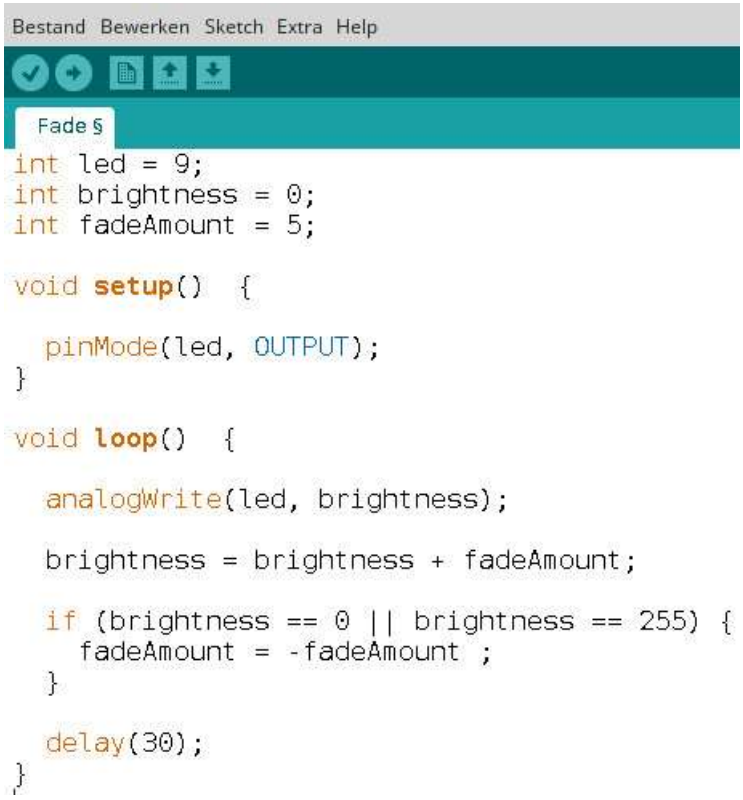
Fade led

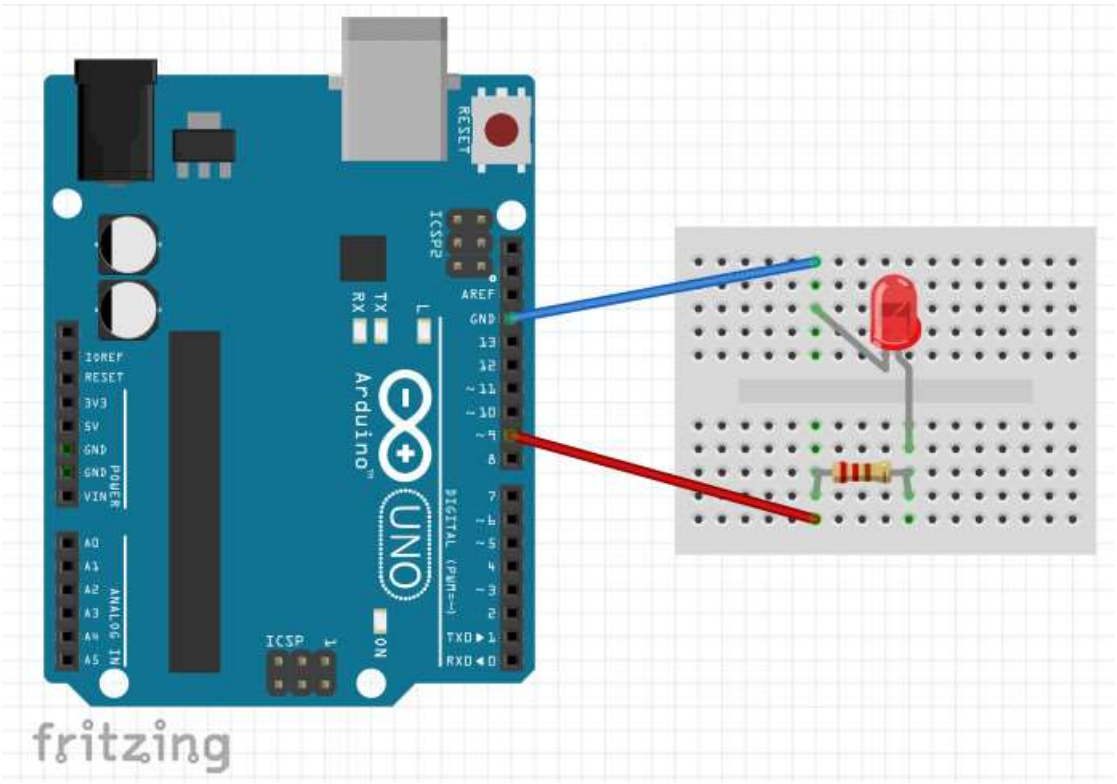
De les

In deze les laten we een led steeds feller branden totdat deze helemaal AAN is.

De code



Fade led



Hoe werkt het programma?

Op de eerste regels worden een aantal variabele gegeven: `int led = 9;` hier wordt dus het pin nummer gegeven, en `int brightness = 0;`, dit is de lichtsterkte van de led en begint dus in het programma met 0, en `int fadeAmount = 5;` de lichtsterkte zal worden aangepast in stappen van 5.

De reden dat in deze sketch niet pin 13 wordt gebruikt is omdat pin 9 een pin is met "Pulse Width Modulation" (PWM). Als je op je Arduiono kijkt zie je dat er voor het nummer 9 een ~ staat. Alle pinnen met een ~ ervoor hebben PWM. PWM betekent dat deze pinnen het aankunnen om de stroomsterkte in een heel snel tempo aan te passen. En dat is presies wat we gaan doen met deze Fade Sketch. De lichtsterkte zal in stappen van 5 toenemen of afnemen.

Dan krijgen we deze regel `pinMode(led, OUTPUT);` en geeft aan dat pin 9 een OUTPUT pin is. Dit staat tussen de { } van de functie `void setup()` en wordt daarom alleen bij de start van het programma uitgevoerd.

Dan krijgen we een loop die voor altijd doorgaat: `void loop() { }` Dus de regels code die tussen de accolades staan worden steeds herhaald totdat je het programma stopt.

Dan krijgen we de regel `analogWrite()` en deze functie krijgt twee gegevens mee: `led` en `brightness`. De led blijft hetzelfde, dit is pin 9, maar de brightness zal steeds veranderen en dat komt door de volgende regel:`brightness = brightness + fadeAmount;`Dus eerst is brightness 0 dus dan staat er 0=0+5. Omdat deze

Fade led

regel in de loop staat zal deze regels elke keer uitgevoerd worden. Dus dan wordt het 5=5+5, en daarna 10=10+5 etc.. Dit mag niet voor altijd doorgaan want de brightness moet tussen de 0 en de 255 blijven. Daarom wordt er in het programma nog een regel toegevoegd die er voor moet zorgen dat brightness tussen de 0 en 255 blijft: `if (brightness == 0 || brightness == 255)` dit betekent: als de lichtsterkte is gelijk aan 0 of de lichtsterkte is gelijk aan 255 dan: `fadeAmount = -fadeAmount ;` in deze regel zie je dat fadeAmount veranderd van + naar - maar het betekent tegelijkertijd ook dat als fadeAmount al - was, deze weer + wordt. Dus als de brightness 255 is dan wordt het fadeAmount -5. Dus dan krijg je 255=255-5, en daarna 250=250-5 en dan 245=245-5 etc, dit gaat weer net zolang door totdat brightness 0 wordt. En dan zal fadeAmount weer +5 worden. Dus de lichtsterkte is op zijn sterkts bij een brightness van 255 en daarna zal de lichtsterkte weer afzwakken tot de brightness weer 0 is. En dan neemt de brightness weer toe, enz

En dan de laatste regel`delay(30);` , deze regel geeft aan hoe lang er gewacht moet worden tot de volgende herhaling van de loop. En dat is maar 30 milliseconde!

Om te expirimenteren kun je de milliseconden wel aanpassen. Probeer maar eens een lagere waarde of een hogere waarde. Om het programma weer te uploaden naar de Arduino klik je weer op de pijl.