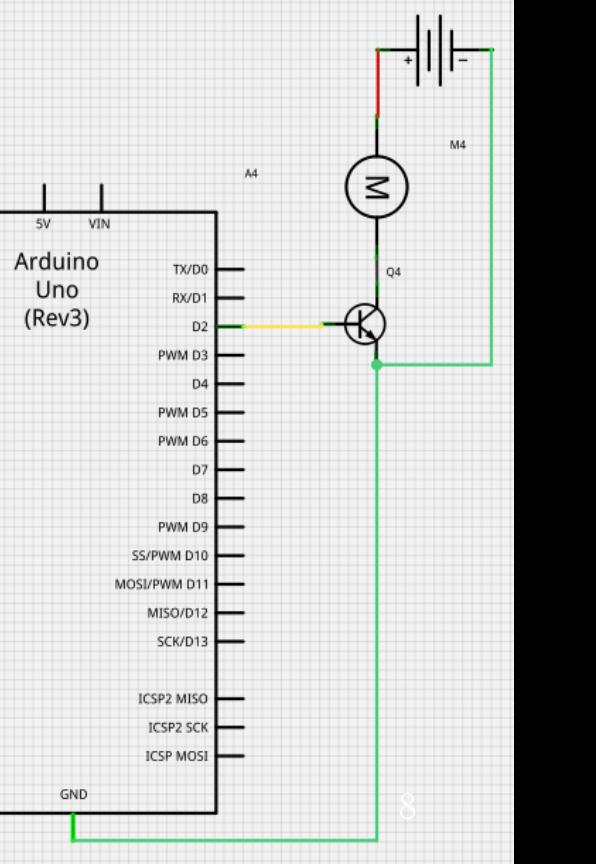
## 0.4 Wat is het probleem?

- Een Arduino kan niet veel<sup>1</sup> stroom leveren
- Een Arduino is een micro-controller
- Iets dat veel vermogen<sup>2</sup> nodig heeft, moet indirect aangestuurd worden

<sup>1</sup>40 mA per pin  
<sup>2</sup>Vermogen = Spanning x stroom

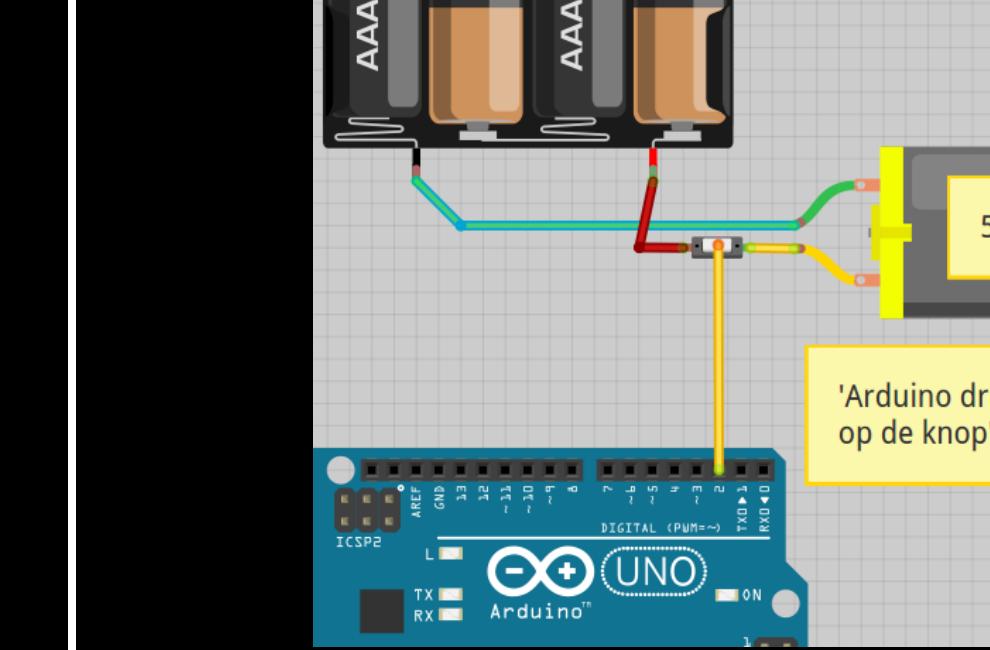
4

## 0.8 Beter



8

## 0.5 Goed idee



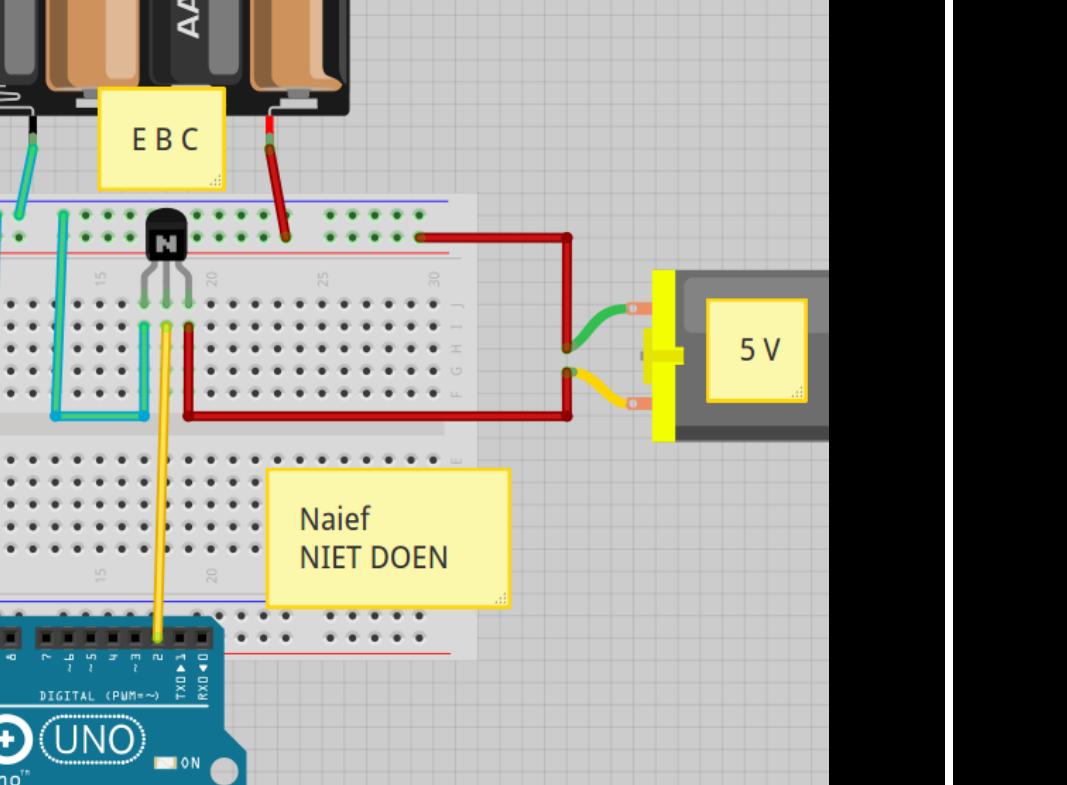
5

## 0.9 Motor

- Een motor kan een terugslag hebben
- Hierdoor wil de motor een omgekeerde stroom laten lopen
- Dit kan de Arduino beschadigen!
- Er is een onderdeel die deze omgekeerde stroom kan laten lopen

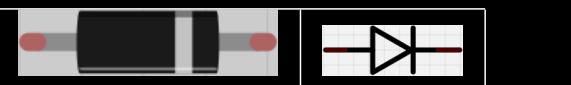
9

## 0.7 Beter



7

## 0.10 Diode



- Eenrichtingsweg voor stroom
- Stroom loopt van plus naar min
- Bijvoorbeeld van 5V naar GND

10

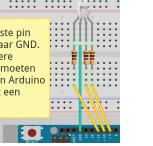


Figure 2: Aansluiten RGB LED op een slimme manier

Om een kleur van een RGB lampje aan te sturen kan je gebruik maken van het voorbeeld 'Fade'.<sup>2</sup>

Het laat schijnen van een LEDje gebeurt met de functie 'analogWrite', bijvoorbeeld als volgt:

`analogWrite(9, 127);`

Hierin wordt naar pin 9 de waarde 127 gestuurd, waardoor de LED op halve sterke gaat branden.

Shift een RGB LED aan. Laat deze elke seconde een andere kleur geven, in de volgorde rood-groen-blauw; groen, cyan (=groen + blauw), blauw, magenta (=rood + blauw)

Algoritme 1 is een mogelijke oplossing.

### 3 Aansluiten lichtsensor

Als je naïf bent, sluit je een lichtsensor aan als in figuur 3. Om de waarde van een lichtsensor te lezen, kan je het voorbeeldprogramma 'AnalogReadSerial'<sup>3</sup> gebruiken.

Het lezen van een sensor gebeurt met 'analogRead', bijvoorbeeld als volgt:

`int mijn_sensort_waarde = analogRead(A0);`

Hierin wordt de waarde van pin A0 gelezen en opgeslagen in de variabele 'mijn\_sensort\_waarde'.

• Kun je niet voorspellen welke waarde je gaat meten? Bouw dit schema en noteer het resultaat.

Als je weet dat figuur 3 niet werkt, bouw je figuur 4.

• Schrijf op welke waarde je meet in het donker en in het volle licht

<sup>2</sup>File -> Examples -> 01\_Basics -> Fade

<sup>3</sup>File -> Examples -> 01\_Basics -> AnalogReadSerial

2

## 0.11 Transistor

- Een transistor heeft weinig weerstand
- Het is gemakkelijk per ongeluk B-E kort te sluiten
- Zet dus een weerstand voor de B
- Begin met een hoge weerstand, bijvoorbeeld 10 kOhm

11

**Algoritme 1 RGB LED voorbeeld 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 GeefKleur(
  const int roodheid,
  const int groothed,
  const int blauwheid)
{
  analogWrite(pin_rood, r);
  analogWrite(pin_groen, g);
  analogWrite(pin_blaauw, b);
}

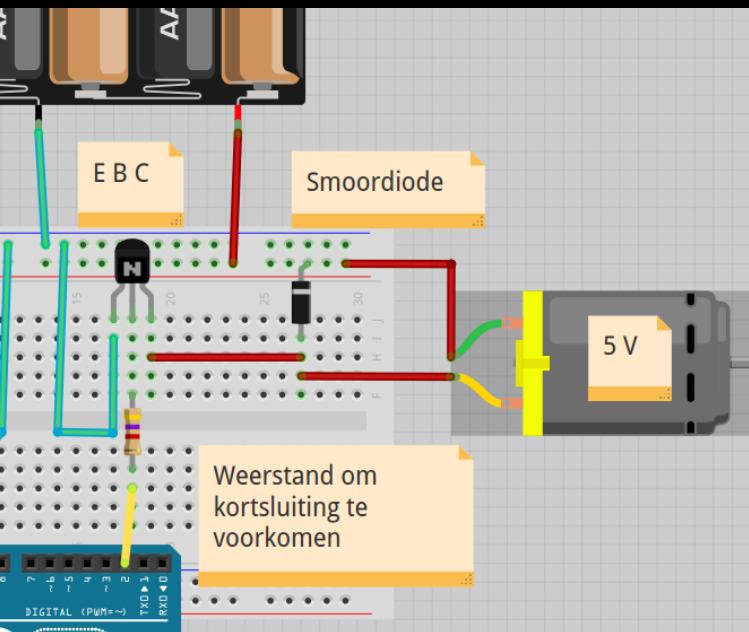
void loop() {
  GeefKleur(255, 0, 0); delay(500);
  GeefKleur(255, 255, 0); delay(500);
  GeefKleur( 0, 255, 0); delay(500);
  GeefKleur( 0, 255, 255); delay(500);
  GeefKleur( 0, 0, 255); delay(500);
  GeefKleur(255, 0, 255); delay(500);
}
```



Figure 3: Lichtsensor naïef aangesloten

3

## 0.12 Correct



12

Project: kleurensensor

(C) Richèl Bilderbeek

August 25, 2014

### 1 Introductie

Een lichtsensor kan geen kleur herkennen. Met behulp van een RGB LED kan dit wel.

Dit project bestaat uit meerdere onderdelen:

- Aansluiten RGB LED

- Aansluiten lichtsensor

- LED laten reageren op sensor

- Omprogrammeren naar kleurensensor

### 2 Aansluiten RGB LED

Als je naïf bent, sluit je een RGB LED aan als figuur 1.

- Wat is onjuist aan figuur 1? Wat zou er gebeuren als je dit zo aansluit?

Als je slim bent, sluit je een RGB LED aan als figuur 2.

(ook wel 'licht-naïf' of 'veelklik' weerstand' of 'LDR' (Light Dependent Resistance) genoemd)

Figure 1: Aansluiten RGB LED op een naïeve manier

1

Hoe voeg ik code samen?

(C) Richèl Bilderbeek

August 25, 2014

4



Figure 4: Lichtsensor juist aangesloten

En dimmer manier is de 'map' functie te gebruiken:

```
const int gemeten_waarde = analogRead(sensort_pin);
```

```
const int waarde_voor_led = map(gemeten_waarde, 400, 600, 0, 255);
```

```
analogWrite(led_pin, waarde_voor_led);
```

• Laat de RGB LED reageren op de gemeten sensorenwaarde. Laat deze bijvoorbeeld wit schijnen als er veel licht is.

### 5 Omprogrammeren naar kleurensensor

Last de LED om de rood, groen en blauw branden. Laat de lichtsensoren de hoeveelte rood, groen en blauwwaarde meten van het ding waarop je schijnt. Laat dan de LED de kleur branden van het ding.

## 0.1 Overzicht

1. Standaard volgorde
2. Regels samenvoegen
3. Conclusie

1

## 0.5 Samenvoegen normale functies

- Hou de niet-setup en niet-loop functies bij de functies
- Houdt de volgorde aan van de losse programma's

```
void DraaiServo() { /* */}  
en  
void ZetAan() { /* */}  
void ZetUit() { /* */}  
void Knipper() { /* Roeft ZetAan en ZetUit aan */ }  
wordt  
void DraaiServo() { /* */}  
void ZetAan() { /* */}  
void ZetUit() { /* */}  
void Knipper() { /* Roeft ZetAan en ZetUit aan */ }
```

5

## 0.6 Samenvoegen setup functies

- Voeg de code van de setup functie samen tot een setup functie
- Houdt de volgorde aan van de losse stukken code

```
void setup() { my_servo.attach(pin_servo); }  
en  
void setup() { const int breedte{16}; const int hoogte{2};  
my_lcd.begin(breedte, hoogte); }  
wordt  
void setup() {  
my_servo.attach(pin_servo);  
const int breedte{16}; const int hoogte{2};  
my_lcd.begin(breedte, hoogte); }
```

6

## 0.2 Standaard volgorde

- Includes
- Globale variabelen
- Functies, waaronder 'setup' en 'loop'

```
#include <LiquidCrystal.h>  
LiquidCrystal lcd(12,11,5,4,3,2);  
  
void ZegHallo() { lcd.print("Hello World!"); }  
  
void setup() { lcd.begin(16, 2); }  
  
void loop() { ZegHallo(); }
```

2

## 0.3 Samenvoegen includes

- Hou de includes bij de includes
- Elke volgorde is goed

```
#include <Servo.h>  
en  
#include <LiquidCrystal.h>  
wordt bijvoorbeeld  
#include <LiquidCrystal.h>  
#include <Servo.h>
```

3

## 0.4 Samenvoegen globale variabelen

- Hou de globale variabelen bij de globale variabelen
- Houdt de volgorde aan van de losse programma's
- Check of bijvoorbeeld pinnummers niet dubbel worden gebruikt

```
const int pin_servo{9};  
en  
const int pin_led_red{8};  
const int pin_led_red{pin_led_red + 1};  
wordt  
const int pin_servo{9};  
const int pin_led_red{10};  
const int pin_led_red{pin_led_red + 1};
```

4

## 0.7 Samenvoegen loop functies

- Voeg de code van de loop functie samen tot een loop functie
- Houdt de volgorde aan van de losse stukken code
- Bedenk wat je wilt dat het samengevoegde programma doet

```
for (int pos{0}; pos!=180; ++pos)  
{  
myservo.write(pos);  
delay(1000);  
}  
en  
void loop()  
{  
Serial.println(analogRead(pin_sensor));  
delay(1000);  
}
```

7

## 0.8 Voorbeeld

```
void loop()  
{  
for (int pos{0}; pos!=180; ++pos)  
{  
myservo.write(pos);  
delay(1000);  
}  
en  
void loop()  
{  
Serial.println(analogRead(pin_sensor));  
delay(1000);  
}
```

8

## 0.9 Naieve samenvoeging

```
void loop ()  
{  
    for (int pos{0}; pos!=180; ++pos)  
    {  
        myservo.write(pos);  
        delay(1000);  
    }  
    Serial.println(analogRead(pin_sensor));  
    delay(1000);  
}
```

9

## 0.1 Overzicht

1. Doel
2. Probleem
3. Oplossing

1

## 0.2 Doel

- Waarvoor kun je arrays gebruiken?
- Wat is een array?
- Hoe gebruik je arrays?

2

## 0.10 Juiste samenvoeging

```
void loop ()  
{  
    for (int pos{0}; pos!=180; ++pos)  
    {  
        Serial.println(analogRead(pin_sensor));  
        myservo.write(pos);  
        delay(1000);  
    }  
}
```

10

## 0.11 Conclusie

- De includes zijn gemakkelijk samen te voegen
- Let op bij het samen voegen van globale variabelen
- Normale functies zijn gemakkelijk samen te voegen
- De setup functies zijn gemakkelijk samen te voegen
- Bij het samenvoegen van de loop functies moet het meest worden nagedacht

3

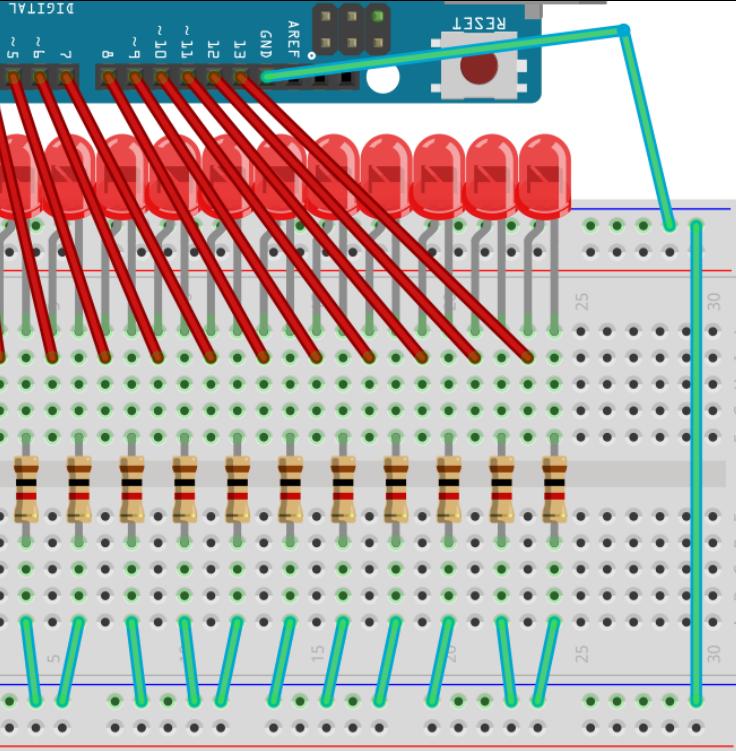
Arrays

(C) Richel Bilderbeek 

August 25, 2014

11

## 0.3 Probleem



3

## 0.4 Probleem

```
const int pin_1 = 3;  
// ...  
const int pin_12 = 13;
```

```
void setup()  
{  
    pinMode(pin_1,OUTPUT);  
    // ...  
    pinMode(pin_12,OUTPUT);  
}
```

4

## 0.5 Probleem

```
void loop ()  
{  
    digitalWrite (pin_1,HIGH)  
    //...  
    digitalWrite (pin_12,HIGH)  
    delay (1000);  
    digitalWrite (pin_1,LOW)  
    //...  
    digitalWrite (pin_12,LOW)  
    delay (1000);  
}
```

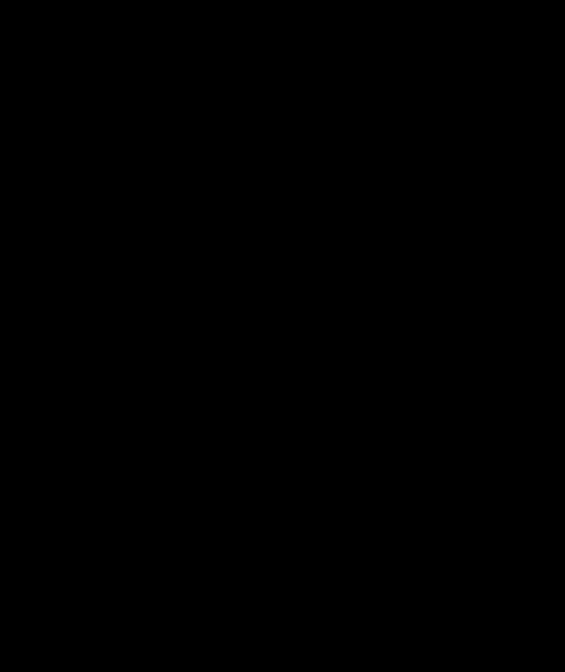
5

## 0.9 Oplossing probleem

```
const int pins [] = { 2,3,4,5,6,7,8,9,10,11,12 };  
const int n_pins = 12;  
  
void setup ()  
{  
    //Vaak in boeken: 'for (int i=0; i<n_pins; i++)'  
    for (int i{0}; i!=n_pins; ++i)  
    {  
        pinMode (pins [ i ] ,OUTPUT);  
    }  
}
```

9

## 0.6 Wat is een array?



6

## 0.10 Oplossing probleem

```
void loop ()  
{  
    for (int i{0}; i!=n_pins; ++i)  
    {  
        digitalWrite (pins [ i ] ,HIGH);  
    }  
    delay (1000);  
  
    for (int i{0}; i!=n_pins; ++i)  
    {  
        digitalWrite (pins [ i ] ,HIGH);  
    }  
    delay (1000);  
}
```

10

## 0.7 Hoe gebruik je een array?

```
int getallen [] = {1,4,8};  
const int eerste_getal = getallen [0];  
getallen [2] = 9;
```

7

## 0.8 Oplossing probleem, naief

```
const int pins [] = { 2,3,4,5,6,7,8,9,10,11,12 };  
const int n_pins = 12;  
  
void setup ()  
{  
    //C++ begint bij index nul te tellen  
    pinMode (pins [ 0 ] ,OUTPUT);  
    pinMode (pins [ 1 ] ,OUTPUT);  
    //...  
    pinMode (pins [ 10 ] ,OUTPUT);  
    pinMode (pins [ 11 ] ,OUTPUT);  
    //Nee, pins [12] bestaat niet,  
    //omdat C++ bij index nul begint te tellen!  
    //pinMode (pins [12] ,OUTPUT);  
}
```

8