



Arduino4Kids





Wat is Arduino?

- Open jullie doos...
- Hierin vind je...





Wat is Arduino?

- Er kan één programma op.
- Opladen van een programma gaat via een USB kabel.





Wat kan je ermee?

- Arduino krijgt stroom via een batterij of USB.
- Op alle poorten kan hij stroom uitsturen en ontvangen.





LED's

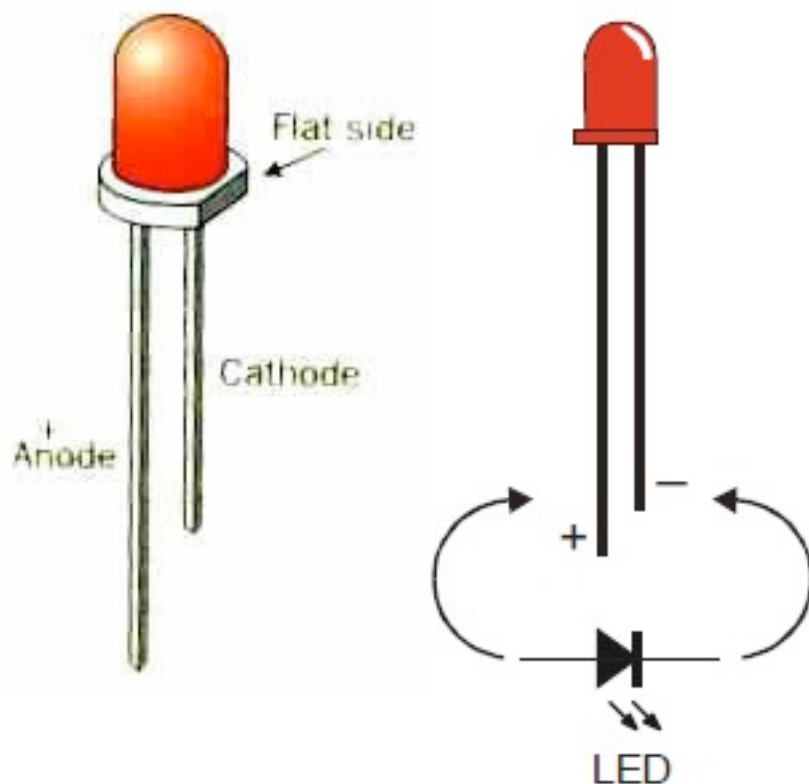
- Een LED is een klein lampje, beschikbaar in verschillende kleuren.
- Zoek een LED lampje in jullie doos...





LED's

- Elke LED heeft een kort en een lang pootje
- Elke LED heeft een 'afgeplat' kantje.



- Lange poot = + = Anode
- Korte poot = - = Kathode

KNAP = **K**athode **N**egatief **A**node **P**ositief

Stroom gaat van + naar - !?

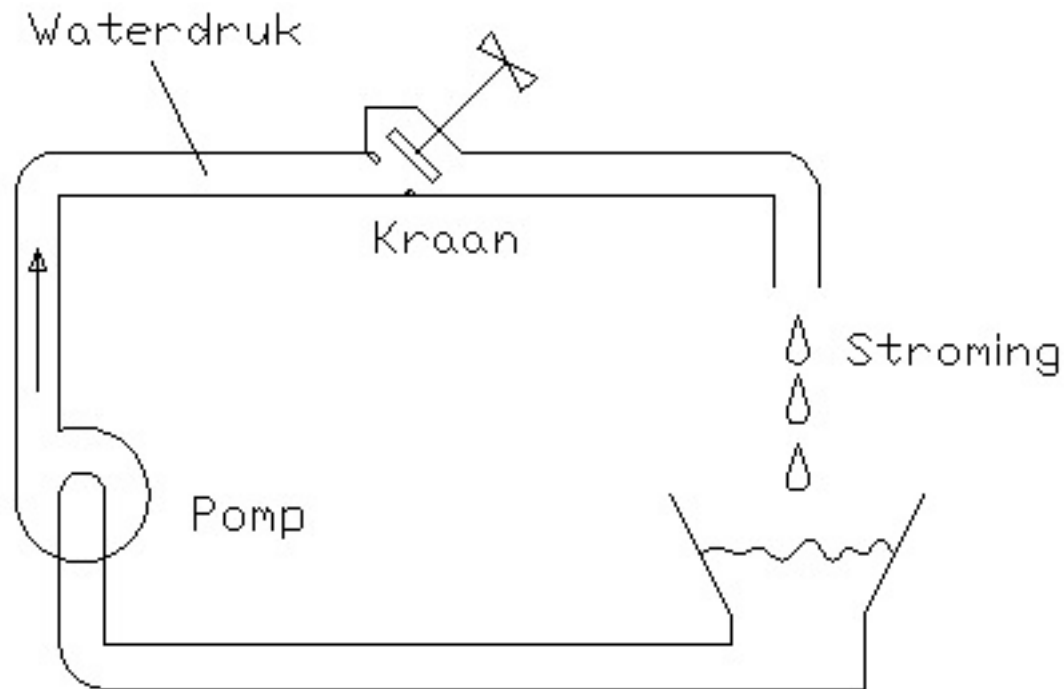
Dat is de afspraak :)





Electriciteit

- **Elektrische spanning, stroom en weerstand:** een waterleiding als voorbeeld.
- Spanning = druk die de pomp van het waterleiding bedrijf levert.
- Kraan bijna dicht = heel grote weerstand, er stroomt erg weinig water.
- Kraan half open = de weerstand is flink minder en heb je al een aardige straal.
- Kraan helemaal open = bijna geen weerstand meer en het water spuit eruit.



Hoe verder de kraan open, des te minder weerstand, en des te meer stroming.





Wet Van Ohm

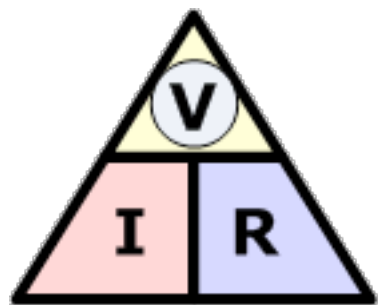
- Spanning = U, uitgedrukt in V = Volt
- Stroom = I, uitgedrukt in A = Ampère
- Weerstand = R, uitgedrukt in Ω = Ohm

Wet van Ohm

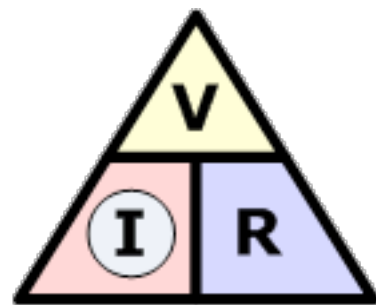
$$U = I \times R$$

Spanning = Stroomsterkte x Weerstand

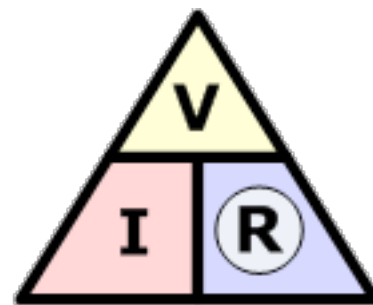
Ohm's Law traingle



$$V = I \times R$$



$$I = \frac{V}{R}$$



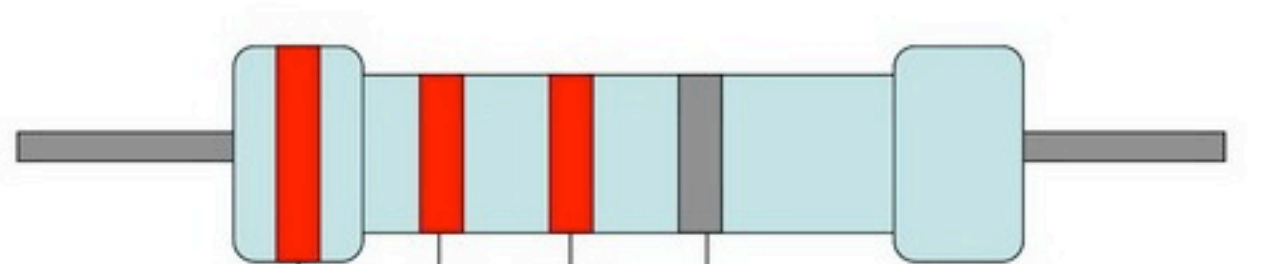
$$R = \frac{V}{I}$$





Weerstand?

- Een weerstand heeft een aantal kleurringen (3 of 4).
- Eerste 3/4 ringen geven de weerstand weer (in Ohm).
- De laatste ring geeft de tolerantie weer (= mogelijke afwijking).



		0,01	10%
		0,1	5%
0	0	1	
1	1	10	1%
2	2	100	2%
3	3	1K	
4	4	10K	
5	5	100K	
6	6	1M	
7	7	10M	
8	8		
9	9		

De 1^e en 2^e ring vormen het getal.

De 3^e ring is de vermenigvuldigingsfactor.

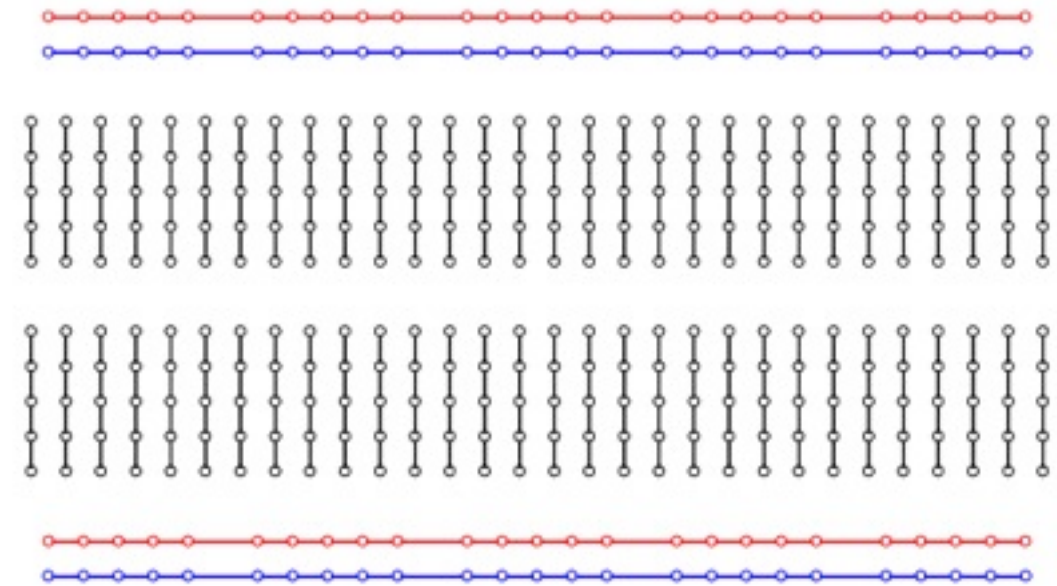
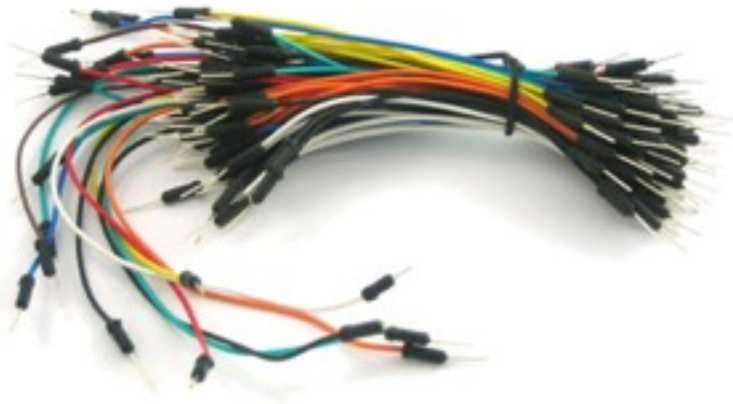
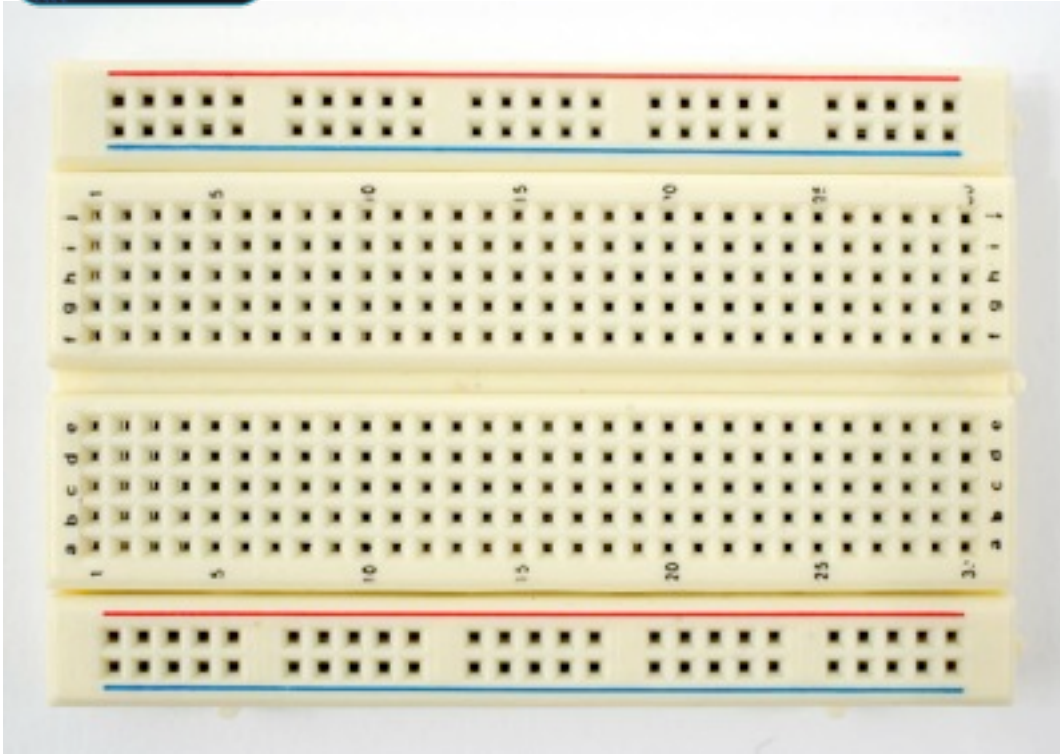
De 4^e ring is de tolerantie.

Deze weerstand:
 $22 \times 100 = 2.200 \text{ Ohm}$
met een tolerantie van 10%





Breadboard





Eerste Project

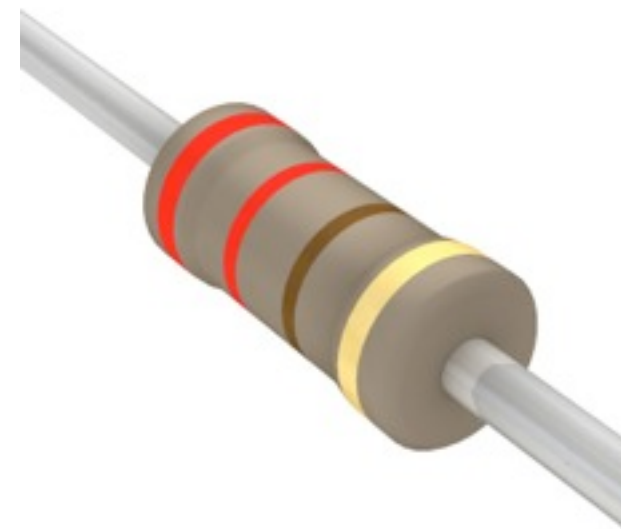
- Oplichten van een LED.
- Forward voltage drop = 1,8V (de LED gaat met 1,8 volt 'lopen').
- Stroom moet beperkt worden tot maximum 15mA (= 0,015A).
- Bereken de benodigde weerstand....

Wet van Ohm

$$U / I = R$$

$$5V - 1,8V / 0,015A = 213\Omega$$

(Afronden naar boven)

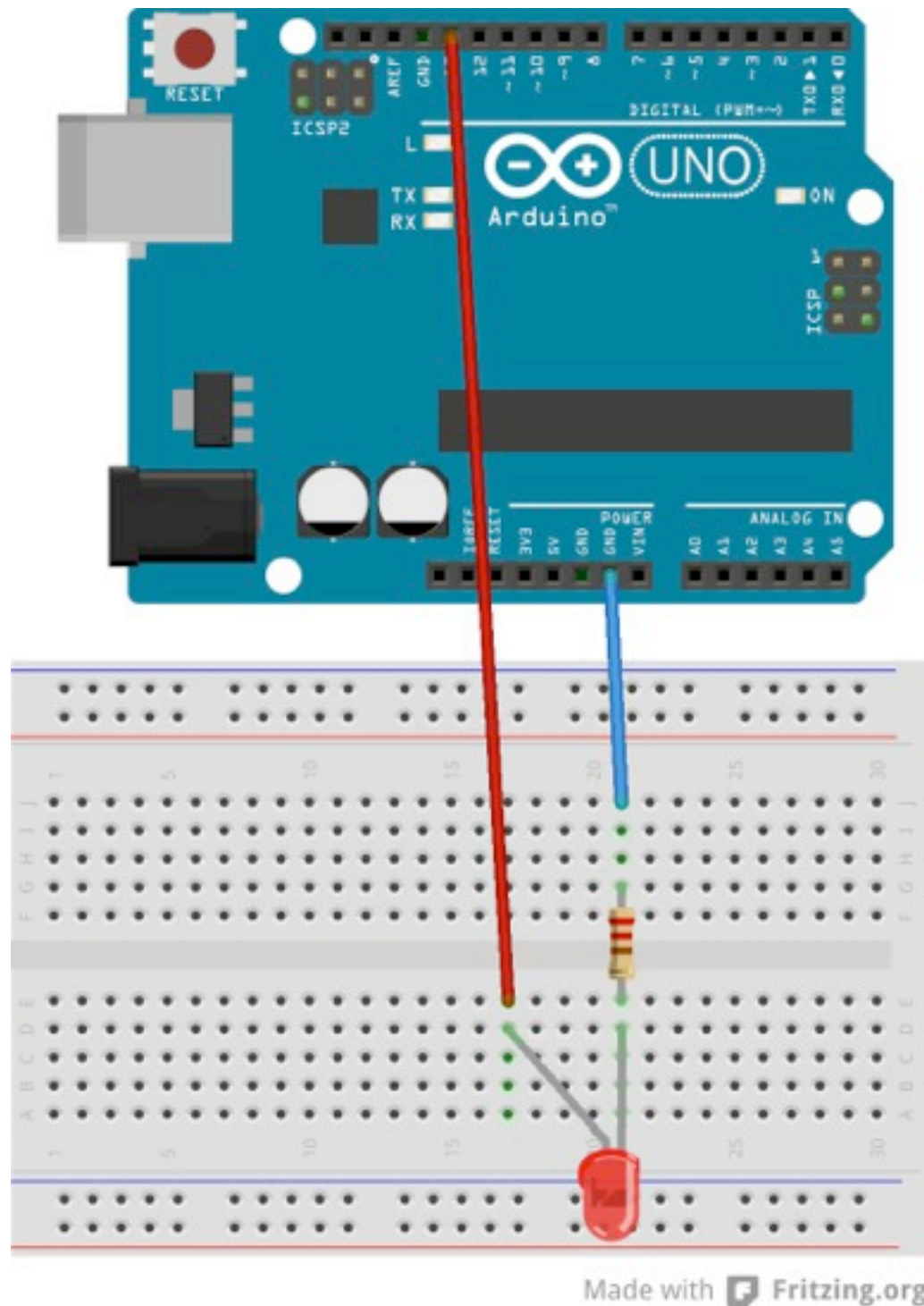


220Ω = Rood - Rood - Bruin





Project I: Blink





Project 1: Code

```
/*  
  Blink  
  */  
  
int led = 13; // Led pin.  
  
// the setup routine runs once when you press reset.  
void setup() {  
  // defineer de led pin als output.  
  pinMode(led, OUTPUT);  
}  
  
// the loop routine runs over and over again forever.  
void loop() {  
  digitalWrite(led, HIGH); // Zet de led AAN.  
  delay(250);               // Wacht.  
  digitalWrite(led, LOW);  // Zet de led AF.  
  delay(250);               // Wacht.  
}
```





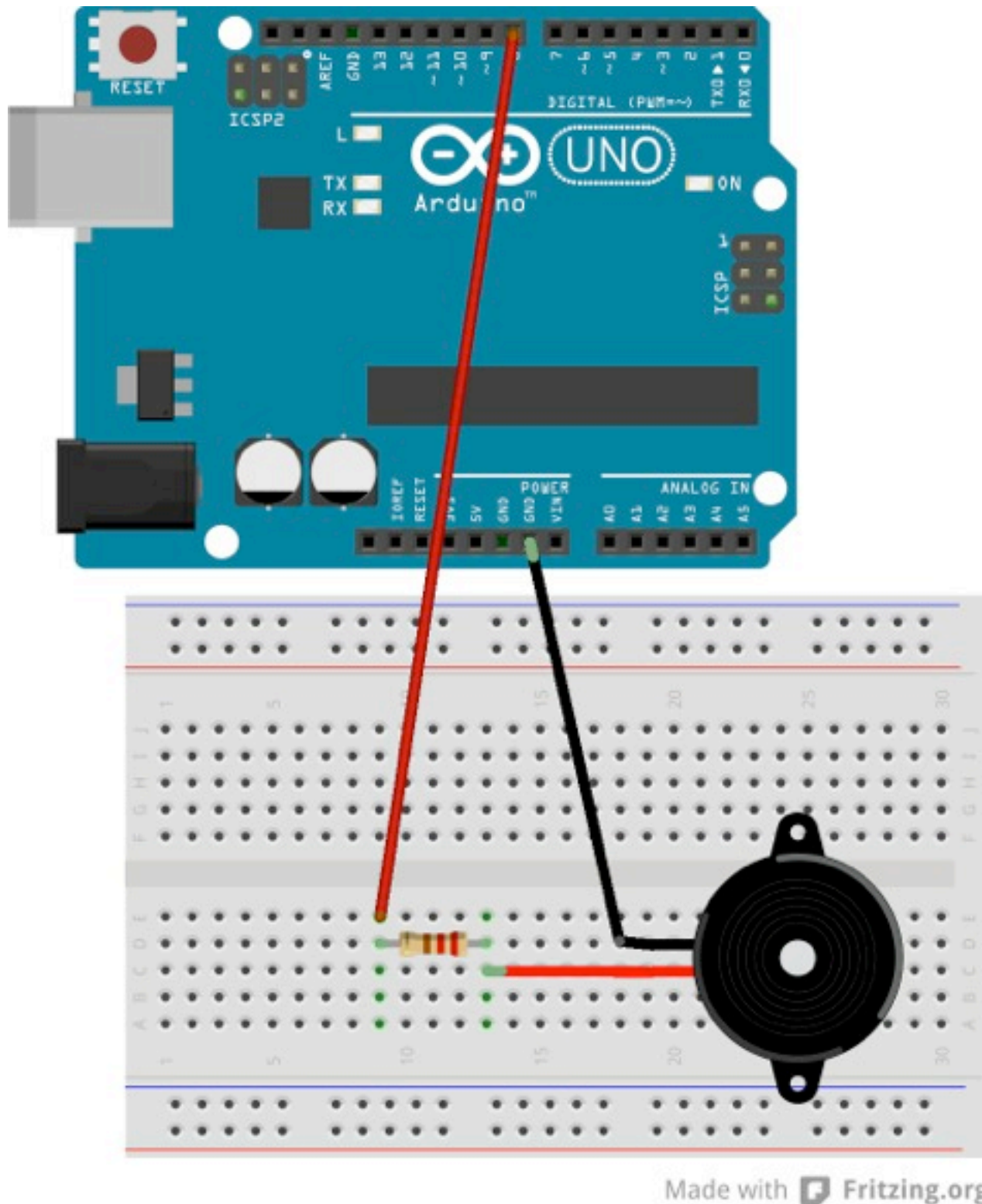
Piezo buzzer

- We gaan gebruik maken van een piezo buzzer
- Dit is een soort mini luidspreker
- Om de stroom te beperken gebruiken we een 220Ω weerstand





Project 2: Melodie





Project 2: Code

```
/*  
  Buzzer  
*/  
  
// Melodie  
int tonen[] = {262, 196, 196, 220, 196, 0, 247, 262};  
int toonDuur[] = {4, 8, 8, 4, 4, 4, 4, 4};  
  
int buzzer = 8; // Buzzer pin.  
  
// the setup routine runs once when you press reset:  
void setup() {  
}  
  
// the loop routine runs over and over again forever:  
void loop() {  
  
  for(int toon = 0; toon < 8; toon++) {  
    int duur = 1000 / toonDuur[toon];  
    tone(buzzer, tonen[toon], 300);  
    delay(duur * 1.4);  
    noTone(buzzer);  
  }  
  
  delay(2000);  
}
```





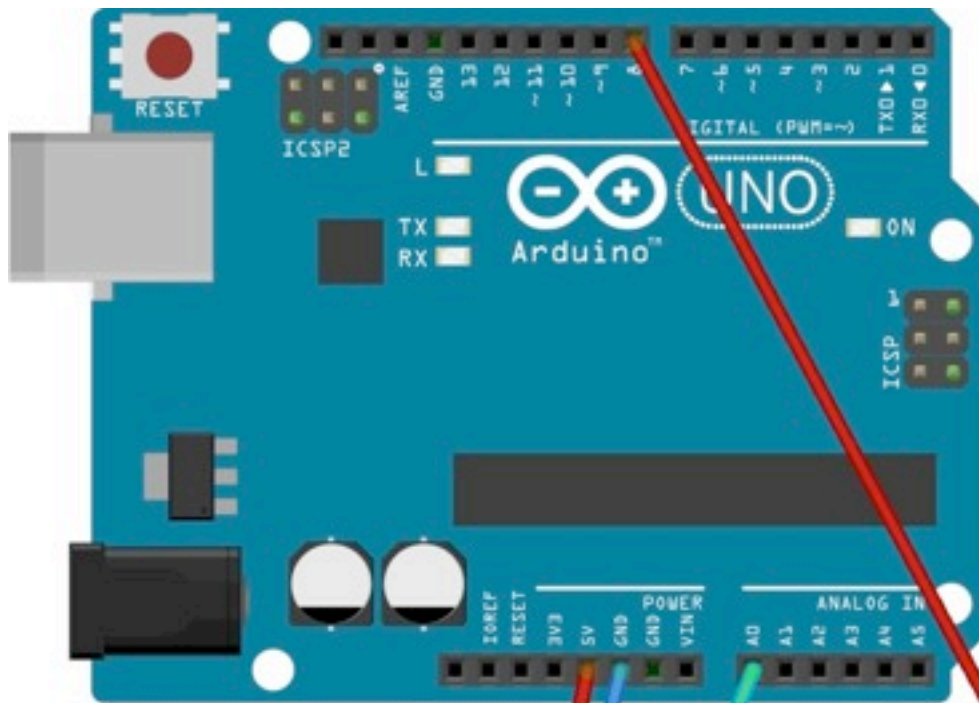
Photoresistor

- De tot nu toe gebruikte weerstanden hadden een **vaste** waarde.
- De weerstand van de photoresistor is afhankelijk van het licht.
- Beperk de stroom met een 10K weerstand.
- Zoek de photoresistor...

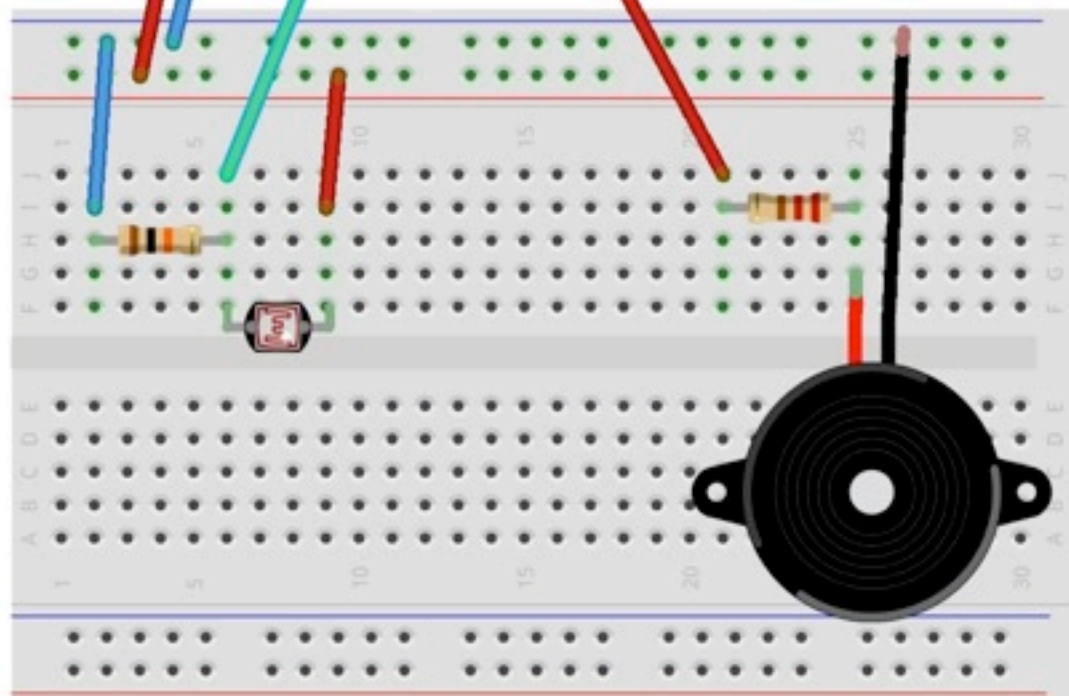




Project 3: Synthesizer



- Een analoge pin leest verschillende waarden, niet enkel AAN/UIT.
- Een analoge pin leest het Voltage dat binnenkomt.



Made with  Fritzing.org





Project 3: Code

// Synthesizer

```
int photoResistor = 0; // Pin voor de photoresistor.  
int buzzer = 8;        // Pin voor de buzzer.
```

```
void setup()
```

```
{
```

```
  // Communicatie initialiseren.
```

```
  Serial.begin(9600);
```

```
  pinMode(buzzer, OUTPUT);
```

```
}
```

```
void loop()
```

```
{
```

```
  // Lees de analoge waarde van de photoresistoren zet om naar een digitaal getal.
```

```
  int input = analogRead(photoResistor);
```

```
  //Print de digitale waarde van de photoresistor.
```

```
  Serial.println(input);
```

```
  /* We vertalen ('mappen') de input waarde tussen 100 en 1000 naar een  
     waarde tussen 150 en 1500. deze laatste stellen de waarden voor in Hz.  
     bvb. 100 -> 150, 1000 -> 1500
```

```
     De input waarden dienen mogelijk te worden aangepast aan de lichtsterkte van de ruimte!
```

```
  */
```

```
  int toon = map(input, 100, 1000, 150, 1500);
```

```
  tone(buzzer, toon, 10);
```

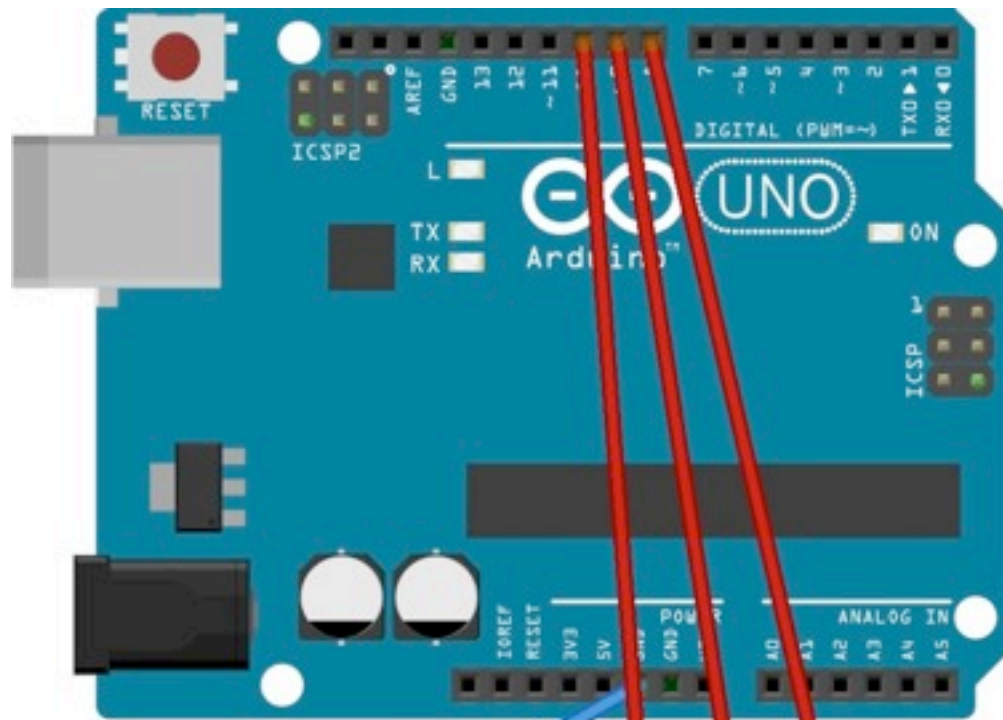
```
  delay(1); //short delay for faster response to light.
```

```
}
```

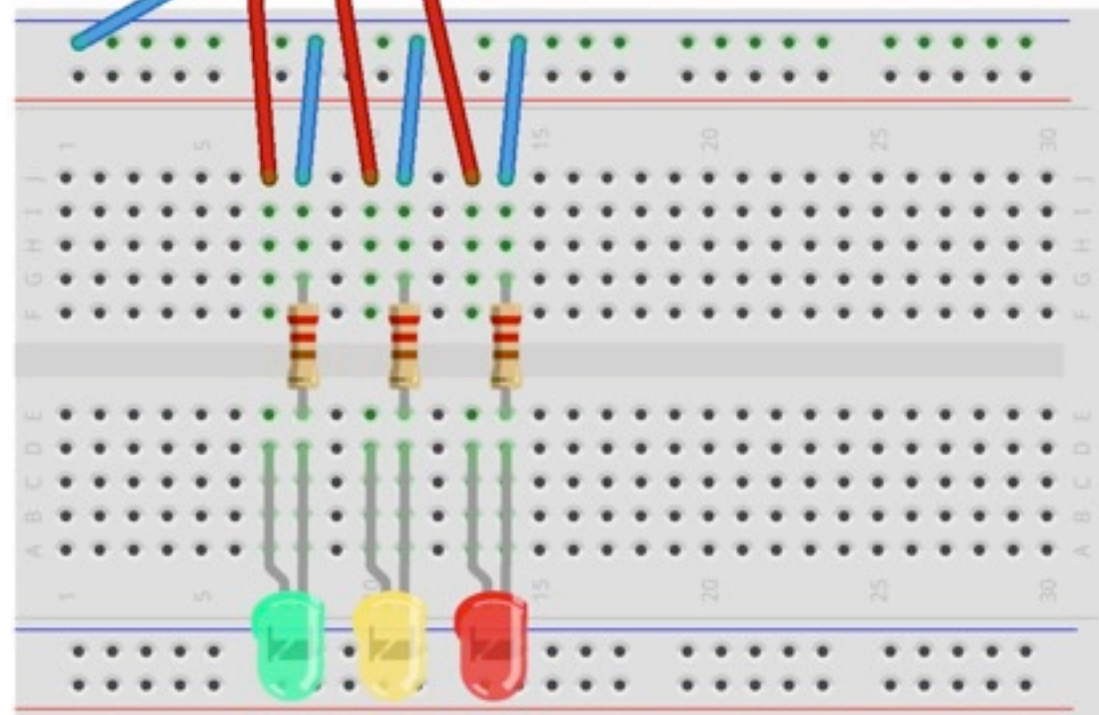




Project 4: Traffic Light



Bouw een verkeerslicht



Made with  Fritzing.org





Project 4: Code

```
/*
```

```
  Traffic light.
```

```
*/
```

```
int red = 8;
```

```
int orange = 9;
```

```
int green = 10;
```

```
// the setup routine runs once when you press reset:
```

```
void setup() {
```

```
  // initialize the digital pin as an output.
```

```
  pinMode(red, OUTPUT);
```

```
  pinMode(orange, OUTPUT);
```

```
  pinMode(green, OUTPUT);
```

```
}
```

```
// the loop routine runs over and over again forever:
```

```
void loop() {
```

```
  digitalWrite(green, HIGH);
```

```
  delay(1000);
```

```
  digitalWrite(green, LOW);
```

```
  digitalWrite(orange, HIGH);
```

```
  delay(1000);
```

```
  digitalWrite(orange, LOW);
```

```
  digitalWrite(red, HIGH);
```

```
  delay(1000);
```

```
  digitalWrite(red, LOW);
```

```
}
```





Meer weten?



<http://fritzing.org/>



<http://www.arduino.cc/>



<http://nl.wikipedia.org>

