

Dag 1

(C) Richèl Bilderbeek 

February 12, 2014

0.2 Wie zijn wij?

- Peter
 - Weet veel van elektronica
 - Werkt veel met Arduino, kent veel Arduino projecten
- Richèl
 - Weet veel van programmeren
 - Weet veel van les geven

0.1 Vandaag

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

0.3 Wie zijn jullie?

- Wie ben jij?
- Wat wil je?

Arduino cursus introductie

(C) Richèl Bilderbeek 

January 30, 2014

0.4 Vorm cursus

- De vorm van de cursus is vrij
 - Niveau van de uitleg hoger/lager?
 - Uitleg door deelnemers?
 - Wedstrijd?
 - Grote projecten?
- Samen zoeken naar onze favoriete vorm

0.1 Overzicht

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

0.5 Vorm cursus vandaag

- Schoolse traject
 - begint bij de basis
 - stap-voor-stap uitleg, oefening en nabespreking
- Projectgebaseerde traject
 - vrij

0.6 Hoe handelen we?

- Concentreren: we zijn alleen bezig met Arduino
- Uittesten: fouten maken is goed
- Slim: we gedragen ons als Einsteins
- Samen: we proberen het goede voorbeeld te geven en ons best te doen

0.7 Tijdsindeling avond

- ☑ introductie
- ☐ Wat is Arduino?
- ☐ Ontwerp: LED
- 20:30-21:00 pauze
- ☐ Bouwen: LED
- ☑ Programmeren: LED
- ☐ Ontwerp: schakelbare LED
- ☐ Bouwen: schakelbare LED
- ☐ Programmeren: schakelbare LED

Wat is Arduino?

(C) Richèl Bilderbeek 

February 12, 2014

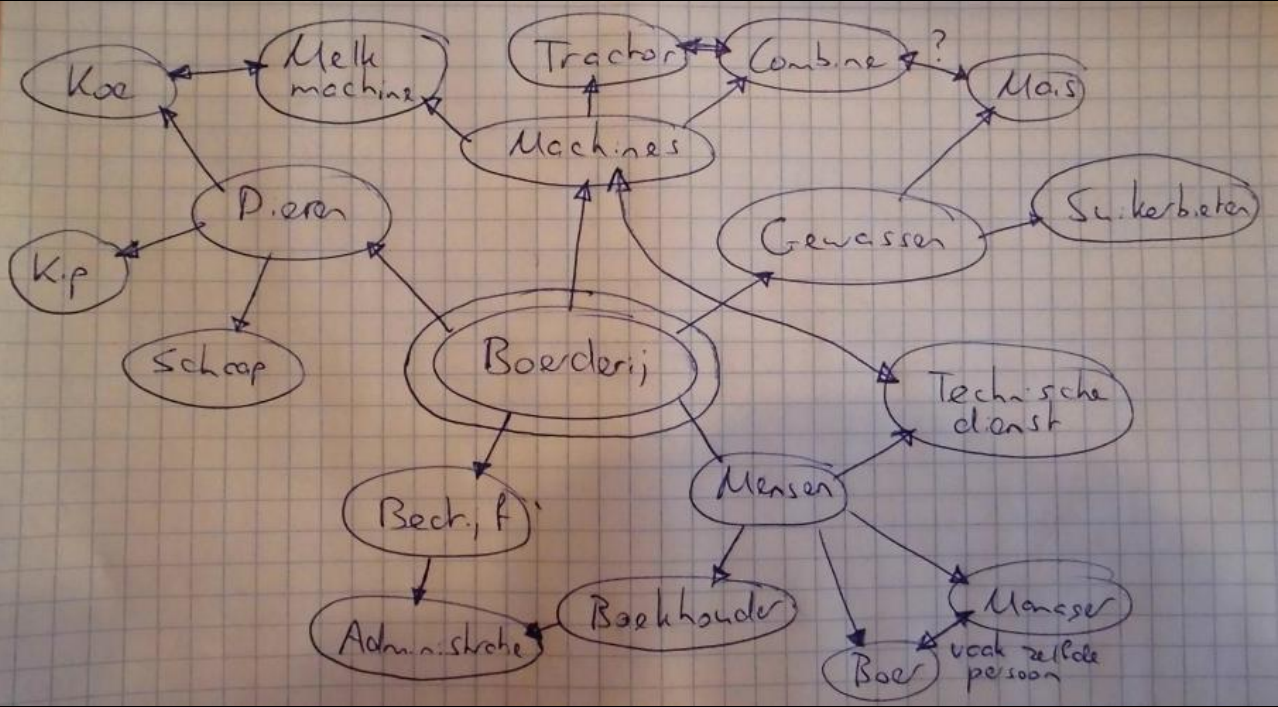
0.1 Overzicht

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

0.2 Wat weten jullie?

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

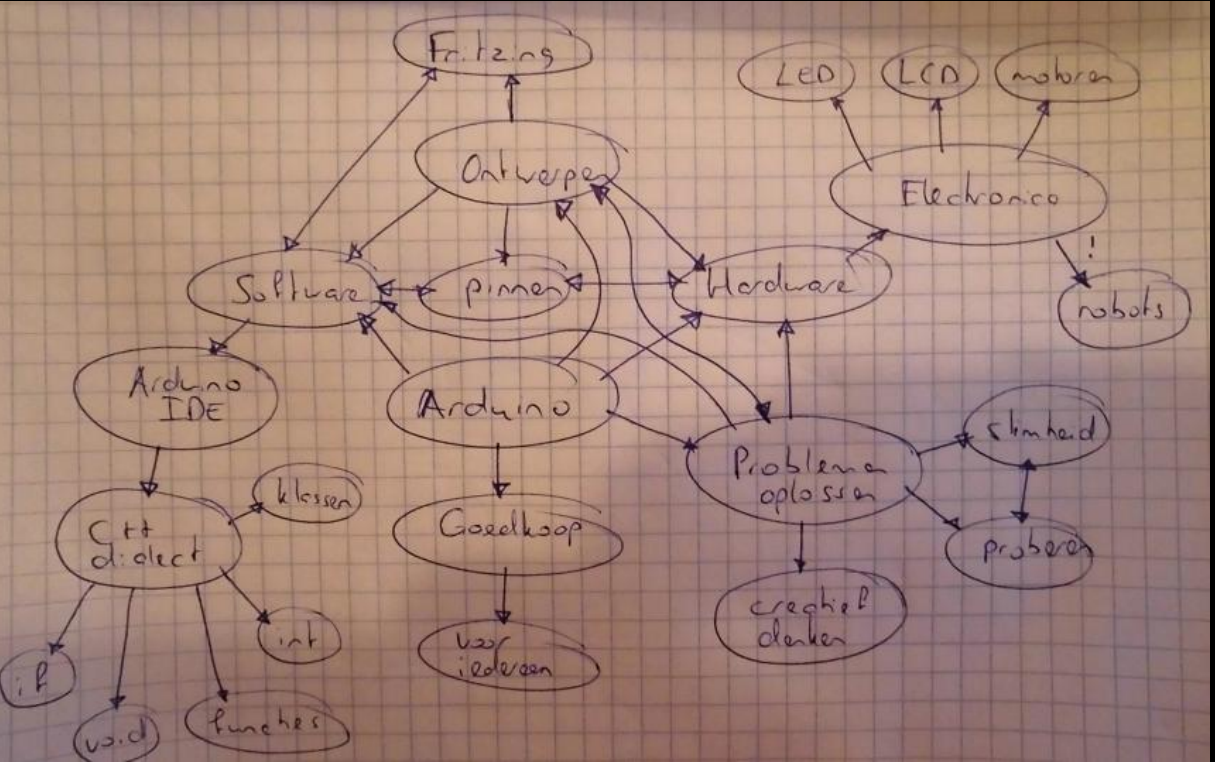
0.3 Wat is een mindmap?



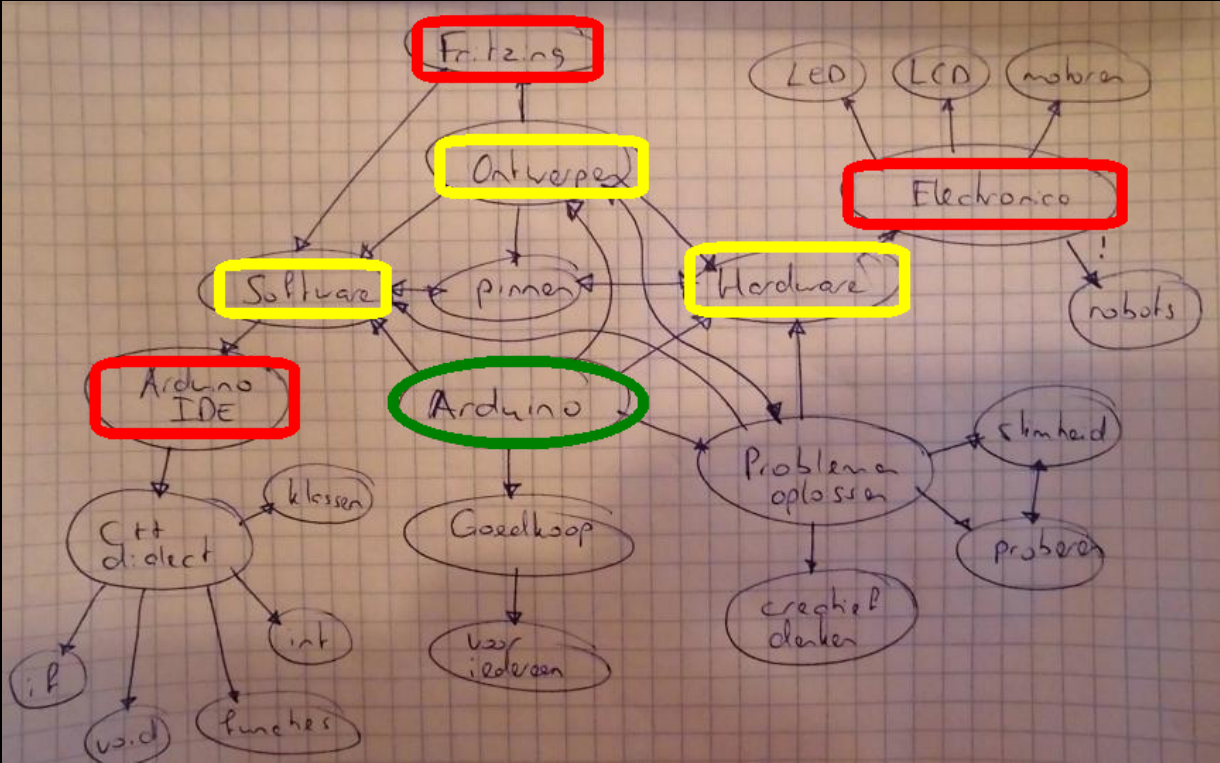
0.4 Maken mindmap

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

0.5 Wat is Arduino volgens Richèl?



0.6 Wat is Arduino volgens Richèl?



6

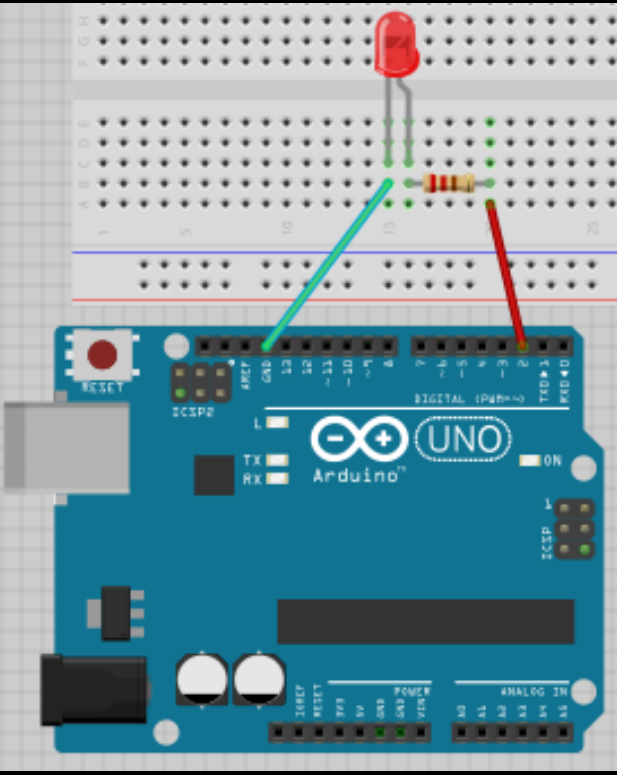
0.10 Conclusie

Wat we doen is divers:

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

Wij bouwen dit stap voor stap op

0.7 Elektronica



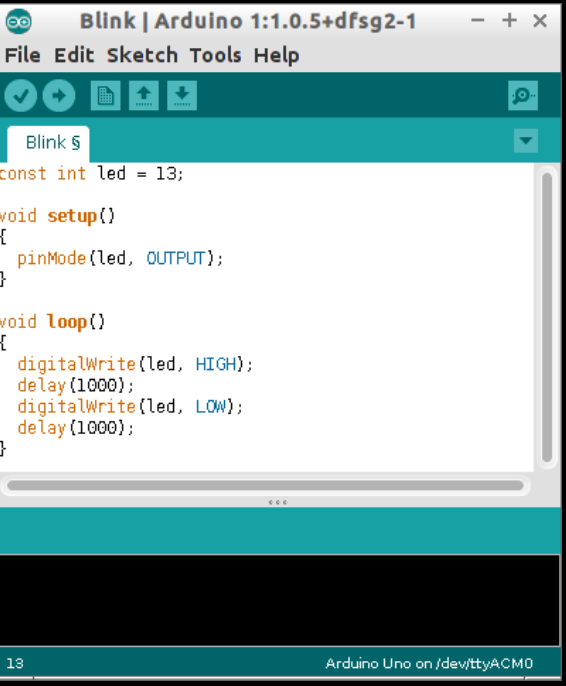
7

Hoe sluit ik een LED aan?

(C) Richèl Bilderbeek 

February 12, 2014

0.8 Software

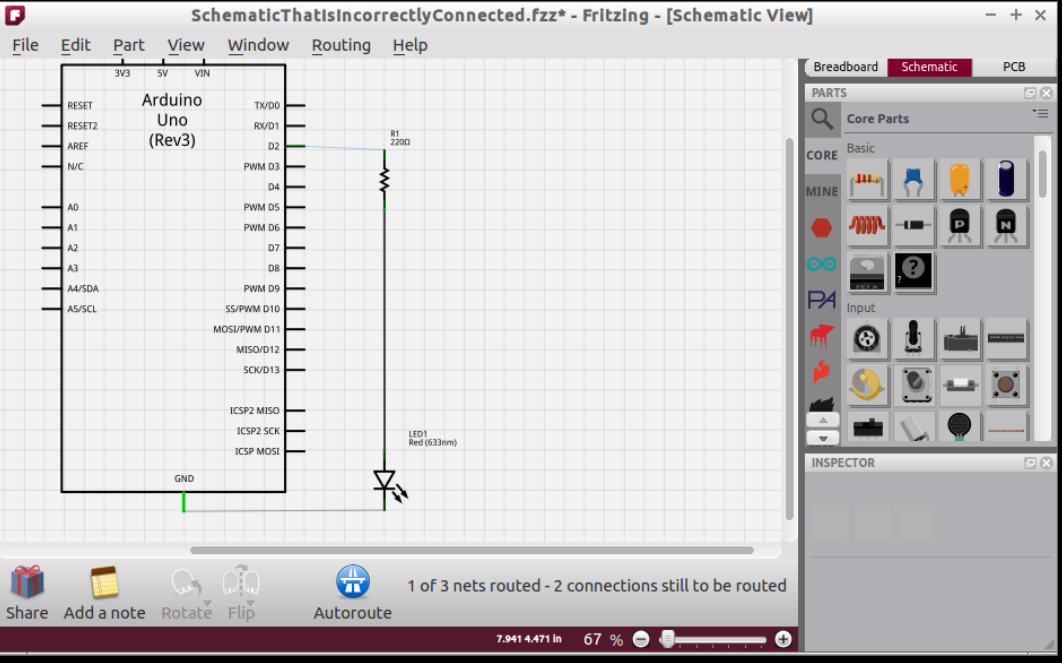


8

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

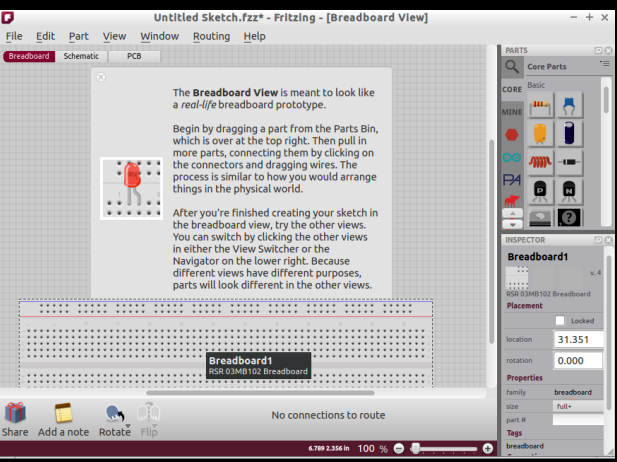
0.9 Ontwerpen



9

0.2 Fritzing

- Vrije software
- Linux, Mac OS X, Windows



0.3 Fritzing installatie

- `sudo apt-get install fritzing`
- `yum install fritzing`
- Download van fritzing.org/download

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

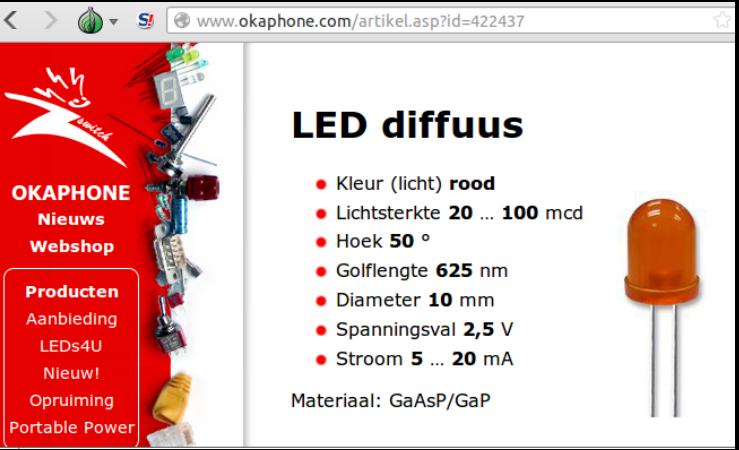
Wat kun hieraan doen?

0.4 Opdracht

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

Welk probleem kom je tegen?

0.8 Antwoorden



Wat kun hieraan doen?

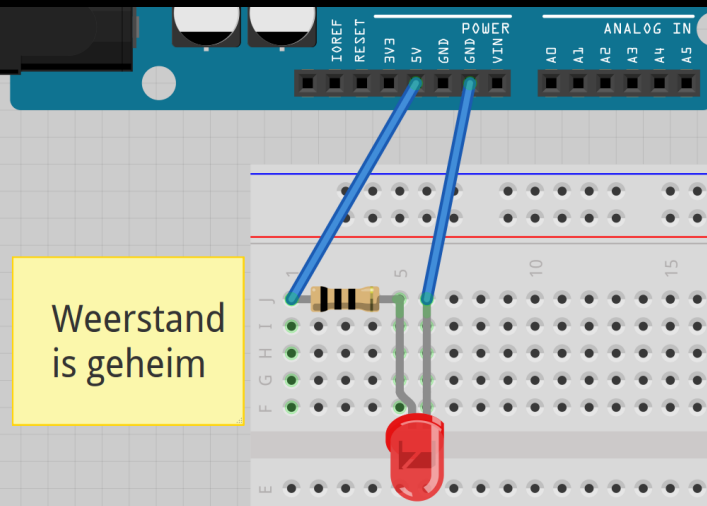
0.5 Er is een probleem?

Ja: je kunt een LED niet direct op een Arduino aansluiten

- Hoe kun je dat merken?
- Dat gaan we doen!
- Waarom is dat zo?

0.9 Hoe hoge spanningen tegen te gaan?

- Met een weerstand!

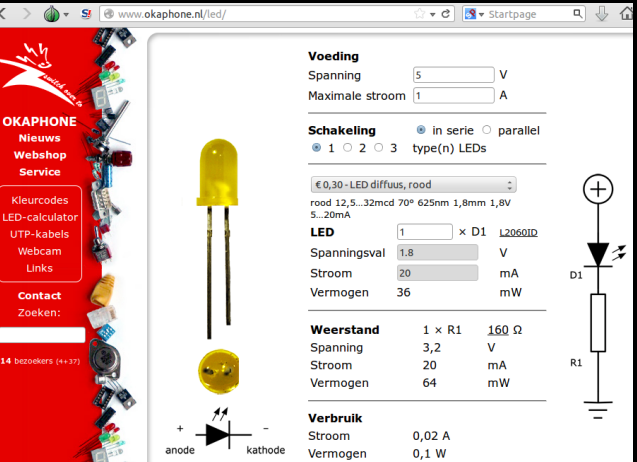


0.6 Wat weten we?

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

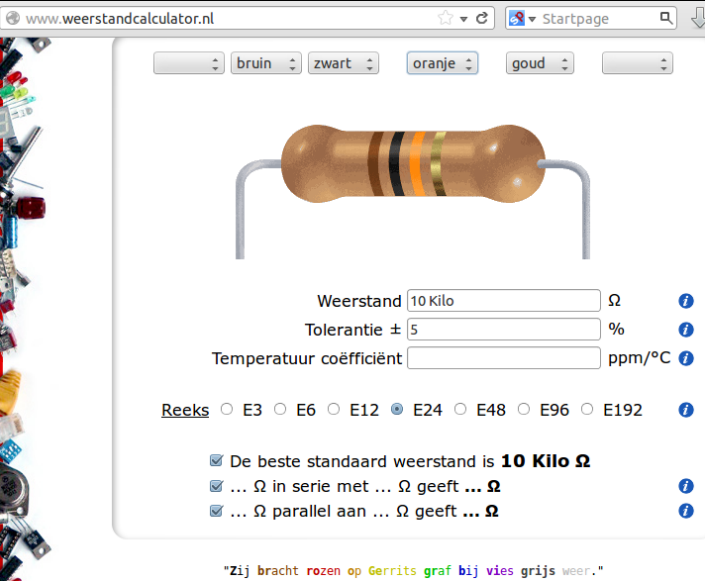
0.10 Welke weerstand?

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



0.11 Welke weerstand?

- Zij Bracht ROzen Op GErrits GRaf Bij Vies GRIJS Weer



0.12 Bouwen!

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

0.13 Geniet

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

Hoe schakel ik een LED?

(C) Richèl Bilderbeek

February 12, 2014

0.1 Overzicht

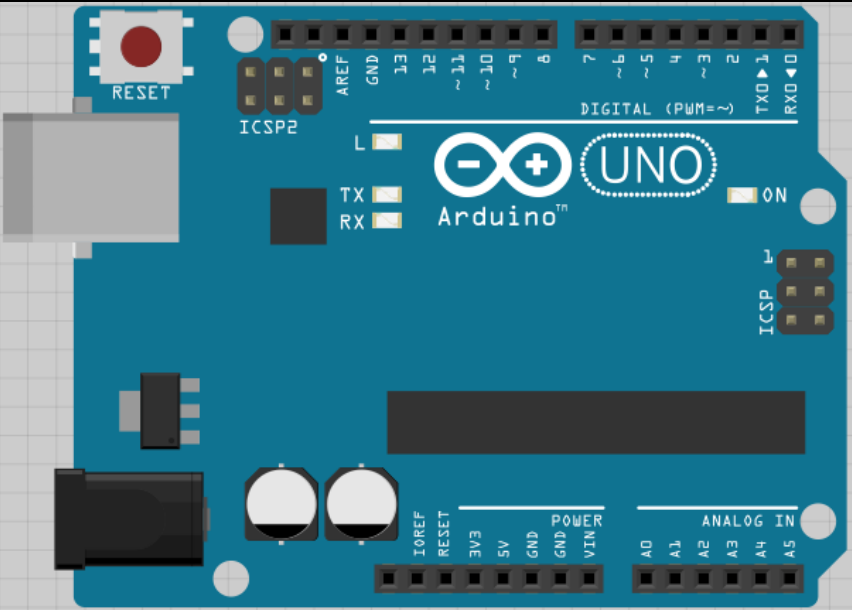
1. Doel
2. Hoe ontwerpen?
3. Arduino IDE
4. Bouwen!

0.2 Doel

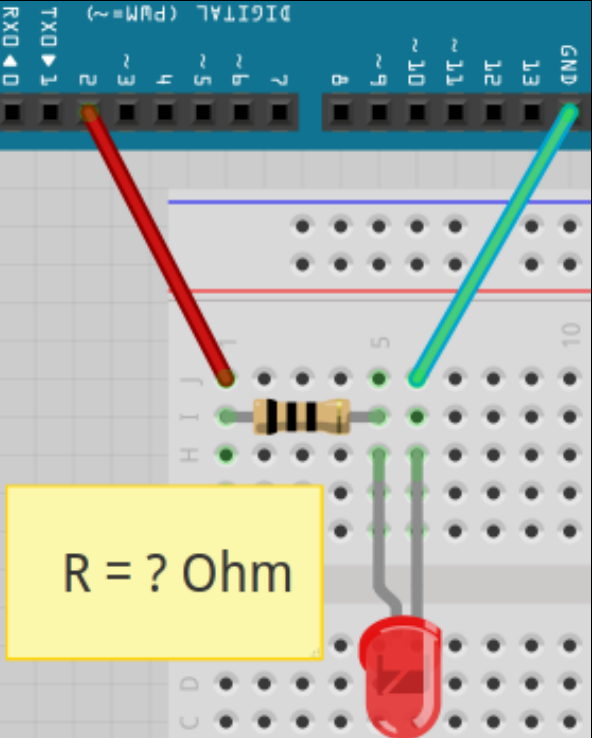
Een machine maken die een LED na een seconde aan doet, en deze een seconde later weer uit doet

0.3 Hoe ontwerpen?

- Waar denk je dat de Arduino dit mee kan?



0.4 Ontwerp



0.5 En nu?

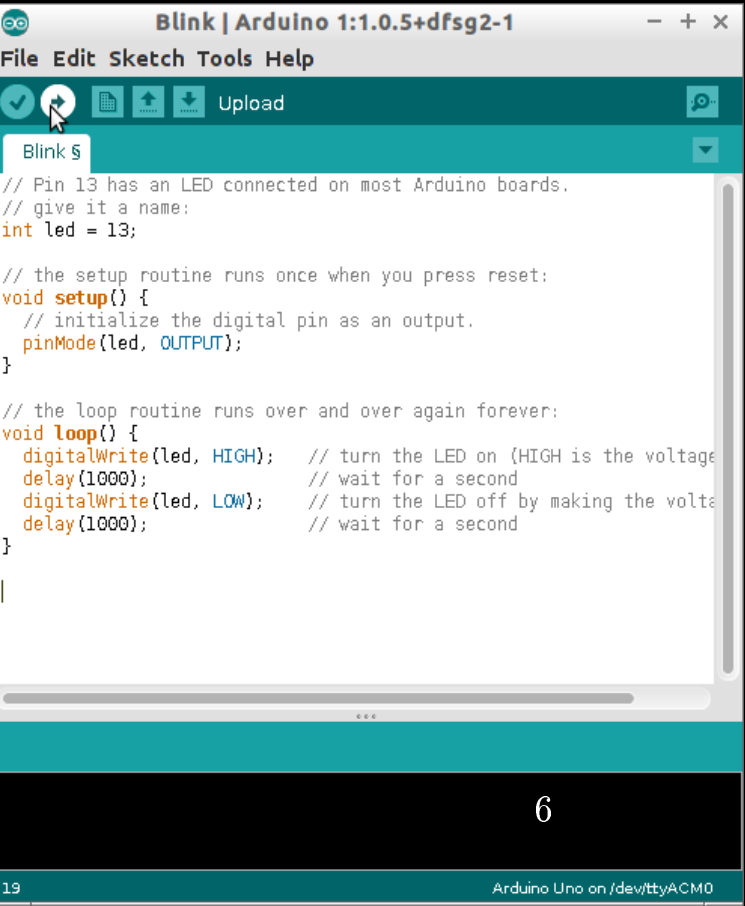
- Programmeren!
- Met Arduino IDE ('Integrated Development Environment' ≈ 'Programmeeromgeving')

Dag 1 Oefeningen

(C) Richèl Bilderbeek 

February 12, 2014

0.6 Arduino IDE



0.1 Overzicht

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

0.7 Arduino IDE installatie

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

0.2 Fritzing installatie

- `sudo apt-get install fritzing`
- `yum install fritzing`
- Download van fritzing.org/download

0.8 Opdracht

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

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

0.4 Arduino IDE installatie

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

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

Dag 2

(C) Richèl Bilderbeek 

February 12, 2014

0.1 Vandaag

1. shortCircuit: kortsluiting & 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

Kortsluiting & Breadboard

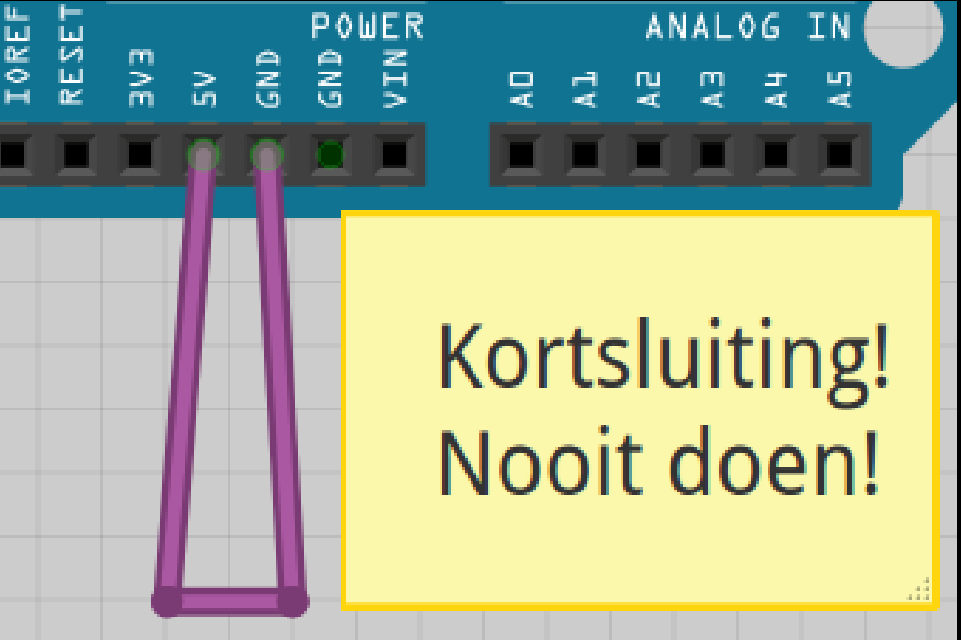
(C) Richèl Bilderbeek 

February 12, 2014

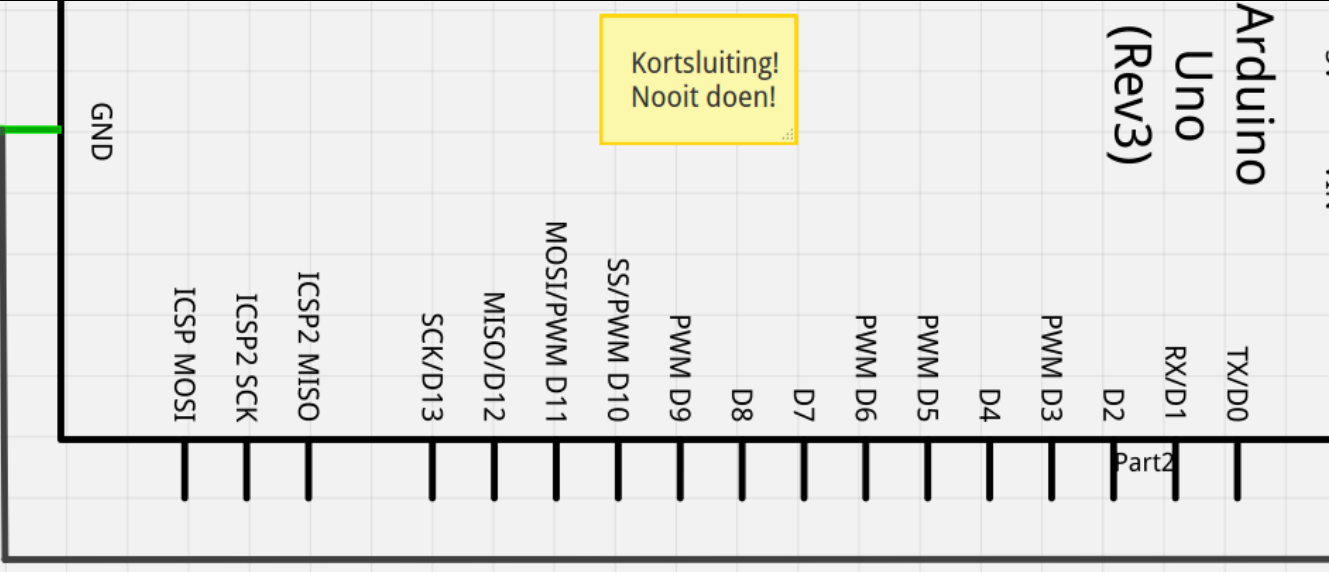
0.1 Overzicht

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

0.2 Wat is kortsluiting?



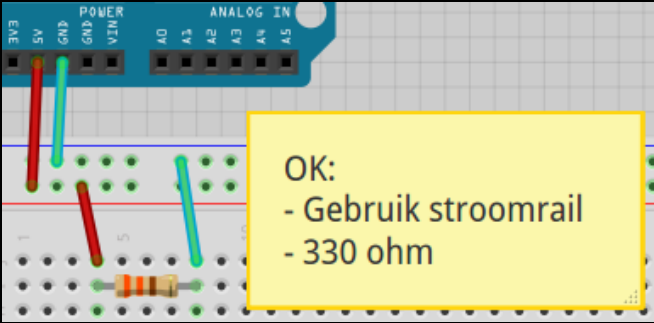
0.3 Wat is kortsluiting?



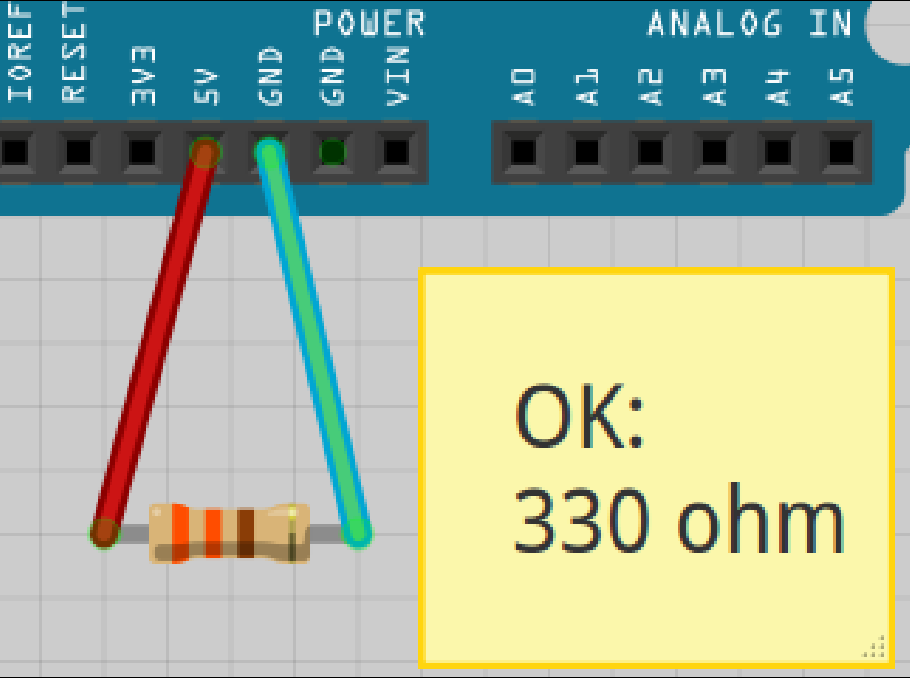
0.4 Hoe dit te voorkomen?

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

0.8 Breadboard



0.5 Wat is kortsluiting?

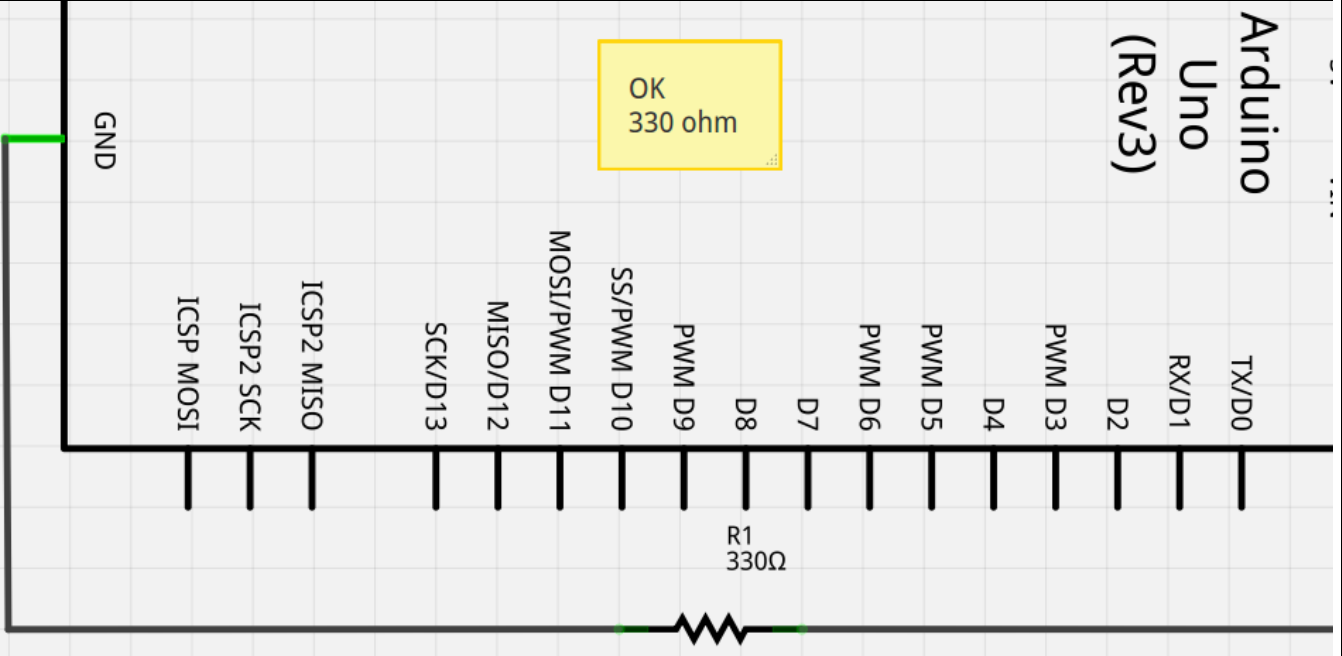


analogRead

(C) Richèl Bilderbeek 

February 12, 2014

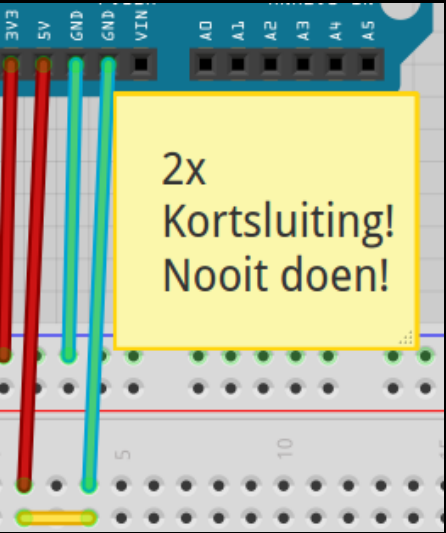
0.6 Wat is kortsluiting?



0.1 Overzicht

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

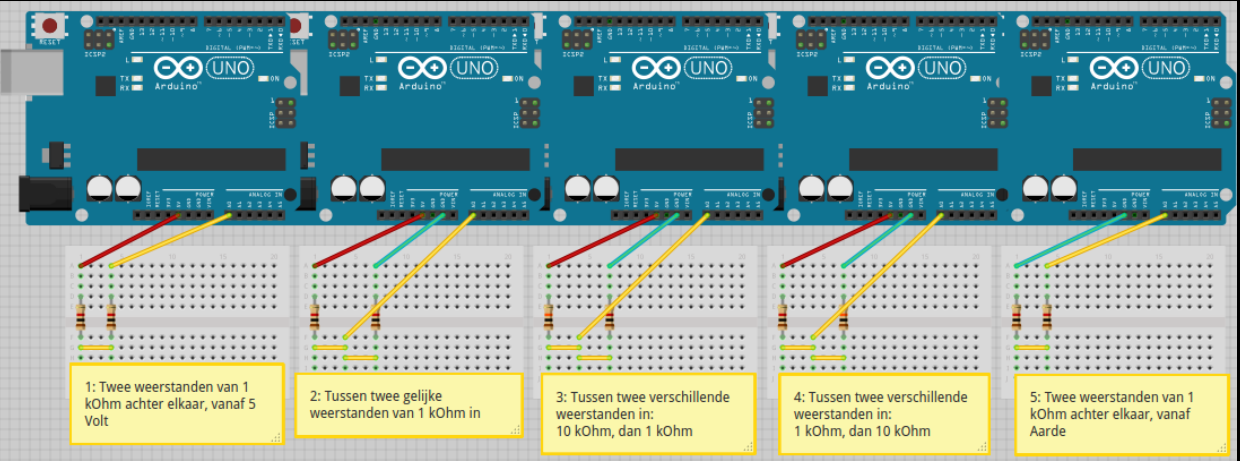
0.7 Breadboard



0.2 Doel

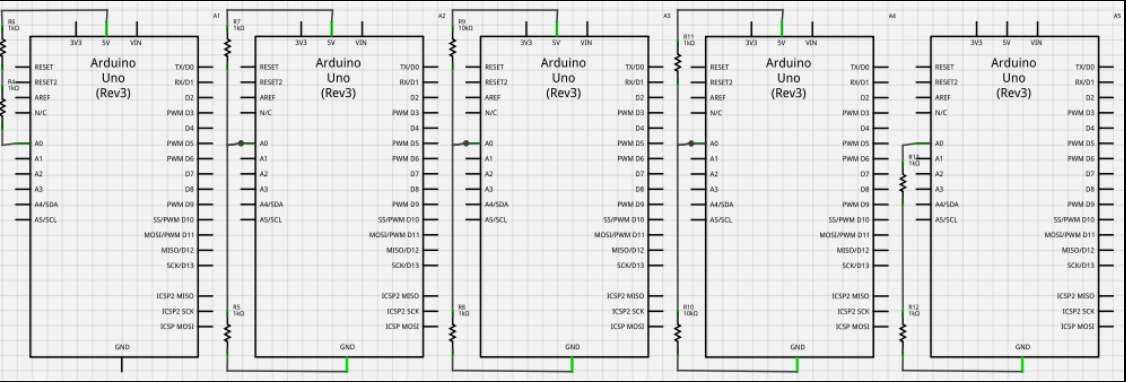
- Spanning meten met Arduino: analogRead
- Basisschakelingen herkennen
- Bouwen van minstens vijf schakelingen
- Dit niet kennen = domme fouten gaan maken: drukknop, sensor, etcetera

0.3 Opdracht



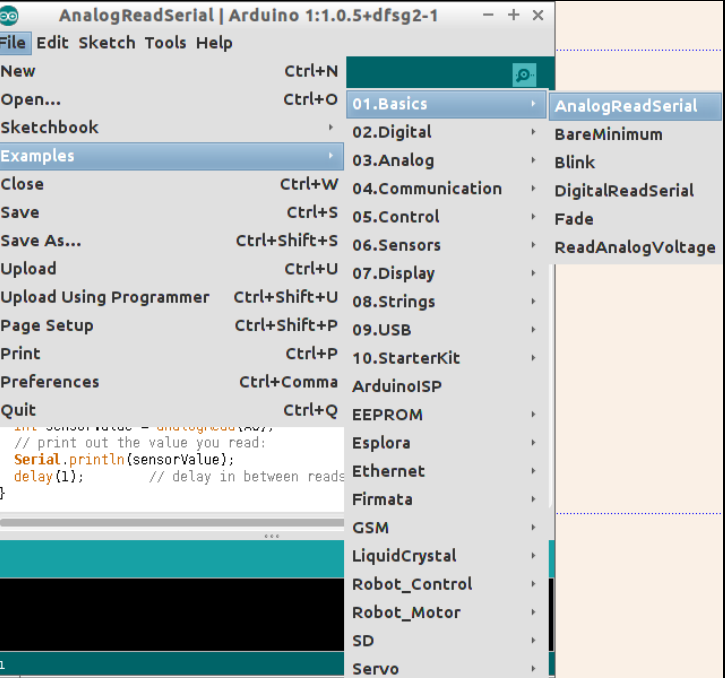
3

0.4 Opdracht



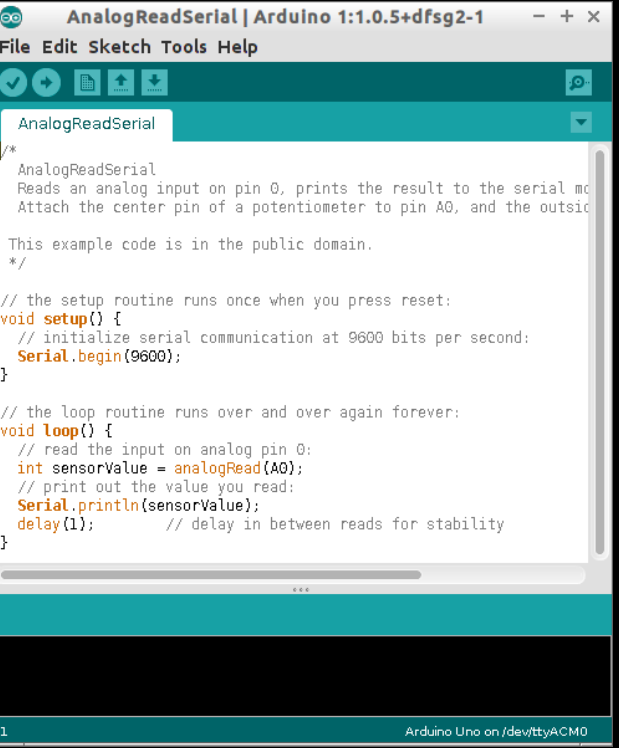
4

0.5 Programma



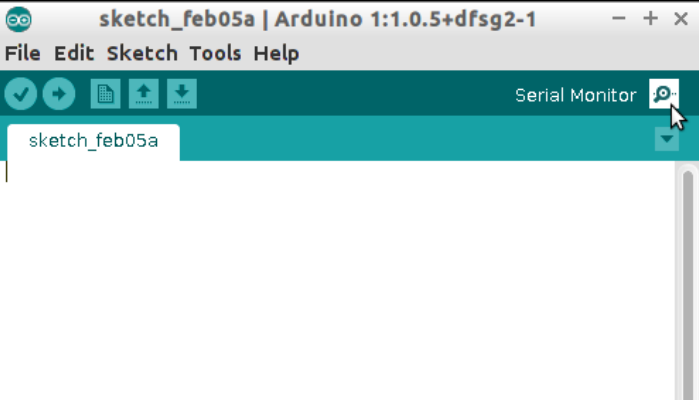
5

0.6 Programma



6

0.7 Bekijken



7

0.8 Opschrijven

- 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!

8

Hoe lees ik input?

(C) Richèl Bilderbeek 

February 12, 2014

0.1 Overzicht

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

1

0.2 Doel

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

0.6 Doen...

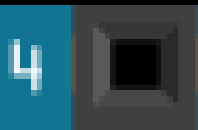
- Volgende slide de oplossing...

0.3 Vraag

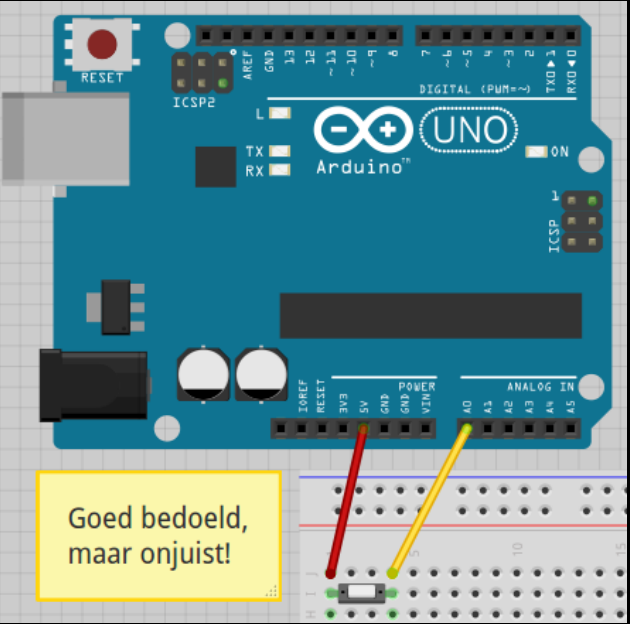
- Bouw een machine die kan reageren op een drukknop
- Gebruik je kennis van 'analogRead'
- ... dit gaat echter anders dan verwacht!

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?



0.4 Ontwerp: naief



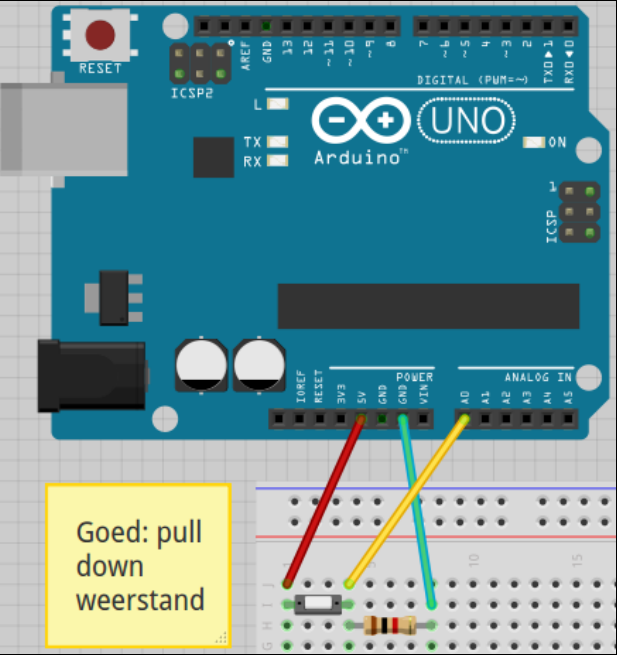
0.8 Ontwerp: oplossing

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

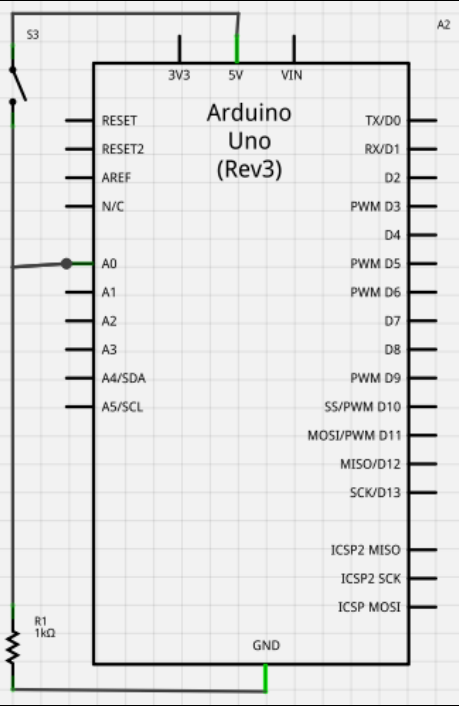
0.5 Opdracht

- 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)

0.9 Ontwerp: oplossing



0.10 Ontwerp: oplossing



10

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)

11

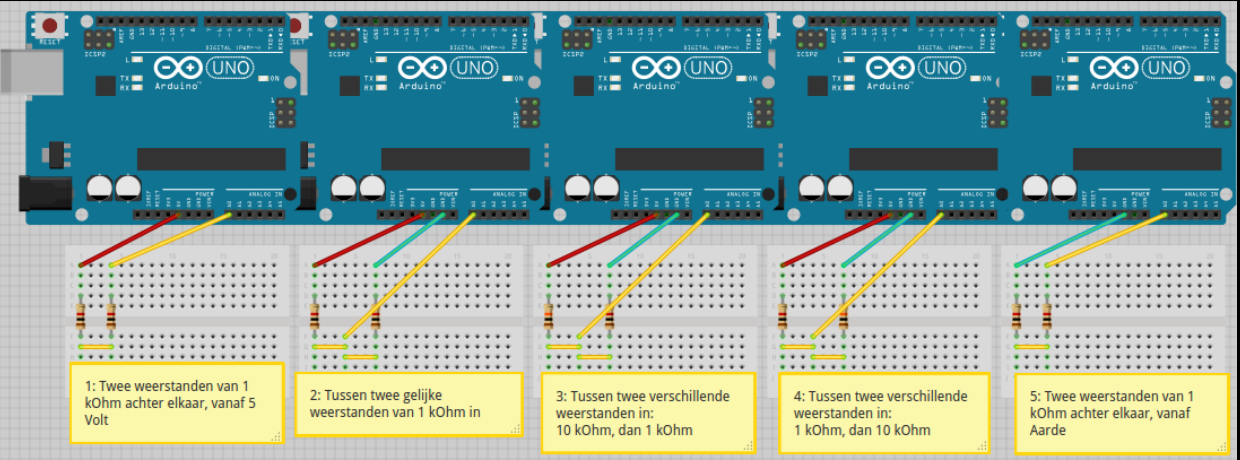
Dag 2 Oefeningen

(C) Richèl Bilderbeek

February 12, 2014

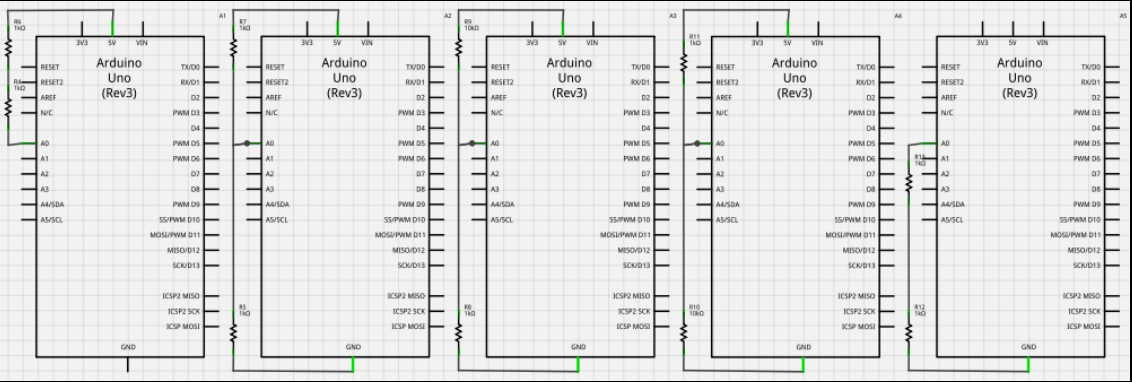
1

0.2 Basisschakelingen



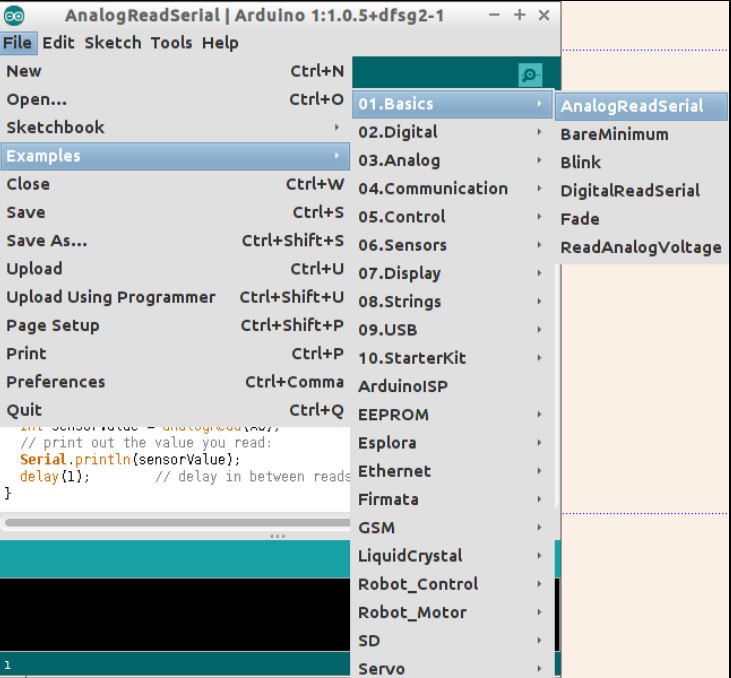
2

0.3 Basisschakelingen



3

0.4 Programma



4

0.1 Overzicht

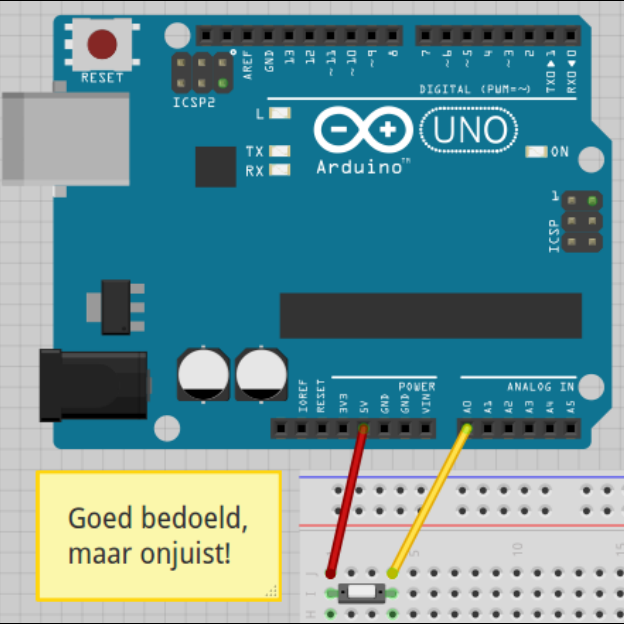
1. De vijf basisschakelingen
2. De drukknop

0.5 De Vijf Basisschakelingen

- Gebruik het programma: 'Examples | 01. Basics | AnalogReadSerial'
- Bouw omstebeurt de vijf basisschakelingen
- Welke waarden meet je bij elk van de vijf schakelingen?
- Noteer! Deze schakelingen zul je vaak gaan bekijken!

5

0.6 De Druknop: naief ontwerp



Arduino C++ #1

(C) Richèl Bilderbeek 

February 12, 2014

0.7 De Druknop

- 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!
- Daarna: krijg de schakeling juist werkend

0.1 Overzicht

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

Dag 3

(C) Richèl Bilderbeek 

February 12, 2014

0.2 Doel

- Fouten leren lezen
- Internet zoektermen
- Basis van programmeren

0.1 Vandaag

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

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)

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!

0.8 Fouten

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

0.5 Minimum programma

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

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

```
const int pin_led = 2;
```

- double: een gebroken getal, bijvoorbeeld: een voltage

```
const double voltage = 3.3; //Volt
```

- bool: een ja/nee, bijvoorbeeld: is de drukknop ingedrukt?

```
const bool is_ingedrukt = /* iets ingewikkelds */;
```

- nog veel meer

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

```
const int pin_led = 2;
```

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

```
void loop()
{
    const int wachttijd = 1000; //milliseconden
    digitalWrite(pin_led ,HIGH);
    delay(wachttijd);
    digitalWrite(pin_led ,LOW);
    delay(wachttijd);
}
```

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 snapt wat de
- Ik vind er met gemak achtien!
- Deze fouten zul je nog vaak tegen komen!

Hoe lees ik een sensor?

(C) Richèl Bilderbeek 

February 12, 2014

0.1 Overzicht

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

0.5 Doen!

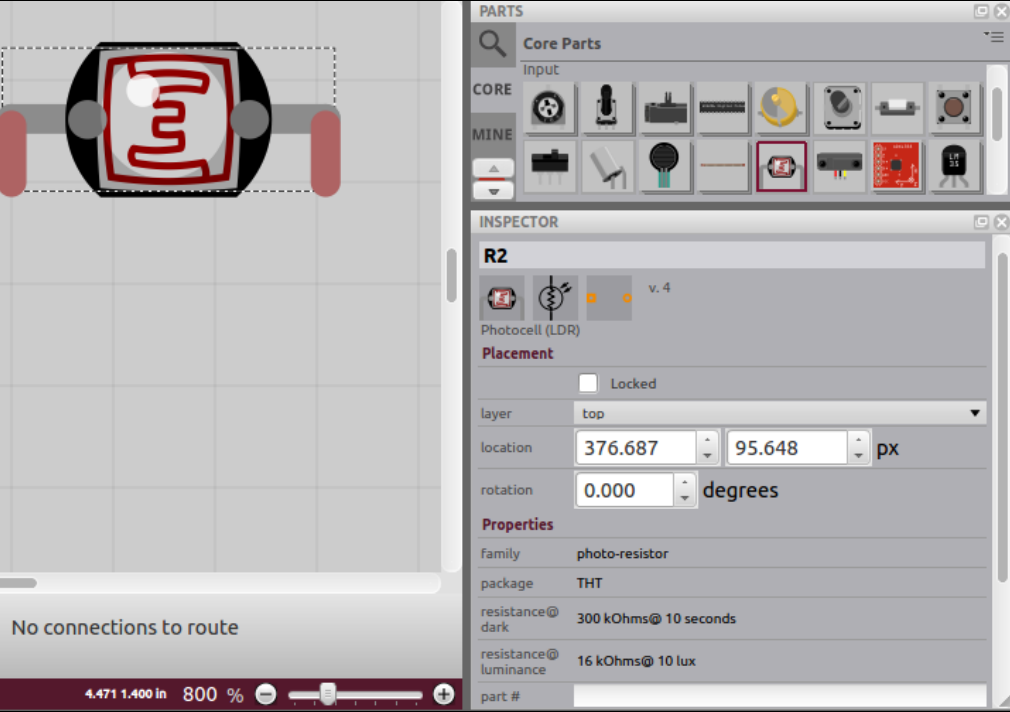
0.2 Doel

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

0.6 Conclusie

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

0.3 Waar in Fritzing?



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)

0.1 Overzicht

- 1. Doel

Hoe stuur ik een zonnemotor?

(C) Richèl Bilderbeek 

February 12, 2014

0.2 Doel

- Zelf uitvinden hoe een zonnemotor werkt

0.6 Conclusie

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

0.3 Waar in Fritzing?

- Niet

Dag 4

(C) Richèl Bilderbeek 

February 12, 2014

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

Arduino & werkwijze

(C) Richèl Bilderbeek 

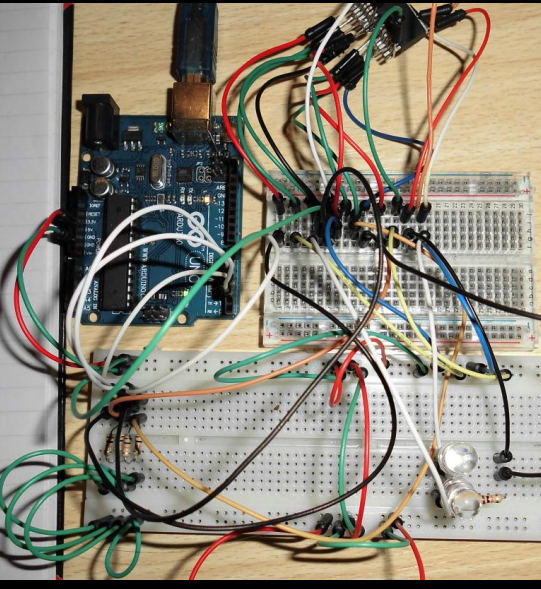
February 12, 2014

0.5 Doen!

0.1 Overzicht

1. Waarom een werkwijze?
2. Welke werkwijze?

0.2 Hoezo?



Hij doet het niet!

0.6 Wat is je programma?

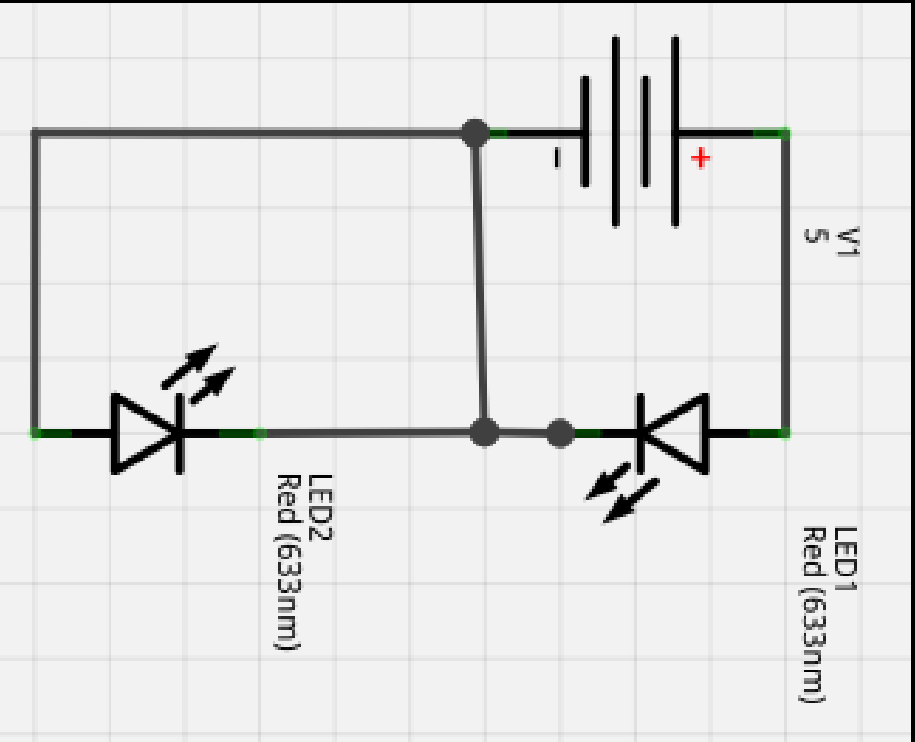
```
void setup()
{
  pinMode(2,OUTPUT);
}

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

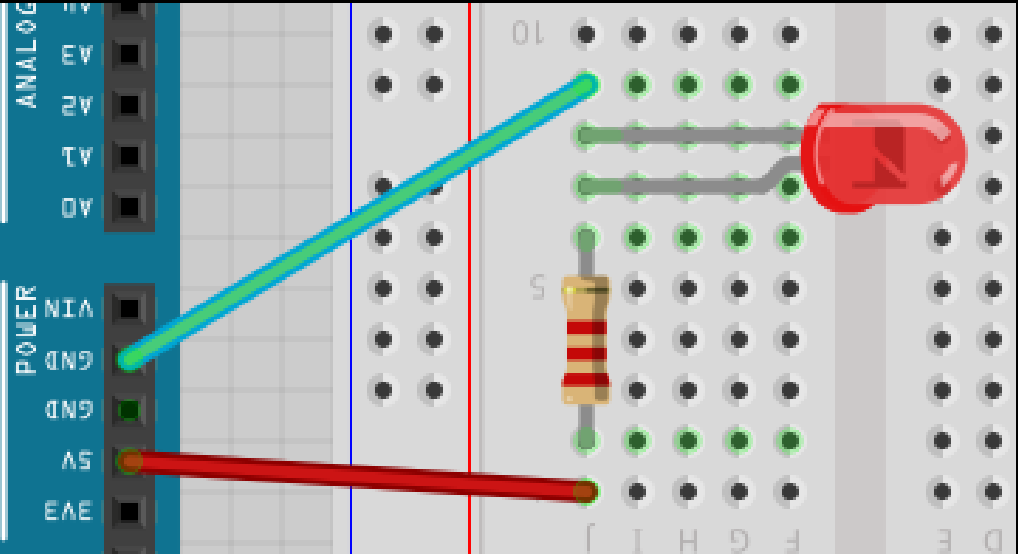
0.3 'Hij doet het niet'

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

0.4 Stroomschema



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);
}
```

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 second 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]

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

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