

Dag 1

(C) Richèl Bilderbeek 

April 19, 2014

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 

January 30, 2014

0.1 Overzicht

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

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

0.3 Wie zijn jullie?

- Wie ben jij?
- Wat wil je?

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

0.5 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 

April 19, 2014

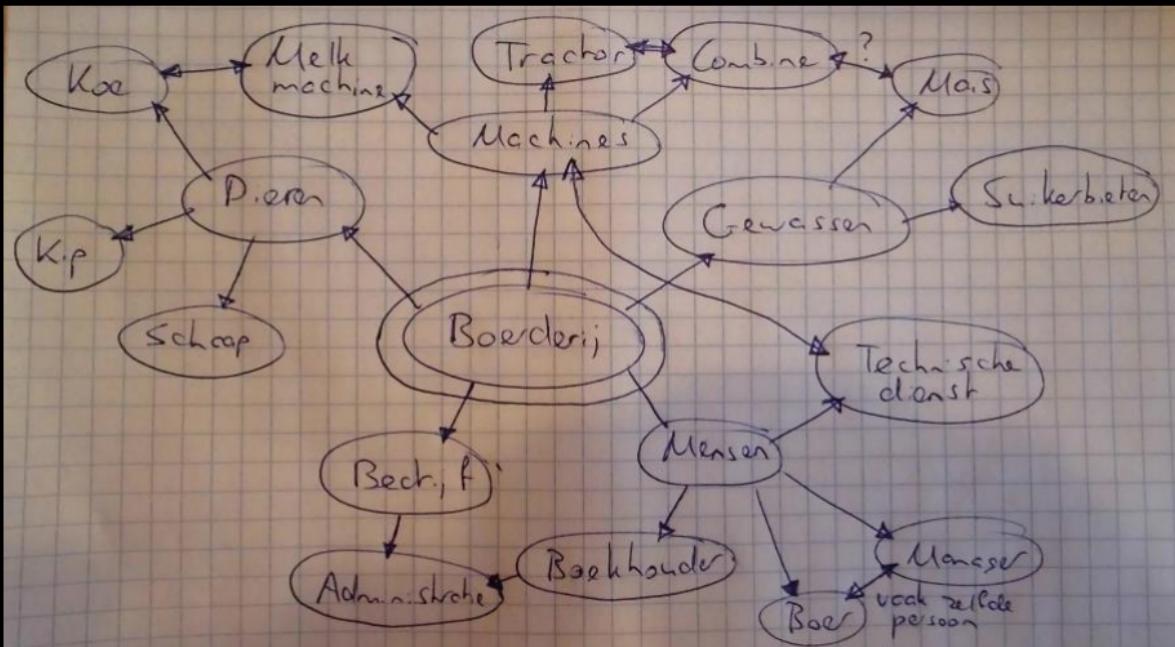
0.1 Overzicht

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

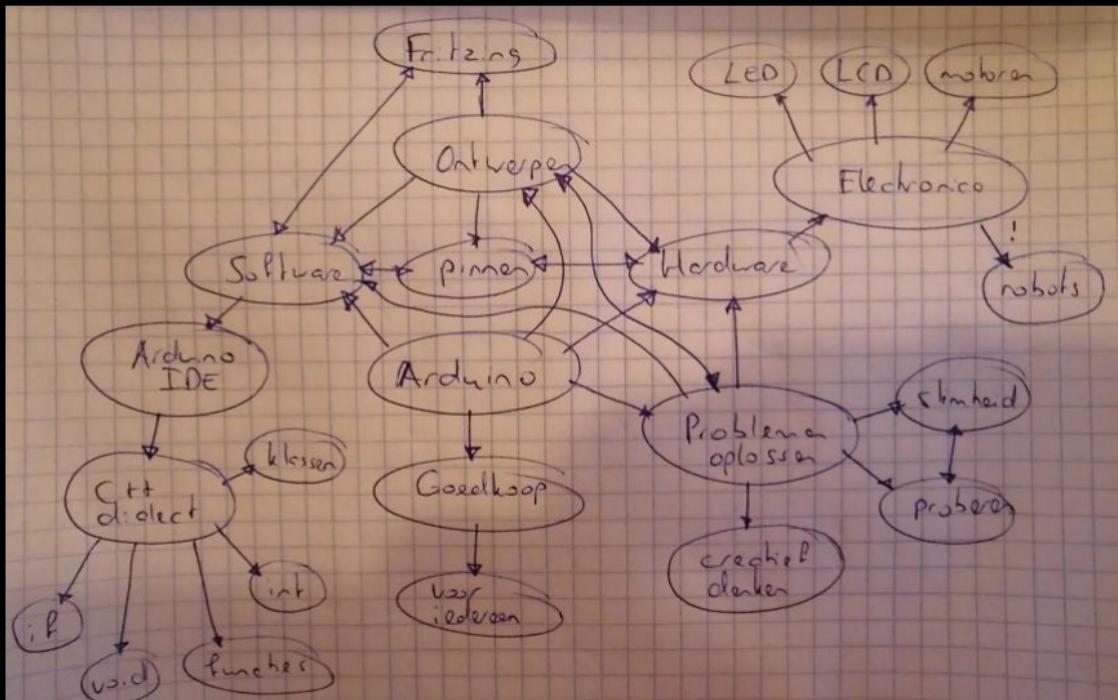
0.3 Wat is een mindmap?



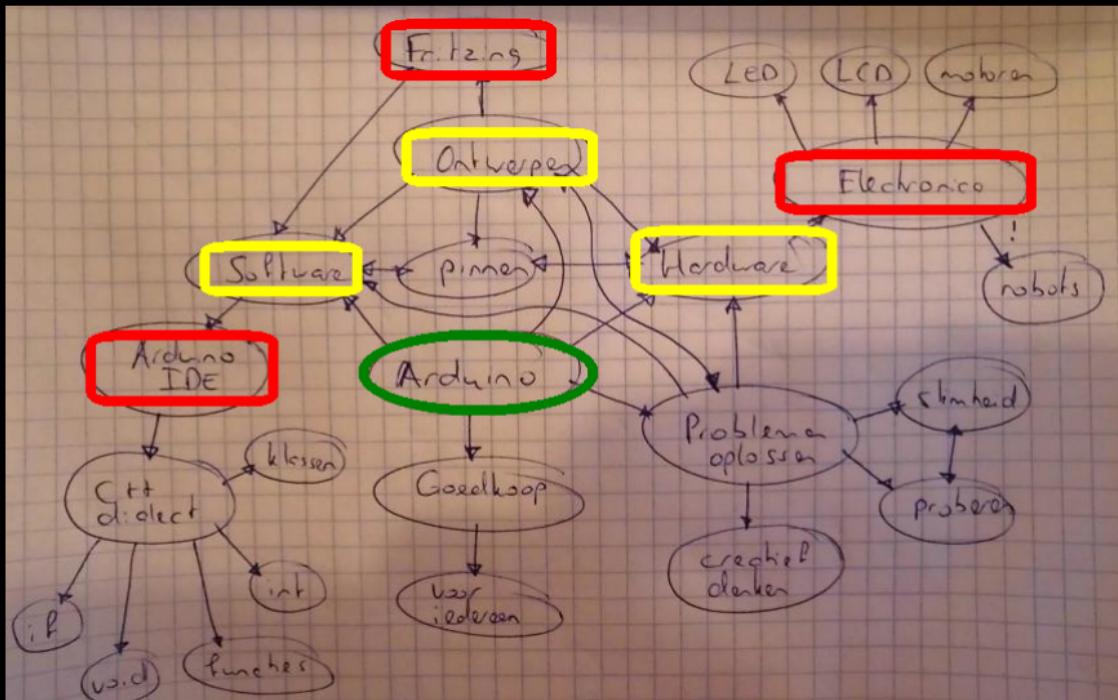
0.4 Maken mindmap

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

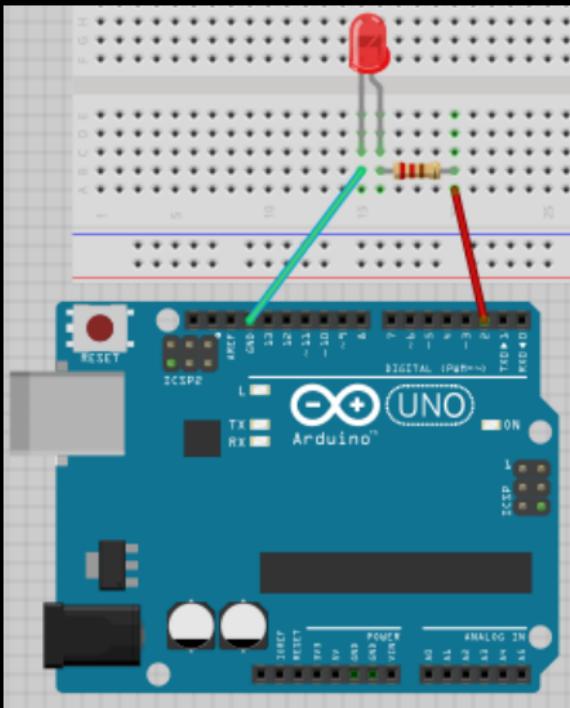
0.5 Wat is Arduino volgens Richèl?



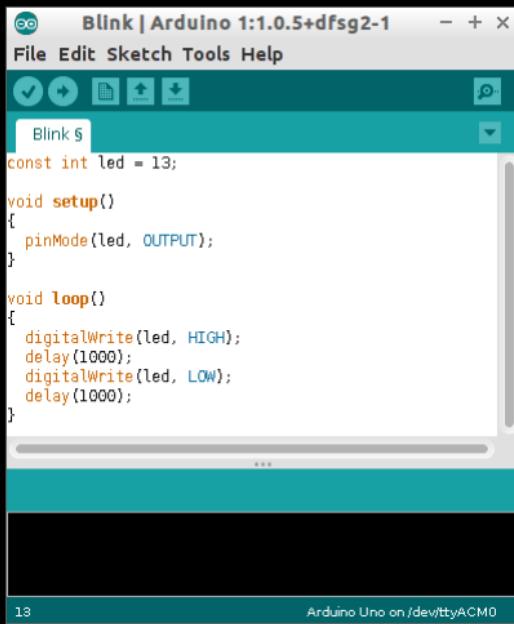
0.6 Wat is Arduino volgens Richèl?



0.7 Elektronica



0.8 Software



The screenshot shows the Arduino IDE interface with the title bar "Blink | Arduino 1:1.0.5+dfsg2-1". The menu bar includes File, Edit, Sketch, Tools, and Help. Below the menu is a toolbar with icons for file operations like Open, Save, and Upload. The main code editor window displays the "Blink" sketch:

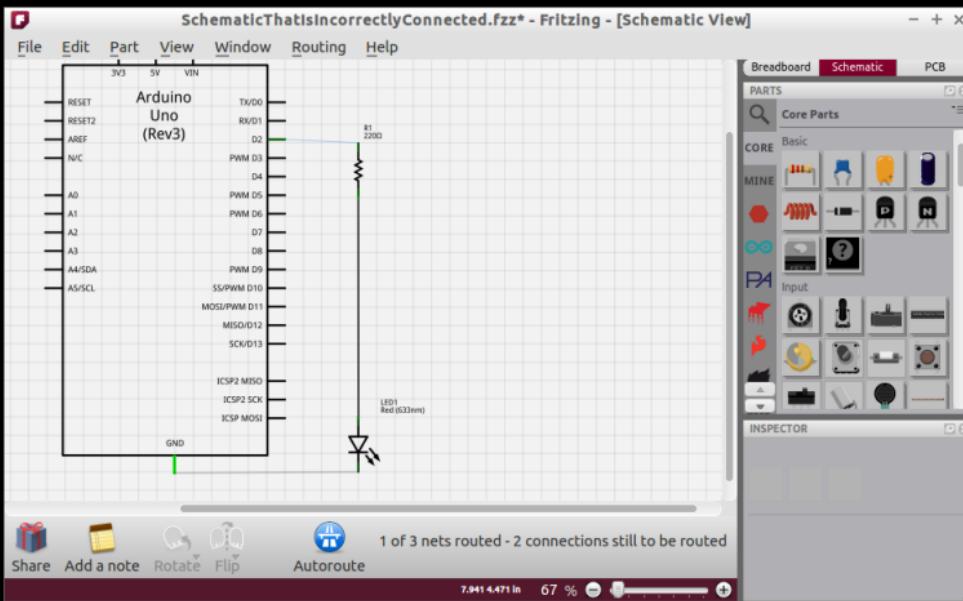
```
const int led = 13;

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

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

The status bar at the bottom shows the port "13" and the connection "Arduino Uno on /dev/ttyACM0".

0.9 Ontwerpen



0.10 Conclusie

Wat we doen is divers:

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

Wij bouwen dit stap voor stap op

Hoe sluit ik een LED aan?

(C) Richèl Bilderbeek 

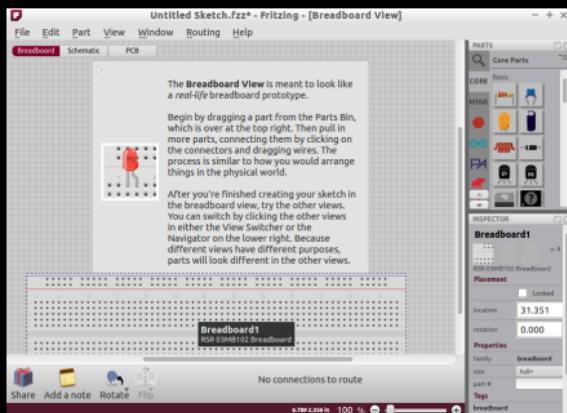
April 19, 2014

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

0.2 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

0.4 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?

0.5 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?

0.6 Wat weten we?

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

0.7 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?

0.8 Antwoorden

The screenshot shows a web browser window with the URL www.okaphone.com/artikel.asp?id=422437. The page content is as follows:

LED diffuus

- Kleur (licht) **rood**
- Lichtsterkte **20 ... 100 mcd**
- Hoek **50 °**
- Golvengte **625 nm**
- Diameter **10 mm**
- Spanningsval **2,5 V**
- Stroom **5 ... 20 mA**

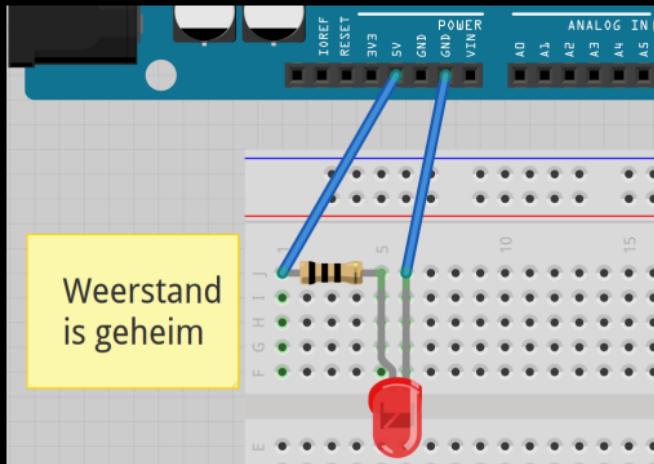
Materiaal: GaAsP/GaP

A photograph of a red, diffused LED is shown on the right side of the page.

Wat kun hieraan doen?

0.9 Hoe hoge spanningen tegen te gaan?

- Met een weerstand!



0.10 Welke weerstand?

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

The screenshot shows a web-based LED calculator interface. On the left, there's a sidebar with links for news, a shop, service, color codes, calculators, cables, cameras, and links. Below that is a contact form and a search bar. A banner at the bottom says "14 bestellers (++D)".

Voeding

Spanning	5	V
Maximale stroom	1	A

Schakeling

1 2 3 type(n) LEDs

LED

€0,30-LED diffusa, rood	x D1 L2060ID	
rood, 12,5...32mcd 70° 625nm 1,8mm 1,8V 5...20mA		
Spanningsval	1,8	V
Stroom	20	mA
Vermogen	36	mW

Weerstand

1 × R1	160	Ω
Spanning	3,2	V
Stroom	20	mA
Vermogen	64	mW

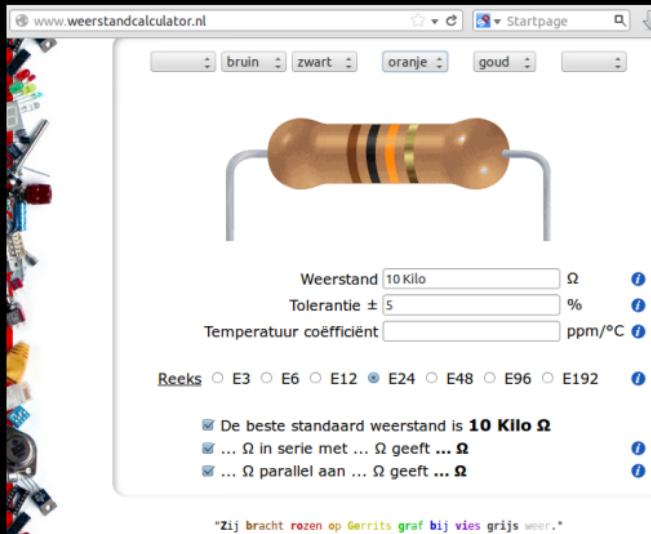
Verbruik

Stroom	0,02	A
Vermogen	0,1	W

Below the table is a schematic diagram of a simple circuit: a battery (5V) connected in series with a diode (D1) and a resistor (R1). The diode is labeled with its symbol and "D1". The resistor is labeled "R1". The circuit is completed by connecting the other end of the resistor back to the negative terminal of the battery.

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 

April 19, 2014

0.1 Overzicht

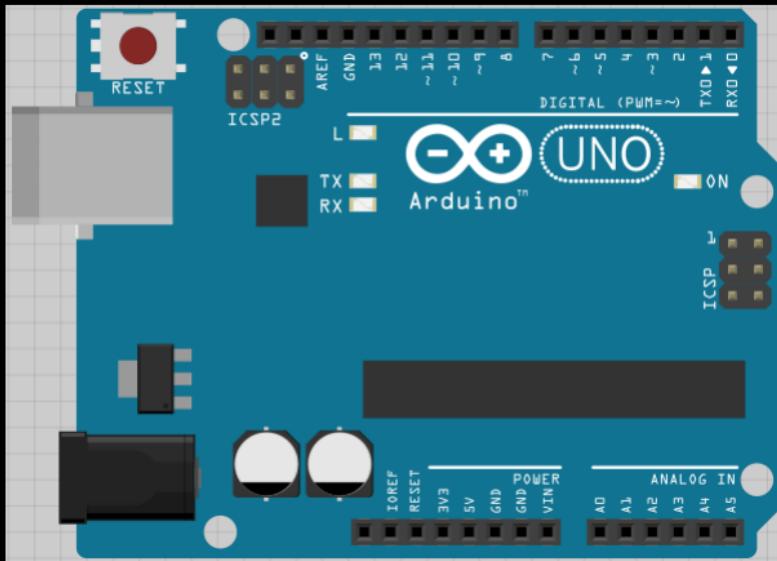
1. Doel
2. Hoe ontwerpen?
3. Arduino IDE
4. Bouwen!

0.2 Doel

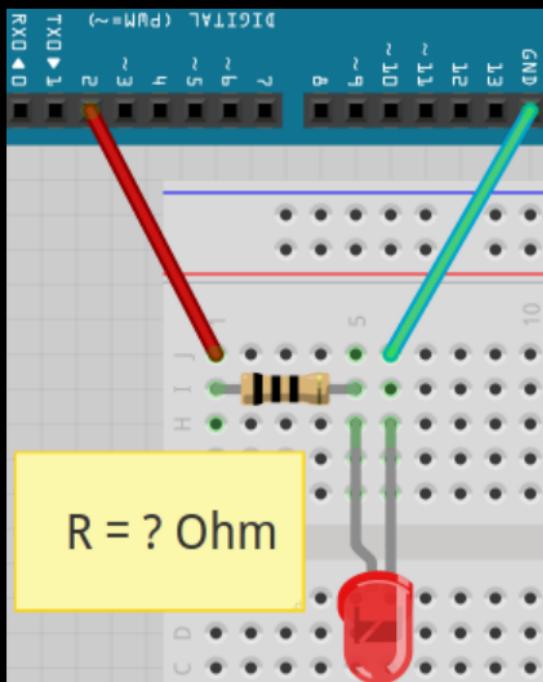
Een machine maken die een LED na een seconde aan doet, en deze een seconde later weer uit doet

0.3 Hoe ontwerpen?

- Waar denk je dat de Arduino dit mee kan?



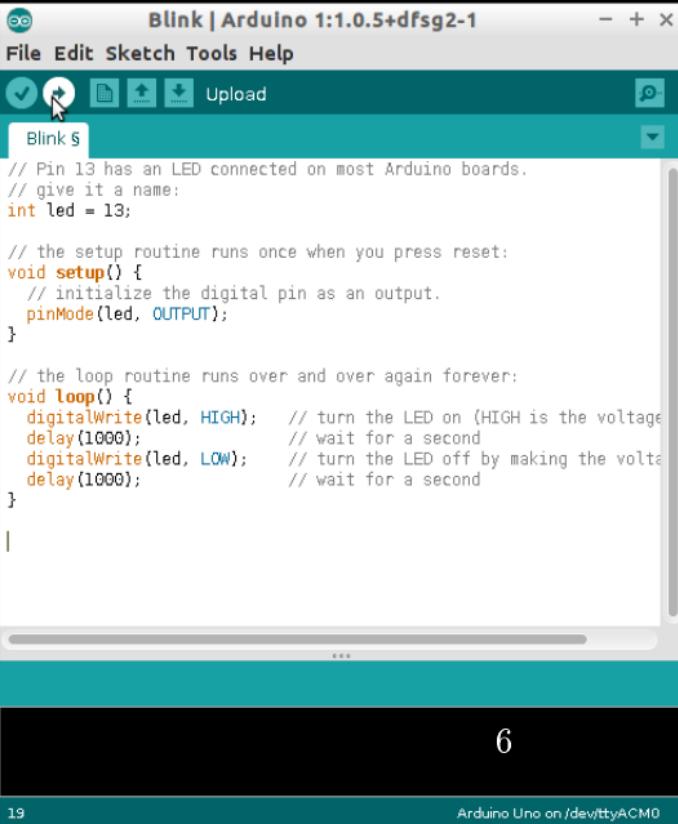
0.4 Ontwerp



0.5 En nu?

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

0.6 Arduino IDE



The screenshot shows the Arduino IDE interface with the title bar "Blink | Arduino 1:1.0.5+dfsg2-1". The menu bar includes File, Edit, Sketch, Tools, and Help. The toolbar has icons for Open, Save, Upload, and others. A dropdown menu is open over the "Upload" button. The code editor contains the "Blink" sketch:

```
// Pin 13 has an LED connected on most Arduino boards.  
// give it a name:  
int led = 13;  
  
// the setup routine runs once when you press reset:  
void setup() {  
  // initialize the digital pin as an output.  
  pinMode(led, OUTPUT);  
}  
  
// the loop routine runs over and over again forever:  
void loop() {  
  digitalWrite(led, HIGH);    // turn the LED on {HIGH is the voltage  
  delay(1000);              // wait for a second  
  digitalWrite(led, LOW);    // turn the LED off by making the voltage  
  delay(1000);              // wait for a second  
}
```

The status bar at the bottom shows "19" and "Arduino Uno on /dev/ttyACM0".

0.7 Arduino IDE installatie

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

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

Dag 1 Oefeningen

(C) Richèl Bilderbeek 

April 19, 2014

0.1 Overzicht

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

0.2 Fritzing installatie

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

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 

April 19, 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 

April 19, 2014

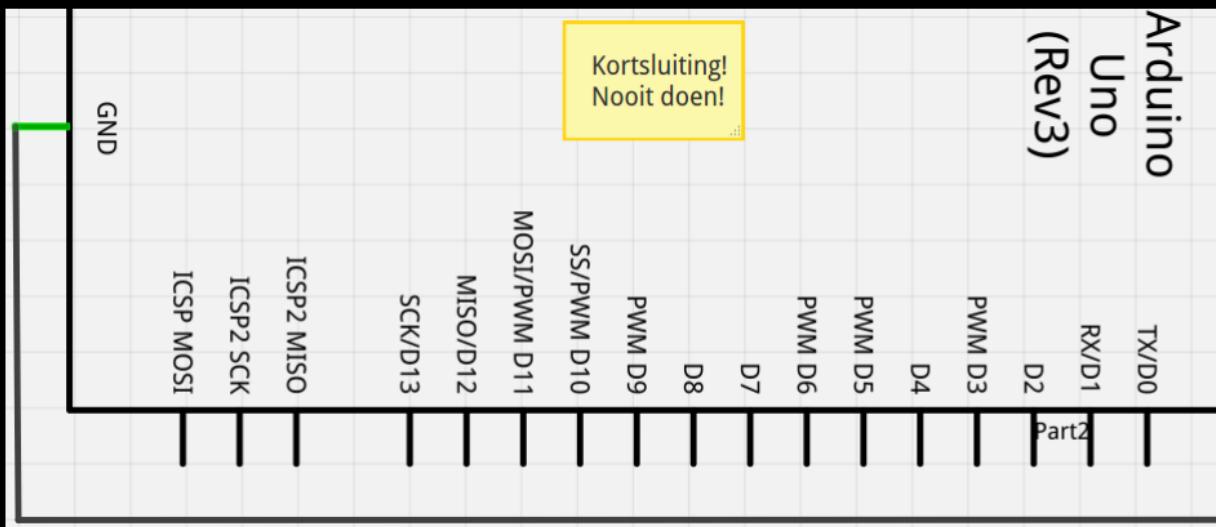
0.1 Overzicht

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

0.2 Wat is kortsluiting?



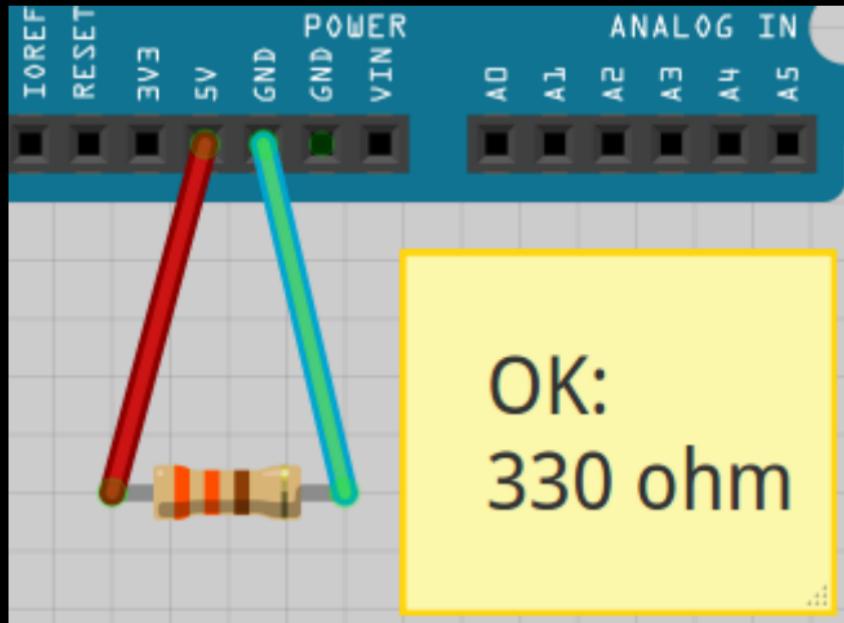
0.3 Wat is kortsluiting?



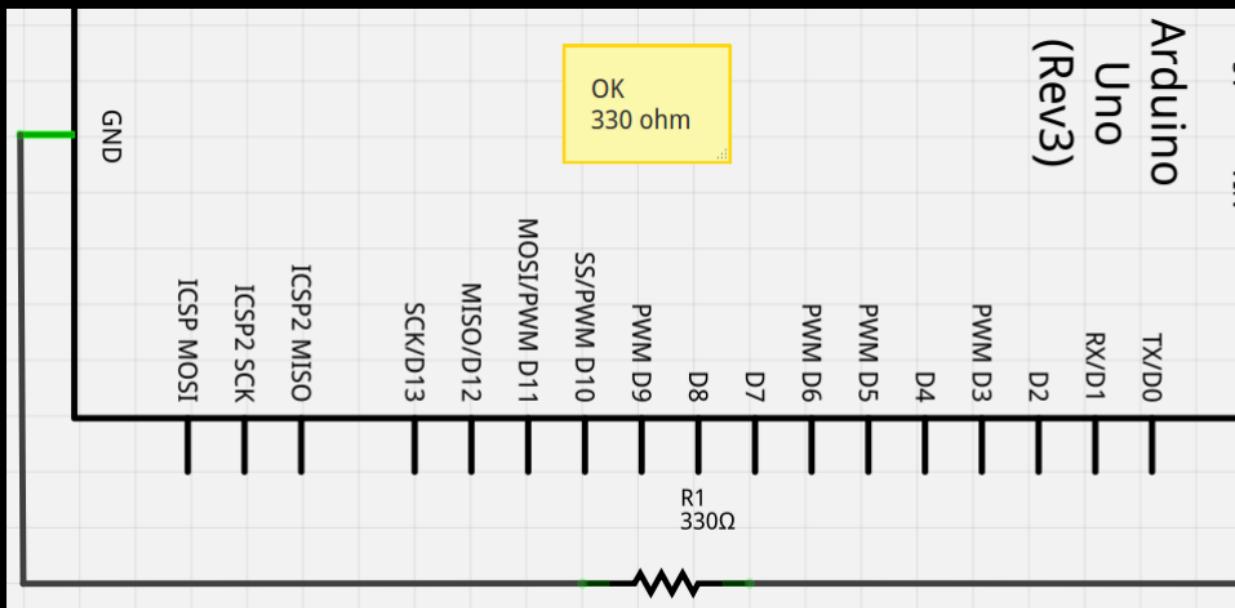
0.4 Hoe dit te voorkomen?

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

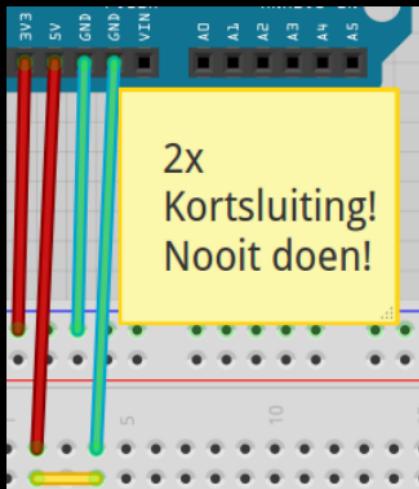
0.5 Wat is kortsluiting?



0.6 Wat is kortsluiting?



0.7 Breadboard



0.8 Breadboard



analogRead

(C) Richèl Bilderbeek 

April 19, 2014

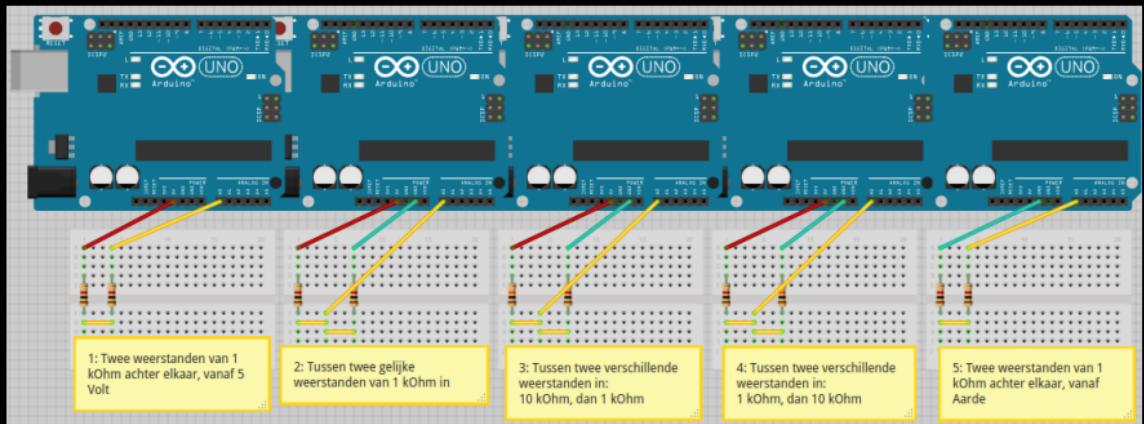
0.1 Overzicht

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

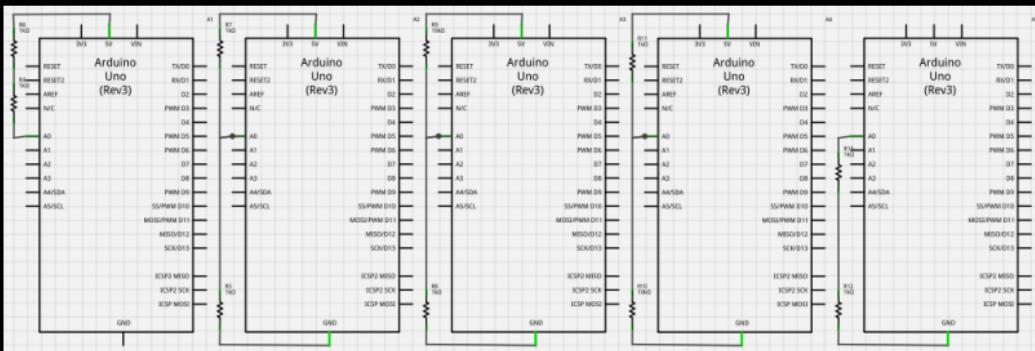
0.2 Doel

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

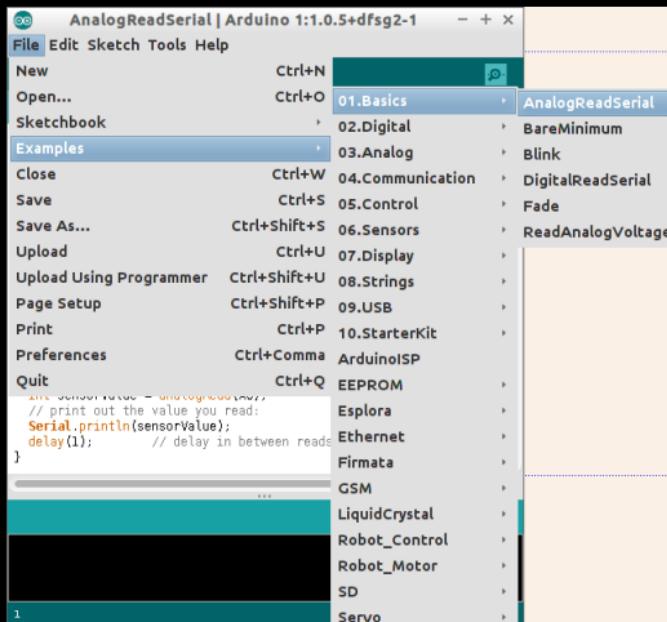
0.3 Opdracht



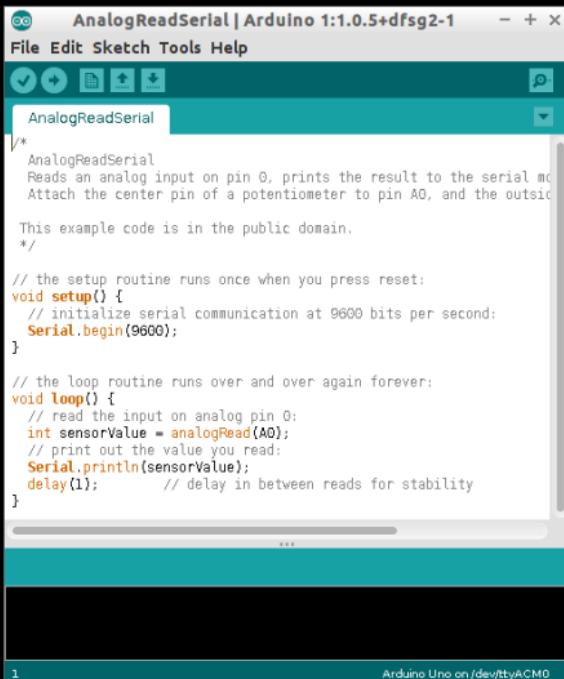
0.4 Opdracht



0.5 Programma



0.6 Programma



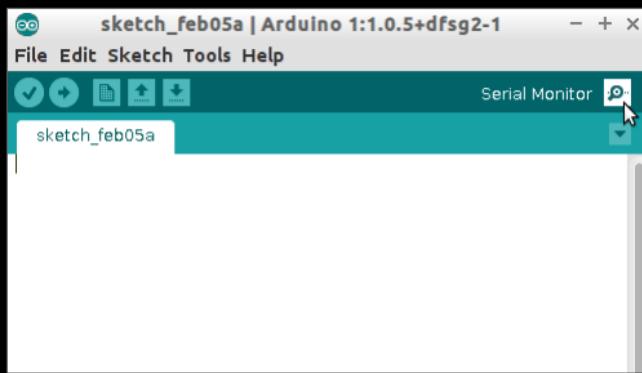
The screenshot shows the Arduino IDE interface with the title bar "AnalogReadSerial | Arduino 1:1.0.5+dfsg2-1". The menu bar includes File, Edit, Sketch, Tools, and Help. The toolbar has icons for Open, Save, Print, and others. The code editor window contains the "AnalogReadSerial" sketch. The code reads an analog input from pin A0 and prints it to the serial monitor. It includes a setup routine for serial communication at 9600 bits per second and a loop routine that reads the analog value, prints it, and adds a delay of 1 millisecond.

```
/*
 * 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
}
```

0.7 Bekijken



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!

Hoe lees ik input?

(C) Richèl Bilderbeek 

April 19, 2014

0.1 Overzicht

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

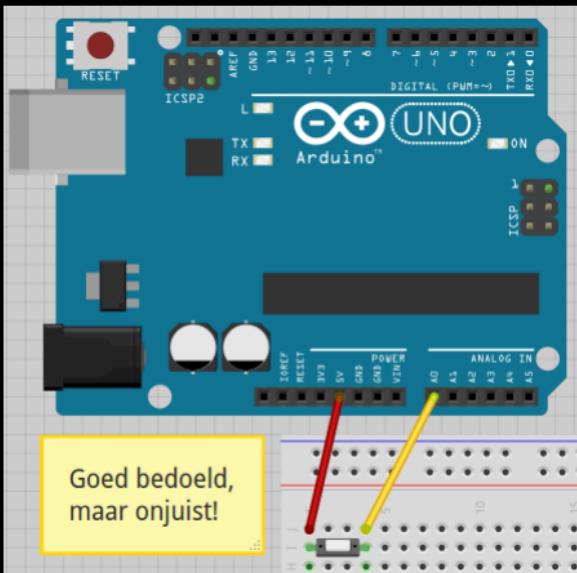
0.2 Doel

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

0.3 Vraag

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

0.4 Ontwerp: naief



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)

0.6 Doen...

- Volgende slide de oplossing...

0.7 Waarom onjuist?

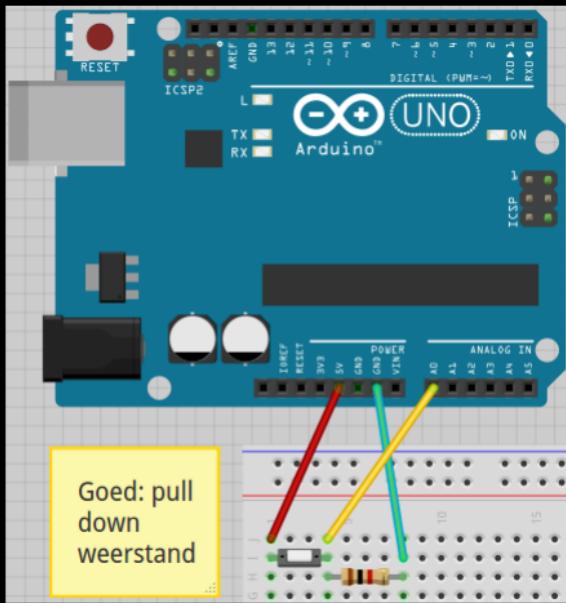
- Als de drukknop open is, is de spanning op de input onbepaalt: dit kan elke waarde tussen nul en vijf Volt zijn!
- Dit wordt een zwevende input genoemd
- Hoe dit op te lossen?



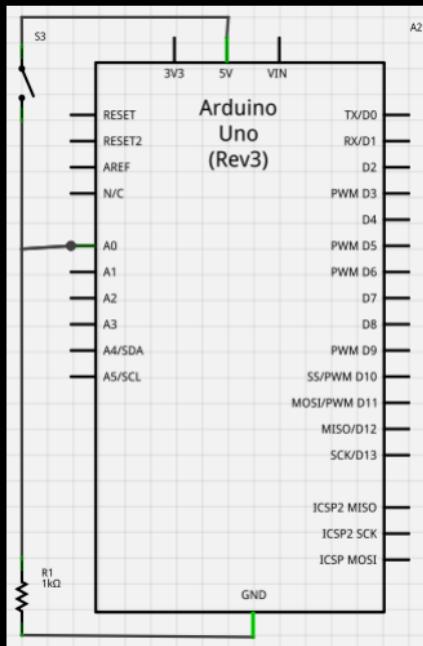
0.8 Ontwerp: oplossing

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

0.9 Ontwerp: oplossing



0.10 Ontwerp: oplossing



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)

Dag 2 Oefeningen

(C) Richèl Bilderbeek

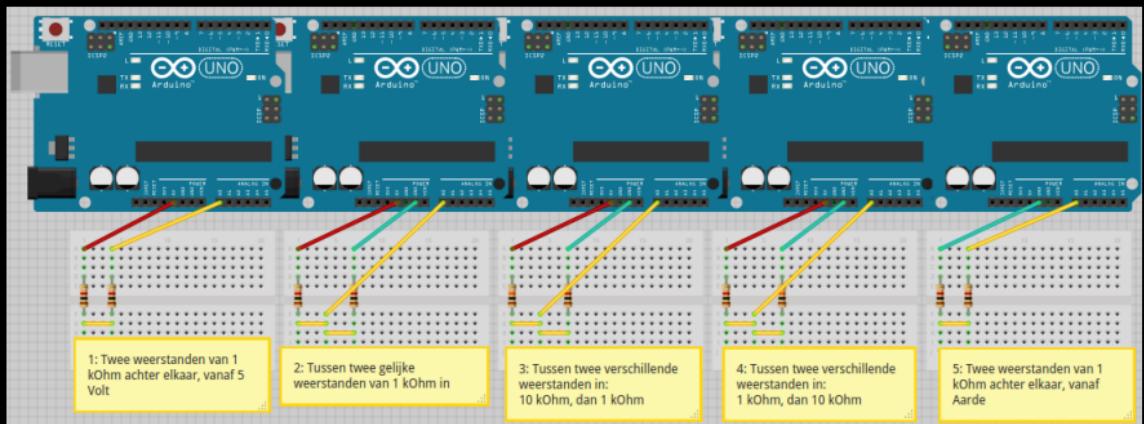


February 13, 2014

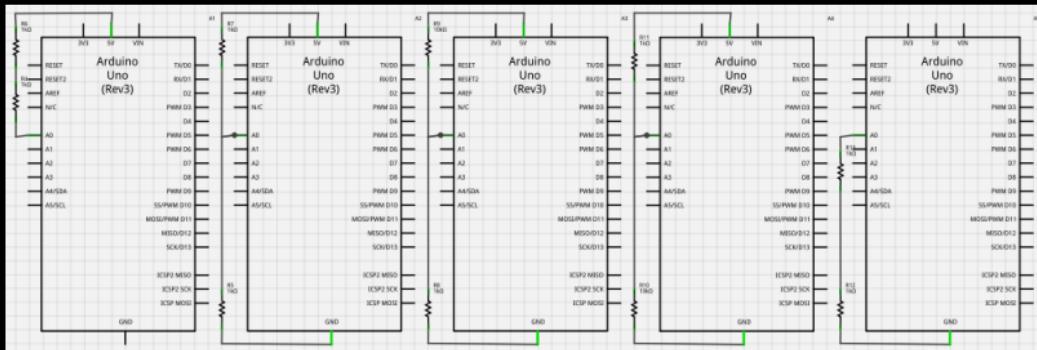
0.1 Overzicht

1. De vijf basisschakelingen
2. De drukknop

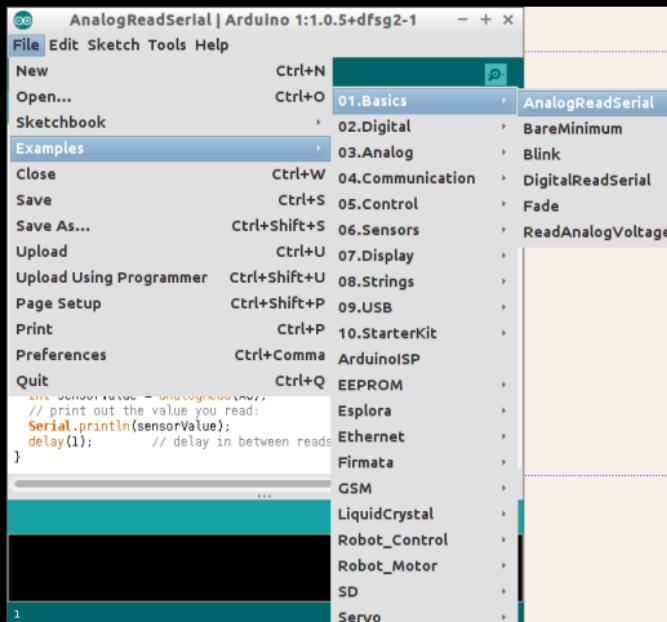
0.2 Basisschakelingen



0.3 Basisschakelingen



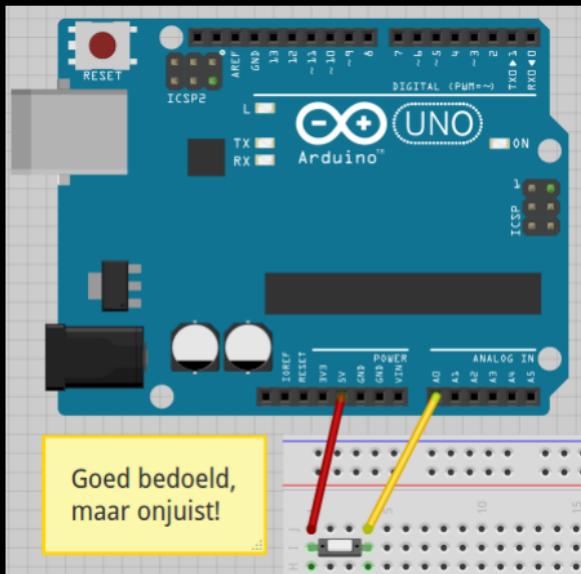
0.4 Programma



0.5 De Vijf Basisschakelingen

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

0.6 De Drukknop: naief ontwerp



0.7 De Drukknop

- 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

Dag 3

(C) Richèl Bilderbeek 

April 19, 2014

0.1 Vandaag

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

Arduino C++ #1

(C) Richèl Bilderbeek 

April 19, 2014

0.1 Overzicht

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

0.2 Doel

- Fouten leren lezen
- Internet zoektermen
- Basis van programmeren

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!

0.5 Minimum programma

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

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

0.8 Fouten

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

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

0.10 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;
```

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

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

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

Hoe lees ik een sensor?

(C) Richèl Bilderbeek



April 19, 2014

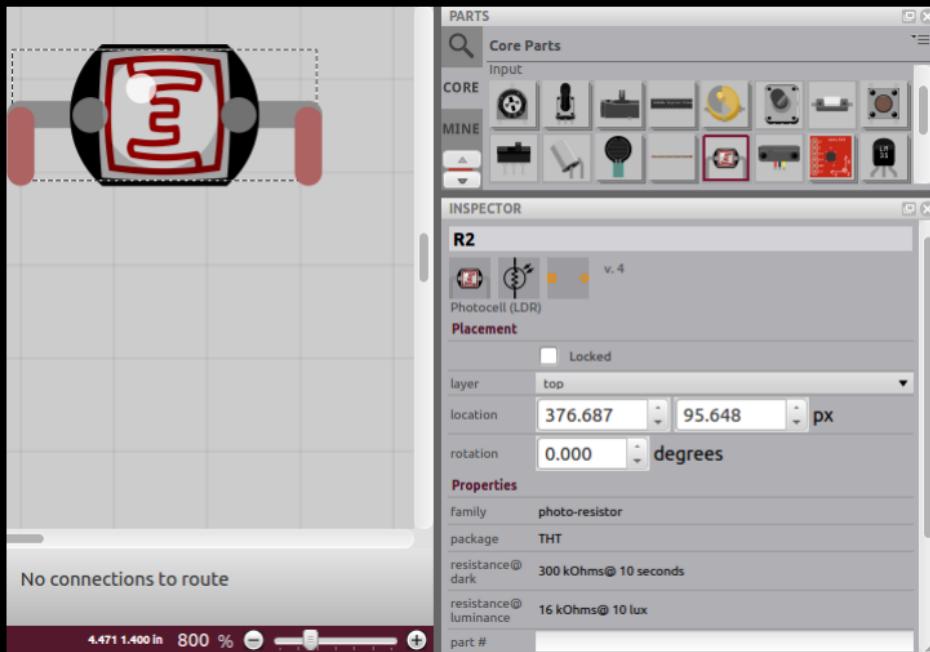
0.1 Overzicht

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

0.2 Doel

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

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)

0.5 Doen!

0.6 Conclusie

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

Hoe stuur ik een zonnemotor?

(C) Richèl Bilderbeek 

April 19, 2014

0.1 Overzicht

1. Doel

0.2 Doel

- Zelf uitvinden hoe een zonnemotor werkt

0.3 Waar in Fritzing?

- Niet

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

0.5 Doen!

0.6 Conclusie

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

Dag 4

(C) Richèl Bilderbeek 

April 19, 2014

Hoe sluit ik een gelijkstroom motor aan?

(C) Richèl Bilderbeek



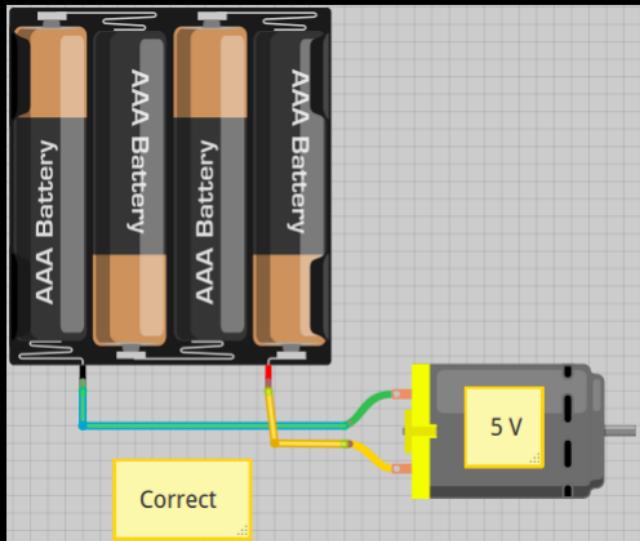
April 19, 2014

0.1 Overzicht

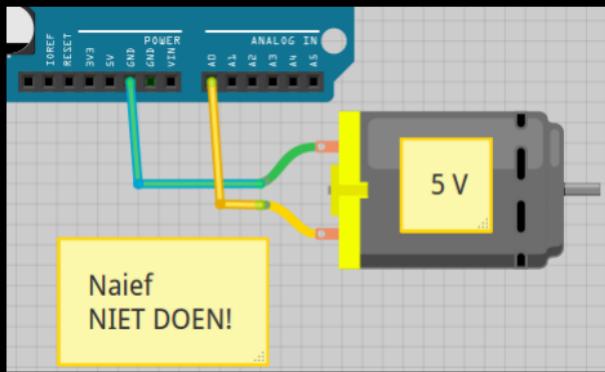
1. Doel
2. Er is een probleem? 40 mA per pin
3. Wat weten we?
4. Er is een probleem?
5. Hoe gaan we daar mee om?
6. Opdracht: vind dit uit! Noteer in je logboek

0.2 Doel

- Sluit een gelijkstroommotor aan op een Arduino



0.3 Naieve oplossing



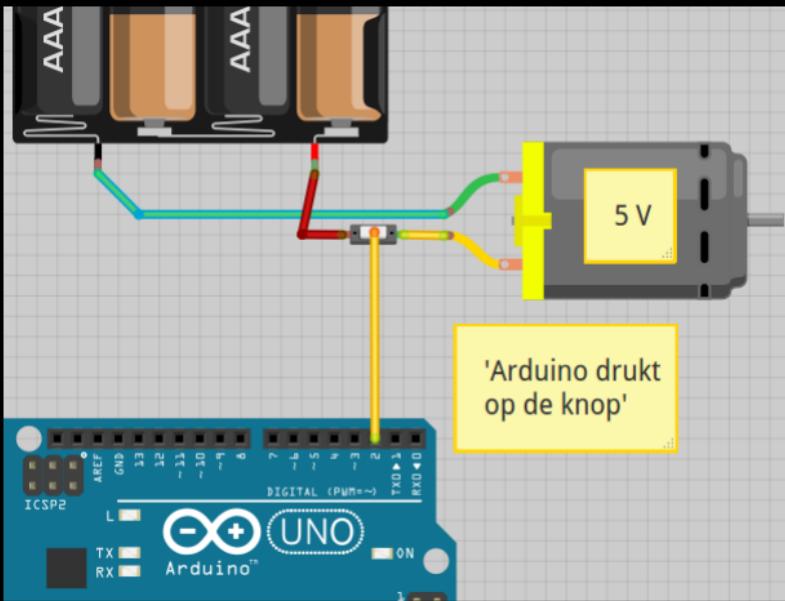
0.4 Wat is het probleem?

- Een Arduino kan niet veel¹ stroom leveren
- Een Arduino is een *micro-controller*
- Iets dat veel vermogen² nodig heeft, moet indirect aangestuurd worden

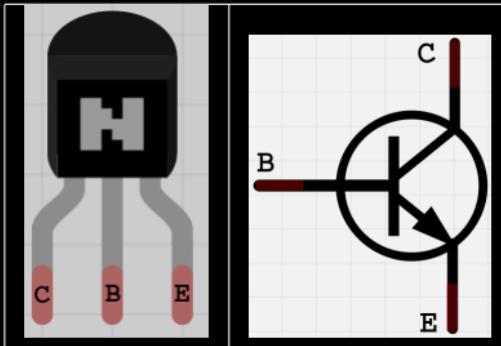
¹40 mA per pin

²Vermogen = Spanning x stroom

0.5 Goed idee

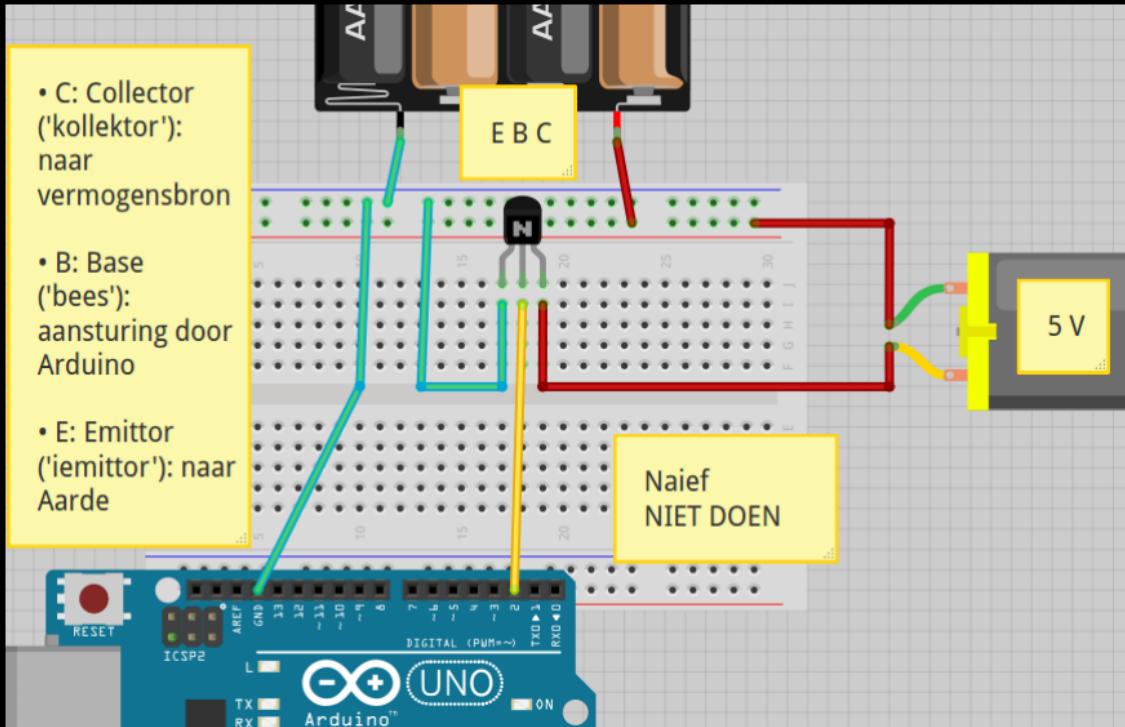


0.6 NPN transistor

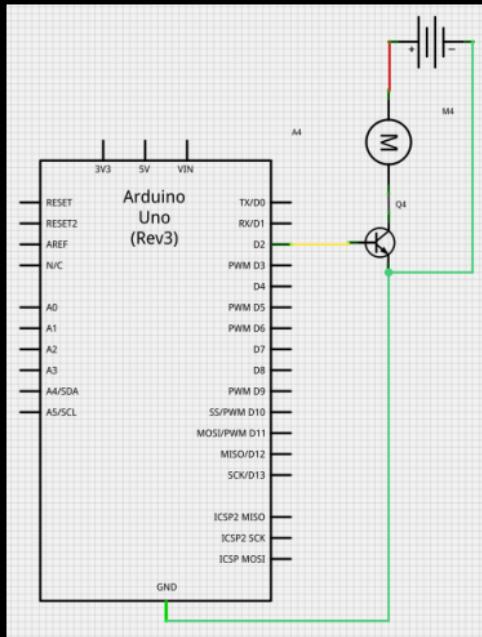


- C: Collector ('kollektor'): naar vermogensbron
- B: Base ('bees'): aansturing door Arduino
- E: Emitter ('iemittor'): naar Aarde

0.7 Beter



0.8 Beter



0.9 Beter

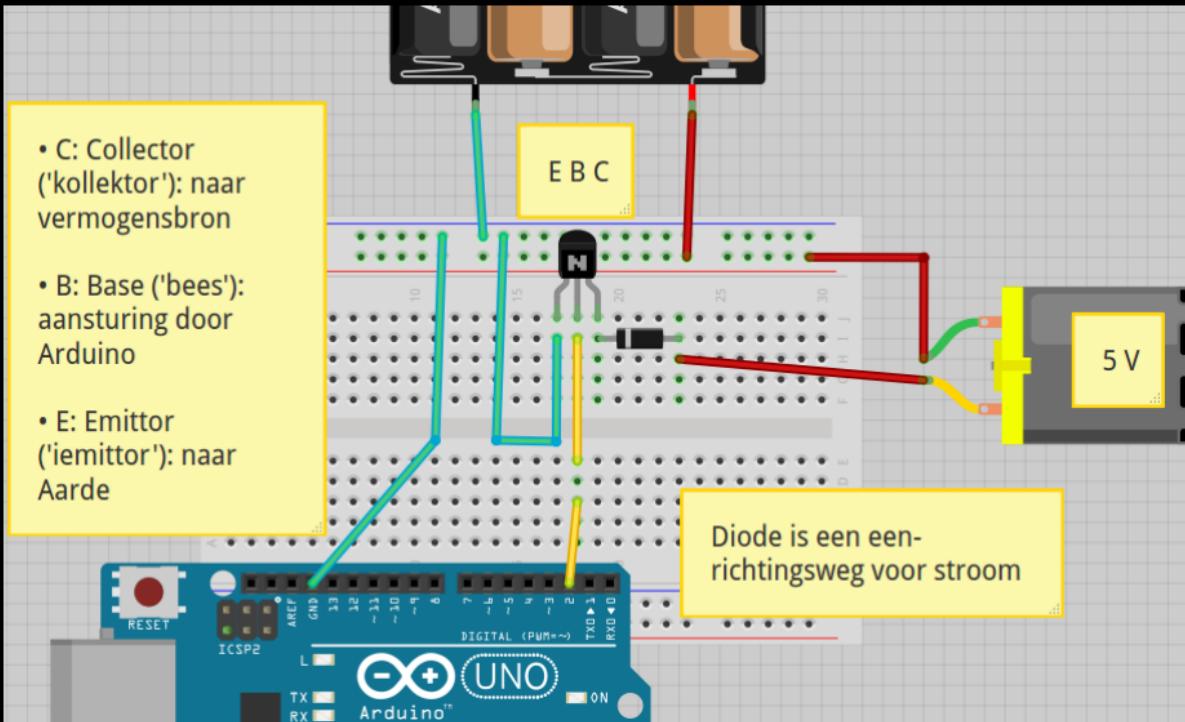
- Een motor kan een terugslag hebben, waardoor er stroom de andere kant op kan gaan lopen
- Dit kan de Arduino beschadigen!
- Er is een onderdeel waardoor stroom maar een kant op kan lopen...

0.10 Diode

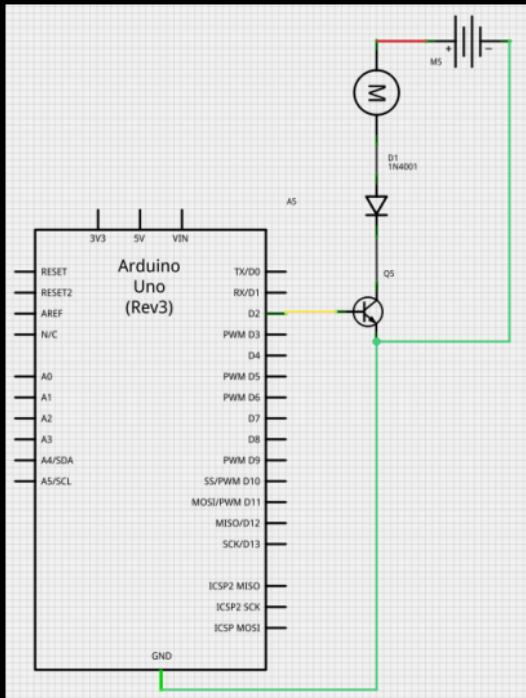


- Stroom loopt van plus naar min
- Bijvoorbeeld van 5V naar GND

0.11 Goed



0.12 Goed



Project: kleurensensor

(C) Richèl Bilderbeek 

March 21, 2014

1 Introductie

Een lichtsensor¹ 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 naief 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.

¹ook wel 'licht-afhankelijk weerstand' of 'LDR' ('Light Dependent Resistance') genoemd



Figure 1: Aansluiten RGB LED op een naieve manier

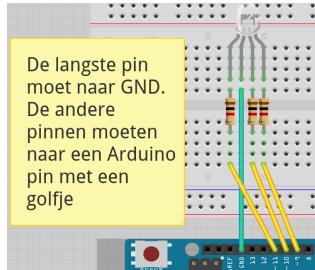


Figure 2: Aansluiten RGB LED op een slimme manier

Om een kleur van een RGB lampje aan te sturen kun je gebruik maken van het voorbeeld Fade².

Het laten 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 sterkte gaan branden.

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

Algoritme 1 is een mogelijke oplossing.

3 Aansluiten lichtsensor

Als je naief bent, sluit je een lichtsensor aan als in figuur 3. Om de waarde van een lichtsensor te lezen, kun je het voorbeeldprogramma AnalogReadSerial³ gebruiken.

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

```
int mijn_sensor_waarde = analogRead(A0);
```

Hierin wordt de waarde van pin A0 gelezen en opgeslagen in de variabele 'mijn_sensor_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

²File -> Examples -> 01. Basics -> Fade

³File -> Examples -> 01. Basics -> AnalogReadSerial

Algorithm 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 groenheid,
    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 naief aangesloten

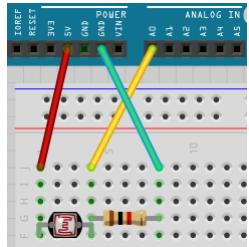


Figure 4: Lichtsensor juist aangesloten

4 LED laten reageren op sensor

Stel je voor, de sensor meet waarden van 400 voor donker en 600 voor licht. Je weet dat een LED als waarde maximaal 255 kan krijgen. Een naieve manier om de LED aan te sturen is als vol:

```
const int gemeten_waarde = analogRead(sensor_pin);
analogWrite(led_pin, gemeten_waarde / 3);
```

- Wat is hiervan het nadeel? Tip: gaat het LEDje ooit volledig branden; gaat het LEDje ooit volledig uit?

Een slimmere manier is de 'map' functie te gebruiken:

```
const int gemeten_waarde = analogRead(sensor_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 sensorwaarde. Laat deze bijvoorbeeld wit schijnen als er veel licht is.

5 Omprogrammeren naar kleurensensor

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

Arduino & werkwijze

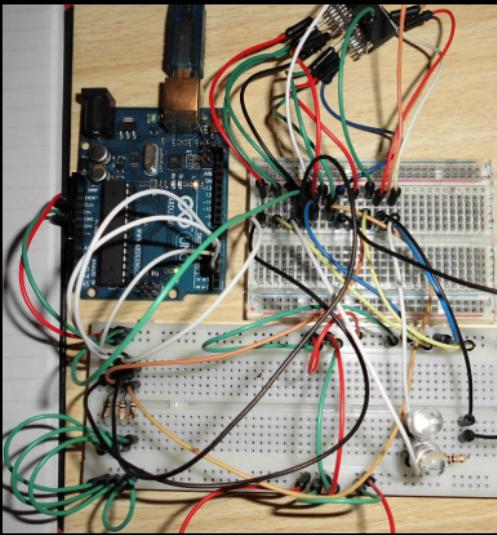
(C) Richèl Bilderbeek 

April 19, 2014

0.1 Overzicht

1. Waarom een werkwijze?
2. Welke werkwijze?

0.2 Hoezo?

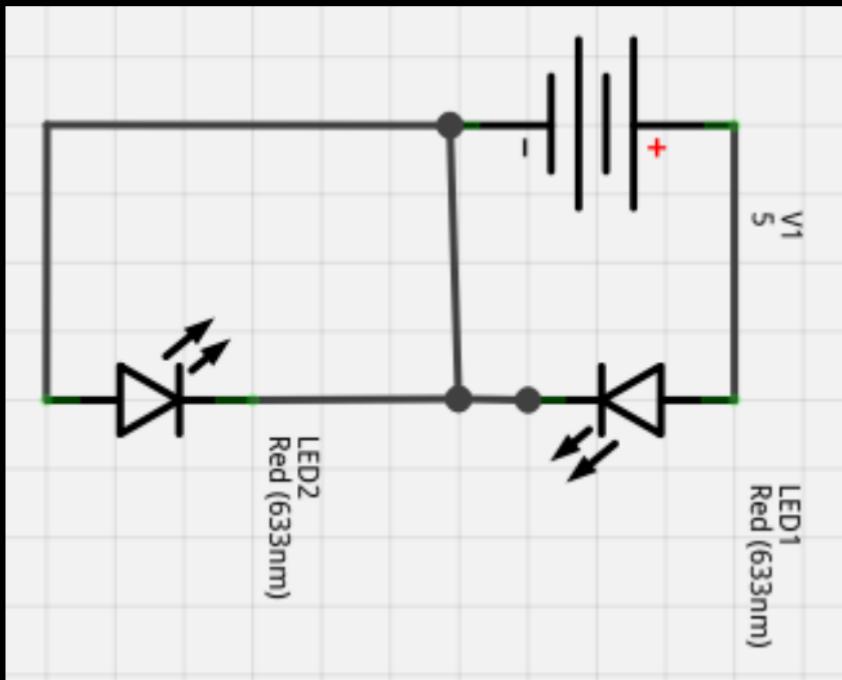


Hij doet het niet!

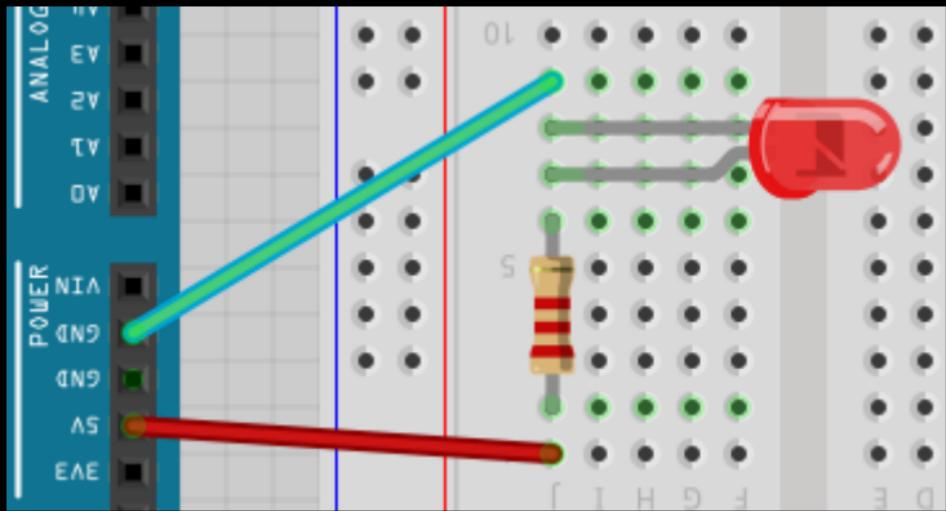
0.3 'Hij doet het niet'

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

0.4 Stroomschema



0.5 Stroomschema



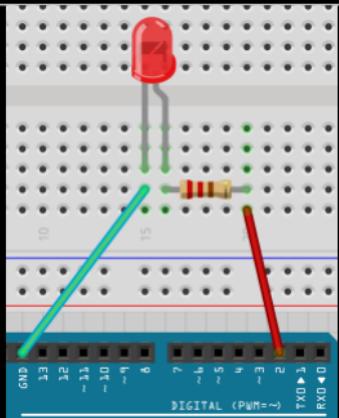
0.6 Wat is je programma?

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

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

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