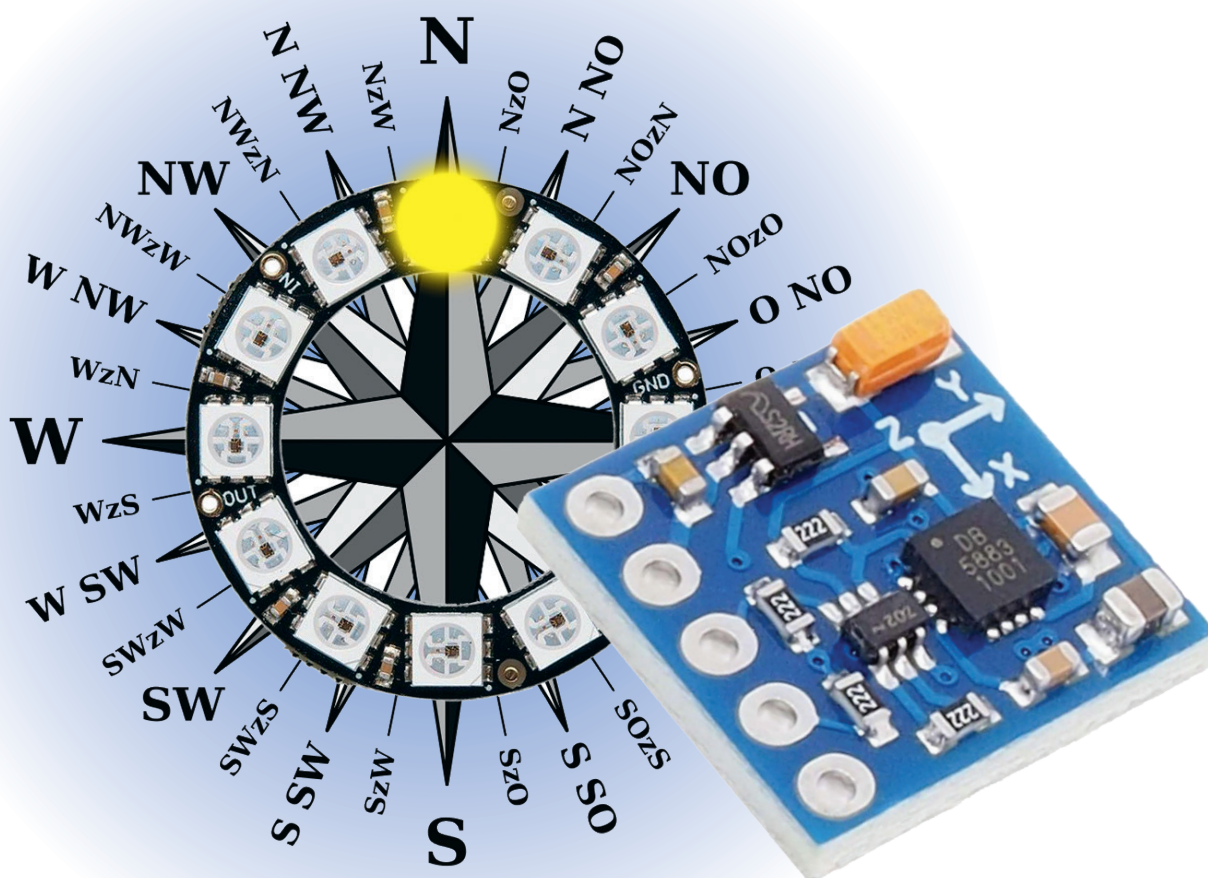


# Een kompasroos met de GY-271

Of waarom we in een acht bewegen om een sensor te kalibreren



Deze prachtige kompasroos is gemaakt voor het public domain door Soeren Gasch: [https://commons.wikimedia.org/wiki/File:Compass\\_rose\\_german.svg](https://commons.wikimedia.org/wiki/File:Compass_rose_german.svg).

**Rolf Hase** (Duitsland)

Het bouwen van een elektronisch kompas uit drie hoofdonderdelen zou kinderspel moeten zijn: een magnetisch veldsensor op een breakout-board, een NeoPixel-LED-ring als kompasroos, en een Arduino met een kant-en-klaar programma om alles aan te sturen. Maar wanneer het kompas niet helemaal werkt zoals je gedacht had, is er geen andere keuze dan terug te gaan naar de basis om een verbijsterend probleem op te sporen. En de oplossing van het probleem is niet minder verrassend!

Vroeger, als je een fietstochtje maakte door onbekend gebied, was een kaarthouder een onmisbare accessoire. Tegenwoordig klem je gewoon een smartphone aan je stuur of,

voor een overzicht op grotere schaal, leg je een tablet in je stuurmandje. Op mijn smartphone heb ik de app OsmAnd geïnstalleerd, die kaarten gebruikt van OpenStreetMap.

Helaas heeft het toestel geen kompas en moet ik het dus doen met de piepkleine koersindicatie op het scherm van OsmAnd; alleen, als ik stop met bewegen, werkt dit helaas niet meer

en heb ik dus weinig aan mijn oriëntatie. De kaart op het scherm draait heen en weer, en dus moet ik een meter of 20 afleggen (eerst in de verkeerde richting, uiteraard) om me weer te kunnen oriënteren.

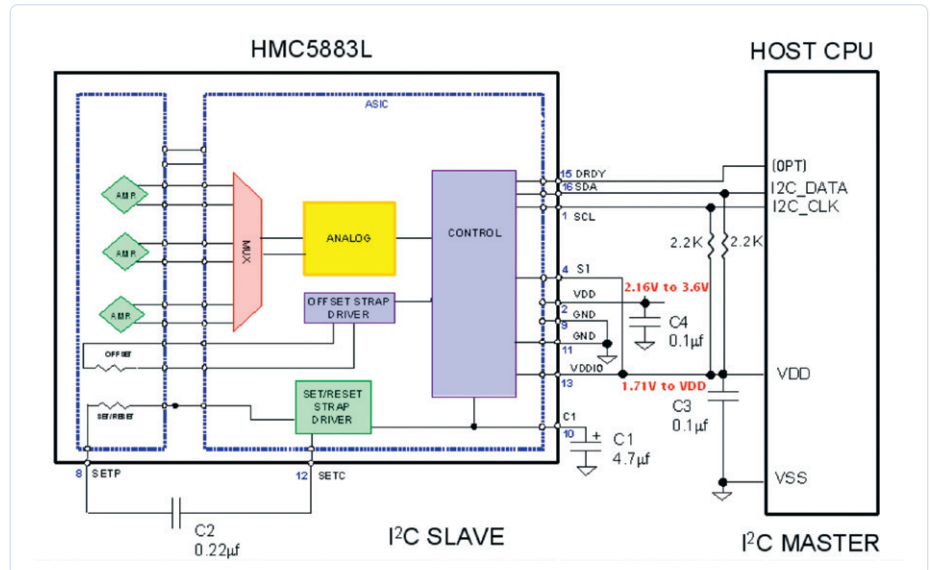
Daarom kwam ik op het idee om een apart, duidelijk zichtbaar kompas mee te nemen, zonder een tere naald die last kan hebben van mechanische trillingen. Bij het doorbladeren van Elektor begon het idee van een elektronische oplossing zich af te tekenen: met elektronica van de huidige generatie zou het mogelijk moeten zijn een betrouwbaar en nauwkeurig apparaat te maken.

Een aanpak met een Arduino Nano, een NeoPixel-LED-ring, en de GY-271 staat beschreven in [1], inclusief schema en broncode. Het bleek alles te doen wat ik wilde, dus trok ik mijn portemonnee en kocht een module. Omdat ik graag wat reserve-onderdelen bij de hand heb en de module niet duur is, kocht ik drie stuks.

## De GY-271 kompasmodule

De GY-271 is een breakout-board ontworpen voor de Arduino, en zijn headerpinnen passen perfect in een gewoon breadboard. Het belangrijkste IC op het bordje is een drie-assig kompas met onderdeelnummer HMC5883L of QMC5883. Dit apparaatje zit vaak in smartphones en is dus een goedkoop massaproduct. De interne schakeling van het IC is afgebeeld in **figuur 1**. De andere componenten op de printplaat zijn slechts een 3,3 V LDO-spanningsregelaar, wat afvlak- en bypass-condensatoren, en twee level-shifters voor de I<sup>2</sup>C bus-interface. Hieruit kunnen we afleiden dat de module gevoed kan worden met een 3 V tot 5 V-voeding en dat hij met een aangesloten microcontroller (dat kan een Arduino, een Raspberry Pi of een ander 'intelligent' component zijn) praat via I<sup>2</sup>C. De details van het communicatieprotocol zijn zeer gedetailleerd beschreven in de begeleidende documentatie en in een gratis e-book met toepassingen, dat in verschillende talen beschikbaar is.

Omdat we niet te maken hebben met analoge signalen, is het bouwen van de schakeling vooral een kwestie van het aansluiten van de voedingen. De LED-ring wordt gevoed via twee soldeerpaden voor voeding en massa, en er is een enkel logisch signaal nodig om de NeoPixels van een klok te voorzien. Om ruimte te besparen, gebruikte ik een Arduino Pro Mini, en de Arduino IDE moet daarvoor geconfigureerd zijn. Voor het programmeren van de Arduino heb je een FTDI USB-naar-serieel kabel nodig.

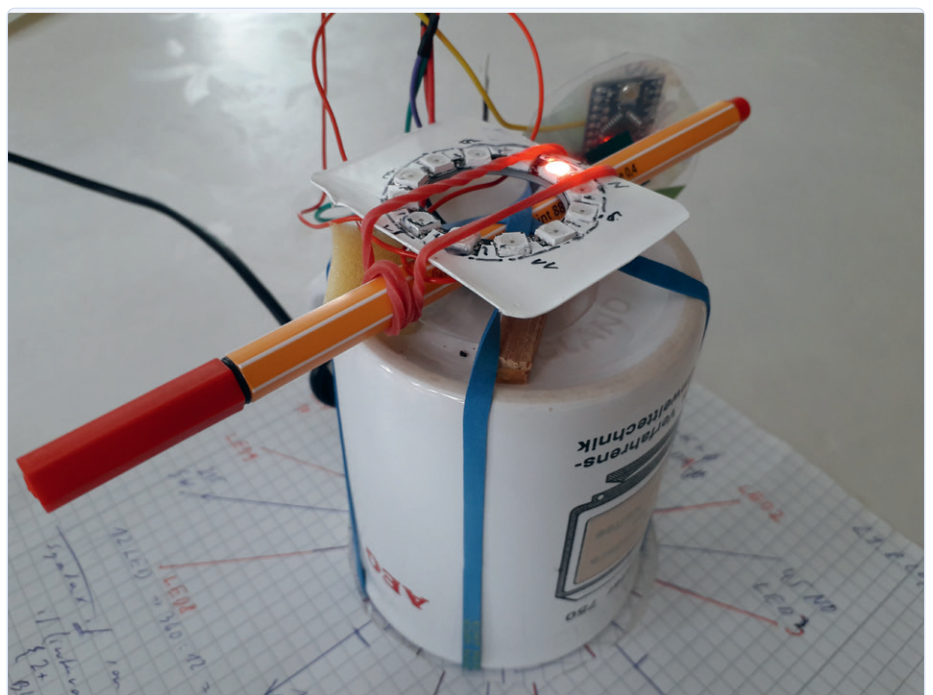


Figuur 1: Interne schakeling van de HMC5883L-kompassensoren (Bron: Honeywell datasheet).

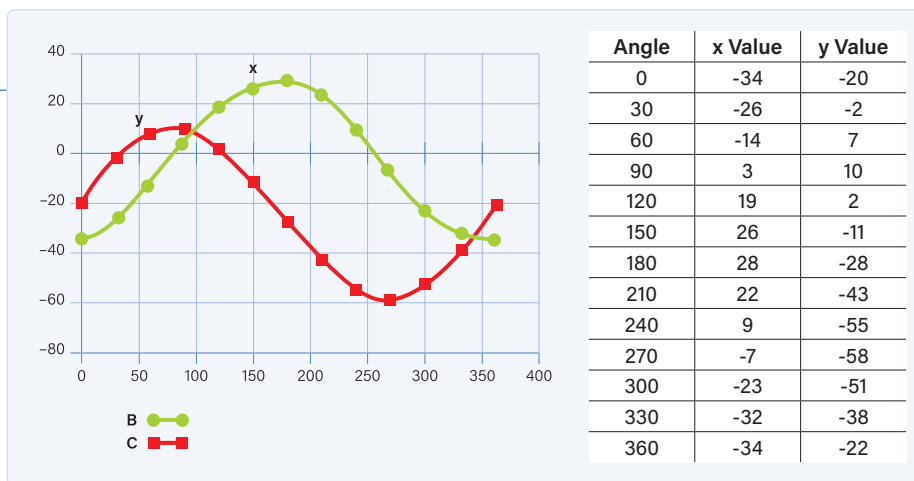
## Tot zover is alles niet goed

In theorie zou het bouwen van het kompas en het installeren ervan op mijn fiets dus geen enkel probleem mogen opleveren. Maar de eerste test maakte daar al snel korte metten mee! Slechts één kwadrant werkte, en daarna sprong de 'naald' met 180 graden. Speuren in de (inmiddels ingetrokken) broncode naar de mogelijke oorzaak leverde niets op, en ik schreef het probleem toe aan op eBay gekochte merkloze producten van slechte kwaliteit.

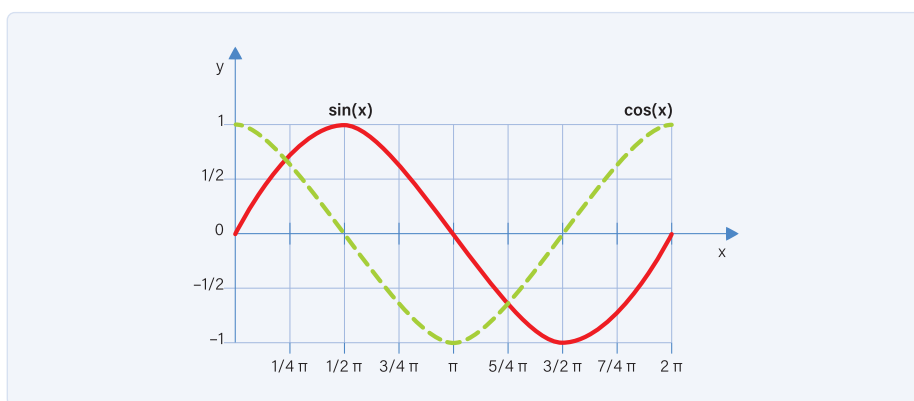
Kortom, het enige pragmatische wat ik kon doen was empirisch te werk gaan en proberen het probleem tot op de bodem uit te zoeken. In de eerste experimenten stelde ik vast dat het apparaat zeer gevoelig was voor externe velden, zoals bijvoorbeeld in de buurt van mijn laptop. Proberen te experimenteren met een opstelling bestaande uit een kwetsbare massa draden, is niet aan te bevelen, en daarom heb ik een robuustere experimentele opstelling geconstrueerd, zoals te zien in **figuur 2**. Om



Figuur 2: Slimme experimentele opstelling om de bug op te sporen..



Figuur 3: De sensormetingen vertonen een offset, en dat leidt tot het buggy gedrag.



Figuur 4: Zo zouden de twee curven er idealiter uit moeten zien.

een beetje uit te leggen wat je ziet, ik koos voor een omgekeerde koffiemok die gegarandeerd vrij is van magnetische anomalieën. De printplaat en (niet zichtbaar op deze foto) de sensor en de ring met LED's zijn met elastiekjes aan de onderkant van de mok bevestigd, met een pen eronder die als 'noord'-indicator fungeert. Een klein stukje hout en een spons fungeren tegelijkertijd als afstandhouder en ondersteuning. De richtingen van de x- en y-assen zijn aangegeven op de componentenzijde van de GY-271-module. De rode dop van de pen komt overeen met de richting van LED 0 en de negatieve x-richting (!) van de sensor, die met de componentenzijde naar beneden is gemonteerd. De z-as van de sensor wordt in dit ontwerp niet gebruikt, maar het apparaat moet zo worden afgesteld dat het als nul wordt uitgelezen om fouten in het systeem te minimaliseren. (Het uitlezen van de z-as, in combinatie met een beetje wiskunde, zou bruikbaar kunnen zijn om die 'hors catégorie'-beklimmingen in de bergen te meten). De mok staat op een vel papier waarop een primitieve kompasroos is getekend. In het midden is een cirkel uitgesneden zodat hij gemakkelijk kan draaien. En

hoewel het er misschien niet bijzonder verfijnd uitziet, leverde deze opstelling niet alleen reproduceerbare resultaten op, maar leidde hij ook tot een opmerkelijk 'aha'-moment.

Ik draaide het koffiemok-apparaat dus rond en voerde de waarden van de x- en y-as in de Excel-spreadsheet van **figuur 3** in (de waarden van de x-as in blauw, die van de y-as in rood). Als je deze waarden vergelijkt met de ideale waarden in **figuur 4**, kun je meteen een afwijking zien. Het gaat om een sinus- en een cosinuscurve (of een met 90° verschoven sinuscurve). Zoals we met de paplepel ingegoten hebben gekregen, is in een rechthoekige driehoek de sinus van een hoek gelijk aan de overstaande rechthoekszijde gedeeld door de schuine zijde, de cosinus is gelijk aan het aanliggende gedeeld door de schuine zijde, en de tangens is gelijk aan de overstaande rechthoekszijde gedeeld door de aanliggende; dat betekent dat de sinus gedeeld door de cosinus de tangens is, want de schuine zijde telt niet mee. Om de hoek te bepalen uit de zijden hebben we de inverse functie nodig, de arctangens, of arctan.

De Arduino bibliotheken bevatten een

arctan2-functie, die de x- en y-waarden als argumenten neemt. De nominale amplitudes van de x- en y-signalen moeten gelijk zijn, maar het is niet nodig om die amplitude te normaliseren naar 1. Nu kunnen we zien dat de verschuiving van het nulpunt de oorzaak van het probleem is. Dit verklaart ook het bizarre recept voor het kalibreren van het kompas van een smartphone, waarbij je hem in patronen van een acht moet bewegen. Deze smartphones gebruiken dus duidelijk HMC magnetische sensoren! In sommige gevallen heb ik gehoord dat zelfs deze formule niet werkt, blijkbaar omdat de afwijking van het nulpunt te groot is om te corrigeren.

Het is ook mogelijk de sensor 'automatisch' te kalibreren op een soortgelijke manier op de Arduino, door de vereiste correcties af te leiden uit de maximale en minimale sensorwaarden. Het schrijven van de kalibratiewaarden naar de sensor vereist communicatie in twee richtingen, wat het aantal interface-pinnen verdubbelt naar twee. Maar aangezien we de volledige karakteristiek van de sensor hebben vastgelegd, kunnen we de benodigde correcties in de software berekenen en de bedrading simpel houden.

## Een nieuw programma

Een belangrijk pluspunt van de originele software is dat deze alle code bevat om het HMC-apparaat te initialiseren en er metingen van te ontvangen. Ik heb een paar wijzigingen aangebracht in de monitorcode om de uitgelezen waarden te schalen naar meer hanteerbare proporties door ze te delen door 100. De NeoPixel-ring heeft slechts 12 posities, dus een nauwkeurigheid van ongeveer  $\pm 10\%$  is voldoende. Als je van plan bent om een grotere NeoPixel-ring of een grafisch display te gebruiken, dan moet je de niet-geschaalde waarden behouden en op de juiste wijze verwerken voor de resolutie die je nodig

### Listing 1: Hard gecodeerde kalibratie van de metingen.

```
Serial.print("X Value: ");
x = x/100;
x = x+2;
Serial.print(x);
Serial.print(" Y Value: ");
y = y/100;
y = y+20;
Serial.print(y);
Serial.print(" Z Value: ");
z = z/100;
Serial.print(z);
```

hebt. Voor het aansturen van de NeoPixels heb ik gebruik gemaakt van een afgeslankte versie van de zeer duidelijk geschreven code in [3], die geschikt lijkt voor het aansturen van 12 LED's. De eigenlijke correctie voor de offset in de HMC-sensor is te zien in het stukje broncode in **listing 1**. Het volledige, en uitgebreid aangepaste, Arduino-programma kun je downloaden in [4]. Bijzonder interessant als je verder wilt experimenteren met de code in de sketch is de monitor-functie.

## Veldproeven

Het kompas moet worden ingebouwd in een doorzichtig plastic doosje en aan het fietsstuur worden bevestigd. Binnenin is de constructie niet moeilijk, je kunt de verschillende lagen van elkaar scheiden door plaatjes plastic, allemaal met dubbelzijdig plakband bevestigd op een bodemplaat. Aan een andere plaat, met een cirkelvormige uitsparing en vertandingen kun je de NeoPixel-ring bevestigen. De kalibratietests werden uitgevoerd met LED 0 naar het noorden gericht, maar om de een of andere reden bleek bij het testen van de software dat deze oplichtte als het apparaat naar het zuiden gericht was. In plaats van de software aan te passen om het probleem op te lossen, was het veel eenvoudiger om het geheel zo op de fiets te monteren dat LED 6 recht vooruit wijst. Als de fiets nu naar het noorden kijkt, licht de 'recht vooruit'-LED op; draai naar rechts, en de volgende LED links licht op, opnieuw om de richting van het noorden aan te geven, precies zoals het hoort. Helaas bleek dat het restmagnetisme in het staal van het stuur de sensormetingen verstoort. Ik ontdekte echter (indirect) dat

## HET KOMPAS OP EEN E-BIKE

Ik heb een voeding voor het circuit op mijn e-bike als volgt geïmplementeerd. Verborgen onder het stuur zit een gewone aansluiting zoals je die in auto's vindt, die via twee ministeckertjes (rood en pastelblauw) verbonden is met het laadcontact van de accu van de e-bike. Een goede isolatie van deze aansluitingen is van vitaal belang, want ondanks de lage uitgangsspanning (meestal 36V) zit er in de accu genoeg energie om veel schade aan te richten. In geval van kortsluiting zal een ongeschikte kabel als een zekering doorbranden! De 5V-voeding voor de schakeling komt van een usb step-down converter met een 24V-ingang, afkomstig van een leverancier van onderdelen voor bedrijfsvoertuigen. Omdat vrachtwagenaccu's vaak met een veel hogere spanning worden opgeladen dan hun nominale spanning, hebben ze meestal een MC34063A-schakelomvormer, die bestand is tegen een ingangsspanning tot 40V. Haal de USB-omvormer uit elkaar en je kunt een tweaderige kabel met headerpinnen rechtstreeks op de 5V-uitgang solderen, het stroomverbruik van het kompas is vrij laag.



Figuur 5: Het gemonteerde kompas bevestigd aan mijn fietsframe.

het frame van mijn fiets in feite van aluminium is: in tegenstelling tot staal heeft aluminium zodanige eigenschappen dat het kompas er geen last van heeft. Omdat het apparaat klein is, kon ik het aan het frame bevestigen terwijl het toch binnen mijn gezichtsveld bleef (zie **figuur 5**). Tests 'in het veld' toonden aan dat het apparaat betrouwbaar werkte, waarbij het enige probleem vlakbij een spoorbrug optrad, waarschijnlijk veroorzaakt door het veld rond de bovenleiding; een volgende test op dezelfde plaats liet geen fout zien. Dus

dankzij dit kompas kan ik nu veel gemakkelijker mijn weg vinden in onbekend gebied!



200597-03

### Vragen of opmerkingen?

Heb je technische vragen of opmerkingen bij dit artikel? Stuur dan een e-mail aan de redactie van Elektor: [editor@elektor.com](mailto:editor@elektor.com).



### GERELATEERDE PRODUCTEN

- > **Elektor Arduino Electronics Bundle**  
[www.elektor.nl/elektor-arduino-electronics-bundle](http://www.elektor.nl/elektor-arduino-electronics-bundle)
- > **Arduino Sensor Kit**  
[www.elektor.nl/arduino-sensor-kit](http://www.elektor.nl/arduino-sensor-kit)



## WEBLINKS

- [1] **GY-271 Module:** [www.az-delivery.de/en/products/gy-271-kompassmodul-kompass-magnet-sensor-fuer-arduino-und-raspberry-pi](http://www.az-delivery.de/en/products/gy-271-kompassmodul-kompass-magnet-sensor-fuer-arduino-und-raspberry-pi)
- [2] **Quick-Start-Guide: GY-271 Kompassmodul Sensor (Free E-Book):**  
[www.az-delivery.de/products/gy-271-kompassmodul-kompass-magnet-sensor-fur-arduino-und-raspberry-pi](http://www.az-delivery.de/products/gy-271-kompassmodul-kompass-magnet-sensor-fur-arduino-und-raspberry-pi)
- [3] **NeoPixel Library:** [https://github.com/adafruit/Adafruit\\_NeoPixel](https://github.com/adafruit/Adafruit_NeoPixel)
- [4] **Project Page:** [www.elektormagazine.nl/200597-01](http://www.elektormagazine.nl/200597-01)