

Video-output met micro- controllers

composiet
video

p. 6

p. 70

THE TUBE

een ongebruikelijke
buizenversterker

p. 16

ESP32-CAMERA

zo simpel –
heeft niet eens WiFi

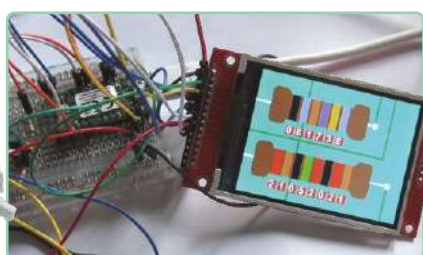
FOCUS ON

Audio & Video



ATX-voeding voor Raspberry Pi:
verschillende spanningen bij
interessant vermogen

p. 20



SDR-radioklokken:
vijf tijdsignalen, zes displays

p. 29



Geluidseffecten met licht:
LDR-synthesizerfilter

p. 98





ONS ASSORTIMENT VAN TECHNICI VOOR TECHNICI

The best part of your project: www.reichelt.com

Voor u alleen het beste - van meer dan 1500 merkfabrikanten

Onze productmanagers zijn al vele jaren bij reichelt actief en kennen de eisen van onze klanten. Zij stellen een breed scala aan kwaliteitsproducten samen, optimaal afgestemd op de behoefte bij onderzoek en ontwikkeling, onderhoud, IT-infrastructuur, kleine serieproducties en makers.

STRUCTURE ENERGY: ENERGIE-EFFICIËNT REGELEN

Shelly

Shelly TRV radiatorthermostaat

Radiatorkraan met regeling via WiFi

Zeer efficiënt verwarmingsbeheer voor het besparen van energiekosten. Door het grote temperatuurbereik kunt u de temperatuur precies op uw behoeften instellen.

- In minder dan twee minuten de oude verwarmingskraan vervangen
- 6500 mAh accu
- Tot maar liefst twee jaar accucapaciteit
- Temperatuurbereik: 5 °C - 30 °C



PRODUCTTIP

Bestelnr.:
SHELLY TRV

81,24
(67,14)



Ontdek ons uitgebreide Shelly-assortiment voor maximaal wooncomfort.

► www.reichelt.com/shelly

SMART. OPTIMAAL. HANDIG.

ENERGIE EFFICIËNT REGELEN

Ontdek nu ►
<https://rch.lt/energy-nl>



reichelt
elektronik **MAGAZIN**

ADVIES:

Efficiënt verwarmen gemakkelijk gemaakt – Deze tips helpen u energie en kosten te besparen

Ontdek nu ►
<https://rch.lt/MG530n>



Betaalwijzen:

DAGPRIJZEN! Prijzenstand: 14. 12. 2022

■ Geweldige prijs-kwaliteitverhouding

■ Meer dan 130.000 geselecteerde producten

■ Betrouwbare levering - vanuit Duitsland over de hele wereld

www.reichelt.com

Bestelservice: +31 85 208 62 94

reichelt
elektronik – The best part of your project

De wettelijke herroepingsregelingen zijn van toepassing. Alle aangegeven prijzen in euro inclusief de wettelijke BTW, excl. verzendkosten voor de totale winkelwagen. Uitsluitend onze Algemene Voorwaarden (zie hiervoor <https://rch.lt/AV-NL> of op aanvraag) zijn van toepassing. Afbeeldingen kunnen afwijken. Drukfouten, vergissingen en prijswijzigingen voorbehouden.
reichelt elektronik GmbH & Co. KG, Elektronikring 1, D-26452 Sande, Tel.: +31 85 208 62 94

Elektor is een uitgave van
Elektor International Media B.V.
Postbus 11, 6114 ZG Susteren, Nederland
Tel.: +31 (0)46- 4389444

Nieuwe abonnementen & bestellingen

service@elektor.nl – Tel. 046-4389444

Elektor International Media B.V. legt gegevens vast voor de uitvoering van de (abonnements) overeenkomst. De door u verstrekte gegevens kunnen gebruikt worden om u te informeren over relevante diensten en producten. Stelt u daar geen prijs op, dan kunt u dit schriftelijk doorgeven aan:

Elektor International Media B.V.,
Afdeling Customer Service
Postbus 11, 6114 ZG Susteren.

Of per email: service@elektor.nl

In overeenstemming met de Wet bescherming persoonsgegevens zijn de verwerkingen van persoonsgegevens aangemeld bij de toezichthouder, Autoriteit Persoonsgegevens te Den Haag.

Druk: Senefelder Misset, Doetinchem
Distributie: Betapress, Gilze

Advertenties Benelux

Raoul Morreau
Tel. +31 (0)6 4403 9907
E-mail: raoul.morreau@elektor.com

Advertentietarieven, nationaal en internationaal, op aanvraag. Alle advertentiecontracten worden afgesloten conform de Regelen voor het Advertentiewezen gedeponeerd bij de rechtbanken in Nederland. Een exemplaar van de Regelen voor het Advertentiewezen is op aanvraag kosteloos verkrijgbaar.

Auteursrecht

Niets uit deze uitgave mag vervaelvoudigd en/of openbaar gemaakt worden door middel van druk, fotokopie, microfilm of op welke wijze dan ook, zonder voorafgaande schriftelijke toestemming van de uitgever. De auteursrechtelijke bescherming van Elektor strekt zich mede uit tot de illustraties met inbegrip van de printed circuits, evenals de ontwerpen daarvoor. In verband met artikel 30 van de Rijksoctrooiwet mogen de in Elektor opgenomen schakelingen slechts voor particuliere of wetenschappelijke doeleinden vervaardigd worden en niet in of voor een bedrijf. Het toepassen van de schakelingen geschiedt buiten de verantwoordelijkheid van de uitgever. De uitgever is niet verplicht ongevraagd ingezonden bijdragen, die hij niet voor publicatie aanvaardt, terug te zenden. Indien de uitgever een ingezonden bijdrage voor publicatie aanvaardt, is hij gerechtigd deze op zijn kosten te (doen) bewerken. De uitgever is tevens gerechtigd een bijdrage te (doen) vertalen en voor haar andere uitgaven en activiteiten te gebruiken tegen de daarvoor bij de uitgever gebruikelijke vergoeding.



Jens Nickel

Hoofredacteur Elektor Magazine

Voor ogen en oren

De mix doet het – en dat is altijd zo geweest bij Elektor. Je kunt evenveel leren van een duik in een schema als van je verdiepen in broncode. In deze editie benaderen we het populaire onderwerp van audio en video – zowel analoog als digitaal, in theorie en in de praktijk. Want niet alleen de ogen en oren willen aan hun trekken komen, ook de grijze massa in ons hoofd wil wat het toekomst.

Ons omslagartikel is bijzonder uitnodigend. Een goedkope besturingskaart, een paar externe componenten en wat open-source software is alles wat je nodig hebt om videosignalen te leveren die op een monitor kunnen worden bekeken. Dit werkt niet alleen met een composiet videosignaal; VGA en HDMI zijn ook mogelijk. Mijn collega Mathias Claussen stelde een artikel samen vol achtergrondkennis, maar het werd zo uitgebreid dat het onze redactionele workflow tot het uiterste op de proef stelde. In plaats van een artikel van 20 pagina's in dit nummer af te drukken, hebben we besloten Mathias' artikel in twee delen te splitsen. Dit nummer laat zien hoe je een composiet signaal uit een kleine 8-bit controller kunt peuteren, in zwart-wit, in grijswaarden of in kleur (pagina 6). In het volgende nummer komen iets sterkere rekentui- gen aan de beurt om VGA-, DVI- en HDMI-signalen te produceren.

Voor gevorderde gebruikers raad ik het artikel van Tam Hanna aan dat het Audio Development Framework van Espressif introduceert met praktische toepassingen zoals een MP3-speler (pagina 88). Ik was verbaasd over wat er op het gebied van audioverwerking mogelijk is met een ESP32 en een paar regels software. Toegegeven, de leercurve is steil. Maar voor iedereen die professionele projecten wil ontwikkelen met spraak- of muziekuutvoer, zal de moeite zich in alle opzichten lonen.

Onze versterker voor 32Ω-koptelefoons werkt zonder microcontroller. Een dual audio-op-amp OPA2134 wordt gecombineerd met twee BUF634A high-speed buffers, die zorgen voor de stroomversterking. Als je wilt, kun je deze IC's vervangen door een discrete schakeling.

Ga aan de slag en leer al doende!

Ons team



Internationaal hoofredacteur: **Jens Nickel** (redactie@elektor.com)

Redactie:

Eric Bogers (redacteur NL-editie), **Rolf Gerstendorf**,
Dr. Thomas Scherer, **Clemens Valens**, **Brian Tristram Williams**

Technische redactie:

Mathias Claussen, **Ton Giesberts**, **Clemens Valens**

Vormgeving & Layout:

Harmen Heida, **Sylvia Sopamena**, **Patrick Wielders**

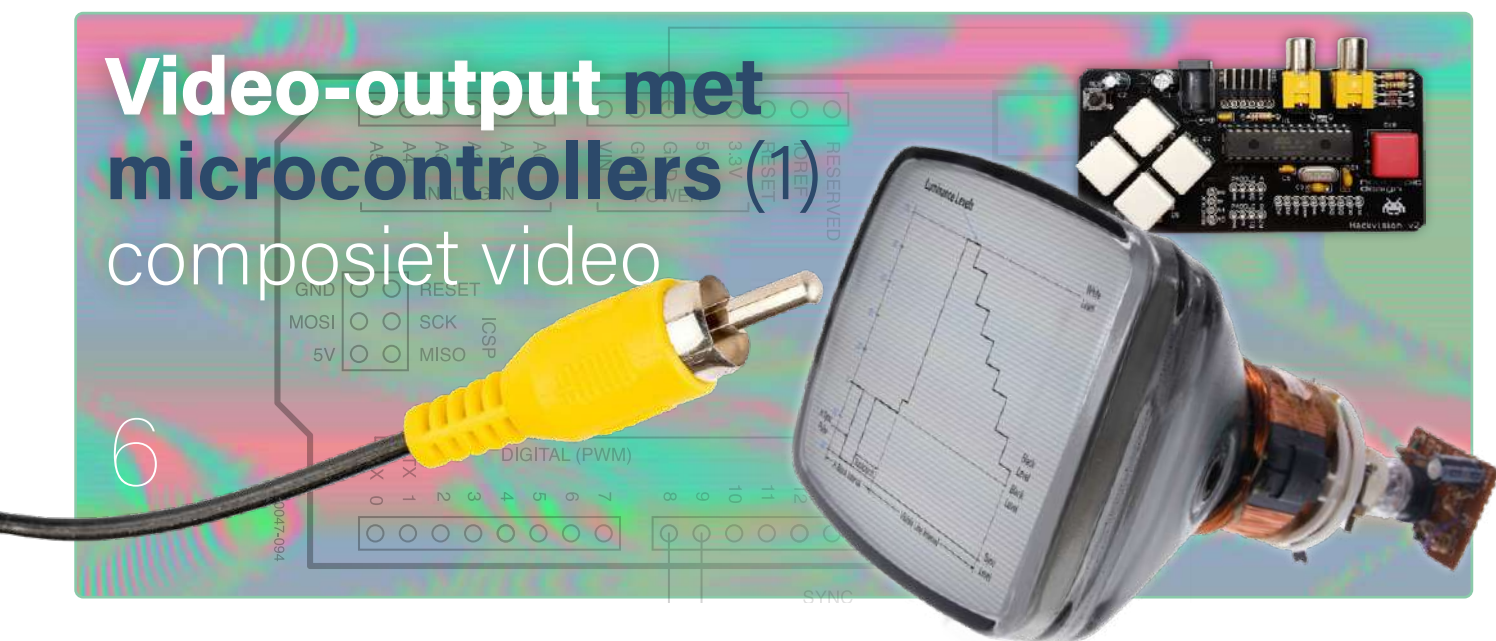
Directeur:

Erik Jansen

DEUTSCHE

FACHPRESSE

Elektor is lid van de VDZ (Verband Deutscher Zeitschriftenverleger), die "de gemeenschappelijke belangen behartigt van 500 Duitse consumenten- en B2B-uitgevers".



Rubrieken

- 3 Voor ogen en oren**
- 34 Alle begin...**
...zet een streep onder de diode
- 38 Uit het leven gegrepen**
over de kwaliteit der dingen
- 52 Developer's Zone**
THD-meting met een oscilloscoop en FFT
- 77 Ethiek in actie**
biomateriaal in elektronica: klaar of niet
- 103 Retrotronica**
de kilowatter van Elektuur
- 106 Oost West Lab Best**
een volumetrisch display in Canada
- 110 Project 2.0**
correcties, updates en brieven van lezers
- 114 Hexadoku**
puzzelen voor elektronici

Achtergrond & info

- FOCUS**
6 Video-output met microcontrollers (1)
composiet video
- 14 electronica 2022**
nieuws van 's werelds grootste elektronica-beurs

- 47 MakePython ESP32 Development Kit**
alles in een doosje
- 80 De Opera Cake antenneswitch voor HackRF One**
sluit tot acht antennes aan op uw SDR
- 82 Q&A met Ashwin Pajankar**
engineering met de Arduino, en meer
- FOCUS**
88 Audiosignalen en de ESP32
de ESP-ADF-omgeving in de praktijk

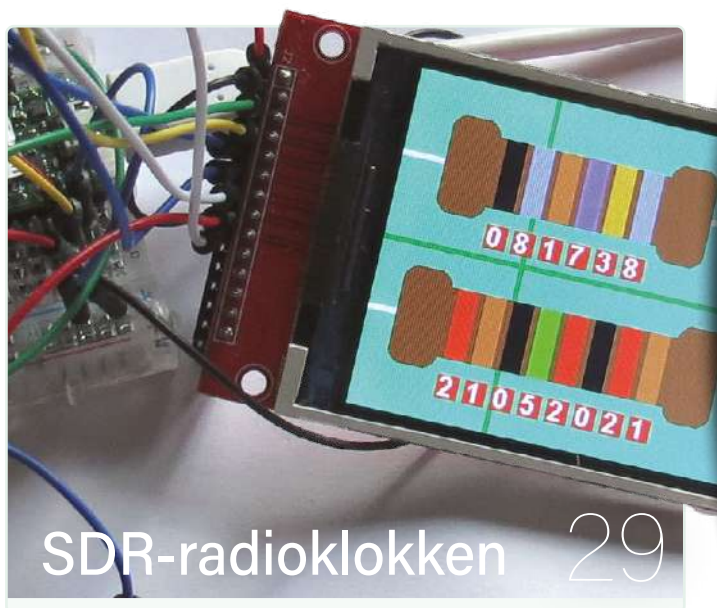
Industry

- FOCUS**
56 Allesziende machines
de technologie achter moderne industriële visionsystemen
- FOCUS**
60 Elektor infographics
feiten en cijfers
- FOCUS**
62 De evolutie van spraak- en audiogestuurde elektronische apparaten
- 66 WEEF 2022 terugblik**
- 68 FFWD electronica 2022 terugblik**
de vernieuwers hebben niet nagelaten indruk te maken

The Tube
een ongebruikelijke
buisenversterker



70



SDR-radioklokken 29



Licht gebruiken voor geluidseffecten 98

Projecten

FOCUS

16 ESP32-camera

zo simpel – heeft niet eens WiFi

20 ATX-voeding voor Raspberry Pi

FOCUS

26 32Ω-koptelefoonversterker

eenvoudige maar hoogwaardige 3-chip oplossing

29 SDR-radioklokken

vijf tijdsignalen, zes displays

40 Reverse-engineering van een Bluetooth Low Energy LED-badge

een BLE-apparaat besturen met een Python-script

FOCUS

70 The Tube

een ongebruikelijke buizenversterker

85 LiDAR-precisiemeter

meet tot 12 meter

FOCUS

98 Licht gebruiken voor geluidseffecten

op LDR gebaseerd spanningsgestuurd 24 dB/oct synthesizerfilter

Binnenkort

Elektor maart/april 2023

We hebben natuurlijk weer onze gebruikelijke mix van projecten, schakelingen, achtergrondinformatie en tips en trucs van en voor elektronici en makers voor u in de aanbieding.

De focus ligt daarbij op embedded en AI.

Een greep uit de inhoud:

- > Wekker 2.0
- > Video met microcontrollers: VGA, HDMI, DVI
- > Raspberry Pi Pico: PIO in de praktijk
- > Het programmeren van spraakgestuurde IoT-applicaties
- > De Android WiFi-API
- > Mijn eerste software defined radio
- > Armeluis chiptweaker
- > Echte USB-toevalsgenerator

...en nog veel meer!

Aankondigingen onder voorbehoud.

Elektor maart/april 2023 verschijnt omstreeks 16 maart 2023.

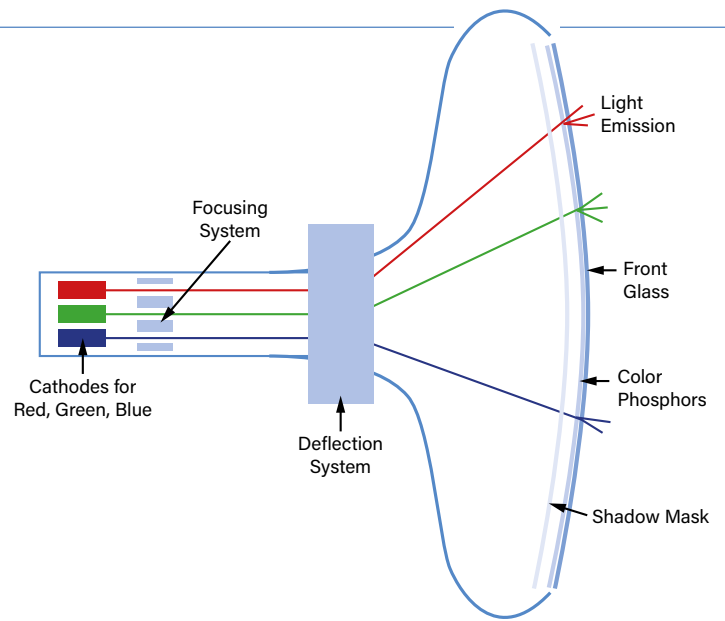
47

MakePython ESP32 Development Kit
alles in een doosje





Figuur 2: Geopend tv-toestel met kathodestraalbuis. (Bron: Shutterstock / Sergio Sergio)



Figuur 3: Schematische opbouw van een kleurenbeeldbuis. (Bron: ITWissen.info)

het PAL formaat is weergegeven in **Figuur 1** als voorbeeld van deze standaarden.

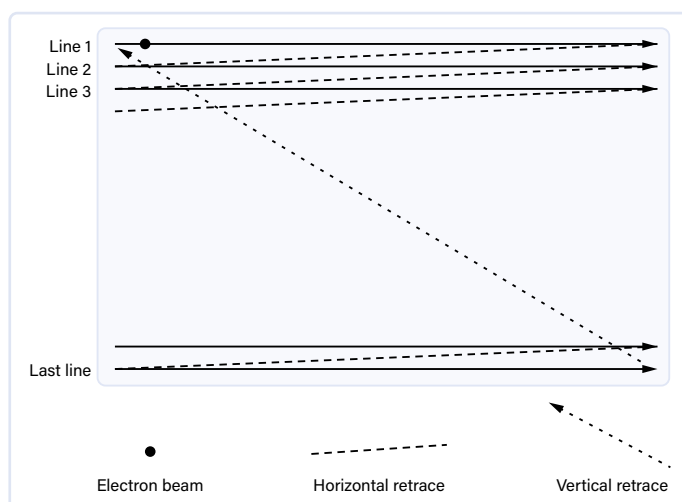
Deze norm was vooral geschikt voor beeldweergave met beeldbuizen (**Figuur 2**). Het werkingsprincipe van zulke beeldbuizen verschilt enorm van LCD's of OLED-schermen. In een beeldbuis wordt een gemoduleerde, horizontaal en verticaal afgebogen elektronenbundel op een fosforlaag gericht. Op het trefpunt gloeit de fosfor dan in verhouding tot de intensiteit van de elektronenbundel (**Figuur 3**). Tijdgesynchroniseerde horizontale en verticale afbuiging levert een tweedimensionaal beeld op.

Vanuit het perspectief van de kijker is een beeld (zwart-wit) opgebouwd in lijnen van links naar rechts en de lijnen vervolgens van boven naar beneden (**Figuur 4**). De eerste "pixel" per beeld bevindt zich dus linksboven. Deze logica is in moderne digitale beeldschermen grotendeels gehandhaafd.

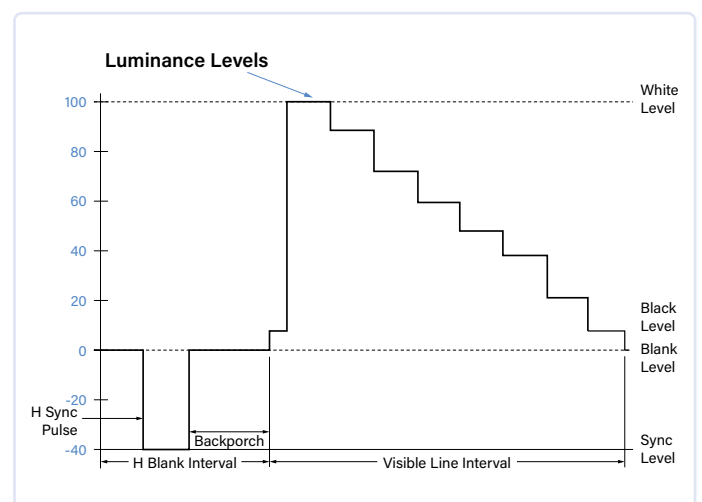
De historische kennis over beeldgeneratie door middel van een bewegende elektronenbundel maakt het gemakkelijker om moderne videosignalen te begrijpen. Het signaal in Figuur 1 bevat de structuur van een beeldlijn. In het voorste gedeelte (**Figuur 5**) van het signaal is de "horizontale blanking"

te zien. Aan het begin van de "horizontale blanking" bevindt de elektronenbundel zich aan de rechterrand van de beeldbuis en moet daarom voor de volgende lijn terug naar uiterst links. Tijdens deze sprong van rechts naar links wordt de elektronenbundel onderdrukt door het helderheidssignaal op zwart of zelfs "zwarter dan zwart" te zetten.

Horizontale blanking duurt $12\ \mu\text{s}$ voor PAL en $10,9\ \mu\text{s}$ voor NTSC. Het bestaat uit drie delen, het "voorportaal", de "sync dip" en het "achterportaal". Het helderheidsniveau van het voorportaal (PAL = $1,65\ \mu\text{s}$ / NTSC = $1,4\ \mu\text{s}$) ligt op of iets onder de waarde van zwart bij $0,3\ \text{V}$,



Figuur 4: Op lijnen gebaseerde beeldopbouw.

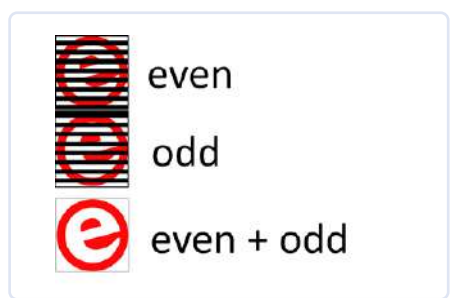


Figuur 5: Horizontale blanking. (Bron: www.edn.com)

terwijl de elektronenbundel nog verder naar rechts buiten het zichtbare beeld komt. De spanning bij de sync-dip (PAL en NTSC = 4,7 μ s) is 0 V en ligt dus aanzienlijk lager dan het zwartniveau. De sync-dip zorgt ervoor dat de elektronenbundel van rechts naar links springt. Het achterportaal (PAL en NTSC = 4,7 μ s) levert de referentie voor de zwartwaarde van 0,3 V voor de lijn. De te tekenen inhoud, d.w.z. de beeldinformatie, begint aan het eind van het achterportaal. Er is dan 52 μ s beschikbaar voor het zichtbare deel van de lijn. Het aantal pixels of sample punten in deze 52 μ s hangt af van het videoformaat. In PAL zijn er meestal 720 zichtbare pixels. Als alle lijnen getrokken zijn en de elektronenbundel rechtsonder heeft bereikt, moet een sprong naar het beginpunt van het beeld (linksboven) worden uitgevoerd. Dit proces wordt in een apart hoofdstuk beschreven.

Synchronisatie en interlacing

De analoge bandbreedte voor het uitzenden van televisiebeelden was voldoende voor slechts 25 beelden per seconde voor PAL en 29,97 beelden per seconde voor NTSC. Voor het menselijk oog resulteert dit in een bewegend beeld, maar het is niet erg vloeiend en geeft een zeer onaangenaam flikkerend effect. Met een snellere beeldsnelheid van 50 of zelfs 59,94 beelden per seconde wordt het flikkeren merkbaar verminderd. De meeste mensen ervaren dit dan als een vloeiende visuele indruk. Omdat de beschikbare radio-bandbreedte beperkt was, werd de interlaced scan technologie geïntroduceerd: in plaats van alle beeldlijnen op een hogere frequentie als volledige frames uit te zenden, worden de lijnen met even of oneven nummers afwisselend op tweemaal de frequentie uitgezonden, dus respectievelijk 50 of 59,94 Hz (**Figuur 6**).



Figuur 6: Frame-opbouw van twee velden.

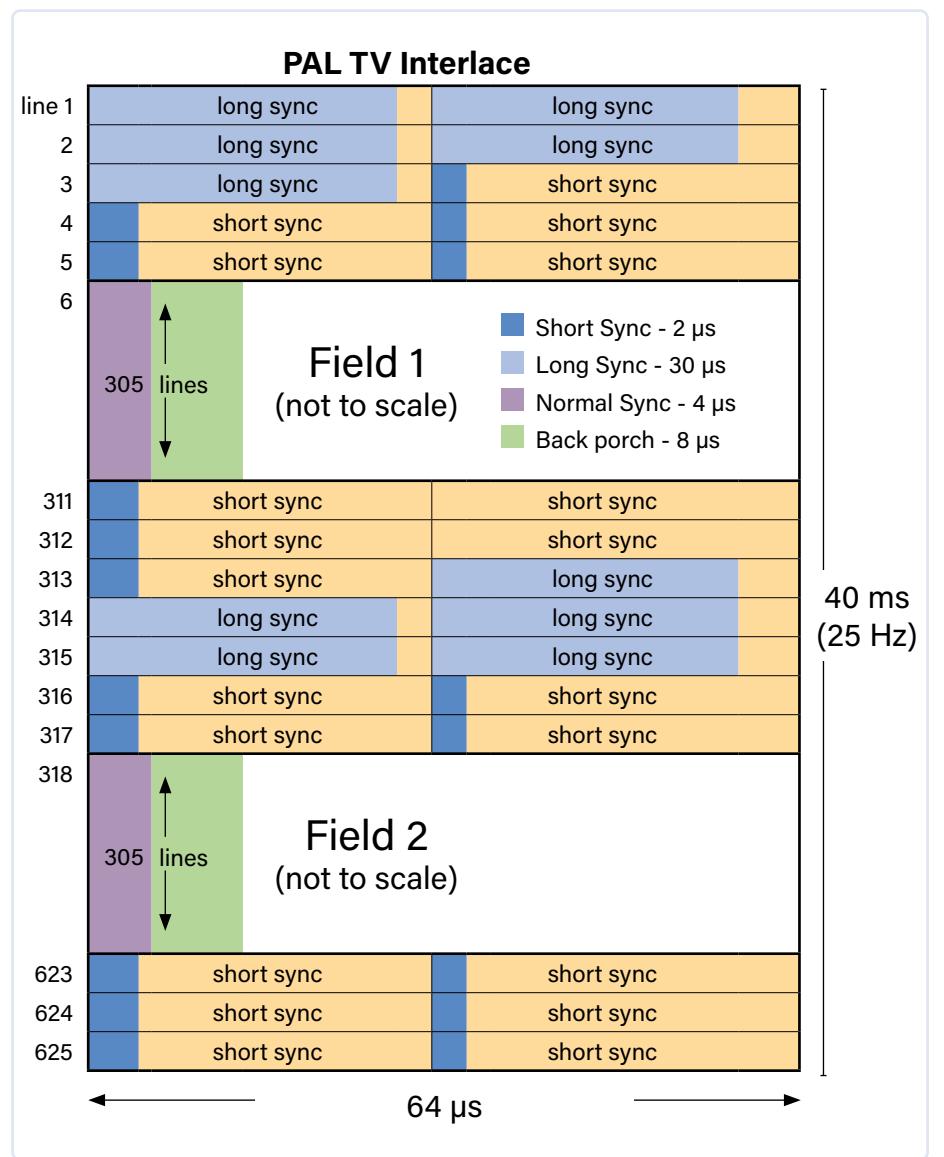
Dus wordt eerst een half frame ("veld") met alle oneven lijnummers uitgezonden, gevolgd door het tweede veld met de even lijnummers. Om de lijnen van de twee velden in de openingen van het respectieve andere veld te laten vallen, is er een detectiesignaal aan het begin van elk veld.

Omdat de lijnen van een compleet frame genummerd zijn volgens de zendvolgorde, heeft het eerste veld in PAL de lijnen 1 tot en met 313 en het tweede veld telt dan door tot en met lijn 625. De verticale synchronisatie voor het einde van een veld en het begin van het volgende is wat lastig: lijnen 311 tot 317 bij de overgang van oneven naar even velden en lijnen 623 tot 5 bij de overgang

van het even naar het oneven veld van het volgende complete frame zijn niet zichtbaar. In deze lijnen worden snellere synchronisatiepulsen met niveaus tussen 0,3 V en 0 V uitgezonden. **Figuur 7** toont de volgorde voor een compleet PAL-frame. Aan de hand van deze synchronisatiepulsen kan de elektronica herkennen of het om het veld met even of oneven lijnummers gaat.

Monochrome video met AVR-controllers

Basisinformatie over composiet signalen zoals CVBS of FBAS kun je vinden op Wikipedia [1]. Als je meer details en vaardigheidjes zoekt, kun je die zeker vinden in het boek *Analogue Video* [2] van Angelo La Spina.



Figuur 7: Samenstelling van een compleet PAL-beeld. (Bron: martin.hinner.info)



Als het videosignaal als VBS signaal (geen kleurinformatie, alleen helderheid) wordt doorgegeven met een geschikte composiet videokabel (**Figuur 8**), kan het genereren van de helderheidsinformatie vrij eenvoudig zijn. Als je alleen wit en zwart gebruikt in plaats van grijsstinten, hoef je slechts twee geschikte spanningen plus synchronisatie uit te zenden. Behalve dat dit de schakeling vereenvoudigt, vermindert het ook het geheugen dat nodig is voor de beeldgeneratie.

De Atmega328P microcontroller van een Arduino Uno kan dus een monochroom beeld genereren met 128x96 pixels. Alle pixels kunnen worden opgeslagen in het interne geheugen van de Arduino Uno, want er zijn slechts 1536 bytes nodig. De bedrading van een Arduino Uno met composietuitgang is weergegeven in **Figuur 9**.

Zoals je ziet zijn twee weerstanden voldoende om een geschikt signaal uit te voeren. Voor de beelduitvoer hoef je het wiel niet opnieuw uit te vinden met je eigen code, maar kun je de TVOut bibliotheek [3] gebruiken. Deze bibliotheek ondersteunt timing voor NTSC en PAL. De video-uitvoer die ermee gedaan kan worden varieert van eenvoudige tekst tot je eigen spelletjes. Hackvision [4] is een platform voor een spelcomputer. Dit is open hardware waarvoor in extreme gevallen alleen een breadboard en een paar onderdelen nodig zijn. Een Arduino kan ook omgezet worden in een Hackvision platform (**Figuur 10**) en zo de bijbehorende bibliotheek van spelletjes gebruiken.

Voor video-output met een AVR microcontroller is één van zijn timers nodig. De horizontale en verticale synchronisatie wordt verzorgd door Timer1 en zijn uitgangspen PB1 (OC1A). Op basis van de timer worden de pixels lijn voor lijn uitgevoerd via een speciale pin (PD7).

De waarden voor de weerstanden bij PB1 en PD7 kunnen vrij eenvoudig worden bepaald. Een composiet ingang op monitoren of TV-toestellen heeft een ingangsimpedantie van 75 Ω . Het niveau voor synchronisatie ligt tussen 0 V en 0,3 V. De spanning voor de helderheidswaarden van de pixels ligt respectievelijk tussen 0,3 V voor zwart en 1 V voor wit. In het volgende wordt aangenomen dat de microcontroller gevoed wordt met 5 V.



Figuur 8: Kabel voor composiet video met RCA-connector. (Bron: Shutterstock / Woodpond)

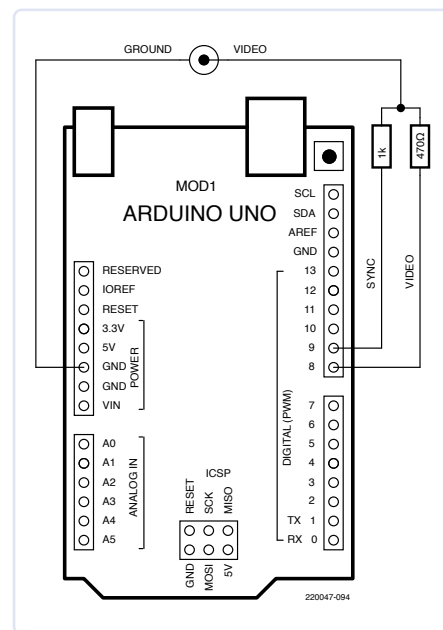
Figuur 11 toont de spanningsdeler voor de video-uitgang bij de typical impedantie van 75 Ω . Pin 9 levert de synchronisatiesignalen. Wiskundig gezien zou een waarde van 1175 Ω nodig zijn voor R1, zodat de spanningsval op R2 0,3 V is met een hoog niveau op pin 9. 1 k Ω levert maximaal 0,34 V op R2, wat nauwkeurig genoeg is voor dit doel. Pin 7 voert de helderheidswaarden uit. Waarden tussen 0,34 V (zwart) en 1 V (wit) zijn nu nodig op R2. Om 1 V bij R2 te bereiken is een waarde van 375 Ω nodig voor de parallelle verbinding van R1 en R3, wat zou resulteren in 600 Ω voor R3. De volgende grotere waarde van 470 Ω uit de E-12 reeks zorgt ervoor dat de spanning op R2 nooit hoger kan worden dan de waarde van 1 V.

Het feit dat dit werkt met een Arduino Uno of een ATmega328P bewijst dat weinig rekenkracht en weinig geheugen volstaat om afbeeldingen op een scherm weer te geven. Omdat het complete beeld in het interne geheugen kan worden bewaard, is het tekenen van een nieuw beeld niet tijdskritisch, alleen het genereren van de videosignalen met behulp van Timer1.

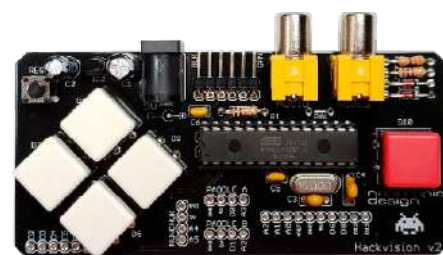
Is het ook mogelijk om met de Arduino grijsstinten = meerdere helderheidswaarden uit te voeren? Dat is mogelijk, maar het interne geheugen is de beperkende factor. Met 16 grijsstinten (4-bits grijsstinten) en geschikte weerstanden zijn 6144 bytes nodig voor het videogeheugen als je de resolutie van 128x96 pixels wilt aanhouden en een compleet beeld in het geheugen moet passen.

Raspberry Pi Pico en Composiet

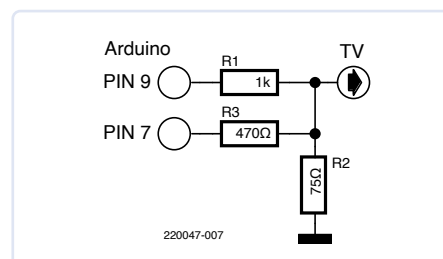
Een Raspberry Pi Pico kan ook een composiet signaal uitvoeren, en ondersteunt gemakkelijker meer dan 50 grijsstinten. Hiervoor wordt een R-2R weerstandsladder [5] gebruikt



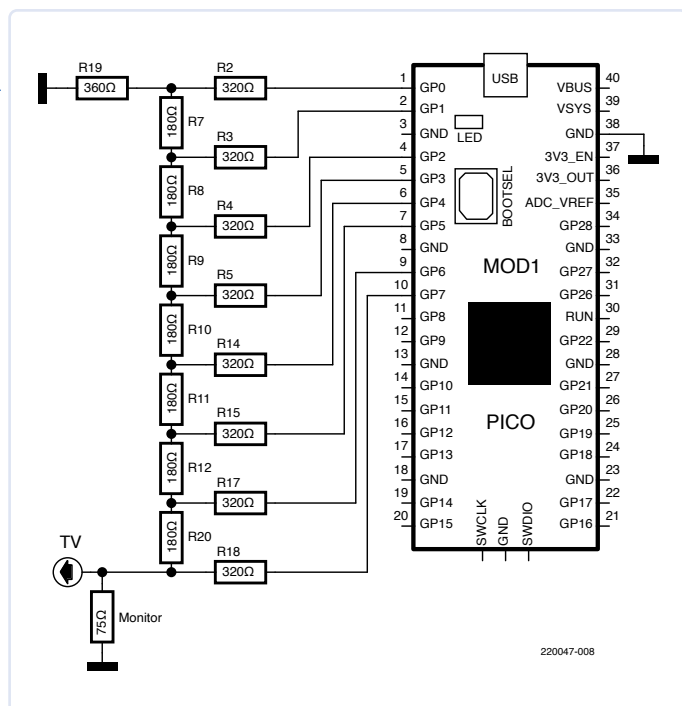
Figuur 9: Z/W video-uitgang met twee weerstanden op een Arduino Uno.



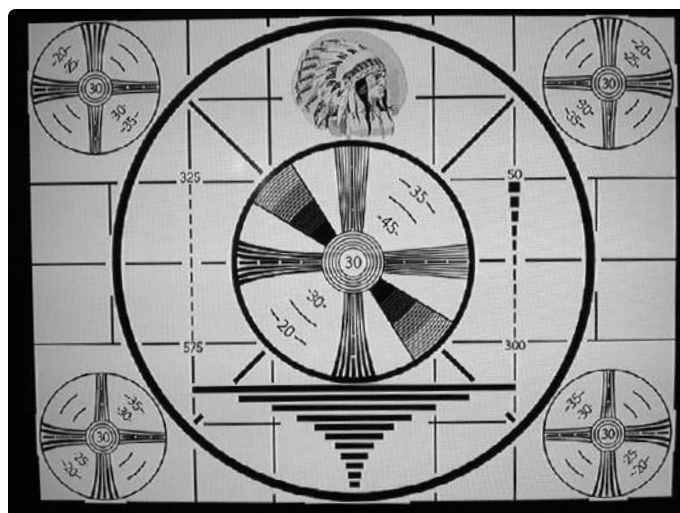
Figuur 10: Hackvision hardware. (Bron: nootropic design)



Figuur 11: De schakeling van de video-uitgang van **Figuur 9**.



Figuur 12: Een R-2R netwerk als DAC op de Raspberry Pi Pico.



Figuur 13: Grijswaardenbeeld van een Raspberry Pi met 512x384 pixels. (Bron: tinyurl.com/2p8z27a2)

als digitaal/analoog converter. **Figuur 12** toont een geschikte schakeling bestaande uit weerstanden van 180, 320 en 360 Ω. Zoals te zien is op de GitHub pagina van het project "pico-composite8" [6], kan de interne weerstand van de GPIO-pinnen hier niet verwaarloosd worden. De ontwikkelaar van het project heeft hiervoor waarden bepaald van ongeveer 40 Ω. De Pico voert hier een composiet signaal uit volgens NTSC. Hij laadt de beelden ofwel uit RAM of Flash en kan maximaal 512x384 pixels uitvoeren. Het resultaat is te bewonderen in **Figuur 13**.

Het project is echter slechts een haalbaarheidsdemonstratie en biedt geen kant-en-klare generieke bibliotheek. Het feit dat een Raspberry Pi Pico grijswaardenvideo met 512x384 pixels kan weergeven laat zien dat het niet zozeer een kwestie is van rekenkracht maar van de juiste timing. Als een compleet beeld in het RAM van de Raspberry Pi Pico wordt bewaard, blijft er slechts ongeveer 64 KB van de 264 KB beschikbaar voor je eigen toepassingen. Maar als je bedenkt dat zelfs een ATmega spellen als Tetris of Pong aankan, inclusief video-uitvoer, zou dit meer dan genoeg ruimte moeten bieden voor je eigen creaties.

Kleur voor Composiet Video

Meer dan 50 tinten grijs is goed, maar kleur is gewoon beter. Als het gaat om composiet video en kleur, wordt het veel moeilijker dan voorheen om een geschikt analoog signaal te genereren. In 2003 demonstreerde Rickard Gunée het genereren van een compo-

siet signaal (PAL of NTSC) op een Hackaday evenement [7] met behulp van een Scenix / Ubicom SX28 op ongeveer 50 MHz.

Maar hoe moeilijk is het om kleur toe te voegen aan een videosignaal? Het antwoord hangt af van de manier waarop kleurinformatie wordt toegevoegd aan het composiet signaal.

Kleuren dankzij PAL en NTSC

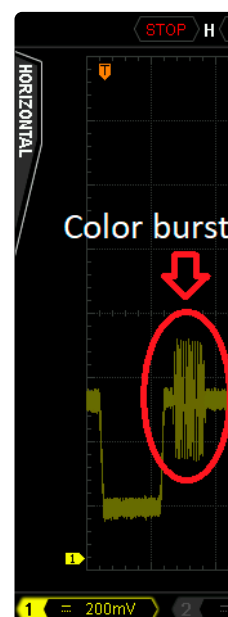
Bij de overgang van zwart-wit naar kleuren-televisie enkele decennia geleden werd geen geheel nieuw signaal gedefinieerd, omdat ervoor gezorgd moest worden dat bestaande zwart-wit televisies het beeld nog steeds konden weergeven. Dit probleem werd internationaal op drie verschillende manieren opgelost, waardoor de normen NTSC, PAL en SECAM naast elkaar kwamen te bestaan. Wat ze gemeen hebben is dat de kleureninformatie werd toegevoegd aan het bestaande monochrome signaal.

Terwijl PAL en NTSC in principe vergelijkbaar zijn (quadratuur modulatie voor kleur), verschilt SECAM in het integreren van kleur door gebruik te maken van frequentie modulatie.

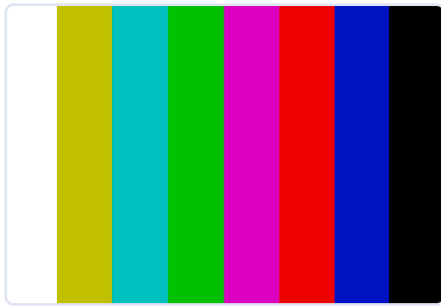
Kleuren die worden geleverd in hun drie basiscomponenten rood, groen en blauw moeten worden omgezet naar een YUV of YCbCr kleurruimte [9] voordat ze kunnen worden uitgevoerd via een composiet videosignaal. De conversie van RGB naar YUV wordt als volgt berekend:

$$\begin{aligned} Y &= 0,299 \times R + 0,578 \times G + 0,144 \times B \\ U &= 0,493 \times (B - Y) \\ V &= 0,877 \times (R - Y) \end{aligned}$$

Dit transformeert de waarden voor R, G en B in een bereik tussen 0 en 1. De formules laten zien dat een implementatie voor microcontrollers ofwel rekenkracht ofwel enige programmeervaardigheid vereist.



Figuur 14: Kleurenburst oscillogram van een PAL-signaal.



Figuur 15: Testpatroon met kleurenbalken.

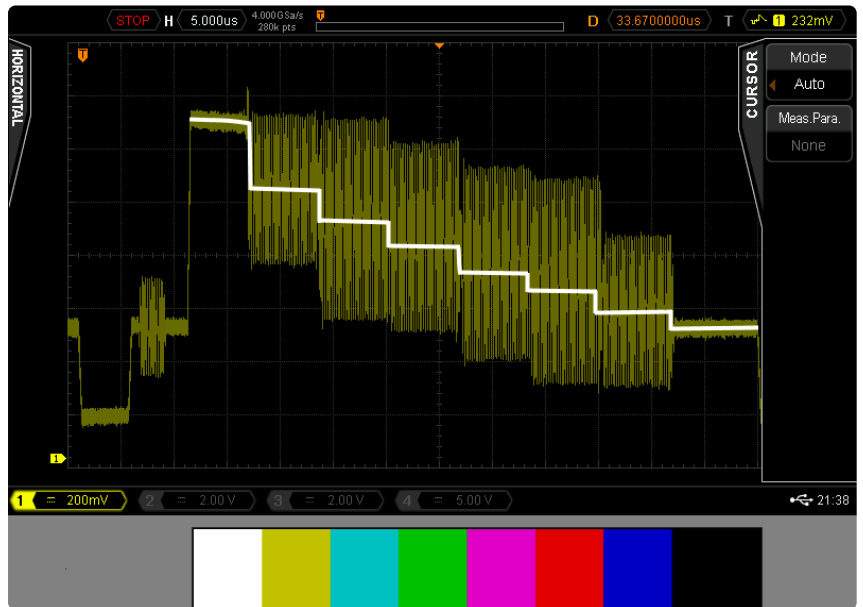
Hieronder volgt een basisuitleg van de implementatie van kleur op basis van PAL en NTSC. **Figuur 1** toont een monochroom PAL-sig-naal waaraan nog geen kleurinformatie is toegevoegd. Voor kleurinformatie gebruikt PAL een subcarrier op 4,43361875 MHz, die gebruikt wordt als referentiesignaal. Deze referentie wordt in elke lijn verzonden als een "color burst" (**Figuur 14**) tijdens de horizontale synchronisatie. Het testpa-troon (**Figuur 15**) wordt gebruikt om uit te leggen hoe de verschillende kleuren van het testbeeld worden gecodeerd. Het testbeeld komt overeen met de EBU kleurenbal-ken [10]; een Raspberry Pi Zero dient als beeldsignaalgenerator.

Het signaal voor één beeldlijn is te zien in **Figuur 16**. De helderheid (witte lijn als gemid-delde waarde van het signaal) en de ampli-tude van de kleurinformatie zijn goed zicht-baar. Er is nog een derde stukje informatie in het signaal, dat verborgen is in de signaalfase. **Figuur 17** toont (in rood) de faseverandering bij de overgang naar een andere kleur.

De informatie voor de helderheid en de ampli-tude van het kleursignaal en de faseverschui-ving zit dus in het signaal. Als er tijdens trans-missie of ontvangst van het signaal een fout optreedt in de fase of de evaluatie daarvan, verandert de kleurindruk van het beeld, wat NTSC ontvangers handmatig konden compenseren door middel van een "tintre-geling" [11] via een draaiknop (later ook met elektronische oplossingen). Bij PAL werd dit probleem vermeden omdat de fase-informatie bij elke tweede lijn 180° wordt verschoven. Een fasefout wordt dus gecompenseerd tussen twee aangrenzende lijnen.

Kleuren Composiet Video met ESP32

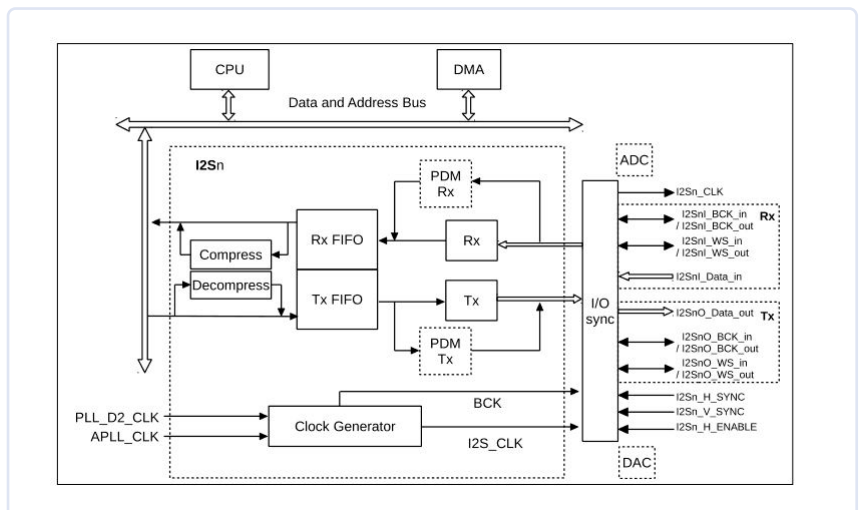
Een ESP32 kan gemakkelijk een monochroom composiet signaal uitzenden met een paar kleine trucjes, en zelfs een Arduino is daartoe in staat. Maar het genereren van een kleuren-composiet signaal stelt aanzienlijk hogere eisen aan de modulatie.



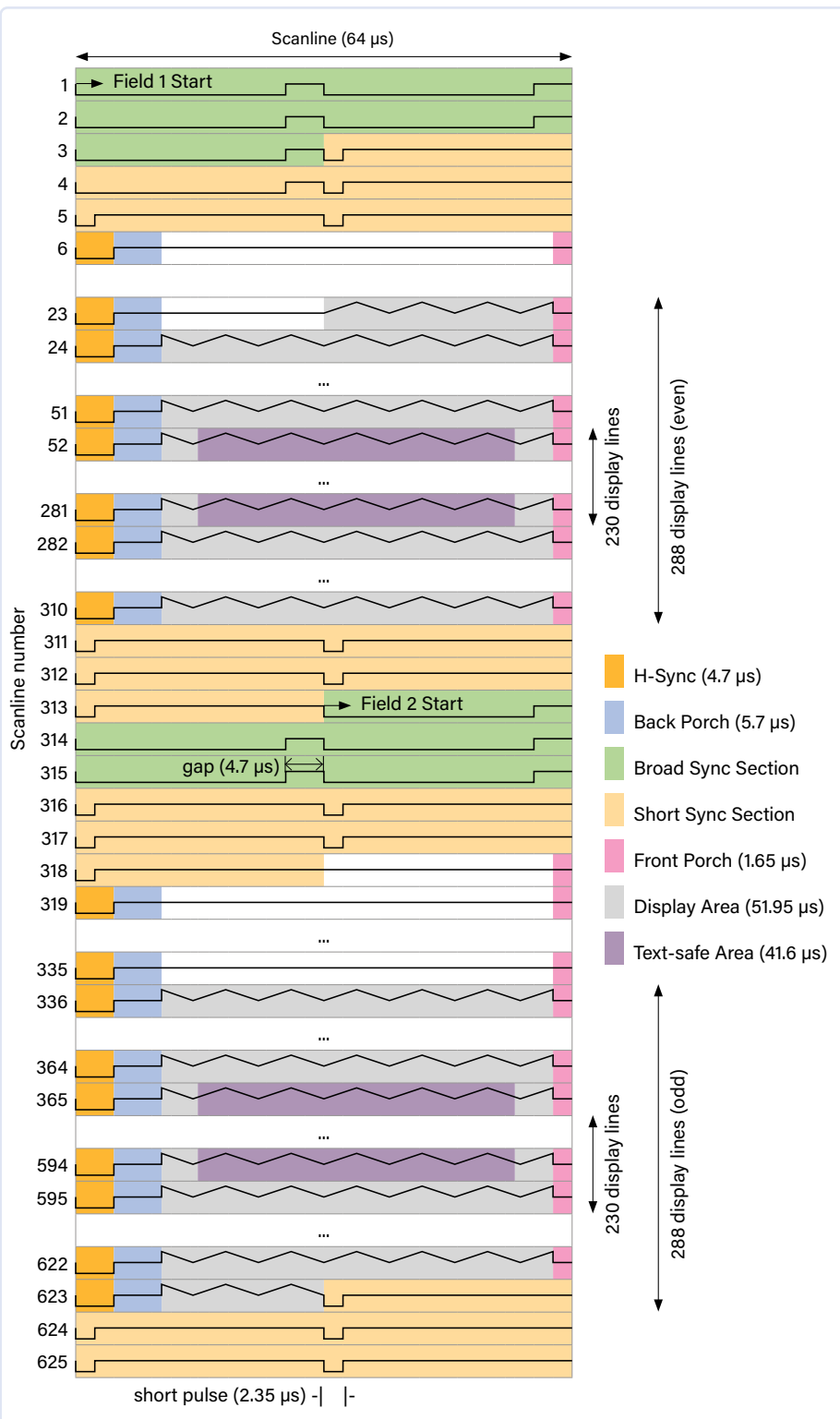
Figuur 16: Oscillogram van een PAL-kleursignaal.



Figuur 17: Faseveranderingen tijdens de kleurverandering zijn gemarkeerd.



Figuur 18: Het I2S-blok van de ESP32. (Bron: Espressif / tinyurl.com/yrrbnjak)

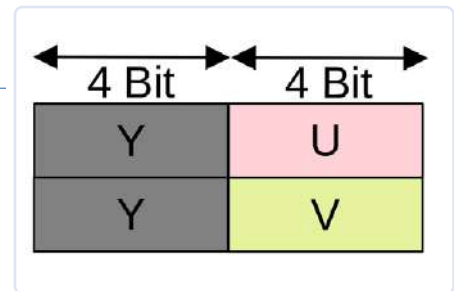


Figuur 19: Signaal van beelduitvoer met halve beeldlijn.
(Bron: batsocks.co.uk / <https://tinyurl.com/4fyhchmk>)

In 2018 demonstreerde bitluni zijn aanpak voor het genereren van een kleurencomposiet signaal met behulp van een ESP32 [12]. Aangetoond werd dat de 13,33 MSa/s van de DAC's van de ESP32 voldoende zijn om een kleurendrager en kleurinformatie in het signaal te integreren. Net als bij het monochrome signaal wordt één processor-kern van de ESP32 gebruikt om het videosignaal te genereren. De opstelling gebruikt het

I2S-blok van de ESP32 om gegevens naar de DAC te sturen (figuur 18).

In PAL en NTSC wordt het beeld interlaced weergegeven, d.w.z. achtereenvolgens in velden met oneven lijnen en even lijnen. Zoals gezegd verdubbelt dit de waargenomen verversingssnelheid (bij dezelfde bandbreedte), wat de waargenomen flikkering aanzienlijk vermindert. De timing van deze twee velden



Figuur 20: Opslag van kleurinformatie door bitluni.

heeft een paar eigenaardigheden, zoals slechts een halve kaderlijn aan het eind van het beeld in een van de twee velden (Figuur 19).

Omdat het RAM van een ESP32 slechts voldoende is om een beeld met de halve resolutie uit te voeren, moeten lijnen tweemaal getekend worden: de inhoud van regel één verschijnt dus ook in regel twee. De interlacing zorgt er dus voor dat dezelfde inhoud tweemaal wordt getekend. Nu rijst de vraag of men werkelijk zowel het oneven veld als het even veld moet uitvoeren, of dat het ook voldoende zou zijn om het even veld tweemaal te gebruiken en zo de code voor de uitvoer aanzienlijk te vereenvoudigen. Precies deze truc werd al gebruikt door het Super Nintendo Entertainment System (SNES). En ook de code van bitluni gebruikt deze methode om de code voor synchronisatie te vereenvoudigen. Zo worden slechts 288 echte lijnen bij 50 Hz uitgevoerd (288p) - voor NTSC zou het equivalent 240 lijnen bij 60 Hz (240p) zijn.

In tegenstelling tot de monochrome beeldweergave moet nu ook de kleureninformatie in het RAM van de ESP32 worden opgeslagen. In het algemeen worden de kleuren weergegeven door hun componenten rood, groen en blauw. Voor het genereren van een standaard composiet signaal kan de kleurinformatie echter beter in het RAM worden opgeslagen als YUV waarden. Dit is de enige manier waarop de ESP32 en zijn DAC deze gegevens snel genoeg kunnen uitvoeren. Omdat de hoeveelheid RAM in de ESP32 beperkt is, zijn enkele trucs nodig bij het coderen van de gegevens. bitluni slaat de informatie in het RAM op als gecombineerde YU-, YV- en V-waarden met elk een 4-bits resolutie (Figuur 20).

Helaas is met de aanpak van bitluni alleen composiet video-uitgang mogelijk volgens de PAL-standaard, met NTSC werkt het niet. De verhouding tussen de 3,579545 MHz van de NTSC color burst en de sample rate van de DAC is zo ongunstig dat er geen bruikbare color burst kan worden uitgevoerd en ontvangers niet kunnen synchroniseren.



Figuur 21: Emulatoren voor 8-bit spelcomputers. (Bron: tinyurl.com/ykd9ezap)

server van het espflix project. Dit vereist echter enkele concessies aan het videomateriaal. Details hierover zijn te vinden op de bijbehorende Github pagina [14].

Deel 2 vooruitblik

Het tweede deel van het artikel gaat over VGA, DVI en sprites. Vooral met een ESP32 of de RP2040 van de Raspberry Pi Pico zijn hier verbazingwekkende effecten mogelijk. Als je niet wilt wachten tot het tweede deel, kun je je inschrijven voor het webinar "Microcontroller als Pixel Artist" [15], waar de onderwerpen VGA, DVI en sprites ook aan bod komen. ◀

220047-03



Webinar "Microcontroller as Pixel Artist"

Omdat bewegende beelden meer zeggen dan woorden, zeker over dit onderwerp, en uitgebreide opsommingen niet in het artikel pasten, kun je het Elektor webinar voor dit nummer bekijken. Je kunt je al inschrijven op [15]. Het zal gaan over graphics met Arduino, ESP32 en Raspberry Pi Pico.
www.elektormagazine.com/webinars

Vragen of opmerkingen?

Heb je technische vragen of opmerkingen over dit artikel? Stuur een e-mail naar de auteur via mathias.claussen@elektor.com of neem contact op met Elektor via redactie@elektor.com.



Gerelateerde producten

- **Raspberry Pi Pico RP2040 (SKU 19562)**
www.elektor.nl/19562
- **ESP32-DevKitC-32D (SKU 18701)**
www.elektor.nl/18701
- **Arduino Uno Rev3 (SKU 15877)**
www.elektor.nl/15877

Kleur met ESP32 in PAL en NTSC

Met zijn project "esp_8_bit" heeft de GitHub gebruiker rossumur bewezen dat het ook mogelijk is om kleur volgens NTSC te maken met een ESP32. Dit project is een verzameling op ESP32 gebaseerde emulaties van verschillende 8-bit consoles, van de Atari 400 tot het Nintendo Entertainment System en het Sega Master System (Figuur 21).

Het project gebruikt eenvoudigweg één pin van de ESP32 om een composiet signaal in NTSC of PAL uit te voeren. De truc zit verborgen in de audio PLL van de ESP32. Deze kan worden gebruikt om de DAC te laten werken met sample rates tot ongeveer 20 MHz. Vier maal de frequentie van de NTSC kleurendrager is 14,318182 MHz (voor PAL 17,734475 MHz). Met de APLL kunnen 14,318180 MHz en 17,734476 MHz worden gegenereerd, wat dicht genoeg ligt bij de benodigde frequenties voor PAL en NTSC. Op deze manier geeft de DAC

nu een geheel veelvoud van de kleurendragerfrequentie uit, wat resulteert in verwerkbare videosignalen.

Met de APLL kan de DAC van de ESP32 niet alleen een videosignaal uitvoeren, deze aanpak is ook zeer interessant voor andere Directe Digitale Synthese (DDS) toepassingen. Het gebruik van de APLL heeft echter een nadeel bij hoge samplesnelheden. De DAC van de ESP32 heeft twee kanalen. Als de APLL één daarvan voorziet van gegevens voor video-uitgang en men probeert het tweede kanaal te voorzien van audiogegevens uit de I2S interface, dan krijgt de DAC last van dropouts. Deze storingen treden op beide kanalen op, zodat je te veelvuldigheid moet nemen tot andere methoden voor audio-uitvoer.

Op basis van het "esp_8_bit" project is "espflix" ontwikkeld. Hiermee kan de ESP32 video's afspelen die zijn opgeslagen op een

WEBLINKS

- [1] Composite Video: https://en.wikipedia.org/wiki/Composite_video
- [2] Angelo La Spina, "Analogue Video": <https://www.elektor.com/analogue-video-e-book>
- [3] Arduino TVOut Library: <https://github.com/Avamander/arduino-tvout>
- [4] Hackvision: <https://nootropicdesign.com/hackvision/>
- [5] Resistor ladder: https://en.wikipedia.org/wiki/Resistor_ladder
- [6] pico-composite8: <https://github.com/obstruse/pico-composite8>
- [7] Hackaday event:
<https://hackaday.com/2022/08/17/chips-remembered-the-scenix-ubicom-parallax-sx>
- [8] Color with SX Chips: <https://elinux.org/images/e/eb/Howtocolour.pdf>
- [9] YCbCr color space: <https://en.wikipedia.org/wiki/YCbCr>
- [10] EBU color bars: <https://en.wikipedia.org/wiki/YUV>
- [11] NTSC tint control: https://en.wikipedia.org/wiki/Tint_control
- [12] bitluni, "ESP32 Composite Video": <https://bitluni.net/esp32-composite-video>
- [13] Github user rossumur: <https://github.com/rossumur>
- [14] espflix: <https://github.com/rossumur/espflix>
- [15] Webinar "Microcontroller as Pixel Artist":
<https://www.elektormagazine.com/webinars>



electronica 2022

nieuws van 's werelds grootste elektronikabeurs

Stuart Cording en Jens Nickel

Dit noem je een compleet succes: ongeveer 70.000 bezoekers kwamen afgelopen november naar Electronica. Soms kon het behoorlijk druk zijn, tenminste bij de stands van de grote distributeurs en halfgeleiderfabrikanten. En op 's werelds grootste beurs in zijn soort namen ze geen halve maatregelen – in plaats daarvan kwamen ze met extravagante stands en aantrekkelijke presentaties. Er waren motorfietsen met ingebouwde AI om je aan te vergapen, speciale busjes voor het opsporen van drones, veel racewagens, en ook een quadcopter-simulator voor één persoon. Hier een kleine selectie van de interessante nieuwe technologie die we te zien kregen.



Aaronia, specialist voor (RF) meettechnologie, toonde dit busje voor de detectie van drones. Het volledig uitgeruste voertuig is verkrijgbaar voor ongeveer 1,5 miljoen euro, maar er zijn ook veel goedkopere oplossingen voor het detecteren van onbemande vliegtuigen verkrijgbaar als rack-mounted apparatuur.
<https://drone-detection-system.com>



Digitale microscopen vervangen steeds vaker optische exemplaren. Ze zijn gebruiksvriendelijk, zelfs voor inspectiepersoneel dat niet vertrouwd is met microscopie, en maken het mogelijk het materiaal automatisch te scannen. Het hier getoonde Olympus-model produceert zelfs 3D-beelden die vanuit alle hoeken bekeken kunnen worden.
<https://olympus-ims.com/en/microscope/dsx1000/high-end-model>



Terwijl virtual reality worstelt om mainstream te worden, biedt augmented reality (AR) tastbare voordelen voor de industrie. Technici kunnen toegang krijgen tot instructies of begeleiding die voor hun ogen worden geprojecteerd terwijl ze een machine repareren, terwijl ze toch volledig zicht hebben op hun omgeving. De kleurenlasermodule van TDK, ingebouwd in enkele prototypes van AR-brillen, straalt kleurenbeelden rechtstreeks op het netvlies - en we hebben het overleefd om over onze ervaring te vertellen!
www.tdk.com/de/news_center/press/20221013_01.html

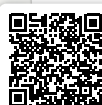




Dit was zeker het hoogtepunt op de stand van halfgeleiderfabrikant Infineon: wie een beetje geduld had, kreeg de kans om een virtuele vlucht te wagen met deze 1-persoons quadcopter-simulator.
<https://infineon.com/cms/en/product/promopages/electronica/>



Epaper-displays zijn in opkomst omdat ze onder elke kijkhoek een scherp beeld bieden. Distributeur Beck Elektronik toonde deze exemplaren, groter dan A4, in monochroom en kleur. Vergeleken met LCD's zijn de displays echter (nog) duur. Bovendien hebben ze geen eigen lichtbron, net als bedrukt papier.
<https://beck-elektronik.de/en/products/displays/e-paper-display-epd>



De auto-industrie zal steeds meer vertrouwen op software om de mogelijkheden van hun nieuwe voertuigen te definiëren met behulp van gestandaardiseerde, programmeerbare hardware. NXP lanceerde zijn S32K39, die twee speciale motorcontrollers bevat. Ze ondersteunen 100kHz-besturingscircuits en zijn klaar om gecombineerd te worden met apparaten met een grote bandbreedte, zoals siliciumcarbide, om zeer efficiënte tractieomvormers voor elektrische voertuigen te leveren.
<https://elektor.link/NXPS32K>



OKdo is een dochteronderneming van RS Components, die boards en kits aanbiedt voor makers en professionals. De single-board computers van het merk ROCK zijn goed uitgerust, krachtig en (meestal) goed verkrijgbaar. De nieuwe Rock 5 Model B heeft een 8K video-uitgang, die werd gedemonstreerd met een bewakingstoepassing.
<https://okdo.com>
<https://wiki.radxa.com/Rock5>



Range anxiety is een belangrijk punt van zorg voor consumenten die overstappen van auto's op fossiele brandstoffen naar elektrische auto's. In de hoop die angst weg te nemen, kondigde Mercedes-Benz onlangs aan dat hun VISION EQXX een traject van 1.202 km (747 mijl) heeft afgelegd op één acculading. Op de stand van onsemi maakt het voertuig gebruik van hun VE-Trac SiC-modules in de tractieconverteer om 95% van de energie van de batterij naar de wielen te leiden.
<https://elektor.link/onsemiSiC>



De Elektor-redactie heeft tijdens Electronica 2022 in München verschillende persoonlijke interviews gehouden met technici en vooraanstaande personen uit de industrie. Bezoek het Elektor TV-kanaal om alle video's te bekijken: www.elektor.tv

220652-03

ESP32-camera

zo simpel – heeft niet eens WiFi



Bera Somnath (India)

Heb je een idee voor een cameratoepassing? Probeer het eerst zelf te doen voordat je een nieuwe kant-en-klare camera koopt. Met een ESP32-board met camera en enkele extra componenten bouw je een eenvoudige maar effectieve oplossing op maat.



Alle smartphones kunnen foto's maken, opslaan en weergeven. Het bouwen van nog een camera is dan ook eigenlijk onzin, tenminste voor mij. Tenzij het echt eenvoudig is. Met een paar goedkope modules zoals een ESP32-board met camera, een klein OLED-display, een realtime-klok en een triggermechanisme kunt je je eigen batterijgevoede camera bouwen die foto's maakt die je volledig onder controle hebt. Door op een knop te drukken of door aanwezigheidsdetectie met een bewegingssensor of zelfs getriggerd door een contactloze temperatuursensor, maakt hij (onbemerkt) een foto van iemand wiens lichaamstemperatuur te hoog is. Zulke toepassingen liggen voor de hand omdat je je smartphone niet op deze manier kunt gebruiken.

ESP32-Cam

Door een PIR-sensor aan het ESP32-Cam-board toe te voegen, kun je het zo aanpassen dat het onbemerkt foto's van indringers maakt! Of je kunt met alleen de ESP32-plus-camera en het OLED-display een handige kleine webcam bouwen die verbinding maakt met een van de beschikbare netwerken in de buurt en de videostream vervolgens op het intranet publiceert. Door de port forwarding-functie van je modem of router te gebruiken, kun je de afbeelding ook op internet publiceren. Deze functie is erg handig voor diegenen die een tweede camera nodig hebben voor demodoeleinden van

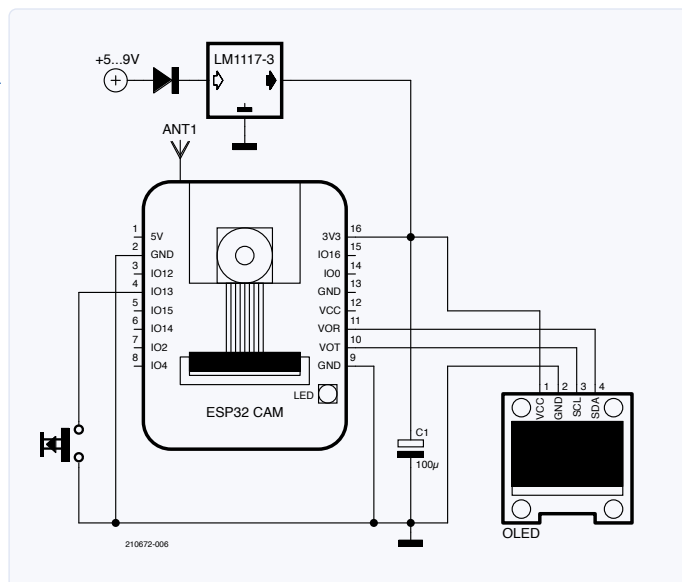
hun webinars en/of online-lessen. De kosten van de ESP32-Cam-module die we in dit artikel gebruiken bedragen ongeveer € 10.

Eerst een coole demo

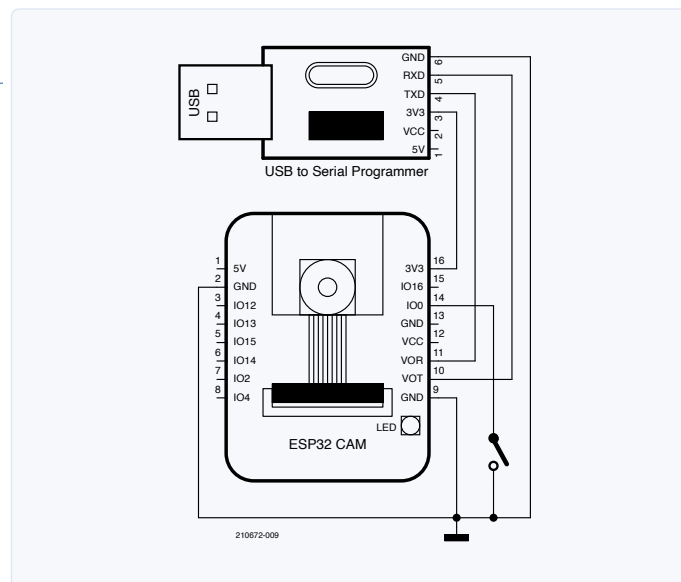
Na het installeren van de nieuwste Arduino IDE en het meest recente ESP32-package met behulp van de Boards Manager van de IDE, selecteer je als board het "AI Thinker ESP32-CAM" board (Tools -> Board -> ESP32 Arduino) en de poort waarop het is aangesloten (Tools -> Port). Open vervolgens het ESP32-webcam voorbeeldproject (File -> Examples -> ESP32 -> Camera -> Webserver). Probeer het uit; het zal je verrassen! De beide onderstaande projecten borduren hierop voort.

Project 1: webcam met OLED-display

Het webcamproject maakt het niet mogelijk om per menu een WiFi-netwerk te selecteren (de netwerk-referenties zijn hardgecodeerd in het programma) en de enige manier om het IP-adres van de webcam te weten te komen, is door deze aan te sluiten op een PC met een actieve seriële terminal. Hier breiden we dit project zodat het verbinding kan maken met elk beschikbaar netwerk in de buurt. Zodra het is verbonden, wordt op het kleine OLED-display weergegeven met welk netwerk het is verbonden en wat het IP-adres is. Als de verbinding om de een of andere reden wordt verbroken,



Figuur 1. Voeg een OLED-display toe aan de ESP32-Cam voor wat extra gemak.



Figuur 2. De ESP32-Cam wordt niet geleverd met ingebouwde USB/serieel-interface. Om de sketch te laden heb je daarom een USB/serieel-adaptor nodig. Om naar de programma-laadmodus te gaan, moet de resetknop van de ESP32 worden ingedrukt (of de voeding moet worden uit- en ingeschakeld) terwijl IO0 naar massa wordt getrokken.

wordt opnieuw geprobeerd verbinding te maken met een van de bekende netwerken.

Bouw van de webcam

Figuur 1 toont hoe je het OLED-display aansluit op het ESP32-Cam-board. Er zijn maar heel weinig GPIO-pinnen over voor gebruik nadat de SD-kaart en de camera zijn aangesloten. De enige beschikbare pinnen die we kunnen gebruiken zijn IO1, IO3, IO4, IO12, IO13 (zie kader **Aantekeningen van het Elektor Lab**). IO1 ('UoT') en IO3 ('UoR') worden ook gebruikt als seriële poort voor het laden van het programma naar de ESP32 (in flash-programmeermodus). Daarom mogen deze pinnen tijdens het laden van programma's helemaal nergens mee zijn verbonden.

In dit webcamproject kunnen we probleemloos pinnen gebruiken die zijn gereserveerd voor de pinnen van de SD-kaart, omdat we de SD-kaartfunctie niet gebruiken. We hebben echter IO1 en IO3 gebruikt als I²C-pinnen voor het aansluiten van het OLED-display. Op die manier hebben we zoveel mogelijk vrije GPIO-pinnen ter beschikking. Vergeet niet de I²C-bus los te koppelen bij het laden van een programma in de ESP32 (**figuur 2**).

Gebruik

Het werkingsprincipe van de ESP32-webcam met OLED-display is heel eenvoudig gebleven. De ESP32 zoekt naar de WiFi-netwerken die als een lijst in het programma zijn opgenomen en probeert verbinding te maken met een van deze. Zodra de verbinding tot stand is gebracht, verschijnt op het OLED-display de statusmelding dat de camera gereed is, samen met het netwerk-ID en het IP-adres dat aan de webcam is toegewezen. Om toegang te krijgen tot de camera, maak je verbinding met het aangegeven netwerk en ga je in een browser naar het IP-adres van de webcam. Het poortadres is en blijft standaard (8080).

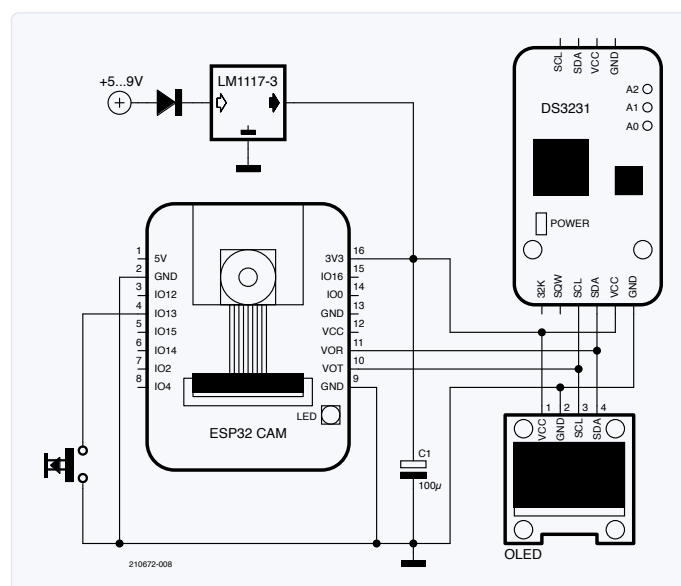
De software voor de ESP32-webcam plus OLED-display kan worden gedownload van [1]. Merk op dat de bibliotheek "ESP8266 and ESP32 OLED driver for SSD1306 displays" vereist is (we gebruikten versie

4.1.0) [2]. Deze kan worden geïnstalleerd met de bibliotheekmanager van de Arduino IDE. Vergeet niet de I²C-bus los te koppelen voordat je een sketch laadt.

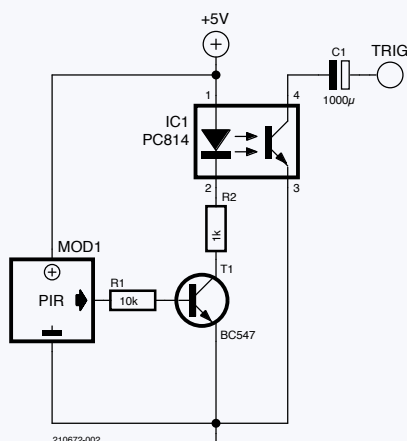
Project 2: getriggerde camera met OLED en RTC

In dit project wordt telkens wanneer er iets gebeurt een foto gemaakt. Foto's worden met een volgnummer en timestamp opgeslagen op een SD-kaart.

De schakeling in kwestie (**figuur 3**) is een uitbreiding van de vorige schakeling. Een DS3231 (of DS1307) realtime-klokmodule (RTC) wordt aan de I²C-bus toegevoegd en een triggersignaal wordt



Figuur 3. Een drukknop en een realtime-klokmodule veranderen de webcam van figuur 1 in een getriggerde bewakingscamera die ook timestamps genereert.



Figuur 4. We kunnen een 5V-PIR-sensor gebruiken – sommige werken zelfs op 3,3 V – om de ESP32-camera te activeren. Voor volledige galvanische scheiding raden we echter aan om een optocoupler (of een relais) te gebruiken.

aangesloten op IO13. Dit kan een drukknop zijn of een PIR-sensor (figuur 4) of iets anders dat een korte laag-actieve puls genereert wanneer een foto moet worden gemaakt.

IO4 wordt gebruikt als lamp: de on-board superheldere witte LED van de ESP32-Cam wordt gebruikt als extra lichtbron voor de camera. IO13 is de trigger-ingang voor het maken van foto's. Het programma voorziet de bestandsnaam van datum en tijd en slaat deze op de SD-kaart op. Vervolgens gaat hij in de deep sleep-modus om de batterij te sparen wanneer er niets te doen is en geen foto's worden gemaakt. Een laag-actieve puls op IO13 wekt de camera om een nieuwe foto te maken.

Dit programma bestaat alleen uit de functie **setup**; de functie **loop** wordt niet gebruikt. De reden hiervoor is het gebruik van de deep sleep-modus. Wanneer het apparaat uit deze modus ontwaakt, start het opnieuw op en voert het de functie **setup** uit en maakt het een foto voordat het terugkeert naar deep sleep, zodat de functie **loop** nooit wordt bereikt.

De software voor de getriggerde camera kan worden gedownload van [1]. Naast de OLED-driverbibliotheek die nodig is voor het eerste project [2], heeft dit project ook de bibliotheek "Rtc by Makuna" nodig (we gebruikten versie 2.3.5) [3]. Ook deze bibliotheek kan worden geïnstalleerd met de bibliotheekmanager van de IDE. Vergeet niet de I²C-bus los te koppelen voordat je een sketch uploadt (figuur 2).

Conclusie

De ESP32-Cam is erg goedkoop, maar biedt toch veel mogelijkheden. Door meerdere van deze camera's op een rij te installeren, kunnen ze worden gebruikt als een vast opgestelde scanner voor grotere objecten, bijvoorbeeld om foto's van de kentekenplaten van

naderende auto's te maken bij snelheidsovertredingen. Een enkele camera met PIR-sensor kan worden gebruikt voor 'snapshots' van vogels of andere dieren. Er zijn ook uitgebreidere toepassingen denkbaar. Door bijvoorbeeld gezichtsdetectie en een contactloze temperatuurscanner toe te voegen, kan het systeem enige beveiliging tegen Covid bieden. ◀

210672-03

Vragen of opmerkingen?

Hebt u technische vragen of opmerkingen naar aanleiding van dit artikel? Stuur een e-mail naar de redactie van Elektor via redactie@elektor.com.



ONDERDELENLIJST

- ESP32-Cam board
- SD-kaart
- 0,96" I²C SSD1306-compatibel OLED-display
- DS3231/DS1307 I²C RTC-module
- USB/serieel-adaptor (kabel)
- LM1117 3,3V-regelaar
- PIR-sensor of drukknop
- Optioneel: batterij, condensatoren enzovoort



GERELATEERDE PRODUCTEN

- **ESP32-Cam-CH340 Development Board (SKU 19333)**
www.elektor.nl/19333
- **0.96" I²C OLED Display (SKU 18747)**
www.elektor.nl/18747
- **SparkFun Real Time Clock Module - RV-8803 (Qwiic) (SKU 19646)**
www.elektor.nl/sparkfun-real-time-clock-module-rv-8803-qwiic

WEB LINKS

- [1] Downloads voor dit artikel op Elektor Labs: <https://www.elektormagazine.com/labs/esp32-camera>
- [2] ESP8266 en ESP32 OLED-drivers voor SSD1306-displays: <https://github.com/ThingPulse/esp8266-oled-ssd1306>
- [3] Rtc van Makuna: <https://github.com/Makuna/Rtc>
- [4] Getriggerde camera (versie van Elektor Labs): <https://www.elektormagazine.com/210672-03>

Aantekeningen van het Elektor Lab

Veel ESP32-Cam-kaarten worden samen met een praktische USB/serieel-converter dochterprint verkocht.* (Misschien moeten we het een moederbord noemen omdat de ESP32-module erop wordt aangesloten en van daaruit gevoed wordt?). Deze combinatie (figuur 5) maakt het programmeren van de ESP32 heel eenvoudig, maar het 'botst' met de I²C-interface op poorten IO1 en IO3 zoals door de auteur in zijn project gebruikt. Wanneer deze pinnen worden gebruikt, moet de I²C-poort telkens worden losgekoppeld telkens wanneer je de ESP32 wilt herprogrammeren; daarom zochten we naar een andere manier om de I²C-bus aan te sluiten.

In theorie kan de I²C-bus op de ESP32 worden geconfigureerd voor gebruik van bijna alle IO-pinnen, maar op het ESP32-Cam-board resulteert in allerlei opstartproblemen en PSRAM-geheugenfouten. Dit komt door de pull-up weerstanden op de I²C-bus.

Het ESP32-Cam-board heeft twee 8-polige uitbreidingsconnectoren. Op het eerste gezicht lijkt dat misschien in orde, maar bij nadere beschouwing is het niet zo eenvoudig. In de eerste plaats worden zes van de zestien pinnen gebruikt voor de voeding, waardoor er slechts tien overblijven voor IO. Zes hiervan (IO2, IO4, IO12, IO13, IO14 en IO15) worden gedeeld met de SD-kaarthouder. Dan blijven nog slechts de poorten IO0, IO1, IO3 en IO16 over.

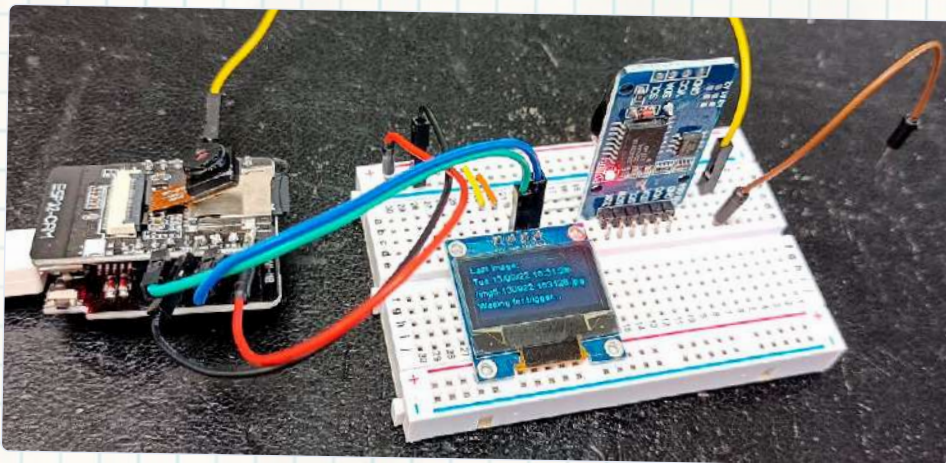
Voor het laden van een programma naar de ESP32 met aangesloten dochterprint zijn de poorten IO0, IO1 en IO3 nodig.

IO16 wordt gedeeld met de chip select-pen van de PSRAM-chip, dus deze kan alleen veilig worden gebruikt wanneer PSRAM niet nodig is.

De SD-kaart bibliotheek is standaard ingesteld op 4-bit datamodus, maar kan met

```
SD_MMC.begin("/sdcard",true);
```

in de 1-bit-modus worden gebracht. Hierdoor komen de poorten IO4, IO12 en IO13 vrij. Poort 4 wordt gedeeld met de (helder)witte flits-LED. Deze heeft een 47-k Ω pull-up weerstand en



Figuur 5. Het prototype van Elektor Labs is opgebouwd op breadboard. De ESP32-Cam met dochterprint is links te zien, de verticale module rechts is de RTC.

een 10+1 k Ω pull-down weerstand die een transistor aanstuurt. Poort 13 heeft ook een pull-up weerstand van 47 k Ω en wordt in dit project gebruikt als camera-triggeringang.

IO12 is een van die ESP32-pinnen die bij het opstarten met zorg moeten worden behandeld, omdat deze de waarde van VDD_SDIO bepaalt, de spanning van de flash-geheugeninterface. Hij moet bij het opstarten op het ESP32-Cam-board omlaag worden getrokken.

Na wat experimenteren bleek dat de I²C-bus veilig kan worden verplaatst naar de poorten IO0 (SDA) en IO3 (SCL – of andersom, dat maakt niet uit, zolang de software maar correct is geconfigureerd). Door deze poorten te gebruiken botst de I²C-bus niet met de seriële programmeerpoort of met de opstartmodus, aangezien zowel IO0 als IO3 geacht worden hoog te zijn bij het opstarten.

Tot slot (en dat staat los van het bovenstaande) moet je beseffen dat IO33 is aangesloten op een rode LED aan de ESP32-modulezijde van het board en die je op elke gewenste manier kunt gebruiken (zolang je er rekening mee houdt dat deze laag-actief is).

We lieten ons een beetje meeslepen door het tweede project en pasten het programma aan (figuur 6). Het gebruikt niet langer geëmuleerd EEPROM voor het opslaan van het afbeeldingsnummer, maar in plaats daarvan een bestand op de SD-kaart. Wis gewoon het bestand "counter.txt" om de teller op nul te zetten. Ons programma kan worden gedownload van [4].



Figuur 6. Het werkt!

* Zie het kader Gerelateerde producten voor nog een andere versie van de ESP32-Cam-module.

ATX-voeding voor Raspberry Pi

Sébastien Guerreiro de Brito (Frankrijk)

Heb je een Raspberry Pi die stroom nodig heeft, en een oude ATX-voeding? Een printplaatje, wat componenten en een ATtiny en het is gefixt.



Figuur 1: ATX-voeding. Bron: Shutterstock.

De ATX PC-voeding (**figuur 1**) werd in 1995 door Intel geïntroduceerd. Het is sinds de Pentium II-processoren de meest voorkomende schakelende voeding voor PC's. Alle elektronische, mechanische, milieu- en andere kenmerken zijn door Intel gespecificeerd om de ATX-norm te definiëren.

Het doel van deze schakeling is om een ATX-voeding te gebruiken om een Raspberry Pi-board en de vele randapparatuur die daarop kan worden aangesloten van stroom te voorzien. Het voordeel van deze voeding is het aantal verschillende spanningen dat beschikbaar is - op interessante vermogensniveaus.

ATX Voedingsaansluiting

In dit artikel gaan we echter niet de interne werking van de ATX schakelende voeding beschrijven, maar alleen proberen hem aan

de praat te krijgen. Laten we om te beginnen in detail kijken naar de connector (**figuur 2**) die moet worden aangesloten op het moederbord van de PC.

De connector bestaat uit

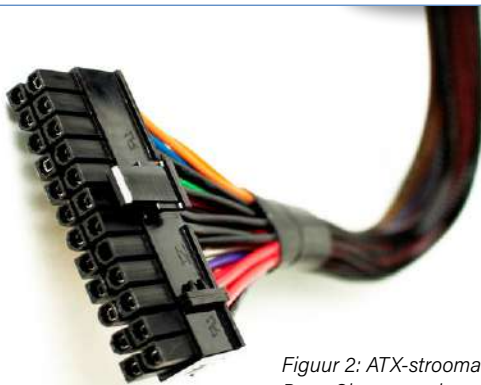
- een 24-polige, twee-rijige Molex Mini-Fit Jr vrouwelijke connector (ref: 39-01-2240) of gelijkwaardig
- Molex Mini-Fit HCS vrouwelijke contacten (ref: 44476-1112)

De pin-out van deze connector staat in **tabel 1**. De signalen die van belang zijn voor het aansturen van de voeding zijn PWR_OK en PS_ON#. De +5 VSB-voeding zal ook worden gebruikt in onze toepassing. Het PWR_OK signaal is een "Power good" signaal dat door de voeding wordt gebruikt

om het systeem te vertellen dat de +5 VDC, +3,3 VDC en +12 VDC uitgangen aanwezig zijn en voldoen aan de drempelwaarden.

Het PS_ON# signaal is het signaal waarmee we de voeding kunnen starten. Het is een TTL-compatibel signaal, actief-laag, waarmee het moederbord de voeding op afstand kan besturen. Dit maakt software-gestuurde voeding AAN/UIT, Wake-on-LAN, wake-on-USB activiteit, enz. mogelijk.

Wanneer PS_ON# laag wordt getrokken, moet de voeding de vier hoofdgelijkspanningen activeren: +12 VDC, +3,3 VDC, +5 VDC en -12 VDC. Wanneer PS_ON# naar een TTL-hoog niveau wordt getrokken, of naar een open circuit, mogen de DC-uitgangssporen geen stroom leveren en moeten ze op nul worden gehouden ten opzichte van massa. Het ATX-voedingscircuit voorziet in een intern



Figuur 2: ATX-stroomaansluiting.
Bron: Shutterstock.

anti-bounce circuit om aan/uit-schommelingen te voorkomen als het PS_ON#-signaal wordt geactiveerd door een mechanische schakelaar. De kenmerken van het PS_ON# signaal staan in **tabel 2**. De PS_ON# signaal-karakteristieken zijn in **figuur 3** als grafiek weergegeven.

Het PS_ON# signaal heeft geen effect op de +5 VSB output. De +5 VSB (SB = stand-by) voeding is een voeding die aanwezig is wanneer de voeding op het lichtnet is aangesloten. Deze output levert stroom voor circuits die operationeel moeten blijven wanneer de hoofdvoedingsspanningen zijn uitgeschakeld.

Beschrijving van de print

Om de ATX-voeding door de Raspberry Pi te laten aansturen, is een printplaatje ontworpen [1]. Dit bevat de benodigde logica om het opstarten en afsluiten van de Raspberry Pi mogelijk te maken door er een kleine controller aan toe te voegen. Dit geeft ook enkele andere leuke voordelen. Aangezien we een ATX-voeding hebben, zullen we de 5 V-rail gebruiken om een set USB-poorten aan de print toe te voegen, zodat randapparatuur die hogere stromen nodig heeft daar gebruik van kan maken. De PCB aan het werk is te zien in **figuur 4**.

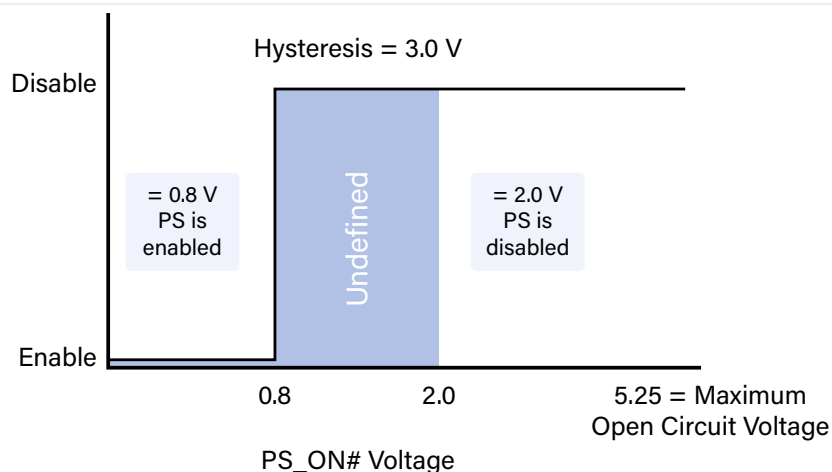
Parameter	Minimum	Maximum
V_{IL}	0 V	0,8 V
I_{IL}	-	-1,6 mA ¹
V_{IH}	2,0 V	-
$V_{IH, open circuit}$	-	-5,25 V
Rimpeling/ Ruis		400 mVpk-pk

¹ Opmerking: Negatieve stroom geeft aan dat er stroom loopt van de voeding naar het moederbord.

Tabel 2: PS_ON# signaalkarakteristieken.

PIn	Signal	Wire color
1	+3.3 VDC	
2	+3.3 VDC	
3	COM	
4	+5 VDC	
5	COM	
6	+5 VDC	
7	COM	
8	PWR_OK	
9	+5 VSB	
10	+12 V1DC	
11	+12 V1DC	
12	+3.3 VDC	
13	+3.3 VDC	
14	-12 VDC	
15	COM	
16	PS_ON#	
17	COM	
18	COM	
19	COM	
20	Reserved	NC
21	+5 VDC	
22	+5 VDC	
23	+5 VDC	
24	COM	

Tabel 1: ATX connector pinout.

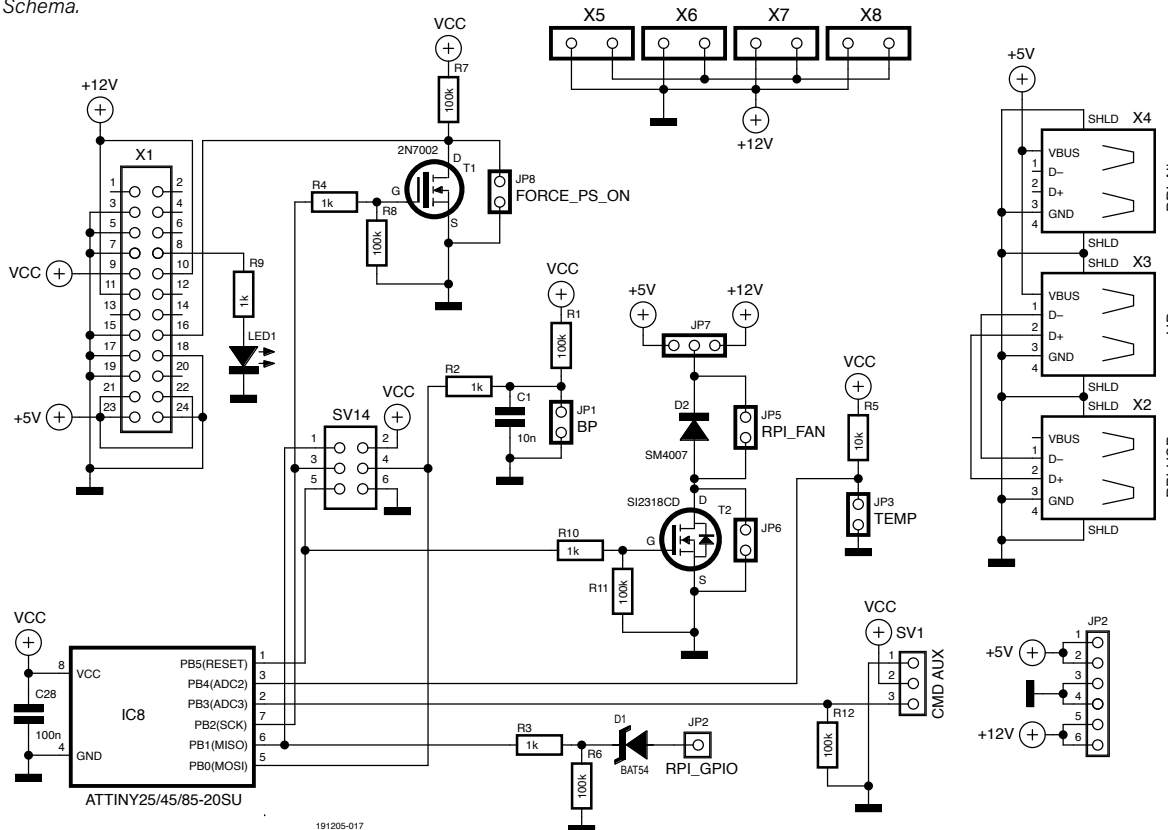


Figuur 3: PS_ON# signaalkarakteristieken. (Bron: ATX/ATX12V Power Supply Design Guide versie 1.1, paragraaf 3.3.2, figuur 3)



Figuur 4: ATX-voeding aangesloten op de printplaat.

Figuur 5: Schema.



Het schema van de printplaat is te zien in **figuur 5** en is relatief eenvoudig. Het is gebaseerd op het gebruik van een ATtiny85 microcontroller die rechtstreeks vanuit +5 VSB wordt gevoed. Het voordeel van het gebruik van een microcontroller is dat, hoewel de regeling van de voeding eenvoudig is, het de mogelijkheid biedt om andere functies toe te voegen.

Starten van de voeding

De ATtiny krijgt de opdracht de voeding te

starten door een externe drukknop die op jumper JP1 wordt aangesloten. Wanneer de drukknop wordt geactiveerd, stuurt de microcontroller transistor T1 aan, die de voeding start. Jumper JP8 maakt het mogelijk de voeding automatisch te starten.

Extra temperatuurbewaking voor de Raspberry Pi

Wij bieden extra luchtkoeling voor het Raspberry Pi bord. Hiervoor wordt een 10 kΩ NTC thermistor geplaatst en afhankelijk van

de gemeten temperatuur zal het board een ventilator activeren. Met behulp van jumper JP7 kunnen we kiezen voor een ventilatorspanning van 12 V of 5 V.

Communicatie met de Raspberry Pi

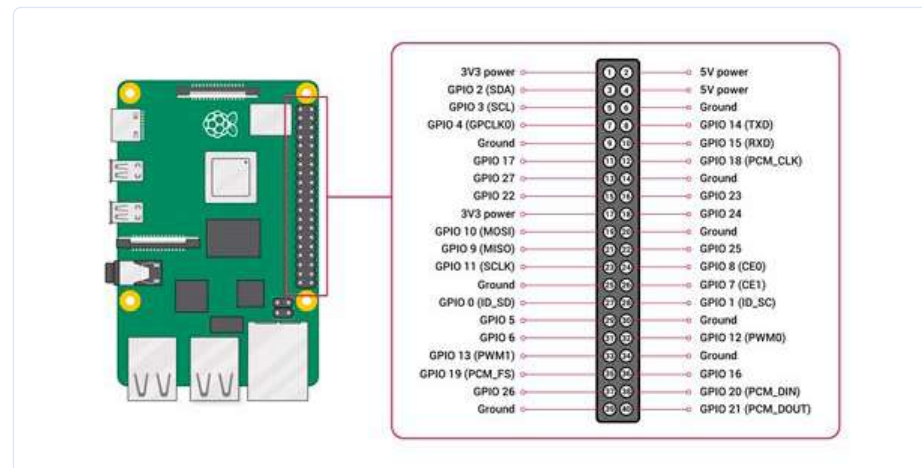
Om de ATX-voeding van de Raspberry Pi te kunnen uitschakelen, zoals op een PC, hebben we een kleine list bedacht. Het RPI_GPIO signaal is verbonden met GPIO27 op het Pi bord. De GPIO-connector van de Raspberry Pi is te zien in **figuur 6**. Wat het besturingssysteem betreft moeten we er eerst voor zorgen dat GPIO27 toegankelijk is. Daartoe maken we het GPIO-initialisatiebestand aan (dat bij het opstarten wordt uitgevoerd) met de nano-editor:

`nano S75gpioinit`

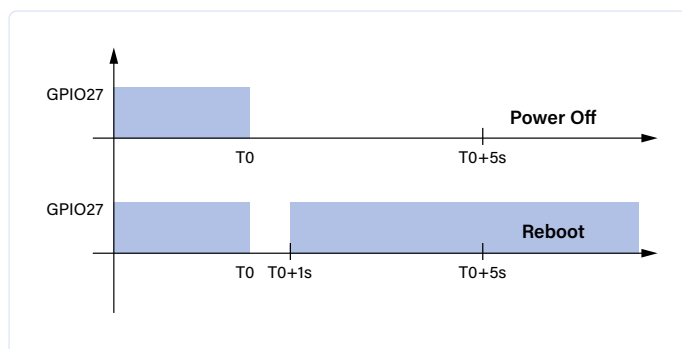
Dan voeren we de code van **Listing 1** in. We maken het bestand uitvoerbaar en plaatsen het in de initialisatiemap.

```
sudo chmod +x S75gpioinit
sudo mv S75gpioinit /etc/init.d
```

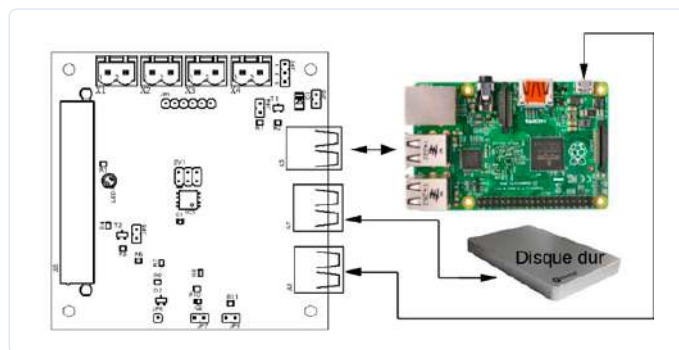
Je zult in het initialisatiescript zien dat GPIO27 is geïnitieerd met een waarde van 1. Dus, wanneer we de Raspberry Pi uitschakelen



Figuur 6: Raspberry Pi pinout (Bron: [2]).



Figuur 7: Beheer van de GPIO27-pin.



Figuur 8: Aansluiten van de Raspberry Pi.

via het OS (poweroff, bijvoorbeeld), zal dit signaal veranderen in 0 zodat de microcontroller weet dat het tijd is om de ATX-voeding uit te schakelen.

Maar, gewoon omdat het kan, willen we een reboot van de Raspberry Pi mogelijk maken met behulp van de reset van de voeding.

Hiervoor gebruiken we de truc om GPIO27 te variëren om de microcontroller het signaal te geven dat niet een poweroff, maar een reset wordt gevraagd.

Zoals je kunt zien in **figuur 7**, is het beheer van GPIO27 als volgt gedetailleerd: Als de microcontroller de GPIO27-pin op 0 ziet

staan, wacht hij 5 seconden en als de pin dan nog steeds op 0 staat, betekent dit dat de Raspberry Pi is uitgeschakeld en dus schakelt hij de ATX-voeding uit. Zo niet, dan gaat het om een reboot commando en zorgt hij ervoor dat de ATX-voeding 2 seconden uitschakelt, voordat hij opnieuw opstart.



Listing 1: Shell-Script

```
#!/bin/sh
#####
# IO Init Script #
# Author : Sebastien Guerreiro #
# Versions : #
# ----- #
# Mars 2019 V1.0 Création #
#####
CHEM_GPIO=/sys/class/gpio
#-----
register()
{
#Registers the outputs
if [ $SENS="out" ]; then
#Outputs
echo "$NUM" > $CHEM_GPIO/export
echo "$SENS" > $CHEM_GPIO/gpio$NUM/direction
echo "$VALEUR" > $CHEM_GPIO/gpio$NUM/value
chmod g+w $CHEM_GPIO/gpio$NUM/value
else
#Inputs
echo "$NUM" > $CHEM_GPIO/export
echo "$SENS" > $CHEM_GPIO/gpio$NUM/direction
fi
ret=$?;
if [ $ret -eq 0 ]; then
#echo_success;
echo "Registering GPIO$NUM : OK"

else
# echo_failure;
echo "Registering GPIO$NUM : ERROR"
exit $ret;
fi
}
#-----
unregister()
{
echo "$NUM" > /sys/class/gpio/unexport
ret=$?;
if [ $ret -eq 0 ]; then
echo_success;
else
echo_failure;
exit $ret;
fi
}
#-----
start()
{
echo "Registering GPIO--Setting Power GPIO ON"
#Output for power GPIO
NUM=27; SENS=out ; VALEUR=1
register;
}
#-----
start
exit 0
```



Listing 2: Het shutdown script.

```
#!/bin/sh
#####
# Reboot Poweroff Management #
# Author : Sebastien Guerreiro (www.sebelectronic.com)#
# Versions : #
# ----- #
# Fev 2020 V1.0 Création #
#####
case $1 in
-r)
echo 0 > /sys/class/gpio/gpio27/value
sleep 1
echo 1 > /sys/class/gpio/gpio27/value
sleep 1
/sbin/shutdownSeb $@
;;
*)
/sbin/shutdownSeb $@
;;
esac
```

Om dit alles mogelijk te maken moeten we ingrijpen met enkele Linux commando's. Aan de kant van het besturingssysteem hernoemen we de `shutdown` en `reboot` instructies in `sbin` naar respectievelijk `shutdownSeb` en `rebootSeb`.

```
sudo mv /sbin/shutdown /sbin/shutdownSeb
sudo mv /sbin/reboot /sbin/rebootSeb
```

Vervolgens maken we twee scripts: `shutdown` en `reboot`, die we in de map `/sbin` plaatsen. Typ in de terminal:

```
nano shutdown
```

en voer het volgende (**Listing 2**) in het bestand in. Sla het bestand op en sluit nano af.

Typ in de terminal:

```
sudo cp shutdown /sbin
```

Dit kopieert uw script naar `/sbin` zodat u het later kunt uitvoeren. Vervolgens moeten we het `reboot` script uitvoeren. Ook in de terminal, voer in:

```
nano reboot
```

en voer het volgende script (**Listing 3**) in dat bestand in. Sla het bestand op en sluit nano af. Typ dan in de terminal:

```
sudo cp reboot /sbin
```

Dit zal uw herstartscript kopiëren naar `/sbin` en het later laten uitvoeren.

Dat is het!



Lijst 3: Het herstartscript.

```
#!/bin/sh
#####
# Reboot Management #
# Author : Sebastien Guerreiro #
# Versions #
# ----- #
# Fev 2020 V1.0 Création #
#####
echo 0 > /sys/class/gpio/gpio27/value
sleep 1
echo 1 > /sys/class/gpio/gpio27/value
sleep 1
#/sbin/rebootSeb
/sbin/shutdownSeb -r now
```



VERWANTE PRODUCTEN

- > **Raspberry Pi 4 2 GB (SKU 18965)**
www.elektor.com/18965
- > **Velleman VTSS220 Temperatuurgestuurd Soldeerstation (SKU 19865)**
www.elektor.com/19865

WEB LINKS

- [1] Project on Elektor Labs: www.elektormagazine.com/labs/atx-powersupply-for-rpi
- [2] Image Source: www.raspberrypi.com/documentation/computers/images/GPIO-Pinout-Diagram-2.png

De randapparatuur aansluiten

Het gebruik van een ATX-voeding maakt het mogelijk het bord aan te sluiten op bijvoorbeeld een IDE harde schijf, of in mijn geval een IDE CD-ROM drive. Kortom, het voordeel is dat je de randapparatuur van je oude PC

kunt gebruiken voor je nieuwe projecten. De aansluiting gebeurt zoals je ziet in **figuur 8**.

Wie meer wil weten over het project kan terecht op de Elektor Labs pagina [1].

191205-03

Vragen of opmerkingen?

Heeft u vragen of opmerkingen over dit artikel? Neem contact op met Elektor via editor@elektor.com of laat een reactie achter op de projectenpagina van Elektor Labs.



ONDERDELENLIJST

Weerstand (0805, 0,1 W)

R1, R3, R6, R7, R8 = 1 kΩ
R2, R4, R5, R9, R10 = 100 kΩ
R11 = 10 kΩ

Condensatoren (0805)

C1 = 100 nF
C2 = 10 nF

Halfgeleiders

D1 = SM4007
D2 = BAT54
IC1 = ATTINY85

T1 = SI2318CDS

T2 = 2N7002

LED1 = LED, groen, 3 mm

Diversen

SV1, JP7 = Header 3 pennen, 2.54 mm pas

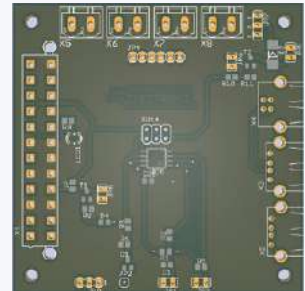
JP1, JP3, JP5, J6, JP8 = Header 2 pennen,
2.54 mm pas

JP2 = Header 1 pen

SV14 = Header 2 rijen van 3 pennen, 2.54 mm
pas

X1 = Molex Mini-Fit Jr verticale connector, 5566,
24-way, 2 rijen

X2, X3 = USB Type A connector



X4 = USB Type B connector

X5, X6, X7, X8 = MSTBVA 2.5 / 2-G-5.08 2-pins
header

Print

Advertisement

Word lid van de Elektor Community

- ✓ Een compleet web-archief t/m 1980!
- ✓ 8x Elektor Magazine (Print)
- ✓ 8x digitaal (PDF)
- ✓ 10% korting in onze webshop, en exclusieve aanbiedingen
- ✓ Toegang tot meer dan 5000 Gerberfiles



Neem **nu** een
lidmaatschap!



www.elektormagazine.nl/Abonnement

elektor
design > share > earn

32Ω

koptelefoonversterker

eenvoudige maar hoogwaardige 3-chip-oplossing

Thierry Clinquart (België)

Vroeger hadden de meeste goede koptelefoons een impedantie van 600 Ω. Tegenwoordig domineren 32Ω-versies in de midden- en hogere prijsklassen. Daarom hebben moderne koptelefoonversterkers iets meer vermogen nodig.

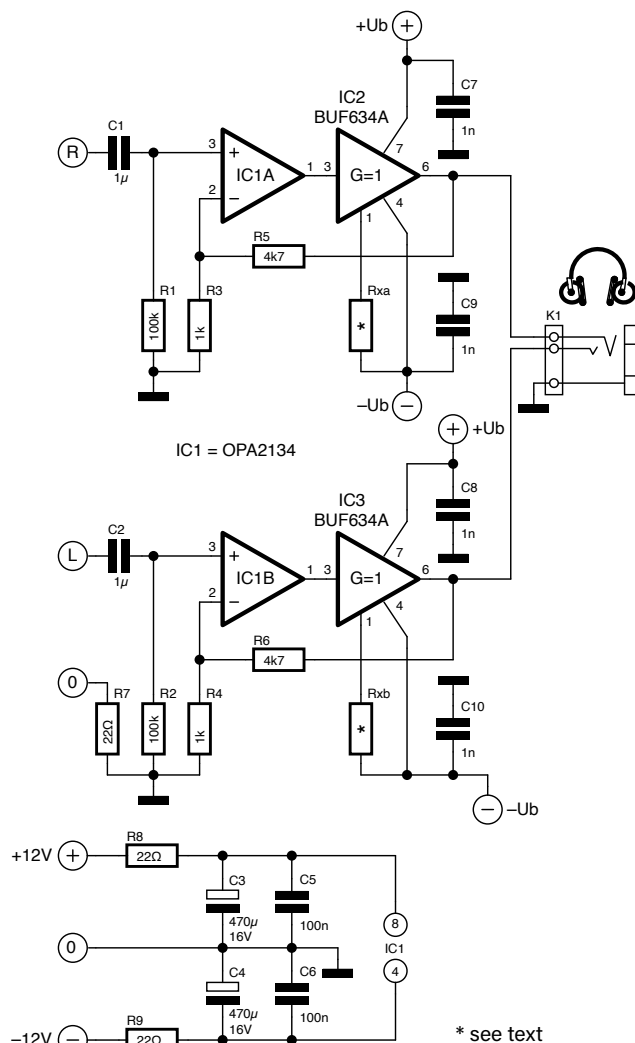
Terwijl een normale opamp in principe volstaat om een 600Ω-koptelefoon zeer fors aan te sturen, vloeien bij een vergelijkbaar volume in de spreekspoelen van moderne 32Ω-koptelefoons aanzienlijk hogere stromen, waardoor normale opamps overstuurd zouden worden. De klassieke oplossing met de oude, maar goede NE5532 audio-opamps en vergelijkbare IC's is dus achterhaald. Om ervoor te zorgen dat moderne koptelefoons niet vervormd klinken, moet de klassieke opampschakeling dus worden gemodificeerd. Tegenwoordig is dit heel eenvoudig, niet alleen theoretisch, maar ook praktisch. Alles wat je nodig hebt, is een BUF634A per kanaal.

De BUF634A

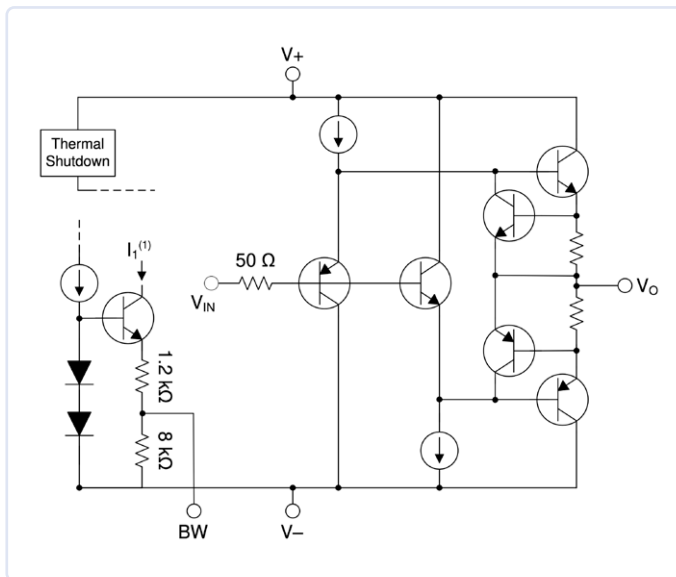
Dit is in feite een kleine en snelle push-pull eindversterker in een IC-behuizing. Je hoeft hem alleen maar achter een opamp te zetten en in de negatieve terugkoppeling op te nemen – klaar!

Figuur 1 toont de volledige stereoschakeling. Een dubbele audio-opamp van het type OPA2134 [1] levert de twee ingangstrappen en is verantwoordelijk voor de spanningsversterking. Twee BUF634A's zorgen voor de stroomversterking. De interne schakeling is te bewonderen in **figuur 2**.

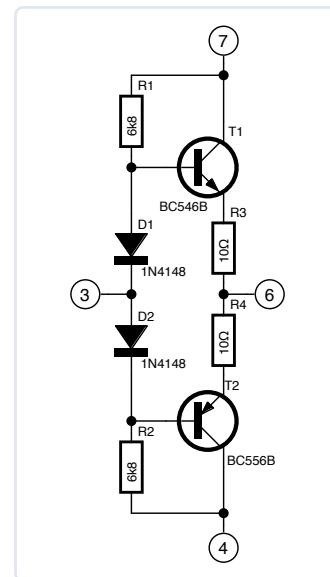
Het bijzondere van deze geïntegreerde push-pull eindtrap is de zeer hoge bandbreedte, die kan worden ingesteld tussen 35 en 210 MHz via de stroom die in pin 1 loopt. Als weerstand Rx in figuur 1 wordt weggelaten (pen 1 = open), is de bandbreedte 35 MHz en de ruststroom slechts ongeveer 1,5 mA. De maximale bandbreedte ontstaat als voor Rx een waarde van 0 Ω wordt gekozen. In dat geval loopt de ruststroom



Figuur 1: De schakeling van de volledige koptelefoonversterker berust op geïntegreerde halfgeleiders.



Figuur 2: De prinsieschakeling van de BUF634A geïntegreerde push-pull eindtrap. Bron: [2].



Figuur 3: Deze discrete schakeling kan worden gebruikt in plaats van het BUF634A IC.

op tot een aanvaardbare 8,5 mA. De eindtrap kan stromen leveren tot 250 mA – genoeg voor koptelefoons met impedanties tot 8 Ω . Zie voor gedetailleerde informatie de datasheet [2].

Welke bandbreedte, en dus welke ruststroom, is de juiste? Voor toepassing als kleine koptelefoonversterker zou normaal gesproken de lage bandbreedte in orde moeten zijn, en kun je Rx weglaten. Voor hogere capacatieve belastingen is echter de hogere ruststroom met $R_x = 0 \Omega$ de betere keuze. Dit laatste geldt ook voor opamps die niet stabiel zijn als spanningsvolger. Ruststromen tussen de extremen zijn eenvoudig instelbaar met andere waarden van Rx.

Fijnafstemming en modificatie

Met SMD-componenten kun je op een kleine ruimte een uitstekende koptelefoonversterker bouwen met minimale vervorming, zeer weinig ruis en een hoge bandbreedte. Maar ook wie van bedrade componenten houdt, zal blij zijn met deze schakeling. IC1 is ook verkrijgbaar in een 8-polige DIL-versie, en voor IC2 en IC3 kun je gemakkelijk een equivalent van het type BUF634 gebruiken. Dit IC is nog steeds verkrijgbaar in een achtpins DIL-behuizing. Het zal werken, hoewel het iets langzamer is, een iets hogere ruststroom heeft, en niet langer aanbevolen wordt voor nieuwe ontwerpen.

Een alternatief is om de geïntegreerde BUF634A te vervangen door een discrete equivalente schakeling met kleinsignaal-transistoren, zoals weergegeven in **figuur 3**. De ruststroom is dan afhankelijk van de respectieve gecombineerde eigenschappen van de halfgeleiders, en kan worden aangepast via de waarden van R1 en R2. Verder moet er met het oog op de thermische stabiliteit voor worden gezorgd dat D1 en D2 in direct contact staan met T1 en T2. Zo nodig kunnen ze ook aan elkaar worden gelijmd. Omdat de ruststroom ook afhankelijk is van de voedingsspanning, moet de versterker worden gebruikt met een gestabiliseerde $\pm 12V$ -voeding, bijvoorbeeld met behulp van 7812 of

7912 spanningsregelaars. Voor een 32 Ω -koptelefoon is een voeding van ± 100 mA voldoende. Voor een 8 Ω -koptelefoon is ongeveer het dubbele nodig. R7 vermindert brom veroorzaakt door eventuele aardlussen.

Natuurlijk kun je experimenteren met andere types voor IC1 of T1 en T2 en goede resultaten krijgen. Bij gebruik van koptelefoons met impedanties $\geq 32 \Omega$ zijn er geen warmteproblemen met de BUF634A. Bij 8 Ω



ONDERDELENLIJST

Weerstanden

(standaard: metaalfilm, 1%)

R1,R2 = 100 k

R3,R4 = 1 k

R5,R6 = 4k7

R7...R9 = 22 Ω

Rxa,Rxb = zie tekst

Condensatoren

C1,C2 = 1 μ , 25 V, film

C3,C4 = 470 μ , 16 V, elektrolytisch

C5,C6 = 100 n, 25 V

C7...C10 = 1 n, 25 V, keramisch

Halfgeleiders

IC1 = OPA2134

IC2,IC3 = BUF634A (zie tekst)

Andere

K1 = Koptelefoonaansluiting, stereo, 6,3 mm



en langdurig luisteren op hoog volume kunnen SOIC-IC's behoorlijk warm worden. Zorg er dan bij het ontwerpen van een print voor dat de warmte via de pads wordt afgevoerd naar voldoende koper op de print, of kies voor de 8-polige DRB-behuizing die aan de onderzijde is voorzien van een "thermal die pad". Het type BUF634 (zonder de "A") is zelfs verkrijgbaar in een gemakkelijk te koelen TO-220 of TO-263 behuizing.

Aan de ingang kun je het beste filmcondensatoren gebruiken voor C1 en C2. Aan de uitgang zijn geen condensatoren nodig omdat de typische offsetspanning slechts een paar mV zal bedragen, wat geen significante uitslag van de spreekspoelen veroorzaakt. Als je hier echter absoluut zeker van wilt zijn, kun je een condensator plaatsen tussen de uitgang en de koptelefoon. Voor 32Ω-impedantie zou een bipolair type van 470 µF (25 V) voldoende zijn. Voor een 8Ω-koptelefoon is minstens 1000 µF nodig. De uitgang is kortsluitvast – maar niet als je de BUF634-vervanging uit **figuur 3** gebruikt.

Conclusie

Moderne analoge elektronica maakt kleine koptelefoonversterkers mogelijk met uitgesproken audiofiele eigenschappen. Vervormingsfactoren in het bereik van <0,01% bij gemiddelde volumes zijn gemakkelijk haalbaar en ruis is bijna onhoorbaar dankzij de goede eigenschappen van IC1. De versterker kan tot 300 mW leveren aan 32Ω-koptelefoons (en, met goede koeling, nog steeds tot 250 mW aan 8 Ω), wat veel meer is dan je je oren moet aandoen. ◀

200441-03 (vertaling: Willem den Hollander)

Over de auteur

Thierry Clinquart is opgeleid tot elektronicus en heeft een passie voor audio. Al zijn projecten draaien om analoge audio – voorversterking, dynamische verwerking, correcties, signaalverdeling enzovoort. Hij maakt zijn eigen PCB's met Sprint-Layout van Abacom en sPlan voor de schema's.

Vragen of opmerkingen?

Als je technische vragen hebt, kun je de Elektor-redactie mailen via redactie@elektor.com.



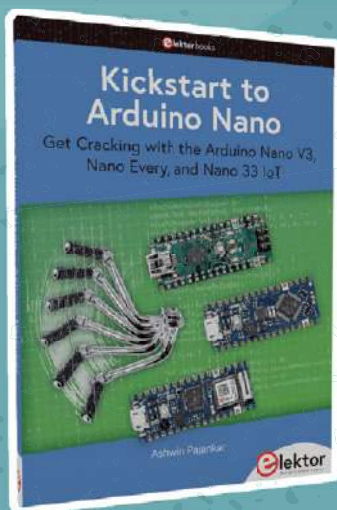
Verwante producten

- > **OWON SDS1102 2-kanaals oscilloscoop (100 MHz) (SKU 18782)**
www.elektor.nl/18782
- > **PeakTech 3350 True RMS digitale multimeter (6000 counts) (SKU 19986)**
www.elektor.nl/19986
- > **Joy-IT JDS6600 Signaalgenerator & Frequentieteller (SKU 18714)**
www.elektor.nl/18714

WEBLINKS

[1] OPA2134 Datasheet: <https://ti.com/product/OPA2134>

[2] BUF634A Datasheet: <https://ti.com/product/BUF634A>



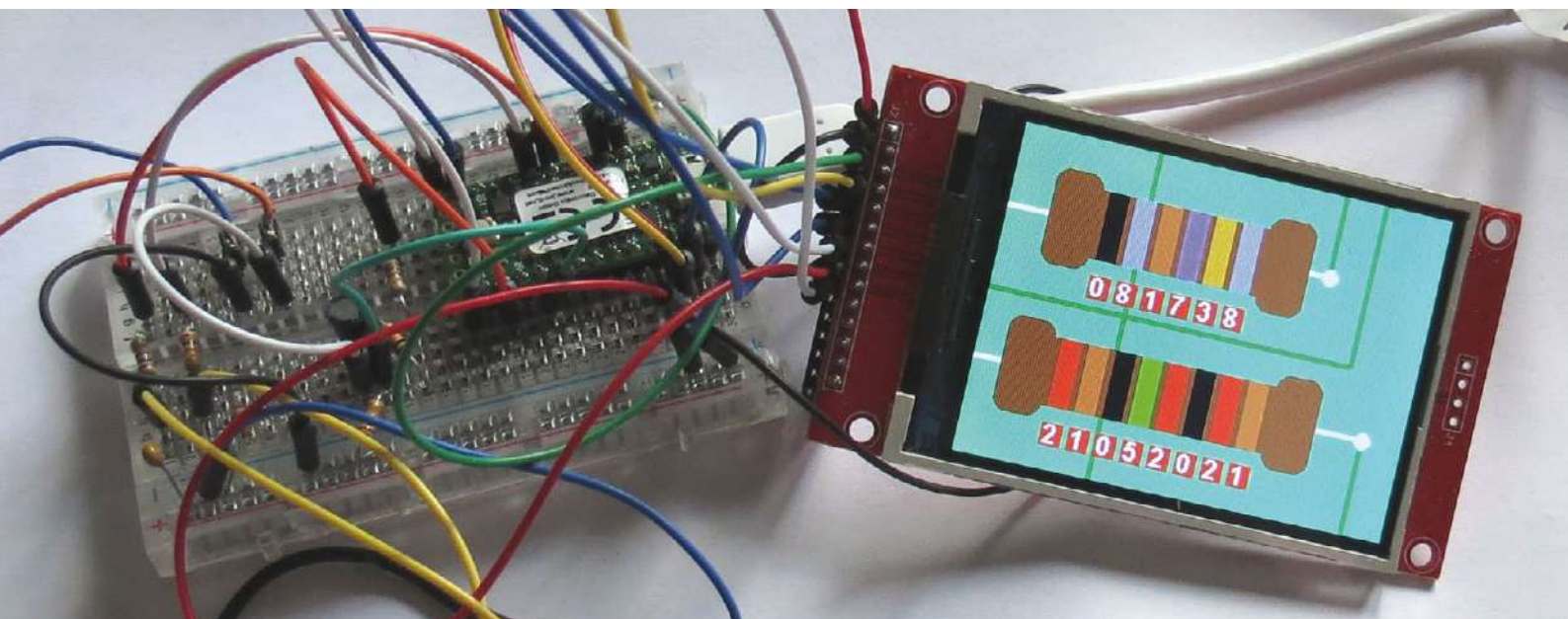
Kickstart to Arduino Nano

Dit boek dient als eerste stap voor beginners en liefhebbers van microcontrollers die een begin willen maken met het programmeren met Arduino. Het boek volgt een stap-voor-stap aanpak om concepten en de werking van dingen uit te leggen. Elk concept wordt stevast gevolgd door een to-the-point schema en codevoorbeelden. Daarna volgt gedetailleerde uitleg over de syntaxis en de gebruikte logica. Door de concepten nauwgezet te volgen, raakt u vertrouwd met het bouwen van schakelingen, het programmeren met Arduino, de werking van de codevoorbeelden en de gepresenteerde schakelschema's.

www.elektor.nl/20241



elektor



De SDR-radioklok bestaat uit het Teensy 4.0 microcontroller-board met aangesloten display, hier opgebouwd op breadboard. De tijd wordt hier weergegeven als weerstand-kleurcode.

SDR- radioklokken

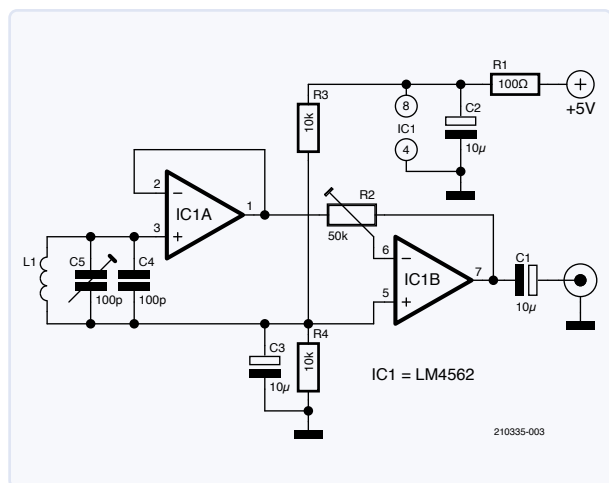
Prof. dr. Martin Ossmann

De huidige microcontrollers zijn zo krachtig dat ze gebruikt kunnen worden om radiogestuurde klokken te bouwen op basis van het software-defined radio (SDR) principe. Hoe dit in zijn werk gaat, is aangetoond door de MSF-radioklok [1], opgebouwd rond een Raspberry Pi Pico. Ook dit artikel gaat over tijdsignalen, maar dit keer wordt het snelle Teensy 4.0-board gebruikt als microcontroller.

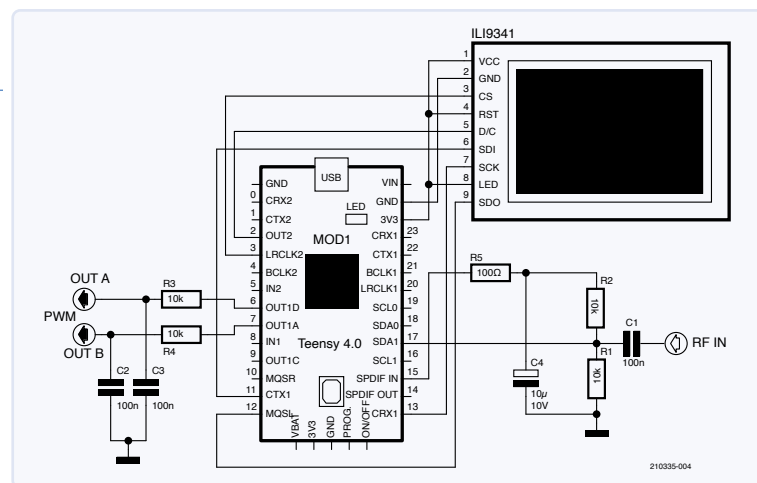
Tabel 1. Voor deze zenders zijn ontvangers geïmplementeerd.

60,0 kHz	MSF	Britse tijdsignaalzender NPL
77,5 kHz	DCF77	Duits tijdsignaal
129,1 kHz	EFR	Radio-rimpelcontrole
162 kHz	TDF	Frans tijdsignaal
198 kHz	BBC	AM-uitzending met fasemodulatie

In dit project ontwikkelen we een ontvanger voor de signalen van de EFR-radio-dienst, die door energieleveranciers wordt gebruikt voor rimpelcontrole. EFR zendt ook met regelmatige tussenpozen tijdinformatie uit. Het bouwen van een radioklok ermee is iets nieuws. Hetzelfde op SDR gebaseerde concept kan met dezelfde hardware worden gebruikt om andere tijdsignalen te ontvangen, die wij van plan zijn uitgebreid te gebruiken (**tabel 1**). Een ander belangrijk doel van dit project is het tonen van de tijd op verschillende innovatieve displays.



Figuur 1. Schema van de actieve antenne voor 129,1 kHz. De 250 windingen van 0,2 mm² geëmailleerd koperdraad zijn gewikkeld op een ETD29-spoelvorm. De spoel is op een ferrietstaaf van 150 × 8 mm geschoven, gemaakt van 3B1-materiaal. Door Cx aan te passen is de schakeling bruikbaar voor frequenties van 60 kHz tot 200 kHz.



Figuur 2. Schema van onze software-gedefinieerde radio voor radiogestuurde klokken.

Eén concept voor alle frequenties

Om de radiosignalen te ontvangen is allereerst een actieve antenne nodig, zoals in het schema van **figuur 1**. Het signaal van de ferrietantenne (150 mm × 8 mm, materiaal 3B1, spoelvorm ETD29, 250 windingen CuL 0,2 mm²) wordt gefilterd met een resonantiekring en vervolgens gebufferd door een opamp. De tweede trap is weer een opamp, waarvan de versterking met instelpot R2 aan de plaatselijke omstandigheden kan worden aangepast. Onze ontvanger werkt het beste wanneer een signaal van ongeveer 50 mV_{tt} op de uitgang van de tweetraps versterkerschakeling staat. De schakeling is geschikt voor alle hier gebruikte tijdsignaalfrequenties; wil je de antenne voor andere frequenties gebruiken, dan zul je condensator C4 moeten aanpassen.

Het volledige systeemschema is te zien in **figuur 2**. De belangrijkste component is een Teensy 4.0-board. Met weerstanden R1 en R2 stellen we de offset van de A/D-converter in, en met C1 koppelen we het signaal van de actieve antenne naar de A/D-converter van het Teensy-board. We hebben ons prototype op breadboard opgebouwd (zie nogmaals figuur 1). De twee laagdoorlaatfilters, R4/C2 en R3/C3, filteren de PWM-signalen van uitgangen 6 en 7. We gebruiken de PWM-signalen voor debugging, bijvoorbeeld om de I- en Q-signalen op een oscilloscoop weer te geven. Het display wordt aangesloten via de SPI-connector en enkele extra pinnen. De voeding gaat via de USB-poort, die ook wordt gebruikt om het board te programmeren en de debugging-teksten uit te voeren. Zo kun je de ontvangst van de gegevens bekijken in

de seriële monitor van de Arduino IDE. Het Teensy-board kan worden geprogrammeerd in de normale Arduino-ontwikkelomgeving.

Tijdcodes ontvangen

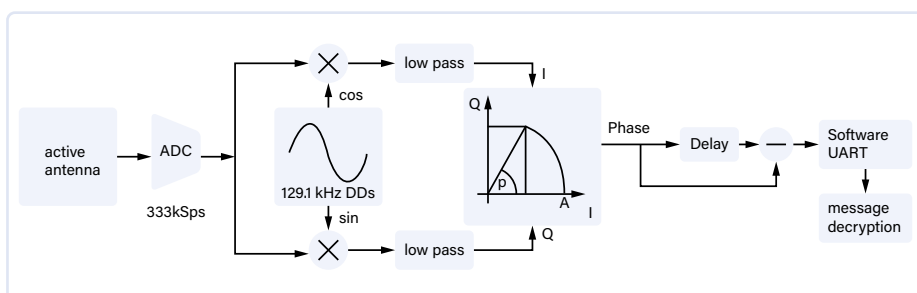
Het begrip 'universeel' heeft niet alleen betrekking op de hardware van de ontvanger, maar ook op de software. De structuur blijft gelijk voor alle hier genoemde tijdsignaalzenders. Maar omdat de demodulatie- en decoderings-routines zeer verschillend zijn, leek het mij zinvoller om voor elke tijdseinzender aparte firmware te schrijven dan alle routines in één firmware te stoppen en de gewenste te activeren via bijvoorbeeld een schakelaar. Het volledige, gedocumenteerde softwarepakket kan worden gedownload van de projectpagina bij dit artikel [2]. De basiswerking van de ontvanger zal echter eerst nader worden beschreven, met als voorbeeld de ontvangst van de EFR-langegolfzender op 129,1 kHz.

EFR-ontvangst

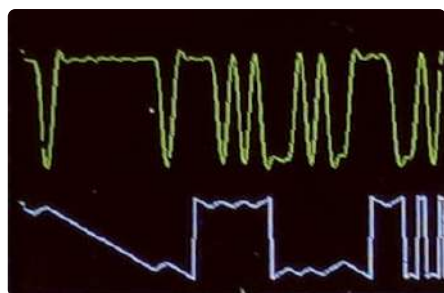
EFR GmbH exploiteert drie langegolfzenders (129,1 kHz, 139 kHz en 135,6 kHz) voor rimpelcontrole in energiecentrales [3]. De zenders

in Duitsland en Hongarije hebben elk een bereik van ten minste 500 km en bestrijken dus het grootste deel van Midden-Europa. Er wordt gebruik gemaakt van frequency-shift keying (FSK) met een afwijking van ±170 Hz en een datasnelheid van 200 bps. Naast de gecodeerde eigen berichten wordt ook tijdinformatie verzonden. Het protocol is gespecificeerd in IEC 60870-5.

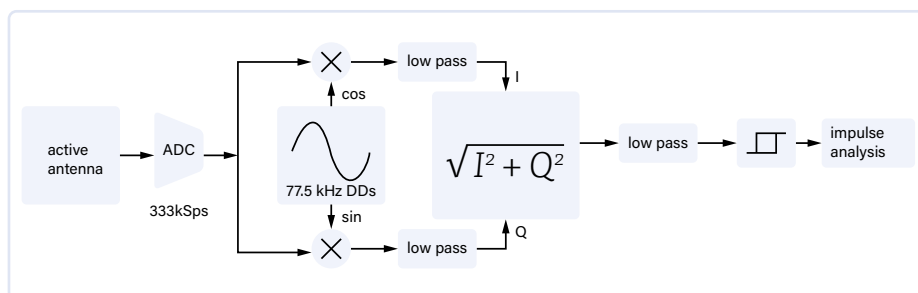
Het concept van onze EFR-ontvanger is geschetst in **figuur 3**. Het signaal van de actieve antenne bereikt de interne ADC van het Teensy 4.0-board, die geklokt is op 333,333 kHz. Er wordt dus voldaan aan het bemonsteringstheorema. Zoals gebruikelijk bij SDR-ontvangers wordt vervolgens een IQ-vermenigvuldiging uitgevoerd. Het oscillatorsignaal wordt gegenereerd met een in software geïmplementeerde DDS-generator. Na vermenigvuldiging van het ingangssignaal met het I- of Q-oscillatorsignaal wordt laagdoorlaatfiltering uitgevoerd. De laagdoorlaatfilters zijn zo gedimensioneerd dat het ontvangen signaal nog niet geclipt wordt, maar dat er ook geen andere signaalcomponenten doorgelaten kunnen worden.



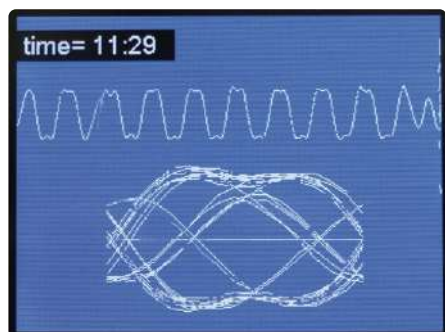
Figuur 3. De opzet van de EFR-ontvanger.



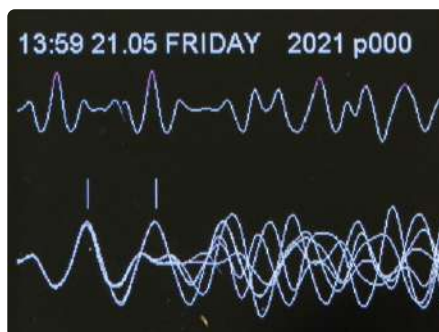
Figuur 4. Debug-informatie op het display. Bovenste grafiek: gedemoduleerd RTTY-sigitaal (momentane frequentie); onderste grafiek: momentane fase.



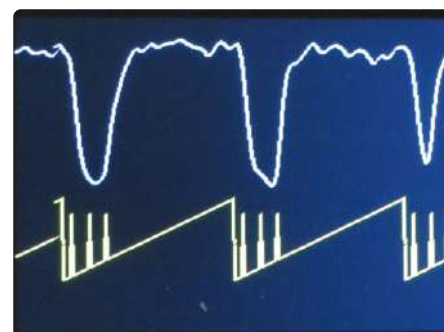
Figuur 5. Opzet van de DCF-ontvanger.



Figuur 6. Debug-informatie van de BBC-ontvanger. Boven: teruggewonnen datasigitaal; onder: oogdiagram.



Figuur 7. Debug-signalen van de TDF-ontvanger.



Figuur 8. MSF-debuginfo. Boven: ontvangsniveau met secundemarkers; onder: timervariabele met bemonsteringspunten.

Om het RTTY-sigitaal te demoduleren, moeten we de momenteel ontvangen frequentie bepalen. Bij +170 Hz offset is het een één, bij -170 Hz is het een nul. Daartoe bepalen we de huidige fase uit de I/Q-signalen. Dit doen we met de functie `atan2()`. Uit de faseverandering in de tijd kunnen we de frequentie bepalen. Daartoe berekenen we het verschil tussen het oorspronkelijke en het vertraagde sigitaal, wat resulteert in het gedemoduleerde sigitaal, dat we doorgeven aan een software-UART om de verzonden bytes terug te krijgen. Zo decoderen we de verzonden berichten, waaruit we dan weer de tijdinformatie kunnen halen. **Figuur 4** toont de debug-signalen op het display, boven het gedemoduleerde RTTY-sigitaal (momentane frequentie), en onder de momentane fase.

DCF-ontvanger

Het concept van de DCF-ontvanger (**figuur 5**) lijkt sterk op het EFR-concept. Ditmaal wordt echter de momentane zendamplitude bepaald aan de hand van de I/Q-signalen. Daartoe berekenen we $\sqrt{I^2 + Q^2}$.

Dit sigitaal wordt door een laagdoorlaatfilter gestuurd en naar een Schmitt-trigger geleid. Hierdoor komen de tweede pulsen beschikbaar voor evaluatie, en kunnen we uit de bits de datum en tijd bepalen.

BBC-ontvanger

De BBC zendt een radiosigitaal uit op 198 kHz. De draaggolf is fasegemoduleerd en wordt gebruikt voor gegevensoverdracht, vergelijkbaar met het RDS-systeem bij FM-radio [4]. De fasemodulatie wordt op dezelfde manier gedemoduleerd als in het EFR-systeem. De bitklok van 25 bps moet worden teruggewonnen uit het gedemoduleerde sigitaal (**figuur 6**). Daartoe vormt de ontvanger als het ware het oogdiagram en varieert de klok zodanig dat het oog zich maximaal opent. De gegevens worden dan bemonsterd in het midden van het oog. Van de aldus verkregen bitstroom moeten nu de blokgrenzen worden hersteld. De gegevens worden verzonden in blokken van elk 50 bits, waarvan 13 bits controlebits zijn. De ontvanger zoekt in de bitstroom naar foutloze blokken en herkent de blokgrenzen aan hun positie. Er zijn verschillende bloktypen die verschillende gegevens bevatten. Bloktype 0 is bedoeld voor de overdracht van de tijdinformatie en wordt door onze ontvanger gedecodeerd.

TDF-ontvanger

Tot 2016 zond TDF (voorheen *Télédiffusion de France*) een radioprogramma uit op 162 kHz, maar tegenwoordig wordt alleen de fasegemoduleerde draaggolf uitgezonden. In de gegevens is tijdinformatie opgenomen. Een

nul wordt verzonden als een enkele puls, een één als een dubbele puls, elk aan het begin van een seconde. **Figuur 7** toont de ontvangen fase-informatie (bovenste curve) en onderaan, als een soort oogdiagram, over elkaar liggende curven met de pulsen links tussen de verticale lijnen. In de 59^e seconde van elke minuut is het sigitaal niet fasegemoduleerd. Onze ontvanger gebruikt deze pauze voor synchronisatie. De structuur van de individuele bitinformatie is vergelijkbaar met die van DCF77.

MSF op 60 kHz

Op 60 kHz heeft het NPL (*National Physical Laboratory*) in Groot-Brittannië een langegolfzender die met 15 kW tijdinformatie uitzendt. Er wordt gebruik gemaakt van eenvoudige aan/uit-draaggolfmodulatie [5]. De minuut begint met een pauze van 500 ms. De volgende seconden beginnen elk met een draaggolf-onderbreking van 100 ms en vervolgens nog twee onderbrekingen van 100 ms indien nodig, afhankelijk van de te versturen bits. Onze MSF-ontvanger is opgebouwd zoals de DCF-ontvanger, alleen de pulsdecoding gebeurt anders. Vergeleken met de andere signalen is het MSF-sigitaal relatief zwak, maar toch heel behoorlijk te ontvangen aan de westelijke grens van Duitsland (**figuur 8**).

Display-opties

Toen de auteur besloot om een op EFR gebaseerde radioklok te bouwen, moest hij ook beslissen welk display moest worden gebruikt. Hij koos voor een 2,8"-LCD met 320 x 240 pixels. Het display is gebaseerd op een ILI9341-controller en is bijvoorbeeld via eBay verkrijgbaar. Als alternatief kan het kleine broertje van het display met een diagonaal van 2,2" maar dezelfde resolutie uit de Elektor-shop worden gebruikt (zie **Gerelateerde producten**). Het display wordt op het Teensy-board aangesloten via de SPI-interface.

Weerstand-display

Onze EFR Klok biedt een enigszins ongebruikelijke nieuwe weergaveoptie, het 'weerstand-display' (figuur 9). De cijfers van tijd en datum worden kleur gecodeerd weergegeven zoals gebruikelijk bij weerstanden. Er zijn twee weerstanden in het display, één met zes ringen voor de zes cijfers van de tijd (uren:minuten:seconden, bijvoorbeeld 12:23:45). De tweede weerstand heeft acht ringen voor de datum (dag.maand.jaar, bijvoorbeeld 23.08.2012). De tijd wordt elke seconde bijgewerkt, en de lokale klok wordt

altijd gesynchroniseerd wanneer de tijdgegevens van de EFR-dienst worden ontvangen. Voor gebruikers die de weerstandscodes nog niet kennen, worden de tijd en datum ook normaal numeriek weergegeven. De puristen onder de lezers wijzen er misschien op dat weerstanden maximaal zes ringen hebben, maar zo kunnen beginners tenminste op een leuke manier de DIN41429-kleurcode leren.

Domino-display

Een ander grappig display-idee is een domino-display geïmplementeerd (figuur 10). Met dominostenen kun je de cijfers 0...9 weergeven. Elke domino biedt ruimte voor twee cijfers, dus je hebt drie domino's nodig voor de tijd en vier voor de datum. Of het weerstand-display, het domino-display of een andere optie wordt gebruikt, wordt in de broncode bepaald met behulp van optiebits (listing 1). De gewenste optie wordt geselecteerd met een "1", de andere opties worden uitgeschakeld met een "0".

Woordklok en analog display

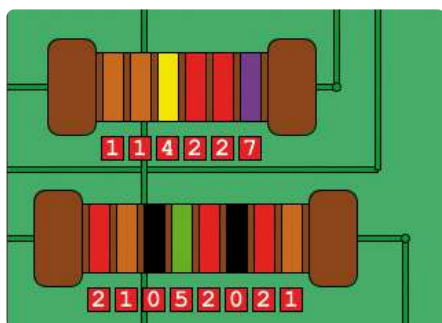
Naast de tot nu toe beschreven weergaven kun je ook kiezen voor een woord-display,

waarbij de tijd als platte tekst (Duitstalig) op het display wordt weergegeven (figuur 11). Bij deze optie wordt alleen de tijd weergegeven, niet de datum. De tijd wordt elke vijf minuten bijgewerkt. Het klassieke displayformaat van een analoge klok is ook geïmplementeerd (figuur 12). De tijd wordt weergegeven op een ronde wijzerplaat, en de datum als tekst eronder.

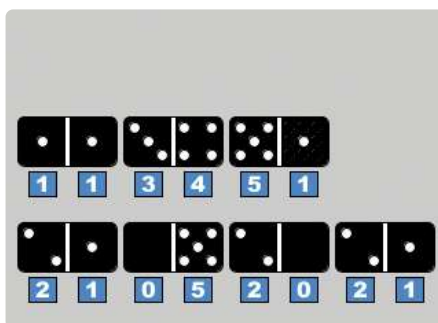


Listing 1. Het display wordt geselecteerd via deze constanten.

```
#define scopeShow      0
#define scopeClkShow  1
#define resShow        0
#define dominoShow    0
#define wordShow       0
#define sevenSegShow   0
#define AP6571Show     0
```



Figuur 9. Weergave als weerstand-kleurcode.



Figuur 10. Domino-display.



Figuur 11. Er is ook een tekstuele klok geïmplementeerd.



Figuur 12. Klassieke analoge klok.



Figuur 13. Nostalgisch zeven-segment display.



Figuur 14. Weergave als bij een dotmatrix-printer.

Zeven-segment-display

Natuurlijk hebben we ook een eenvoudig, nostalgisch zeven-segment-display in de aanbieding. Dus als je aan vroeger dagen wilt worden herinnerd, is dit de juiste keuze (figuur 13).

Dotmatrix-display

Als laatste optie bieden wij een eenvoudige tekstweergave op basis van de legendarische 6571AP ROM-karaktergenerator, ooit veel gebruikt in 9-pins dotmatrix-printers (figuur 14).

Over het geheel genomen biedt dit project veel combinatiemogelijkheden en opties. Je kunt kiezen tussen vijf radioklokvarianten van

EFR tot DCF en tussen zes display-opties, dus er moet echt voor elk wat wils tussen zitten. Als je wilt, kun je de software, bijvoorbeeld de display-opties, ook gebruiken als uitgangspunt voor eigen ontwikkelingen. ◀

210335-03

Vragen of opmerkingen?

Hebt u technische vragen of opmerkingen naar aanleiding van dit artikel? Stuur een e-mail naar de auteur via ossmann@fh-aachen.de of naar de redactie van Elektor via redactie@elektor.com.



Gerelateerde producten

- > **Teensy 4.1 Development Board (SKU 19311)**
www.elektor.nl/teensy-4-1-development-board
- > **2.2" SPI TFT display module ILI9341 (240x320) (SKU 18419)**
www.elektor.nl/2-2-spi-tft-display-module-ili9341-240x320

WEB LINKS

- [1] Martin Ossmann, "Een MSF-SDR met de Raspberry Pi Pico", Elektor juli/augustus 2022: <https://www.elektormagazine.nl/magazine/elektor-266/60631>
- [2] Firmware op de projectpagina: <https://elektormagazine.nl/210335-03>
- [3] EFR-signaal: https://mee.hu/files/images/5/B_Sbick_EFR-CEE_Lakihegy.pdf
- [4] L.F. Radio Data specification: <https://downloads.bbc.co.uk/rd/pubs/reports/1984-19.pdf>
- [5] MSF 60 kHz tijd- en datumcode: https://www.npl.co.uk/products-services/time-frequency/msf-radio-time-signal/msf_time_date_code

elektor e-zine

Your dose of electronics



Elke week dat u zich niet inschrijft voor onze nieuwsbrief mist u leuke elektronica-gerelateerde artikelen en projecten!

Dus, waarom nog langer wachten? Abonneer u vandaag nog op www.elektor.nl/ezine en ontvang bovendien een gratis Raspberry Pi projectboek!



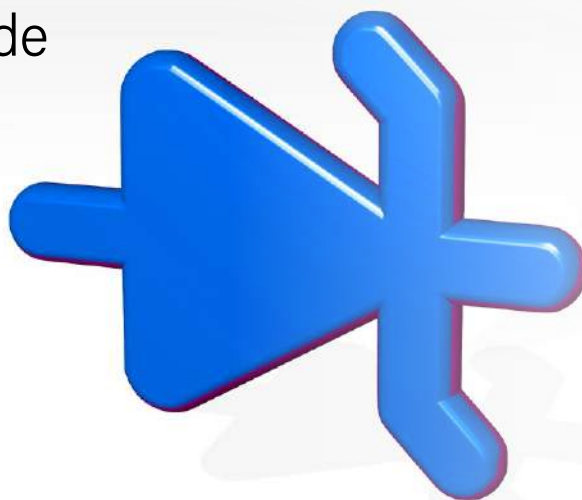
elektor
design > share > earn

Alle begin...

...zet een streep onder de diode

Eric Bogers (Elektor)

In deze aflevering sluiten we het hoofdstuk over diodes af met enkele min of meer exotische exemplaren, die echter hun bestaansrecht hadden en nog steeds hebben.



De eerste generatie dimmerschakelingen werden nog met thyristoren gebouwd – triacs bestonden nog niet (althans niet voor de gewenste vermogens). Tegenwoordig worden in dimmers vrijwel uitsluitend triacs gebruikt.

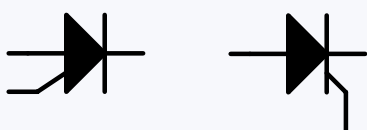
De thyristor

Een thyristor (**figuur 1**) is in principe niets anders dan een diode met een stuuraansluiting – als de thyristor geleidt, doet hij dat slechts in de doorlaatrichting, net als een gewone diode. Daarom hebben we voor AC-toepassingen twee thyristoren (of een gelijkrichter) nodig.

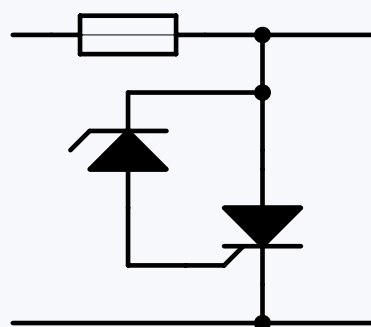
Om een thyristor in geleiding te brengen, heeft hij een 'ontsteekpuls' nodig op zijn derde aansluiting, de *gate* (de beide andere aansluiting heten net als bij een gewone diode *anode* en *kathode*). Als een thyristor eenmaal ontstoken is, dan blijft hij geleiden zolang er een stroom doorheen loopt – er hoeft geen spanning meer op de gate te staan. Om de thyristor uit te schakelen (weer in sperrende toestand te brengen) moet de stroom erdoorheen worden onderbroken. In een gelijkstroomkring zal een thyristor 'eeuwig' geleiden; in een wisselstroomkring slechts tot de eerstvolgende nuldoorgang. Er zijn twee mogelijkheden om met een thyristor een dimmer te

construeren: de wisselspanning kan worden gelijkgericht, of we kunnen twee thyristoren gebruiken die antiparallel zijn geschakeld. Omdat op deze manier opgebouwde dimmerschakelingen tegenwoordig geen praktisch nut meer hebben, gaan we daar niet nader op in.

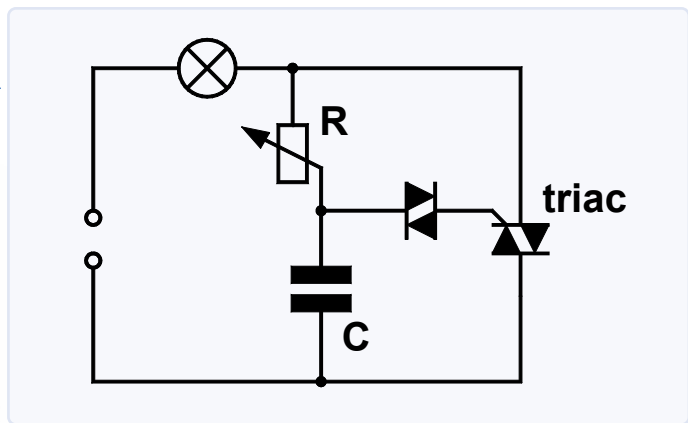
Thyristoren worden vaak gebruikt als beveiliging tegen overspanning, zoals geschetst in **figuur 2**. Als in de netvoeding de stabilisator of een vermogenstransistor doorbrandt, dan komt er zonder beveiliging een (veel) te hoge spanning op de aangesloten componenten en apparaten te staan, waardoor deze vernield kunnen worden. Dat is een situatie die we moeten proberen te voorkomen. Zodra echter in de schakeling van figuur 2 de spanning te hoog oploopt, slaat de zenerdioden door en ontsteekt de thyristor die vervolgens continu in geleiding gaat. Dat veroorzaakt een



Figuur 1. De thyristor.



Figuur 2. Overspanningsbeveiliging (principe).



Figuur 3. Conventionele (lees: verouderde) dimmerschakeling.



Figuur 4. Fase-aansnijding.

kortsluiting die de zekering doet aanspreken. Dat is minder ‘erg’ dan een heleboel kapotte elektronica.

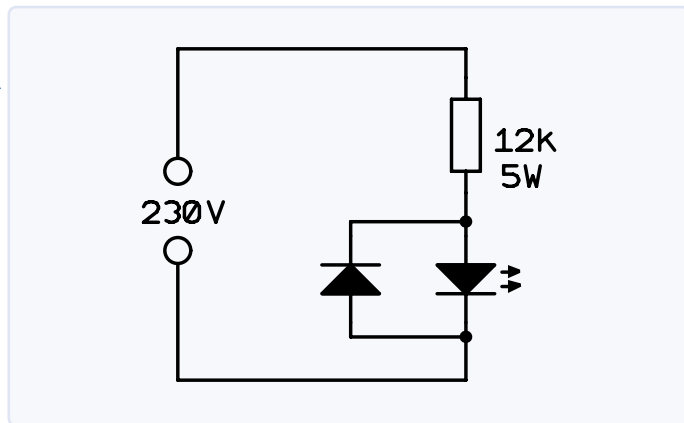
Afhankelijk van het type hebben thyristoren voor het ontsteken een spanning van 0,8...2,5 V nodig. De zenerspanning moet dus zo worden gekozen dat bij een overspanning deze ontsteekspanning gegarandeerd wordt bereikt.

De triac

Net zoals een diac als een combinatie van twee zenerdiodes kan worden beschouwd, is een triac een combinatie van twee thyristoren – met geen ander doel dan dat we voor AC-toepassingen maar één component nodig hebben. Laten we de triac meteen maar in een praktische schakeling bekijken – de conventionele dimmerschakeling van **figuur 3**. Zo’n dimmer werkt volgens het principe van fase-aansnijding: al naar gelang een kleiner of groter deel van de fase (periode) wordt doorgelaten, zal de aangesloten lamp zwakker of helderder branden. In **figuur 4** hebben we dit geschetst.

De triac schakelt door op het moment dat de spanning over de diac groter wordt dan ongeveer 33 V en deze de ontsteekpuls voor de triac levert. Wanneer potmeter R in **figuur 3** tegen de onderste aanslag is gedraaid, wordt de spanning enerzijds door het RC-netwerk gedeeld en anderzijds 90° in fase verschoven – de spanning over de diac bereikt dan nooit een waarde van 33 V. Als vervolgens de potmeter langzaam wordt verdraaid, dan neemt de spanning over de diac toe (en verschuift de fase ervan ook steeds minder) met als gevolg dat de triac op een steeds vroeger tijdstip zal ontsteken. De schakeling van **figuur 3** is zo ongeveer de eenvoudigst denkbare (lichtnet)dimmer: elke vorm van ontstoring ontbreekt, terwijl ook geen maatregelen zijn getroffen tegen de inschakel-hysterese. Er bestaan tegenwoordig veel betere dimmerschakelingen, maar die zijn een stuk ingewikkelder.

Natuurlijk kun je een lamp dimmen door er een regelbare (vermogens)potmeter mee in serie te schakelen, maar dan wordt een groot gedeelte van de lichtnet-wisselspanning in die potmeter in nutte-



Figuur 5. LED met vrijlooptiode.

loze warmte omgezet. Het voordeel van fase-aansnijding is dat dit vermogensverlies niet optreedt.

Lichtgevende dioden en optocouplers

Lichtgevende dioden zijn dioden die licht geven zodra er een stroom doorheen loopt. De doorlaatspanning ligt bij ongeveer 2 V terwijl de maximale stroom normaliter zo’n 20 mA bedraagt (bij zogenaamde *low current*-LED’s bedraagt die maximale stroom slechts ongeveer 2 mA).

LED’s hebben al sinds jaar en dag in bedieningspanelen de plaats van gewone indicatielampjes ingenomen. Ze munten uit door een zeer lange levensduur, een relatief gering stroomverbruik en een grote schok- en trillingsbestendigheid. De Engelse benaming is *Light Emitting Diode*, vandaar het acroniem LED.

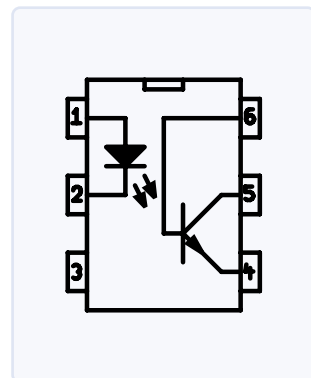
Inmiddels zijn LED’s in alle mogelijke kleuren verkrijgbaar, bestaan er ook laser-LED’s (waar we de CD- en DVD-speler aan danken) terwijl onder de naam Neopixel LED’s met ingebouwde microcontroller verkrijgbaar zijn waarmee alle mogelijke kleuren en lichteffecten haalbaar zijn. Ook hebben LED’s hun intrede gedaan in de verlichtingstechniek als (verplichte) vervanging van gloeilampen. Hoewel dat alles bijzonder interessant is, valt een bespreking daarvan ver buiten het kader van deze basiscursus. We beperken ons dus tot een korte algemene bespreking, ter ere van deze kleine maar o zo belangrijke component.

LED’s op wisselspanning

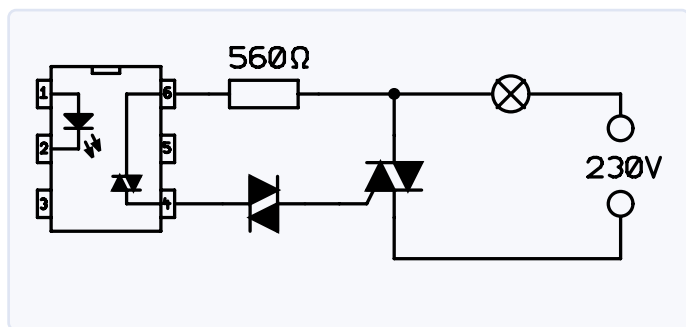
De toelaatbare sperspanning van LED’s bedraagt in veel gevallen slechts een paar volt. Als ze uit een wisselspanning moeten worden gevoed, moeten ze per se van een vrijlooptiode worden voorzien die de sperspanning over de LED tot een veilige waarde begrenst (zie **figuur 5**). A propos: je ziet dat het schemasymbool van een LED een gewone diode is – met een tweetal pijltjes die het uittredende licht symboliseren.

Optocouplers

Een optocoupler bestaat uit een lichtgevende diode en een lichtgevoelige component (vaak een fototransistor) in één behuizing (**figuur 6**). Zodra er een spanning over de LED wordt gezet, zal die oplichten waardoor de fototransistor gaat geleiden. Het grote voordeel van deze op het eerste



Figuur 6. Optocoupler.



Figuur 7. Optocoupler met fototriac.

gezicht nodeloos ingewikkelde aanpak is de galvanische scheiding tussen LED en fototransistor: tussen deze componenten kan een spanningsverschil van (vaak) meerdere honderden volt bestaan zonder dat dit tot problemen leidt. Optocouplers worden veel gebruikt om het vermogensgedeelte van dimmer- of andere schakelingen gescheiden te houden van de stuurlektronica: het vermogensgedeelte kan bijvoorbeeld de lichtnetpotentiaal voeren terwijl de stuurlektronica daarvan gevrijwaard blijft (al was het alleen maar om veiligheidsredenen).

Optocoupler met fototriac

Een transistor geleidt maar in één richting (zoals we in de volgende aflevering zullen zien), en dat kan voor sommige toepassingen onhandig zijn. Wanneer we daarmee een triac willen aansturen, zouden we een aparte spanning nodig hebben en zou de triac elke tweede halve wisselspanningsperiode met een ontsteekpuls van tegengestelde polariteit worden ontstoken, wat een asymmetrisch gedrag tot gevolg zou hebben. Om een lang verhaal kort te maken: een gewone optocoupler is voor dergelijke toepassingen allesbehalve ideaal. Daarom gebruiken we in die gevallen bij voorkeur een optocoupler met een fototriac in plaats van een fototransistor (**figuur 7**).

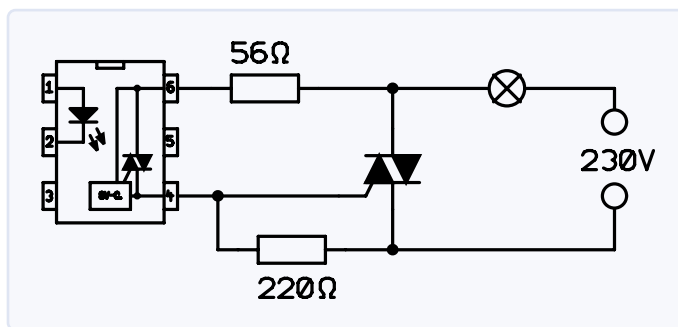
Wanneer de LED stroom voert en oplicht, gaat de fototriac geleiden en schakelt op zijn beurt de externe triac door. Bij heel bescheiden belastingen (zoals een kleine ventilator) kan de externe triac desnoods vervallen en kan de belasting rechtstreeks door de fototriac worden geschakeld.

De schakeling van figuur 7 zou eigenlijk ook zonder weerstand en diac moeten functioneren. Het is echter gebleken dat er zich dan twee problemen kunnen voordoen:

- zonder de diac vertoont de schakeling een neiging om ongecontroleerd door te schakelen: de triac gaat dan geleiden zonder dat de LED een stuurpuls heeft gegeven;
- veel vermogenstriacs doen de fototriac doorbranden wanneer de stroom niet door een weerstand wordt begrensd.

Optocouplers met nuldoorgangsdetector

Wanneer de stroom ergens in het midden van een periode wordt ingeschakeld, dan ontstaat daarbij een hoogfrequente stoerpuls die bijvoorbeeld de geluidsinstallatie hoorbaar kan storen. Deze HF-storingen zijn de oorzaak van het beruchte 'dimmer-zoemen'.



Figuur 8. Optocoupler met nuldoorgangsdetector.

Bij dimmers die volgens het principe van fase-aansnijding werken valt daar helaas niet veel aan te doen: we kunnen slechts proberen het probleem met behulp van deugdelijke ontstoorfilters binnen de perken te houden.

Maar in alle situaties waar een verbruiker slechts hoeft te worden in- of uitgeschakeld, dient het schakelmoment samen te vallen met de nuldoorgang van de lichtnet-wisselspanning: in dat geval kan er namelijk helemaal geen storing ontstaan. Hiervoor kun je eventueel zelf ingewikkelde schakelingen bedenken en bouwen – of je kunt een optocoupler gebruiken waar al een nuldoorgangsdetector in is geïntegreerd.

Natuurlijk is het niet mogelijk om hiermee een triac exact op de nuldoorgang te schakelen – want op dat moment is er geen spanning voorhanden die de benodigde ontsteekstroom kan leveren. Maar het is wel mogelijk om zo dicht in de buurt van de nuldoorgang te schakelen dat de ontstane storingen een verwaarloosbaar niveau hebben.

Hier sluiten we onze verkenning van de wonderbaarlijke wereld van de diodes af. In de volgende aflevering maken we kennis met de transistor. ◀

220446-03

De artikelreeks "Alle begin..." is gebaseerd op het boek "Basiscursus elektronica" van Michael Ebner, dat bij Elektor is verschenen.

Vragen of opmerkingen?

Hebt u technische vragen of opmerkingen naar aanleiding van dit artikel? Stuur een e-mail naar de redactie van Elektor via redactie@elektor.com.



Related Products

- Michael Ebner, *Basiscursus elektronica* (e-boek, SKU 18232) www.elektor.nl/18232



Een high-end UHF reader ontwikkelen

Toen Kiman (42) 6 jaar geleden met Nedap in aanraking kwam via een squashmaatje vond hij het maar bijzonder: onbeperkte vakantiedagen, geen vaste opleidingsbudgetten, veel vrijheid om je baan in te vullen zoals je wil en bovendien korte lijntjes met de in-house productieafdeling. Een verschil van dag en nacht met het ingenieursbureau waar hij voorheen werkte als Electrical Engineer. Daar ging alles uurtje factuurtje, werd de productie uitbesteed en voelde hij zich nooit de eigenaar van de producten waar hij aan werkte. Bij Nedap veranderde dat en zag 'zijn kindje' het licht: de uPASS Target. Als electrical engineer bij Nedap is Kiman dagelijks bezig met de ontwikkeling van UHF-technologie.

Rolls Royce onder de UHF readers

De uPASS Target is een high-end UHF RFID-lezer voor voertuigidentificatie op lange afstand. Op basis van passieve UHF-technologie (± 900 MHz) worden voertuigen geïdentificeerd op een afstand tot 10 meter. Naast voertuigidentificatie kan de uPASS Target ook worden gebruikt om personen of rollend materieel te identificeren. Typische toepassingen zijn onder meer toegangscontrole van beveiligde zones en monitoring van verkeersactiviteiten op industriële locaties en logistieke depots. "We noemen de uPASS Target ook wel de Rolls Royce van de UHF-readers," zegt Kiman lachend, „het antenneontwerp maakt het product uniek, de

reader die we gebruikt hebben heeft een grote ontvangstgevoeligheid. Dat heeft te maken met de module die we gebruiken. Doordat we een high-end product ontwikkelden was prijs echter geen issue en konden we gaan voor maximale performance. Dit maakte het ontwikkelproces leuk, nog minder kaders!"

On the spot problemen oplossen

"Eén van de projecten waar ik trots op ben is de toepassing van de uPASS Target in de parkeergarage van een bekende Europese voetbalclub. In zo'n garage staan mooie auto's die ik niet elke dag zie, dus tijdens het werken op locatie keek ik mijn ogen uit!" Maar zoals bij elk project loop je soms tegen dingen aan. "In dit geval hadden we last van zogenaamde 'cross-reads', veroorzaakt door de nauwe ruimtes in de parkeergarage. Een cross-read krijg je als je twee readers hebt die elkaar storen. We hebben goed bestudeerd waarom het probleem zich voordeed en we hebben advies gegeven over hoe de readers het beste opgehangen konden worden. Probleem opgelost! Wat mooi is, is dat we deze ervaring later bij andere klanten weer hebben kunnen gebruiken om problemen sneller op te lossen."

Lees het volledige verhaal van Kiman op onze website:
www.nedap.com/uhf



Foto: Shutterstock/ Jackrit Singhanutra

Uit het leven gegrepen

over de kwaliteit der dingen

Ilse Joostens (België)

‘Vroeger was alles beter’ is een gevleugelde uitspraak van menig ouder persoon, refererend aan de ‘goeie ouwe tijd’, al dan niet vergezeld van een sneer naar ‘de jeugd van tegenwoordig’. Hoewel ik ook al meer dan een halve eeuw op deze aardkloot rondloop weet ik wel beter. Het idee dat het vroeger beter was, is van alle tijden en voornamelijk het gevolg van ons brein dat ons voor de gek houdt omdat positieve ervaringen ons beter blijven dan negatieve.

Toch denk ik niet dat mijn brein me beduvelt als ik het gevoel heb dat de kwaliteit van consumentenartikelen vroeger in het algemeen beter was. Niet dat er toen geen rommel

werd verkocht maar tegenwoordig is het toch wel heel erg. De vooruitgang in technologie leidt niet alleen tot verbeteringen en meer mogelijkheden maar helaas ook tot het

beknotten van de vrijheid van de consument en erger nog het geniepig verkorten van de levensduur van producten.

Koninklijke lampen en onvervangbare batterijen

De ontdekking van de heldere blauwe LED in 1993 heeft een revolutie in gang gezet op het vlak van verlichtingstechniek. Het heeft nog even geduurd vooraleer de LED-lamp in de winkel op het schap lag en naast een zeer laag verbruik werd ook een fenomenale levensduur van enkele tienduizenden uren beloofd. In de praktijk lijkt het soms eerder om minuten te gaan dan uren, want ik heb al behoorlijk wat kapotte LED-lampen vervangen. Bedrijven willen nu eenmaal graag verkopen en dan zijn producten met een lange levensduur minder goed voor de omzet. In de elektronica is ontwerpen op het randje van – of zelfs over – de maximale specificaties van componenten een sport op zich. Dat is moeilijker dan het lijkt want bij voorkeur dient het product zo snel mogelijk – maar natuurlijk wel na de garantietermijn – te falen, om niet te moeten opdraaien voor de geleden schade. De consument in kwestie laat zich toch weer keer op keer beduvelen en koopt zuchtend gewoon een nieuw en beter model.

Dat is met LED-lampen niet anders en vele zijn zo ontworpen dat de LED's de maximaal toelaatbare stroom voor hun kiezen krijgen of net ietsje meer. Het verband tussen licht-opbrengst en stroom is niet lineair waardoor het rendement afneemt en meer vermogen in warmte wordt omgezet. Met deze aanpak krijgen zowel de LED's als de andere elektronica het zwaar te verduren waardoor de lamp er voortijdig de brui aan geeft. Mag je weer een nieuwe kopen, kassa!

Het kan ook anders. Om de stad Dubai energiezuiniger te maken, is Sjeik Mohammad Bin Rashid Al Maktoum een partnerschap aangegaan met Philips voor de productie van miljoenen LED-lampen, de zogenaamde Dubai-lampen [1][2][3]. Door de LED's aan te sturen tot de helft van hun maximaal vermogen is niet alleen de levensduur veel langer, maar deze lampen verbruiken voor dezelfde lichtopbrengst de helft van onze LED-lampen. Het zal u niet verbazen dat deze lampen hier lange tijd niet te koop waren, wat had u gedacht? Pas recent heeft Philips – al dan niet onder politieke druk – hier vergelijkbare LED-lampen uitgebracht met hun Master Ultra Efficiency-serie.

Jaren geleden werd me gevraagd een satellietontvanger te herstellen die leed aan een of andere vorm van vroege dementie, wat zich uitte als willekeurige tekens op het display. Zowat alle elco's in de 12V-voedingslijn op de hoofdprint bleken 10V-types te zijn. Ik heb geen idee hoelang de gemiddelde 10V-elco het uithoudt bij 12 V, maar er moet over nagedacht geweest zijn omdat de garantietermijn net verstreken was. Nadat ik ze vervangen had, door 16V-exemplaren, werkte het ding weer als een zonnetje. Het is geen alleenstaand geval want zelfs in een defecte accula-



der van een grasmaaier van een gerenommeerd A-merk vond ik een 10V-elco terug die als afvlakcondensator piekspanningen tot wel 14 V te verwerken kreeg.

Iedereen weet dat lithium-ion accu's niet het eeuwige leven hebben en dus vervangbaar zouden moeten zijn. Niet zo bij mijn draadloze steelstofzuiger dus. Kort na de garantietermijn – alweer – begon de zuigtijd sterk terug te lopen. Na veel gepriegel, gevloek en een opgehaalde vinger heb ik de 18650-cellen succesvol kunnen vervangen. Helaas is puntlassen en solderen niet meteen weggelegd voor de gemiddelde stofzuigergebruiker. Onze oude energievretende sledestofzuiger daarentegen doet het anders na 24 jaar nog prima. Hetzelfde met accu's van recente smartphones waar een vervanging stilaan meer weg heeft van een openhartoperatie.

De softwareduivel en het Nieuw Europees Bauhaus

Het voorgaande was maar 'old school', door de toenemende digitalisering en het gebruik van software zijn de mogelijkheden om consumenten te pesten zo ongeveer eindeloos geworden. Van inktpatronen voor printers met chips tot eindeloze software-updates die uw apparaten zoals computers en smartphones niet alleen vertragen maar op termijn

zelfs onbruikbaar maken wegens zogezegd 'verouderd'. Of erger nog, op een mooie dag laat uw printer doodleuk weten dat hij het einde van zijn levensduur heeft bereikt waarna hij er prompt mee ophoudt [4]. Een bekende fabrikant van labelprinters [5] heeft het zelfs gepresteerd om de labels te voorzien van RFID-tags zodat je hun duur papier moet kopen, uiteraard om de gebruikservaring te verbeteren. Nou ja, dit soort dingetjes wordt altijd als goed voor de gebruiker, consument of burger voorgesteld, maar uiteindelijk is die onveranderlijk de pineut.

In het kader van 'planned obsolescence' zijn de groene bobo's met hun Nieuw Europees Bauhaus [6] misschien nog wel de ergsten. Ik heb het niet zo op die Duitse bouwstijl begrepen, maar ook de toenemende druk op burgers om hun nog perfect werkende apparaten op de schroothoop te gooien en te vervangen door energiezuinigere, 'groene' varianten baart me zorgen. Los van de zotte kosten lijkt het me dat de productie en recyclage van huishoudapparaten ook behoorlijk wat CO₂ uitstoot. Ook ik ondergetekende ben sinds kort bij wet verplicht mijn nog perfect functionerende gasketel op het stort te gooien en te vervangen door een nieuwe omwille van een paar procenten rendement. We weten weer waarvoor we moeten gaan werken dus... 

220461-03



WEB LINKS

- [1] Leds from Dubai, the royal lights you can't buy: <https://hackaday.com/2021/01/17/leds-from-dubai-the-royal-lights-you-cant-buy/>
- [2] The lamps you're not allowed to have. Exploring the Dubai lamps: <http://www.youtube.com/watch?v=klaJqofCsu4>
- [3] Making "Dubai lamps" from 13 W Philips bulbs with a simple hack!: <http://www.youtube.com/watch?v=rXJq2vLJhLA>
- [4] Planned obsolescence rears its ugly head in Epson printer spat: <http://www.extremetech.com/computing/338810-planned-obsolescence-rears-its-ugly-head-in-epson-printer-spat>
- [5] Why Dymo Label Printers SUCK!: <http://www.youtube.com/watch?v=xzSDJRC0F6c>
- [6] European Parliament: Report on the New European Bauhaus: https://www.europarl.europa.eu/doceo/document/A-9-2022-0213_EN.html

Reverse-engineering van een Bluetooth Low Energy LED-badge

een BLE-apparaat besturen met een Python-script

Koen Vervloesem (België)

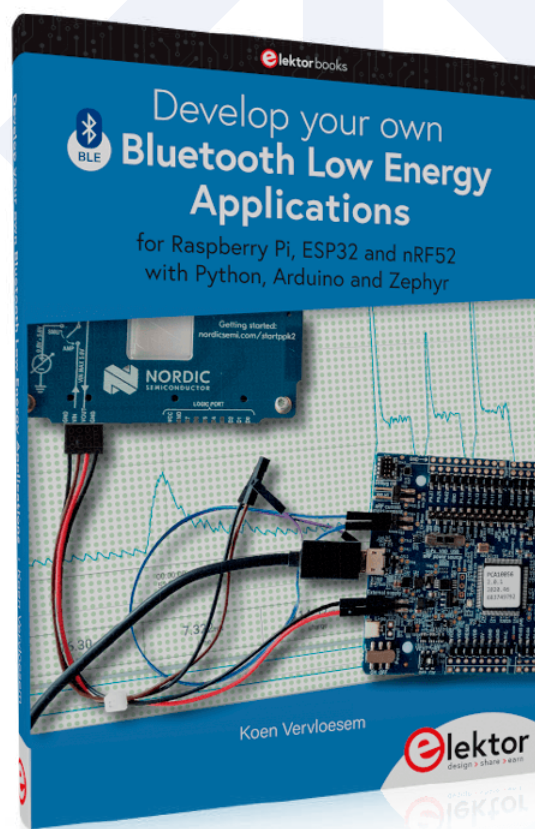
Veel Bluetooth Low Energy (BLE)-apparaten worden geleverd met hun eigen mobiele app die gebruikerscontrole biedt en doorgaans een aangepast protocol implementeert dat alleen door het apparaat en de bijbehorende app wordt begrepen. En zoals bijna altijd is er geen specificatie die je kunt bestuderen om je eigen implementatie te maken. Het goede nieuws is echter: reverse-engineering van het BLE-apparaat is de manier om het te 'kraken' en met je eigen software te gebruiken. Dat gaat zo.

Opmerking van de redactie: Hoewel dit artikel gebaseerd is op een hoofdstuk uit het 257 pagina's tellende boek *Develop your own Bluetooth Low Energy Applications* (Elektor 2022), werd de inhoud door de auteur speciaal voor Elektor Magazine bewerkt tot een compleet en volledig reproduceerbaar project.

In dit artikel ga ik een BLE LED-badge reverse-engineeren als voorbeeld van deze uitdagende en uiterst leerzame activiteit. Doe mee en ontdek hoe het apparaat werkt door de BLE-services en -kenmerken te onderzoeken. Kijk met me mee hoe ik de bijbehorende mobiele app decompileer en BLE-verkeer tussen de app en het apparaat afluister. Mijn doel is om een Python-script te schrijven om de LED-badge te besturen, zodat ik de officiële mobiele app niet langer nodig heb.

Onderzoek van de LED-badge

De LED-badge van AliExpress zoals afgebeeld in **figuur 1** heeft een 11 x 55 LED-pixeldisplay dat in verschillende kleuren verkrijgbaar is. Het ondersteunt Bluetooth zonder gespecificeerde versie. Het scannen van de QR-code op de achterkant van de badge resulteert in een "HTTP 404"-fout. Op Google Play vond ik een app met de naam "Bluetooth LED Name Badge", afkomstig van Shenzhen Lesun



Electronics Co. Ltd (**figuur 2**). Deze app kan tekst en pictogrammen naar de badge sturen en biedt een aantal instelmogelijkheden, waaronder het scrolltempo.

Laten we de LED-badge inschakelen en de BLE-scanner-app "nRF Connect for Mobile" [1] gebruiken om te zien wat het apparaat zoal 'adverteert' zoals dat heet. Druk twee keer op de onderste knop. Het display toont dan een BLE-pictogram. In nRF Connect zie je dat het nu fabrikantsspecifieke gegevens en twee eigen BLE-services aangeeft: 0xfee7 en 0xfee0 (**figuur 3**). Het apparaat heeft de naam **LSLED**. Door er verbinding mee te maken, worden de services onthuld, samen met de kenmerken die in **tabel 1** zijn opgesomd.

Tabel 1. Karakteristieken van de BLE-badge.

Service	Karakteristiek	Eigenschap
0xfee7	0xfec7	WRITE
0xfee7	0xfec8	INDICATE
0xfee7	0xfec9	READ
0xfee0	0xfee1	NOTIFY, READ, WRITE

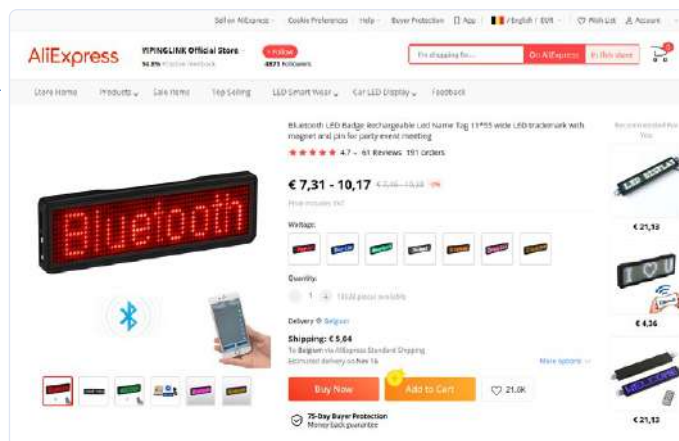
Het lezen van het kenmerk "0xfec9" in nRF Connect for Mobile levert dezelfde waarde op als de fabrikantsspecifieke gegevens. Lezen van 0xfee1 retourneert geen gegevens en abonneren op de meldingen levert ook niets op. De Characteristic User Description van 0xfee1 is "Data." Deze methode levert niets bruikbaars op.

De mobiele app decompileren

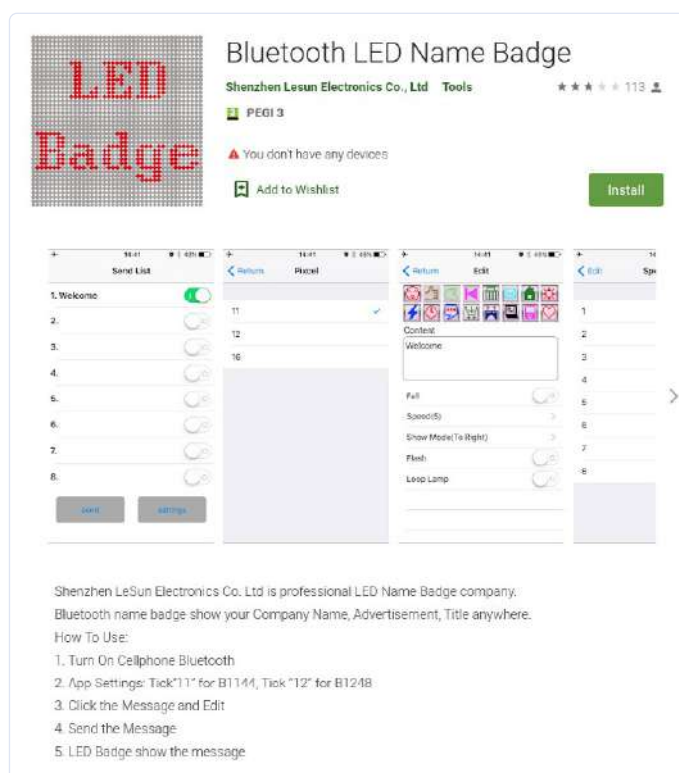
Aangezien duidelijk is dat de Bluetooth LED Name Badge-app voor Android met de LED-badge kan communiceren, zullen we nu eens kijken wat de app eigenlijk doet. We doen dit door de app te decompileren en de werking van de broncode uit te pluizen.

Download het APK-bestand voor de Android-app met behulp van een externe APK-downloadersite zoals [2]. Plak gewoon de Google Play Store-URL [3] van de app in het zoekveld van APKPure. Vervolgens kun je het bestand downloaden. Dit APK-bestand bevat de Dalvik-bytecode die Android kan uitvoeren. Om te begrijpen wat de app doet, hebben we de broncode nodig. Terwijl de ontwikkelaar zijn broncode naar Dalvik heeft gecompileerd, kunnen we dit andersom doen met een decompiler zoals jadx [4], die Dalvik-bytecode omzet in Java-broncode. Met de meest recente release voor Windows [5] kunnen we de grafische interface starten door te dubbelklikken op het **jadx-gui**-bestand in de **bin**-directory. Start op andere besturingssystemen (Linux, MacOS) de grafische interface door bin/jadx-gui uit te voeren vanaf de opdrachtregel.

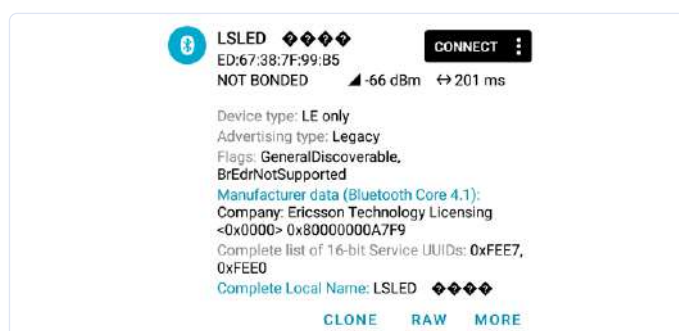
Open het gedownloade APK-bestand. Het programma decompileert nu de app en toont links een boomstructuur van de packages en Java-bestanden. Nu begint de zoektocht. We hebben al een paar aanknopingspunten: de UUID's van de services en karakteristieken die gevonden worden in nRF Connect for Mobile. In het menu **Navigation / Text search** kun je een paar zoektermen invoeren.



Figuur 1. Deze Bluetooth LED-badge lijkt een interessante kandidaat voor reverse-engineering!



Figuur 2. De "Bluetooth LED Name Badge app" in de Google Play Store kan commando's naar de LED-badge sturen.



Figuur 3. De Bluetooth LED-badge in nRF Connect for Mobile.



Listing 1.

```
public static final String UUID_CHARACTERISTICS_WRITE = "fee1";
public static final String UUID_SERVICE = "fee0";
```



Listing 2.

```
public static byte[] get64(List<SendContent> list, int i) {
    Iterator<SendContent> it = list.iterator();
    while (it.hasNext()) {
        Log.d("abcdef", "get64-----SendContent:" + it.next().toString());
    }
    byte[] bArr = new byte[64];
    bArr[0] = 119;
    bArr[1] = 97;
    bArr[2] = 110;
    bArr[3] = 103;
    bArr[4] = 0;
    bArr[5] = 0;
    bArr[6] = getFlash(list);
    bArr[7] = getMarquee(list);
    byte[] modeAndSpeed = getModeAndSpeed(list);
    for (int i2 = 0; i2 < 8; i2++) {
        bArr[i2 + 8] = modeAndSpeed[i2];
    }
    byte[] msgLength = getMsgLength(list, i);
    for (int i3 = 0; i3 < 16; i3++) {
        bArr[i3 + 16] = msgLength[i3];
    }
    bArr[32] = 0;
    bArr[33] = 0;
    bArr[34] = 0;
    bArr[35] = 0;
    bArr[36] = 0;
    bArr[37] = 0;
    byte[] date = getDate();
    for (int i4 = 0; i4 < 6; i4++) {
        bArr[i4 + 38] = date[i4];
    }
    for (int i5 = 0; i5 < 19; i5++) {
        bArr[i5 + 44] = 0;
    }
    bArr[63] = 0;
    return bArr;
}
```

Dat is makkelijker gezegd dan gedaan. De code vermeldt de 0xfee7-service of de kenmerken ervan niet! Het vermeldt wel de andere service en zijn kenmerk in de klasse `com.yannis.ledcard.ble.BleDevice`. Relevante delen ervan zijn te zien in **listing 1**.

In de code kun je nu zoeken naar deze twee constanten:

```
UUID_CHARACTERISTICS_WRITE
UUID_SERVICE
```

De eenvoudigste manier om dit te doen is waarschijnlijk door met de rechter muisknop op de naam van de constante te klikken en vervolgens **Find usage** te selecteren. Klik vervolgens op een van de gevonden instanties om het bijbehorende bestand op die plaats te openen.

Tijdens het zoeken naar enkele andere aanwijzingen in de code, kwam ik een veelbelovende codefragment tegen dat afbeeldingen en weergavemodi in de klasse `com.yannis.ledcard.util.LedDataUtil` betreft. Concreet vond ik de Java-methode van **listing 2** die een soort header voor gegevens lijkt te creëren.

We zien hier een vaste header (6 bytes), modus en snelheid, berichtlength en datum. Hij wordt aangeroepen door een `getSendHeader()` in de genoemde klasse. Het was niet voldoende om uit te vinden wat de app doet door alleen maar verder in deze code te duiken. Dus begon ik de app met de LED-badge te gebruiken terwijl ik het dataverkeer tussen beide apparaten afluisterde. Voor een beter begrip kon ik nu de opgedane kennis uit de broncode van de app combineren met live-dataverkeer.

Afluisteren van het BLE-verkeer tussen badge en app

Wireshark [6] vangt het live Bluetooth-verkeer van je telefoon op met behulp van de Android Debug Bridge. Sluit je telefoon aan op je computer met een USB-kabel en sta de debug-verbinding op je telefoon toe. Start Wireshark. Dit zou **Android Bluetooth Btsnoop Net** moeten tonen als een van de beschikbare interfaces. Na dubbelklikken op de interface, zie je live Bluetooth-verkeer voorbij komen! Een nog betere optie is om de nRF Sniffer voor Bluetooth LE plug-in voor Wireshark [7] te gebruiken, waarbij een nRF52840-dongle wordt gebruikt als het hardwaregedeelte van de 'sniffer'.

Druk op de BLE-badge twee keer op de onderste knop totdat het Bluetooth-pictogram wordt weergegeven. Selecteer vervolgens het apparaat in Wireshark zodat alleen pakketten van en naar dat specifieke apparaat worden weergegeven.

Installeer "Bluetooth LED Name Badge" op je Android-telefoon en open de app. In Wireshark zie je dat je telefoon een scan request doet en de BLE-badge antwoordt met de apparaatnaam in het scan response. Selecteer het apparaattype in de app. Voor de LED-badge van 11 x 55 pixels is dat type 11. Tik op **yes**.

De app toont een lijst met te verzenden berichten. De standaardconfiguratie is om één bericht te verzenden: **Welcome**. Tik op **Send**. De app maakt nu verbinding met de LED-badge en toont het welkomstbericht op het display (**figuur 4**).

In Wireshark zien we een CONNECT_REQ-pakket gevolgd door een verzoek om attributen. Dan komen er een paar Write Requests en reacties. Je krijgt een beter overzicht door met de rechter muisknop op de Write Request-opcode in het Bluetooth Attribute Protocol-ge-deelte van de packet-details te klikken en vervolgens **Apply as Filter / Selected** te kiezen. Je kunt ook het volgende invoeren: `btatt.opcode == 0x12` als displayfilter.

In dit geval heeft de app negen Write Request-packets verzonden, elk met 16 bytes aan gegevens, zoals hieronder weergegeven:

```
77 61 6e 67 00 00 00 00 30 30 30 30 30 30 30 30
00 07 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 e5 0a 18 0c 37 25 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 c6 c6 c6 c6 d6 fe ee c6 82 00 00 00 00 00 7c
c6 fe c0 c6 7c 00 00 38 18 18 18 18 18 18 3c
00 00 00 00 00 7c c6 c0 c0 c6 7c 00 00 00 00 00
7c c6 c6 c6 c6 7c 00 00 00 00 00 ec fe d6 d6 d6
c6 00 00 00 00 00 7c c6 fe c0 c6 7c 00 00 00 00
```

Als je de Java-methode `get64()` uit de vorige paragraaf bekijkt, herken je de delen van de header: 77 61 6e 67 (hexadecimaal equivalent van decimaal 119 97 110 103). Dan krijgen we twee keer 00 en dan weer twee keer 00, wat de waarden zijn voor respectievelijk `getFlash()` en `getMarquee()`. De volgende acht bytes zijn allemaal 0x30; deze staan voor de modus en snelheid. De volgende 16 bytes zijn de berichtlengte. Omdat de app een lijst van acht berichten toont, zou dit de lijst met de lengtes van deze berichten kunnen zijn. De eerste twee bytes hier zijn 00 07, wat inderdaad het aantal tekens is in de tekst **Welcome**. De veronderstelling is dat de volgende 14 bytes de lengte vertegenwoordigen van de volgende zeven berichten (in dit geval geen).

Dan zijn er zes nullen; de volgende zes bytes vertegenwoordigen de datum, in dit geval e5 0a 18 0c 37 25. Omgerekend naar decimaal is dat 229 10 24 12 55 37. Ik heb deze app uitgevoerd op 24 oktober 2021 om 12:55:37. De maand, dag en tijd kloppen. De header eindigt met 20 stuks 00.

Hierna vertegenwoordigen vijf pakketten van 16 bytes op de een of andere manier de zeven tekens van de **Welcome**-tekst. Laten we eens kijken hoe het werkt, te beginnen met een eenvoudiger bericht. Open de app opnieuw en klik op het eerste bericht. Wijzig de tekst in **W** en schakel **Flash** in. Keer terug naar het hoofdscherm, klik op het tweede bericht en voeg deze tekst toe: **e**. Stel voor dit tweede bericht de snelheid in op 8, de modus op **right** en schakel **Marquee** in. Keer terug naar het hoofdscherm. Schakel vervolgens de schuifregelaar naast het tweede bericht in en klik op **Send** terwijl je de Bluetooth Attribute Protocol-packets in Wireshark in het oog houdt.

De LED-badge toont nu de letter W die naar links beweegt en aan en uit knippert. Hierna toont het de letter e die met dubbele snelheid naar rechts beweegt en een frame van bewegende pixels eromheen. In dit geval stuurde de app zes Write Request-packets, elk met 16 bytes aan data – zie hieronder:



Figuur 4. De Bluetooth LED-badge toont een welkomstbericht.

```
77 61 6e 67 00 00 01 02 30 71 30 30 30 30 30 30
00 01 00 01 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 e5 0a 18 10 06 13 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 c6 c6 c6 c6 d6 fe ee c6 82 00 00 00 00 00 7c
c6 fe c0 c6 7c 00 00 00 00 00 00 00 00 00 00 00
```

Het eerste packet begint met dezelfde 6 bytes, maar dan is de waarde van `getFlash()` 01 en die van `getMarquee()` 02. Dit komt doordat we **Flash** hebben ingeschakeld voor het eerste bericht en **Marquee** voor het tweede bericht.

De volgende byte is nog steeds 0x30, omdat we niets hebben veranderd aan de modus en snelheid van het eerste bericht. Maar de byte hierna luidt nu 0x71. We hebben de snelheid gewijzigd in 8 en de modus in Right. Dus de linker nibble van die byte codeert blijkbaar de snelheid (snelheid 4 is gecodeerd als 3 en snelheid 8 als 7) en de rechter nibble de modus. Het tweede packet toont 00 01 als de lengte van het eerste bericht en hetzelfde voor de lengte van het tweede bericht. Dit bevestigt onze veronderstelling. Dan is de datum gecodeerd.

Er zijn nu nog twee packets over. Converteer de hexadecimale waarden naar binair en formatteer ze byte voor byte en op volgorde, en splits ze vervolgens in elf rijen (ja, want het scherm is 11 pixels hoog), zoals hieronder:

```
00000000
11000110
11000110
11000110
11000110
11010110
11111110
11101110
11000110
10000010
00000000

00000000
00000000
00000000
00000000
01111100
11000110
11111110
11000000
11000110
01111100
00000000
```



Listing 3.

```
"""Find BLE LED badges.

Copyright (c) 2022 Koen Vervloessem
SPDX-License-Identifier: MIT

"""

import asyncio
from bleak import BleakScanner

num_devices = 0

def device_found(device, advertisement_data):
    """Show device details if it's a BLE LED badge."""
    global num_devices
    if device.name.startswith("LSLED"):
        num_devices += 1
        print(
            f"() - RSSI: "
        )

async def main():
    """Scan for BLE devices."""
    print("Searching for LED badges...")
    scanner = BleakScanner()
    scanner.register_detection_callback(device_found)
    await scanner.start()
    await asyncio.sleep(5.0)
    await scanner.stop()
    if not num_devices:
        print("No devices found")

if __name__ == "__main__":
    asyncio.run(main())
```

Tegen het einde worden ter opvulling ('filler') nullen gebruikt om een blok van 16 bytes op te vullen (hier niet getoond). Als je dezelfde decodering toepast op het eerste bericht met **Welcome** en vervolgens de bitmaps van de letters naast elkaar legt, zie je een 1 voor elke ingeschakelde pixel en een 0 voor elke uitgeschakelde, zoals hier:

[illegible]

We hebben nu een vrij goed idee van het formaat van de gegevens die naar de LED-badge moeten worden gestuurd om tekst op het display weer te geven. Laten we dit nu allemaal combineren en een Python-script schrijven om eigen afbeeldingen naar de LED-badge te sturen.

Willekeurige afbeeldingen naar de LED-badge schrijven met Bleak

Met de Bluetooth LED-naambadge kun je ook enkele voorgedefinieerde kleine 11 x 11-bitmaps verzenden. Maar nu je weet in welk formaat je data moet verzenden, kun je bijvoorbeeld ook een willekeurige bitmap sturen die het 11 x 55-display helemaal vult. Laten we dit doen met behulp van Bleak [8], een platformonafhankelijke Python-bibliotheek voor BLE. Tik in:

```
pip3 install bleak
```

Laten we eerst een script maken dat gedurende vijf seconden naar BLE-badges scant op basis van hun naam en vervolgens hun Bluetooth-adressen en RSSI's toont – zie **listing 3**.

Als je dit uitvoert met twee LED-badges die beide in de luister-modus staan, toont het script:

```
$ python3 find_led_badge.py
Searching for LED badges...
ED:67:38:80:0D:E2 (LSLED) - RSSI: -74
ED:67:38:7F:99:B5 (LSLED) - RSSI: -83
```

Laten we nu voortbouwen op de kennis die we in de vorige paragraaf hebben opgedaan. Bekijk dit script dat de juiste commando's naar de LED-badge schrijft om een afbeelding weer te geven – zie **listing 4**. Naast Bleak gebruikt dit script ook de Pillow-bibliotheek [9]:

```
pip3 install Pillow
```

De eerste vier BLE-commando's zijn hardgecodeerd. In het eerste commando definieert 0x34 de modus en snelheid voor het eerste bericht. De '4' staat voor statische modus: het bericht wordt weergegeven als een stilstaand beeld. In de tweede opdracht definieert 0x07 de lengte van het bericht als het aantal 8-pixel-tekens. Het display is 55 pixels breed en 7 keer 8 is gelijk aan 56, dus we moeten de afbeelding coderen als 7 tekens. De volgende twee commando's zouden de huidige datum moeten coderen, maar dat is niet echt nodig. Stuur gewoon 16 nullen voor elk van deze commando's.

`chunks()` is een hulpfunctie om een byte-array te splitsen in een lijst met byte-arrays van elk 16 bytes. Dit wordt gebruikt in de functie `commands_for_image()` die een afbeeldingsbestand converteert naar bytes die tekens op het scherm vertegenwoordigen. Aan het einde van de functie worden deze bytes opgesplitst in blokken van 16 bytes. `commands_for_image()` opent een afbeelding met Pillow en laadt de pixels. Voor elk van de zeven tekens op het scherm codeert de functie het teken als 11 bytes: één voor elke rij. Al deze bytes worden toegevoegd aan `image bytes`.

Aan het einde van de afbeelding wordt de byte-array opgevuld met extra bytes, zodat het een veelvoud is van 16 bytes. En uiteindelijk wordt het opgesplitst in hapklare stukken van 16 bytes.

De functie `main()` veegt alles bij elkaar. Hij maakt verbinding met het apparaat en maakt een lijst met commando's: de vier commando's van de header worden uitgebreid met de commando's voor de afbeelding. Vervolgens wordt elk van deze opdrachten naar het kenmerk geschreven met UUID 0xfef1. De hoofdcode onderaan controleert of je twee



Listing 4.

```
"""Display an image on a BLE LED badge.
Copyright (c) 2022 Koen Vervloesem
SPDX-License-Identifier: MIT
"""

import asyncio
import sys
from PIL import Image
import bleak

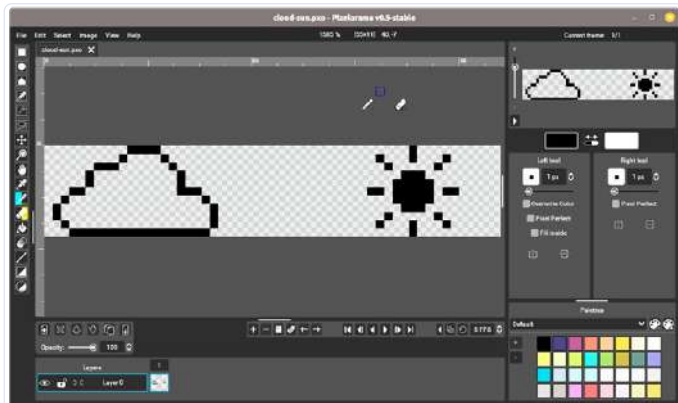
WRITE_CHAR_UUID = "0000fee1-0000-1000-8000-00805f9b34fb"
COMMAND1 = bytes([0x77, 0x61, 0x6E, 0x67, 0x00, 0x00, 0x00, 0x00, 0x34, 0x30, 0x30, 0x30, 0x30, 0x30, 0x30, 0x30])
COMMAND2 = bytes([0x00, 0x07, *(14 * [0x00])])
COMMAND3 = bytes(16 * [0x00])
COMMAND4 = bytes(16 * [0x00])

def chunks(lst, n):
    """Yield successive n-sized chunks from lst."""
    for i in range(0, len(lst), n):
        yield lst[i : i + n]

def commands_for_image(image):
    """Return commands to show an image on the BLE LED badge."""
    image_bytes = bytearray()
    with Image.open(image) as im:
        px = im.load()
        for i in range(7): # 7x8 = 56 -> 7 bytes next to each other
            for row in range(11):
                row_byte = 0
                for column in range(8):
                    try:
                        pixel = int(
                            px[(i * 8) + column, row][3] / 255
                        )
                    except IndexError:
                        pass # Ignore the 56th pixel in a full row
                row_byte = row_byte | (pixel << (7 - column))
            image_bytes.append(row_byte)
    # Fill the end with zeroes to have a multiple of 16 bytes
    image_bytes.extend(bytes(16 - len(image_bytes) % 16))
    # Split image bytes into 16-byte chunks
    return list(chunks(image_bytes, 16))

async def main(address, filename):
    """Connect to BLE LED badge and send commands to show an image."""
    try:
        async with bleak.BleakClient(address) as client:
            commands = [COMMAND1, COMMAND2, COMMAND3, COMMAND4]
            commands.extend(commands_for_image(filename))
            for command in commands:
                await client.write_gatt_char(WRITE_CHAR_UUID, command)
    except asyncio.exceptions.TimeoutError:
        print(f"Can't connect to device .")
    except bleak.exc.BleakError as e:
        print(f"Can't write to device : ")

if __name__ == "__main__":
    if len(sys.argv) == 3:
        address = sys.argv[1]
        filename = sys.argv[2]
        asyncio.run(main(address, filename))
    else:
        print(
            "Please specify the BLE MAC address and image filename."
        )
```

Figuur 5. Pixelorama is een open-source 2D-afbeeldingseditor.



Figuur 6. Door de juiste BLE-commando's te verzenden, kun je opscheppen met je eigen afbeeldingen van 11 x 55 pixels op de LED-badge.

argumenten op de opdrachtregel hebt opgegeven. De eerste is toegewezen aan het Bluetooth-adres en de tweede aan de bestandsnaam. Voordat je deze code uitvoert, moet je een afbeelding maken van exact 11 x 55 pixels, met behulp van een pixeleditor zoals Pixelorama [10]. Vul een pixel voor elke LED die je op het display wilt laten oplichten. **Figuur 5** geeft een voorbeeld van een afbeelding van een wolk en zon die ik in Pixelorama heb gemaakt.

Exporteer de afbeelding als een .png-bestand. Zet nu de LED-badge in Discovery-modus door twee keer op de knop onderaan te drukken, totdat het Bluetooth-pictogram verschijnt. Voer het Python-script uit met twee argumenten; eerst het Bluetooth-adres van de badge en vervolgens de bestandsnaam van de afbeelding:

```
$ python3 display_led_badge.py ED:67:38:7F:99:B5 cloud-sun.png
```

Tot slot

De LED-badge zou nu 'jouw' afbeelding moeten weergeven (**figuur 6**). Het is dus gelukt: je hebt met succes de BLE-badge 'gekraakt' en kunt deze nu met je eigen code gebruiken. Het script dat ik hier heb gepresenteerd is zeker voor verbetering vatbaar, vooral om het algemener toegankelijk te maken. Ik hoop dat het je inspireert om soortgelijke reverse engineering toe te passen op andere BLE-apparaten. 

220439-03

Vragen of opmerkingen?

Hebt u technische vragen of opmerkingen naar aanleiding van dit artikel? Stuur een e-mail naar de auteur via koen@vervolesem.eu of naar de redactie van Elektor via redactie@elektor.com.



GERELATEERDE PRODUCTEN



> **K. Vervloesem, Develop your own Bluetooth Low Energy Applications, Elektor 2022 (boek, SKU 20200)**
www.elektor.nl/20200

Dit boek wordt met een GRATIS
nRF52840 USB-dongle geleverd!

> **K. Vervloesem, Develop your own Bluetooth Low Energy Applications, Elektor 2022 (e-boek, SKU 20201)**
www.elektor.nl/20201

WEB LINKS

- [1] nRF Connect for Mobile-app: <https://www.nordicsemi.com/Products/Development-tools/nrf-connect-for-mobile>
- [2] APK-bestand : <https://apkpure.com>
- [3] De app in Google Play Store: <https://play.google.com/store/apps/details?id=com.yannis.ledcard>
- [4] De jadx-decompiler : <https://github.com/skylot/jadx>
- [5] Laatste release: <https://github.com/skylot/jadx/releases>
- [6] Wireshark: <https://www.wireshark.org>
- [7] nRF Sniffer for Bluetooth LE : <https://www.nordicsemi.com/Products/Development-tools/nrf-sniffer-for-bluetooth-le>
- [8] Bleak : <https://bleak.readthedocs.io>
- [9] De Pillow-bibliotheek: <https://pillow.readthedocs.io>
- [10] Pixelorama: <https://orama-interactive.itch.io/pixelorama>
- [11] Info en resources bij het boek: <https://www.elektor.com/develop-your-own-bluetooth-low-energy-applications>
- [12] Code op GitHub: <https://github.com/koenvervolesem/bluetooth-low-energy-applications>

MakePython ESP32 Development Kit

alles in een doosje

Tam Hanna (Slowakije)

Moderne microcontrollers, zoals de ESP32, zijn zo krachtig dat ze in MicroPython geprogrammeerd kunnen worden. Dankzij de krachtige bibliotheken kunt u zo snel een afgerond project realiseren. Met de MakePython ESP32 Development Kit, enerzijds een (Engelstalig) leerboek en anderzijds een hardwarekit, presenteert de bekende Elektor-auteur Dogan Ibrahim nu een starterspakket dat aan de hand van praktijkvoorbeelden illustreert hoe u met MicroPython kunt werken.



We hoeven hier niet uit te leggen dat MicroPython niet de manier is om maximaal efficiënte software te maken. Maar moderne microcontrollers zoals de ESP32 kunnen het qua prestaties gemakkelijk opnemen tegen een 486. En daarom kan het, zeker voor kleine series, heel redelijk zijn om wat snelheid op te offeren en daarbij werk te besparen door een hogere programmeertaal te gebruiken.

Het boek

We beginnen met het Engelstalige leerboek: Het ziet er best flitsend uit, met een (imitatie van) een spiraalbinding.

De eigenlijke didactische structuur begint met een korte uitleg van de ESP32 als geheel. Een Elektor-lezer met uitgebreide of op zijn minst basiskennis over microcontrollers, vind hier een “snel” overzicht van de periferie die Espressif in deze controller levert. Als u geen ervaring hebt met microcontrollers, is de uitleg (op sommige punten) misschien te kort om volledig te begrijpen.

De installatie van de IDE's (Ibrahim introduceert zowel uPyCraft als Thonny, maar werkt verder bijna uitsluitend met Thonny) wordt in detail uitgelegd voor Windows, maar over Linux staat er weinig. Dan volgt een hoofdstuk dat de uitvoering van enkele “kleine” stukjes Python met Thonny illustreert, wie nog geen kennis heeft van de syntax van Python zal hier weinig van begrijpen. Maar de uitleg is meer dan voldoende om de basis van het werken met MicroPython en de IDE te begrijpen.

De eigenlijke presentatie van de 46 projecten in het boek gaat in de klassieke Dogan Ibrahim-stijl: als eerste stap presenteert de professor altijd de “te volbrengen missie”, en dan volgt de code en uitleg over het bouwen van de hardware. Fans van compact coderen zullen zich misschien ergeren aan het feit dat de listings altijd beginnen met een standaard header van meer dan tien regels. Maar de voorbeelden (die niet te ingewikkeld zijn) laten heel duidelijk de essentiële aspecten van het werken met MicroPython zien.



Figuur 1: De kit. De plastic behuizing beschermt de componenten.

Als u dit hebt gelezen, hoeft u niet meer bang te zijn dat u geen idee hebt hoe u bepaalde periferie aan de praat moet krijgen. Wat ik ook heel positief vind, is dat Ibrahim uitlegt hoe een “verprutste” kaart opnieuw met MicroPython-firmware kan worden geladen.

De ontwikkelkaart

De ontwikkelkaart is in combinatie met het boek verkrijgbaar in de kit in **figuur 1**, die ook geschikt is om mee op reis te nemen dankzij de behoorlijk stevige plastic behuizing. Maar laten we de eigenlijke hardware eens bekijken. De auteur heeft door praktische ondervinding geleerd dat vooral ontwikkelaars die niet bekend zijn met embedded systemen een veel prettiger leerervaring hebben als het ontwikkelplatform beschikt over een (lieft volledig grafisch) display.

Elektor lost dat slim op met de rode print die u ziet in **figuur 2**. Het zwarte gebied is geen zwart gat, maar één van die bekende SSD1306-OLED's met een resolutie van 128x64 pixels.

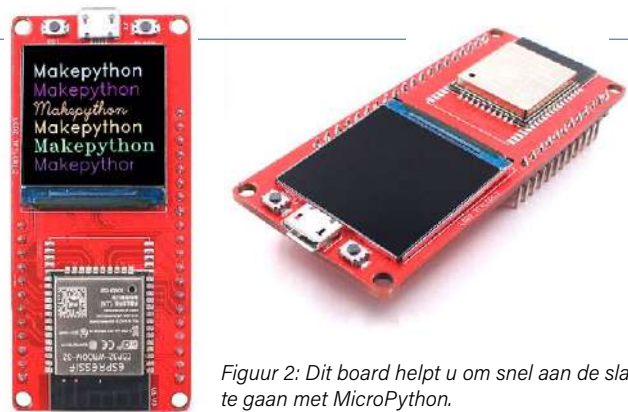
Mijn grootste kritiekpunt is hier niet te zien: De 2,54mm-headers moeten door de gebruiker worden ingesoldeerd. Dat is heel jammer, want als iemand de kit koopt in een winkel en hem meteen meeneemt op vakantie zit hij klem, tenzij hij ook een soldeerbout in de koffer heeft, of de telefoonwinkel om de hoek zo vriendelijk is om hier even mee te helpen.

Voor de rest is er niets aan te merken op de kaart. De gebruikte ESP32-variant is één van de grotere modellen en heeft genoeg geheugen, de MicroUSB-interface garandeert dat een “besturingskabel” wel te vinden is bij een oudere mobiele telefoon. En de kit bevat zelf ook een (heel korte) kabel van behoorlijke kwaliteit.

Een demo

Hoewel het leerboek voornamelijk gericht is op de behoeften van ontwikkelaars die werken met Windows, ga ik bij de volgende stappen gebruik maken van Ubuntu 20.04 LTS, alleen maar voor het gemak. De op [1] beschikbare IDE is een slechte keus, want op alle Ubuntu-versies later dan 16.04 geeft die een foutmelding zoals:

```
ImportError: /tmp/_MEIOhQKhZ/libz.so.1: version
'ZLIB_1.2.9' not found (required by /usr/lib/x86_64-
linux-gnu/libpng16.so.16)
```



Figuur 2: Dit board helpt u om snel aan de slag te gaan met MicroPython.

Thonny is dan een stuk “vriendelijker”. Die is automatisch te installeren met het commando:

```
bash <(wget -O - https://thonny.org/installer-for-linux)
```

en kan dan worden gestart met:

```
/home/tamhan/apps/thonny/bin/thonny
```

Om Thonny te verwijderen, gebruiken we `/home/tamhan/apps/thonny/bin/uninstall`. Alles wat we nog hoeven te doen bij de eerste start is de taalkeuze te bevestigen.

De volgende stap is het verbinden van het evaluatieboard met een workstation. Gelukkig heeft Elektor er aan gedacht om de kaart te leveren met een voorgeconfigureerde MicroPython-runtime. Het is heel handig, dat de kaart ook meteen het display inschakelt, zodat we kunnen zien dat het werkt. Als converter wordt een Silicon Labs CP210x UART bridge chip gebruikt.

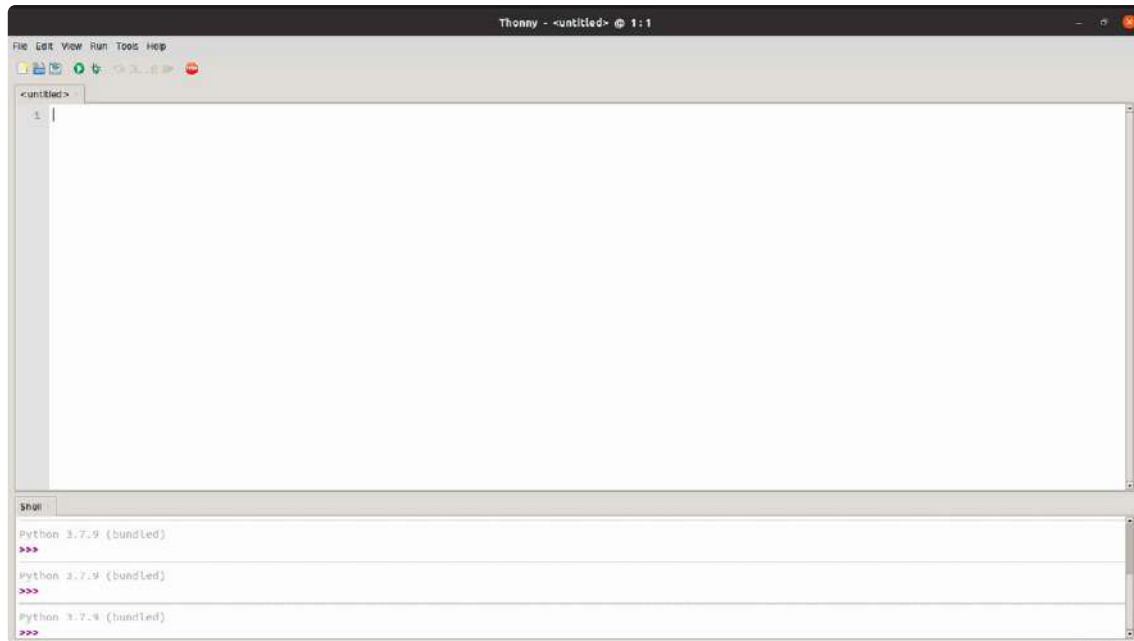
U kunt nu overschakelen naar Thonny (**figuur 3**). Klik in het keuzemenu op de Python-versie die rechtsonder verschijnt en kies *MicroPython (ESP32)*. Thonny gaat dan automatisch op zoek naar ESP32-kaarten. In mijn geval werd de aangesloten kaart automatisch gedetecteerd.

Het is de vraag of het nodig is om Thonny te draaien met superuser-privileges. Mijn eigen user-account is lid van de groep *plugdev*, dus ik kon Thonny gewoon draaien op mijn account en had toegang tot de ESP32.

Een eerste welkom

Het wordt nu tijd om het OLED-display tot leven te brengen. Open de URL: www.makerfabs.com/makepython-esp32-starter-kit.html. Download het archief *MakePython ESP32 Lessons Source Code*. Werken met RAR is niet zo prettig onder Linux. Pak het archief uit naar een gemakkelijk toegankelijke locatie. Voor de volgende stappen hebben we vooral het bestand *ssd1306.py* nodig, dat de driver voor het display bevat.

Klik om te beginnen op *File -> Open* en kies dan de optie *This Computer*. Navigeer dan naar het bestand en laad het in de editor.



Figuur 3: Screenshot van Thonny.

Dan kiezen we *File -> Save* en de optie *MicroPython Board*.

Als Thonny daarbij begint te mopperen dat het backend “busy” is, klik dan op het rode stop-pictogram in de toolbar om hem te stoppen. Sla het bestand dan op in het bestandssysteem onder de bekende naam (alleen als u de originele firmware hebt veranderd). Zo niet, dan staat de file al klaar.

Zoals gebruikelijk bij MicroPython moeten we eerst enkele bibliotheken installeren en ook constanten definiëren met extra informatie over de specificaties van het display:

```
import machine
import ssd1306
import time
WIDTH = const(128)
HEIGHT = const(64)
```

Voor de eigenlijke communicatie met het display gebruiken we I2C (dat is niet heel gebruikelijk, meestal zou je SPI verwachten). We hebben daarvoor de software-I2C-class nodig:

```
pscl = machine.Pin(5, machine.Pin.OUT)
psda = machine.Pin(4, machine.Pin.OUT)
i2c = machine.I2C(scl=pscl, sda=psda)
oled = ssd1306.SSD1306_I2C(WIDTH, HEIGHT, i2c)
```

Tenslotte sturen we tekst naar het display op de volgende manier:

```
while True:
    oled.fill(0)
    oled.text('Hello World!', 10, 0)
    oled.show()
    time.sleep(1)
```

Nu kunt u op het play-pictogram in de IDE drukken en genieten van de tekst op het display van de ontwikkelkaart.

MQTT opzetten

Nu willen we een beetje voorbij de “gewone” projecten in de kit gaan en een geavanceerde taak uitvoeren, om te laten zien dat de kit geschikt is voor moeilijker taken. Om precies te zijn, willen we het bekende MQTT-protocol gaan gebruiken. Het wordt vooral interessant omdat we willen werken met een gloednieuwe versie van de MQTT-broker Mosquitto. Bij [2] vindt u een (aanbevelenswaardige) beschrijving van de problemen die kunnen optreden met de parsers (die om veiligheidsredenen steeds strikter worden geconfigureerd) in interactie met de MicroPython-drivers.

Nu we het toch over MicroPython-drivers hebben: er zijn intussen twee concurrerende implementaties van MQTT voor MicroPython. We gaan hier gebruik maken van de eenvoudigste, die te vinden is op [3]. Download de file:

<https://github.com/micropython/micropython-lib/blob/master/micropython/umqtt.simple/umqtt/simple.py>

En sla die op als *simple.py* op de computer.

Nu moeten we de WiFi-interface van de kaart in station-modus zetten op de volgende manier:

```
...
wlan=network.WLAN(network.STA_IF)
```

Voor het opzetten van de eigenlijke verbinding gebruiken we een methode gebouwd volgens het volgende ontwerp, deze is grotendeels afkomstig van het projectskelet van MakerFabs en implementeert een countdown-teller inclusief het maken van de verbinding.



Listing 1.

```
def connectWiFi():
    i=0
    wlan.active(True)
    wlan.disconnect()
    wlan.connect("ssid", "pass")
    while(wlan.ifconfig()[0]!='0.0.0.0'):
        i=i+1
        oled.fill(0)
        oled.text('connecting WiFi',0,16)
        oled.text('Countdown: '+str(20-i)+'s',0,48)
        oled.show()
        time.sleep(1)
        if(i>20):
            break
    oled.fill(0)
    oled.text('Makerfabs',25,0)
    oled.text('MakePython ESP32',0,32)
    if(i<20):
        oled.text('WIFI connected',0,16)
    else:
        oled.text('NOT connected!',0,16)
    oled.show()
    time.sleep(3)
    return True
```

Natuurlijk moet u de strings voor `wlan.connect` aanpassen aan uw lokale netwerk (zie **listing 1**).

Het maken van de eigenlijke testverbinding is dan heel eenvoudig:

```
connectWiFi()
while True:
    time.sleep(1)
```

Vervolgens willen we een Mosquitto-server. Mosquitto is niet de snelste MQTT-broker, maar hij is open source, gratis en zeer strikt in zijn MQTT-implementatie.

Omdat hij zo veel gebruikt wordt, is er ook een gebruiksklaar image in de Docker Hub, waarmee u de server kunt draaien zonder ingrijpende herconfiguratie van de hostcomputer.

Voor een eerste poging volstaat het volgende commando:

```
docker run -it -p 1883:1883 -p 9001:9001 -v mosquitto.
conf:/mosquitto/config/mosquitto.conf eclipse-mosquitto
```

De parameters `-p 1883:1883` en `-p 9001:9001` zijn belangrijk. Elke Docker-container heeft immers beschikking over alle TCP/IP-poorten van het systeem. Elk van de parameters verbindt één van deze poorten met de netwerkkaart van het hostcomputer. Met de parameter `-v mosquitto.conf:/mosquitto/config/mosquitto.conf` integreren we ook een configuratiefile.



Op dit moment is het een goed idee om een eerste download-poging te doen met een bestaande Internetverbinding. Docker zal normaal gesproken verbinden met de hub, de vereiste componenten downloaden en stoppen met de volgende foutmelding:

```
docker: Error response from daemon: source /var/lib/
docker/aufs/mnt/a22b9e557c37d99eb71f17e7bc6d38df6e7677
d09225376d416612adf0977ccd/mosquitto/config/mosquitto.
conf is not directory.
```

De oorzaak van die fout is dat er nog geen bestand `mosquitto.conf` staat in de hostcomputer.

Die file kunnen we gewoon zelf maken met `gedit`:

```
(base) tamhan@tamhan-thinkpad:~$ gedit mosquitto.conf
```

De overgang van Mosquitto versie 1 naar versie 2 is gepaard gegaan met een heftige verstrakking van de veiligheidsconfiguratie. Een Mosquitto 1.X (gestart met default parameters) aanvaardde server-verzoeken van alle klanten, Mosquitto 2 weigert dat. Maar als u de volgende regels in het configuratiebestand zet, werkt alles weer net zo gemakkelijk (en onveilig) als vroeger:

```
listener 1883
allow_anonymous true
```

De eigenlijke uitvoering kan dan worden gestart met het onderstaande commando. Let op: de `-v` parameter heeft altijd een volledig pad nodig, daarom gebruiken we het `pwd`-commando en “bouwen” het commando op met shell-functionaliteit. `pwd`, staat trouwens voor Print Working Directory, **figuur 4** laat zien hoe het werkt:

```
(base) tamhan@tamhan-thinkpad:~$ docker run -it -p
1883:1883 -p 9001:9001 -v $(pwd)/mosquitto.conf:/
mosquitto/config/mosquitto.conf eclipse-mosquitto
. . .
```

Als de Docker-container de start bevestigt met `1658673183: mosquitto version 2.0.14 running`, hebben we een server draaien.

```
tamhan@tamhan-thinkpad: ~/Desktop/Stuff
Support: command not found
(base) tamhan@tamhan-thinkpad:~$ pwd
/home/tamhan
(base) tamhan@tamhan-thinkpad:~$ cd Desktop/Stuff/
(base) tamhan@tamhan-thinkpad:~/Desktop/Stuff$ pwd
/home/tamhan/Desktop/Stuff
(base) tamhan@tamhan-thinkpad:~/Desktop/Stuff$
```

Figuur 4: Het commando `pwd` geeft de huidige werkdirectory van de shell.



Initialiseren van de MQTT-software op de ESP32

Om te beginnen hebben we enkele constanten nodig om te werken met MQTT. Pas het mee te geven IP-adres in de serverstring aan voor uw lokale situatie:

```
SERVER = "192.168.178.146"
CLIENT_ID = ubinascii.hexlify(machine.unique_id())
TOPIC = b"led"
state = 0
```

We hebben ook een callback-functie nodig die de MQTT-driver altijd aanroept als er een bericht is ontvangen:

```
def sub_cb(topic, msg):
    global state
    print((topic, msg))
    . . .
```

Het eigenlijke opzetten van de MQTT-verbinding is dan relatief eenvoudig:

```
connectWiFi()
c = MQTTClient(CLIENT_ID, SERVER, keepalive=30)
c.set_callback(sub_cb)
c.connect()
c.subscribe(TOPIC)
print("Connected to %s, subscribed to %s topic" % (SERVER,
TOPIC))

while True:
    c.wait_msg()
```

De aanroep `c.subscribe` zorgt dat de MQTT-driver de server vertelt, welke kanalen we willen ontvangen. Het is ook belangrijk om periodiek de methode `c.wait_msg()` aan te roepen om de MQTT-server CPU-tijd te geven voor het verwerken van de informatie die binnenkomt of wordt verzonden. U ziet de volgende output `Connected to 192.168.178.146, subscribed to b'led' topic`.

Nu willen we één van de beschikbare LED's verbinden met GPIO-pen 12. Natuurlijk via een serieweerstand, we gaan ervan uit dat u voldoende kennis hebt van elektronica. Voor de initialisatie van de GPIO-poort is alleen gewone ESP32-code nodig:

```
led = Pin(12, Pin.OUT, value=1)
```

Om binnenkomende berichten te verwerken, moeten we de callback uitbreiden door de string in `topic` te combineren met de constanten voor de diverse commando's:

```
def sub_cb(topic, msg):
    global state
    print((topic, msg))
    if msg == b"on":
        led.value(1)
        state = 1
    elif msg == b"off":
        led.value(0)
        state = 0
    elif msg == b"toggle":
        led.value(state)
```

Voor de eigenlijke test gebruiken we het commando `mosquitto_pub`: Dit is een Linux-tool om commando's rechtstreeks naar een MQTT-server te sturen. De volgende commando's schakelen de LED aan en uit:

```
(base) tamhan@tamhan-thinkpad:~$ mosquitto_pub -t led
-m 'on'
(base) tamhan@tamhan-thinkpad:~$ mosquitto_pub -t led
-m 'off'
```

Als de ESP32 en de server in hetzelfde netwerk zitten, kunt u nu de LED aan- en uitschakelen.

Conclusie en vooruitblik

De MakePython ESP32 Development Kit is een verbluffend compacte gereedschapskist, waarmee Python-ontwikkelaars en/of elektronici snel die twee werelden kunnen verenigen. Vooral omdat alles zo handig verpakt en klaar voor gebruik is, is het een doosje dat ik van harte kan aanbevelen. ◀

220429-03



Related Products

> **MakePython ESP32 Development Kit (SKU 20137)**
www.elektor.nl/20137

WEBLINKS

- [1] uPyCraft IDE: <https://github.com/DFRobot/uPyCraft>
- [2] umqtt.simple en mosquitto 2.0.12: <https://github.com/micropython/micropython-lib/issues/445>
- [3] umqtt.simple : eenvoudige MQTT-client voor MicroPython:
<https://github.com/micropython/micropython-lib/tree/master/micropython/umqtt.simple>

THD-meting met een oscilloscoop en FFT

gemakkelijke berekening van de vervormingsfactor

Sebastian Westerhold (AI5GW) (Duitsland)

Totale harmonische vervorming, kortweg THD, is een belangrijke maat voor de harmonische vervorming van signalen in voedingsnetwerken en elektronische assemblages, zoals audioversterkers. Meestal zijn speciale meetapparaten nodig om de THD te bepalen. Als je echter een digitale oscilloscoop met FFT-functie hebt, kun je met een beetje wiskunde de THD berekenen.

Eerst wat terminologie: harmonischen beschrijven zowel de grondfrequentie als de boventonen van een signaal door hun ordenummers. Boventonen beschrijven alle gehele veelvouden van de grondfrequentie, te beginnen met tweemaal de frequentie van de grondgolf. De eerste harmonische komt overeen met de grondtoon, de tweede harmonische komt overeen met de eerste boventoon.

Er zijn twee belangrijke vertegenwoordigers van de THD-familie, THD_R en THD_F . Voor beide geldt het volgende: Hoe kleiner de THD, hoe lager de harmonische inhoud van het gemeten signaal. Hoe groter de harmonische inhoud, hoe groter de THD. Omdat harmonischen typisch voorkomen als effect van vervorming, is het in feite een maat voor het kwantificeren van niet-lineaire vervorming.

In dit artikel wordt de term "boventonen" gebruikt als het gaat om veelvouden van de grondfrequentie in hun geheel. De term "harmonischen" wordt gebruikt als ofwel de totaliteit van alle frequentiecomponenten (inclusief de grondgolf) wordt bedoeld, ofwel een specifiek veelvoud van de grondfrequentie, gekenmerkt door een volgnummer. Voorbeeld: 3e harmonische = 3 maal de frequentie van de fundamentele golf.

THD_F geeft de verhouding aan tussen alle boventonen en de grondgolf. Deze grootheid wordt bijvoorbeeld gebruikt bij het meten van

harmonischen in voedingsnetten. THD_R daarentegen betekent de verhouding van alle boventooncomponenten tot het totale signaal. Deze grootheid is vooral populair in de geluidstechniek en wordt ook wel "vervormingsfactor" genoemd."

Het onderscheid tussen de twee metrieken is erg belangrijk omdat THD_F en THD_R , hoewel nauw verwant, verschillende resultaten opleveren. In literatuur en datasheets worden de twee waarden vaak met elkaar verward [1]. Op sommige plaatsen vind je alleen de term "THD" zonder verdere definitie van de gebruikte methode, bijvoorbeeld in het datablad van Texas Instruments voor de LM386.

In de praktijk kan THD_F relatief gemakkelijk worden berekend uit de som van de vermogensniveauverschillen van alle boventonen ten opzichte van de grondtoon. Hieruit kan dan weer THD_R berekend worden.

$$THD_F = 100 \cdot \sqrt{\sum_{n=2}^{\infty} 10^{\left(\frac{P_n}{10}\right)}} = 100 \cdot \sqrt{10^{\left(\frac{P_2}{10}\right)} + 10^{\left(\frac{P_3}{10}\right)} + 10^{\left(\frac{P_4}{10}\right)} \dots}$$

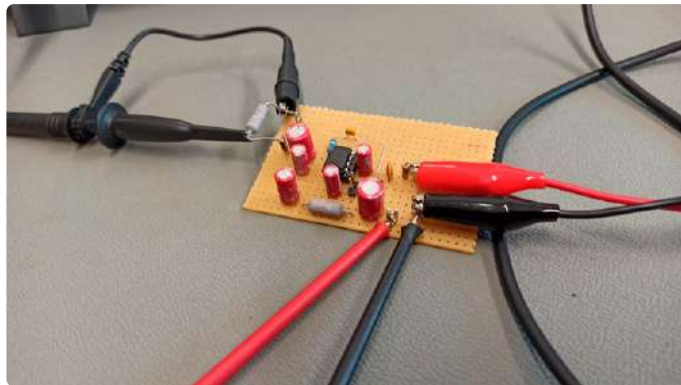
Formule 1: THD_F (in %), n = orde aantal harmonischen, P = vermogensverschil met fundamentele golf in dB

De formule die dit principe beschrijft kan op het eerste gezicht ontmoedigend lijken. In vereenvoudigde termen is THD_F de vierkantswortel van de som van alle spanningsverhoudingen berekend uit de relatieve vermogensverschillen van een oneindig aantal harmonischen ten opzichte van de grondgolf. Op het eerste gezicht lijkt deze zin misschien net zo ontmoedigend als de formule zelf. De materie is echter niet zo ingewikkeld als ze lijkt.

De aldus berekende THD_F -waarde kan dan worden omgezet in de vervormingsfactor met de volgende formule:

$$THD_R = \frac{THD_F}{\sqrt{1 - THD_F^2}}$$

Formule 2: THD_R (in %), THD_F (in %).



Figuur 1: Audioversterker met een LM386N-1 als DUT.

Het verschil tussen THD_F en THD_R neemt af naarmate de waarden zelf kleiner zijn. Onder de 1% is het verschil bijna verwaarloosbaar.

In voedingsnetwerken wordt, om voor de hand liggende redenen, de netfrequentie, bijvoorbeeld 50 Hz, gebruikt als grondfrequentie bij het berekenen van de THD. In de audiotechniek is 1 kHz een gebruikelijke standaard.

In de praktijk is het natuurlijk onmogelijk met een oneindig aantal harmonischen te rekenen. In voedingsnetwerken worden bijvoorbeeld maximaal 50 harmonischen beschouwd. In de audiotechniek ligt het voor de hand om onhoorbare harmonischen te verwaarlozen.

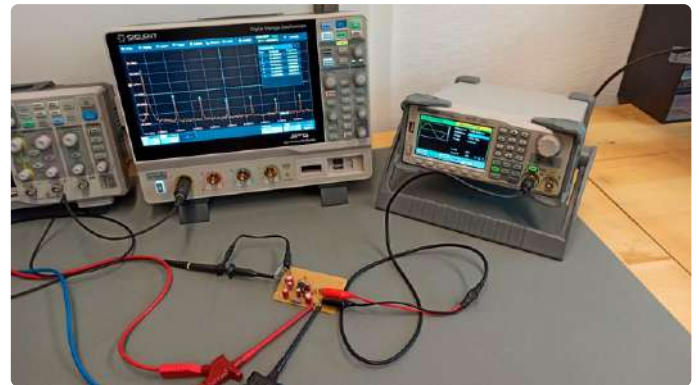
Praktijkvoorbeeld

Om te laten zien dat de saaie wiskundige theorie in de praktijk zo slecht nog niet is, bekijk je het volgende voorbeeld: een sinus signaal met een frequentie van 1 kHz wordt toegevoerd aan de ingang van een versterker (LM386N-1) (Figuur 1). Het uitgangssignaal van de versterker wordt verbonden met een weerstand van $10\ \Omega$ (die dient als ruwe vervanging voor de impedantie van een luidspreker) en via een 1:10 probe naar een Siglent SDS2104X HD oscilloscoop geleid (Figuur 2).

Bekeken in het tijddomein ziet het uitgangssignaal er aanvankelijk vrij schoon en sinusvormig uit (Figuur 3). Deze eerste indruk is echter



Figuur 3: Uitgangssignaal van de DUT in het tijddomein.



Figuur 2: Complete meetopstelling met signaalgenerator, oscilloscoop en DUT.

misleidend, zoals een blik op het FFT-spectrum duidelijk laat zien: de harmonische componenten - hier uitgesneden tot 7 kHz - zijn duidelijk zichtbaar naast de grondtoon op 1 kHz (Figuur 4).

Voor de duidelijkheid heb ik mijn aanvankelijke idee om het aantal beschouwde harmonischen tot in het oneindige te benaderen, althans tot aan de bandbreedtelimiet van de oscilloscoop, van tafel geveegd. Uiteindelijk registreerde ik de vermogensniveaus van de eerste zeven harmonischen, overeenkomend met het vermogensniveau van de grondgolf en de eerste zes boventonen.

De meeste oscilloscopen bieden de mogelijkheid om automatisch de gemeten vermogensniveaus weer te geven in een handige tabel. Vermogensniveaus worden meestal weergegeven in de eenheid dBm. Dit is tienmaal delogaritme bij grondtal 10 van de verhouding van de gemeten waarde ten opzichte van een referentiegrootheid. In het geval van dBm is de referentiegrootheid 1 mW.

Voordat berekeningen kunnen worden gemaakt, moeten de absolute vermogensniveaus worden omgezet in een signaalniveauverschil ten opzichte van de basisgolf. Hier wordt de eenheid dBc (dB ten opzichte van de draaggolf) gebruikt. Voor de omrekening moet het vermogensniveau van de fundamentele golf worden afgetrokken van de vermogensniveaus van alle harmonischen. Bijvoorbeeld, voor de 2e harmonische (= 1e boventoon): $-31,7\ \text{dBm} - 22,1\ \text{dBm} = -53,8\ \text{dBc}$.



Figuur 4: FFT-spectrum van het uitgangssignaal.

Tabel 1: Absolute en relatieve amplitudes van de eerste zeven harmonischen (afgerond).

Ordnummer van de harmonische	Vermogen (dBm)	Relatief vermogen (dBc)
1. (1 kHz) = grondtoon	22,1	0
2. (2 kHz) = 1e boventoon	-31,7	-53,8
3. (3 kHz) = 2e boventoon	-38,7	-60,8
4. (4 kHz) = 3e boventoon	-43,8	-65,9
5. (5 kHz) = 4e boventoon	-63,5	-85,6
6. (6 kHz) = 5e boventoon	-47,0	-69,1
7. (7 kHz) = 6e boventoon	-58,0	-80,1

Tabel 1 geeft een overzicht van de harmonischen met hun volgnummers, de absolute vermogensniveaus in dBm (afgerond) en de berekende vermogensniveauverschillen ten opzichte van de 1e harmonische. De relatieve vermogensniveaus uit Tabel 1 kunnen direct worden ingevoegd in Formule 1.

$$100 \cdot \sqrt{10^{\frac{-53,8}{10}} + 10^{\frac{-60,8}{10}} + 10^{\frac{-65,9}{10}} + 10^{\frac{-85,6}{10}} + 10^{\frac{-69,1}{10}} + 10^{\frac{-80,1}{10}}} \approx 0,232\%$$

De THD_F-waarde van de versterker (beschouwd tot 7 kHz) is dus ongeveer 0,232%. Met de tweede formule kan dan ook THD_R (de vervormingsfactor) worden berekend.

$$THD_R = \frac{0,232}{\sqrt{1 - 0,232^2}} \approx 0,239\%$$

De vervormingsfactor is dus ongeveer 0,239%. De datasheet van de LM386 specificeert een THD van 0,2%. Dit komt aardig in de buurt van de hier berekende waarden. Helaas onthult Texas Instruments niet welk type THD in de datasheet staat, noch hoeveel harmonischen in de meting zijn meegenomen.

Beperkingen en opmerkingen

De hier gepresenteerde methode vindt zijn grenzen in het dynamische bereik van de analoog-digitaal omzetter (ADC) van de gebruikte oscilloscoop. Gewoonlijk zijn dit 8-bits ADC's met een theoretisch dynamisch bereik van ongeveer 48 dB. Deze kunnen gebruikt worden tot een minimale THD van ongeveer 3%.

Oplettende lezers hebben misschien opgemerkt dat de THD die hierboven als voorbeeld is berekend duidelijk onder deze grens ligt. Dit is mogelijk omdat ik een 12bit-oscilloscoop heb gebruikt. Met 12bit-oscilloscopen is het theoretische dynamische bereik ongeveer 72 dB. Dit betekent dat THD-waarden van ongeveer 0,2% binnen de mogelijkheden liggen.

Voor een optimaal gebruik van deze grenzen is echter voorzichtigheid geboden: het werkelijk beschikbare dynamische bereik hangt ook sterk af van de verticale afbuiging! Als de ADC wordt overstuurd, treedt vervorming op; als hij niet genoeg wordt aangestuurd, gaat waardevol dynamisch bereik verloren.

Als je deze methode rechtstreeks wilt uitproberen, vraag je je misschien af waar je referentiesignalen met bekende THD-waarden kunt verkrijgen. Een eenvoudige signaalgenerator kan hierbij helpen: Een redelijk symmetrisch blokvolgsignaal met een duty cycle van 50% heeft een THD_F van ongeveer 48,3%. Met een even symmetrisch driehoekssignaal met dezelfde duty cycle is de THD_F nog steeds ongeveer 12,1%.

Als je wat dieper in de materie wilt duiken en ook de afleiding van de formules wilt begrijpen, kun je meer informatie vinden onder [2] en [3].

220398-03

Vragen of opmerkingen?

Heb je technische vragen of opmerkingen over dit artikel? E-mail de auteur op sebastian@baltic-lab.com of neem contact op met Elektor via redactie@elektor.com.

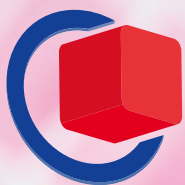


Gerelateerde producten

- > **Siglent SDS1204X-E 4-ch Oscilloscope (200 MHz)**
(SKU 18524)
www.elektor.nl/18524
- > **Siglent SDG1032X 2-ch Signal Generator (30 MHz)**
(SKU 20276)
www.elektor.nl/20276
- > **Adafruit 2.5 W Class D Mono Amplifier (PAM8302)**
(SKU 18745)
www.elektor.nl/18745

WEBLINKS

- [1] Doron Shmilovitz, "On the definition of total harmonic distortion and its effect on measurement interpretation," Power Delivery, IEEE Transactions, S. 526 – 528, 2005: <https://doi.org/10.1109/TPWRD.2004.839744>
- [2] Sebastian Westerhold, "Total Harmonic Distortion (THD) from dBc," 2022: <https://baltic-lab.com/thd/>
- [3] Sebastian Westerhold, "Total Harmonic Distortion (THD) analysis utilizing the FFT capabilities of modern digital storage oscilloscopes," 2022: <http://dx.doi.org/10.5281/zenodo.6969825>



embeddedworld2023

Exhibition & Conference

... it's a smarter world



JOIN THE EMBEDDED
COMMUNITY

14–16.3.2023



Get your
free ticket now!

embedded-world.com/voucher

Use the voucher code **GG4ew23**

Media partners

Markt & Technik
Das unabhängige Wochenblatt für Elektronik

Elektronik

SmarterWorld
Solutions for a Smarter World

DESIGN &
ELEKTRONIK
KNOW-HOW FÜR ENTWICKLER

Elektronik
automotive

•medical-design

computer &
automation

elektroniknet.de

NÜRNBERG MESSE



Allesziende machines

de technologie achter moderne industriële visionsystemen

Stuart Cording (Elektor)

Met de komst van CMOS-beeldsensoren zijn camera's met een laag energieverbruik nu gemakkelijk in massa te produceren in een rendabel halfgeleider-productieproces. Tegenwoordig hebben we allemaal minstens twee van dergelijke onderdelen in onze smartphones. Dankzij steeds krachtigere microcontrollers zijn beeldopname en objectbewerking voor iedereen beschikbaar, en ze ondersteunen zelfs 3D-metingen in industriële sensoren.

In de huidige maatschappij zijn camera's nooit ver weg. Als jij er geen op zak hebt, is de kans groot dat de persoon die naast je in de trein zit er wel een heeft. En zo niet, dan is er waarschijnlijk een geïntegreerd in het treinstel, het station of het beveiligingssysteem in het café waar je onderweg langs liep. Camera's zijn zo alomtegenwoordig geworden dankzij CMOS-beeldsensoren, een technologie die de kosten, het stroomverbruik en de afmetingen aanzienlijk heeft verminderd in vergelijking met de oude CCD-technologie die ze vervangen hebben. Terwijl het vastleggen van beelden en video één taak is, heeft de snelle groei van de rekenkracht van computers in het afgelopen decennium het mogelijk gemaakt dat binnenkomende visuele gegevens in real-time worden geanalyseerd. Dit maakt geautomatiseerde inspectie mogelijk voor diverse industriële toepassingen, zoals het

opsporen van verkeerd geplaatste onderdelen op printplaten, het lokaliseren van onderdelen op transportbanden en zelfs het beoordelen van de kwaliteit van een lasverbinding.

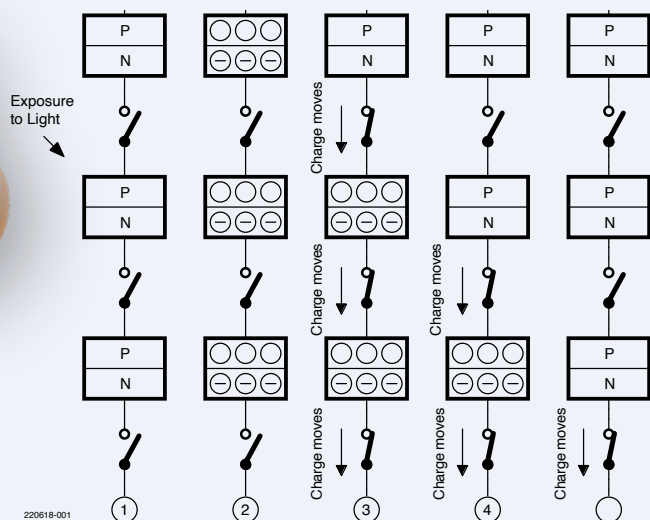
CCD's - beeldtechniek uit het verleden

Beeldsensoren gebruiken een serie fotodiodes om fotonen om te zetten in elektrische stroom. In CCD's is een reeks fotodiodes via schakelaars met elkaar verbonden. Nadat het licht op het beeldoppervlak is gevallen, worden de ladingen in elke pixel één voor één door de string geleid (lading doorgeven) naar de rand van de sensor (**figuur 1**). Hier wordt de lading omgezet in een spanning alvorens door een analoog/digitaal-converter (ADC) te worden omgezet in een digitale waarde. CCD-sensoren worden vervaardigd in een metaaloxide-halfgeleiderproces (MOS), dat

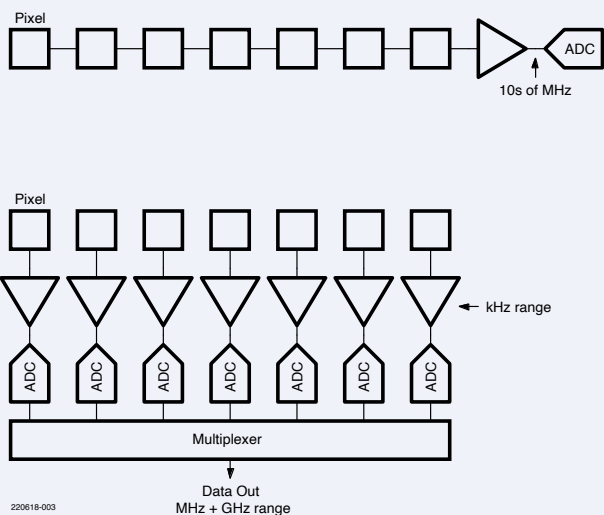
verschilt van de complementaire MOS-technologie (CMOS) die voor de meeste silicium-componenten wordt gebruikt.

De lange levensduur van CCD als sensortechnologie hangt samen met de hoge kwantum-efficiëntie, een maatstaf voor hoe goed het apparaat fotonen omzet in elektronen, die kan oplopen tot 95%. Naast commerciële en consumenten-cameratoepassingen werden CCD's gebruikt in wetenschappelijke apparatuur, zoals spectrometers en zelfs de Hubble-telescoop [1]. Voor de evaluatie van de beelden die ze vastlegden waren echter externe systemen nodig.

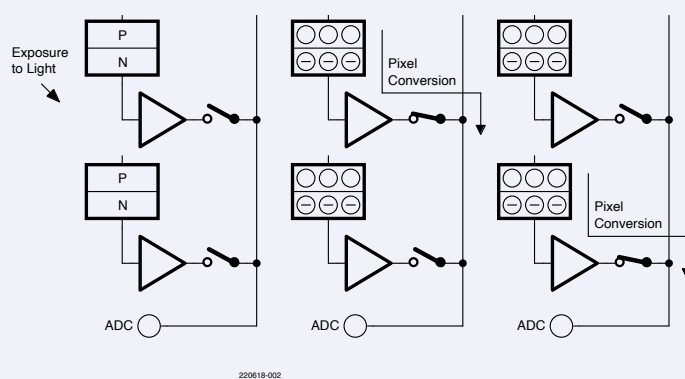
CMOS-beeldsensoren kwamen in de jaren negentig op als een levensvatbare technologie. Aanvankelijk was de beeldkwaliteit slecht, met een lage resolutie, ruis en een slechte kleurdefinitie. Door noodzakelijke verschillen in ontwerp waren de fotodiodes in deze sensoren veel kleiner dan die in een CCD-sensor met een vergelijkbaar aantal pixels. In plaats van een oplaadcircuit te implementeren, had elke pixel zijn eigen versterker nodig, waardoor het oppervlak van elke pixel kleiner werd (**figuur 2**). Maar dankzij de immense ontwikkeling en het onderzoek, gedreven door duidelijke marktbehoeften, is het kwaliteitsverschil tussen beide technologieën tegenwoordig minimaal. Het productiegemak van een CMOS-fabricageproces heeft ook de prijzen gedrukt, waardoor de integratie is verbeterd



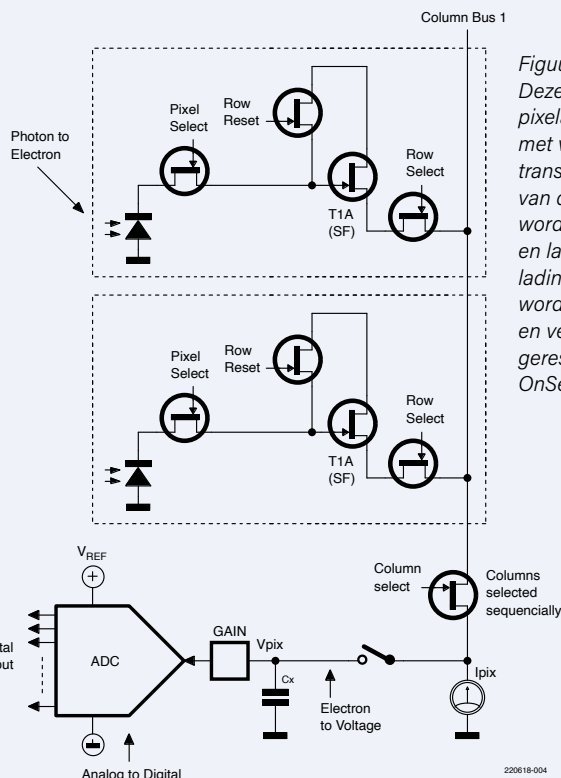
Figuur 1: CCD-beeldsensoren gebruiken een ladingsoverdracht tussen pixels om hun waarden uit te lezen. (Bron: Canon Inc.)



Figuur 3: Omdat CMOS-beeldsensoren grootschalig parallel zijn, hoeft het analoge front-end niet zo snel te zijn als bij CCD-sensoren. (Bron: Teledyne Digital Imaging Inc.)



Figuur 2: CMOS-beeldsensoren gebruiken een ADC om de waarde van elke pixel afzonderlijk om te zetten. (Bron: Canon Inc.)



Figuur 4: Deze CMOS-pixelarchitectuur met vier transistoren is een van de vele die worden gebruikt en laat zien hoe de lading van de pixel wordt geëvalueerd en vervolgens gereset. (Bron: OnSemi)

en veel van de noodzakelijke beeldverwerking op de chip van de sensor zelf zit. Bovendien is het stroomverbruik ongeveer 100 maal minder dan dat van een gelijkwaardige CCD-sensor, iets wat essentieel is voor de huidige draagbare apparaten die op batterijen werken.

Vermindering van ruis in pixels en verhoging van de beeldsnelheid wordt bereikt door een grootschalige parallelle gegevensomzetting, waarbij meerdere ADC's de ladingen in een rij pixels tegelijk omzetten. Vergeleken met CCD's kunnen daardoor versterkers en ADC's met een kleinere bandbreedte worden gebruikt (figuur 3). Pixelontwerpen variëren, maar bestaan meestal uit een versterker met hoge impedantie verbonden met de fotodiode en

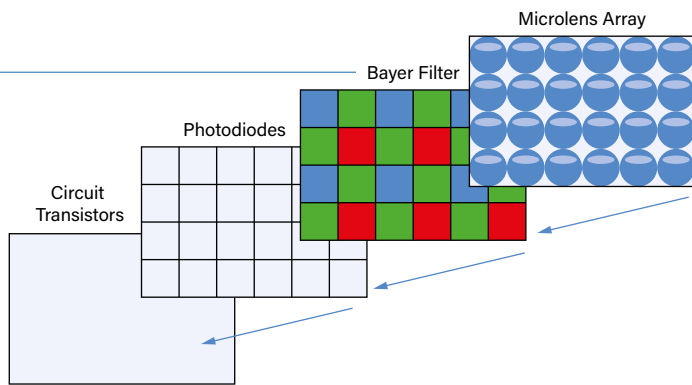
schakelaars voor rij- en kolomkeuze. Tenslotte is er een schakelaar om de lading van de pixel te wissen (figuur 4).

Kleur toevoegen

Het gebruik van het populaire CMOS-productieproces heeft de kosten van beeldsensoren weliswaar verlaagd, maar ze zijn allesbehalve normaal als het gaat om de fabricage. De fotodiodes zijn gevoelig voor al het zichtbare licht, zodat elk vastgelegd beeld alleen een grijswaardenbeeld oplevert. Om kleurenbeelden te produceren is een kleurenfilter-array (CFA) nodig. Meestal wordt een Bayer-filter gebruikt, genoemd naar Bryce Bayer van Eastman Kodak, die het uitvond. Het gebruikt rode, blauwe en twee groene kleurenfilters in

een patroon dat ervoor zorgt dat elk vierkant van vier pixels half groen, kwart rood en kwart blauw is. De hogere weging voor groen komt voort uit de fysiologie van het oog, dat groen beter waarneemt [2].

Het resulterende ruwe beeld is zonder nabewerking niet geschikt voor onmiddellijke menselijke consumptie. Dit gebeurt met een proces dat **demosaicing** heet. In het eenvoudigste geval wordt elke pixel omgezet in een RGB-waarde (rood/groen/blauw) op basis van de waarden van specifieke pixels eromheen. Sommige demosaicing-benaderingen, zoals lineaire interpolatie, kunnen echter ongewenste effecten veroorzaken in het uiteindelijke beeld, vooral rond de randen



Figuur 5: In tegenstelling tot gebruikelijke halfgeleiders hebben beeldsensoren bij hun constructie een kleurenfilter en een microlens-array nodig.

van objecten. Om dit tegen te gaan is een aantal algoritmen beschikbaar die proberen de artefacten te minimaliseren en toch een snel eindbeeld te leveren [3].

Een ander verschil met de gebruikelijke chipfabricage is de noodzaak van een microlens-laag. Omdat het lichtgevoelige oppervlak van elke pixel zo klein is, helpt de lens om het binnenkomende licht op de pixel te concentreren, wat op zijn beurt de kwantumefficiëntie verhoogt (**figuur 5**).

Elektronische sluiters begrijpen

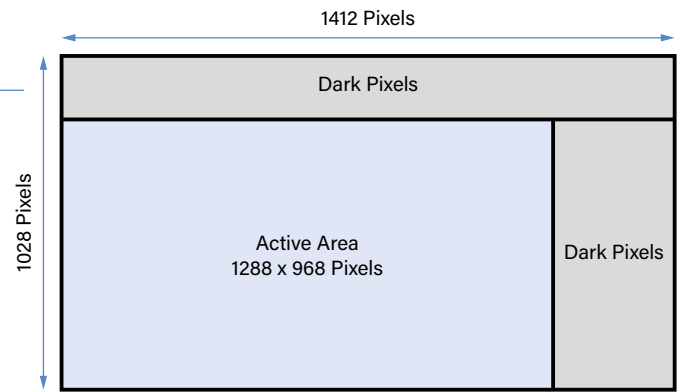
Hoe de sluiters van de beeldsensor werkt kan voor niet-ingewijden ook moeilijk te begrijpen zijn. Een elektronische sluiters wordt gebruikt om de pixels gedurende de juiste tijd bloot te stellen aan licht. Doorgaans zijn er twee mogelijkheden, globaal en rollend, en sommige beeldsensoren ondersteunen ze allebei. Bij een globale sluiters worden alle pixels van de beeldsensor elektronisch aangestuurd om tegelijkertijd belicht te worden. Daarna wordt elke rij pixels omgezet in digitale waarden en doorgegeven aan de hostprocessor die het beeld verwerkt. Bij een rollende sluiters wordt elke rij pixels één voor één aan het licht blootgesteld, waarbij de omzetting voor elke regel onmiddellijk plaatsvindt.

Het beeldverschil dat ontstaat wordt deels beïnvloed door de gebruikte sluiters-methode. Een globale sluiters legt het beeld in één moment vast. Ter vergelijking: de rollende sluiters legt elke lijn vast met een kleine vertraging ten opzichte van de vorige lijn. Wanneer het object dat door de sensor wordt vastgelegd beweegt, kunnen ongewenste effecten in het beeld ontstaan. Als je bijvoorbeeld een locomotief fotografeert die van links naar rechts beweegt, zal het uiteindelijke beeld van links naar rechts vegen vertonen vanaf de bovenkant van het beeld.

Hoewel het voor de hand liggende antwoord zou zijn om een globale sluiters te kiezen, zijn er zoals altijd, meer dingen die begrepen moeten worden. De rollende sluiters methode vereist minder transistoren rond elke pixel, waardoor er meer ruimte overblijft voor het lichtgevoelige deel, wat resulteert in een betere beeldkwaliteit. Het vereenvoudigde ontwerp leidt ook tot minder ruis in de pixel. Tenslotte maakt de rollende sluiters hogere beeldsnelheden mogelijk dan een globaal sluitersontwerp, omdat de bovenste rijen van het volgende beeld kunnen worden belicht terwijl de laatste rijen worden omgezet.

De beslissing wat beter is hangt af van de toepassing; de problemen veroorzaakt door de rollende sluiters kunnen worden opgelost door een flitsers te gebruiken die gesynchroniseerd is met de sluiters. Sommige wetenschappelijke instrumenten vereisen een globale sluiters, terwijl een camera voor visuele inspectie prima uit de voeten kan met een rollende sluiters en gesynchroniseerde flitslichtbron. Een nuttige bron om de problemen in meer detail te begrijpen is geleverd door PCO AG [4].

Een andere potentiële bron van verwarring bij beeldsensoren is het verschil tussen het aantal pixels op de sensor en het aantal dat beschikbaar is voor de toepassing van de gebruiker. De OnSemi AR0130CS [5] is een voorbeeld, een 1/3-inch CMOS digitale beeldsensor met een rollende sluiters. Deze 1,2-megapixel sensor heeft 1280 × 960 actieve pixels in de lijst met belangrijkste prestatieparameters van de datasheet. De werkelijke sensor meet echter 1412 × 1028 pixels met een actief gebied van 1288 × 968. Sommige van de bovenste rijen en een blok van de kolommen aan de rechterkant zijn optisch zwart. Deze worden intern gebruikt om het zwartniveau te controleren en zorgen voor zwartniveaurectie (**figuur 6**).



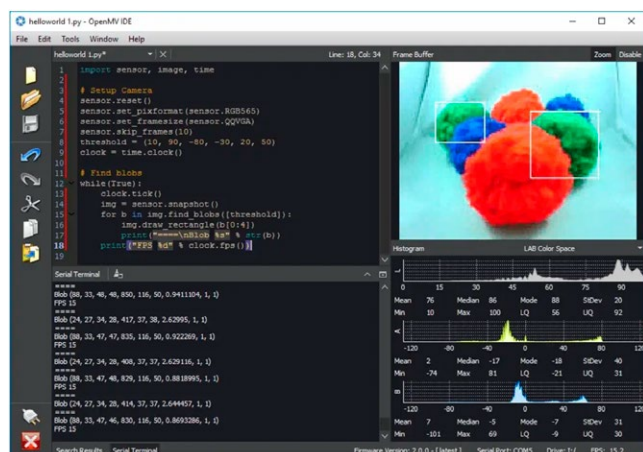
Figuur 6: In een CMOS-beeldsensor kunnen extra pixels worden aangebracht om het zwartniveau te compenseren. (Bron: OnSemi)

Een andere uitdaging bij het verkrijgen van een beeld van goede kwaliteit ligt in de voeding. De PSRR (power supply rejection ratio) van de gebruikte low dropout-regelaars (LDO) moet voorkomen dat er ruis in de voedingsaansluitingen van de CMOS beeldsensor komt. Zo niet, dan neemt het risico van pixelruis toe. Volgens een application note van OnSemi [6] wordt, naarmate de beeldsensoren verder gaan dan 50 megapixels, een PSRR van meer dan 90 dB tot 10 kHz en meer dan 45 dB tussen 1 en 3 MHz aanbevolen. De dynamische belasting kan ook vrij aanzienlijk zijn, met een paar honderd milliampère belastingverandering tijdens rij-overgangen. Hier zijn bulkcondensatoren essentieel om optimale prestaties te bereiken. Tenslotte moet rekening worden gehouden met de uitgangsruis van de LDO ($nV/\sqrt{Hz}$), want alles in het bereik van 10 Hz tot 1 MHz kan resulteren in extra ruis in de pixels, waardoor hun dynamisch bereik afneemt.

Machine vision met microcontrollers

Met hun 32-bits processoren en een megabyte SRAM of meer zijn de grotere microcontrollers van tegenwoordig zeer geschikt voor eenvoudige vision-toepassingen en kunnen ze zelfs enkele nuttige machine learning-algoritmen (ML) voor objectdetectie integreren. Een project om te overwegen is OpenMV [7], een combinatie van hardware en een software-ontwikkelomgeving speciaal gericht op machine vision. Met de STM32-familie microcontrollers biedt hun hardware toegang tot beeldsensoren tot 5 megapixels die naar behoefte uitwisselbaar zijn.

Softwareontwikkeling wordt ondersteund door de OpenMV geïntegreerde ontwikkelomgeving (IDE) die is afgestemd op de behoeften van machine vision (**figuur 7**). Aan de linkerkant is een editor voor het schrijven



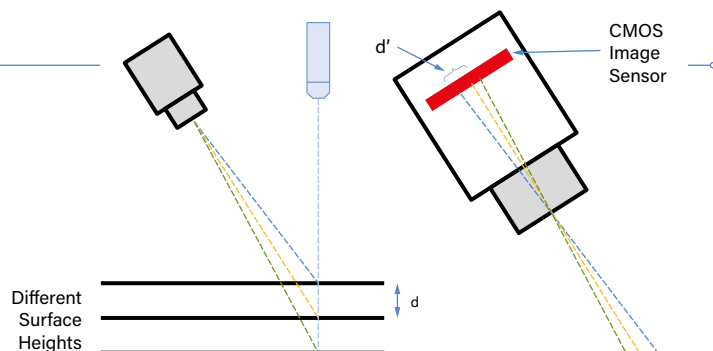
Figuur 7: De OpenMV-ontwikkelomgeving biedt een editor voor codering en toont het beeld dat door de aangesloten sensor is vastgelegd. (Bron: OpenMV)

ven van code in MicroPython en een seriële terminal, die tekstuitvoer van de applicatie-code weergeeft. Aan de rechterkant staat de uitvoer van de camera. Bovenaan staat de framebuffer die het beeld van de camera weergeeft, gecombineerd met uitvoer die door de toepassing in de videostream wordt geïnjecteerd. Dit kunnen tekstberichten zijn of grensvlakken die objecten markeren die met ML zijn gedetecteerd. Aan de onderkant worden histogrammen getoond, zoals de rood-, blauw- en groenwaarden van het beeld. Net als in de Arduino-omgeving zijn er voorbeeldprojecten beschikbaar. Deze bevatten ook ML-modellen voor toepassingen als gezichts- en persoonsdetectie.

3D-beeldvorming met behulp van laser-driehoeksmeting

Hoewel camera's met CMOS-beeldsensoren kunnen worden gebruikt voor een groot aantal vision-toepassingen, behoren die meestal tot het domein van de 2D-beeldvorming. Een groot aantal toepassingen vereist echter nauwkeurige oppervlaktemetingen die een 3D-meetcapaciteit vereisen. In de industrie is de laser-driehoeksmeting een gebruikelijke oplossing om het oppervlak van eindproducten te meten, zoals lasnaden, gelijmde delen en de positie van componenten op printplaten.

Bij deze sensoren wordt een laserpunt of -lijn op het meetobject geprojecteerd. Het door het oppervlak gereflecteerde licht valt op een geïntegreerde CMOS-sensor die met behulp van driehoeksmeting de afstand tot het object bepaalt (figuur 8). De positie van de beeldsensor en de laser bepaalt de complexiteit van de berekeningen die nodig zijn om de afstand, de resolutie en de kans op meetfouten te bepalen. De laser kan bijvoorbeeld loodrecht op het object staan (standaard-geometrie) met de camera onder een hoek, waardoor een goede sensor voor algemeen gebruik ontstaat.



Figuur 8: Laser-driehoeksmeting gebruikt een CMOS-beeldsensor om de reflectie van een laser op een oppervlak te detecteren om de hoogte ervan te bepalen. (Bron: MoviMED)

Maar bij het meten van sterk reflecterende oppervlakken zoals glas en gepolijst metaal is een look-away geometrie met zowel de beeldsensor als de laser onder een hoek ten opzichte van de meting een beter ontwerp.

Sensorfabrikanten als SmartRay bieden een breed scala aan 3D-sensoren op basis van deze technologie. Hun ECCO 95.015G-sensoren [8] zijn speciaal ontworpen om vlak glas en speculaire, sterk reflecterende oppervlakken te scannen. Gemonteerd op een afstand van 23,5 mm van het meetobject, bieden ze een verticale resolutie van 0,42...0,54 μm . Het apparaat levert zijn metingen als een reeks punten via een gigabit Ethernet-interface die kan worden gekoppeld aan typische industriële beeldvormingssoftware om de gegevens te evalueren.

CMOS-beeldsensoren voor iedereen

Met hun lage kosten en energieverbruik zijn CMOS-beeldsensoren alomtegenwoordig geworden in de huidige maatschappij. In detectietoepassingen waar ultrasone of infraroodsensoren zouden zijn ingezet,

is het heel goed denkbaar dat vandaag de dag een CMOS-beeldsensor wordt gebruikt. Maar met miljoenen pixels en een lastig te begrijpen functionaliteit kunnen ze moeilijk te koppelen zijn aan microcontrollers. Gelukkig maken de rekenkracht en geheugencapaciteit van de huidige microcontrollers relatief complexe machine-learning functies voor objectdetectie mogelijk, in combinatie met beeldsensoren van bescheiden afmetingen. En bij deze apparaten gaat het niet alleen om 2D-beelden. Industriële sensoren die lasergebaseerde driehoeksmeting gebruiken, bevatten CMOS-beeldvormers om tot op de micrometer nauwkeurige metingen in drie dimensies uit te voeren, zodat de visuele inspectie van werkstukken kan worden geautomatiseerd om de kwaliteit ervan te garanderen. ◀

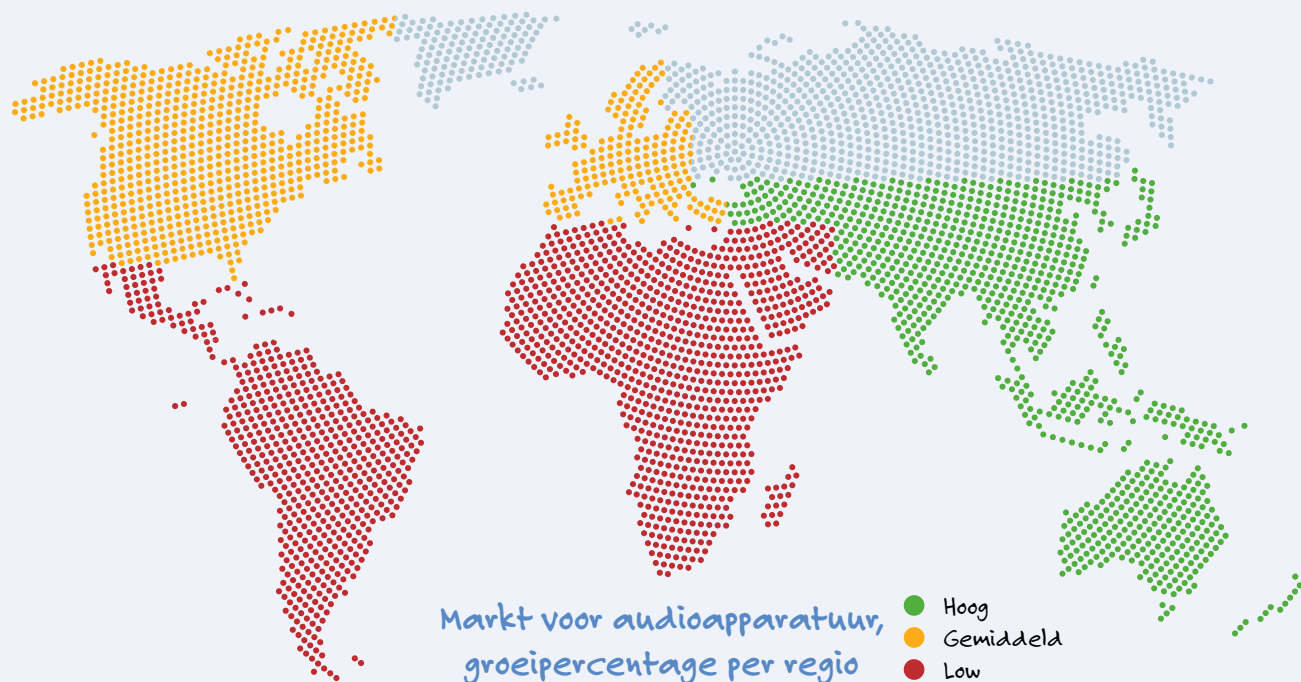
220618-03

Vragen of opmerkingen?

Heb je technische vragen of opmerkingen over dit artikel? E-mail de auteur op stuart.cording@elektor.com of neem contact op met Elektor via redactie@elektor.com.

WEBLINKS

- [1] "Hubble's Instruments: WFPC2 Wide Field Planetary Camera 2," European Space Agency: <https://bit.ly/3WBhN1Z>
- [2] "The Human Eye's Response to Light," Iowa State University: <https://bit.ly/3UsMVDU>
- [3] A. Rajwade, "Color Image Demosaicing," Indian Institute of Technology Bombay: <https://bit.ly/3UneoCg>
- [4] "What are all the discussions about global vs. rolling shutter?," PCO AG: <https://bit.ly/3h90h4R>
- [5] Product Page: AR0130CS CMOS Image Sensor: <https://bit.ly/3hgWq63>
- [6] M. Dadafshar, "Understanding Challenges in Powering High Resolution, High Frame Rate CMOS Image Sensors," OnSemi, July 2022: <https://bit.ly/3Dz7EtS>
- [7] Website: OpenMV: <https://openmv.io/>
- [8] Product Page: ECCO 95.015G: <https://bit.ly/3E7HakZ>



De Asia-Pacific regio loopt voorop

Kijkende naar audioapparatuur voor professionele toepassingen - mixers, versterkers, microfoons en luidsprekers/monitoren - verwachten experts voor de komende vier jaar een groeipercentage van ongeveer 6%. Dit is inclusief alle audioapparatuur binnen de

automobielsector. De markt zal groeien van ongeveer USD \$ 13,5 miljard in 2022 tot USD \$ 18 miljard in 2027. Asia-Pacific is hierbij de snelst groeiende regio, gevolgd door Europa en Noord-Amerika. Het verschil in groeicijfers tussen Asia-Pacific, Europa/Noord-Amerika en Afrika/

Zuid-Amerika kan voor een groot deel worden verklaard door de sterk groeiende automarkt in China en India. Eén auto kan wel iets van 10 luidsprekers hebben, soms zelfs wel meer.

(Bron: Fortune Business Insights, Mordor Intelligence)

Marktconcentratie in de markt voor audioapparatuur



Bron: Mordor Intelligence (Report Audio Equipment Market, 2022-2027)

Is er hier sprake van marktconcentratie? Nou nee

De toekomst van de markt voor professionele audioapparatuur ziet er goed uit. Daarom is er ook geen gebrek aan interesse van leveranciers om zich in deze markt te begeven. Dit draagt ertoe bij dat de vooruitzichten voor kopers van audioapparatuur nóg beter wordt. Het gebrek aan marktconcentratie betekent dat concurrentie op prijs in de markt voor audioapparatuur zeer stevig zal zijn. Daarbij zie je ook veel innovaties, of het nu gaat om versterkers, om microfoons, of om luidsprekers. Kortom: de professionele klant kan hoogwaardige geluidsapparatuur verwachten voor redelijke prijzen. Ook aan videoconferencing zal dit een boost geven.

(Bronnen: Microsoft, Mordor Intelligence)

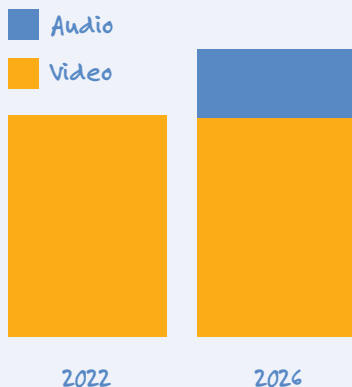
Bij de auto van de toekomst gaat het om... geluid uit/geluid aan

Bij het analyseren van de cijfers blijkt al snel dat de markt voor professionele audioapparatuur voornamelijk bestaat uit audioapparatuur in auto's. Zie de grafiek. Waarom is gebruik hiervan in de auto dan zo dominant? De eerste reden is dat we een groot deel van onze tijd in de auto doorbrengen, met schattingen van alleen al zo'n 45 uur per jaar extra door verkeersvertragingen. De tweede reden is dat het gebruik van infotainment-diensten in de auto gevaarlijk kan zijn: ze kunnen de bestuurder afleiden, waardoor er niet goed op de weg gelet wordt. Daarom is voice recognition ook zo belangrijk. En spraakherkenning vereist audioprocessors, en ook deze behoren tot die markt voor audioapparatuur.

(Bronnen: Fortune Business Insights, Mordor Intelligence)



Een overzicht: video wint het van audio



Er is overigens verschil tussen de professionele audiomarkt en de totale audiomarkt. Deze laatste beslaat namelijk ook de thuisgebruikers. Die totale audiomarkt betreft allerlei soorten recorders en players, dus niet alleen professionele mixers, etc. Hetzelfde geldt voor videoapparatuur: een televisiecamera is heel wat anders dan een smartphonecamera. Indien we professioneel gebruik en thuisgebruik voor zowel audio als video zouden

optellen, wat zouden dan de cijfers zijn? Welnu, deze markt bedroeg vorig jaar ongeveer USD \$ 158 miljard, met een verwachte groei tot USD \$ 196 miljard in 2026. Ongeveer 80% van die totale markt bestaat uit videoapparatuur. Curved tv's zullen daarin de komende jaren een groot aandeel hebben.

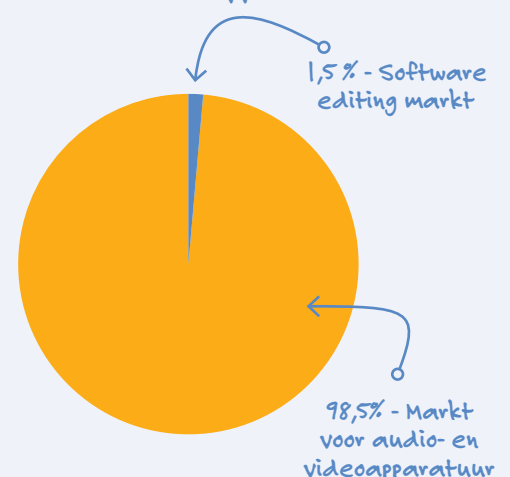
(Bronnen: Marktvooruitzichten, Onderzoek en Markten)

Hardware? Ok, duidelijk, maar hoe zit het dan met software?

Tot nu toe hebben we alleen naar hardware gekeken. Maar hoe zit het dan met software? Uiteraard is er een markt voor audio- en video-editingsoftware, alleen al vanwege de vele YouTubers. De softwaremarkt groeit in een nog hoger tempo dan de markt voor audio- en videoapparatuur: het gaat dan om 7%, in plaats van de ongeveer 6% voor de komende jaren. In vergelijking met audio- en videohardware is de markt voor audio- en videosoftware echter maar bescheiden. Volgens cijfers van Transparency Market Research zal de markt voor audio- en video-editingsoftware groeien van ongeveer USD \$ 2 miljard in 2020 tot ongeveer USD \$ 3 miljard in 2030. Dit betekent dat de markt voor audio/video software slechts ongeveer 1,5% van de hardwaremarkt beslaat.

(Bron: Research and Markets, Transparency Market Research)

Software editing markt vs. audio-/videoapparatuur





De evolutie van spraak- en audiogestuurde elektronische apparaten

▲

*Figuur 1: Spraakgestuurde robotassistent.
(Bron: PaO_STUDIO op Shutterstock)*

Mark Patrick (Mouser Electronics)

Spraak is een efficiënte manier voor mensen om ideeën en wensen uit te drukken. Reeds vóór het industriële tijdperk ontdekte de mens dat dieren konden worden getraind op het herkennen van elementaire commando's, bijvoorbeeld om bepaalde taken uit te voeren. De volgende logische stap was het vinden van een manier om met onze machines te communiceren en ze aan te sturen met behulp van onze stem. Het gebruik van spraak en audio als bedieningsinterface voor elektronische apparaten is de laatste jaren steeds populairder geworden en de ontwikkelingen gaan door om te voldoen aan de verwachtingen van gebruikers en de eisen van nieuwe toepassingen. In dit artikel gaan we dieper in op de voordelen van spraak en audio voor het aansturen van elektronische apparatuur en machines, en we bespreken enkele toepassingen. We laten ook zien hoe deze bedieningsinterface nu kan worden ingebed in offline apparaten en hoe de audio-ervaring die ze bieden sterk verbeterd kan worden.

Het gebruik van spraak voor het aansturen van elektronische apparatuur

De interactie tussen spraak en machine biedt een aantal duidelijke voordelen:

- Spraak is voor mensen een intuïtieve vorm van communicatie en dus een makkelijke manier om verbaal opdrachten over te brengen.
- Communicatie door middel van spraak is ook mogelijk wanneer iemands handen en ogen met iets anders bezig zijn. Hoewel aanvullende aansturing met spraak handig kan zijn, is het in bepaalde omstandigheden, zoals tijdens het autorijden, verboden om te proberen een ander apparaat met aanraking aan te sturen.
- Spraak is een efficiënt medium voor het aansturen van slimme machines die kunnen luisteren en reageren zonder de noodzaak van complexe commando's.
- Spraakintegratie maakt een touchscreen op veel apparaten vrijwel overbodig. Dit is vooral wenselijk voor op afstand geplaatste of draagbare, door batterijen gevoede apparatuur, waarbij kleinere afmetingen en minder stroomverbruik een gemeenschappelijke uitdaging vormen voor ontwerpers. Het wegnemen van de noodzaak om een apparaat fysiek aan te raken is ook hygiënischer, zeker voor toepassingen met meerdere gebruikers.
- zoals te zien is in **figuur 1**, kan het een hulpmiddel zijn voor mensen met een lichamelijke beperking voor wie fysiek contact geen haalbare optie is. Spraakcommunicatie met machines kan worden gebruikt voor het uitvoeren van verschillende taken, zoals het openen van deuren, of om updates over iemands gezondheid door te geven op afstand.

Het audio-ingangsdeel (audio front-end, AFE) van een spraakgestuurd apparaat omvat een microfoonarray en signaalverwerkingsblokken. Het AFE verwerkt het signaal van een meerkanaals microfoonarray om interferentie van eventuele achtergrondruis of van geluiden die het apparaat zelf maakt te neutraliseren. Dit signaal wordt vervolgens naar een detectiefunctie voor een activeringswoord ('wake-word') gestuurd die op het apparaat voorgeprogrammeerde woorden herkent, zoals „Alexa” of „OK Google”. Er worden meerdere signaalverwerkingsalgoritmen gebruikt om ongewenste stoorsignalen te elimineren. Een oplossing voor spraakaansturing bevat onder meer de volgende componenten:

Microfoonarray: Systemen voor spraakactivering vereisen één of meerdere microfoons om het audiosignaal op te vangen. Grootte, kosten, prestaties en robuustheid zijn de belangrijkste overwegingen bij de

keuze voor een microfoonarray. Door verschillende signalen van een multi-microfoonarray te combineren, kan de verhouding tussen signaal en ruis (signal to noise ratio, SNR) voor de audiosignaalketen worden verbeterd.

Detector voor de richting van aankomst (direction-of-arrival, DoA): Bepaalt de positie van de gebruiker ten opzichte van het apparaat, waarna de microfoonarray zich kan richten op de stem.

Bundelvormer (beamformer): Registreert geluiden van de DoA en onderdrukt geluidssignalen uit overige richtingen. De prestaties zijn afhankelijk van de geometrie en van de SNR van de microfoonarray, evenals de bundelbreedte en de hoeveelheid achtergrondruis.

Akoestische echowisser (acoustic echo canceller, AEC): Onderdrukt het playbacksignaal op de speaker van het apparaat (bijvoorbeeld als de speaker van het apparaat muziek afspeelt), zodat het spraakcommando van de gebruiker duidelijk en zonder stoorsignalen binnenkomt.

Adaptieve interferentiewisser (adaptive interference canceller, AIC): Neutraliseert externe ruis van andere geluidsbronnen die moeilijk te onderdrukken is met een traditionele bundelvormer, bijvoorbeeld lawaai dat door andere apparatuur wordt voortgebracht.

Activeringswoorddetector: Vergelijkt het verwerkte spraaksignaal van de AFE met een bibliotheek van activeringswoorden, zoals „Hey Google”. Hierbij wordt gebruikgemaakt van een detectiealgoritme voor activeringswoorden dat onderdeel is van een machine-learning model. Hoe groter het model, des te nauwkeuriger het resultaat. Een getraind model van 1 MB is bijvoorbeeld nauwkeuriger dan een model van 64 kB, maar vergt ook meer processorkracht. Om de activeringswoorden nauwkeurig te detecteren en het aantal valse alarmen te verminderen zijn grote modellen voor activeringswoorden nodig.

Klasse D audioversterkers

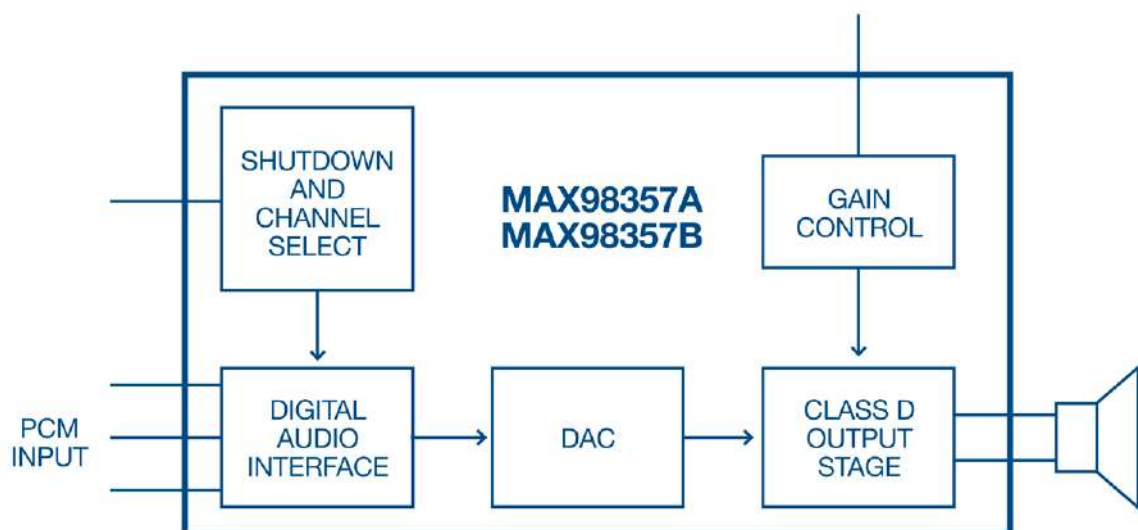
Het spraakverwerkingsgedeelte van deze bedienings-interface heeft een grote ontwikkeling doorgemaakt, zodat nu zelfs goedkope apparaten nauwkeurige spraakherkenning kunnen bieden. Het audiogedeelte van de interface heeft echter aanzienlijk minder aandacht gekregen, waardoor de geluidskwaliteit van veel oudere slimme speakers en andere audioapparatuur voor het Internet of Things (IoT) inferieur was aan de geluidskwaliteit van duurdere audioapparatuur.

Het nieuwtje van de spraakbesturing zelf werd misschien beschouwd als voldoende afleiding om de aandacht af te leiden van deze tekortkoming.

Maar nu slimme apparaten steeds breder worden toegepast, worden de verwachtingen van de consument m.b.t. de audio-ervaring ook steeds hoger. De lage efficiëntie van traditionele klasse AB audioversterkers maakt ze onpraktisch voor gebruik in IoT-apparaten met een laag vermogen. Daarom hebben verschillende chipfabrikanten onlangs een reeks geavanceerde klasse D audioversterkers geïntroduceerd die een aanzienlijke verbetering zijn ten opzichte van de eerder beschikbare versterkers. Veel van de nieuwe versterkers zijn speciaal ontwikkeld voor hoogwaardige audio in slimme technologie en IoT-apparatuur.

De TAS2770 [1] 15-W ingangsversterker van Texas Instruments [2] verhoogt het volume en de audiokwaliteit. Zijn grotere vermogen om spraak op te vangen betekent een eenvoudigere en natuurlijkere bediening van spraakgestuurde apparaten. De TAS2770 is het eerste audio-ingangsdeel dat een digitale microfooningang combineert met een krachtige I/V-versterker, waardoor spraak en omgevingsgeluiden kunnen worden geregistreerd voor echo- of ruisonderdrukking in spraakgestuurde toepassingen. Maxim Integrated [3] (nu onderdeel van Analog Devices [4]) biedt met zijn MAX98357 [5] en MAX98358 [6] klasse D versterkers een rendement van 92% en een klasse AB audioprestatie van 3,2 W. **Figuur 2** toont een vereenvoudigd blokschema voor deze versterkers. De PAM8106 [7] van Diodes Incorporated [8] heeft een laag stroomverbruik, waardoor deze uitermate geschikt is voor apparaten met 1,5 V loodbatterijen en 3,5 V lithium-ion batterijen. Deze versterker heeft een efficiëntie van 92% en kan zonder omvangrijk koellichaam worden geïmplementeerd in een compacte ontwerpstructuur.

Figuur 2: Vereenvoudigd blokschema klasse D audioversterkers van Maxim Integrated. (Bron: Maxim Integrated)



Figuur 3: SLN-LOCAL2-IOT offline spraakaansturingsoptelling van NXP. (Bron: NXP [10])

Offline spraakaansturing

Er zijn cloudgebaseerde oplossingen, zoals Alexa van Amazon en Google Assistant, verkrijgbaar voor apparaten met een stabiele internetverbinding, maar voor apparaten met een slechte of geen verbinding is offline spraakaansturing een betere oplossing. Als een product bijvoorbeeld moet reageren op eenvoudige commando's van één woord, zoals go, stop, reset, enz. (vaak aangeduid als keyword spotting), dan is het zinvol om de verwerking lokaal op het apparaat zelf uit te voeren. Een goedkope ingebedde microcontroller, zoals de EdgeReady MCU-gebaseerde oplossing van NXP voor offline lokale spraakaansturing, maakt de implementatie van een eenvoudig systeem voor commando's met keywords mogelijk. Deze maakt gebruik van de i.MX RT crossover MCU en stelt ontwikkelaars in staat om snel spraakbesturing toe te voegen aan hun producten. De op i.MX RT106S gebaseerde oplossing van NXP omvat de SLN-LOCAL2-IOT development kit [9]. (Zie **Figuur 3**).

Deze oplossing wordt geleverd met volledig geïntegreerde software, draait onder FreeRTOS en is voorzien van een software-ontwikkelingskit (software development kit, SDK) voor een snelle proof of concept. Met deze oplossing kunnen fabrikanten hun producten in het kader van smart home, smart appliance, smart building en smart industrial, eenvoudig voorzien van spraakbesturing, zonder de noodzaak van WiFi of cloud-connectiviteit. Offline spraakaansturing pakt tevens de privacyzorgen van veel consumenten aan die vrezen dat hun systeem kwetsbaar is voor online hackers. De kit heeft ook een op Windows™ gebaseerde tool waarmee spraakmodellen kunnen worden gemaakt voor meer dan honderd commando's op maat en voor meerdere activeringswoorden uit tekstinvoer in meer dan veertig talen.

Conclusie

Spraak en audio worden snel de favoriete interfaces bij uitstek voor veel slimme apparaten. Omdat ze geen dure en energieverlindende digitale displays vereisen, zijn ze namelijk bijzonder geschikt voor gebruik in draagbare IoT-apparaten met een laag stroomverbruik. Veel oudere systemen hadden een inferieure audiokwaliteit en konden alleen worden geïmplementeerd met behulp van een met de cloud verbonden oplossing.

Dankzij een nieuwe generatie zeer efficiënte klasse D audioversterkers kunnen fabrikanten de consument apparatuur aanbieden met een hoogwaardige audiokwaliteit. Er zijn nu ook oplossingen verkrijgbaar om apparaten met een slechte of geen internetverbinding aan te sturen. Deze innovaties tonen aan dat

deze technologie eenvoudig aan de nieuwe eisen die telkens ontstaan is aan te passen. Naarmate mensen meer gewend raken aan deze bedieningsinterface, zal deze trend zich waarschijnlijk voortzetten. ◀

220620-03



Over de auteur

Als Technical Marketing Manager voor EMEA bij Mouser Electronics is Mark Patrick verantwoordelijk voor het maken en verspreiden van technische content in die regio – content

die essentieel is voor de strategie van Mouser om zijn technische publiek te ondersteunen, te informeren en te inspireren.

Voordat hij leiding gaf aan het Technical Marketing-team, maakte Patrick deel uit van het EMEA Supplier Marketing-team en speelde hij een cruciale rol bij het aangaan en ontwikkelen van relaties met belangrijke productiepartners. Naast een verscheidenheid aan technische en marketingfuncties, heeft Patrick onder andere acht jaar bij Texas Instruments in Applications Support en Technical Sales gewerkt. Als 'hands-on' engineer in hart en nieren, met een passie voor vintage synthesizers en motorfietsen, draait hij zijn hand er niet voor om die te repareren. Patrick heeft een eerstegraads Honours Degree in Electronics Engineering van Coventry University.

WEB LINKS

- [1] TAS2770 Audio Amplifier Evaluation Module : <http://elektor.link/MouserTAS2770>
- [2] Texas Instruments @ Mouser: <http://elektor.link/MouserTexasInstruments>
- [3] Maxim Integrated @ Mouser: <http://elektor.link/MouserMaxim>
- [4] Analog Devices @ Mouser: <http://elektor.link/MouserAnalogDevices>
- [5] MAX98357: <http://elektor.link/MouserMAX98357>
- [6] MAX98358: <http://elektor.link/MouserMAX98358>
- [7] PAM8106 10W Audio Amplifier : <http://elektor.link/MouserPAM8106>
- [8] Diodes Incorporated @ Mouser: <http://elektor.link/MouserDiodesIncorporated>
- [9] SLN-LOCAL2-IOT Solution for Local Voice Control: <http://elektor.link/MouserSLNLOCAL2IOTDevkit>
- [10] SLN-LOCAL2-IOT Product: <http://elektor.link/MouserSLNLOCAL2IOT>

WEEF 2022

terugblik

C. J. Abate (Elektor)

De ethische beweging in de elektronica kwam scherp in beeld tijdens het tweede World Ethical Electronics Forum van de Lenthe Foundation dat plaatsvond tijdens electronica 2022 in München, Duitsland.



Het 2e World Ethical Electronics Forum vond plaats tijdens electronica 2022 in München.
(Bron: Messe München)

Op 15 november 2022, op de elektronica-beurs in München, bracht het World Ethical Electronics Forum (WEEF) ingenieurs, leidinggevendenden uit de elektronica-industrie, academici en techjournalisten samen om te discussiëren over veel van de belangrijkste ethische elektronica vragen van deze tijd. Tijdens het eendaagse evenement boden de Lenthe Foundation – een in Amsterdam gevestigde non-profit organisator van evenementen over sociaal ondernemerschap – en haar mediapartners – Elektor, Elektronikpraxis en electronica – een platform voor vooruitziende leiders om de volgende onderwerpen te bespreken: milieuvriendelijke ontwerpmethoden, het belang van verantwoorde ontwikkelingsdoelstellingen, de planeet en mensen vóór winst, ethische HR-praktijken, duurzame resources, en nog veel meer.

Het evenement bood stimulerende gesprekken en inzichten van een breed scala aan thought leaders: prof. dr. Stefan Heine-mann (professor, FOM University of Applied Sciences), dr. Reinhard Pfeiffer (CEO, Messe München), Florian Kurz (KK Ventures), Michel Lorient (AGIGA), Tessa Quandt (VARTA), Dr. Ralf Bodelier (auteur, journalist), Hendrik-Jan Overmeer (medeoprichter van het team Deedmob), Andrea Barrett (VP Social Responsibility and Sustainability, RS Group), Robert van Kats (medeoprichter bkvv architecten), Milda Pladaitė (initiatiefneemster van Engineers Enterprise),



De World Ethical Electronics Forum Award 2022.
(Bron: Messe München)



WEEF Award-winnares Milda Pladaitė –
uitgereikt door Beatriz Sousa (Elektor). (Bron:
Messe München)



Prijswinnaar Frank Stührenberg (CEO van
Phoenix Contact) met dr. Reinhard Pfeiffer
(CEO van Messe München) en Don Akkermans
(Stichting Lenthe / voorzitter van het
programmacomité). (Bron: Messe München)

Kelly Heaton (kunstenaar, VS), Johann Wiesböck (hoofdredacteur, Elektronik-praxis), C. J. Abate (Content Director, Elektor), Udo Bormann (project director van de electronica fast forward 2022 Startup Award), Stuart Cording (journalist, Elektor) en Don Akkermans (voorzitter, WEEF). Naast de voordrachten huldigde de WEEF-jury vier WEEF-prijswinnaars:

- **Sven Krumpel:** de eigenaar van CODICO, die zijn werknemers de beschikking geeft over grond voor de teelt van groenten en fruit.
- **Gopal Kumar Mohoto:** een ingenieur die werkt aan elektrisch vervoer in Bangladesh, hij is de medeoprichter en CTO van Advanced Dynamics.
- **Frank Stührenberg:** CEO van Phoenix Contact, bestuurslid van de Stichting

KlimaWirtschaft en een belangrijke medewerker van de All Electric Society.

- **Milda Pladaitė:** een civiel ingenieur uit Litouwen die deel uitmaakt van de World Federation of Engineering Organizations (WFEO) Young Engineers Future Leaders Committee.

Het volledige event werd opgenomen door Elektor – kijk maar op het Elektor TV-kanal op elektor.tv!



Nieuwsgierig naar de toekomst van WEEF? Bezoek **worldethicalelectronicsforum.com** voor actueel nieuws over WEEF, de 2023 WEEF index en ethische elektronica-gerelateerde nieuwsberichten en artikelen. Begin tot medio 2023 zal de Lenthe Foundation (de producent van WEEF) de jaarlijkse WEEF GUIDE lanceren. Deze zal worden verspreid (online en in gedrukte vorm) onder duizenden WEEF-partners en geïnteresseerden. De WEEF GUIDE zal de profielen bevatten van thought leaders uit de 100 WEEF INDEX, interviews met winnaars, controversiële commentaren van buiten de sector, interviews met ethische leiders, en nog veel meer. Als je een bijdrage wilt leveren, neem dan contact op met Shenja Panik via shenja.panik@elektor.com

220642-03



Moderator Johann Wiesböck en Don Akkermans interviewen hoogleraar bedrijfsethiek dr. Heinemann (FOM).



Stuart Cording was een van de twee moderatoren die een dag vol ethische debatten leidde. (Bron: Messe München)



Don Akkermans en C. J. Abate bespreken onderzoekresultaten over elektronica-exposanten en hun ethische representatie.

FFWD

electronica 2022 terugblik

de vernieuwers hebben niet nagelaten indruk te maken



Brian Tristam Williams

Na maanden van voorbereiding en opwinding heeft de uitreiking van de Fast Forward 2022 start-up en scale-up awards, mogelijk gemaakt door Elektor, plaatsgevonden tijdens electronica 2022 in München.

In november, op de electronica Fast Forward 2022 – powered by Elektor, kwamen we om eenhoorns te zien, en we werden niet teleurgesteld. We kregen eindelijk alle start-ups en scale-ups te zien waar we de afgelopen maanden naar hebben uitgekeken. Op de stand van Elektor ontvingen we al deze vrienden in een speciaal voor dit doel opgezette Fast Forward 2022 tentoonstellings- en presentatieruimte.

De deelnemers

In hun eigen stand stelden de start-ups hun innovaties voor:

- > **Opulo**, met hun innovatieve DIY open-source pick-and-place machine.
- > **AMSEL** gaf ons een diepgaande blik op hun baanbrekende oplossing voor oppervlaktevoorbereiding.
- > **AirHood** toonde ons hun afzuigkap

voor fornuizen/kookplaten die geen afzuigkap is, maar een draagbare, esthetisch verantwoorde oplossing op het aanrecht voor het elimineren van keukendampen.

- > **V-Juice toonde** hun elegante en slanke ideeën voor draadloos opladen, met meer toepassingsperspectief dan alleen de techniek - marketing, gamification en incentivization behoren allemaal tot de mogelijkheden.
- > **Treesense** stelde milieusensoren voor, waardoor bomen met ons kunnen communiceren en ons kunnen vertellen wat ze willen, wat veel handmatige middelen bij het beheer en onderhoud van de flora bespaart.

De scale-up:

- > **wheel.me** was al eerder een favoriet en toonde zijn autonome wieloplossingen, die zeker een einde zullen maken aan het rugbrekende industriële en commerciële heffen en transporteren.

Innovation Lane:

- > **Voltera** demonstreerde hun bestaande producten, plus een spannend nieuw product dat de mogelijkheid biedt om (bijna) alles op alles af te drukken.
- > **Quadruped** was een favoriet bij de bezoekers van de beurs en trok mensenmassa's aan met hun viervoetige robots die de softwareoplossingen van het bedrijf aanprezen.

En een universiteitsproject:

- > **Ecurie Aix** presenteerde met trots hun RWTH Aachen University project dat elk jaar verder werkt aan hun volledig autonome elektrische Formula Student racewagens.



De Fast Forward 2022 tentoonstellings- en presentatieruimte – powered by Elektor.



Eersteprijs-winnaar Kai Kreos-Nemcock van wheel.me.



Een autonoom wiel van wheel.me doet het zware werk op een intelligente manier.



Alba Dervishi van Treesense is winnaar van de derde prijs.

Alle negen deelnemers mochten op dinsdag en woensdag hun ontwikkelingen uitgebreid tentoonstellen, en vervolgens, op donderdag, verbaasden de start-ups en scale-ups ons met hun officiële korte presentaties

Voor de juryleden was het schijnbaar niet makkelijk de winnaars te bepalen. Donderdag werd tot diep in de nacht beraadslaagd waarna op vrijdag de winnaars werden bekendgemaakt.

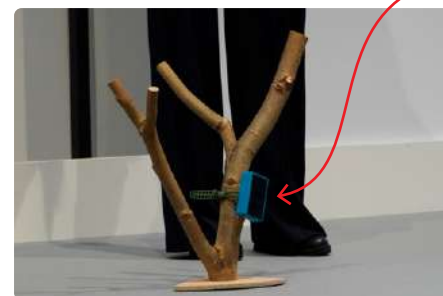
1e prijs

Met een mediabudget van 75.000 euro van Elektor kwam wheel.me weg, die grote vooruitgang hebben geboekt in het veren-

voudigen van zwaar werk, met toepassingen in logistiek en magazijnbeheer tot in ziekenhuizen [1]. Hun wielen zijn niet alleen krachtig en nemen een grote last weg waar het nodig is, maar ze kunnen ook met elkaar communiceren, zijn voorzien van sensors of kunnen autonoom zijn. We kijken ernaar uit om ze verder te zien groeien en welke invloed ze hebben op de industrie!

2e prijs

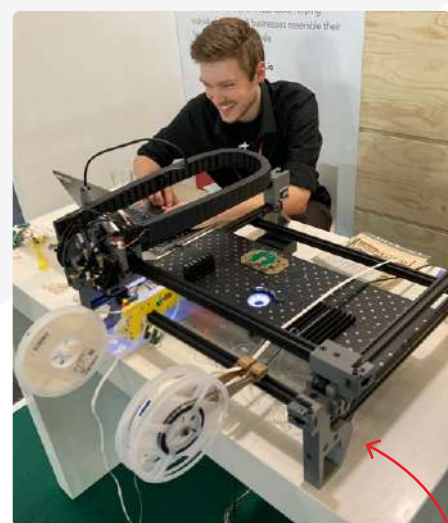
Stephen Hawes was de hele week aanstekelijk enthousiast, en zijn Opulo open-source pick-and-place-machine, die hij stap voor stap in elkaar zette, nauwgezet documenteerde en het hele proces 'vlogde' op YouTube, won de 2e prijs, wat betekent dat Opulo de kans krijgt om zijn aanbod aan te prijzen via het marketing-budget van € 50.000 uit de Elektor Media Kit [2].



Dankzij Treesense-sensoren kunnen bomen ons vertellen wat ze nodig hebben.



Winnaar van de aanmoedigingsprijs Dr. Salman Asad van AMSEL.



Stephen Hawes van Opulo stelt zijn open source pick-and-place machine op.

3e prijs

Treesense was de milieufavoriet van de stands, met elektronische innovatie in de vorm van embedded sensoren om de pols van bomen te voelen en de mensen die voor ze zorgen, te laten weten wat ze nodig hebben [3]. Met hun Elektor mediabudget van 25.000 euro kan Treesense wat licht werpen op een onderwerp dat niet die aandacht krijgt dat het verdient: de zorg voor de bomen op onze planeet.

Dat zou het moeten zijn. We hebben het eigenlijk nooit gehad over een 4e prijs. Maar, met een gepassioneerde late-night jury, konden we niet verder zonder een kanshebber te herkennen over wie we gewoon moesten toegeven bij het overwegen van het potentieel: "Daar zit zeker wat in."



Tweede prijs-winnaar Stephen Hawes van Opulo.

Aanmoedigingsprijs

AMSEL won een mediabudget van 10.000 euro om hun ongelooflijke eBlaze-technologie van de grond te krijgen. Ja, we waren genooddaakt om dit jaar een andere prijs uit te vinden omdat deze geavanceerde technologie voor oppervlaktevoorbehan-

deling, hoewel nog in de kinderschoenen, niet onopgemerkt kon blijven. Let op, want we hebben hun CEO, Dr. Salman Asad [4], opnieuw uitgenodigd voor Fast Forward 2024, waar hij mag bewijzen dat dit echt iets is.

Gelukwensen voor alle winnaars. We zullen jullie vorderingen de komende jaren met grote belangstelling volgen! ◀

220651-03

WEBLINKS

- [1] wheel.me: <https://wheel.me>
- [2] Opulo: <https://opulo.io>
- [3] Treesense: <https://treesense.net>
- [4] AMSEL's Salman Asad op LinkedIn: <https://linkedin.com/in/syed-salman-asad-86337216>

The Tube

een ongebruikelijke buizenversterker

Door Gerd Reime (Duitsland)

De belangrijkste kenmerken van deze bijzondere versterker zijn goede prestaties en een indrukwekkende verschijning. Dit project combineert vakmanschap in de constructie met audiofiele schakelingen en koppelt dankzij de buizentechnologie nostalgische elementen aan een moderne verschijning.

Ik bouwde mijn eerste buizenversterker in 1971, toen ik net 15 jaar oud was. Hij was opgebouwd rond twee EL84 buizen die in push-pull modus werkten, met behulp van een schakeling die ik in Elektor vond. De rest van de onderdelen waren afkomstig van oude radio's. Even later ging ik in de leer als radio- en televisietechnicus bij Grundig. Daarna werkte ik in R&D. Toen ik met pensioen ging, wilde ik aan een speciaal project werken. Wat ik in gedachten had was een mooie, kleine eindversterker van zo'n 10 tot 15 W uitgangsvermogen - met twee EL84's, net als vroeger, maar dan beter.

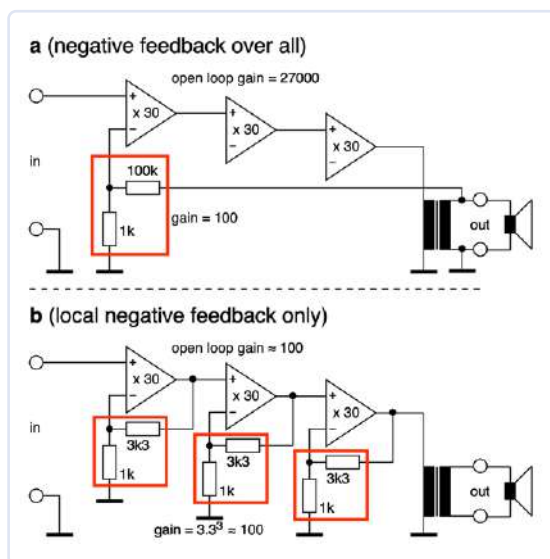
Eerst wilde ik alleen een speciale behuizing, maar toen raakte ik gefascineerd door de elektronische aspecten en probeerde met vele proefopstellingen en metingen iets zo goed mogelijk te construeren. Wat begon als een leuk project werd een serieuze onderneming. Als je inspiratie wilt putten uit mijn ideeën, vormt de volgende beschrijving een goede inleiding. De volledige bouwbeschrijving is nogal uitgebreid en eigenlijk te groot voor een Elektor-artikel. Voor alle details kun je het uitgebreide pdf-bestand raadplegen dat je kunt downloaden van [1].

Voorafgaande overwegingen

Waarom zweren zo veel audioliefhebbers bij buizenversterkers? De zogenaamd warme klank en de matige reactie op oversturing worden vaak genoemd. Bijna alle schakelingen die ik ken werken met negatieve terugkoppeling, maar wat zijn de effecten van dit soort terugkoppeling?

Een audioversterker verwerkt het zwakke signaal van een audiobron zodat het door een luidspreker kan worden weergegeven. Daartoe moeten zowel de spanning als de stroom worden versterkt, zodat de spoel en de conus van de (gewoonlijk gebruikte) dynamische luidspreker voldoende kunnen worden aangedreven om muziek of andere audiosignalen luid genoeg te maken om te worden gehoord. Dit is gemakkelijk met solid-state versterkers, maar buizenversterkers werken met relatief hoge spanningen en navenant geringe stromen, die niet rechtstreeks op de luidspreker kunnen worden aangesloten. Meestal is een uitgangstransformator nodig voor impedantieaanpassing. Vele generaties hifi-liefhebbers hebben zich gebogen over de vraag wat de beste transformator is.

Een ander aspect is dat het werkingsbereik van buizen niet zo lineair is als je zou willen. Negatieve terugkoppeling wordt gebruikt om de resulterende vervorming van het audiosignaal te verminderen. Een deel van het uitgangssignaal wordt met tegengestelde fase teruggevoerd naar de ingang, en de doorgaans hoge, "overtollige" open-lus versterking (vergeleken met de eigenlijke versterking van de versterker) vermindert de niet-linea-



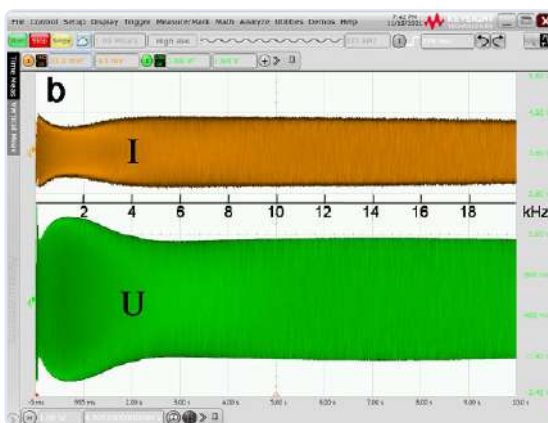
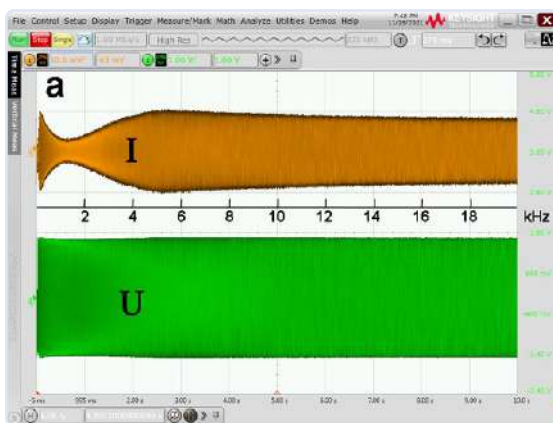
Figuur 1: Verschillende benaderingen van negatieve terugkoppeling: (a) conventionele algemene negatieve terugkoppeling en (b) lokale negatieve terugkoppeling, afzonderlijk in elke versterkertrap.

riteit die anders zou optreden. In **figuur 1a** wordt de negatieve terugkoppeling verzorgd door de spanningsdeler in het rode kader. Alle niet-lineariteit en vervorming in de versterker worden verminderd met een factor die gelijk is aan de verhouding tussen de open-lus versterking en de door de negatieve terugkoppeling ingestelde versterking. In het voorbeeld in figuur 1a is deze factor $27000 / 100 = 270$. Omdat de negatieve terugkoppeling vermoedelijk alle vervorming met deze factor vermindert, hebben tekortkomingen in het ontwerp minder effect.

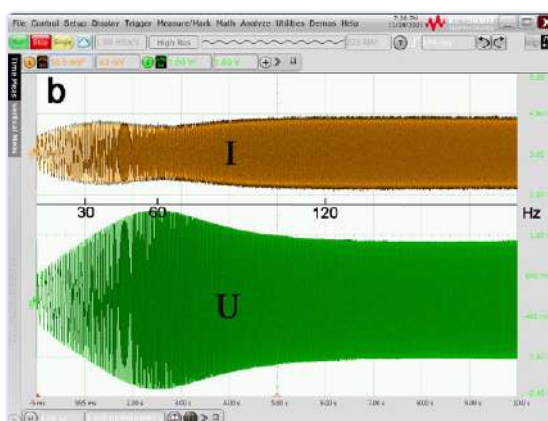
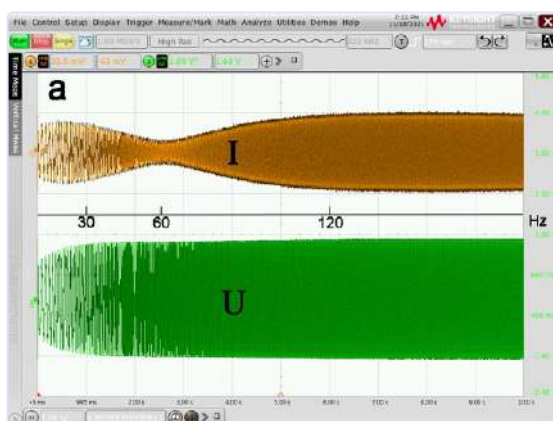
Een alternatief voor een sterke algemene negatieve terugkoppeling (van de luidsprekeruitgang naar de ingang van de eerste trap van de versterker) is het beperken van de versterking van elke trap van de versterker door lokale negatieve terugkoppeling (in **figuur 1b** wordt deze bepaald door de spanningsdelers in de drie rode kaders, die functioneel overeenkomen met de anode- en kathodeweerstanden van de buizen). Op deze manier worden de afzonderlijke versterkertrappen gelineariseerd, wat leidt tot een vrij lineaire versterker met weinig vervorming die geen totale negatieve terugkoppeling nodig heeft. De vraag is of dit een verschil maakt en of het werkelijk anders klinkt.

Vergeleken met een solid-state versterker is het opvallendste verschil dat er een uitgangstransformator voor de luidspreker zit. In de schakeling van **figuur 1a** bevindt de transformator zich in de terugkoppeling, maar in figuur 1b niet.

Het is te verwachten dat het ontwerp volgens figuur 1a een vrij vlakke frequentiecarakteristiek heeft wat betreft de spanning. Mijn metingen aan een dergelijke versterker, weergegeven in **figuur 2a**, bevestigen dit. De spanningskarakteristiek is zeer vlak, maar de stroomkarakteristiek heeft een scherpe dip rond 1 kHz omdat de impedantie van de aangesloten luidspreker op dit punt het grootst is. Daardoor is bij 1 kHz ook een vermogensdip te zien. Uit **figuur 2b** blijkt dat door de hogere belastingimpedantie bij 1 kHz de spanning stijgt en de stroomdip minder sterk is. Dit betekent dat de invloed van de luidsprekerimpedantie op het uitgangsvermogen gedeeltelijk wordt gecompenseerd, wat zou moeten leiden tot een ander (beter) geluid. Iets soortgelijks doet zich voor in het laagfrequente gebied bij de resonantiefrequentie van de woofer rond 60 Hz (**figuur 3**). Ook hier neemt de invloed van de luidsprekerimpedantie af als de uitgangstransformator niet in de negatieve terugkoppeling zit. Ik heb deze

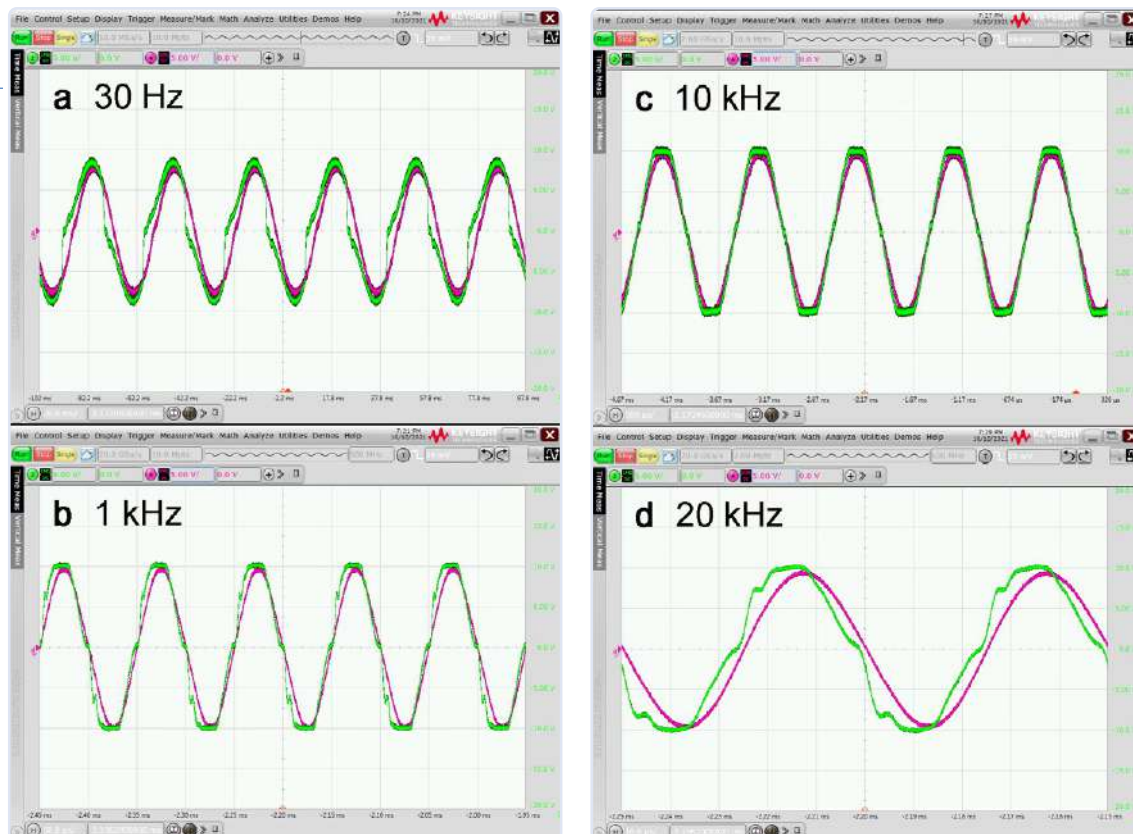


Figuur 2: Stroom (geelbruin) en spanning (groen) over het frequentiebereik (20 Hz tot 20 kHz): (a) met conventionele negatieve terugkoppeling en (b) met lokale negatieve terugkoppeling.



Figuur 3: Stroom (geelbruin) en spanning (groen) in het laagfrequente gebied: (a) met conventionele negatieve terugkoppeling en (b) met lokale negatieve terugkoppeling.

Figuur 4: Overdrive-gedrag bij verschillende frequenties (groen = conventionele negatieve terugkoppeling; rood = lokale negatieve terugkoppeling).



metingen herhaald met verschillende luidsprekers, en die laten over het algemeen een vergelijkbaar beeld zien.

De conclusie is dat een buizenversterker zonder negatieve terugkoppeling van de luidspreker naar een van de eerdere trappen (versie b) zich significant anders gedraagt dan een conventionele versterker (versie a). Natuurlijk heb ik het niet gelaten bij eenvoudige metingen van de twee versterkerontwerpen, maar heb ik ook collega's gevraagd deel te nemen aan luistertests. En de resultaten waren dat versie b, zonder totale negatieve terugkoppeling, in deze vergelijkingen altijd als beste uit de bus kwam. Het geluid werd als aanzienlijk warmer en evenwichtiger ervaren. Deze resultaten zijn niet verrassend, gezien de verschillende meetcurves.

Een ander verschil is dat een push-pull A/B eindtrap bij weinig vermogen of volume in klasse A werkt en daardoor van nature minder vervorming heeft. Daarom zou de harmonische vervorming met een goede uitgangstransformator bijna gelijk moeten zijn bij een laag volume, met of zonder negatieve terugkoppeling. Verrassend genoeg had het ontwerp zonder totale negatieve terugkoppeling een duidelijk geringere harmonische inhoud, vooral in klasse A. Overigens zijn naar mijn mening de beste uitgangstransformatoren die van Menno van der Veen [2].

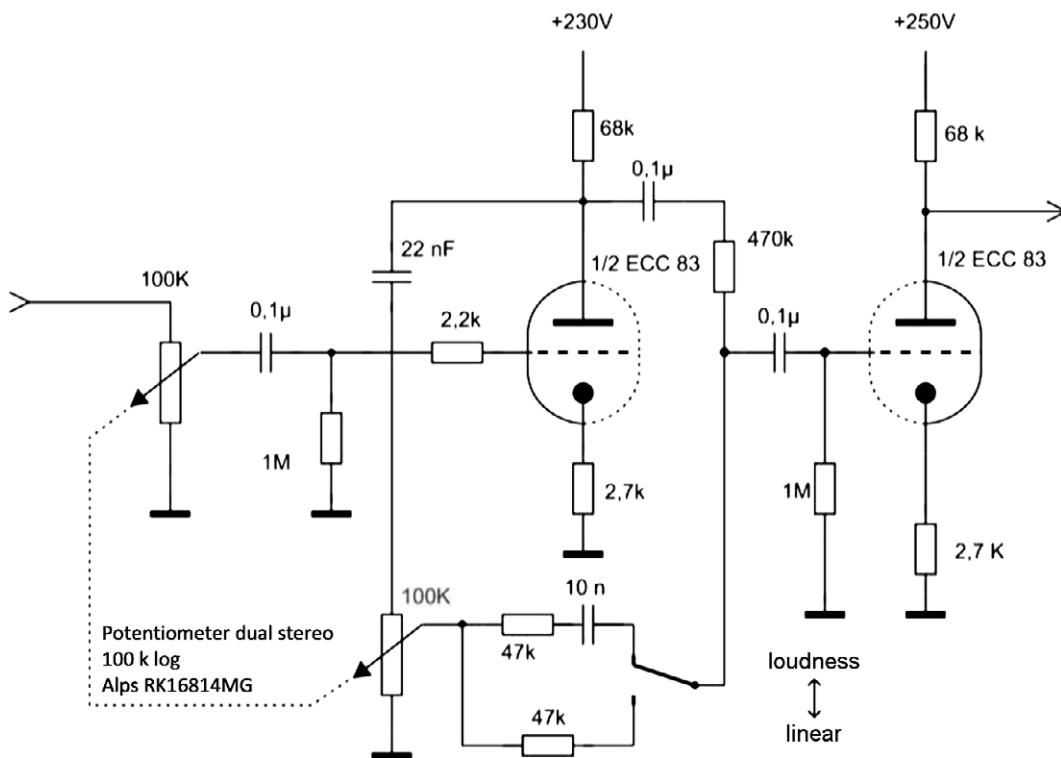
Oversturing

Bijna iedereen weet dat buizenversterkers een matigere begrenzing vertonen bij hoge geluidsterkte, wanneer de piekwaarden van het signaal de voedingsspanningen benaderen. Gewoonlijk begrenzen de eindbuizen de amplitude, samen met de uitgangstransformator, die

op dat punt in verzadiging raakt. Vooral in deze situatie wordt totale negatieve terugkoppeling problematisch, omdat bij clipping de terugkoppeling ervoor zorgt dat de niet-overgestuurde upstream-trappen de eindtrap nog forser aansturen met het correctiesignaal, wat direct leidt tot een sterke toename van de harmonische vervorming. Zonder deze vorm van negatieve terugkoppeling is het harmonische vervormingsgedrag bij overdrive aanzienlijk geringer.

Vooral bij lage frequenties in het bereik van 20 tot 30 Hz kunnen uitgangstransformatoren vaak geen groot vermogen omzetten. Als gevolg daarvan worden de negatieve effecten van negatieve terugkoppeling hoorbaar in het lage frequentiegebied, zelfs bij weinig vermogen. Daar komen nog de effecten bij van de fasekarakteristiek van de transformator, die eveneens ongewenste geluidseffecten veroorzaken als gevolg van negatieve terugkoppeling.

Mijn metingen met 10% overdrive bij verschillende frequenties ondersteunen deze overwegingen. De rode curven van de versterker zonder totale negatieve terugkoppeling in **figuur 4** zien er consequent beter uit dan de min of meer vervormde groene signalen van de versterker met totale negatieve terugkoppeling. De verschillen bij 20 kHz zijn bijzonder sterk, maar relevante verslechteringen bij een conventioneel versterkerontwerp zijn ook te zien bij 30 Hz. Interessant genoeg is de 10% overdrive bij deze frequentie goed voor 85% van het nominale signaalniveau. Samenvattend kan met recht worden gesteld dat echt matig clippen met nauwelijks storend effect alleen mogelijk is zonder totale negatieve terugkoppeling.



Figuur 5: Schakeling voor auditief gecompenseerde volumeregeling met een Alps quadpotentiometer.

Auditief gecompenseerde volumeregeling

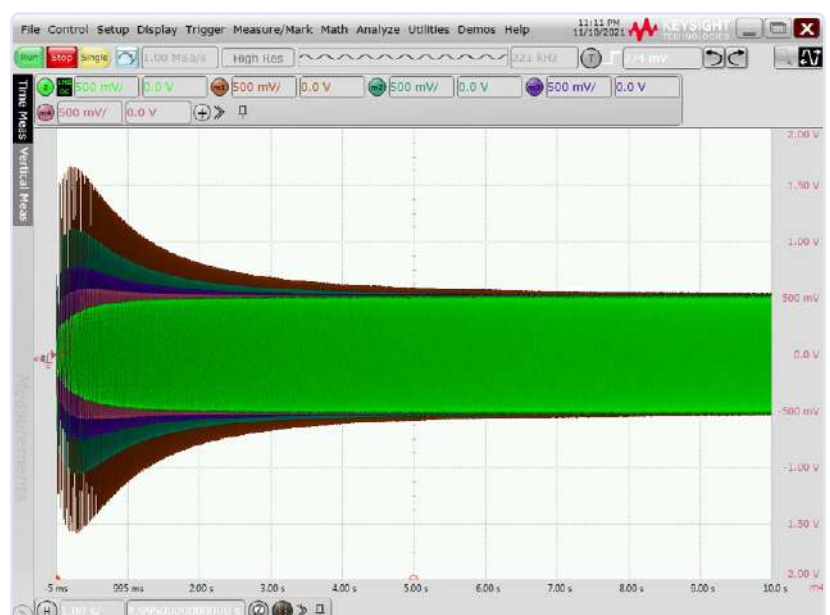
Je leest vaak dat een lineaire respons de enige juiste is. Volgens dit argument moeten topversterkers een vlakke frequentierespons hebben. Ik vermoed daarentegen dat de eis van een vlakke frequentierespons alleen gebaseerd is op het feit dat potentiometers met taps nu niet meer verkrijgbaar zijn. De equivalente schakelingen die online te vinden zijn werken slechts matig. Wat hierachter zit is dat het menselijk gehoor minder gevoelig is voor hoge en lage frequenties bij een laag volume, of (afhankelijk van hoe je het bekijkt) dat onze oren de voorkeur geven aan de frequenties in het middengebied.

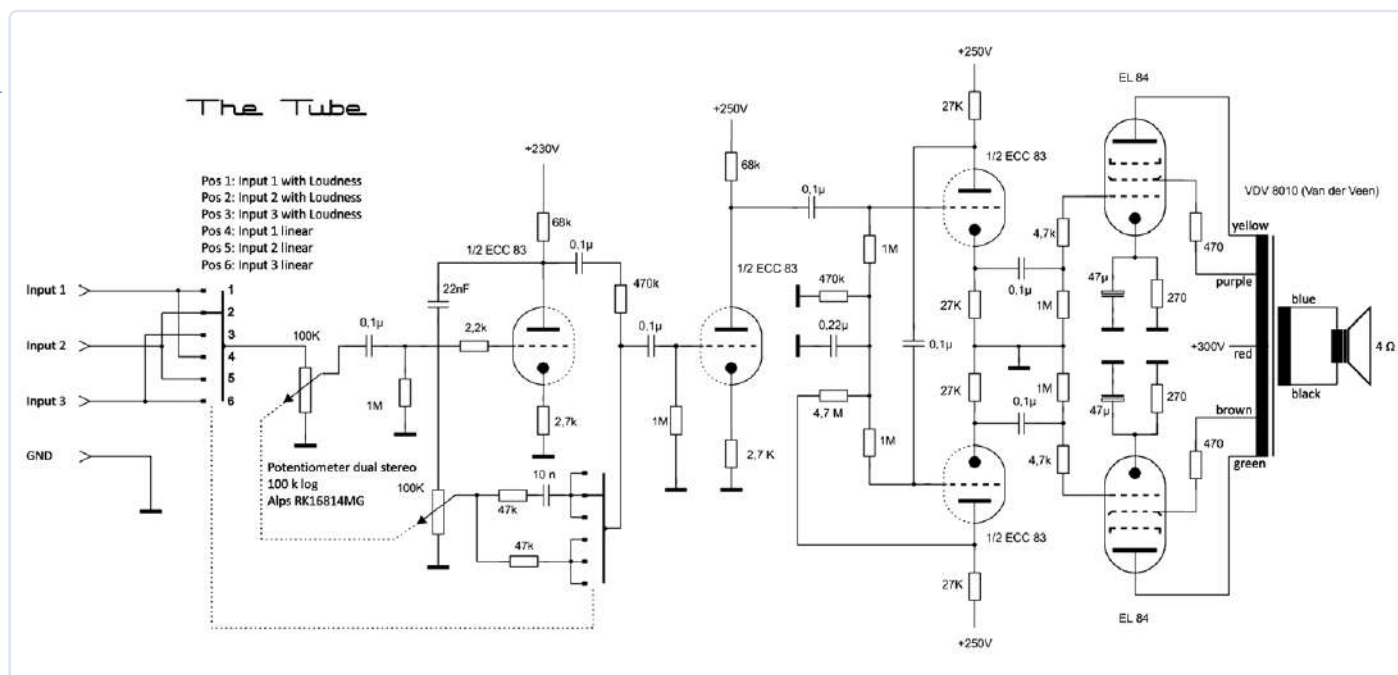
In het verleden was auditief gecompenseerde volumeregeling (ook wel fysiologische volumeregeling genoemd) de gebruikelijke norm. Ze benadrukten de hoge en lage frequenties bij een laag volume, waarbij deze nadruk afnam naarmate de geluidssterkte toenam. Maar hoeveel moderne versterkers bieden deze functie? Bijna geen.

In plaats van vintage potentiometers te gebruiken met twee of drie taps, die daarom sowieso niet ideaal waren, kan dit probleem op een andere en elegantere manier worden opgelost. Je gebruikt een dubbele stereopotentiometer, wat in feite een quadpotentiometer is, zoals die onder andere bij Alps nog verkrijgbaar is. Hiermee en met een eenvoudige schakeling kun je een uitstekende volumeregeling met auditieve compensatie bouwen. Bovendien kun je met een eenvoudige schakelaar wisselen tussen auditieve compensatie en lineair, zonder enige verandering aan de versterker.

Figuur 5 toont de schakeling die ik gebouwd heb met de Alps RK16814MG potentiometer. Hoewel dit een motor-aangedreven potentiometer is, kan de motor wanneer gewenst worden verwijderd zodat de potentiometer met de hand kan worden bediend. Om het effect van de auditief gecompenseerde volumeregeling te verduidelijken, heb ik de frequentierespons bij verschillende instellingen gemeten, deze genormaliseerd naar hetzelfde signaalniveau bij 1 kHz, en de curven over elkaar gelegd. Het effect in het lage frequentiegebied is te zien in **figuur 6**. Zoals duidelijk te zien is, neemt de frequentierespons-correctie geleidelijk af met toenemend volume.

Figuur 6: Effect van auditief gecompenseerde volumeregeling: **bruin** = zeer stil; **donkergroen** = stil (achtergrondmuziek); **blauw** = matig volume; **rood** = hoog volume (lineaire frequentierespons); **groen** = maximum volume vlak voor overdrive.





Figuur 7: Schema van één kanaal van The Tube. Bijzondere kenmerken zijn de auditief gecompenseerde volumeregeling en de symmetrische aansturing van de push-pull eindbuizen.

Aansturing van de eindbuizen

Zoals bekend hebben push-pull trappen signalen nodig met een faseverschil van 180°. Vergeleken met eenvoudige schakelingen met slechts één buis, kun je met wat meer complexiteit een betere kwaliteit bereiken. **Figuur 7** toont de schakeling van één kanaal van The Tube. Elk van de twee EL84's die dienst doen als eindbuis wordt aangestuurd door zijn eigen kathodevolger, bestaande uit de helft van een ECC83. Dankzij identieke anode- en kathodeweerstanden kan de bovenste triode het rooster van de onderste triode precies aansturen om een bijna perfecte, symmetrisch geconfigureerde faseverschuiving te produceren. Deze opstelling vermindert vervorming

en fasefouten. Op deze manier kan een goede versterker worden geproduceerd met een typisch buisengeluid, maar zonder harmonische vervorming, zelfs zonder totale negatieve terugkoppeling. Diverse vergelijkende luistertests bevestigen dat dit ontwerp (subjectief) zelfs beter is dan een dure en energieverblindende buizenversterker van topklasse A, en het laat zelfs goede solid-state versterkers achter zich.

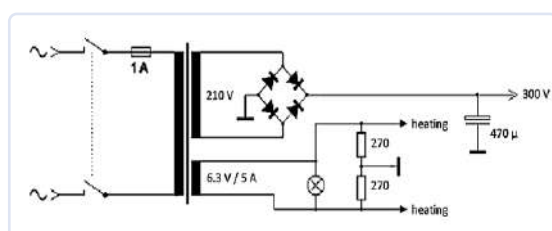
Het relatief eenvoudige basisontwerp van de voeding is weergegeven in **figuur 8**. Details zijn te vinden in [1].

Bouw

Het begon allemaal toen mijn zoons ergens een verzameling grote metalen buizen vandaan haalden. En ik dacht, met zulke buizen kun je eigenlijk maar één ding maken: een buizenversterker.

De massieve aluminium bovenplaat dient als basis voor de mooie buisvoeten. Zoals te zien is in **figuur 9** zijn de buizen van de tussentrap dicht bij de eindbuizen geplaatst. Vanwege de verwachte thermische belasting kan hier geen kunststof worden gebruikt.

Figuur 8: De basisschakeling van de voeding is vrij eenvoudig.



Figuur 9: Aantrekkelijke mechanische constructie voor de elektronische componenten: de voeten voor alle buizen van de stereoversterker.



Figuur 10: De afstandhouders staan ook mooi op de massieve aluminiumplaat.



Figuur 11: Geheel passend bij de buizentechniek: een "circuit board" van massief messing.

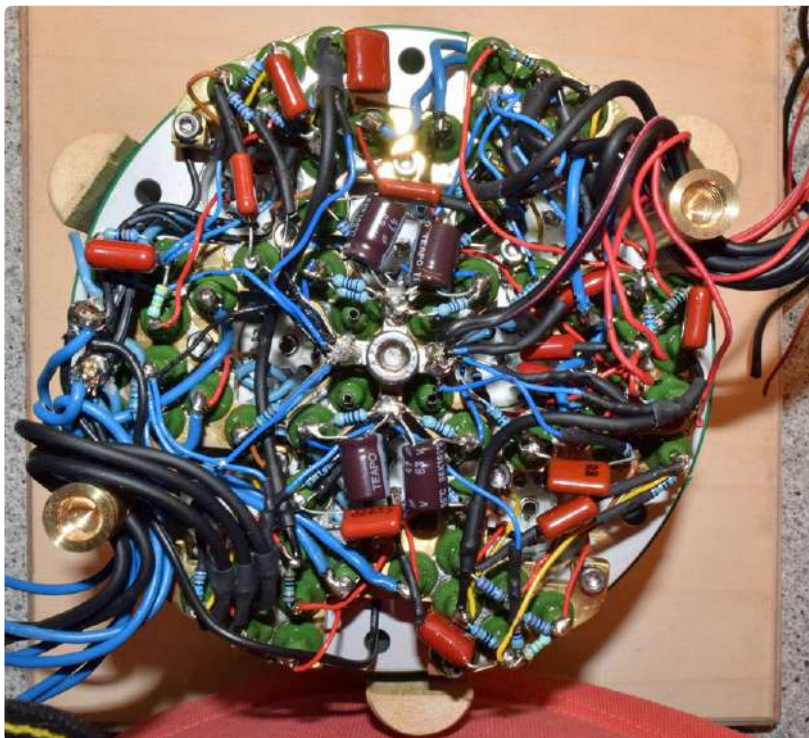


Figuur 12: Soldeerterminals met bijna vintage groene glazen isolatoren.

De onderkant van de aluminiumplaat (**figuur 10**) is voorzien van verschillende afstandhouders, waaraan het "circuit board" met schroeven is bevestigd. De term "circuit board" staat hier tussen aanhalingstekens om twee redenen: ten eerste omdat een board (print) niet gebruikelijk is in een buizenversterker, en ten tweede omdat het bestaat uit componentenhouders (**figuur 11**) gefreesd uit massief messing, zonder epoxy. Het inwendige van een dergelijke buizenversterker moet toch ook mooi zijn, nietwaar? **Figuur 12** toont de onderkant van



Figuur 14: De volumieuze elektrolytische condensatoren benodigen hun eigen messing plaat.



de aluminiumplaat met de afgewerkte messing plaat en de fraaie soldeerterminals met groene glazen isolatoren.

In volledig geassembleerde vorm ziet deze plaat eruit als **figuur 13**. Een ander "circuit board" bevat relatief grote componenten, zoals elektrolytische condensatoren (**figuur 14**). Ook voor de nettrafo, die zoals dat voor een ringkerntransformator hoort, rond is, zijn dragende messing platen gemaakt, die te bewonderen zijn in **figuur 15**.

▲

Figuur 13: Onderaanzicht (soldeerzijde) van het volledig gemonteerde "circuit board". Een beetje een rats nest, maar perfect stabiel.



Figuur 15: Een ronde transformator voor een ronde versterker. Ook hier zijn extra mechanische onderdelen nodig voor de bevestiging.

◀

► **Figuur 16:** De onderdelen van de gebruikersinterface (keuzeschakelaar, aan/uit-schakelaar en volumepotmeter) zijn gemonteerd op hun eigen aluminium ring.



Figuur 17: Voorlopige montage van de volledige versterker, zonder de beschermende buisbehuizing.



Figuur 18: De afgewerkte versterker in zijn volle glorie. Een waar juweeltje - meer dan alleen geschikt voor de woonkamer.

Dat is natuurlijk niet het hele verhaal over de mechanische kant. Het frontpaneel bestaat logischerwijs uit een aluminium ring waarop de ingangskeuzeschakelaar, de aan/uit-schakelaar en de potentiometer voor de volumeregeling zijn gemonteerd (**figuur 16**). Het provisorisch gemonteerde geheel van **figuur 17** geeft een goede indruk van deze opstelling. En nu – tromgeroffel! – de volttooide versterker in zijn behuizing (**figuur 18**).

Met de nodige aandacht voor detail is een buizenversterker als deze altijd een bijzonder esthetisch stukje techniek. Maar ontbreekt er niet iets belangrijks? Zoals bij elke versterker heeft The Tube connectoren nodig waar



Figuur 19: Een uitsparing aan de achterzijde bevat de ingangs- en uitgangconnectoren en de netspanningsconnector.

de zwakke audiosignalen binnenkomen en connectoren waar de versterkte signalen naar de luidsprekers gaan. En op de een of andere manier moet de netspanning de geïnstalleerde transformator bereiken. De oplossing voor dit probleem wordt getoond in **figuur 19**. Deze uitsparing in de buisvormige behuizing van de buizenversterker bevat de vergulde cinch-bussen voor de ingangen, de XLR-bussen voor de luidsprekers en een voedingsaan-sluiting voor het netsnoer.

Voel je je geïnspireerd?

Zoals uit mijn project blijkt, is voor een buizenversterker een goede schakeling slechts de helft van het verhaal. Om Thomas A. Edison te parafraseren, samen met de inspiratie (elektronica) is veel transpiratie (mechanica) nodig. Als je inspiratie wilt putten uit mijn project, moet je bereid zijn veel werk in de constructie te steken om een buizenversterker te bouwen die niet alleen goed klinkt, maar er ook geweldig uitziet. Het oude gezegde "no pain, no gain" is vooral waar als het gaat om buizen-technologie. ◀

220089-03 (vertaling: Willem den Hollander)

Over de auteur

Na zijn opleiding tot radio- en televisietechnicus werkte Gerd Reime achtereenvolgens in de ontwikkelingsafdelingen van Grundig en Nokia. Hij bezit verschillende octrooien.

Vragen of opmerkingen?

Heb je technische vragen of opmerkingen over dit artikel? Neem contact op met de Elektor-redactie via redactie@elektor.com.



Verwante producten

► **Elektor Audio Collection (USB Stick)** (SKU 19892)
www.elektor.nl/19892

► **R. A. Honeycutt, The State of Hollow State Audio (Elektor 2020)**
www.elektor.nl/19170

WEBLINKS

- [1] Uitgebreide projectbeschrijving (PDF): <https://www.elektormagazine.nl/220089-03>
- [2] Van der Veen-transformatoren <https://www.mennovanderveen.nl>

Biomateriaal in elektronica: klaar of niet



Bron: Shutterstock, Unsplash

Priscilla Haring-Kuipers (Nederland)

Veel van de schade ten gevolge van elektronica is een gevolg van de gebruikte materialen. Zijn we klaar om over te schakelen op milieuvriendelijkere biomaterialen bij de productie van elektronica?

Recycling

Duurzaamheid in elektronica verwijst meestal naar recycling, en daar moet inderdaad nog veel aan gedaan worden. Nog geen 20% van de elektronica in de wereld wordt gerecycled, en van dat aantal wordt het grootste deel afgehandeld door twijfelachtige recyclebedrijven of zelfs 'urban miners' en oud-ijzer-handelaren [1]. Als we kijken naar de keerzijde van de elektronica-productie, is er nog veel ruimte naar boven. Door een

ander basismateriaal te gebruiken, kunnen we ook het recycleproces veranderen. Biomaterialen kunnen worden ontworpen met kringloop in het achterhoofd. Schoon, niet giftig en makkelijker te verwerken.

Het probleem met FR4

Bijna alle elektronica wordt gebouwd op printen die bestaan uit FR4 en koperfolie. FR4 staat voor Flame Retardant klasse 4 (vlamvertragend dus) en wordt gemaakt door glasvezeldoek in een epoxyhars te dompelen. Epoxyhars is een thermohardend polymeer met ergens in de moleculaire structuur een 'epoxy ring' [2]. Epoxyhars is meestal afkomstig uit de petrochemie en heeft een grote CO₂-footprint (5,7...7,6 kg CO₂ per kg) [3]. De chemische verwerking van epoxy is ook een bekende oorzaak van beroepsastma, wat blijvende schade aan de longen kan veroorzaken. De hoge viscositeit van hars bij kamertemperatuur maakt daarnaast de toevoeging van petrochemische verdunningsmiddelen noodzakelijk.

Epoxyhars is een thermohardende kunststof die onmogelijk kan worden afgebroken tot zijn chemische bestanddelen en waarvan de bouwstenen niet kunnen worden gerecycleerd – zoals een eenmaal gebakken cake niet kan worden ‘ontbakken’ tot bloem, eieren en boter. De chemische reactie die dit materiaal zijn fantastische mechanische, diëlektrische, vlamvertragende en hitte- en vochtbestendige eigenschappen geeft, maakt het ook zeer moeilijk te recyclen. “Duurzame circulariteit zal meer onderzoek vereisen om hoogwaardige bio-gebaseerde polymeren te vinden ter vervanging van synthetische polymeren in PCB’s.” [4]

Het maken, verwerken en verwijderen van FR4 levert veel problemen op die zouden kunnen worden opgelost door andere materialen te gebruiken. Een van de ingrediënten van FR4 was vroeger koolteer, maar we hebben ons ontwikkeld tot andere derivaten van fossiele brandstoffen. Het lijkt erop dat we klaar zijn om door te ontwikkelen naar andere grondstoffen. Bij enkele projecten zijn papier, vlas en biogebaseerde harsmatrices gebruikt, in combinatie met minder giftige vlamvertragers of vervangende materialen op basis van soja en hennepolie. Duurzaamheid is een probleem bij deze ‘bioboards’ omdat ze – in tegenstelling tot FR4 – wel gevoelig zijn voor vocht.

Meer bio

PLA (polylactic acid, polymelkzuur) is de bekendste van de bioplastics die volledig uit duurzame materialen wordt vervaardigd. Helaas gedraagt het zich niet goed onder hittestress; de processen voor het maken van elektronica stellen echter vaak hoge eisen aan het materiaal (dampdepositie van metalen, reflow-ovens of golfsolderen). Er worden extra chemische bewerkingen van PLA ontwikkeld om het hittebestendiger te maken. Zoals een chemicus me onlangs vertelde: “We kunnen nu alle soorten bioplastic kweken. Je kunt gewoon je wensen aangeven.”

Op koolstof gebaseerd geleidend materiaal is beter bestand tegen hitte en heeft een beter elektrisch geleidingsvermogen. Er zijn een heleboel potentieel relevante geleidende koolstoffen die allemaal kunnen worden gesynthetiseerd uit hernieuwbare grondstoffen en die het gebruik van verschillende metalen in elektronica kunnen vervangen. Koolstof-nanobuisen kunnen worden gekweekt uit vrijwel elke organische verbinding en grafeen kan worden gemaakt door exfoliatie van grafiet. Verschillende universiteiten in de VS werken aan biogebaseerde grafeeninkt om elektronica te printen “die momenteel alleen in het laboratorium of in de verbeelding bestaat” en de biogebaseerde substraten om ze op te printen.



Bron: Michael Schiffer, Unsplash



Bron: Fukayamamo,
UnSplash

Verschillende Finse universiteiten werken aan flexibele piëzo-elektrische sensoren van houtvezels en bacteriën en aan supercondensatoren van paardenbloemen. Zij ontwikkelen “nieuwe groene materialen” zoals biosensoren, elektroden, printplaten en componenten op basis van cellulose en andere biogebaseerde materialen.

Ook de definitie van biogebaseerd is in ontwikkeling. Volgens de Amerikaanse Food and Drug Administration (FDA) omvat ‘bioplastic’ nu ook producten die gemaakt zijn van polymeren op basis van aardolie en gemengd zijn met natuurlijke vezels. Dergelijke materialen zouden niet biologisch afbreekbaar zijn en dat is misschien niet wat we in gedachten hebben als we aan bioplastic denken.

Ten langen leste

De problemen van petrochemische kunststoffen en biomaterialen voor elektronica zijn verschillende keerzijden van dezelfde medaille: extreem duurzaam versus biologisch afbreekbaar. De in elektronica gebruikte petrochemische materialen hebben geen last van natuurlijke veroudering ten gevolge van vocht en temperatuur. Ze zijn zo duurzaam dat ze helemaal niet vergaan en bergen e-afval veroorzaken die we niet goed recyclen. Biomaterialen breken na verloop van tijd af, wat geweldig klinkt als een apparaat aan het eind van zijn levensduur is, maar niet werkelijk leuk klinkt als je het apparaat nog aan het ontwerpen bent.

We willen eigenlijk dat de meeste van onze elektronische apparaten uiteindelijk vergaan. Sommige dingen zijn voor eenmalig gebruik, zoals RFID-tags, sensoren en medische kits. De meeste andere apparaten willen we niet tientallen jaren – of zelfs maar één decennium – gebruiken. Misschien zou in plaats van onbeperkte duurzaamheid biologisch afbreekbare elektronica de standaard moeten zijn bij de fabricage. En als je echt

iets moet produceren met kunststoffen of harsen die eeuwig meegaan, moet je ook een plan presenteren over hoe je het kostbare elektronische apparaat de komende 50 jaar (minstens) gaat onderhouden.

Een verantwoord beheer van materialen is aan de gang en is opgenomen in beleid zoals de EU/VS Green Deals en de komende klimaatwetgeving. Er zijn enkele fantastische ontwikkelingen in de materiaalkunde die ‘de verwezenlijking van een biogebaseerde economie’ moeten versnellen en die relevant zijn voor de productie van elektronica. Maar ik heb er nog geen gevonden die op grotere schaal beschikbaar is. De meeste onderzoeksprojecten lijken klaar om in de komende vijf jaar de chemische en/of productiemarkt te betreden. Ik verwacht grote bio-zaken. ◀

220556-03

WORLD ETHICAL ELECTRONICS FORUM

World Ethical Electronics Forum 2023

In november 2021 lanceerde Elektor het **World Ethical Electronics Forum** (WEEF) in München, Duitsland. Het evenement inspireerde wereldwijde vernieuwers in elektronica met een open discussie over ethiek en duurzame ontwikkelingsdoelen (SDG's). Naast engineers en redacteurs van Elektor bestond de lijst van sprekers en panelleden uit dr. Stefan Heinemann (Professor of Business Ethics aan de FOM University of Applied Sciences), dr. Paula Palade (PhD, Jaguar Land Rover), Margot Coojmans (directeur, Philips Foundation), en verschillende andere thought leaders, waaronder Priscilla Haring-Kuipers. Bezoek de WEEF-webpagina om op de hoogte te blijven en mee te doen: www.worldethicalelectronicsforum.com.

WEB LINKS

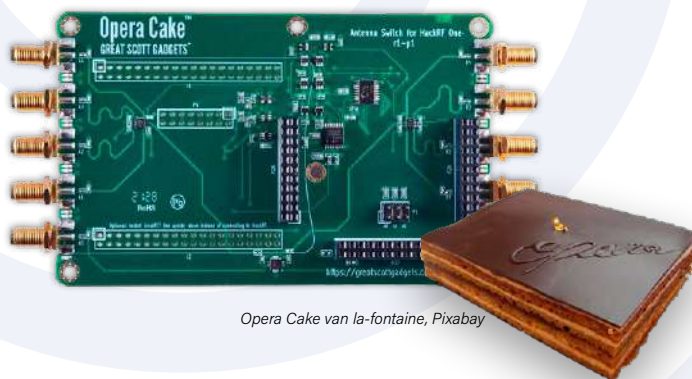
- [1] eWaste Ben: <https://www.youtube.com/c/eWasteBen>
- [2] Custom Materials, Inc., “Is Epoxy Resin Plastic?” January 2, 2022: <https://www.youtube.com/watch?v=i8nEQQ9S0V0>
- [3] NIH, “Bioplastics and Carbon-Based Sustainable Materials, Components, and Devices: Toward Green Electronics,” October 20, 2021: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8532127/>
- [4] O. Ogunseitan, et al, “Biobased materials for sustainable printed circuit boards,” Nature Reviews, September 12, 2022: <https://www.nature.com/articles/s41578-022-00485-2#citeas>

De Opera Cake antenneswitch voor HackRF One

sluit tot acht antennes aan op uw SDR

Clemens Valens (Elektor)

De Opera Cake van Great Scott Gadgets is een antenneswitch voor de HackRF One SDR. Hiermee kunt u maximaal acht antennes op uw apparaat aansluiten of een schakelbare filterbank tussenvoegen.



Opera Cake van la-fontaine, Pixabay

In navolging van een goede traditie om elektronische gadgets te vernoemen naar cakes, gebak en snoep (denk aan Raspberry & Banana Pi, Snickerdoodle [1] en de vroege versies van Android), is de Opera Cake antenneswitch een apparaat dat ook al geen enkele relatie heeft met zijn naam (Pi verwees tenminste nog naar Python).

De Opera-cake die ik zelf ken (en zelfs een paar keer heb gebakken) is een meerlaagse cake die vooral naar koffie smaakt. Misschien is dat dan de relatie met het apparaat dat hieronder wordt besproken? Je hebt namelijk veel koffie nodig om wakker te blijven tijdens de lange dagen van plezier met de SDR.

De Opera Cake antenneswitch...

Zoals hierboven al vermeld is de Opera Cake een antenneswitch voor de HackRF One [2], te zien in **figuur 1**. Eigenlijk is het een dubbele vierweg-antenneswitch, d.w.z. twee enkelpolige viervoudige (SP4T) schakelaars. U kunt de twee schakelaars - ook wel banks genoemd - parallel gebruiken als een DP4T-schakelaar, of in serie (soort van) als een SP8T-schakelaar. Deze schakelaars zijn bidirectioneel, dus ze werken zowel in RX- als TX-modus.

Laten we beginnen met hoe de Opera Cake van de Great Scott Gadgets site eruitziet. Het is eigenlijk gewoon een 120 bij 75 mm vierlaags print (wacht eens even, vier lagen, dus gelaagde cake, hmm...) met veel connectoren. Er zitten vijf female SMA-connectoren aan beide uiteinden van de print, en drie stackable headers erop gemonteerd, met voorbereiding voor nog eens drie.

... is een add-on board

Het board heeft verrassend weinig elektronische onderdelen. Dit komt omdat de Opera Cake een add-on board is. Hij heeft geen intelligentie nodig, want hij wordt bestuurd door het board waarop hij wordt aangesloten. Qua afmeting is hij ontworpen om op een HackRF One te worden gepluigd. Maar indien u de juiste connectoren op het board soldeert, kunt u er ook een GreatFET One [3] op aansluiten (ondersteboven).

Merk op dat, wanneer uw HackRF One in een behuizing zit, u deze er eerst uit moet halen (voorzichtig!) voordat u de Opera Cake erop kunt pluggen.

Configureerbaar

De eerste toepassing van de antenne switch die vast in u opkomt, is het aansluiten van meerdere antennes op een enkele HackRF One, zodat u deze niet elke keer opnieuw hoeft om te pluggen als u van RF-band wisselt.

Het schakelen van antennes kan handmatig worden gedaan met een klein software hulpprogrammaatje, maar het kan ook automatisch worden gedaan met de HackRF One-firmware, en dan op basis van frequentie of tijd. Frequentie-gebaseerd schakelen maakt bijvoorbeeld breedband spectrum analyzer toepassingen mogelijk, terwijl tijd-gebaseerd schakelen andere spannende dingen ondersteunt, zoals pseudo-Doppler richtingsbepaling. [4]

Wanneer u de switch in de DP4T-modus gebruikt kunt u deze gebruiken om bijvoorbeeld verzwakkers of filters in het antenepad op te nemen. Op deze manier kan de Opera Cake functioneren als een geschakelde filterbank.

Stapelbaar

Weet ook dat er maximaal acht Opera Cakes op elkaar kunnen worden gestapeld (een gelaagde SDR-cake ...), waardoor verschillende switch-configuraties mogelijk zijn. In de handmatige modus kan elk board afzonderlijk worden bediend, maar in de automatische frequentie en tijd schakelopties schakelen alle boards op hetzelfde moment en op dezelfde manier. Een stapel van twee boards zou een DP8T-configuratie mogelijk maken, waarbij bijvoorbeeld tussen acht filters kan worden geschakeld.

Waarom zou u er een willen?

De Opera Cake is geen nieuw ontwerp, het dateert uit 2016 of misschien nog wel langer geleden. Wat er nieuw aan is, is dat je hem nu kunt kopen als een volledig geassembleerde en geteste module. Want tot nu toe, als je er een wilde, moest je hem zelf in elkaar zetten, op basis van de ontwerpbestanden die op GitHub werden gepubliceerd. [5]

Het zal u niet verbazen dat de mooi geassembleerde Opera Cake antenne switch de perfecte metgezel is voor de HackRF One. In plaats van elke keer uw hardware opnieuw te bedraden, of de lengte van uw antenne aan te passen telkens wanneer u in een andere band wilt werken, kunt u tot acht van uw favoriete antennes tegelijk aansluiten (en aangesloten houden).

Een paar opmerkingen

Afhankelijk van het bouwjaar van uw HackRF One moet u mogelijk de firmware naar de nieuwste versie upgraden, om deze goed te laten werken met de antenne switch. Instructies over hoe dit te doen zijn beschikbaar op de Great Scott Gadgets website [6].

Houd er rekening mee dat, net als bij HackRF One, de SMA-connectoren op de Opera Cake female types zijn. Dit betekent dat het alleen werkt met antennes die zijn uitgerust met een male connector (d.w.z. met een pinnetje in het midden, zie **figuur 2**). Wees daarom voorzichtig en selecteer verstandig.

Een HackRF One met één of meerdere Opera Cake boards erop gestapeld past niet meer in zijn (krappe) behuizing. Dit maakt het systeem natuurlijk kwetsbaar voor stof en schroevendraaiers, of andere metalen voorwerpen die er per ongeluk in en op kunnen vallen. Afhankelijk van hoe u uw antennes en filters bedient kunnen de **Reset** en **Power** drukknoppen enigszins ontoegankelijk worden (**figuur 3**). Omdat de stackable headers P20, P22 en P28 alleen stroom en wat digitale signalen transporteren, en geen RF-signalen, is het eventueel mogelijk om headers tussen te voegen om de afstand tussen de boards te vergroten. Tot slot: een (kort) male-male SMA-kabeltje, om de Opera Cake aan te sluiten op de antenne-ingang van HackRF One, is helaas niet inbegrepen, dus dient u zelf te regelen. ◀

220602-03

Vragen of opmerkingen?

Heeft u technische vragen of opmerkingen over dit artikel? Neem dan contact op met Clemens Valens via clemens.valens@elektor.com.



Gerelateerde producten

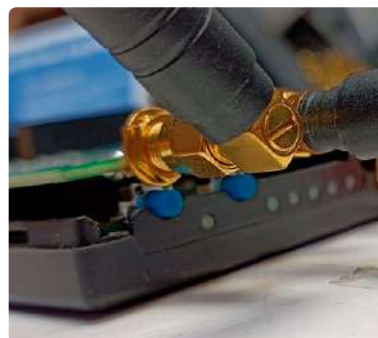
- Opera Cake - Antenneschakelaar voor HackRF One (SKU 20083) www.elektor.nl/20083
- HackRF One Software Defined Radio (1 MHz tot 6 GHz) (SKU 18306) www.elektor.nl/18306
- ANT500 Telescopic Antenna (75 MHz tot 1 GHz) (SKU 18481) www.elektor.nl/18481
- ANT700 Telescopic Antenna (300 MHz tot 1100 MHz) (SKU 18480) www.elektor.nl/18480



Figuur 1: De Opera Cake van Great Scott Gadgets is een antenneswitch voor de HackRF One.



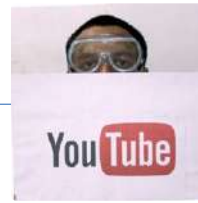
Figuur 2: Antennes met SMA-connectoren. Links is correct, en rechts dus niet, bij de Opera Cake en HackRF One.



Figuur 3: Misschien wilt u stackable headers tussenvoegen om zo de ruimte tussen de twee boards te vergroten.

WEBLINKS

- [1] Snickerdoodle on Elektor Labs: <https://www.elektormagazine.com/labs/snickerdoodles-with-zynq>
- [2] D. Meyer, "HackRF One SDR Transceiver: From 1 MHz to 6 GHz," Elektormagazine.com: <https://www.elektormagazine.com/news/hack-rf-one-sdr-transceiver>
- [3] C. Valens, "Introducing the GreatFET One Dual-Core Microcontroller Board for Python," Elektormagazine.com: <https://www.elektormagazine.com/news/greatfet-one>
- [4] Pseudo-Doppler direction finding: <https://www.rtl-sdr.com/pseudo-doppler-direction-finding-with-a-hackrf-and-opera-cake/>
- [5] Opera Cake on GitHub: <https://github.com/greatscottgadgets/hackrf/tree/master/hardware/operacake>
- [6] HackRF One firmware: <https://github.com/greatscottgadgets/hackrf/releases/latest>



Engineering met de **Arduino**, en meer

een interview met auteur Ashwin Pajankar

By Alina Neacsu (Elektor)

De Elektor-community zit vol met technici die graag hun kennis delen met gelijkgestemde elektronica-liefhebbers. Neem Ashwin Pajankar, een ingenieur, docent, Elektor-auteur en YouTuber die werkt vanuit Nashik, India. Als hij al niet bezig is met een nieuw elektronica project, dan is hij wel bezig met het helpen van zijn peers – door middel van boeken, cursussen en video's - om hen zo te leren engineeren met de Arduino en met andere moderne technologie.

Alina Neacsu: Allereerst, bedankt dat je de tijd neemt om deze vragen te beantwoorden. Vertel eens iets over jezelf. Wat is je huidige beroep?

Ashwin Pajankar: Ik woon in de buitenwijken van de stad Nashik in India. Ik heb een Bachelor of Engineering in Computer Science and Engineering, behaald aan het Shri Guru Gobind Singhji Institute of Engineering and Technology te Nanded. Ik heb ook een Master of Technology behaald (ook in Computer Science and Engineering) aan het International Institute of Information Technology te Hyderabad. Ik spreek vijf talen en heb in drie deelstaten in India gewoond en gewerkt.

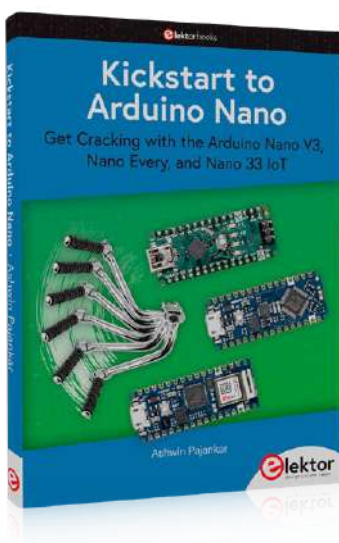
Ik werk momenteel als freelance technisch schrijver, YouTuber en instructeur bij Udemy. Ik organiseer ook bootcamps rond programmeren en elektronica voor professionals en studenten. Ik heb een voorliefde voor wiskunde, natuurkunde, computers en elektronica.

Alina: Herinner je je nog je eerste elektronica project met microcontrollers? Kun je daar iets over vertellen, zoals welke technologie je toen hebt gebruikt?

Ashwin: Mijn allereerste microcontroller-project was een heel bescheiden project. In groep 12 ben ik voor het eerst aan de slag gegaan met microcontrollers (zoals de 8085 en 8086 microcontrollers). Ik heb les gehad in de 8051 microcontroller. (Het is de standaard microcontroller die in heel India wordt gebruikt bij lessen over microcontrollers.) Daar hadden we in groep 12 echter nog geen hands-on onderwijs mee. Toen ik daarna engineering ging studeren (een vierjarige opleiding op bachelorniveau) had ik tijdens het derde jaar van deze leergang de kans om met assembler op de 8051-kit te werken. Mijn allereerste project met de 8051 was een zeer bescheiden project: een knipperende LED met behulp van 8051 assembler-code.

Alina: Het Arduino ecosysteem, wat is dat nou precies? En wat zijn zoal de voor- en nadelen?

Ashwin: Arduino is een open source hardware- en software-ecosysteem. Omdat het ontwerp open source is, kunnen mensen en organisaties hun eigen ontwerpen maken en ideeën toevoegen. De software is ook open source en de broncode is te vinden op github.com/arduino. Men kan hier een bijdrage aan leveren, of dit aanpassen om een eigen variant te maken. Ook zijn er veel ondersteunende bronnen beschikbaar op het internet. Je kunt dus met behulp van de Arduino meteen aan



Lees *Kickstart to Arduino Nano* voor een eerste kennismaking met de Arduino Nano V3, de Nano Every, en de Nano 33 IoT.



de slag met elektronica en programmeren in C zonder heel veel geld uit te hoeven geven. Hieruit blijken meteen de voordelen van het Arduino ecosysteem. Het enige nadeel dat ik kan bedenken is dat er soms namaakproducten met het merk Arduino worden verkocht aan nietsvermoedende enthousiastelingen.

Alina: Wat vond je zo bijzonder aan het Arduino Nano 33 IoT board?

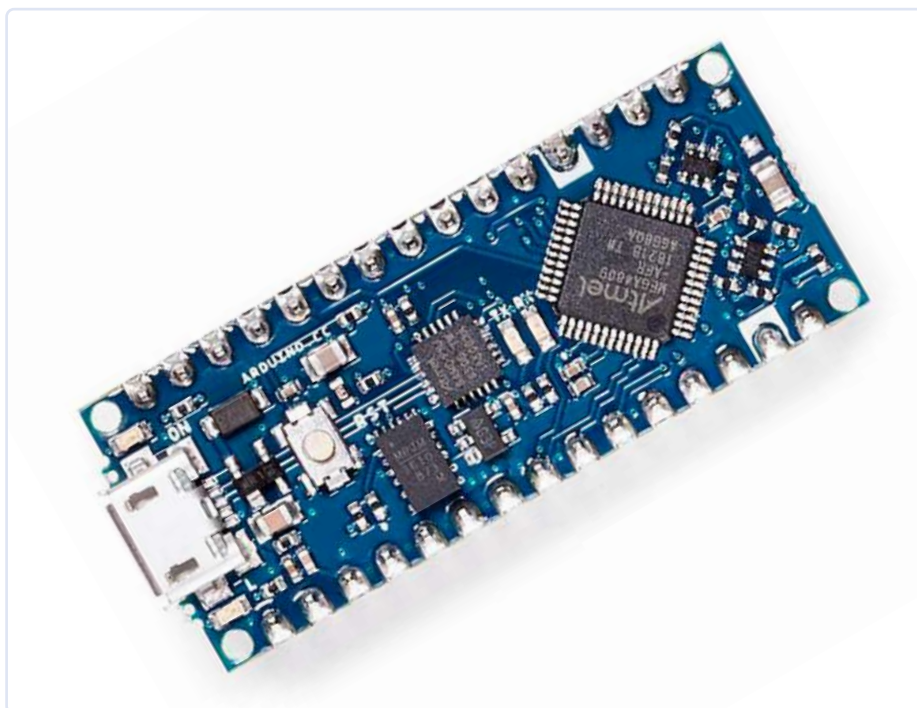
Ashwin: Welnu, voorheen, toen we elk Arduino board nog apart op WiFi moesten aansluiten, moesten we speciale shields gebruiken (die een beetje moeilijk te krijgen zijn in India), of de ESP-01 gebruiken. Werken met de ESP-01 is een beetje ingewikkeld en beginners vinden het vaak een uitdaging. De Arduino Nano 33 IoT is dan een perfecte kant-en-klare IoT-oplossing. Deze heeft al een NINA-W102 WiFi-module ingebouwd. Ik controleer regelmatig *arduino.cc* op nieuwe producten en was erg blij om het nieuwe board met ingebouwde WiFi te zien. Dus zodra deze in India beschikbaar kwam kocht ik hem, en ging ik ermee experimenteren.

Alina: Heb je advies voor wie geïnteresseerd is om aan de slag te gaan met de Arduino Nano?

Ashwin: Ja. Raadpleeg de online documentatie op de Arduino homepage. Mijn boek, *Kickstart to Arduino Nano* (Elektor, 2022), bevat gedetailleerde en stapsgewijze instructies.

Alina: Hoe ben je tot het schrijven van je eerste boek gekomen. Was het een uitdaging?

Ashwin: Ik werkte nog als IT-technicus in Bangalore, India, toen ik mijn allereerste boek schreef en publiceerde. Het boek ging over de Raspberry Pi. Het richtte zich op computer vision. Hoewel dit mijn allereerste ervaring was met het publiceren van een boek ging het me makkelijk af, want ik hield toch al van programmeren, elektronica en schrijven. En het schrijven van boeken stelt me in staat om al mijn passies samen te brengen tot iets heel tastbaars.



Het boek van Ashwin Pandakar bevat een uitgebreide kickstart met de Arduino Nano.

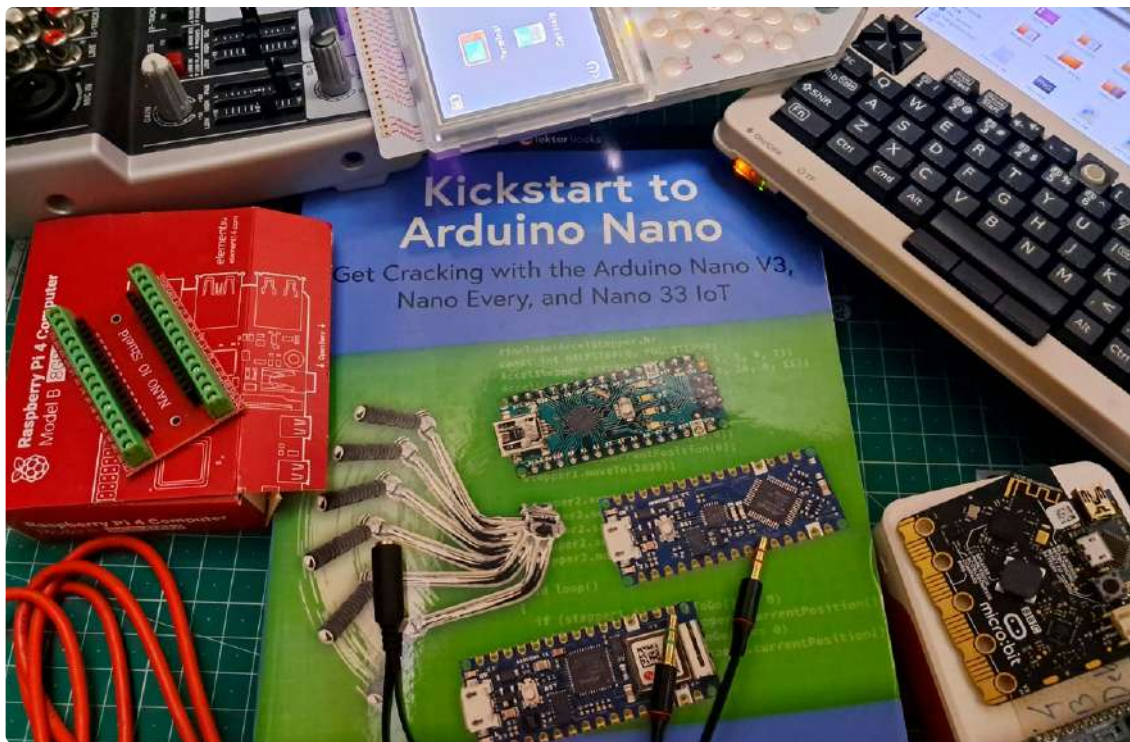
Alina: Je vermeldt in je boek dat je tijdens het schrijven in India was, waar je momenteel woont. Hoe kwam het boek dan tot stand tijdens de COVID-19 pandemie?

Ashwin: De COVID-19 pandemie was een beproeving voor de hele menselijke beschaving. India behoorde tot de zwaarst getroffen landen, met een van de hoogste ziekenhuis-opnamecijfers en het hoogste dodental. Nashik, mijn huidige woonplaats, had dan ook nog het hoogste dodental in India. Dat was zeer verontrustend, omdat niemand ooit eerder zo'n catastrofe had meegemaakt. Er waren constant lockdowns, avondklokken en beperkingen. Er was voortdurend een tekort aan voedsel, medicijnen en andere benodigdheden. Er waren geen bedden beschikbaar in ziekenhuizen. Ambulance- en uitvaartdiensten waren overbelast. Zelf heb ik de ziekte zelfs twee keer opgelopen. De eerste keer werd ik opgenomen in een door de overheid gerund ziekenhuis, en de tweede keer werd ik thuis behandeld. Ik herinner me nog goed dat ik rondstruinde om anti-griep medicijnen voor mezelf en mijn burens te vinden, omdat deze erg schaars waren.

Het schrijven van het boek gaf me een doel in deze donkere periode. Door mijn isolatie, en het bijna stilvallen van nagenoeg alle andere aspecten van het leven, had ik de mogelijkheid om me volledig te concentreren op het schrijven van het boek. Ook is het team van Elektor zeer behulpzaam geweest en hebben ze mij bij elke stap begeleid. Er zijn bijna geen woorden voor om mijn oprechte dankbaarheid naar hen uit te spreken voor de grote steun die ze me steeds geboden hebben.

Alina: Je bent ook sterk betrokken bij online lesgeven, vooral via Udemy. Kun je je ervaringen hiermee met ons delen? Hoe gaan studenten met je om in vergelijking met traditionele klassieke lessen?

Ashwin: Ik geef de voorkeur aan online lesgeven omdat het mijn bereik vergroot. Ik heb in het verleden ook programmeerbootcamps gehouden met live-publiek. Online lesgeven helpt me om honderdduizenden studenten te bereiken. Mensen kunnen met mij in contact komen door de vragen op het portal te stellen. En omdat alle cursussen op eigen tempo kunnen



In de werkruimte van Ashwin Pajankar zie je allerlei items

worden gevolgd is de betrokkenheid van studenten veel hoger.

Alina: Wanneer realiseerde je je dat je het leuk vindt om aan andere mensen les te geven over je favoriete onderwerpen?

Ashwin: Tijdens de zomervakanties van mijn engineering-studies gaf ik les aan de studenten op de middelbare school, en hielp ik hen zich voor te bereiden op toelatingsexamens voor engineering. Ik gaf dan les in wiskunde, natuurkunde, informatica en Engels. Toen realiseerde ik me dat ik graag lesgeef en kennis toegankelijker maak.

Alina: Wat is uiteindelijk het belangrijkste dat je je lezers zou willen leren?

Ashwin: De belangrijkste takeaway die ik voor mijn lezers heb, is om zelf de wereld van de technologie te leren verkennen. Binnen deze wereld is het heel makkelijk manoeuvreren als je eenmaal weet waar je relevante kennis kunt vinden. Men kan al veel leren door alle online documentatie, codevoorbeelden en verschillende technische discussieforums zelf door te nemen.

Alina: Ben je momenteel met iets bezig? Nog nieuwe ideeën voor een boek?

Ashwin: Ja. Ik heb gewerkt met de Raspberry Pi 4. Tijdens de lockdown was er een tekort aan aanbod. De situatie is echter recent verbeterd en kon ik een Raspberry

Pi 4 met 8 GB RAM aanschaffen, waar ik mee heb geëxperimenteerd. Zodra ik klaar ben met mijn eigen project, ben ik klaar om een boek over de Raspberry Pi voor Elektor te schrijven. Ik ben dol op het format van Elektor, en Elektor heeft het beste ontwerp-team. Ik ben dan ook van plan om met Elektor nog veel meer technische boeken te schrijven over uiteenlopende onderwerpen.

Alina: Kun je ons bijpraten over je eigen projecten die je momenteel hebt lopen?

Ashwin: Ik ben recent begonnen aan het vernieuwen van mijn YouTube-kanaal en het werven van meer publiek hiervoor. Binnenkort ga ik veel video's uploaden die wiskunde op schoolniveau in Hindi onderwijzen (een taal die door een miljard mensen over de hele wereld wordt gesproken). Daarna zal ik basis natuurkunde en elektrotechniek behandelen (opnieuw in het Hindi).

Alina: Heb je iets bereikt, of ergens aan bijgedragen, waar je nu het meest trots op bent?

Ashwin: Wiskunde, programmeren en elektronica zijn de dingen die ik doe om de kost te verdienen. Afgezien van mijn professionele werk, ben ik ook altijd actief geweest om wat terug te geven aan de gemeenschap, via social outreach-projecten op plekken en universiteiten waar ik studeerde. Nadat ik als freelancer aan

de slag ben gegaan doe ik dat op persoonlijke titel nog steeds. Mijn deelname aan de educatieve outreach voor kansarme kinderen via het 'IIIT-H community outreach programma' van mijn universiteit had een interview tot gevolg in een van de meest prominente kranten in Hyderabad. Verder heb ik verschillende prijzen ontvangen voor het dienen van de gemeenschap met educatieve outreach. Dit zijn zaken in mijn leven waar ik het meest trots op ben. ◀

220626-03

Vragen of opmerkingen?

Indien u vragen heeft over dit interview stuur dan gerust een e-mail naar Alina Neacsu via alina.neacsu@elektor.com, of naar de redactie van Elektor via redactie@elektor.com.



Gerelateerde producten

- > **A. Pajankar, Kickstart to Arduino Nano (Elektor 2022)**
www.elektor.nl/20241
- > **A. Pajankar, Kickstart to Arduino Nano (PDF, Elektor 2022)**
www.elektor.nl/20242
- > **Arduino Nano Every met Headers (SKU 19939)**
www.elektor.nl/19939

LiDAR-precisiemeter

meet tot 12 meter



Somnath Bera (India)

LiDAR's zijn een prachtige uitvinding voor het detecteren van obstakels en het meten van afstanden. Hier gebruiken we er een om precieze afstandsmetingen uit te voeren tot twaalf meter met een resolutie van 1 cm.

Mijn aanvankelijke gedachte was om een TFMini-S LiDAR te kopen en er een soort radar mee te bouwen. Maar, zoals zo vaak, kwam daar niet veel van – tot ik op mijn werk een inval kreeg.

Electric Resistance Welded-leidingen (ERW) bestaan uit lange buizen met een diameter van 300 mm of meer en met een lengte tot twaalf meter. Ze worden in onze elektriciteitscentrale gebruikt voor de afvoer van as-slurry. Op een dag, toen ik bezig was met deze lange buizen voor de assemblage van de afvoer-pijpleidingen, viel me op dat mijn team elke buis nauwkeurig moest meten voordat deze door de uitgangspoort naar buiten werden gebracht. Elke dag moest er minstens 30 tot 50 keer iemand tijd vrijmaken om deze metingen uit te voeren. Buiten onder een brandende zon van 47°C, met het meetlint aan het ene uiteinde en een opzichter aan het andere uiteinde, en dan de lengte opnemen, is een uitputtende klus. Hier is het idee geboren om een precisie-metinstrument te ontwerpen.

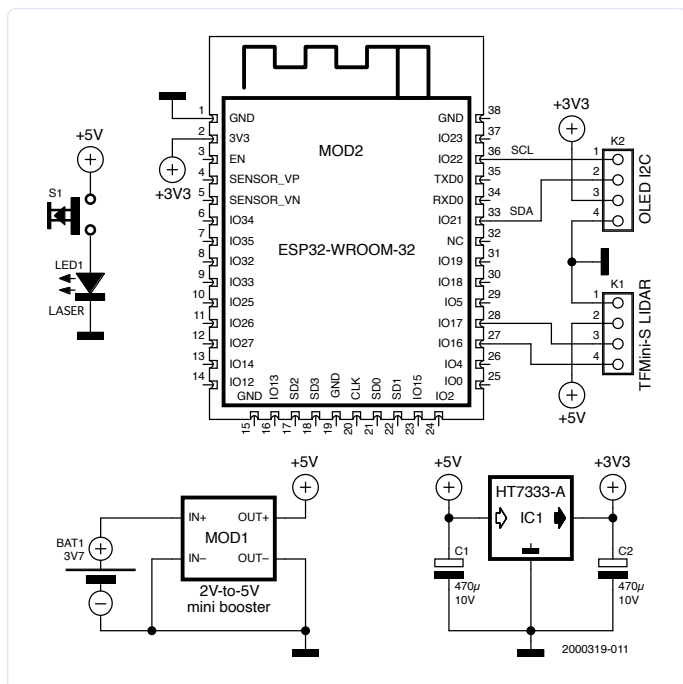
LiDAR-modules

De *TFMini-S single-point ranging LiDAR* (figuur 1) kan zeer snel (100 Hz) en zeer nauwkeurig (1 cm resolutie, 1% nauwkeurigheid) afstandsmetingen uitvoeren tot maximaal 12 m, wat perfect is voor de beoogde taak. Er bestaat ook een Plus-versie die sneller is (1 kHz) en een IP65-behuizing heeft. Beide zijn voorzien van een seriële interface. Door de LiDAR aan het ene uiteinde van de buis te plaatsen en vervolgens op de eindplaat aan het andere uiteinde te richten, kan de lengte van de buis nauwkeurig worden gemeten. Beide versies van de LiDAR hebben een gezichtsveld (*Field of View, FOV*) van 3,4 graden; daarom is een goede uitrichting van de laser zeer belangrijk voor het verkrijgen

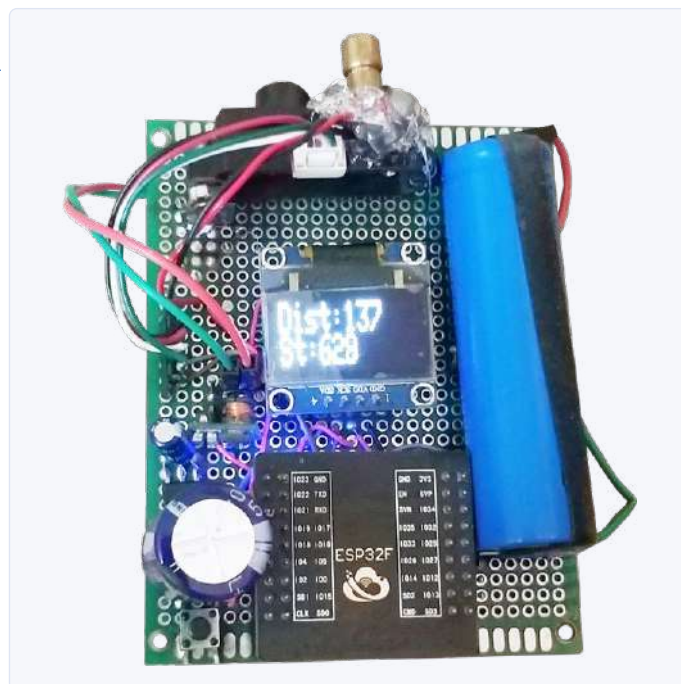
van correcte metingen. Om dit gemakkelijker te maken, hebben we een laserpointer met drukknopbediening aan het systeem toegevoegd. Ik gebruikte een ESP32 module om de LiDAR aan te sturen en de berekeningen uit te voeren. Een klein OLED-display toont de resultaten van de afstandsmetingen. Een lopend gemiddelde van een paar metingen verhoogt de nauwkeurigheid. Het resultaat kan dan worden weergegeven als een floating-point waarde, maar omwille van de snelheid en eenvoud houden we het op een gewone integer-waarde. De TFMini LiDAR's maken gebruik van de reflectie van onzichtbaar infrarood laserlicht van het oppervlak van het object waarop ze zijn gericht. Indien het voorwerp het binnenkomende laserlicht niet reflecteert, volledig absorbeert of breekt, zal de meting foutief zijn. Vermijd bijvoorbeeld water en schuine en sterk reflecterende glazen ramen. De meeste andere voorwerpen weerkaatsen voldoende licht zodat zelfs wanneer het voorwerp beweegt, goede metingen kunnen worden uitgevoerd.

Figuur 1. De goedkope TFMini-S LiDAR heeft een bereik van 12 m met een resolutie van 1 cm.





Figuur 2. Het prototype maakt gebruik van een ESP32-WROOM-module zonder 5V-regelaar. Bij gebruik van een DevMod-C- of Pico-Kit-module kan IC1 worden weggelaten. Sluit in dat geval de 5V-voeding aan op de 5V-ingang van de module, niet op de 3,3V-pin!



Figuur 3. Het prototype van de op LiDAR gebaseerde precisiemeter is opgebouwd op gaatjesprint. Merk op hoe de laserpointer boven op de LiDAR-module is gelijmd.

Bouw van de afstandsmeter

Bij het bouwen van het apparaat moest ik er rekening mee houden dat het draagbaar moest zijn, zodat mijn team het in het veld kon gebruiken. Het apparaat moet dus op batterijen werken. De schakeling waar ik uiteindelijk op uitkwam is te zien in **figuur 2**. Zowel de LiDAR als de laserdiode die ik voor het richten gebruik, hebben 5 V nodig. De ESP32 WROOM module draait op 3,3 V. Daarom heb ik een mini-boostconverter toegevoegd die een 5V-voedingsspanning genereert uit een ingangsspanning van slechts 2 V. Ik heb een low-dropout HT7333-A spanningsregelaar aangesloten op de 5V-uitgang om 3,3 V te maken voor de ESP32. Als je een ESP32 module met ingebouwde spanningsregelaar gebruikt, dan heb je de HT7333-A niet nodig. De volledige opstelling werkt op twee batterijen van 1,5 V of op een LiPo-accu van 3,7 V. Merk op dat de HT7333-A een SMD-component is; wees daarom voorzichtig bij het solderen.

Een woordje over de software

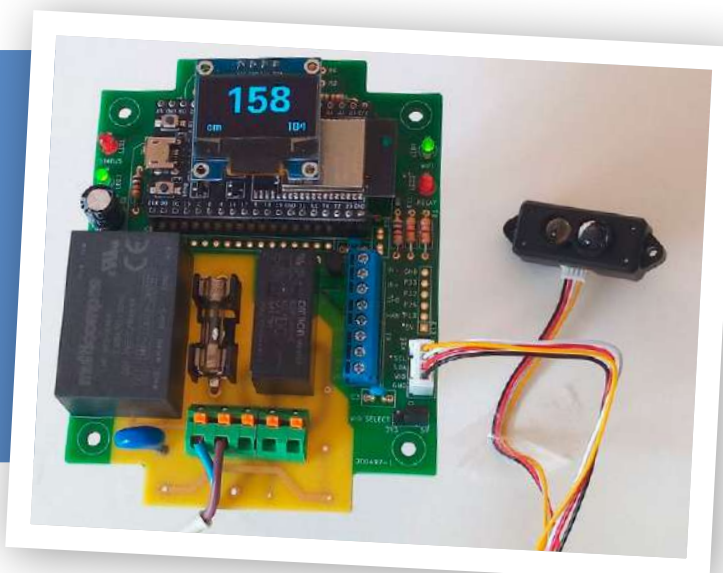
De software die ik voor het apparaat schreef is een Arduino-sketch en kan worden gedownload van [1]. Nodig zijn de TFMMini-bibliotheek en de GFX SSD1306-bibliotheken van Adafruit, die beschikbaar zijn via de bibliotheekmanager van de Arduino IDE. De LiDAR en de ESP32 communiceren met elkaar via een seriële poort (Serial1). Elke 25 ms wordt een nieuw datapunt opgevraagd. Al het zware werk wordt gedaan door de TFMMini-bibliotheek, terwijl de sketch slechts een gebruikersinterface biedt. De waarden van de afstand en de signaalsterkte worden weergegeven op het OLED-display en worden ook doorgegeven via de seriële 'Arduino'-poort (Serial0).

Gebruik van het apparaat

Nadat alle onderdelen op een stuk gaatjesprint zijn gemonteerd (**figuur 3**), wordt de LiDAR zodanig op de rand van de print beves-

Gebruik de Automator!

De Elektor Automator [2] is het perfecte platform om met de TFMMini-S LiDAR te spelen. Hij is gebaseerd op een ESP32-module, heeft een Grove-connector die compatibel is met de LiDAR-module, een OLED-display en de nodige voedingen. Wij (van Elektor Labs) hebben daarom voor dit project een Arduino-sketch gemaakt die op de Automator draait. Deze maakt gebruik van de TFLidar- en U8g2-bibliotheken, die beschikbaar zijn via de bibliotheekmanager van de Arduino IDE. De software kan gemakkelijk worden uitgebreid om WiFi- of Bluetooth-connectiviteit toe te voegen en om het relais en de LED's van de Automator te bedienen. De software kan worden gedownload van [2].





tigd dat hij een onbelemmerd uitzicht heeft. De laseraanwijzer is een gewone rode laserdiode die met een drukknop wordt bediend. Indien mogelijk wordt deze met hotmelt (of iets dergelijks) boven op de LiDAR bevestigd, zodat beide in dezelfde richting wijzen.

De minimale afstand die de TFMini-S LiDAR aankan, bedraagt 30 cm. Richt hem daarom op een object of muur op een afstand van 30 cm tot 12 m. Meet de afstand en controleer deze met een meetlint. Uiteindelijk zul je merken dat de meting tot op een centimeter nauwkeurig is. De tweede regel op het display geeft de sterkte van het ontvangen signaal aan. Naarmate het voorwerp zich verder weg bevindt, neemt de signaalsterkte af. ◀

200316-03

Vragen of opmerkingen?

Hebt u technische vragen of opmerkingen naar aanleiding van dit artikel? Stuur een e-mail naar de redactie van Elektor via redactie@elektor.com.



GERELATEERDE PRODUCTEN

- TFMini-S LiDAR (SKU 19691)
www.elektor.nl/19691
- TFmini Plus IP65 LiDAR (SKU 19690)
www.elektor.nl/19690
- ESP32-WROOM-32 (SKU 18421)
www.elektor.nl/18421 0.96"
- OLED Display (Blue, I²C, 4-Pin) (SKU 18747)
www.elektor.nl/18747

WEBLINKS

[1] Dit project op Elektor Labs: www.elektormagazine.com/labs/tfmini-lidar-precision-gauge-30-cm-to-1200-cm

[2] De Elektor Automator: www.elektormagazine.com/labs/automator

Advertentie

Deel je ideeën en elektronica projecten

*eender welke
moeilijkheidsgraad*

op www.elektor-labs.com
en wordt beroemd!!



Start nu een nieuw project op:
www.elektor-labs.com

design > share > earn



Audiosignalen en de ESP32

de ESP-ADF-omgeving in de praktijk

Tam Hanna (Hongarije)

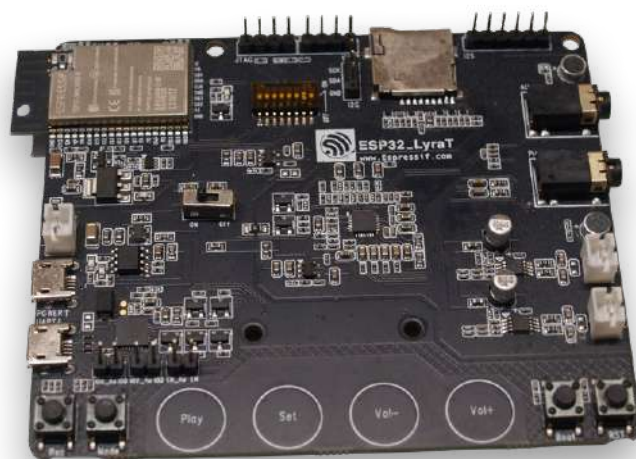
Het ontwikkelen van nieuwe consumentenelektronica is een echte uitdaging. Die producten hebben meestal een zeer korte levenscyclus, dus een snelle ontwikkeling is essentieel voor succes. Dat geldt zeker voor audiotoeepassingen, waar algoritmes worden gebruikt voor het implementeren van verschillende standaard codecfuncties. Het Espressif Audio Development Framework (ESP-ADF) is een krachtig gereedschap met functies die tijd en inspanning besparen voor ontwikkelaars van zulke toepassingen.

Espressif's Audio Development Framework biedt een verzameling van algoritmen en codecs in een gestandaardiseerd formaat. Om een toepassing te ontwikkelen, hoeft de ontwerper alleen maar de juiste "software-IC's" aan elkaar te koppelen, zonder zich bezig te houden met de interne details van die bouwstenen.

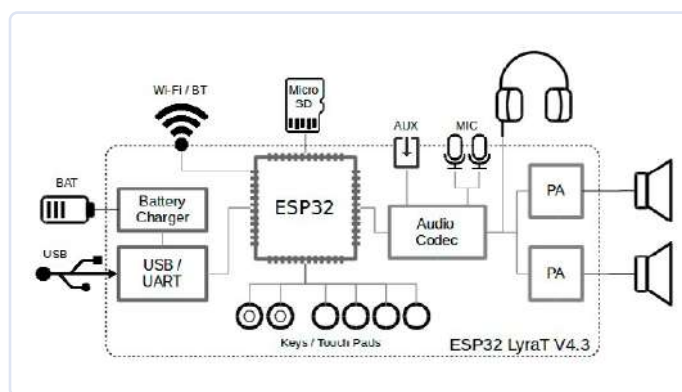
Het doel van dit artikel is om u enig begrip te geven van hoe ADF in de praktijk werkt. Ik heb Espressif's ADF-tool vaak gebruikt bij het implementeren van oplossingen volgens de ontwerpeisen van mijn klanten. Ik kan nu veel dingen met u delen, die ik graag had willen weten toen ik begon ADF te gebruiken.

Hardware to Go

Om te beginnen iets vanzelfsprekends. ESP-ADF is compatibel met alle ESP32-systemen. De library biedt een gestandaardiseerde driver-interface voor het "audiogedeelte" dat de data bewerkt.



Figuur 1: De kleine zwarte print bevat alles...



Figuur 2: ...wat u nodig hebt om audiotoeepassingen te ontwikkelen (bron: [6]).

Voordat u het hardwareontwerp gaat afronden, kunt u al veel van de software schrijven en testen met behulp van een standaard ontwikkelkaart. Espressif levert verschillende van die ontwikkelkaarten. Zelf gebruik ik graag het LyraT-board (**figuur 1 en 2**).



Het belangrijkste is het gedigitaliseerde “analog baseband”-signaal, dat wordt geleverd door de ES8388-chip (met behulp van twee analoge microfoons). De communicatie gaat via I2S en via een aantal gereserveerde pennen; het schema en de verbindingen zijn beschikbaar gesteld door Espressif en zijn eenvoudig te repliceren in het ontwerp van een apparaat. De chip is gemakkelijk verkrijgbaar bij UTSOURCE en LCSC.

Bij het werken met de LyraT is het aantal beschikbare GPIO-lijnen van de ESP32 op de kaart tamelijk beperkt. Als het hardwareontwerp van het apparaat ingewikkelder wordt, kunnen dat er te weinig zijn. In zulke gevallen kan het nodig zijn om een tweede processor te overwegen voor de hardwarecommunicatie.

Installeren van ADF

Espressif levert ADF als een plugin voor de bestaande IDF-omgeving. ADF kan niet worden gebruikt met Arduino, en ook niet met enkele versies van Espressif's eigen IDF.

De gemakkelijkste manier om de werkomgeving te installeren is de complete ESP-ADF-repository te downloaden. Om te beginnen gebruik ik de volgende commando's om een subdirectory, *esp4*, aan te maken, waar de repository lokaal wordt opgeslagen:

```
(base)tamhan@tamhan-thinkpad:~$ mkdir esp4
(base)tamhan@tamhan-thinkpad:~$ cd esp4/
```

Dan download ik de volledige repository met behulp van Git:

```
(base)tamhan@tamhan-thinkpad:~/esp4$ git clone
--recursive https://github.com/espressif/esp-adf.git
Cloning into 'esp-adf'...
```

Aan de output (die hier niet is weergegeven), is te zien dat de ADF-repository verschillende referenties maakt. Daarom is downloaden via de in GitHub geïntegreerde browser niet mogelijk.

Er zit ook een versie van de Espressif IDF bij de ADF. Voor sommige componenten is de aanwezigheid van de omgevingsvariabele `ADF_PATH` nodig. Als u vanaf de commandoregel wilt werken aan ADF-toepassingen, moet u die omgevingsvariabele definiëren met de volgende commando's:

```
(base)tamhan@tamhan-thinkpad:~/esp4$ cd esp-adf
(base)tamhan@tamhan-thinkpad:~/esp4/esp-adf$ export
ADF_PATH=$PWD
```

De aanwezigheid van de variabele `ADF_PATH` beïnvloedt soms normale IDF-projecten, ik probeer dat meestal te vermijden door een ander terminalvenster te gebruiken voor “normaal” ESP32-werk.

Voor de geïntegreerde IDF moet het gebruikelijke installatiescript worden gedraaid voor de compilers en andere afhankelijkheden:

```
(base)tamhan@tamhan-thinkpad:~/esp4/esp-adf$ cd
$ADF_PATH/esp-idf
(base)tamhan@tamhan-thinkpad:~/esp4/esp-adf/esp-idf$ ./
install.sh
Detecting the Python interpreter
. . .All done! You can now run:

. ./export.sh
```

Dan moet een script worden gedraaid voor het opzetten van de toolchain (de twee opeenvolgende punten zijn geen tikfout):

```
(base)tamhan@tamhan-thinkpad:~/esp4/esp-adf/esp-idf$ .
./export.sh
```

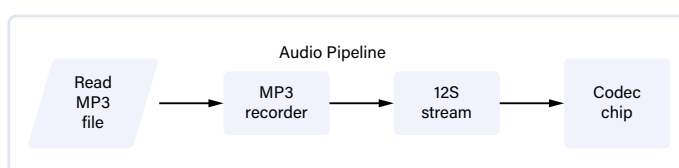
Het is aan te raden om nu een shell-script voor de parameterisatie aan te maken, zodat u met Bash automatisch kunt laten configureren als dat nodig is.

Softwarearchitectuur: de pijplijn

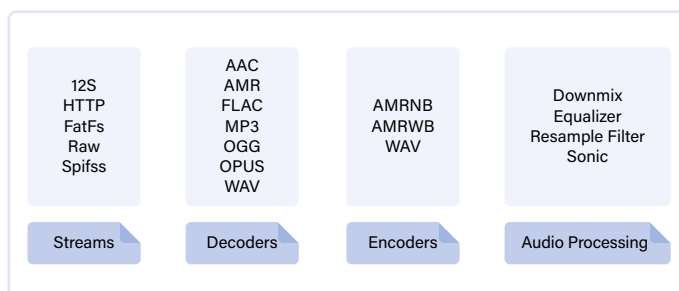
Na het downloaden en installeren van ADF is het tijd om na te denken over de theoretische structuur van het framework. Conform het concept van een “software-IC”, zoals ontwikkeld door The Stepstone Corporation, implementeert de ontwikkelaar met ADF in essentie een pijplijn die bestaat uit een reeks van verwerkingsstappen. In **figuur 3** ziet u een workflowmodel voor het implementeren van een MP3-speler.

In **figuur 4** ziet u de verschillende rollen en voorbeeldimplementaties die Espressif aanbiedt.

Het enige bijzondere aan het huidige systeem is dat het Unix-principe “alles is een file” hier ook van toepassing is op periferie.



Figuur 3: Deze pijplijn decodeert MP3's (bron: [3]).



Figuur 4: Pijplijnelementen zijn te verdelen in verschillende types (bron: [3]).

Met de API, die uitvoerig gedocumenteerd is onder [1] en het gemakkelijkst te begrijpen met behulp van de button-functies onder [2], kunt u “wrappers” aanmaken die u rechtstreeks kunt integreren in de pijplijn. Maar voor ontwikkelaars van toepassingen is dit alleen relevant omdat het idee om ook periferie te gebruiken als pijplijnelement soms niet vanzelfsprekend is.

API-documentatie voor de individuele pijplijnelementen is te vinden onder [3].

Softwarearchitectuur: bouw een MP3-speler

Laten we als praktische illustratie eens kijken naar één van de voorbeelden van Espressif: Als u een project wilt implementeren met behulp van ADF, kunt u het beste uitgaan van een soortgelijk, of minstens verwant, project tussen de voorbeelden. Ga naar de directory met voorbeelden met het commando:

```
cd $ADF_PATH/examples/
```

In de volgende stappen zien we hoe een klassieke MP3-speler is te bouwen. Ga naar de map `~/esp4/esp-adf/examples/get-started/play_mp3_control/main` en open de file `play_mp3_control_example.c` met uw favoriete editor.

Het belangrijkste is het startpunt, waar enkele member-variabelen worden aangemaakt. Ons voorbeeld heeft een pijplijn-object nodig en twee element-handles voor de individuele elementen:

```
void app_main(void)
{
    audio_pipeline_handle_t pipeline;
    audio_element_handle_t i2s_stream_writer, mp3_decoder;
    ...
}
```

Aan de hardware wordt in ADF gerefereerd met de abstractie *Board*. De parameterisatie kiest u in het framework *Menuconfig*. De commando's in de achterliggende code verwerken voornamelijk de compilatieconstanten:

```
audio_board_handle_t board_handle = audio_board_init();
audio_hal_ctrl_codec(board_handle->audio_hal,
AUDIO_HAL_CODEC_MODE_BOTH, AUDIO_HAL_CTRL_START);
. . .
int player_volume;
audio_hal_get_volume(board_handle->audio_hal,
    &player_volume);
```

Het is interessant dat de HAL ook de volumeregeling bevat. Daarna wordt het pijplijnobject opgezet:

```
audio_pipeline_cfg_t pipeline_cfg =
    DEFAULT_AUDIO_PIPELINE_CONFIG();
pipeline = audio_pipeline_init(&pipeline_cfg);
mem_assert(pipeline);
```

Dit zal ontwikkelaars met ervaring met ESP-IDF bekend voorkomen. Het pijplijnobject wordt gemaakt door een configuratiestruct mee te geven, dat dan van gebruiker naar gebruiker wordt doorgegeven.

Vervolgens worden de eigenlijke pijplijnelementen gegenereerd. Eerst de MP3-decoder:

```
mp3_decoder_cfg_t mp3_cfg =
    DEFAULT_MP3_DECODER_CONFIG();
mp3_decoder = mp3_decoder_init(&mp3_cfg);
audio_element_set_read_cb(mp3_decoder,
    mp3_music_read_cb, NULL);
```

Nu is de vraag (zeker als we kijken naar de pijplijn in figuur 3) hoe de data moet worden aangeleverd. Dat gaat met de methode `audio_element_set_read_cb`, die een functie als parameter krijgt. Deze functie heeft de volgende structuur:

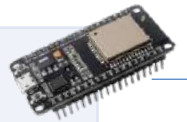
```
int mp3_music_read_cb(audio_element_handle_t el,
    char *buf, int len,
    TickType_t wait_time, void *ctx) {
    int read_size = file_marker.end
        - file_marker.start - file_marker.pos;
    if (read_size == 0) {
        return AEL_IO_DONE;
    } else if (len < read_size) {
        read_size = len;
    }
    memcpy(buf, file_marker.start +
        file_marker.pos, read_size);
    file_marker.pos += read_size;
    return read_size;
}
```

De callback levert de informatie die de codec moet verwerken via de buffer. De I2S-stream is dan eenvoudiger; die wordt doorgegeven van `AUDIO_STREAM_WRITER` en geparametriseerd als een output zodat de binnenkomende informatie naar de I2S-geluidshardware wordt geleid:

```
i2s_stream_cfg_t i2s_cfg = I2S_STREAM_CFG_DEFAULT();
i2s_cfg.type = AUDIO_STREAM_WRITER;
i2s_stream_writer = i2s_stream_init(&i2s_cfg);
```

Het aanmaken van pijplijnelementen maakt niet automatisch dat ze worden gedeclareerd. Dat is ten behoeve van ontwikkelaars die meerdere pijplijnen tegelijk willen gebruiken. Het opzetten van de pijplijn vindt plaats door het registreren van de individuele delen, die elk worden geïdentificeerd door een string:

```
audio_pipeline_register(pipeline, mp3_decoder, "mp3");
audio_pipeline_register(pipeline, i2s_stream_writer,
    "i2s");
```

Listing 1: Eventverwerking voor een MP3-speler.

```
while (1) {
    audio_event_iface_msg_t msg;
    esp_err_t ret = audio_event_iface_listen(evt, &msg, portMAX_DELAY);
    if (ret != ESP_OK) {
        continue;
    }
    if (msg.source_type == AUDIO_ELEMENT_TYPE_ELEMENT && msg.source == (void *) mp3_decoder
        && msg.cmd == AEL_MSG_CMD_REPORT_MUSIC_INFO) {
        audio_element_info_t music_info = ;
        audio_element_getinfo(mp3_decoder, &music_info);
        ESP_LOGI(TAG, "[ * ] Receive music info from mp3 decoder, sample_rates=%d, bits=%d, ch=%d",
            music_info.sample_rates, music_info.bits, music_info.channels);
        audio_element_setinfo(i2s_stream_writer, &music_info);
        i2s_stream_set_clk(i2s_stream_writer, music_info.sample_rates,
            music_info.bits, music_info.channels);
        continue;
    }
}
```

```
const char *link_tag[2] = {"mp3", "i2s"};
audio_pipeline_link(pipeline, &link_tag[0], 2);
```

De pijplijn wordt dan opgezet met behulp van een array waarin de individuele string-ID's in de juiste volgorde worden opgenomen. Voor de MP3-speler ontbreekt nu alleen nog het activeren van de pijplijn. Dat gaat als volgt:

```
ESP_LOGI(TAG, "[ 5.1 ] Start audio_pipeline");
...
audio_pipeline_run(pipeline);
```

Er is voor de MP3-speler ook nog een vrij uitgebreide eventverwerking om te reageren op verschillende gebeurtenissen (**listing 1**). De exacte eventverwerking is nu niet belangrijk, het is veel interessanter om de compilatie uit te voeren als test:

```
(base)tamhan@tamhan-thinkpad:~/esp4/esp-adf/examples/
get-started/play_mp3_control$ make build
```

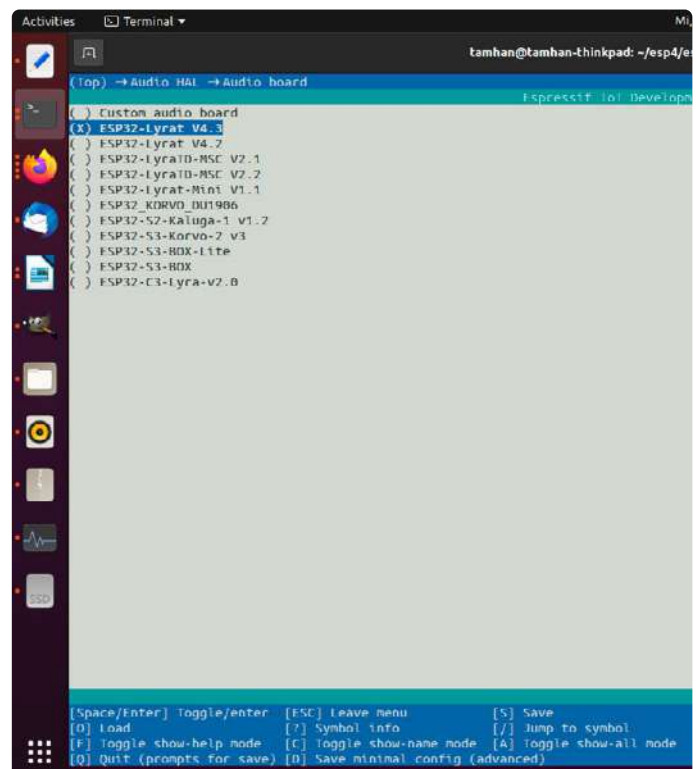
Een opmerking hierover: om de privacy van mijn klanten te beschermen, heb ik bij het schrijven van dit artikel gewerkt op mijn reislap-top. De Ubuntu-versie die daarop draait werkt met versie 3.4.3 van CMake, en daarom kon ik idf.py niet draaien. In de praktijk is idf.py natuurlijk te prefereren boven het gebruik van make. En dat zal ook verplicht zijn in toekomstige versies van IDF.

Bij een maagdelijk project is de beloning voor het werk het *menuconfig*-scherm, dat enkele extra opties biedt vergeleken met dat van een gewoon IDF-project. De belangrijkste is de optie (Top) Audio HAL Audio board, waar u de board-configuratie kunt kiezen (**figuur 5**).

Na het opslaan begint het bouwproces op dezelfde manier als voor elk ander IDF-project. Het is heel belangrijk dat de kaart twee MicroUSB-poorten heeft: de POWER-poort is om de kaart te voeden

en de UART-poort wordt alleen gebruikt voor datacommunicatie. Meestal gebruik ik gewoon twee aparte USB-kabels om mijn PC met de kaart te koppelen.

Het LyraT-board kan (in tegenstelling tot veel ander ontwikkelkaarten) niet automatisch opnieuw worden geprogrammeerd; zelfs voordat een verbinding kan worden opgebouwd, moet u de



Figuur 5: ADF breidt Menuconfig uit met enkele instellingen.

Boot-knop ingedrukt houden en dan kort op RST drukken. Dan ziet u als het goed is:

```
Serial port /dev/ttyUSB0
Connecting.....
```

Als u niet op de juiste manier op de knoppen hebt gedrukt, ziet het systeem dit als een corrupt communicatiebericht... Probeer het opnieuw:

```
A fatal error occurred: Failed to connect to ESP32:
Invalid head of packet (0x1B): Possible serial noise or
corruption.
```

Als het flashen is gelukt, kunt u een koptelefoon aansluiten en opnieuw op *Reset* drukken. Als alles goed gegaan is, moet u een audio-track kunnen horen.

Vergeet niet dat het uitvoeren van het configuratiescript *export.sh* niet voldoende is om de omgeving te parameteriseren. U moet altijd eerst zorgen dat de omgevingsvariabele *ADF_PATH* goed is ingesteld:

```
(base)tamhan@tamhan-thinkpad:~/esp4/esp-adf$ export
ADF_PATH=$PWD
(base)tamhan@tamhan-thinkpad:~/esp4/esp-adf/esp-idf$ .
./export.sh
Setting IDF_PATH to '/home/tamhan/esp4/esp-adf/esp-idf'
. . .
```

Verwerken van inputdata

Het gebruik van de ESP-ADF-pijplijn is niet alleen zinvol bij het weergeven van geluid: We kunnen net zo goed berekeningen in de pijplijn zetten. Onlangs moest ik nog voor een klant microfoon-data laten verwerken door een algoritme. De eigenlijke berekening zette ik in een pijplijnelement, waarvan ik de algemene structuur hier graag wil laten zien.

We beginnen met het initialiseren van de pijplijn, en omdat de audioinformatie binnenkomt via de I2S-bus, is een I2S-stream-element nodig in de eerste stap. Maar in dit geval bevat de configuratie nu de vlag *AUDIO_STREAM_READER*, die aangeeft dat het een input of een databron is:

```
i2s_stream_cfg_t i2s_cfg = I2S_STREAM_CFG_DEFAULT();
i2s_cfg.type = AUDIO_STREAM_READER;
#if defined CONFIG_ESP_LYRAT_MINI_V1_1_BOARD
    i2s_cfg.i2s_port = 1;
#endif
i2s_stream_reader = i2s_stream_init(&i2s_cfg);
```

Naast de I2S-stream hebben we ook een element nodig waar het algoritme in zit. Dat wordt hier geïnitieerd:

```
tams_stream_cfg_t fatfs_cfg = TAMS_STREAM_CFG_DEFAULT();
fatfs_cfg.type = AUDIO_STREAM_WRITER;
tams_stream = tams_stream_init(&fatfs_cfg);
```

De eigenlijke pijplijn wordt dan opgebouwd door strings toe te wijzen en die door te geven naar *audio_pipeline_link()*:

```
audio_pipeline_register(pipeline, i2s_stream_reader,
    "i2s");
audio_pipeline_register(pipeline, tams_stream, "tam");
audio_pipeline_link(pipeline, (const char *[]) {"i2s",
    "tam"}, 2);
```

Nu kunnen we overgaan naar de headerfile die de elementen voor het gebruik van de “Tam”-taak bevat. Het belangrijkste is de configuratie-struct, die onder meer de meta-informatie voor FreeRTOS bevat (**listing 2**).

TAMS_STREAM_CFG_DEFAULT is een macro voor het aanmaken van de structs met default parameters. We hebben hier geen ruimte om de constanten weer te geven:



Listing 2: Configuratie van de pijplijnelementen.

```
typedef struct {
    audio_stream_type_t    type;          /*!< Stream type */
    int                    buf_sz;        /*!< Audio Element Buffer size */
    int                    out_rb_size;   /*!< Size of output ring buffer */
    int                    task_stack;    /*!< Task stack size */
    int                    task_core;     /*!< Task running in core (0 or 1) */
    int                    task_prio;     /*!< Task priority (based on freeRTOS priority) */
} tams_stream_cfg_t;
```



Listing 3: Initialisatie.

```
audio_element_handle_t tams_stream_init(tams_stream_cfg_t *config)
{
    audio_element_handle_t el;
    tams_stream_t *fatfs = audio_calloc(1, sizeof(tams_stream_t));

    AUDIO_MEM_CHECK(TAG, fatfs, return NULL);

    audio_element_cfg_t cfg = DEFAULT_AUDIO_ELEMENT_CONFIG();
    cfg.open = _tams_open;
    cfg.close = _tams_close;
    cfg.process = _tams_process;
    cfg.destroy = _tams_destroy;
    cfg.task_stack = config->task_stack;
    cfg.task_prio = config->task_prio;
    cfg.task_core = config->task_core;
    cfg.out_rb_size = config->out_rb_size;
    cfg.buffer_len = config->buf_sz;
    if (cfg.buffer_len == 0) {
        cfg.buffer_len = TAMS_STREAM_BUF_SIZE;
    }
    cfg.tag = "file";
    ...
}
```

```
#define TAMS_STREAM_CFG_DEFAULT() {\
    .task_prio = TAMS_STREAM_TASK_PRIO, \
    . . .
}
```

Alleen voor `BUF_SIZE` maken we een uitzondering. De aangeleverde data is 16 bits breed, daarom hebben we de volgende constante nodig om blokken met een lengte van 1024 slots te verwerken:

```
#define TAMS_STREAM_BUF_SIZE (1024*2)
```

Nu kunnen we overgaan naar de initialisatie, zie **listing 3**.

De waarde van `cfg.buffer_len` bepaalt het aantal datawoorden dat per run wordt geleverd. Voor de rest is er eigenlijk alleen standaard code voor de administratie:

```
if (config->type == AUDIO_STREAM_WRITER) {
    cfg.write = _tams_write;
} else {
    cfg.read = _tams_read;
}
el = audio_element_init(&cfg);

AUDIO_MEM_CHECK(TAG, el, goto _tams_init_exit);
audio_element_setdata(el, fatfs);
return el;

_tams_init_exit:
audio_free(fatfs);
return NULL;
```

De eigen audio-elementen van de ontwikkelaar moeten in een pijplijn worden geïntegreerd zoals figuur 3. `audio_element_init()` heeft daar wat informatie voor nodig: naast de thread-parameters zet de routine ook een aantal callbacks op om events te signaleren.

Zo is er bijvoorbeeld de methode `tams_open`, die zorgt voor de initialisatie van het element volgens het volgende schema:

```
static esp_err_t _tams_open(audio_element_handle_t self) {
    audio_element_info_t info;
    audio_element_getinfo(self, &info);

    initTamsWorkerAlgo();

    return audio_element_setinfo(self, &info);
}
```

Omdat het huidige algoritme geen informatie in `audio_element_info_t`, nodig heeft is de implementatie beperkt tot het doorgeven van de parameterinformatie naar de pijplijn.

De events `read` en `write` zorgen voor de uitwisseling van gegevens. Onze stream wordt alleen gebruikt als een “write point” en kan niet worden gelezen, daarom genereert hij alleen een foutmelding in de logfile als iemand probeert hem uit te lezen:

```
static int _tams_read(audio_element_handle_t self,
char *buffer, int len,
TickType_t ticks_to_wait, void *context)
{
    ESP_LOGE(TAG, "CENSORED");
    return 0;
}
```

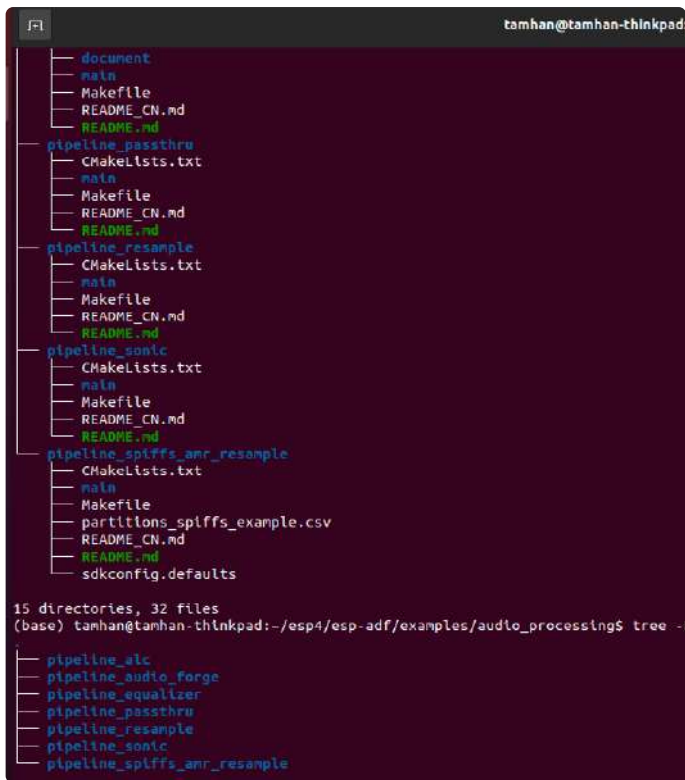
Schrijfoperaties zijn ook geïmplementeerd als “passthrough” en voeren geen echte operaties uit:

```
static int _tams_write(audio_element_handle_t self,
char *buffer, int len,
TickType_t ticks_to_wait, void *context) {
    return len;
}
```

Voor de volgende stap is een inputbuffer van het volgende formaat nodig:

```
int16_t DspBuf[4096];
```

`_tams_process()` verzorgt de eigenlijke verwerking van de data.



Figuur 6: Er zijn heel veel pijplijnen om te verkennen!

Het belangrijkste hier is de call van `audio_element_input`, die de informatie naar de werkbuffer kopieert:

```
static int _tams_process(audio_element_handle_t self,
char *in_buffer, int in_len) {
    audio_element_input(self, (char *)DspBuf, in_len);
```

Als een eenvoudig voorbeeld van dataverwerking doe ik hier een eenvoudige normalisatie op een reeks van waarden:

```
for (int i=0 ; i< in_len ; i++) {
    y_cf[i] = ((float)DspBuf[i]) / (float)32768;
}
```

Vergeet ook de administratie niet:

```
int r_size = audio_element_input(self,
    in_buffer, in_len);
int w_size = 0;
if (r_size > 0) {
    w_size = audio_element_output(self,
        in_buffer, r_size);
} else {
    w_size = r_size;
}
return w_size;
}
```

Operaties voor het vrijmaken van geheugen dat eerder was toegevoegd voor streams:

```
static esp_err_t _tams_close(audio_element_handle_t self)
{
    return ESP_OK;
}
```

```
}
static esp_err_t _tams_destroy(audio_element_handle_t
self) {
    tams_stream_t *fatfs =
    (tams_stream_t *)audio_element_getdata(self);
    audio_free(fatfs);
    return ESP_OK;
}
```

De rest van de code in de oplossing hoeft alleen nog de kern te stoppen met behulp van de volgende instructies:

```
audio_pipeline_run(pipeline);
printf("halting.\n");
for(;;);
```

De periodiek aangeroepen werkerfunctie in de pijplijn kan in feite doen wat hij wil. In dit project communiceert hij met I2C- of SPI-collega's als reactie op binnenkomende geluidsdata.

Meer pijplijnelementen

Er zijn veel meer audio-voorbeelden beschikbaar in de ESP-ADF (**figuur 6**). Ga naar `~/esp4/esp-adf/examples/audio_processing`. De algoritmen implementeren allerlei extra pijplijnoperaties.

Het voorbeeld `examples/audio_processing/pipeline_equalizer` vond ik bijzonder interessant. Dit implementeert een complete grafische equaliserfunctie en is een klasse in het ESP-ADF-raamwerk, dus heeft geen geavanceerde wiskunde nodig. Het gebruik ervan gaat op de volgende, bekende manier:

```
equalizer_cfg_t eq_cfg = DEFAULT_EQUALIZER_CONFIG();
int set_gain[] = {-13, -13, -13, -13, -13, -13, -13,
-13, -13, -13, -13, -13, -13, -13, -13, -13, -13,
-13, -13};
eq_cfg.set_gain = set_gain;
equalizer = equalizer_init(&eq_cfg);
```

Zie [4] voor meer over datavelden.

Integratie in de weergavestream gaat dan, zoals gewoonlijk, met `audio_pipeline_register()`:

```
audio_pipeline_register(pipeline, fatfs_stream_reader,
"file_read");
audio_pipeline_register(pipeline, wav_decoder, "wavdec");
audio_pipeline_register(pipeline, equalizer,
"equalizer");
audio_pipeline_register(pipeline, i2s_stream_writer,
"i2s");
```

Als een toepassing een bepaalde graad van complexiteit bereikt, schieten de aanwezige signaalverwerkingselementen tekort. Espres-sif levert dan hulp met de ESP-DSP-library, beschikbaar onder [5]:



Het is een soort raamwerk met diverse DSP-algoritmen met een grote algoritmische stabiliteit en met optimalisaties voor de verschillende chips van Espressif.

Dankzij de mappenstructuur is elk ESP32-project “direct” ingericht voor integratie van extra ESP-IDF-componenten. De installatie van DSP is dus gemakkelijk. Maak eerst een kopie van een project:

```
(base)tamhan@tamhan-thinkpad:~/esp4/esp-ADF/examples$  
cp -r audio_processing/pipeline_equalizer/ tamsdspstest1  
(base)tamhan@tamhan-thinkpad:~/esp4/esp-ADF/examples$  
cd tamsdspstest1/
```

Maak daar een map met de naam *components*. Daar komt dan de volledige versie van de library, die is te downloaden van GitHub:

```
(base)tamhan@tamhan-thinkpad:~/esp4/esp-ADF/examples/  
tamsdspstest1$ mkdir components  
(base)tamhan@tamhan-thinkpad:~/esp4/esp-ADF/examples/  
tamsdspstest1$ cd components/  
(base)tamhan@tamhan-thinkpad:~/esp4/esp-ADF/examples/  
tamsdspstest1/components$ git clone https://github.com/  
espressif/esp-dsp.git
```

Als u dan de compilatie-toolchain draait, ziet u een nieuwe optie voor het configureren van het gedrag van de DSP-library, zoals in **figuur 7**.

Samenvatting

Met de ondersteuning van ESP-ADF geeft Espressif ontwikkelaars een krachtig en flexibel raamwerk voor het vereenvoudigen van de implementatie van audiotoeepassingen. Met die hulp heb ik in mijn eigen bedrijf al vele honderden uren aan ontwikkeltijd bespaard, ik kan het zeer aanbevelen. ◀

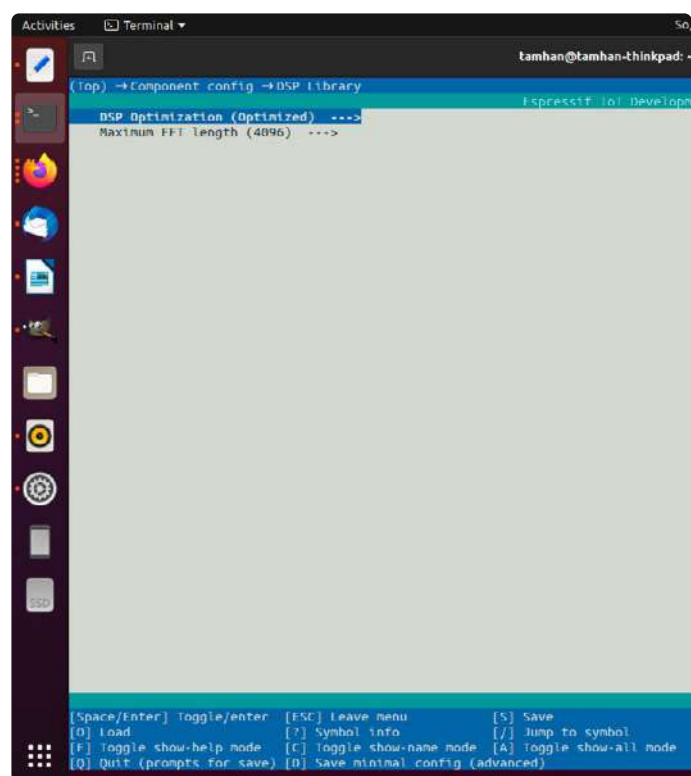
220600-03

Vragen of opmerkingen?

Als u vragen of opmerkingen hebt over dit artikel neem dan contact op met de auteur via tamhan@tamoggemon.com of met het redactieteam via redactie@elektor.com.

Over de auteur

Tam Hanna is freelance ontwikkelaar, auteur en journalist (www.instagram.com/tam.hanna). Hij ontwikkelt al meer dan 20 jaar elektronische producten, computers en software. In zijn vrije tijd ontwerpt en produceert hij 3D-geprinte oplossingen en hij is een liefhebber van goede sigaren.



Figuur 7: ESP-DSP verschijnt onder (Top) Component config DSP Library in het compilatieproces.



Gerelateerde producten

➤ **ESP32-DevKitC-32D**
www.elektor.nl/18701

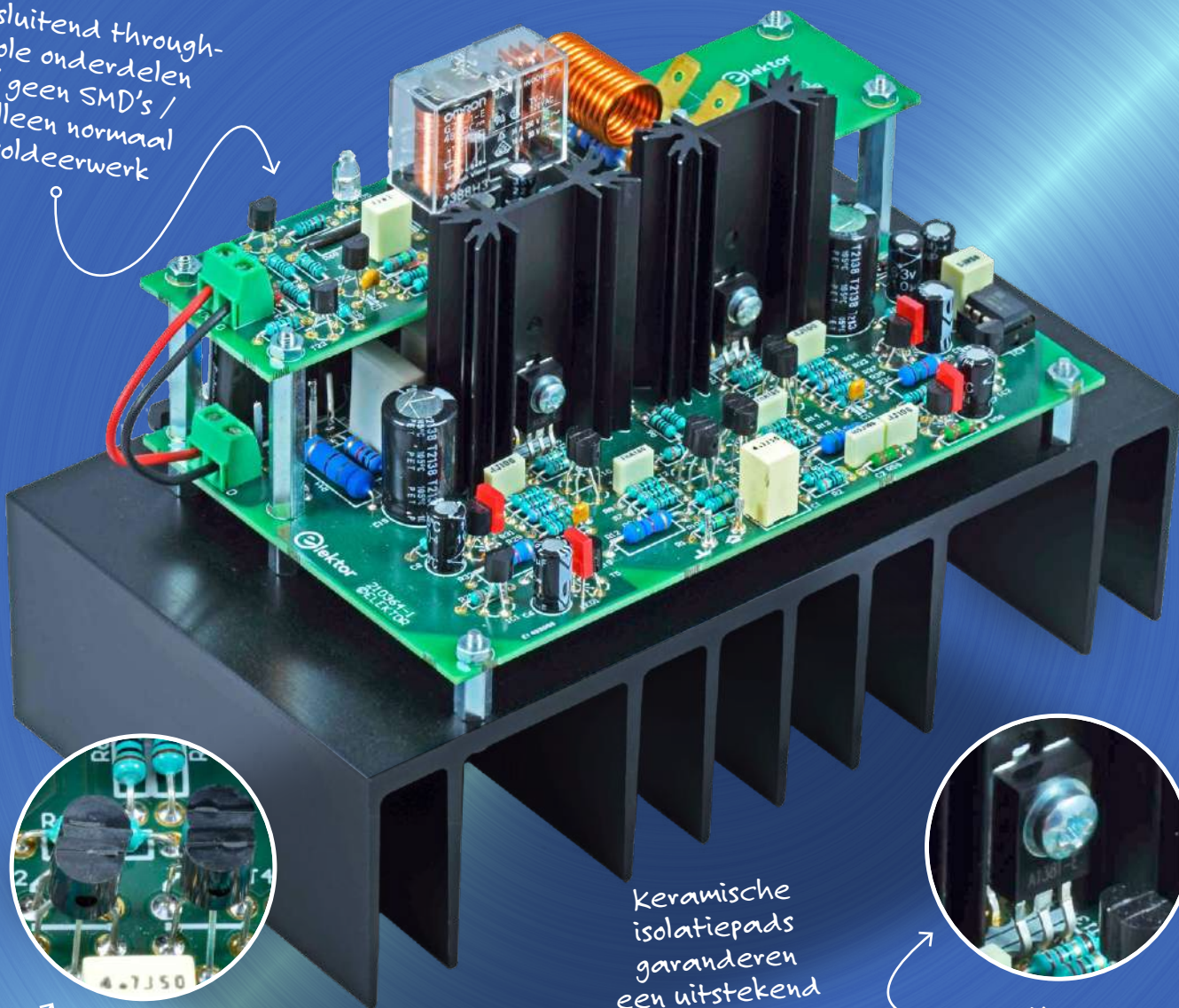


WEBLINKS

- [1] ESP-ADF Peripherals : <https://elektor.link/ESPPeripherals>
- [2] ESP-ADF Button Peripheral:
<https://elektor.link/ESPButtonPeripheral>
- [3] ESP-ADF API Reference:
<https://elektor.link/ESPAPIReference>
- [4] ESP-ADF Equalizer: <https://elektor.link/ESPEqualizer>
- [5] esp-dsp: <https://github.com/espressif/esp-dsp>
- [6] ESP32-LyraT Getting Started Guide:
<https://elektor.link/ESP32LyraT>

Elektor Fortissimo-100 eindversterker-kit

uitsluitend through-
hole onderdelen
/ geen SMD's /
alleen normaal
soldeerwerk



BC546B en
BC556B
kleinsignaal-
transistoren

TO-264-5
eindtransistoren
en TO-220 drivers
half onder de print
gemonteerd

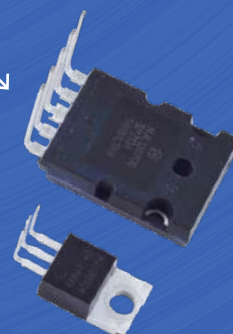


keramische
isolatiepads
garanderen
een uitstekend
thermisch contact
met het koellichaam



pootjes van de push-
pull transistoren met
thermal relief

Uitgangstransistoren
met haaks
omgebogen pootjes





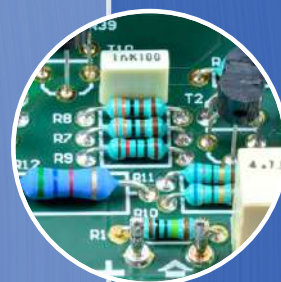
Het ontwerp

- geheel analoge klasse AB-versterker
- 100% symmetrisch
- extreem geringe vervorming en ruis
- ThermalTrack™ eindtransistoren maken ruststroominstelling in de uitgangstrap onnodig
- symmetrische-bootstrap uitgangstrap voor maximaal uitgangsspanningsbereik
- geregelde 40V-voeding nodig (bijvoorbeeld SMPS800RE)
- belastingimpedantie minimaal 3 Ω
- stevige en compacte print-op-koellichaam constructie
- mechanisch werk is beperkt tot 12 M3-gaten die door het koellichaam worden geboord
- +40 V voedingsspanningsdetectie
- uitgangs-DC-beveiliging



De onderdelen

- Elektor-kit = printen + alle through-hole componenten + koellichamen
- ThermalTrack™ transistoren met geïntegreerde insteldioden
- MJE15023- en MJE15033-drivers
- KSC3503- en KSA1381-transistoren voor hoge lineariteit van de push-pull versterkertrap
- OPA177 (DIP-8) gestuurde DC-regellus
- 0.6 W 1% metaalfilmweerstanden voor alle kleinsignaal-schakelingen
- duurzame hoge-temperatuur elektrolytische condensatoren (lage ESR) voor bootstrapping en voedingsontkoppeling



De specificaties

Ingangsgevoeligheid	1,076 V (94 W/8 Ω, THD = 0,1 %, B = 22 kHz)
Ingangsimpedantie	10 kΩ
Vermogen (sinus)	94 W (8 Ω, THD = 0,1 %) 181 W (4 Ω, THD = 0,1 %)
Bandbreedte	3,3 Hz – 237 kHz (–3 dB, 1 W/8 Ω)
Open-lus bandbreedte	≈ 20 kHz
Open-lus versterking	≈ 140000 (8 Ω load)
Stijgsnelheid	45 V/μs
Signaal/ruis-verhouding	103 dB (B = 22 Hz – 22 kHz linear)
Harmonische vervorming + ruis	0,0008 % (1 kHz, 50 W, 8 Ω, B = 80 kHz) 0,002 % (20 kHz, 50 W, 8 Ω, B = 80 kHz) 0,0042 % (20 kHz, 100 W, 4 Ω, B = 80 kHz)
Intermodulatievervorming (50 Hz : 7 kHz = 4 : 1)	0,0015 % (50 W, 8 Ω) 0,0041 % (100W, 4 Ω)

Meer informatie
www.elektor.nl/20273



Lichtgestuurde geluidseffecten

Spanningsgestuurd
24 dB/oct synthesizerfilter (VCF) met LDR's

Raymond Schouten (Nederland)

Dit artikel beschrijft een eenvoudige afstembare filterschakeling waarvoor geen speciale chips nodig zijn, alleen twee op-amps die kunnen werken op een enkele 5-V voeding. Ondanks zijn eenvoud biedt het filter een zeer lage vervorming ($\text{THD} < 0,01\%$), goede groot-signaalverwerking en een groot dynamisch bereik ($> 90 \text{ dB}$).

Het hieronder gepresenteerde 24 dB/oct audio laagdoorlaatfilter is gebaseerd op vier lichtafhankelijke weerstanden, beter bekend als LDR's, die worden aangestuurd door een enkele LED. De helderheid van de LED regelt de afsnijfrequentie van het filter van ongeveer 20 Hz tot 20 kHz. Damping (d.w.z. de Q-factor, ook bekend als resonantie) kan worden ingesteld tot volledige oscillatie, wat aantrekkelijk is voor toepassin-

gen in muzieksynthesizers. De schakeling is eenvoudig te bouwen, en de kosten van de componenten bedragen ongeveer € 4.

Dit filter is ontworpen met het oog op muzieksynthesizers, maar de hoge signaalkwaliteit zou ook andere audio-toepassingen mogelijk kunnen maken. De schakeling werd eerst gesimuleerd voordat een prototype op een breadboard

werd gebouwd om de prestaties te meten. Geluidsvoorbeelden en een demonstratievideo zijn beschikbaar op de Elektor Labs projectpagina [1].

Twee versies

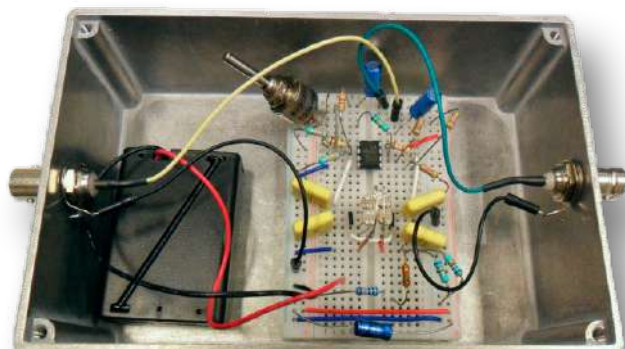
Er zijn twee versies van het filter beschikbaar, waarvan de eenvoudigste werkt op een symmetrische dubbele voeding. Door toevoeging van enkele weerstanden en condensatoren kan het worden veranderd in een versie die werkt op een enkele voeding. Deze versie werd gebouwd en gemeten (**figuur 1**).

De schakeling (dubbele voeding)

We zullen de schakeling van het filter met symmetrische voeding uitleggen, omdat die minder componenten gebruikt, zie **figuur 2**. De single-supply versie (**figuur 3**) werkt precies hetzelfde.

Het hart van het filter is een 6 dB/octaaf RC laagdoorlaatfilter waarbij de vaste weerstand R is vervangen door een LDR. Vier van deze filters zijn in serie gezet om een helling van 24 dB/octaaf te verkrijgen. Door de weerstand van de vier LDR's tegelijkertijd door een enkele LED te regelen, ontstaat een filter met een afsnijfrequentie die over een groot bereik ($> 1 : 1000$) kan worden ingesteld.

Cascadering van vier identieke RC-schakelingen zonder buffers lijkt misschien een recept voor een ramp, met als resultaat een zeer slecht filter met slappe overgang bij



Figuur 1: De single-supply versie van het filter werd gebouwd op een breadboard en in een metalen doosje geplaatst om de prestaties te controleren terwijl het gevoed werd door drie AAA-batterijen.

de afsnijfrequentie, maar aangezien het beroemde Moog-ladderfilter ook op dit principe is gebaseerd, kan het niet helemaal slecht zijn. Robert Moog loste het probleem op een slimme manier op, een oplossing die ook in onze schakeling wordt gebruikt. Het is gebaseerd op negatieve terugkoppeling die pas bij de afsnijfrequentie overgaat in positieve terugkoppeling. Het kader *Filter Q controle uitgelegd* gaat hier dieper op in.

Op-amp U1a buffert de ingang en voegt het feedbacksignaal toe via R9 en P1. Op-amp U1b is een hoogohmige buffer voor de LDR/C-cascade en voegt enige versterking toe ter compensatie van het verlies bij de afsnijfrequentie.

P1 regelt de hoeveelheid totale negatieve terugkoppeling. Het filter gaat van gedempte respons naar volledige oscillatie door P1 te regelen van 100 kΩ naar 0 Ω. Vervanging van P1 door nog een LED-LDR combinatie maakt een elektronisch geregelde Q-factor mogelijk. Zie het kader getiteld *Simulatie resultaten filterrespons* voor de frequentieafstemming en de Q-regelkrommen.

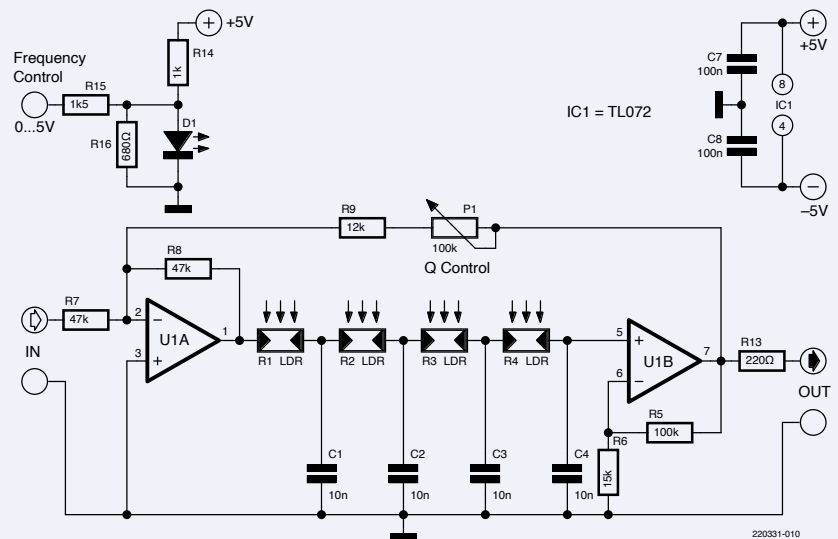
Een gedetailleerde uitleg over de keuze van LED D1 en het principe achter het

weerstandscircuit is te vinden in het kader *De LED-LDR combinatie*.

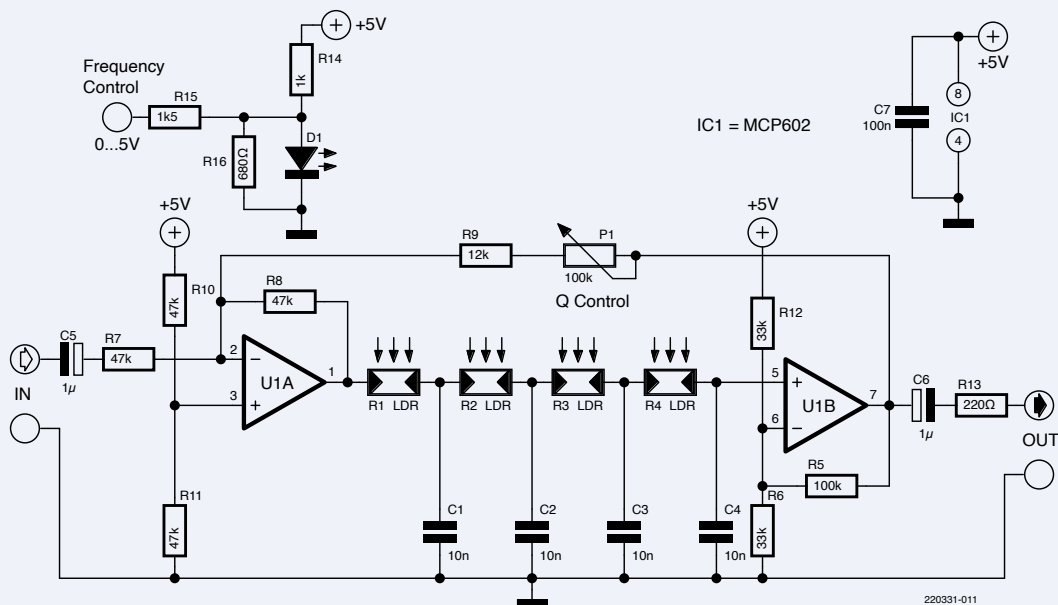
Voordelen van het concept

De meeste analoge afstembare filters gebruiken complexere schakelingen die alleen werken bij kleine signaalniveaus ($< 100 \text{ mV}_{PP}$) om de vervorming op een aanvaardbaar niveau te houden. Voorbeelden hiervan zijn filters gebaseerd op tuned-bias transistors of diodes, discreet of geïntegreerd in speciale IC's (OTA's zoals de beroemde LM13700). Dit op LDR's

gebaseerde ontwerp daarentegen kan signalen van meerdere volts aan terwijl de vervorming zeer laag blijft. Eigenlijk wordt de spanningszwaai voornamelijk beperkt door de opamps en de voedingsspanning, want de meeste LDR's kunnen tot 100 V of meer aan. Je zou dit filter zelfs met buizen kunnen bouwen! Aangezien het ruisniveau van de schakeling vrij laag is -die wordt voornamelijk bepaald door de opamps en de equivalente weerstandsruijs van de LDR's- kan een groot dynamisch bereik worden gerealiseerd.



Figuur 2: Schema van de dual-supply versie van het filter.



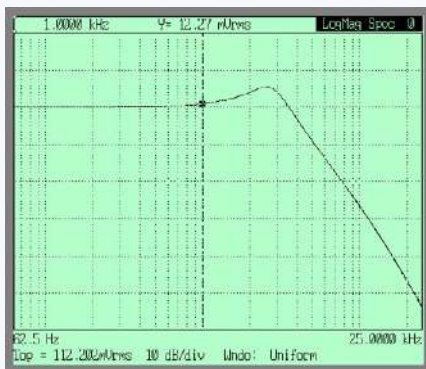
Figuur 3: Door toevoeging van enkele weerstanden en condensatoren kan het filter met dubbele voeding worden veranderd in een filter met één voeding.



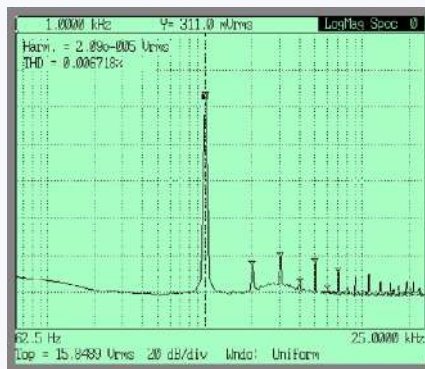
Vervormingsmetingen

Voor deze metingen werd een Stanford Research SR770 FFT spectrum analyzer met ingebouwde signaalbron gebruikt. Om het oppikken van elektrische en optische storingen te voorkomen, werd de schakeling geplaatst in een gesloten metalen doos waarin ook de batterijvoeding was ondergebracht (zie figuur 1). De schakeling werkte hier op een enkele voeding van 4,5 V (3x AAA).

Eerst werd het instrument ingesteld om de filterresponscurve te meten. Vervolgens werd het ingesteld om een sinus van 1 kHz te genereren en het uitgangsspectrum van het filter te meten tot 25 kHz. Daaruit werd de totale harmonische vervorming (THD) berekend voor de eerste zes harmonischen. Het filter werd ingesteld op een afsnijfrequentie van 4 kHz. De reden voor deze keuze is dat er positieve terugkoppeling is bij de afsnijfrequentie en dat verhoogt de vervorming. Dit is dus een worst-case scenario voor de tweede en derde harmonische vervorming (meestal de dominante). Men zou kunnen aanvoeren dat het een deel van de hogere harmonischen onderdrukt, maar een controle bij $f_c = 13$ kHz liet vrijwel hetzelfde harmonische patroon zien.



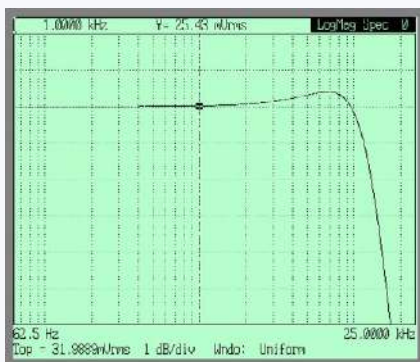
Gemeten filterrespons.



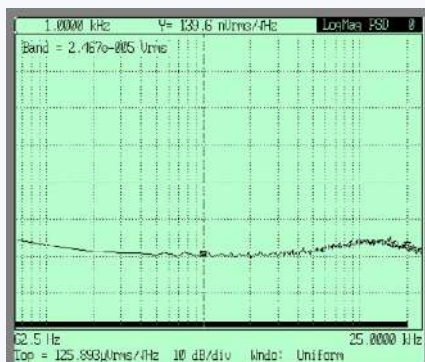
Gemeten vervorming.

Ruismeting en dynamisch bereik

Het uitgangsspectrum is gemeten met uitgeschakelde signaalbron en geeft in totaal 25 μ Vrms in de audioband tot 20 kHz. In combinatie met de maximale signaalverwerking van 4 Vpp (1,4 Vrms) leidt dit tot een dynamisch bereik van 95 dB. Dit zou verbeterd kunnen worden door hogere voedingsspanningen te gebruiken.



Gemeten filterresponscurve (1 dB/div). Het filter was ingesteld op -3 dB bij 13 kHz.



Het gemeten spectrum aan de uitgang van het filter.

Een ander voordeel van de schakeling is de extreem lage (afwezige?) stuurspanningslekage in het audiopad vanwege de optische koppeling.

Met geschikte op-amps werkt hij op een enkele lage voeding (3 V tot 5 V), waardoor hij compatibel is met microproceessor-toepassingen zoals Arduino.

Met een spanningsregeling op de LED (zie kader *De LED-LDR Combinatie*) kan een ruwe, grofweg exponentiële regelkromme worden verkregen die nodig is voor synthesizertoepassingen. Voor een preciezere (maar lineaire) regeling kan een stroombron of PWM-sigitaal voor de LED worden toegepast.

Beperkingen van het concept

Om eerlijk te zijn heeft het hier gepresenteerde filter ook enkele nadelen in vergelijking met andere concepten. De instelling van de afsnijfrequentie is niet erg precies. Het is voldoende voor een muzieksynthesizer, maar voor nauwkeuriger toepassingen kan een kalibratietabel of een extra terugkoppelingsschakeling nodig zijn.

Bovendien wordt de regelsnelheid van de afsnijfrequentie beperkt door de relatief trage LDR's. Bij het verhogen van de filterfrequentie blijft de gemeten respons in de orde van een milliseconde, maar het verlagen kan tot 50 ms duren en zelfs tot 500 ms voor zeer lage afsnijfrequenties. Korte 'attack'-tijden zijn dus mogelijk voor het openen van het filter aan het begin van een toon. In de muziek is de aanslag tijd van een geluid meestal korter dan de uitsterftijd, waardoor dit filter geschikt is voor muzikale toepassingen.

Suggesties en opmerkingen

Het dynamische bereik kan verder worden vergroot door hogere voedingsspanningen te gebruiken. Zoals gezegd kunnen LDR's doorgaans signalen tot 100 V verwerken.

Om een nauwkeurigere en stabielere frequentieregeling te krijgen, zou je vijf LDR's kunnen gebruiken in plaats van vier en er eentje in de feedbacklus opnemen van een op-amp die de LED aanstuurt. Omdat het filter is ontworpen met toepassingen voor muzieksynthesizers in gedachten, is de

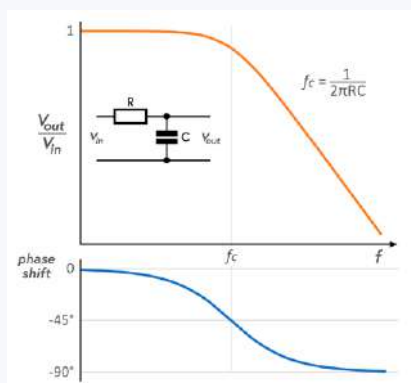
versterking sterk afhankelijk gemaakt van de Q-waarde om grote verhogingen van het uitgangssignaal te voorkomen. Als alternatief zou het ingangssignaal kunnen worden toegepast op de niet-inverterende ingang van U1a in plaats van op de inverterende ingang. Dit zou de invloed van de Q-waarde op de filterversterking beperken en het filter zou een niet-inverterende overdrachtsfunctie krijgen. **K**

220331-03

Vragen of opmerkingen?

Heeft u technische vragen of opmerkingen over dit artikel? E-mail de auteur op rs.elc.projects@gmail.com of Elektor op editor@elektor.com.

Filter Q-control uitgelegd



Een RC-laagdoorlaatfilter is niet alleen een filter, maar ook een frequentie-afhankelijke faseverschuiver, beginnend bij 0°, oplopend tot -45° bij de afsnijfrequentie en daarna tot -90°. Met vier van deze RC-filters in serie wordt de faseverschuiving bij de afsnijfrequentie $4 \times -45 = -180^\circ$. Dit betekent dat het signaal bij die frequentie van teken verandert. Wanneer we dus negatieve terugkoppeling toepassen van de uitgang naar de ingang van het filter, is het resultaat positieve terugkoppeling bij de afsnijfrequentie, waardoor de versterking bij die frequentie piekt. Dit geeft het filter de scherpe overgang bij de afsnijfrequentie. Door de hoeveelheid negatieve terugkoppeling aan te passen, kan de versterking worden gevarieerd. Bij een bepaalde maximale terugkoppeling kan het een (sinus)oscillator worden die regelbaar is van 20 Hz tot 20 kHz.

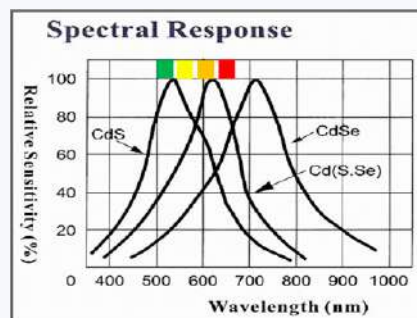
De combinatie LED-LDR

Bouw

De LED hangt zo'n 5 tot 7 mm boven de vier LDR's. De mechanische precisie-eisen zijn bescheiden, metingen tonen geen significant verschil in respons als de LED 1 tot 2 mm uit de as wordt geduwd. Zorg voor een goede afscherming van de schakeling tegen omgevingslicht, want dat zal ook het filter moduleren!

Keuze van de LED-kleur

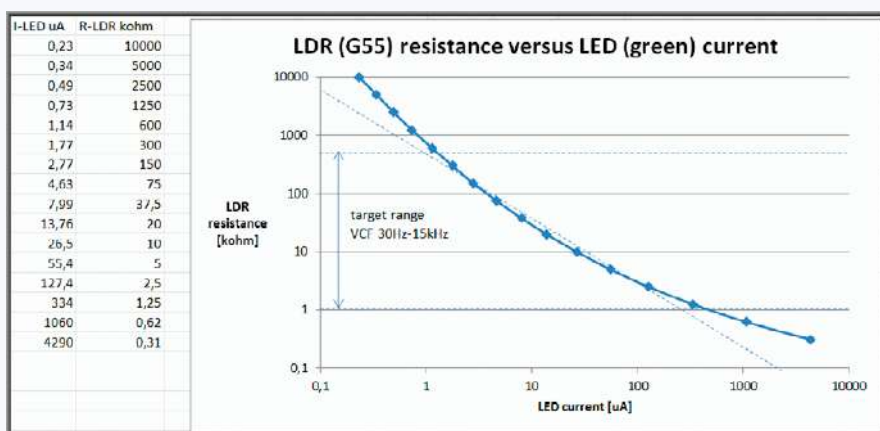
Ik heb LDR's op basis van CdS gebruikt, dan werkt groen of geel licht het beste, zoals blijkt uit dit spectrale responsdiagram.



Bron: JCLH, datasheet van LDR GL5537-1

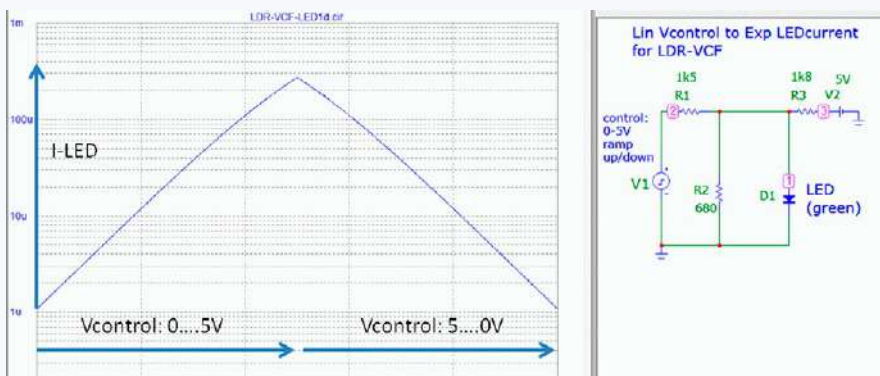
Gevoeligheid

Uit de metingen bleek dat de weerstand van de LDR bijna een lineaire functie is van de LED-stroom in het weerstandsbereik van 500 kΩ tot 1 kΩ. In totaal kon de LDR-weerstand worden geregeld van 10 MΩ tot 300 Ω, zoals blijkt uit deze gemeten curve.



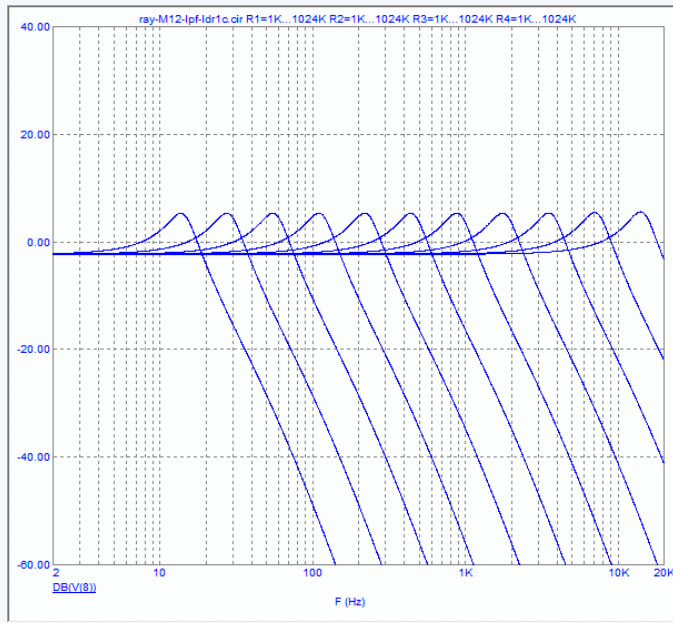
Hier werd een zeer heldere groene LED gebruikt (OVLBG4C7), 0,5 μA produceerde genoeg licht om de LDR-weerstand te verlagen.

Hieronder wordt een zeer eenvoudige maar onnauwkeurige manier getoond om een exponentiële regelkromme te benaderen zonder gebruik te maken van extra schakelingen. Het principe is als volgt. Wanneer een diode (hier de LED) met een spanning wordt aangestuurd, is de resulterende stroom een exponentiële functie van die spanning. Een spanningsbron rechtstreeks aansluiten op een LED is geen goed idee, maar hier gebruiken we voldoende lage weerstandswaarden om dit te benaderen. De simulatie hieronder laat zien dat wanneer we de stuurspanning lineair sweepen van 0 tot 5 V, we een exponentiële toename van de LED-stroom krijgen van 1 μA tot 300 μA. In de praktijk moeten de weerstandswaarden misschien een beetje aangepast worden aan de LED. Zoals gezegd is dit nogal onnauwkeurig en temperatuurafhankelijk, maar de eenvoudigste manier om exponentiële regeling voor synthesesizetoepassingen te benaderen. De geluidsvoorbeelden zijn gemaakt met deze regeling.

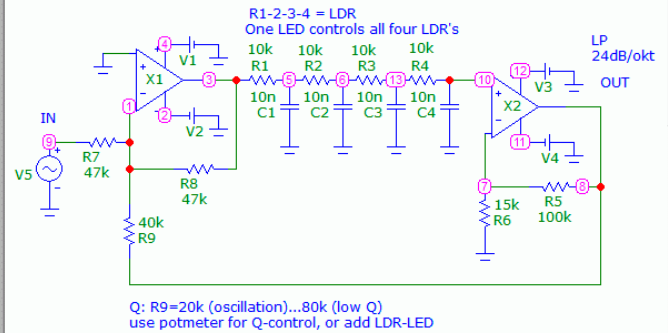




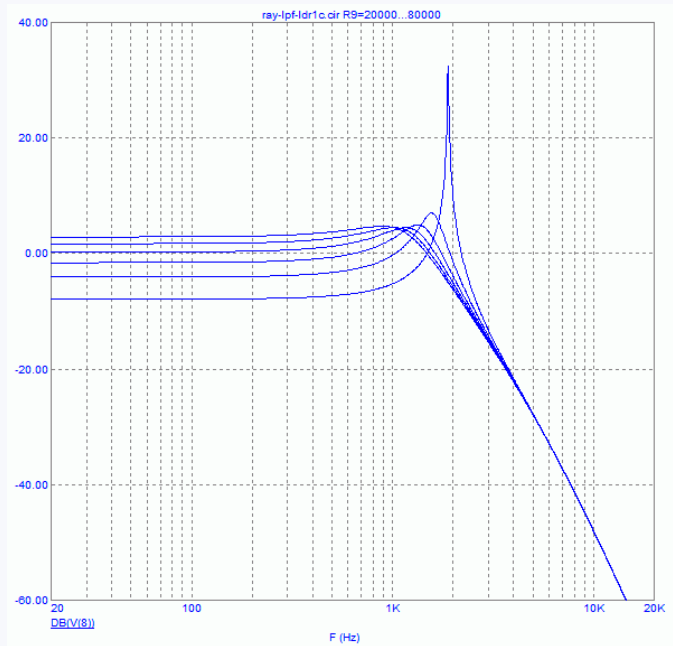
Simulatieresultaten van de filterrespons



LDR based VC LPF 24dB/oct 20Hz-20kHz

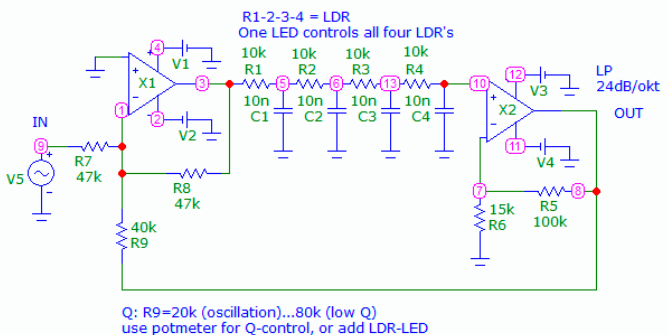


In deze simulatie wordt de afsnijfrequentie van het filter gesweept van 20 Hz tot 20 kHz terwijl de Q-regelaar op een constante waarde wordt gehouden.



LDR based VC LPF 24dB/oct 20Hz-20kHz

www.RS-ELC.nl



Hier gaat de Q-regelaar van het filter (R9) van 80 kΩ naar 20 kΩ terwijl de de afsnijfrequentie op een constante waarde wordt gehouden.



Verwante producten

- Joy-IT ScopeMega50 USB Oscilloscoop (SKU 18277) www.elektor.nl/18277
- MIDI Input Board - Kale PCB (SKU 18382) www.elektor.nl/18382
- De LTspice XVII Simulator (SKU 19741) www.elektor.nl/19741

WEBLINKS

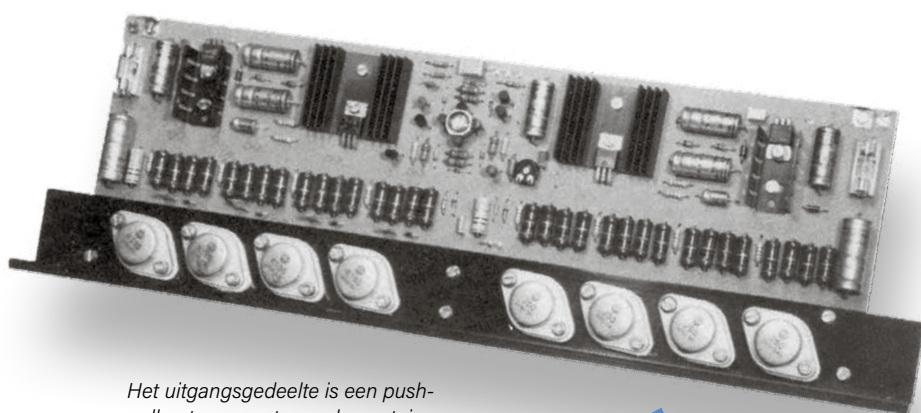
- [1] This project at Elektor Labs:
www.elektormagazine.com/labs/voltage-controlled-24dboct-synthesizer-filter-using-ldrs

De kilo-watter van Elektuur

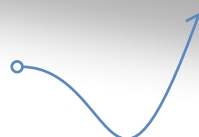
meer volume gaat niet!

Larry Kossek (Elektor)

Audioversterkers maken vanaf het begin deel uit van het DNA van Elektuur. Thomas Scherer bouwde ze allemaal, van 40 W tot 200 W, voordat hij in 1986 probeerde door te breken met zijn eigen ontwerp, dat een hoogwaardige sound kon leveren bij een destijds ongelooflijk uitgangsvermogen van 1000 W.



Het uitgangsgedeelte is een push-pull ontwerp met complementaire N- en P-kanal horizontale powerMOSFET's, geselecteerd op een goede transiëntresponsie en lineariteit bij alle mogelijke aansturingsniveaus.

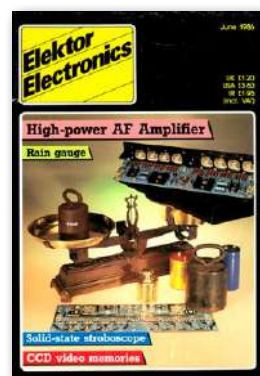


Segway Cloners Club

Een duik in de Elektuur-archieven roept herinneringen op en brengt verrassende associaties aan het licht. Thomas Scherer, de man die in 1986 deze gigantische audioversterker ontwierp, werd later een fan en promotor van de Elektor Wheelie. Wat hebben een audioversterker van 1000 W en een Segway-kloon gemeen? Een onderdeel met drie pootjes, een metalen behuizing en bestendigheid tegen grote stromen. Inderdaad – de MOSFET-vermogenstransistor, het werkpaard dat een revolutie teweegbracht in de moderne vermogens-elektronica. In de jaren tachtig waren deze vermogenscomponenten nog niet vanzelfsprekend in Elektuur-projecten. En Thomas had al een voorliefde voor sterkstroomschakelingen die de zekeringen in het lab op de proef stelden.

“Wer Großes will, muß sich zusammenraffen”

Vrij vertaald: *als je het groots wilt aanpakken, moet je jezelf vermennen*. Natuurlijk maakten ‘gewone’ eindversterkerprojecten vanaf het begin deel uit van het DNA van Elektuur. Versterkers volgden elkaar in snel tempo op en Thomas bouwde ze allemaal, van 40 tot meer dan 200 W. In 1986 was hij lid van de Elektuur-redactie en droomde hij van een audioversterker die genoeg power had om high-end audio te leveren bij het toen nog onvoorstelbare vermogen van 1000 W. Uiteindelijk ontwierp hij er een, thuis, in zijn vrije tijd en zonder dat zijn collega's het wisten. Hij ging laat naar bed en in veel van zijn dromen kwamen complementaire paartjes uitgangstransistoren voor. “Ik ontwierp de analoge schakeling met behulp van de een gezonde



Specificaties

- › Uitgangsvermogen: 2 × 250 W (8 Ω), 2 × 500 W (4 Ω) of 1 × 1000 W (8 Ω)
- › Bandbreedte: 8 Hz tot 100 kHz
- › Vervorming: 0,1% (@ 1 kW); 0,01% (@ 600 W)
- › Dempingsfactor: > 100
- › Inschakelvertraging, ventilatorregeling, spanningsbewaking
- › Ingangsniveau: 0,775 VRMS voor maximaal uitgangsvermogen
- › Gewicht: dat wil je niet weten!

portie hoofdrekenen en een beetje hulp van een programmeerbare Sharp-rekenmachine. [...] Mijn voorstel werd ter beoordeling door Elektuur geaccepteerd."

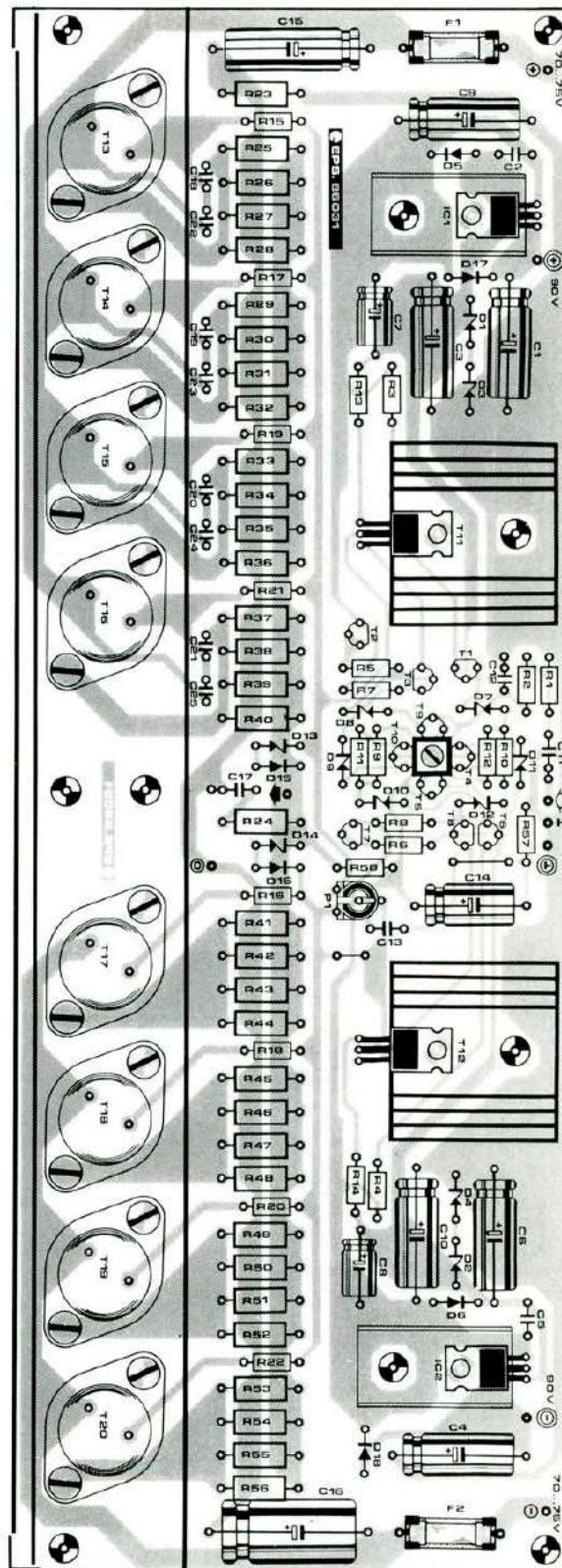
Exorbitant dure MOSFET's

Al snel moest Thomas van het papieren schema een tot het uiterste beproefd en reproduceerbaar project proberen te maken: de "Kilo-watter", en zonder al te veel onderdelen op te blazen, vooral die exorbitant dure Hitachi MOSFET's van het type 2SK135/2SK175 (T13...T16) en 2SJ50/2SJ55 (T17...T20).

"Met voedingsspanningen tot ± 90 V kon ik piekspanningen van meer dan 125 V over de luidsprekeraansluitingen verwachten. Tot dan was ik altijd aan de veilige kant gebleven met 48 V. [...] Het was essentieel de luidspreker vertraagd in te schakelen en voor de voedings-transformator een 'soft start' te hebben voor de voedings-transformator' in te bouwen. [...] Voor meting van de vervorming en enkele andere zaken was mijn referentie de schrikbarende dure Brüel&Kjær spectrumanalyzer die het audiolab van Elektuur sierde waartoe ik nu toegang had."

De één-kilowatt-versterker (door cynici ook wel *Killerwatts* genoemd) wekte grote belangstelling en weerklonk jarenlang [1][2][3]. Af en toe duikt er, meer dan 35 jaar na de publicatie, een exemplaar van de kilowattversterker op internet te koop op. Veel mensen hebben met de kilo-watter opgescheept en mogelijk zelfs 'een zekering opgeblazen' met nummers die op oorlogsterkte door dat beest van een versterker de omgeving werden ingeslingerd.▶

220234-03



Beveiligingen zijn bedoeld om desastreuze knallen uit de luidspreker(s) en opgeblazen netzekeringen bij het inschakelen van de versterker te voorkomen.

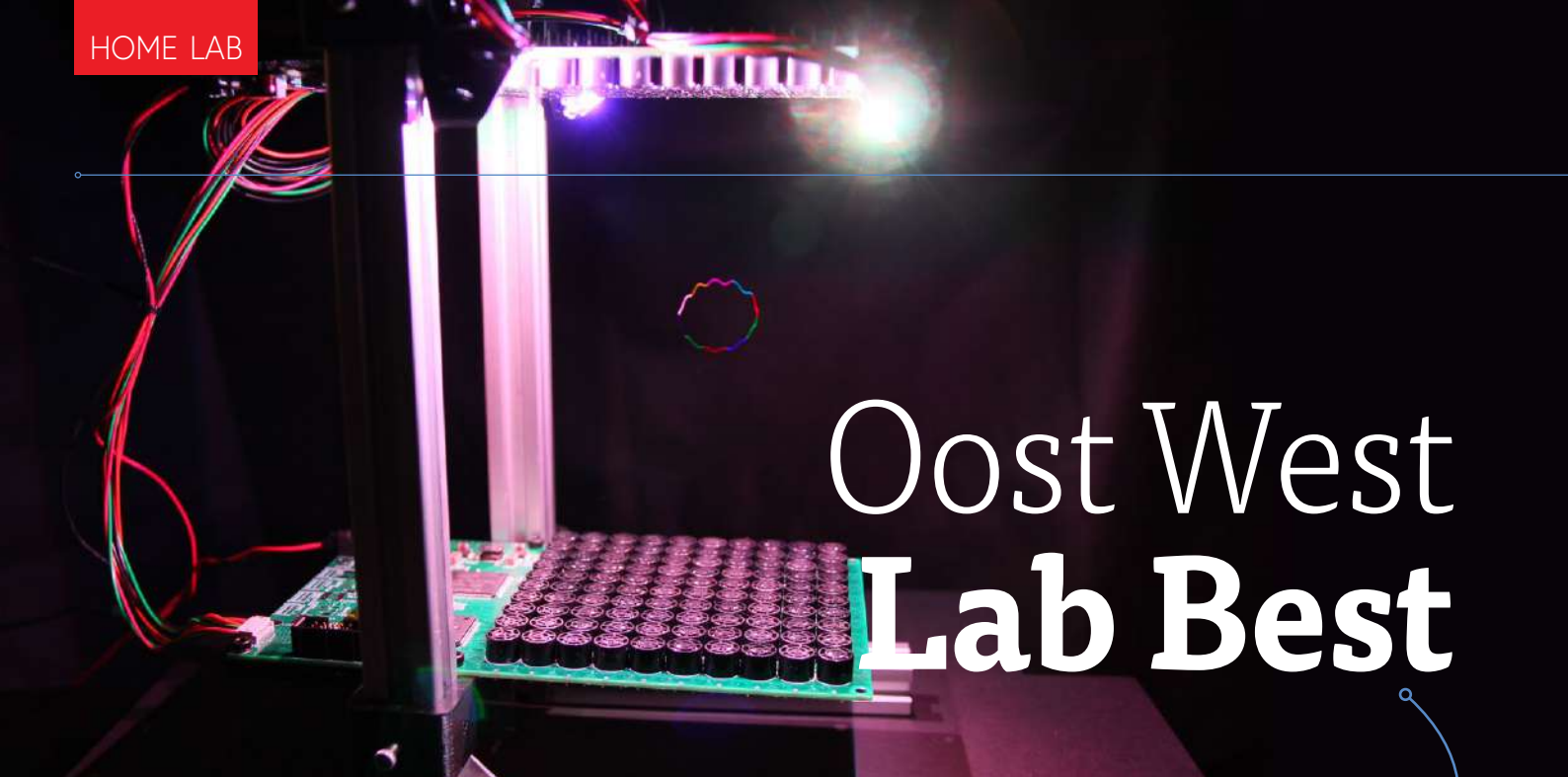


Gerelateerde producten

- **Elektor Archive 1974-2021 (USB Stick) (SKU 20071)**
www.elektor.nl/20071

WEB LINKS

- [1] T. Scherer, "Kilo-watter (1)", Elektor mei 1986: <http://www.elektormagazine.nl/magazine/elektor-198605/44002>
- [2] T. Scherer, "Kilo-watter (2)", Elektor juni 1986: <http://www.elektormagazine.nl/magazine/elektor-198606/44020>
- [3] T. Scherer, "Elektor High-Power AF Amplifier" (Retronics installment), Elektor 5/2016: <https://www.elektormagazine.com/magazine/elektor-201605/28977>



Oost West Lab Best

een volumetrisch display in Canada

Dan Foisy (Canada) en Eric Bogers (Elektor)

Met behulp van een fasegestuurd array van ultrasone transducers kan een klein piepschuim bolletje niet alleen aan het zweven worden gebracht, maar ook met grote snelheid heen en weer worden bewogen. Dit biedt met gesynchroniseerde LED-verlichting en dankzij de traagheid van onze ogen (persistence of vision, POV) de mogelijkheid een 'echt' driedimensionaal display te construeren.

Het idee om een klein piepschuim bolletje op een 'kussen' van ultrasone geluidsgolven te laten zweven, is niet werkelijk nieuw. En niet alleen dat: deze techniek is inmiddels voor de doorsnee hobbyist beschikbaar – zie het artikel "Zweven op geluidsgolven" van de hand van Elektor-redacteur Luc Lemmens, waarin hij de Maker-fabs Acoustic Levitation Kit bespreekt [1].

De inspiratie voor een project kan overal vandaan komen; in het geval van het Volumetric Display using an Acoustically Trapped Particle (dat is een hele mondvol, vandaar de afkorting VDATP) was dat een artikel van onderzoekers aan de University of Sussex [2] waarin een methode werd beschreven om een in geluidsgolven 'gevangen' deeltje voor displaydoeleinden te gebruiken. Toen Dan Foisy uit Toronto (Canada) de bijbehorende video [3] zag, was hij verkocht; gedreven door de bij hobby-elektronici maar al te bekende wens "Dat wil ik ook..." ging hij in zijn hobbylab aan de slag. Laten we Dan aan het woord.

"Met behulp van een fasegestuurd array van ultrasone transducers is het mogelijk een klein (1 mm) piepschuim balletje te laten zweven en snel heen en weer te doen

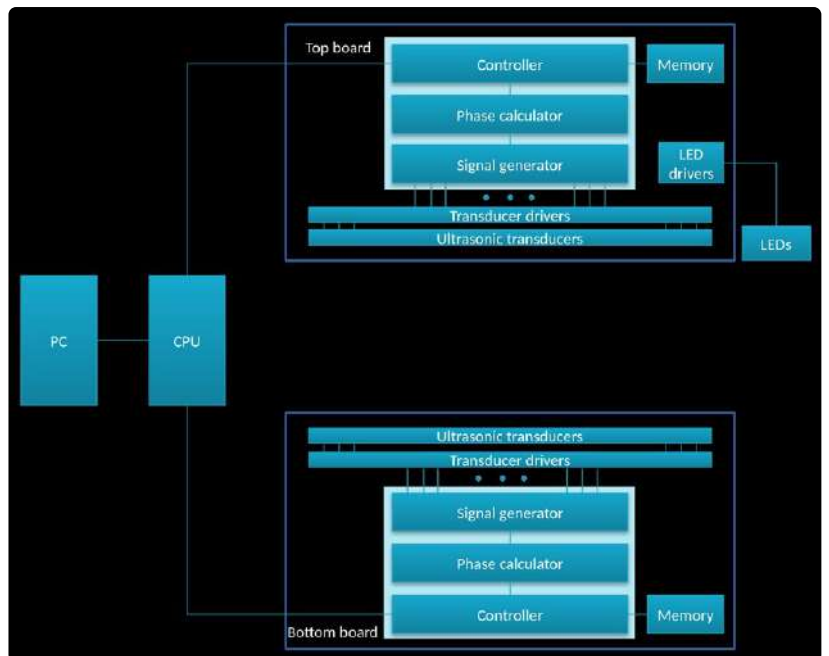
bewegen (meer dan 1 m/s). Door dan gebruik te maken van de traagheid van ons oog, kan ik met het balletje figuren in de lucht 'tekenen' – in een volume van ongeveer 100 x 100 x 140 mm. Met behulp van RGB-LED's wordt het balletje op specifieke momenten gedurende zijn traject belicht, zodat meerkleurige afbeeldingen ontstaan."

Zo'n fasegestuurd array is niet werkelijk nieuw: in 'omgekeerde' richting wordt het principe al lang gebruikt in bijvoorbeeld de radio-astronomie, waar de signalen van vele kleine antennes worden gecombineerd om een zeer grote radiotelescoop (zo groot dat die in de praktijk nooit geconstrueerd zou kunnen worden) te simuleren. Een fraai voorbeeld is LOFAR [4][5], een van oorsprong Nederlands project dat inmiddels over heel Europa is uitgebreid en uit ongeveer 20.000 afzonderlijke (kleine) antennes bestaat. Die vormen samen een fasegestuurd array met een ongeëvenaard oplossend vermogen van 0,2 boogseconden.

"Om meer te doen dan het balletje alleen maar te laten zweven – dus om het te kunnen besturen – is heel



Figuur 1. Twee array's van elk 100 ultrasone transducers worden gebruikt om een piepschuim balletje te laten zweven.



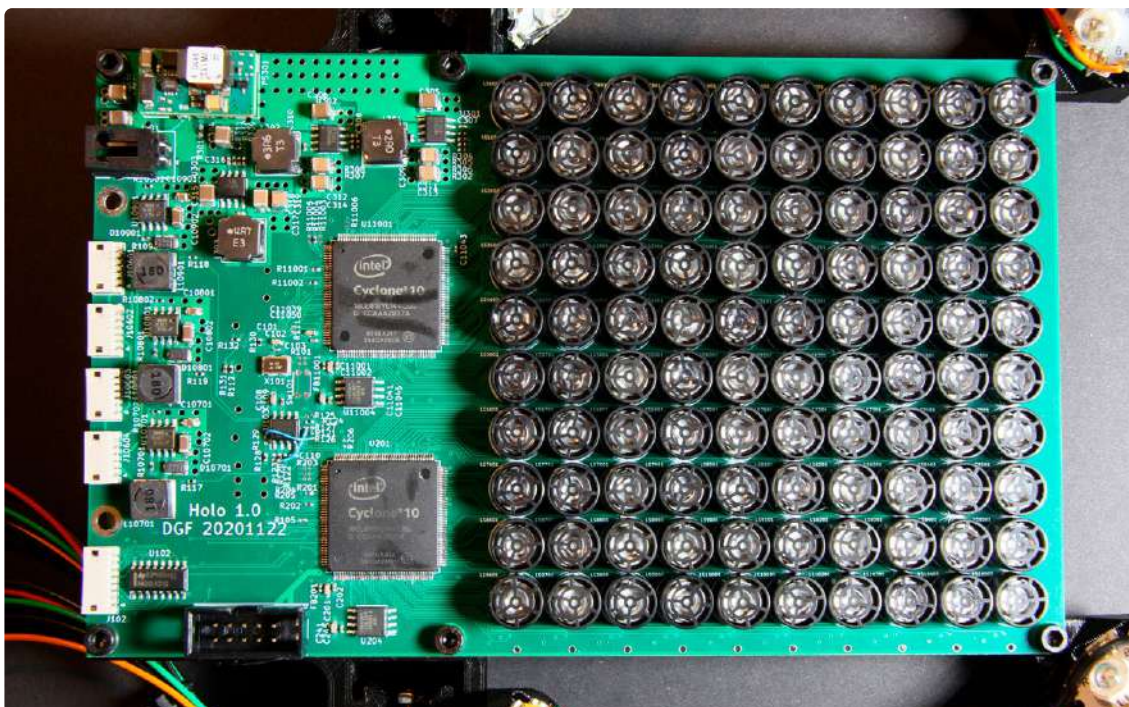
wat rekenkracht nodig. Het VDTP zoals ik het heb opgebouwd telt naast de centrale CPU vier FPGA's die een positie omzetten naar de fasegestuurde signalen voor de ultrasone transducers. De FPGA's kunnen de weergegeven afbeeldingen overigens ook roteren, verschuiven en schalen. Zelfs animaties behoren tot de mogelijkheden – ik heb er een gemaakt van een vlinder die met zijn vleugels klappt [6]."

A propos transducers: in dit project worden 200 exemplaren gebruikt op twee printplaten. Figuur 1 geeft

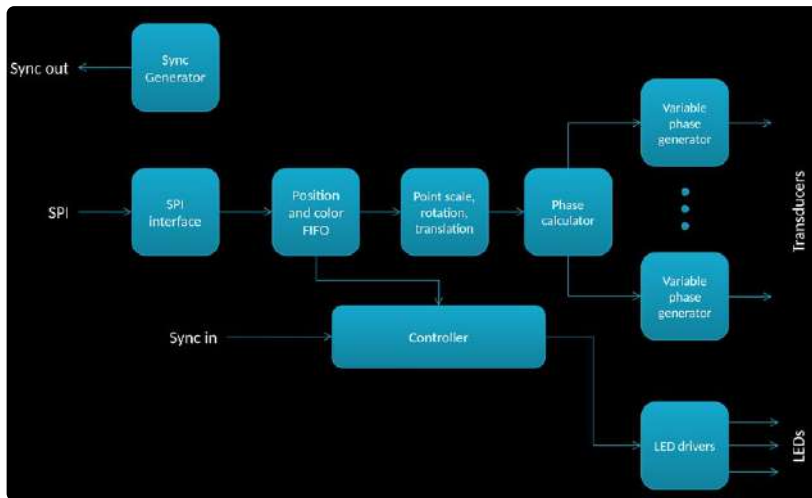
een goede indruk van de opbouw. Het moge overigens duidelijk zijn dat dit project veel te uitgebreid is om tot in detail in Elektor te beschrijven; iedereen die in het VDTP geïnteresseerd is, kan daartoe per e-mail contact opnemen met Dan Foisy (zie het kader Vragen of opmerkingen?).

"Alleen al het blokschema van het VDTP vormde een uitdaging, zie **figuur 2**. De hoofdmoot wordt gevormd door de printen met de transducers, eentje boven en eentje onder. **Figuur 3** toont een van die printen in meer detail.

Figuur 2. Blokschema van het VDTP. De bovenste en onderste print zijn afgezien van de LED-drivers eigenlijk identiek.



Figuur 3. Een van de printen in volle glorie. Elk van de 100 transducers is voor montage doorgemeten.

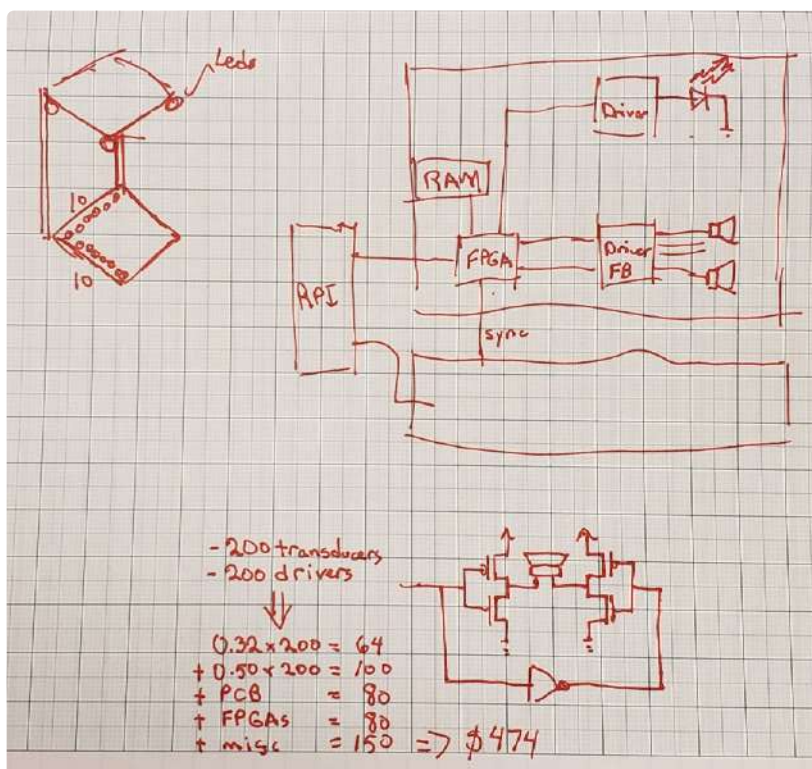


Figuur 4. Hier is het blokschema van de bovenste print verder uitgewerkt.

En in **figuur 4** is een verder uitgewerkt blokschema van de bovenste print te zien (die ook voor de aansturing van de LED's verantwoordelijk is):"

Zoals iedere maker weet, is het een lange weg van idee tot functionerend apparaat; het VDATP bleek in dit opzicht geen uitzondering. **Figuur 5** toont een van de eerste schetsen, en **figuur 6** een uitgewerkt deel van het definitieve ontwerp. Uit **figuur 7** blijkt dat KiCAD voor het printontwerp een zegen is.

Figuur 5. Een van de eerste schetsen van het VDATP.



"Dit was een erg leuk project – het bezorgt me nog altijd veel pret als ik het VDATP in actie zie. Het heeft iets magisch als je een voorwerpje ziet dat in de lucht zweeft en dan zo snel heen en weer schiet dat je het niet meer ziet. Zo iets geeft mij in elk geval veel voldoening." ◀

210237-03

Over de auteur

Dan Foisy is medio de zeventiger jaren in Montreal (Canada) geboren, en groeide op toen personal computers en het internet hun intrede deden – een spannende tijd. Tegen het einde van de jaren '70 verhuisde hij naar Toronto.

Op achtjarige leeftijd kreeg hij een Radio Shack 160-in-1 elektronica kit (het equivalent van de bekende Philips-experimenteerdozen), waarmee hij alarmen, radio's, versterkers en dergelijke bouwde. Ook leerde hij schakelingen naar eigen ontwerp te bouwen.

Aan de Universiteit van Toronto behaalde hij een Bachelor en een Masters of Applied Science. Tijdens zijn Masters-opleiding ging hij voor het pas opgerichte Space Flight Lab werken, en werkte daar aan de eerste ruimtetelescoop van Canada (MOST – Microvariability and Oscillation of Stars [7]). Daarna leidde hij nog vijf andere satellietprogramma's.

Na diverse omzwervingen werkt hij nu bij de Royal Bank of Canada en geeft daar leiding aan diverse teams op AI-gebied, met name Natural Language Processing en de verantwoordelijke en etisch verantwoorde toepassingen van AI.

Toen hij zijn vrouw leerde kennen, gaf hij haar als verlovingsgeschenk een traditionele verlovingsring met diamant. Zijn verloofde vond het toen alleen maar passend hem iets terug te schenken – en kocht een heleboel apparatuur voor zijn lab, die op het moment van schrijven nog altijd in gebruik is. "Voor mij was dat helemaal geen slechte ruil :-)" !"

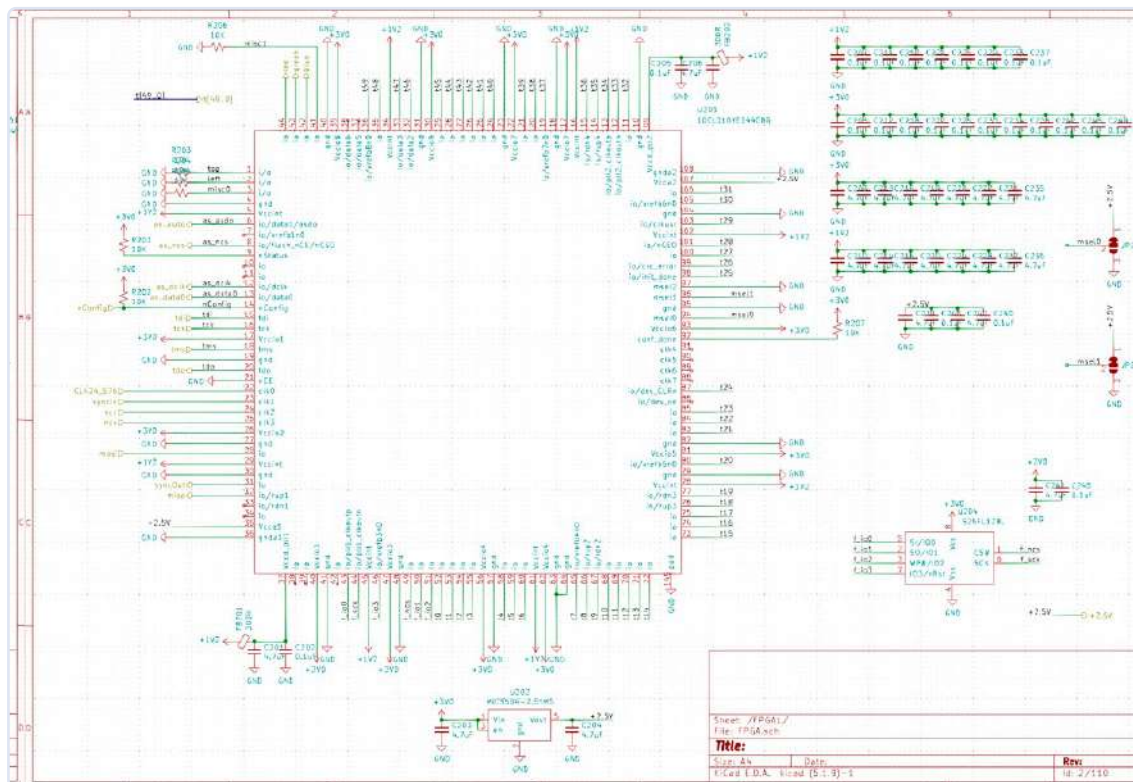
Vragen of opmerkingen?

Wie geïnteresseerd is in nabouw of verdere ontwikkeling het volumetrische display, kan altijd contact met mij opnemen via danfoisy@gmail.com.

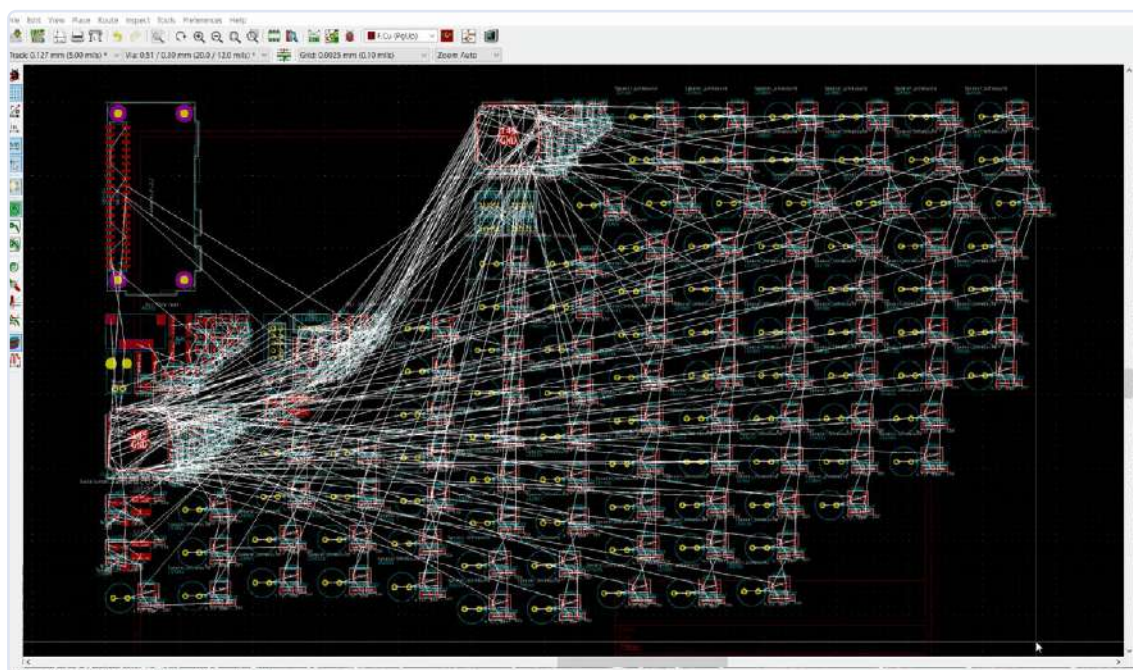


Gerelateerde producten

➤ **Makerfabs Acoustic Levitator DIY Kit (SKU 19984)**
www.elektor.nl/19984



Figuur 6. Een uitgewerkte deelschakeling. Ontwerpsoftware zoals KiCad is bij projecten van deze omvang haast onmisbaar – met potlood en papier is dit onbegonnen werk.



Figuur 7. Het was een hele klus dit 'rats nest' te ontwarren.

WEB LINKS

- [1] "Zweven op geluidsgolven", Elektor juli/augustus 2022: <https://www.elektormagazine.nl/magazine/elektor-266/60629>
- [2] Oorspronkelijk artikel University of Sussex (2019): <https://sro.sussex.ac.uk/id/eprint/86930/>
- [3] VDATP-video University of Sussex: <https://www.youtube.com/watch?v=Tm8JRI1q50>
- [4] LOFAR bij Astron: <https://www.astron.nl/telescopes/lofar/>
- [5] LOFAR bij Wikipedia: <https://nl.wikipedia.org/wiki/LOFAR>
- [6] VDATP-video met (onder andere) vlinder-animatie: <https://www.youtube.com/watch?v=hCC1C5KleUA>
- [7] MOST ruimtetelescoop: https://astro-canada.ca/le_telescope_spatial_most-the_most_space_telescope-eng

Project 2.0

correcties, updates en brieven van lezers

samengesteld door Jens Nickel (Elektor)

Tester voor vermogens-halfgeleiders

Elektor Summer Circuits 2022, p. 56 (210707)

Dit artikel heeft me enorm geholpen bij het repareren van een schakelende voeding: Ik heb de "verdachte" losgesoldeerd en kon toen duidelijk aantonen dat hij defect was. Daarna heb ik hem vervangen door een werkend onderdeel van een andere print.

De tester was gemakkelijk te bouwen uit onderdelen die ik in voorraad had, ik had alleen geen geschikte 12V-lamp. Dus heb ik een weerstand van $47\ \Omega / 5\ W$ (in een metalen behuizing) gebruikt. De spanningsval (via een DIL-bruggeleijkrichter vanwege de polariteit en een serieweerstand van $820\ \Omega$) doet nu een LED oplichten (zie foto).

Dank aan de auteur, David Ashton!

Ulrich Strohmeyer



Fortissimo-100 high-end versterker

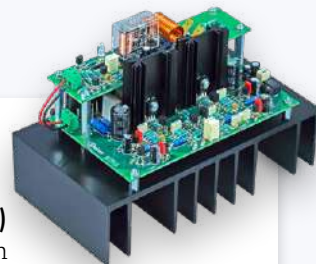
Elektor 11-12/2022, p. 6 (210364)

Ik vind de versterker interessant en het artikel bevat veel gegevens. Maar er is geen informatie over het vermogensverbruik. Mijn vraag is: Hoeveel watt gebruikt een kanaal in rust en bij gemiddeld vermogen?

René Reynders

In rust is de totale stroom maximaal ongeveer 350 mA na het opwarmen, het vermogensverbruik is dan 28 W. Omdat de voeding een stabiele spanning van $\pm 40\ V$ levert, is het maximale gemiddelde vermogen gelijk aan het gespecificeerde continuvermogen. Maar het gemiddelde uitgangsvermogen voor onvervormde muziekweergave is aanzienlijk lager dan het gespecificeerde continue (maximum) uitgangsvermogen. Het totale vermogen van een wisselstroom en een gelijkspanning is het gemiddelde van de stroom maal de gelijkspanning. Voor 181 W bij $4\ \Omega$ is de gemiddelde stroom $\sqrt{(181/2)/\pi} = 3,03\ A$. Dus voor een maximale onvervormde sinusgolf is het totale vermogensverbruik bij $4\ \Omega$ $2 \times 3,03\ A \times 40\ V = 242\ W$. De efficiëntie van de uitgangstrap is 74,7 %.

Ton Giesberts



In memoriam

Helaas is Peter Krengel, één van onze auteurs, overleden. We kenden Peter als een creatieve "Maker", die altijd met nieuwe ideeën kwam. Van reflowovens tot milliohmeters. En hij had nog veel meer projecten en boeken in de pijplijn.

Peter, we zullen je missen!

Het Elektor-team

Een Fliege-notchfilter voor audiometingen

Elektor 9-10/2022, p. 80 (210551)

Ik vond dit artikel meteen interessant: eindelijk een echt nuttig "old school" analoog project, met moderne componenten en hoge kwaliteit!

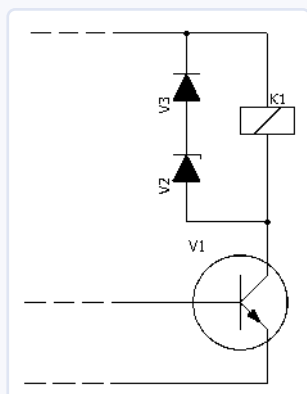
Ik heb contact opgenomen met de auteur van het artikel, en hij was erg vriendelijk en behulpzaam! Hij had nog wat printen over en stuurde die voor een aantrekkelijke prijs naar me op. We hebben ook meerdere vragen en varianten besproken.

Een goede mix maakt Elektor een geweldig tijdschrift: Ga zo door!

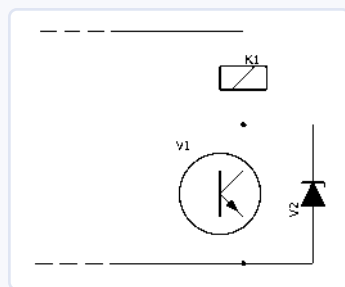
Jens Lemkamp

Dank voor de feedback, fijn dat het zo goed heeft uitgepakt!

Jens Nickel, redactieteam



Figuur 1



Figuur 2

Alle begin...

Elektor 9-10/2022, p. 34 (220256)

Hartelijk dank voor uw artikel in het september/oktober-nummer, waar u heel mooi verschillende toepassingen van diodes bespreekt, zoals de vrijlooptiode voor een relais (in figuur 7 van het artikel).

Ik heb ooit zo'n schakeling gebruikt voor het aansturen van een magneetklep. Het viel me op dat de klep meteen inschakelde, maar uitschakelde met een vertraging. De vertragingstijd verschilde ook van klep tot klep, zodat ik het helaas niet kon compenseren, bijvoorbeeld door iets eerder uit te schakelen.

De oplossing van het raadsel was, natuurlijk, dat de stroom door de spoel maar geleidelijk afnam, omdat de vrijlooptiode een voorwaartse spanning van maar 0,6 V had, zodat het even duurde voordat de sluitveer voldoende kracht had om het van de magnetische kracht te winnen.

Ik heb het probleem opgelost door een zenerdiode te gebruiken, zoals in **figuur 1**. Dankzij de zenerspanning neemt de stroom door de spoel veel sneller af, wat niet alleen de uitschakeltijd aanzienlijk reduceert, maar ook hoorbaar is: een zachte "pling" veranderde in een luide "tonk".

Opmerkelijk dat ik deze truc nooit ergens anders heb gezien, noch in de literatuur noch in echte apparaten.

De schakeling kan nog wat vereenvoudigd worden, door de eigenlijke vrijlooptiode weg te laten en de zenerdiode parallel met de transistor aan te sluiten (**figuur 2**).

Dit kan een interessante variant zijn voor andere Elektor-lezers.

Thomas Klingbeil

Wat ons schoolsysteem kan leren van de maker-community

Ik ben nu 34 en ik volg een technische opleiding in Duitsland om een geavanceerd technisch certificaat te halen (om aan een studie te mogen beginnen). Intussen ben ik ook toegelaten tot een lokale universiteit voor "junior studies" als onderdeel van een programma voor hoogbegaafde leerlingen. Daar volg ik cursussen om alvast wat studiepunten te verdienen.

In principe is er in meer dan tien jaar niets veranderd aan de school: er staat een leraar voor de klas en die lepelt les na les theoretische dingen op, waarvoor de studenten zich nauwelijks praktische toepassingen kunnen voorstellen. De leraren lijken maar niet te begrijpen waarom er, na weken theorie over de Wet van Ohm, zo weinig belangstelling is voor een experiment om de inwendige weerstand van een spanningsbron te bepalen. Telkens weer komt dezelfde vraag: "Waarom zouden we dat willen weten?"

Als we in de maker-community zo zouden redeneren, hadden we wel een probleem. Bij Arduino en dergelijke apparaten valt één ding meteen op: je kunt al snel resultaten zien, en dat werkt heel motiverend. Uitgebreide codevoorbeelden en gerelateerde projecten bieden bijna altijd een startpunt waarmee je je eigen project zo snel mogelijk van de grond krijgt. Onze natuurlijke nieuwsgierigheid naar de vraag "Waarom?" wordt gewekt als er aanpassingen nodig zijn. Ik neem aan, dat vrijwel alle lezers van dit tijdschrift, net als ik, min of meer "blind" zijn begonnen met het bouwen van schakelingen. Zou iemand een projectartikel in Elektor lezen als er geen afgeronde schakeling zou zijn? Of als die pas vier weken later zou worden gepubliceerd? Waarschijnlijk niet.

Natuurlijk kun je zoiets enorms als het onderwijssysteem niet snel dramatisch veranderen. Maar we zouden wel een begin kunnen maken. We praten al heel lang over "de jeugd van tegenwoordig". In mijn jeugd kregen computerspelletjes de schuld; tegenwoordig wijst men naar de sociale media. Maar zowel computerspelletjes als sociale media bieden nou precies dat waar ik voor pleit: snelle bevrediging. Stel je voor dat voor elk computerspel of sociaal platform eerst een wekenlange theoretische voorbereiding nodig zou zijn!

Een heel positief voorbeeld vinden bij Christian Albrechts University in Kiel, in Duitsland, in de vorm van het "Electrical Engineering Studies Entry Project" (www.einfachgutelehre.uni-kiel.de/allgemein/studieneingangprojekt-elektrotechnik). Hier bouwen de studenten in het eerste semester in groepen een metaaldetector, voordat de colleges beginnen. De benodigde kennis vereist voor dat project wordt door de instructeurs gepresenteerd in korte, duidelijke lessen. Elke les geeft precies de kennis die nodig is voor de praktische implementatie. De praktijk is het belangrijkste deel van de projectweek; de vragen over de achtergrond komen dan vanzelf.

Sebastian Westerhold

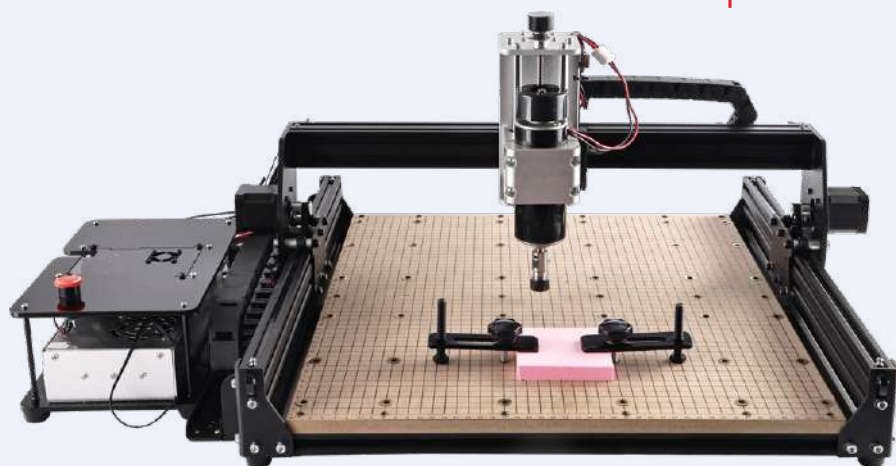
De Elektor Store

Nooit duur, altijd verrassend!

De Elektor-store heeft zich ontwikkeld van de community-shop voor de eigen producten van Elektor (boeken, tijdschriften, kits en modules) tot een volwassen webshop die veel waardevolle elektronica-aanbiedingen heeft.

We bieden hier producten aan waar we zelf enthousiast over zijn of die we gewoon willen uitproberen. Suggesties zijn altijd welkom (sale@elektor.nl).
Ons motto: **nooit duur, altijd verrassend!**

Anet 4540 Desktop CNC-freesmachine



-€ 200

€ 699,00

in plaats van

€ 899,00

Ledenprijs: € 629,10

 www.elektor.nl/20260

Infrarood reflow-oven T-962

Herziene Elektor versie

-€ 40

€ 229,00

in plaats van

€ 269,00

 www.elektor.nl/20346





Great Scott Gadgets HackRF One SDR

-€ 40



€ 299,00

in plaats van

€ 339,00

www.elektor.nl/18306

Raspberry Pi Pico W



Prijs: ~~€ 47,90~~

Actieprijs: € 39,95

www.elektor.nl/20335

RISC-V Assembly Language Programming

+
GRATIS ESP32
RISC-V Board



Prijs: ~~€ 59,90~~

Actieprijs: € 39,95

www.elektor.nl/20296

JOY-IT RD6006 Labvoeding Bundel



-€ 35

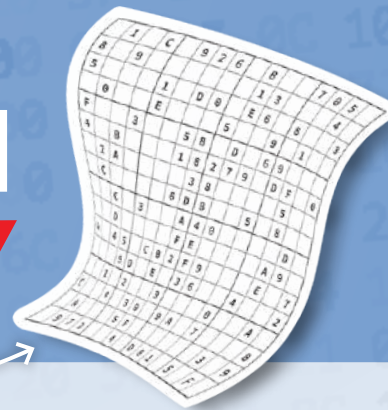
Prijs: ~~€ 199,95~~

Actieprijs: € 164,95

www.elektor.nl/19211

Hexadoku

puzzelen voor elektronici



PC, oscilloscoop en soldeerbout kunnen weer even op adem komen terwijl u uw hersenen pijnigt met onze Hexadoku. De instructies voor deze puzzel zijn heel eenvoudig. De Hexadoku werkt met de hexadecimale cijfers 0 t/m F, helemaal in de stijl van elektronici en programmeurs.

Vul het diagram van 16 x 16 hokjes zodanig in dat **alle** hexadecimale cijfers van 0 t/m F (dus 0...9 en A...F) precies éénmaal voorkomen in elke rij, in elke kolom en in elk vak van 4 x 4 hokjes (gemarkeerd door de dikkere zwarte lijnen). Een aantal cijfers is al aangegeven en deze bepalen de uitgangssituatie voor de puzzel.

Onder de inzenders met de goede oplossing verloten we vijf waardebonnen. Om mee te dingen naar een van deze prijzen dient u **de cijfers in de grijze hokjes** naar ons op te sturen.



Doe mee en win!

Onder de internationale inzenders met het juiste antwoord verloten we vijf Elektor-waardebonnen, elk ter waarde van **50 Euro**.

Het is dus zeker de moeite waard om mee te doen!

Stuur uw antwoord (de getallen in de grijze hokjes) vóór **15 februari 2023** naar:

hexadoku@elektor.nl

DE PRIJSWINNAARS

De juiste oplossing van de Hexadoku uit het november/december-nummer 2022 is: **DBE39**.

Oplossingen die ons vóór 15 december 2022 hebben bereikt, deden mee aan de trekking van vijf Elektor-waardebonnen.

De winnaars zijn bekendgemaakt op www.elektormagazine.com/hexadoku.

Allemaal van harte gefeliciteerd!

E				B	8			3	F						1
	3			6	F		7	D		1	A				8
			7				A	B					2		
	D	8			1	3	5	C	7	E				A	F
	8	F	3			4			B			9	D	C	
B					2		9	8		A					7
	A	E		8		D	1	7	F		4		B	0	
	0	9		B							D		2	5	
	9	7		E							C		3	A	
	E	5		7		A	2	F	1		9		4	6	
D					3		B	4		6					5
	6	A	1			F			2			8	9	B	
	C	3			5	B	E	9	D	4			6	2	
			0				6	A				3			
	B			F	4		3	2		8	1			D	
2					9	C			6	0					B

7	B	8	C	2	A	6	D	1	4	5	E	F	0	9	3
9	D	E	5	1	0	3	8	A	F	6	B	2	7	4	C
F	0	2	3	4	B	E	7	9	C	D	8	5	6	1	A
A	6	4	1	9	C	5	F	7	0	2	3	B	8	D	E
4	7	9	8	A	6	B	1	5	D	E	0	3	C	F	2
B	1	F	A	C	3	0	E	2	6	4	7	D	9	5	8
C	2	D	6	5	7	8	9	B	1	3	F	E	4	A	0
E	3	5	0	D	F	2	4	8	9	A	C	1	B	6	7
D	8	B	9	7	4	C	5	F	2	0	1	A	E	3	6
2	A	7	F	6	8	D	B	E	3	9	4	0	5	C	1
1	C	0	E	F	9	A	3	6	7	8	5	4	D	2	B
6	5	3	4	E	2	1	0	C	A	B	D	7	F	8	9
5	9	1	2	8	E	F	6	D	B	7	A	C	3	0	4
0	E	A	B	3	D	9	C	4	8	F	2	6	1	7	5
8	4	C	D	0	5	7	2	3	E	1	6	9	A	B	F
3	F	6	7	B	1	4	A	0	5	C	9	8	2	E	D

Medewerkers van Elektor International Media en hun familieleden zijn van deelname uitgesloten.

Van **Arduino** en **Elektor** met **Engineering Love**

NU VERKRIJGBAAR

Een unieke editie van Elektor Magazine met gastredactie door Arduino!

DIY elektronica projecten, tutorials en meer van Arduino en Elektor engineers

Vol met projecten, tips en tutorials

Duik in populaire onderwerpen als **MicroPython**, **TinyML** en home automation met Arduino

Leer het Arduino team kennen: Inzichten van **Fabio**, **Massimo** en **David**

Aan de slag met de **Portenta X8**

Koop uw exemplaar in uw favoriete kiosk of bestel het online in de Elektor Store!

Directe links in artikelen geven u gemakkelijk toegang tot Arduino producten en oplossingen



Meer informatie!
www.elektor.nl/arduino-magazine



