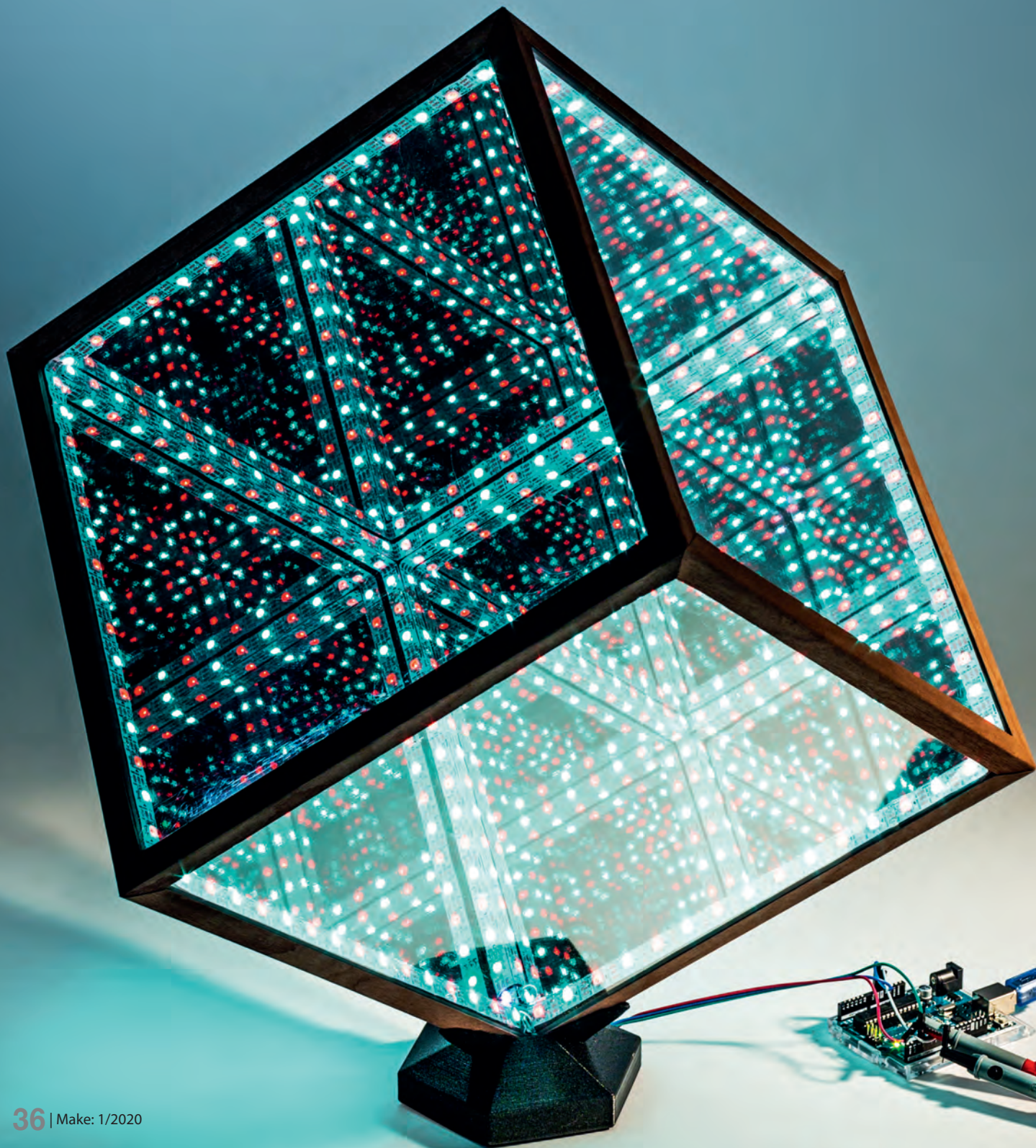


Infinity mirror in 3D

Dit is de overtreffende trap van het bekende infinity mirror-concept, oftewel de oneindige spiegel. Met meer spiegels en meer leds kun je kubussen, obelisken en veelvlakken bouwen. En met zulke flitsende 3D-objecten zul je gegarandeerd de aandacht trekken.

door Jan Thar



Semi-transparante spiegels worden samen met leds al lang gebruikt voor verrassende effecten in de vorm van oneindige spiegels of een smart mirror. In een driedimensionale vorm zoals een kubus zorgt dit concept voor nog meer fascinerende effecten, omdat je het oneindigheidseffect in alle richtingen kunt toepassen. De kubusvorm is erg geschikt voor beginners, omdat deze gemakkelijk te bouwen is en de symmetrie voor een chique kolomstructuur zorgt. Natuurlijk kun je ook complexere vormen bouwen. Bijzonder geschikt zijn zeer symmetrische objecten (veelvlakken) of vormen die er door de weerspiegelingen veel ingewikkelder en groter uitzien dan ze eigenlijk zijn.

Het basisprincipe is altijd hetzelfde: op een geraamte bevestig je ledstrips, bij voorkeur de populaire WS2812b of andere individueel aanstuurbare leds. Het geraamte bedek je vervolgens met semi-transparante spiegels. Met een zelf geprogrammeerde microcontroller of de bij de leds geleverde bediening stuur je het systeem aan. Tot slot kun je voor een strakker design de randen van de spiegels aan de buitenkant nog bekleden.

Afhankelijk van de grootte en complexiteit van de vorm zijn verschillende draagconstructies mogelijk. Van simpele houten balkjes tot lasergesneden panelen, 3D-prints, aluminiumstructuren en aluminium profielen, alles kan. Bouwdelen die met een lasercutter of CNC-freesmachine zijn gemaakt, hebben het voordeel dat ze precies op maat zijn en alleen nog gelijmd moeten worden. 3D-geprinte onderdelen zijn vooral geschikt voor kleine objecten. Hout is goedkoop en met een zaag snel op maat te maken. Hoe groter het object, hoe belangrijker de stabiliteit. Aluminium of zelfs een profielsysteem zoals OpenBuilds is hiervoor ideaal, maar dat loopt wel in de papieren. Hetzelfde geldt voor de bekleding. In het ideale geval is plakband voldoende.

Bij ledstrips is het aantal leds bepalend voor de kosten. Je hebt minstens 30 leds per meter nodig om mooie lichteffecten te bereiken. Voor een optimaal effect gebruik je 144 leds per meter. Maar dat is wel erg duur. Op de ledstrips kun je diffuserstroken of hete lijm uit een lijpistool aanbrengen, zodat de afzonderlijke lichtpunten niet opvallen maar meer in elkaar overgaan. Met monochrome ledstrips is het project erg eenvoudig omdat de controller dan weg kan blijven. Wij kozen een ledstrip met 60 individueel aanstuurbare leds per meter.

Voor de aansturing en voeding gebruiken we de wifi-microcontroller ESP32. Deze verstoppen we samen met alle kabels in de 3D-

Korte info

- » Oneindige kubussen bouwen
- » 3D-geprinte onderdelen ontwerpen met openSCAD
- » De FastLED-bibliotheek voor Neopixel gebruiken

Checklist



Tijdsduur:
6 uur



Kosten:
150 Euro (afhankelijk van de grootte)



Solderen:
basiskennis



3D-printen:
basiskennis



Materiaalbewerking:
zagen of Lasercutter



Vaardigheden:
basiskennis Arduino/ESP32

Materiaal

- » 6 m WS2812-strips, 60 led/m
- » Mikrocontroller (ESP32 Node-MCU)
- » 6 Doorkijkspiegels (30 x 30 cm)
- » 3D-geprinte voet
- » MDF-plaat 30 cm x 30 cm, 5 mm dik (of: 4 m vierkant hout 1 cm x 1 cm)
- » Houtlijm
- » Plakband
- » 3 m lintkabel 4-aderig
- » Optioneel: pinheader

Online info en downloads

- » make-magazin.de/xehw

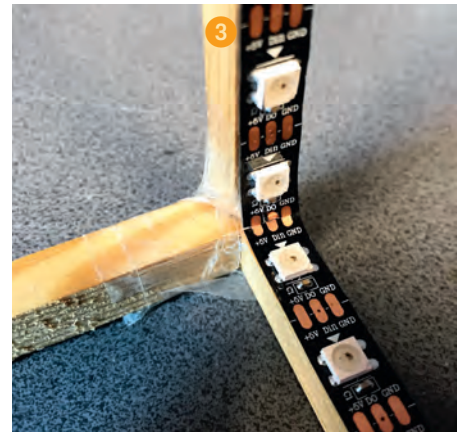
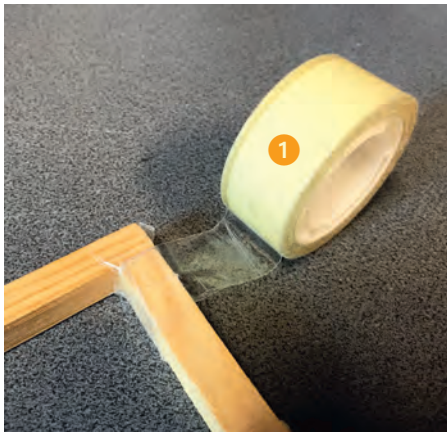
geprinte voet. Zo slaan we twee vliegen in één klap, want de lichteffecten van de kubus zijn het meest effectief wanneer de kubus op zijn punt staat. Je kunt ook andere microcontrollers gebruiken, zolang ze de leds maar kunnen aansturen. Hoe kleiner de controller, hoe kleiner de voet kan zijn.

Het grootste probleem was vreemd genoeg het vinden van een doorkijkspiegel van acryl. Het voordeel van acrylglas ten opzichte van normaal glas is dat het niet zo snel breekt als de kubus omvalt. Het is ook gemakkelijk om op maat te snijden met een lasersnijder. Als je alles zelf wilt doen, kun je een zonwerende spiegel folie op het acryl aanbrengen. Maar zelfs als je dat zorgvuldig doet, is het spiegel-effect niet zo goed als bij kant-en-klare doorkijkspiegels. Uiteindelijk hebben we onze acrylspiegel bij Pyrasied.nl gevonden, waar ook andere interessante materialen verkrijgbaar zijn die met een laser te snijden zijn. Als je geen lasersnijder hebt, kun je gewoon zes passende tegels bestellen.

Hier volgen nu de instructies voor een oneindige kubus. Wil je hierna nog complexere vormen uitproberen, dan staan er online (zie



De Obelisk op de Make Munich



de link in de Korte info) meer tips voor het bouwen van een oneindige icosaeëder, obelisk en oorhangers.

Het geraamte

Het geraamte van de kubus bestaat uit vierkante houten balkjes met een geschikte dikte voor de ledstrips. De strips zijn iets meer dan 1 cm breed. Het beste is om de grootte van de kubus af te stemmen op de rastergrootte van de gebruikte ledstrip. We gebruiken in dit geval 60 leds per meter en 15 leds per zijde. Omdat de ledstrips aan de binnenkant vastgelijmd worden, moet de binnenrand 25 cm lang zijn. Je kunt de kubus ook kleiner of groter maken. Als materiaal kun je op lengte gezaagde stukjes hout of stukken uit gelaserde MDF-platen gebruiken. Eerst maken we twee rechthoekige frames van telkens vier houten balkjes van elk 26 cm lang. We hebben het hout met houtlijm gelijmd en aan de buitenkant met plakband vastgezet. 1

De twee rechthoekige frames stellen we de kubus samen met vier andere houten balkjes. Deze balkjes zijn elk 25 cm lang. Ook hier gebruiken we plakband om het hout te fixeren. 2

Leds

Nu bevestigen we de ledstrips aan de binnenranden. We moeten rekening houden met de totale lengte en de aansluitpinnen van de strips.

De WS2812-strips bestaan uit losse strips van 50 cm lengte die aan elkaar gesoldeerd zijn. Voor onze kubus betekent dit dat de soldeerverbindingen op de hoeken liggen. Omdat dit meestal leidt tot een lichte verschuiving in de striplengte en de soldeerverbindingen moeilijk te buigen zijn, is het makkelijker om de strips eerst los te solderen, ze in de kubus te bevestigen en ze uiteindelijk weer aan elkaar te solderen. 3

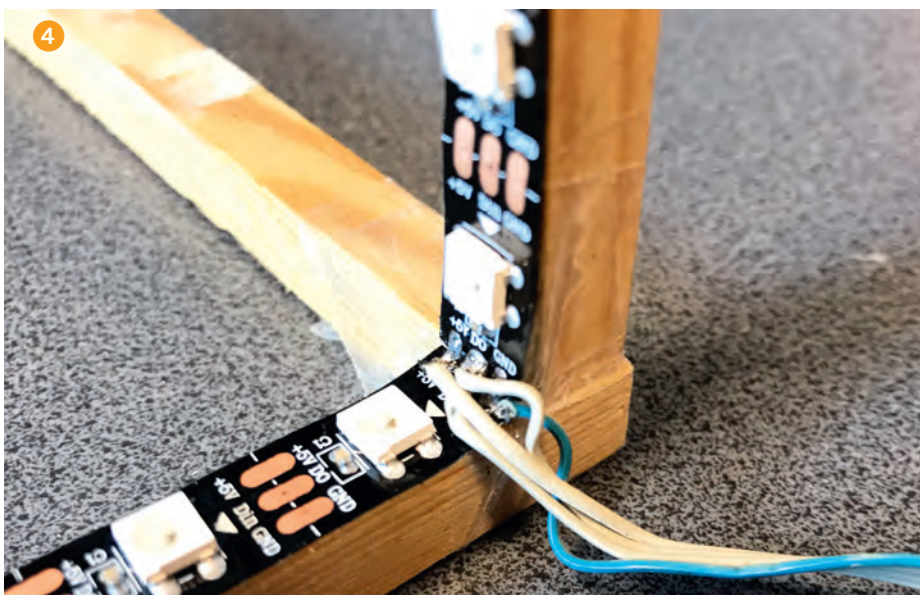
Bij het monteren is het belangrijk om rekening te houden met de oriëntatie van de strips. Naast de pinnen +5V en GND voor de voeding, is de data-ingang (Din) zichtbaar aan de ene kant en de data-output (D0) aan de andere kant. Kleine pijltjes naast de leds geven de richting ook aan. Om later verwarring bij het solderen van de kabels te voorkomen, plaatsen we de ledstrips aan drie zijden zo, dat de data-ingangen in een hoek van de kubus samenkomen. De strips van de andere zijden moeten samenkomen in de diagonaal tegenovergestelde hoek. De controller wordt later ook op één van de hoeken aangesloten.

Bedrading

Nu bevestigen we kabels aan beide hoeken. Voor elke ledstrip hebben we een 4-polige platte lintkabel nodig - met twee hoeken met drie strips wordt dat zes kabels.

We solderen twee van de vier kabelaansluitingen aan de pinnen van de voeding (+5V en GND), waarbij de soldeerverbinding ook het begin en einde van de ledstrip kan aansluiten. De twee overige kabels gaan gescheiden naar de datapinnen van de strip (Din en D0). Let op dat er geen kortsluiting ontstaat. Omdat de pinnen vaak dicht bij elkaar zitten, kun je de strook tussen de pinnen met een schaar inknippen en de soldeerpads een beetje uit elkaar buigen. 4

Op de hoek waar de controller later zal worden bevestigd zijn korte kabels van ca. 10 cm lengte voldoende. In de tegenoverliggende hoek moeten de kabels echter 90 cm lang zijn, zodat ze langs de randen van de kubus naar de controller kunnen lopen. Je kunt ook korte kabeltjes trekken van de data-uitgang van de ene strip naar de data-ingang van de volgende strip om alle strips aan te sluiten. Onze manier om alle data-ingangen en -uitgangen afzonderlijk naar buiten te leiden maakt het echter makkelijker om defecte leds op te sporen. Bovendien kun je dan de volgorde waarin de strips oplichten aanpas-



sen. De langere lintkabels leiden we langs de buitenranden van de kubus naar de tegenoverliggende zijde. Het beste is om de kabels langs verschillende randen te laten lopen zodat er nergens bobbel is.

Tot slot zet je alle kabels en ledstrips nog eens vast met doorzichtig plakband, dat je spiraalsgewijs daar omheen en om het hout wikkelt. Als de losgemaakte soldeerverbindingen van de ledstrips nog niet opnieuw gesoldeerd zijn, kun je dat nu doen. Vervolgens is het raadzaam om de strips even te testen. Je kunt een pinheader solderen of vastknippen aan de uiteinden van de lintkabel om deze aan te sluiten op een microcontroller. De benodigde stappen om een voorbeeldprogramma voor de leds te installeren en te starten, vind je verderop in dit artikel.

Als hele stripsecties niet werken, is het zaak de soldeerverbindingen tussen de strips en de kabel te controleren. Als er maar een led kapot is, kun je deze eenvoudig vervangen. Snijd hiervoor de snijkanten voor en achter de led af met een stanleymes - bij voorkeur zo, dat de volledige soldeerverbindingen op de strip blijven zitten. Verwijder vervolgens de led en snijd een nieuwe van een strip af. Bij deze strip moeten de soldeerverbindingen zoals aangegeven in het midden worden doorgesneden. Lijm het ledje nu in de juiste richting vast op de lege plek en soldeer de overlappende verbindingen aan elkaar.

Doorkijkspiegel

Na de bedrading zijn de doorkijkspiegels aan de beurt. Deze moeten een iets grotere afmeting hebben dan de 26 cm buitenrand van de houten balkjes, omdat er ruimte

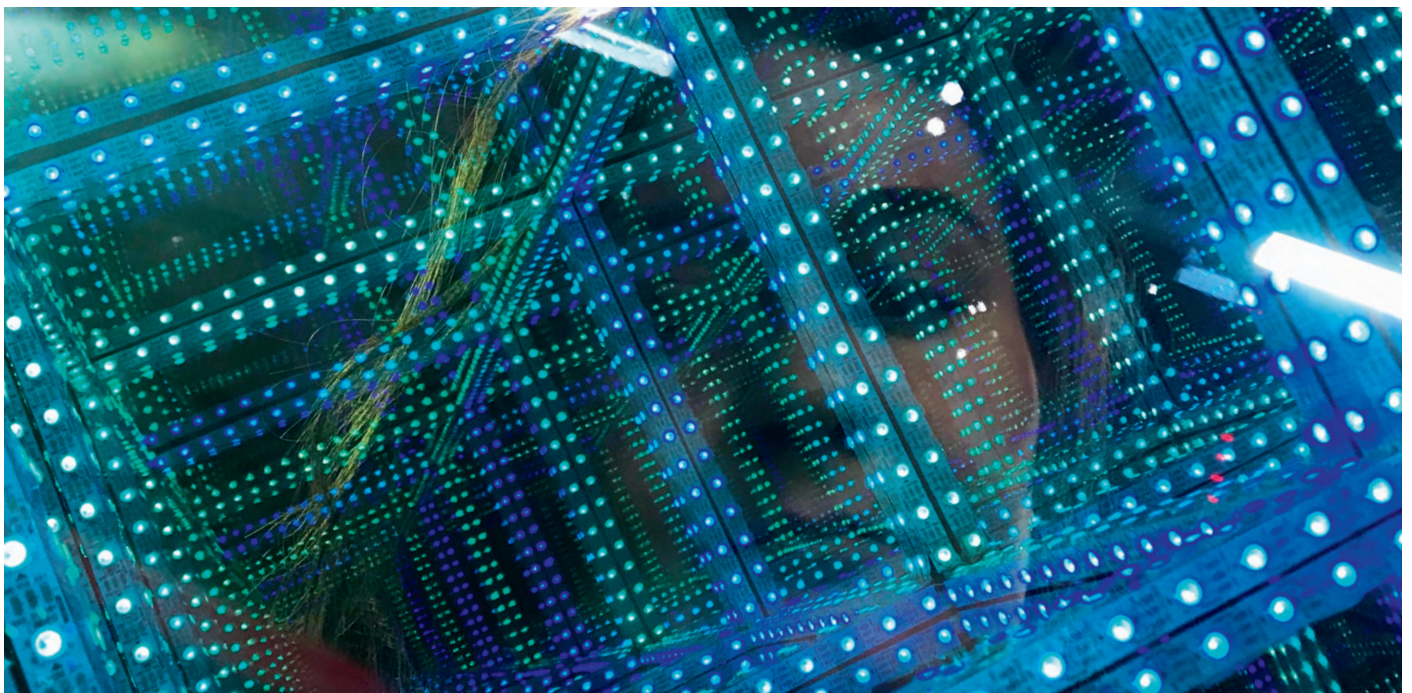


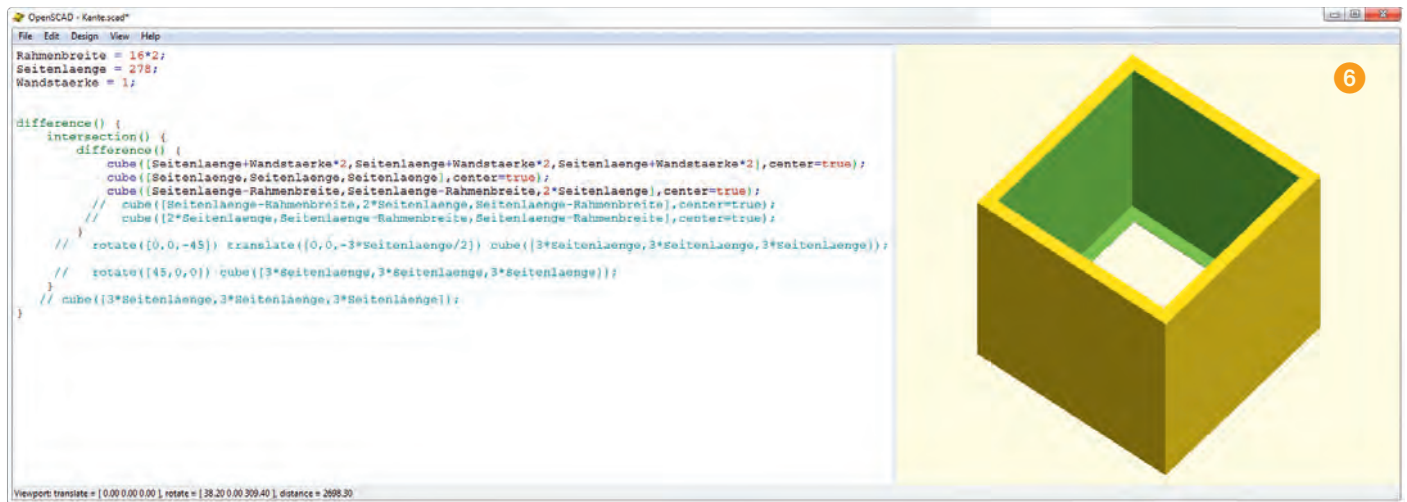
moet zijn voor de kabels tussen het hout en de doorkijkspiegel. In principe passen ook vierkanten van 26 cm, maar dan zijn de zijken en hoeken van de kubus niet hermetisch afgesloten. Als je een lasersnijder gebruikt, kun je de buitenkant van de kubus perfect passend maken. Je kunt bijvoorbeeld op de website makerbase.com vlakken met insteekbare randen ontwerpen en als snijsjabloon voor de cutter downloaden. Vergeet niet om in een hoek een uitsparing te maken waarlangs je de kabels naar buiten leidt.

Let er bij de bevestiging op dat je de zilveren coating van het acryl aan de binnenkant van de kubus plaatst. De spiegels maak je in eerste instantie gewoon met plakband aan elkaar vast **5**.

Randafdekking en voet

Om de kubus aan de buitenkant te verfraaien, hebben we randen ontworpen en geprint met de 3D-printer. Daarmee verdwijnen de spiegelranden en het plakband en ook de houten balkjes zijn dan afgedekt. Met een voet kun je de kubus decoratief op zijn punt zetten. De voet fungeert meteen als behuizing voor de microcontroller. De ontwerpen hebben we berekend met het gratis constructieprogramma OpenSCAD. Dit biedt verschillende geometrische basisvormen zoals kubussen en bollen. Deze kun je met rekenkundige bewerkingen in coderegels in elkaar integreren. Je kunt ons kant-en-klare SCAD-bestand via de link downloaden en aanpassen.





Om de randen aan je eigen kubus aan te passen, kun je de parameters aan het begin wijzigen. De afmetingen worden door de meeste 3D-printers opgevat als millimeters. De lengte van de zijde is de lengte van de binnenkant (in ons geval 278 millimeter). De randbreedte is 16 millimeter - maar omdat we in eerste instantie met de zijken van de kubus werken, wordt de framebreedte twee keer opgegeven ($16 \cdot 2$).

Dankzij de kubusvorm is het maken van de randbedekking vrij eenvoudig. Onder `difference()` maken we een buitenste kubus, waarvan we kleinere kubussen aftrekken tot alleen het frame overblijft. Om de spiegelkubus te omvatten, moet de buitenste kubus nog iets groter zijn en we voegen twee keer de wanddikte toe aan de randlengte. Vervolgens trekken we binnenin een kubus af. Om openingen in de zijvlakken te snijden, trekken we nog drie blokken af, die telkens in twee dimensies de grootte van de uitsparing hebben en twee keer zo lang zijn als onze zijken **6**.

Tot slot trekken we nog twee kubussen af om het frame tot één kant te reduceren. Omdat de randen te lang zijn voor het print-

bed van onze 3D-printer, delen we de kanten in de laatste regel middendoor **7**.

Als je de afzonderlijke functies beter wilt bekijken, kun je bepaalde regels van de programmacode uitcommentariëren. Druk op de F4-toets om de preview rechts te verversen. In OpenSCAD kun je het resultaat tot slot laten renderen (F6-toets) en exporteren als een STL-bestand voor de 3D-printer. Als het printbed niet groot genoeg is om een rand in zijn geheel uit te printen, moet hij in stukken worden gedeeld zoals hierboven beschreven. Bij het afwerken van de prints kun je de stukken met een verwarmde spatel aan elkaar smelten. Gewoon lijmen met seconden- of tweecomponentenlijm werkt ook, maar test dat eerst op een onopvallende plaats.

Het ontwerp van de voet is ingewikkelder omdat we de kubus op een punt willen zetten. Een dubbele rotatie van 45 graden in twee richtingen is niet voldoende om de kubus in de juiste positie te brengen. In plaats daarvan werken we met een boogtangens, waarbij we gewoon op internet naar de juiste formule hebben gezocht. Omdat een echt parametrisch ontwerp te complex

zou zijn geweest, is ook het schalen naar andere maten beperkt. Onze sjablonen en enkele andere tips voor het maken van je eigen ontwerpen vind je via de downloadlink.

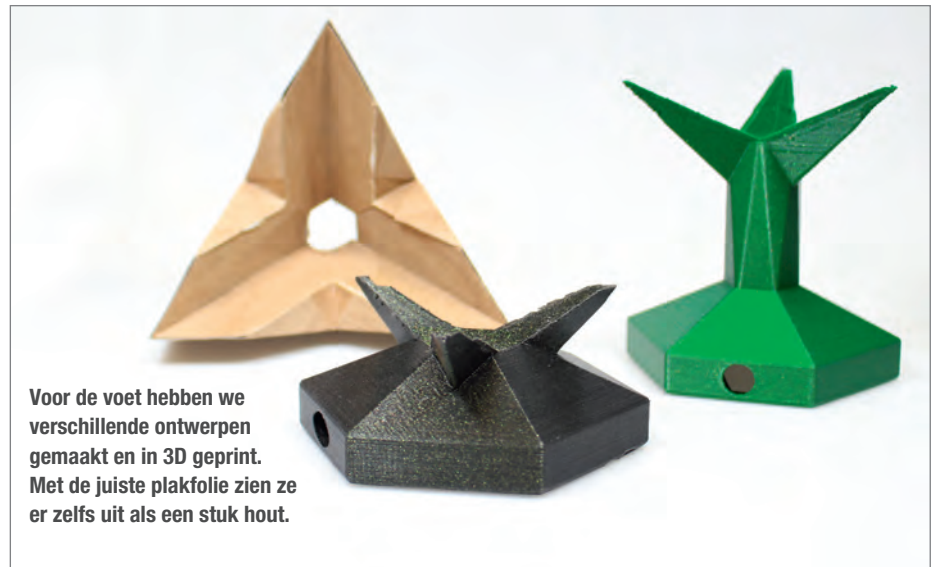
Microcontroller aansluiten

Als brein van het geheel gebruiken we de wifi-microcontroller ESP32 in de NodeMCU-versie. In principe is een Arduino Nano voldoende, maar we willen het project later uitbreiden en de netwerkverbinding van de ESP gebruiken. Eerst sluiten we de data-ingangen en -uitgangen van de afzonderlijke strips aan elkaar aan. De volgorde van de strips is voor nu niet belangrijk. VCC en GND van de strips sluiten we ook aan elkaar aan. Vervolgens solderen we de data-ingang van de eerste strip aan de ESP32 (in ons geval aan pin 17), daarna VCC en GND van de strips aan Vin en GND van de ESP-module. Als je elke strip afzonderlijk wilt aansturen, moet je deze op aparte pinnen van de controller aansluiten. Om de ESP in de voet te plaatsen, hebben we de gesoldeerde pin headers nog omgebogen. Natuurlijk zou het nog ruimtebesparender zijn om een bordje zonder pin-

headers te gebruiken.

Over pinnen gesproken: De data-uitgangen van de ESP32 leveren slechts 3,3 volt, wat officieel te weinig is voor de WS2812. Maar als je de ESP NodeMCU via een usb-aansluiting gebruikt, levert de Vin-pin 5 volt voor de voeding. Daarmee kunnen we deze kubus in principe laten werken, maar erg betrouwbaar ging dat niet. Voor het gebruik met een Arduino raden we dan ook een externe voeding aan die ook voldoende stroom (ampère) levert.

Om de kubus te laten oplichten, gebruiken we de Arduino- programmeeromgeving en de FastLED-bibliotheek, waar mooie voorbeeldprogramma's bij zitten (zie link in de Korte info). Naast de WS2812-leds ondersteunt hij ook veel andere ledstrips en -pixels zoals LPD8806, Neopixel en andere. Als je de bibliotheek nog niet gebruikt, kun je deze via het bibliotheekbeheer eenvoudig downloaden. Ga hiervoor in de programmeeromgeving naar *Hulpmiddelen/Bibliotheeken beheren*. Typ vervolgens FastLED in de zoekregel rechtsboven en installeer de bibliotheek van Daniel Garcia. Wanneer je de muisaanwijzer over de kop van het tekstveld beweegt, verschijnt de knop *Installeren*.



Voor de voet hebben we verschillende ontwerpen gemaakt en in 3D geprint. Met de juiste plakfolie zien ze er zelfs uit als een stuk hout.

In tegenstelling tot de Arduino-boards wordt de ESP32 van huis uit niet door de programmeeromgeving ondersteund. Om de controller te kunnen gebruiken, moeten we de juiste driver installeren via het Board Beheer (zie de link in de korte info). Nu kunnen we eindelijk onder *Hulpmiddelen/Board* de juiste

ESP selecteren, in ons geval de NodeMCU-32s. Voor een ander board moet je natuurlijk de betreffende optie kiezen. Als de microcontroller met de computer is verbonden, moet je tot slot de juiste COM-poort onder *Hulpmiddelen/Poort* aanklikken. Gebruik je niet de nieuwste Arduino-versie,

Boek en Hardware Bundels



www.elektor.nl



Bekijk **alle** Elektor Bundels



We hebben de drie kabels gecombineerd in een Dupont-stekker.

dan kan het zijn dat je problemen ondervindt met het integreren van de bibliotheek of de ESP32, en ook de keuze van de COM-poort loopt niet altijd soepel. Onder de link in de Korte info vind je instructies die in deze gevallen kunnen helpen.

FastLED gebruiken

Als je direct aan de slag wilt, vindt je een aantal voorbeeldprogramma's onder *Bestand/Voorbeelden/Voorbeelden van Custom Libraries/FastLED*. Om de belangrijkste functies te begrijpen, zijn de voorbeeldbestanden "Blink" en "FirstLight" nuttig, die uitleg van de programmacode in de commentaren bevatten. Bijzonder indrukwekkend is het voorbeeld "ColorPalette" 8. Om het te kunnen gebruiken voor onze kubus, voeren we eerst

het nummer van de gebruikte pin (17) in bij LED_PIN. NUM_LEDS geeft het aantal leds aan, in ons geval 360. De helderheid kan worden ingesteld van 0 tot 255, waarbij de standaardwaarde van 64 voldoende is om mee te beginnen. Tot slot moet onder LED_TYPE de gebruikte ledstrip (zoals onze WS2812-leds) worden aangegeven. Een overzicht van de ondersteunde leds is te vinden in de FastLED-documentatie en in het genoemde Blink-voorbeeld. Nu kun je het aangepaste ColorPalette-bestand uploaden naar de ESP via de knop *Uploaden*. Daarna moet de kubus helder oplichten.

Vergeet bij het schrijven van eigen programma's niet om de bibliotheek in de eerste regel van de listing met `#include <FastLED.h>` op te nemen. Met `#define COLOR_ORDER` kun je de volgorde bepalen waarin de waarden voor de primaire kleuren rood, groen en blauw worden aangegeven waaruit de afzonderlijke kleuren samengesteld worden. Naast de RGB-kleurruimte ondersteunt de bibliotheek ook de HSV-kleurruimte, waarin naast de kleurtoon ook verzadiging en helderheid worden gebruikt. Het is ook zinvol om na het starten van het programma een korte vertraging in te bouwen (`delay(2000);`), zodat je de controller opnieuw kunt programmeren als de leds te veel spanning trekken.

Uitbreidingsmogelijkheden

In feite ziet de kubus er met het voorbeeldprogramma al geweldig uit. Natuurlijk zijn er vele uitbreidingsmogelijkheden, het is maar net wat je leuk vindt. Het ligt voor de hand

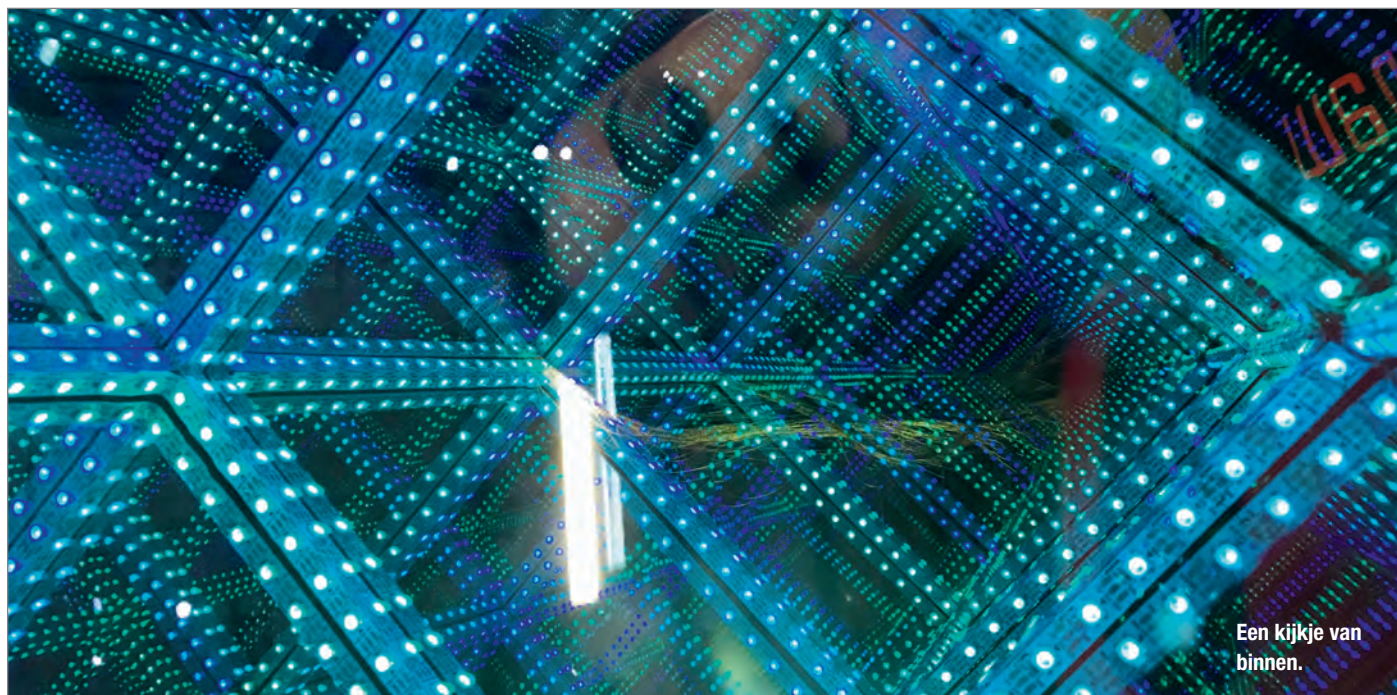
FastLED configureren

```
#include FastLED.h
#define LED_PIN 17
#define NUM_LEDS 360
#define BRIGHTNESS 64
#define LED_TYPE WS2812B
#define COLOR_ORDER GRB
```

8

om een microfoon aan te sluiten om de kubus op de maat van de omgevingsgeluiden te laten oplichten. De ESP32 kan ook via wifi of bluetooth aangestuurd worden. Andere mogelijkheden zijn omgevingsensoren, bijvoorbeeld om de polsslagen van mensen te gebruiken of zelfs de Arduino-Mindflex hack om hersengolven met de kubus te visualiseren. Geef je fantasie de ruimte!

—hch



Een kijkje van binnen.