

# LED-kweekbox



met state-of-the-art assimilatie-LED's

## MicroSupply



labvoeding  
voor 'connected' apparaten

## DDS-signalgenerator

veel signaal voor weinig geld



MQTT-sensorhub ▷ Flikkervrije LED-dimmer ▷ Het SCCC-project (4) ▷ LED-kweekbox ▷ WSPR-berichten verzenden ▷ Hover-catamaran  
 ▷ A/D-converter met PLD ▷ Draadloze accu-voltmeter ▷ Tijdcorrecte basweergever ▷ USB/RS232 break-out-board ▷ MicroSupply ▷  
 Hardware-ontwerp met (V)HDL (4) ▷ Fab Academy ▷ Qi-laders ▷ Retrotronica ▷ en meer...

## DELOCK® – HET STERKE MERK VOOR IT-TECHNOLOGIE, RANDAPPARATUUR EN ACCESSOIRES

**DELOCK®**  
we move the world

### DisplayPort Splitter

Spiegel of breid het beeldschermseignaal uit tot 3 displays in 4K-resolutie!

- Dubbele Wijze geschikt (DP++)
- DisplayPort 1.4 Multi-Stream Transport
- ondersteunt HDR
- ondersteunt HDCP 1.4 en 2.2.

Bestelnr.: DELOCK 87737

**TECHNOLOGIE  
TIP** **90,39**  
(€ 74,71)



(Geleverd zonder monitoren)



[www.reichelt.com/delock](http://www.reichelt.com/delock)

### LoRa-antenne, 868 MHz

SMA-aansluiting

- Frequentiebereik: 860 - 870 MHz
- ZigBee 868 MHz, Z-Wave 868 MHz
- Antenneversterking: 3 dBi
- Impedantie: 50 Ohm
- Efficiëntie: 45,91
- VSWR: 1,5



Bestelnr.: DELOCK 89769

**9,46**  
(€ 7,82) **BEST  
SELLER**

**DELOCK**  
we move the world

### LoRa-antenne, 863-928 MHz

MHF-aansluiting

- LoRa, NB-IoT, ZigBee, Z-Wave (908, 921), ISM, GSM
- Antenneversterking: 1,68 dBi
- Impedantie: 50 Ohm
- zelfklevende bevestiging
- Kabeltype: 1.13

Bestelnr.: DELOCK 12540

**7,12**  
(€ 5,88) **BEST  
SELLER**



**SCHRIJF U NU IN OP DE NIEUWSBRIEF  
EN PROFITEER VAN DE VOORDELEN!**

Wees altijd de eerste die het weet -  
Topaanbiedingen, interessante  
onderwerpen, acties en nieuwigheden



REGISTREER NU ► <http://rch.lt/v3>

Dagprijzen · Prijzenstand: 1. 8. 2019

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.:+49 (0)4422 955-360

Betaalwijzen:



**www.reichelt.nl**

BESTEL-HOTLINE: +49 (0)4422 955-360

## Colofon

59<sup>e</sup> jaargang nr. 655,  
september/oktober 2019  
ISSN 2590-0765



Elektorlabs is een uitgave van

### Elektor International Media B.V.

Postbus 11, 6114 ZG Susteren, Nederland  
Tel.: +31 (0)46- 4389444,  
Fax: +31 (0)46-4370161

### Nieuwe abonnementen & bestellingen

service@elektor.nl Tel. 046-4389444

Elektor International Media B.V. legt gegevens vast voor de uitvoering van de (abonnementen) 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: Pijper Media, Groningen  
Distributie: Betapress, Gilze



## Hortikulti

Wanneer u dit nummer van Elektorlabs, waaraan ik de laatste hand leg tijdens de tweede hittegolf van dit jaar waarbij de temperaturen de 40 graden benaderen en overschrijden, vluchtig doorbladert, dan zou u kunnen denken dat het een MQTT-special betreft: maar liefst drie bijdragen betreffen projecten die van deze techniek (*Message Queuing Telemetry Transport*) gebruik maken! Maar dat is werkelijk toeval – of misschien een indicatie dat deze methode voor de draadloze overdracht van data steeds meer aan populariteit wint. En dat komt dan niet in de laatste plaats door de beschikbaarheid van goedkope platforms zoals Raspberry Pi, Arduino en de diverse Espressif-modules die voldoende rekenkracht voor deze techniek aan boord hebben.

Die projecten verschillen trouwens als dag en nacht van elkaar: een sensor-hub op basis van onze gigantische LED-klok, een afstandsbediende hover-catamaran en een draadloze accu-voltmeter (voor in de auto). Voor elk wat wils dus.

Natuurlijk blijft het daar niet bij: de audioliefhebbers komen volop aan hun trekken met de *Tijdcorrecte basweergever*, waarbij problemen bij het weergeven van het laagste laag op een slimme manier zijn verholpen. Dat is overigens geen kant-en-klaar nabouwproject maar veeleer een uitnodiging om zelf te gaan ontwerpen, zagen en timmeren.

De multiculturele samenleving werd en wordt door tegenstanders wel denigrerend multikulti genoemd – maar dat woord heeft zich (wat mij betreft) inmiddels tot een soort geuzennaam ontwikkeld. Ik moest daarom voor de hand liggende redenen aan denken bij het lezen van de oorspronkelijke titel van ons hoofdartikel over de LED-kweekbox: *Horticulture Box...*

Een opmerking ten behoeve van lezers van deze uitgave in het buitenland: dat project nodigt wellicht uit tot het kweken van zekere geestverruimende groenten, maar hoewel dat in Nederland min of meer legaal is of in elk geval getolereerd wordt, is dat in andere landen doorgaans niet het geval. U bent gewaarschuwd...

### Eric Bogers

Hoofdredacteur NL

### Advertenties Benelux

Margriet Debeij  
Tel. +49 2 41/955 09 174  
Mobiel: +49 (170) 550 539 6  
E-Mail: margriet.debeij@eimworld.com

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

### Auteursrecht

Niets uit deze uitgave mag verveelvoudigd 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 Rijks{o}ctrooiwet 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.

### Ons team

Hoofdredacteur:

Eric Bogers (eric.bogers@elektor.com)

Internationale redactie:

Jan Buiting, Jens Nickel, Marilene Thiebaut-Brodier

Technische redactie:

Mathias Claussen, Ton Giesberts, Luc Lemmens,

Denis Meyer, Clemens Valens, Jan Visser

Aan dit nummer werkten mee:

Jelle Aarnoudse, Rolf Blijlevens, Kees de Groot,

Martien Jansen, Evelien Snel

Ledenmanagement:

Denise Bodrone

Vormgeving & Layout:

Giel Dols, Jack Jamar

Hoofd Online:

Daniëlle Mertens

Directeur:

Don Akkermans

# in dit nummer

59<sup>e</sup> jaargang – nr. 655

september/oktober 2019

# LED-kweekbox

**16** Elektor Labs Pipeline

**74** Project 2.0

correcties, updates en  
brieven van lezers

**80** HomeLab Helicopter

de wondere wereld van de elektronica  
in vogelvlucht

**83** Shop-highlight

De nieuwe Raspberry...

**99** Vreemde onderdelen

Motorola MC14500  
Industrial Control Unit

**106** JOY-iT JDS6600 DDS-signaalgenerator

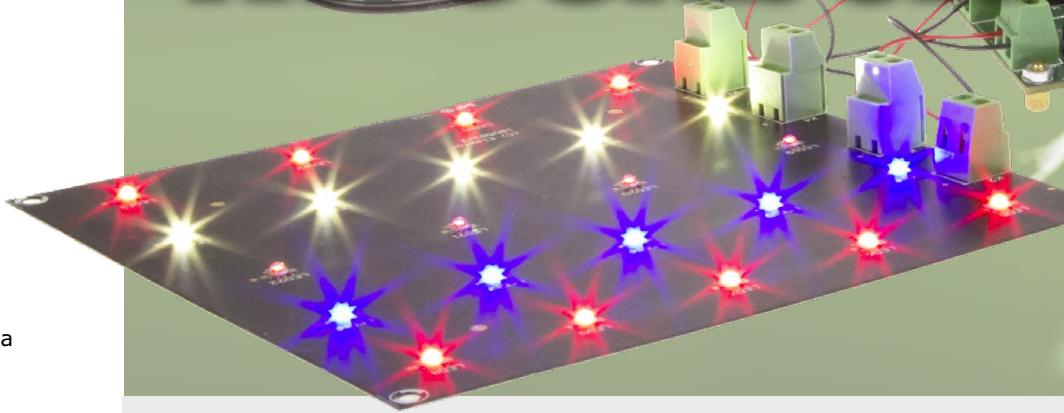
veel signaal voor weinig geld

**108** Retrotronica

Jeweeltjes of e-schroot?

**112** Elektor Store

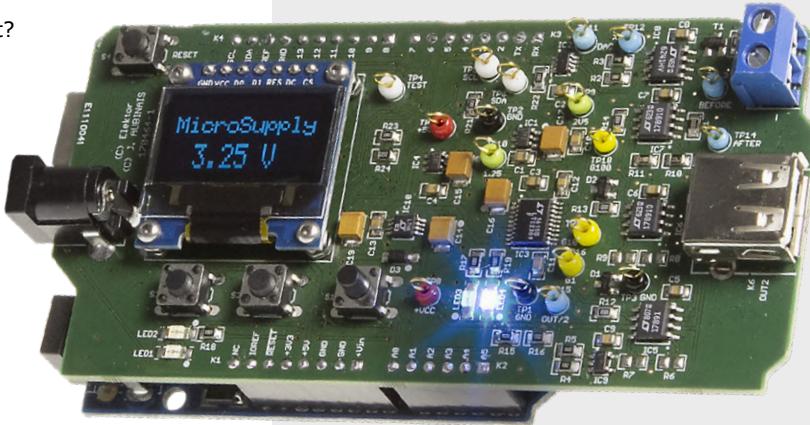
**114** Hexadoku



## MicroSupply

### labvoeding voor 'connected' apparaten

Hoeveel stroom verbruikt mijn IoT-apparaat? Komt het verbruik in de slaapmodus overeen met de specificaties van de fabrikant? Hoelang gaat de CR2032-batterij van het apparaat mee? Als u dit soort vragen hebt, dan is deze MicroSupply (een regelbare voeding in de vorm van een Arduino-shield die zeer kleine stromen kan meten) iets voor u. Met de bijbehorende PC-software kunt u het verbruik van uw IoT-apparaat weergeven en opslaan.



92

## achtergrond & info

**68** Qi-laders

zo gaat inductief opladen van smartphones

**90** Een dag met LoRa

conferentie van de LoRa Alliance in Berlijn

**100** Hardware-ontwerp met (V)HDL (4)

reactietijdmeter

**110** Elektor Ethisiek

Fab Academy

## projecten

**6** MQTT-sensorhub

sensorwaarden inlezen en verzenden met  
de ESP32 Pico-kit

**12** Flikkervrije LED-dimmer

constante stroom en hoog rendement

**17** Het SCCC-project (4)

softcore-processor  
en C-compiler voor zelfbouw

USB/RS232-  
break-out-board  
zeg maar BoB

76



# 22

## met 'assimilatie-LED's' van Würth Elektronik

### JOY-iT JDS6600 DDS-signalgenerator

veel signaal voor weinig geld

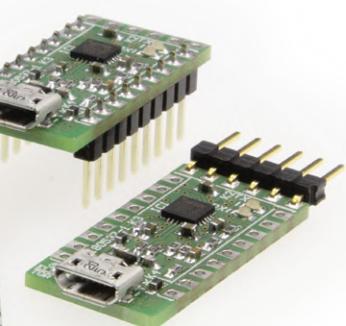
Vroeger had elke elektronicus in zijn (hobby-)lab wel een functiegenerator met een XR2206 staan, maar die tijden zijn al lang voorbij. Tegenwoordig werkt zo'n generator met een DDS-chip die frequenties digitaal opwekt.

Helaas is een fatsoenlijke DDS-functiegenerator niet echt goedkoop – of toch wel? De JOY-iT DDS-functiegenerator belooft een groot frequentiebereik en een heleboel mogelijkheden voor een bodemprijs.



# 106

- 22 **LED-kweekbox**  
assimilatieverlichting met 'horticulture'-LED's van Würth Elektronik
- 32 **WSPR-berichten verzenden**  
bereik heel Europa  
met het Elektor SDR-shield
- 40 **Hover-catamaran**  
met ingebouwde MQTT-broker



**Elektorlabs**  
magazine

- 49 **Eenvoudige A/D-converter met PLD**  
met weinig moeite een sigma-delta-ADC
- 56 **Draadloze accu-voltmeter**  
laadtoestand altijd in de peiling
- 62 **Tijdcorrecte basweergave**  
voor het laagste laag
- 76 **USB/RS232 break-out-board**  
zeg maar BoB
- 86 **ICL7660: reductie van de uitgangsruis**  
met een tweede-orde pi-filter
- 92 **MicroSupply**  
labvoeding voor 'connected' apparaten



binnenkort

**Elektorlabs magazine**  
**november/december 2019**

- 8-kanaals timer met OLED-display • verbeterde stralingsmeter • LED-kaars • softstart voor versterker • zuinig relais • solderen • RFID-reader/writer • Retrotronica • breedband-radio-ontvanger • FreeRTOS voor ESP32 • het SCCC-project (5) • Arduino-temperatuurrecorder ...en nog veel meer.

Aankondigingen onder voorbehoud.

Elektorlabs november/december 2019 verschijnt op 17 oktober 2019.

**Elektor Industry 3/2019**

Elektor Industry 3/2019 is een speciale uitgave gewijd aan Productronica 2019 in München (Duitsland) van 12 tot en met 15 november 2019. Speciale aandacht wordt besteed aan innovatieve technologieën, onderzoek, hardware en software – inclusief IoT, 5G, sensoren, CAD, SMT, PCB-productie en meer. Plus natuurlijk nieuwe afleveringen van de vaste rubrieken als Infographics en Industry Store.

Elektor Industry 3/2019 verschijnt begin november 2019 in druk voor Elektor Gold-leden en als pdf-download voor Elektor Green-leden.

Het nummer is voor belangstellenden ook te koop via [www.elektormagazine.com](http://www.elektormagazine.com). Aankondigingen onder voorbehoud.

# MQTT-sensorhub

## sensorwaarden inlezen en verzenden met de ESP32 Pico-kit

Mathias Claußen (Elektor Labs)

In het mei/juni-nummer 2019 van Elektorlabs presenteerden we de Gigantische LED-klok. Voorzien van vier enorme 7-segment LED-displays kan hiermee niet alleen de tijd worden weergegeven, maar ook via WiFi en MQTT meetdata van een sensorprint. In deze aflevering presenteren we alternatieve firmware die rechtstreekse aansluiting van sensoren op de ESP32 mogelijk maakt. De data-overdracht verloopt via MQTT naar een centrale broker, zoals een RPi met Mosquitto en Node-RED. Voor de firmware hebben we dankbaar gebruikgemaakt van diverse softwaremodules uit andere projecten, wat de ontwikkeltijd voor deze MQTT-sensorhub enorm bekortte.

Het fijne van modulaire software-ontwikkeling is dat gedeelten die al getest zijn in andere projecten hergebruikt kunnen worden. Dat spaart ontwikkeltijd en het scheelt ook veel werk bij het testen en documenteren van de software.

Mathias Claußen van Elektor Labs werkt

inmiddels (onder andere) aan een tweede versie van het ESP32-weerstation uit Elektorlabs januari/februari 2019 [1]. Daarin worden onder andere ook wensen van lezers verwerkt die de eerste versie gebruiken – voor de voortgang en firmware-updates zie [2]. Zo is onder andere voorgesteld om onderstaande I<sup>2</sup>C-sensoren rechtstreeks op de ESP32 aan te sluiten:

- BMP280: sensor voor meet luchtdruk, temperatuur en luchtvochtigheid [5]
- VEML6070: UV-sensor [6]
- TSL2561: helderheidssensor [7]

Aangezien de Gigantische LED-klok gebaseerd is op de ESP32 Pico-kit, was het vrij eenvoudig om de uitlezing van deze sensoren in de code voor dit project op te nemen (op dezelfde manier als we dat hebben gedaan voor de DHT11-sensoren voor luchtvochtigheid en temperatuur in het project Luchtvervuilingsmonitor [13]). Codefragmenten voor het MQTT-datatransport van sensorwaarden naar een MQTT-broker zijn eveneens ontleend aan de weerstation-software. Het is niet handig als we voor elke aanpassing of firmware-update de klok van de wand moeten halen, daarom heb-

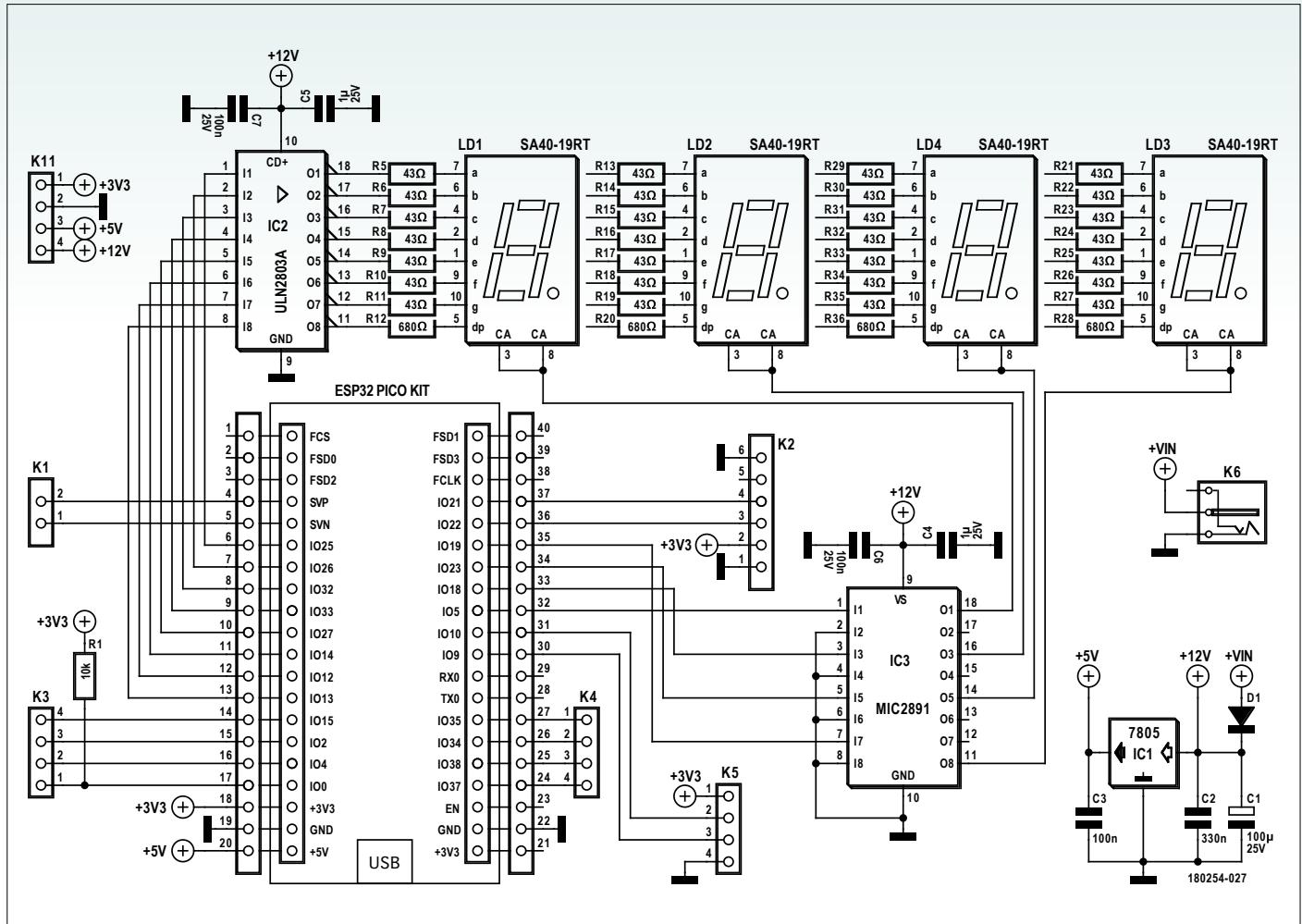
ben we de OTA-library van de Arduino toegevoegd aan de ESP32. Dan kunnen we hem voortaan via Wifi updaten (zie het kader **OTA programmeren met Arduino**).

### Sensorkoppeling

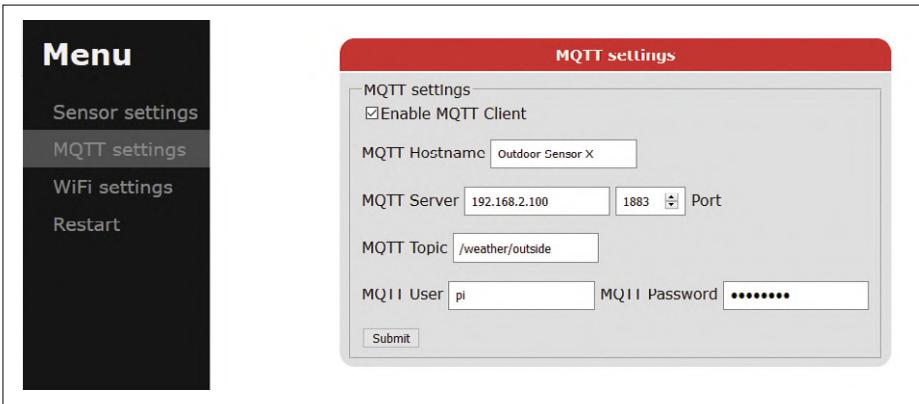
Op de print zijn er geen speciale connectoren voorzien voor het aansluiten van sensoren. I/O-pinnen van de ESP32 die niet nodig zijn voor het display, zijn aan de onderkant naar buiten uitgevoerd (zie **figuur 1**). Pinheaders of bedrading kunnen daar worden gemonteerd. De I<sup>2</sup>C-pennen zijn gemerkt, voor DHT11/DHT22 gebruiken we IO15 als datalijn. Bijzonder bij de ESP32 is zijn I/O-matrix. Bij microcontrollers van bijvoorbeeld de ATmega-reeks ligt vast welke pinnen welke functie hebben, en kunt u zelfs conflicten krijgen wanneer u speciale functies wilt gebruiken. De ESP32 doet dit beter: dankzij een I/O-matrix kunnen we in principe elke functie toewijzen aan (bijna) elke ESP-pin.

We kunnen dus niet alleen de I<sup>2</sup>C-interface van de klok gebruiken, maar ook een van de UART's aan een vrije pin toewijzen en gebruiken. We zouden zelfs een audio-codec op delen van de I<sup>2</sup>S-interface kunnen aansluiten. Uitbreidingen kunnen ook probleemloos via SPI wor-





Figuur 1. Schema van de Gigantische LED-klok met vrij te gebruiken pinnen van de ESP32.



Figuur 2. Configureerbare MQTT-instellingen.

den aangestuurd. Met de ESP32-Arduino-core-library's kan de matrix worden geconfigureerd. Zo kunnen we in de code per parameter vastleggen welke pin we waarvoor willen gebruiken. Een voorbeeld daarvan is het project 'ESP32 als tijdserver' [8]. Voor het OLED-display bij dat project worden de GPIO-pinnen 5 en 4 van de ESP32 gebruikt voor SDA en SCL. Met het commando:

```
Wire.begin(5,4,100000);
```

stellen we de pinnen in, waarbij de derde parameter de gewenste I<sup>2</sup>C-frequentie is. Instellen van de ESP-UART's gaat net zo. Nadelen van deze vrijheid zijn een langere signaalduur en een geringere bandbreedte. De matrix-hardware legt de grens bij 40 MHz. Snellere signalen kunnen niet gerouteerd worden. Vanwege latentietijden kan bij meer dan 20 MHz voor SPI en I<sup>2</sup>C een verschuiving van de data ten opzichte van de klok optreden, waardoor mogelijk bits verkeerd worden

geïnterpreteerd.

### Zenden met MQTT

Gaan we nu over tot wat de sensorhub als nieuwe versie van de LED-klok via MQTT stuurt. Dat is een JSON-string met de volgende opbouw:

```
{"data":  
  {"temperature":31.1,  
   "humidity":25.9,  
   "airpressure":0,  
   "PM2_5":-1,  
   "PM10":-1,  
   "Lux":-1,  
   "UV":-1}  
}
```

Het object `data` bevat de meetwaarden. Als er geen sensor herkend wordt, krijgen de parameters een default-waarde. Voor `temperature`, `humidity` en `air pressure` is dat nul. Alle temperaturen hebben de eenheid °C. De overige default-waarden

zijn –1. Als de betreffende sensor bij het opstarten beschikbaar is, dan staan hier echte meetwaarden.

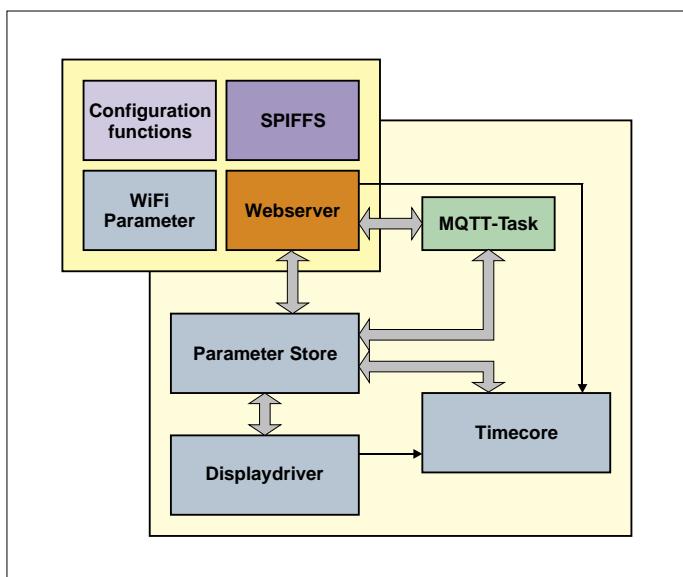
Net als bij de eerste versie van de firmware voor de LED-klok kunnen we de server (de MQTT-broker) en het topic van de verstuurde berichten instellen via de configuratie-webpagina van de ESP32-webserver (**figuur 2**). Hoe u een RPi als MQTT-broker laat werken is beschreven onder [4]. Daar vindt u ook een toelichting op de installatie van Node-RED [9], waarmee snel kleine applicaties gebouwd kunnen worden die sensordata uitlezen en weergeven.

### Softwarecomponenten

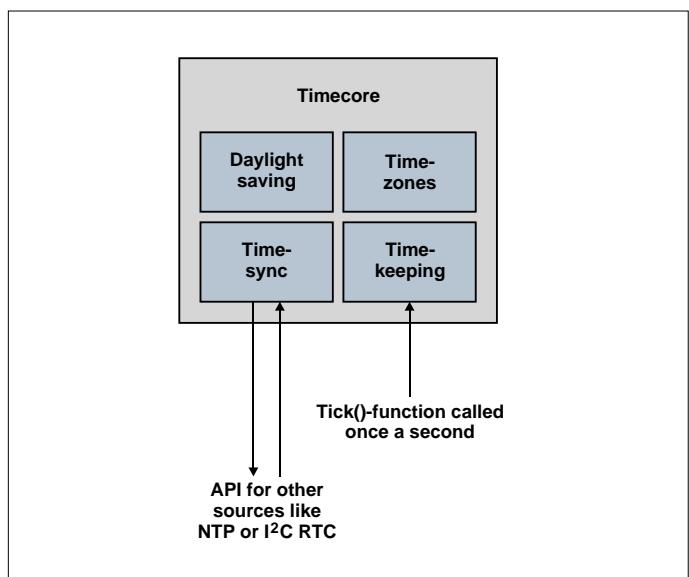
Tot zover *wat* we hebben ingebouwd; nu is het tijd voor het *hoe*: wie software schrijft, houdt er niet van om steeds weer opnieuw het wiel te moeten uitvinden. Als u steeds op hetzelfde platform werkt (bijvoorbeeld een specifieke MCU-familie), dan verzamelt u bijna vanzelf allerlei routines, bibliotheken en softwaremodules die u vaker gebruikt. Alleen al door het grote aantal ESP-projecten in het Elektor Lab was dat ook bij dit project het geval. In principe hebben vergelijkbare netwerkprojecten zoals het weerstation of de LED-klok dezelfde kern, die uit een webserver, WLAN-parametrering via een webbrowser en de configuratieroutines. Dat hoeven we dus niet steeds opnieuw te ontwikkelen.

### Ticker

Voor de klok voegen we een deel van de Elektor Flipperklok [10] toe, dat de tijd en de synchronisatie daarvan, tijd-



Figuur 3. Overzicht van de gebruikte softwaremodules.



Figuur 4. De functies van TimeCore.

zones en zomer- en wintertijd voor zijn rekening neemt. Dit stuk code is ontwikkeld onder de interne benaming *TimeCore*. **FIGUUR 3** geeft een overzicht van de modules.

De TimeCore-bibliotheek importeert zelf al een NTP-client en kan in runtime verbinding maken met andere tijdbronnen zoals bijvoorbeeld een I<sup>2</sup>C-RTC. We moeten er wel voor zorgen dat er elke seconde een TimeCore-functie wordt aangeroepen om intern één seconde verder te tellen (zie **FIGUUR 4**). Daarvoor gebruiken we de bibliotheek *Ticker* waarmee we op een simpele manier periodiek functies kunnen laten uitvoeren. De bijbehorende code is:

```
#include
...
Ticker OneSecondTick;
...
OneSecondTick.attach_ms(1000, FNC);
```

De uitdrukking **FNC** is een substituut voor de aan te roepen functie. Net als bijvoorbeeld **void fnc(void)** mag deze geen argumenten krijgen, noch enige waarde retourneren. De eerste parameter van de functie **attach\_ms** is de tijd in milliseconden tussen twee aanroepen. Ook andere functies zijn op deze manier periodiek aan te roepen, alleen zit daar wel een ‘maar’ aan. Op een ESP32 draait het besturingssysteem FreeRTOS, maar in de Arduino-IDE is dat van ons afgeschermd. Bij **Ticker** moeten we het dus doen met een tweede *task* die met het aangegeven interval de opgegeven functie met heel lage prioriteit aanroept. Daardoor kan het voorkomen dat de op te roepen functie door een andere task met hogere prioriteit onderbroken wordt, want interrupts worden hier niet gebruikt.

## Tasks

In de code speelt MQTT een bijzondere rol. We zouden weliswaar alle functies in de functie **loop()** kunnen zetten, maar dan is het geheel niet meer zo ‘lekker’ modulair. In plaats daarvan geven we de voorkeur aan een eigen task. Zodra MQTT geen data meer te verwerken heeft, mag deze task gaan slapen totdat er nieuwe instellingen voor hem zijn. Het kan de MQTT-task enige seconden kosten om verbinding te maken met de server. Dat merken we vooral als de aangeroepen server bij het opbouwen van de verbinding een timeout genereert. Wanneer het opbouwen van die verbinding in een aparte task plaatsvindt, dan kan Free-

RTOS dat in het ideale geval doen wanneer andere tasks (zoals de loop-functie) even niets te doen hebben.

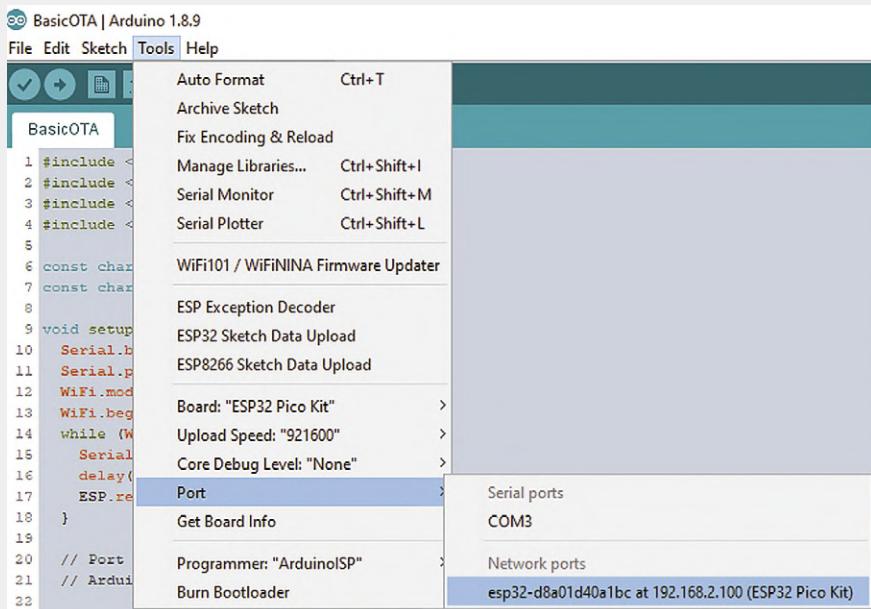
Die task moet echter wel weten dat er nieuwe parameters klaarstaan. Dat is makkelijk te realiseren, omdat elke task

in FreeRTOS zogenaamde *notifications* kan ontvangen (zie **FIGUUR 5**) wanneer de zogenaamde *handle* van die task tenminste bekend is. In **xTaskCreatePin nedToCore** is de zesde parameter een pointer naar een variabele van het type **TaskHandle\_t**.

## OTA-programmeren met Arduino

Een ESP32 kunnen we voorzien van nieuwe software via zijn seriële poort. Dat is wat spartaans, maar meestal wel toereikend. Echt comfortabel wordt het echter met OTA-programmering (*Over The Air*). Om dat te kunnen laten werken moet er firmware in flash geladen zijn die OTA-updates ondersteunt. Een minimalistisch voorbeeld vindt u in de IDE onder *Example → ArduinoOTA → BasicOTA*. Als we dit gedeelte hebben toegevoegd aan onze eigen firmware, en als we de firewall van onze ontwikkel-PC op de juiste manier hebben ingesteld, dan zien we een nieuwe netwerkpoort die we kunnen kiezen in de IDE (zie **afbeelding**).

Bij het uploaden van nieuwe firmware schrijft een uploadskech de data naar een eigen OTA-partitie in het flashgeheugen. Dat is de reden dat op veel boards maar maximaal de helft van het totale vrije flashgeheugen in de MCU-hardware voor applicaties kan worden gebruikt. Behalve de firmware kunt u ook de inhoud van het SPIFFS-geheugen overschrijven, waar zich normaliter de pagina’s van de webserver bevinden. Aangezien de OTA-functie zelf deel uitmaakt van de firmware, lukken verdere firmware-updates alleen als die nieuwe firmware het OTA-gedeelte uitvoert zonder te crashen.



Om ervoor te zorgen dat de Arduino-IDE zichtbaar is, stuurt de ESP32 op de achtergrond informatie over zijn OTA-capaciteit en het bijbehorende IP-adres via mDNS aan de andere aanwezigen op het netwerk. Bij het starten van een update worden er dan UDP-pakketjes uitgewisseld en wordt er zo nodig naar een wachtwoord gevraagd. Is er geen wachtwoord ingesteld, dan kan iedereen in het netwerk updates uitvoeren, wat niet geheel zonder risico is. Is er daarentegen wel een wachtwoord ingesteld, dan kunt u vanuit de Arduino-IDE niet meer het SPIFFS-geheugen updaten. Aan de andere kant riskeert u met OTA-update zonder wachtwoord natuurlijk dat een of andere grappenmaker of hacker iets met de firmware van uw ESP32 uithaalt. Daarom dient u zorgvuldig af te wegen of en hoe u OTA toepast. Mogelijk zijn er extra maatregelen nodig om ongewenste updates te verhinderen.

```

xTaskCreatePinnedToCore(
    MQTT_Task,
    /* Function to implement the
     * task */
    "MQTT_Task",
    /* Name of the task */
    10000,
    /* Stack size in words */
    NULL,
    /* Task input parameter */
    1,
    /* Priority of the task */
    &MQTTTaskHandle,
    /* Task handle */
    0);
/* CORE to pin */

```

Wanneer de task met goed gevolg geïmplementeerd is, krijgt de variabele waar de pointer naar verwijst de waarde van de handle. Daarmee kunnen andere functies notifications naar die task sturen. Een task kan zelfs wachten op notifications. In de MQTT-task wordt dat bewerkstelligd met de volgende regel:

```

ulNotificationValue =
    ulTaskNotifyTake(pdTRUE,
    portMAX_DELAY);

```

De eerste parameter zorgt ervoor dat alle bits na het lezen worden gereset, de tweede parameter bevat de maximale wachttijd (hier oneindig). Er wordt een 32-bit-waarde gereturneerd waarvan we de bits met de functie `mqttsettings_update` in `webserverfunctions.cpp` kunnen zetten.

```

if(MQTTTaskHandle != NULL){
    xTaskNotify(MQTTTaskHandle, 0x01,
    eSetBits);
}

```

Dit codefragment kijkt eerst of de handle gezet is. Daarna wordt bit 0 in de notification-waarde van de MQTT-taks gezet. Als die op dat moment slaapt, dan wordt hij hierdoor wakker en gaat hij aan het

werk. Met deze notification krijgt de MQTT-task het signaal dat de parameters voor het datatransport gewijzigd zijn. De task kan dan de nieuwe instellingen inlezen en verwerken.

Een verdere beschrijving van FreeRTOS en zijn inter-task-communicatie zou voor dit artikel te omvangrijk zijn, maar we hebben voor de komende uitgave een artikel in de planning dat dieper op de mogelijkheden van FreeRTOS en task-programmering in zal gaan.

### Modulaire display-aansturing

De belangrijkste taak bij het weergeven van data is de aansturing van de 7-segment LED-displays. Ook hier hebben we gekozen voor een modulaire opzet, zodat we de code ook in andere projecten kunnen gebruiken. Anders dan in de oorspronkelijk versie moet er nu echter een hardware-versie als parameter worden meegegeven. Zoals al aangekondigd in het artikel over de gigantische LED-klok [3] staat er een mini-versie op de planning, maar het Elektor Lab is op dit moment nog niet helemaal tevreden met het printlayout daarvoor. Om die reden houdt de driver al rekening met eventuele aanpassingen van de layout, en is dan niet meer gebonden aan deze specifieke klok: u kunt hem ook eenvoudig met eigen projecten combineren.

Nu volgen nog een paar tips met betrekking tot de plaatsen die interessant kunnen zijn wanneer u de code aan eigen hardware wilt aanpassen. De functie `SevenSegmentSetup(BedroomclockHW_t HW)` zorgt in het array `DisplaySegmentPins` voor de pintoewijzing aan de LED-segmenten A tot en met G en de decimale punt. In `sevensegments.cpp` vinden we:

```

volatile int DisplaySegmentPins[] = ;

```

De parameters komen overeen met de segmenten A t/m G, de laatste parameter is de decimale punt. Hebt u een display met andere pinout, dan kunt u deze waarden daaraan aanpassen. Hetzelfde geldt voor de common-aansluitingen van de LED-displays. Met dit array legt u de common-aansluitingen 0 tot en met 3 vast:

```

volatile int DisplayCommonPins[] = ;

```

Bij het aanroepen van de functie `SevenSegmentSetup()` moeten we opgeven om welke hardware het gaat. Op die manier wordt gezorgd dat de juiste pinout voor de desbetreffende hardware in beide

boven genoemde arrays geschreven wordt. Deze gegevens slaan we op in het flashgeheugen, maar omdat we een timer-interrupt gebruiken moeten we die data ook nog een keer kopiëren naar het RAM-geheugen.

Dit laatste heeft de volgende reden: we gebruiken een timer van de ESP32 om ervoor te zorgen dat de timing voor het multiplexen van het display klopt. Deze timer triggert met 400 Hz de functie voor het aansturen van de individuele segmenten. Met een microcontroller type AVR of Cortex-M is het heel makkelijk om data in flashgeheugen uit te lezen. Met een ESP32 komen we daar echter niet mee weg, want de eigenlijke MCU heeft geen geïntegreerd flashgeheugen, maar alleen RAM en boot-ROM. De software zit altijd in een externe QSPI-flash-chip. Omdat de leessnelheid daarvan beperkt is, heeft de MCU ook een cache die de zojuist gebruikte of de vermoedelijk weldra nodige data bevat. Daardoor draait de code veel sneller dan het QSPI-flash zou kunnen leveren. Bij een interrupt-routine kan het echter voorkomen dat de betreffende code niet in het cache-geheugen staat maar eerst uit het flash-geheugen moet worden opgehaald. Dat vertraagt de boel, vooral als er tussendoor ook nog andere taken draaien, of (erger) als er in het flash-geheugen geschreven wordt, wat het uitlezen in de weg zit. Een vlotte interrupt-afhandeling is dan niet mogelijk. Het is niet alleen dat lezen tijd kost, als het lezen botst met een schrijfproces, dan veroorzaakt dat een exception en een herstart van de ESP32. Vandaar dat alle data die een interrupt nodig zou kunnen hebben, in RAM beschikbaar moet zijn. De functie die de timer aanroeft moeten we declareren met `IRAM_ATTR()`, waardoor de compiler en de linker de code zodanig inrichten dat die direct na opstarten in RAM wordt gezet en van daaruit wordt uitgevoerd. Op die manier omzeilen we dus het probleem van trage flash-operaties.

Dan is er nog de vraag **waar** de interrupt uitgevoerd wordt. Het antwoord "op de MCU" lokt onmiddellijk de wedervraag uit "en waar op de MCU dan?", want een ESP32-MCU heeft twee cores. In principe wordt een interrupt afgehandeld op de core die hem aanriep. Als u meer wilt weten over hoe de ESP32 dergelijke dingen afhandelt, dan is het aan te bevelen om de documentatie van de ESP-IDF [12]



#### IN DE STORE

→ Print voor de Gigantische

LED-klok

[www.elektrormagazine.nl/180254-1](http://www.elektrormagazine.nl/180254-1)

→ ESP32-PICO-Kit V4

[www.elektrormagazine.nl/esp32-pico-kit-v4](http://www.elektrormagazine.nl/esp32-pico-kit-v4)

→ Boek (Duitstalig): "IoT-

Programmierung mit Node-RED"

[www.elektrormagazine.nl/18357](http://www.elektrormagazine.nl/18357)

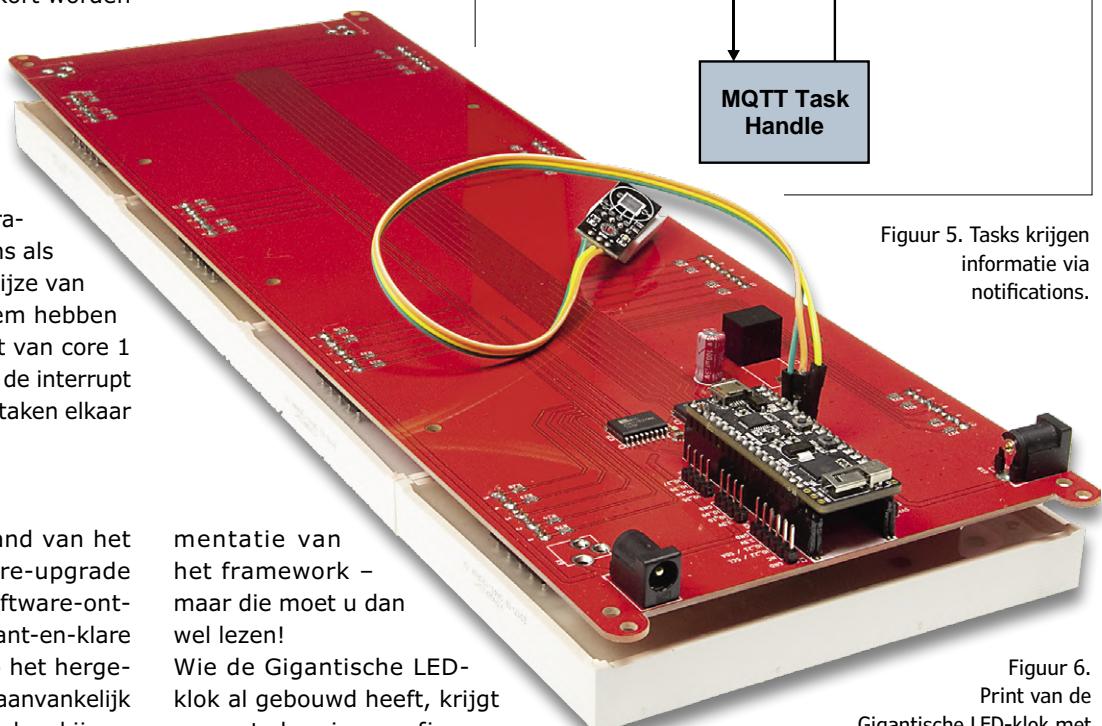
door te nemen.

In het Elektor Lab hebben we onze ESP32 voorzien van een DHT11-sensor voor het meten van temperatuur en luchtvochtigheid. Een nadeel van de daarbij gebruikte bibliotheek is dat tijdens het oproepen van de sensordata bij de actuele kern de interrupts kort worden uitgeschakeld. Toen we de software aan het ontwikkelen waren, gingen daardoor de timer-interrupt voor de display-multiplexing en de taak voor het weergeven van tijd of temperatuur uit de pas lopen, telkens als we de sensor uitzetten. Bij wijze van simpele fix voor dit probleem hebben we de MQTT-task verplaatst van core 1 naar core 0. Zodoende loopt de interrupt ongestoord en zitten ook de taken elkaar niet in de weg.

## Conclusie

Dit artikel geeft aan de hand van het voorbeeld van een firmware-upgrade enig inzicht in efficiënte software-ontwikkeling met behulp van kant-en-klare libraries. De nadruk ligt op het hergebruik van code. Ook als dat aanvankelijk wat meer werk vergt, plukt u daar bij een later project de vruchten van. Niettemin kan het ondanks software-recycling en libraries geen kwaad om zo nu en dan eens een blik onder de motorkap te werpen. Hoewel het Arduino-framework en de ESP-IDF ons veel werk uit handen nemen, zijn er vaak verborgen details die belangrijk zijn om te weten. De oplos-

sing van gemene instinkters en foutsituaties waar u vaak urenlang op zit te puzzelen, is in veel gevallen te vinden in een voetnoot in een datasheet of in de docu-



Figuur 5. Tasks krijgen informatie via notifications.

mentatie van het framework – maar die moet u dan wel lezen!

Wie de Gigantische LED-klok al gebouwd heeft, krijgt er met de nieuwe firmware veel nieuwe functies bij die van de klok een MQTT-sensorhub maken. In **figuur 6** ziet u de print van de klok met een DHT-sensor die rechtstreeks op de onderzijde van de print is aangesloten. U zou zelf bijvoorbeeld de eendraads-thermometer DS18B20 kunnen gebruiken, of kunnen proberen om er een sensor op een



Figuur 6. Print van de Gigantische LED-klok met rechtstreeks aangesloten sensor.

andere locatie via een LoRa-module mee te verbinden. Interessante mogelijkheden te over! ▶

(180254-B-04)

## Weblinks

- [1] Weerstation met ESP32: [www.elektormagazine.nl/180468-01](http://www.elektormagazine.nl/180468-01)
- [2] Weerstation met ESP32 op Elektor Labs: [www.elektormagazine.nl/labs/esp32-weather-station-180468](http://www.elektormagazine.nl/labs/esp32-weather-station-180468)
- [3] Gigantische LED-klok: [www.elektormagazine.nl/180254-03](http://www.elektormagazine.nl/180254-03)
- [4] Gigantische LED-klok op Elektor Labs: [www.elektormagazine.nl/labs/bedroom-clock-with-out-side-temperature-based-on-esp32](http://www.elektormagazine.nl/labs/bedroom-clock-with-out-side-temperature-based-on-esp32)
- [5] BMP280: [www.bosch-sensortec.com/bst/products/all\\_products/bmp280](http://www.bosch-sensortec.com/bst/products/all_products/bmp280)
- [6] VEML6070: [www.vishay.com/ppg?84277](http://www.vishay.com/ppg?84277)
- [7] TSL2561: <https://ams.com/tsl2561>
- [8] ESP32 als tijdserver: [www.elektormagazine.nl/180662-03](http://www.elektormagazine.nl/180662-03)
- [9] Node-RED: <https://nodered.org>
- [10] Elektor Flipperklok: [www.elektormagazine.nl/180307-03](http://www.elektormagazine.nl/180307-03)
- [11] FreeRTOS: [www.freertos.org](http://www.freertos.org)
- [12] ESP-IDF: <https://docs.espressif.com/projects/esp-idf/en/latest/>
- [13] Luchtvervuilingsmonitor: [www.elektormagazine.nl/170182-02](http://www.elektormagazine.nl/170182-02)

# Flikkervrije LED-dimmer

## constante stroom en hoog rendement

Ing. Joost Waegebaert (België)

Het dimmen van LED-lampen lijkt op het eerste gezicht tamelijk simpel. Neem een PWM-uitgang van de een of andere microcontroller, jaag het signaal door een power-MOSFET en klaar is Kees. In dit artikel gaan we echter van een andere opzet uit – helemaal niet nieuw, maar vaak over het hoofd gezien – die een LED perfect en flikkervrij dimt, en dat bij een hoog rendement. De schakeling is opgebouwd met standaard onderdelen, zodat hij ook voor educatieve doeleinden geschikt is.



De wens om LED's te dimmen is niet nieuw. Er circuleren inmiddels ontelbare oplossingen – ook die welke we hier presenteren. Maar deze wordt vaak over het hoofd gezien, en dat terwijl hij een belangrijk voordeel heeft ten opzichte van de 'standaard' PWM-methode.

### **Het probleem met PWM**

Het gebruik van een pulsbreedte-gemoduleerde stroom (*Pulse Width Modulation*, PWM) om de helderheid van een LED te regelen is een bekende methode. Bij een PWM-signaal gaat het om een aan/uit-signaal met vaste frequentie maar met variabele pulsbreedte. Hoe breder de puls, des te langer is de LED

aan en des te groter zijn gemiddelde lichtopbrengst [1].

Gedurende de uit-tijd loopt er geen stroom en geeft de LED effectief geen licht. Hierdoor 'flikkert' de LED. Wanneer de PWM-frequentie hoog genoeg is ziet ons oog de 'gemiddelde' lichtopbrengst, maar dat interval waar de LED geen licht geeft blijft natuurlijk bestaan. Dit kan vermoeiend zijn voor het oog – en brengt ook andere problemen met zich mee.

Flikkering kan merkbaar worden wanneer we bijvoorbeeld een foto maken of een video opnemen: hoewel wijzelf een perfect uitgelicht onderwerp zien, kan een camera in de problemen komen wanneer

die niet synchroon is met de gepulste LED's. Wanneer een frame wordt opgenomen tijdens de uit-tijd van de LED's, is het resultaat een niet of onvoldoende belicht frame. Ook snel bewegende objecten kunnen er 'flikkerend' uit zien ten gevolge van het gepulste licht.

Bij PWM wordt de LED-stroom zo snel mogelijk in- en uitgeschakeld. Hierdoor ontstaan grote stroompieken, compleet met de bijbehorende elektromagnetische storingen (*Electromagnetic Interference*, EMI). LED's kunnen om die reden dan ook niet met lange draden op de PWM-driver worden aangesloten zonder dat die draden als een stoerstraler gaan werken.

## Alternatieve aanpak

Het is heel goed mogelijk om een gewone buck-converter te gebruiken om een bepaalde stroom voor de LED te genereren. In **figuur 1** laat S1 periodiek een stroom naar L1 en D2 door. Het nadeel van deze specifieke opzet is echter juist die schakelaar S1: een MOSFET die direct in de positieve voedingsaansluiting is opgenomen, is moeilijker aan te sturen dan een MOSFET die naar massa schakelt. Om te garanderen dat de MOSFET de stroom met zo min mogelijk verliezen doorlaat, is een spanning nodig die hoger is dan de voedingsspanning, en dat maakt het geheel ingewikkelder.

Dit probleem kan worden opgelost door de onderdelen iets anders te arrangeren, met als resultaat het schema van **figuur 2**: de zwevende buck-converter. Deze wordt ‘zwevend’ genoemd omdat de belasting (D2) niet (direct) aan massa ligt. In deze opzet kunnen we een gewone N-kanaal MOSFET voor S1 gebruiken.

Als S1 gesloten is, neemt de stroom door L1 toe. Deze stroom wordt gemeten en wanneer een bepaalde waarde is bereikt, wordt S1 geopend. De stroom door L1 neemt dan af. S1 wordt weer gesloten voordat die stroom nul wordt.

In deze opzet loopt de stroom door L1

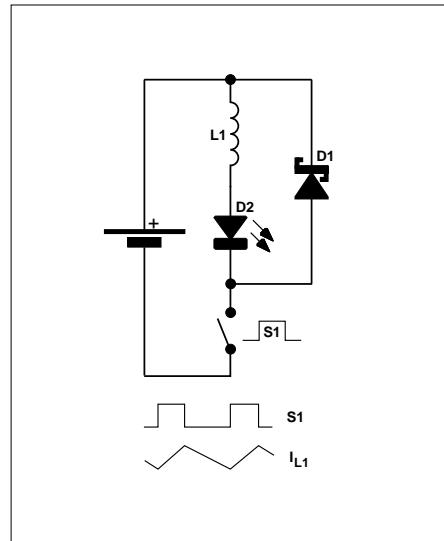
ook door LED D2. Deze stroom is niet intermitterend en ziet er in principe als een driehoek uit (zoals in figuur 2 geschetsst). Door deze ‘rimpel’ wordt de helderheid van de LED nog altijd gemondeerd, maar die modulatie is enkele orden van grootte kleiner dan bij klassieke PWM, waardoor de flikkering vrijwel helemaal wordt geëlimineerd. Datzelfde geldt ook voor de EMI die door de aansluitdraden van de LED wordt afgestraald (een driehoek met kleine amplitude, in tegenstelling tot een blokgolf met grote amplitude).

## De schakeling

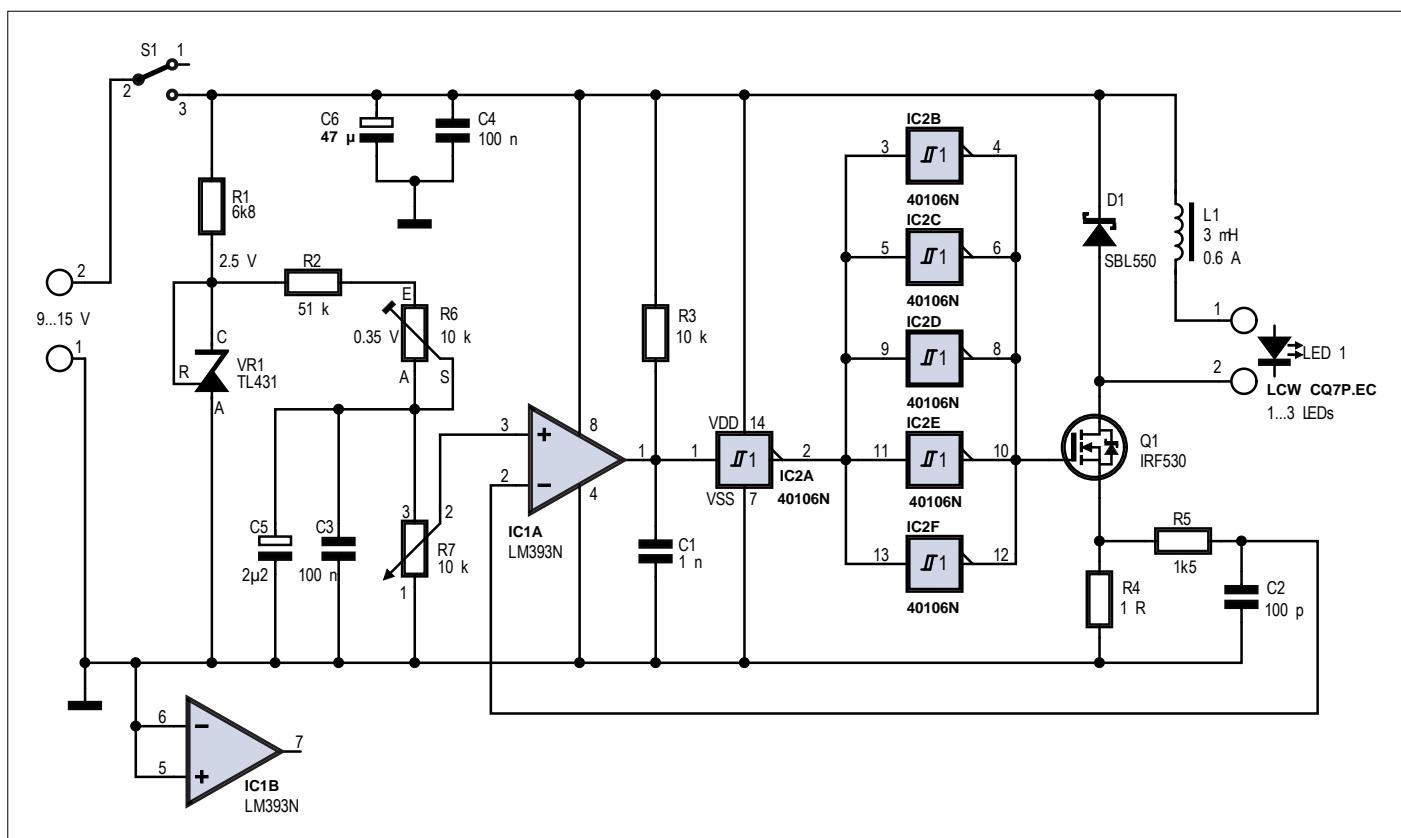
Een praktische uitvoering van de zwevende buck-dimmer is getekend in **figuur 3**. Om het achterliggende principe duidelijk te maken, zijn hier standaard componenten tot een werkende schakeling gecombineerd.

Comparator IC1A vergelijkt de stroom door schakelaar T1 met een variabele referentiespanning op de loper van R7. Bij inschakelen zal T1 doorschakelen; de stroom door L1 neemt toe en veroorzaakt een spanningsval over R4. Zodra deze spanning even groot is als die op de loper van R7, zal de open-collector uitgang van IC1A doorschakelen, waardoor C1 wordt ontladen en T1 wordt uitgeschakeld.

Figuur 1. Principe van een buck-converter.



Figuur 2. De ‘zwevende’ buck-converter.



Figuur 3. Het uitgewerkte schema van de constante-stroom LED-dimmer.

## Een beetje rekenarij

De uit-tijd van de converter wordt bepaald door de R3/C1-tijdconstante. Het triggerniveau van de 40106 Schmitt-trigger ligt bij ongeveer de helft van de voedingsspanning. Een RC-laadcyclus tot ongeveer de helft van de voedingsspanning duurt:

$$t_{\text{OFF}} \approx 0,7 \times R3 \times C1 = 0,7 \times 10^4 \times 10^{-9} = 7 \mu\text{s}$$

Voor de spanning over L1 als T1 aan is, geldt:

$$V_{L1\_ON} \approx V_{\text{voeding}} - V_{\text{LED}} - I_{\text{LED}} \times R4 = 12 - 3,2 - 0,35 \times 1 = 8,45 \text{ V}$$

En voor de spanning over L1 als T1 uit is:

$$V_{L1\_OFF} \approx V_{\text{LED}} + V_{D1} = 3,2 + 0,5 = 3,7 \text{ V}$$

De verandering van de stroom door L1 gedurende de uit-tijd:

$$\Delta I_{L1} \approx V_{L1\_OFF} \times t_{\text{OFF}} / L1 = 3,7 \times 7 \times 10^{-6} / 0,003 = 8,6 \text{ mA}$$

Optisch flikkerpercentage FP [3] (aannemend dat de helderheid (*luminous output*) van de LED lineair met  $I_{\text{LED}}$  varieert):

$$FP = (LO_{\text{MAX}} - LO_{\text{MIN}}) / (LO_{\text{MAX}} + LO_{\text{MIN}}) \approx (350 - 341,4) / (350 + 341,4) = 1,2\%$$

Bij dimmen tot 1/2 en 1/5 van het maximale vermogen wordt dat:

$$FP_{50} \approx (175 - 166,4) / (175 + 166,4) = 2,5\%$$

$$FP_{20} \approx (70 - 61,4) / (70 + 61,4) = 6,5\%$$

De aan-tijd van de converter bedraagt:

$$t_{\text{ON}} \approx L1 \times \Delta I_{L1} / V_{L1\_ON} = 0,003 \times 0,0086 / 8,45 = 3,1 \mu\text{s}$$

De frequentie van de converter tenslotte is:

$$f = 1 / (t_{\text{ON}} + t_{\text{OFF}}) = 106 / (7 + 3,1) = 99 \text{ kHz}$$

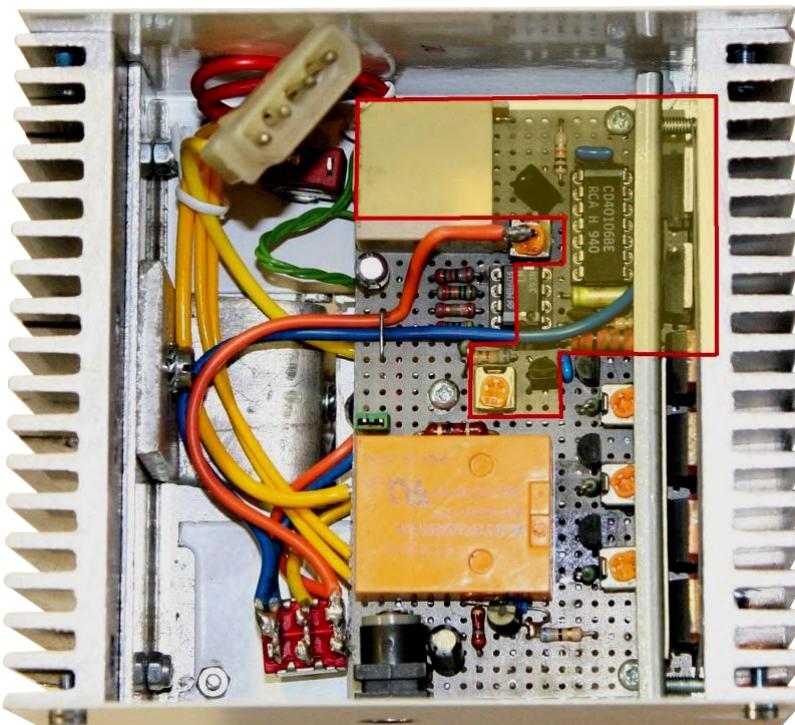
Met T1 uitgeschakeld neemt de spanning over R4 af en klappt de uitgang van de comparator om: C1 wordt geladen. Zodra de spanning over C1 de schakeldrempel van IC2A bereikt, wordt T1 weer ingeschakeld en de cyclus begint opnieuw. De spanning op de loper van R7 correspondeert met de piekstroom door L1. De tijdconstante van R3/C1 representeert een constante uit-tijd en bepaalt zo de rimpel van de LED-stroom (de uit-tijd is immers het interval gedurende welke de stroom afneemt).

R7 regelt de LED-stroom vanaf ongeveer 0 tot een maximumwaarde die met instelpot R6 wordt ingesteld. Voor 1-W vermogens-LED's dient dit maximum 0,35 V te bedragen, wat resulteert in piekstromen van maximaal 0,35 A (aangezien R4 een waarde van 1 Ω heeft).

Het is mogelijk om meerdere LED's in serie op de uitgang aan te sluiten zolang de doorlaatspanning van de LED's lager blijft dan de aangelegde voedingsspanning (vergeet niet dat we met een buck-converter te maken hebben). Wanneer 15 V niet genoeg is, kan het knooppunt van D1 en L1 aan een separate, hogere voedingsspanning worden gelegd (die natuurlijk gescheiden moet blijven van de stuurschakeling).

De schakelfrequentie bedraagt ongeveer 100 kHz (zoals wordt bepaald door L1, R3, C1 en de voedingsspanning). Aan de huidige stand van de techniek gemeten is dit vrij weinig. Een hogere frequentie is mogelijk met een snellere comparator en een 'stijvere' gate-driver, maar in dat geval is het beter om een geïntegreerde versie van deze schakeling te gebruiken. Een voorbeeld is de ZXLD1350 [2], een complete 1-chip controller die bij een hogere frequentie werkt. Een hogere frequentie betekent ook een kleinere waarde voor L1, resulterend in een bijzonder compacte implementatie.

D1 en T1 zijn niet kritisch. De typen die in het schema vermeld zijn, had de auteur toevallig liggen. Voor T1 kan elke N-kanalen MOSFET worden gebruikt met een  $R_{DSon}$  kleiner dan 0,1 Ω, een  $I_{Dmax}$  van meer dan 2 A en een  $V_{DSmax}$  van minstens 50 V. D1 is een Schottky-diode met een  $I_{Fmax}$  van tenminste 2 A en een  $V_{RRM}$  van minstens 50 V. Koellichamen zijn niet nodig. Wanneer u de schakeling opbouwt, moet u ervoor zorgen dat alle massa-aansluitingen naar één sterpunkt (bij de massa-aansluiting van R4) worden gevoerd.



Figuur 4. Het rood omkaderde deel is de door de auteur opgebouwde dimmer, als onderdeel van een schakeling die twee 1-W LED's uit drie Li-Ion cellen voedt; de koellichamen zijn deel van een (lineaire) zonnecel-lader.

## Weblinks

- [1] Pulsbreedtemodulatie:  
[https://en.wikipedia.org/wiki/Pulse-width\\_modulation](https://en.wikipedia.org/wiki/Pulse-width_modulation)
- [2] Datasheet ZXLD1350:  
[www.diodes.com/assets/Datasheets/ZXLD1350.pdf](http://www.diodes.com/assets/Datasheets/ZXLD1350.pdf)
- [3] Flikkerpercentage:  
[www.energy.gov/sites/prod/files/2015/05/f22/miller%2Blehrman\\_flicker\\_lightfair2015.pdf](http://www.energy.gov/sites/prod/files/2015/05/f22/miller%2Blehrman_flicker_lightfair2015.pdf)

Advertentie



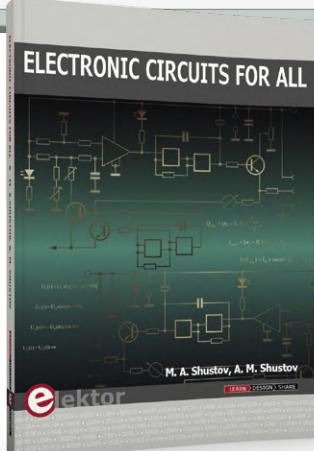
**MOUSER**  
ELECTRONICS<sup>®</sup>

**TEXAS**  
INSTRUMENTS



## IN DE STORE

→ Boek (Engelstalig): Electronic circuits for all  
[www.elektron.nl/electronic-circuits-for-all](http://www.elektron.nl/electronic-circuits-for-all)



**Eén adres  
voor uw hele  
stuklijst**

Nieuwste en breedste  
selectie elektronische  
componenten op voorraad

# Elektor Labs Pipeline

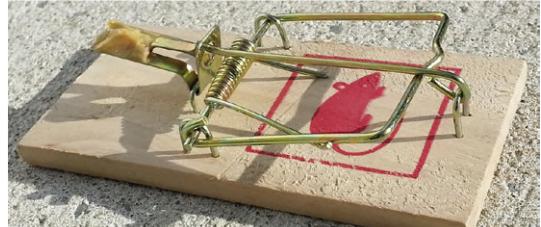


We presenteren weer een greep uit de interessante projecten op de Elektor Labs-website. Sommige lijken op het eerste gezicht bizar, maar voor alle zijn veel (en soms onverwachte) toepassingsgebieden en/of gebruikersgroepen te bedenken.

**Rijk en beroemd worden? Post uw projecten @ Elektor Labs**

## Hang uw muizenval aan het IoT

Nieuwe technologieën hebben vaak overwachte toepassingen. Neem bijvoorbeeld LoRa, Sigfox en aanverwante IoT-zaken. Iedereen is bezig met IoT-thermometers en -luchtvochtigheidssensoren voor thuis, maar het echte probleem is dat je wilt weten of er knaagdierde ergens onderin een groot gebouw of fabriekscomplex in een val is gelopen. Zodat de conciërge geen tijd kwijtraakt aan het controleren van lege vallen.



@ Elektor Labs: [www.elektormagazine.com/labs/1761](http://www.elektormagazine.com/labs/1761)

## Ingangsregelaar voor schakelende voeding

Labvoedingen met vermogenstrafo's zijn zwaar en duur, vooral als ze een paar honderd watt moeten leveren. Bovendien zijn ze ouderwets... Tegenwoordig kunt u met schakelende configuraties 250W-voedingen ontwerpen met een ingangsbereik van 85..265 V<sub>AC</sub> netspanning die toch minder dan 500 gram wegen. Dit ontwerp is zelfs *open source*.



@ Elektor Labs: [www.elektormagazine.com/labs/1820](http://www.elektormagazine.com/labs/1820)

## Robuuste doorgangstester met instelbare drempel

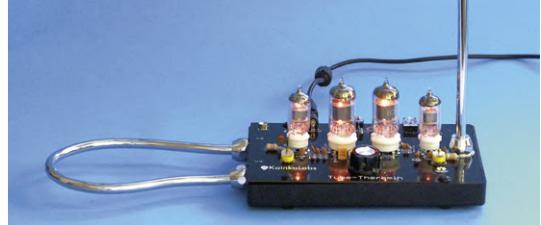
De meeste zelfbouw-doorgangstesters bestaan slechts uit een batterij plus een zoemer of misschien alleen een LED. Dit ontwerp maakt echter gebruik van een Wheatstone-brug. Het voordeel is dat de detectiedrempel instelbaar is van 1...20 Ω. En daar blijft het niet bij. Dit ontwerp kan de weerstand van de meetkabels compenseren en is ook beveiligd tegen mensen die vergeten om het te testen apparaat uit te schakelen.



@ Elektor Labs: [www.elektormagazine.com/labs/1823](http://www.elektormagazine.com/labs/1823)

## Doe als Leon, en bouw een buizen-Theremin

Toen Leon Theremin zijn inmiddels beroemde muziekinstrument bouwde, had hij uitsluitend buizen tot zijn beschikking. Tegenwoordig kunnen we kiezen tussen analoge en digitale technieken en tussen BJT's, FET's en buizen, of zelfs een combinatie daarvan. Ter gelegenheid van de 99<sup>ste</sup> verjaardag van de Theremin presenteren we een versie met uitsluitend buizen, om de puristen onder u te plezieren... ▶



@ Elektor Labs: [www.elektormagazine.com/labs/1828](http://www.elektormagazine.com/labs/1828)

# Het SCCC-project (4)

## softcore-processor en C-compiler voor zelfbouw

Martin Oßmann (Duitsland)

In dit deel van de serie houden we ons bezig met de uitvoer van analoge signalen. We maken daarbij gebruik van sigma-delta-DAC's en interrupts.

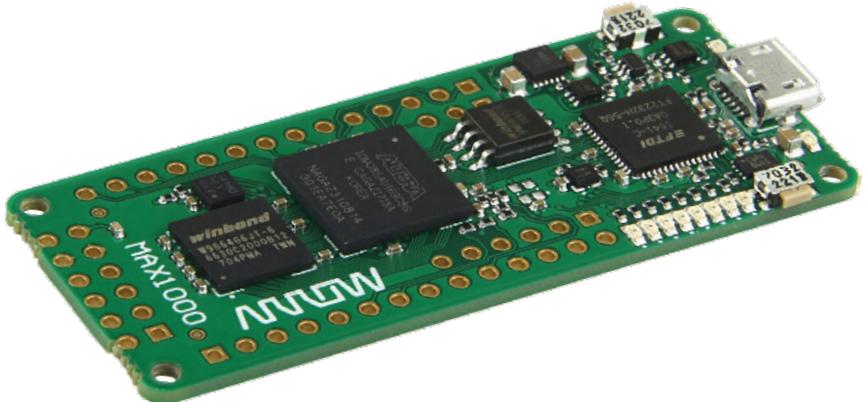
In het vorige deel van de serie [3] hebben we gezien welke periferie-eenheden onze zelfbouw-CPU heeft. Dat zijn onder meer een pulsbreedtemodulator (PWM), een A/D-converter, een GPIO-interface en *last but not least* een gebruikers-UART. We hebben in de Verilog-code al in- en uitvoerkanalen toegewezen aan deze eenheden. Ook hebben we gezien, hoe de gebruiker deze periferie met additionele Verilog-commando's kan verbinden met de pennen van de FPGA om er externe bouwstenen zoals een versnellingssensor mee uit te lezen (of aan te sturen). En tenslotte hebben we gekeken hoe de periferie wordt aangesproken in het C-programma dat onze processor uitvoert. Het is beslist aan te raden om het geheel nog een keer te bekijken, voordat we een volgend interessant experiment gaan uitvoeren.

### Experiment 6

In experiment 6 van onze cursus leren we sigma-delta-DAC's en interrupts gebruiken. We beginnen met de DAC's. De MAX10-FPGA heeft zelf geen DAC-uitgangen. We zouden die, zoals bij veel microcontrollers, met PWM kunnen realiseren. Maar in plaats daarvan gebruiken we DAC's die volgens het sigma-delta-principe werken.

In de toplevel-module instantiëren we twee modules van het type `firstOrderSigmaDeltaDac1v01`, die hun signalen naar de pennen PMOD1 en PMOD2 van de FPGA sturen. De data is 8 bit breed en wordt met het `OUTA`-commando gekoppeld met de kanaalnummers 19 en 20 (de eerder genoemde periferie-eenheden bezetten alleen de kanalen 0...7 en 18):

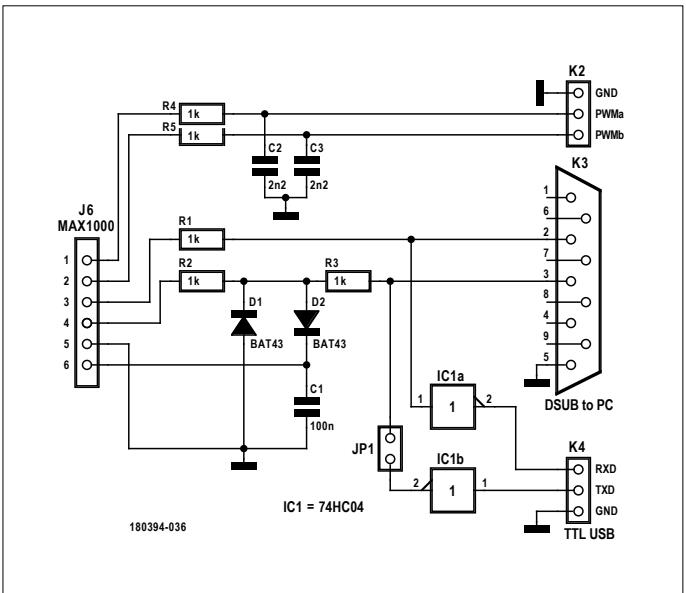
```
reg[8-1:0] aDacSignal ;
reg[8-1:0] bDacSignal ;
firstOrderSigmaDeltaDac1v01 #(.MSBI(7)) DACa
  (clk_100, aDacSignal, PMOD1) ;
firstOrderSigmaDeltaDac1v01 #(.MSBI(7)) DACb
  (clk_100, bDacSignal, PMOD2) ;
```



```
always @(posedge sCclk) begin
  if(outAstrobe) begin
    if( outAchannel==0) begin ... end
    else if( outAchannel==19) begin aDacSignal
      <=outA[8-1:0] ; end
    else if( outAchannel==20) begin bDacSignal
      <=outA[8-1:0] ; end
  end
end
```

De sigma-delta-code zelf ziet er als volgt uit:

```
module firstOrderSigmaDeltaDac1v01 #(parameter
  MSBI=7)
  (input          clk ,
   input [MSBI:0] DACsigIn ,
   output reg     DACout
  );
  reg [MSBI:0]   DACin;      // DAC input (excess
                           2**MSBI)
  reg [MSBI+2:0] DeltaAdder; // Output of Delta adder
  reg [MSBI+2:0] SigmaAdder; // Output of Sigma adder
  reg [MSBI+2:0] SigmaLatch; // Latched output of Sigma
                           adder
  reg [MSBI+2:0] DeltaB;      // B input of Delta adder
  always @ (SigmaLatch)           DeltaB = <<
                           (MSBI+1);
  always @ (DACin or DeltaB)       DeltaAdder = DACin
    + DeltaB;
  always @ (DeltaAdder or SigmaLatch) SigmaAdder =
    DeltaAdder + SigmaLatch;
```



Figuur 1. Uitvoer van de signalen.

```
always @(posedge clk) begin
    DACin <= DACsigIn ;
    SigmaLatch <= SigmaAdder;
    DACout <= SigmaLatch[MSBI+2];
end

endmodule
```

In vergelijking met een PWM-DAC genereren de sigma-delta-DAC's minder laagfrequente ruis. In **figuur 1** is te zien hoe we de hardware aansluiten. In de C-software gebruiken we weer OUTA-commando's in een passende functie. Die wacht in dit programma steeds tot de interruptroutine de variabele mail op 1 zet. De variabele wordt dan teruggezet, en de functieparameters x en y worden uitgevoerd via kanaal 19 en 20.

```
#asm
SigmaDeltaDacA EQU 19
SigmaDeltaDacB EQU 20
#endasm
```

```
outDacs(int x, int y) {
    while(mail==0){ }
    mail=0 ;
    x ;
#asm
    OUTA SigmaDeltaDacA
#endasm
    y ;
#asm
    OUTA SigmaDeltaDacB
#endasm
}
```

De routine `outDacs` is vrij kort, dus we kunnen volledig weergeven wat de compiler daarvan maakt:

```
outDacs      EQU $ // Beginn der Funktion outDacs
// auf Adresse 2 relativ zum Stack liegt Parameter x
// auf Adresse 1 relativ zum Stack liegt Parameter y
// auf Adresse 0 relativ zum Stack liegt die
// Rückkehradresse

__LBL00000   EQU $ ; whileStackBase=0
                LDPUSH    @ __mail
// Wert von mail auf den Stack
                LDI       0 ; hex=00000000
// Wert von 0 in R0
                POP      R1
// R1 = Wert von mail vom Stack
                CMPEQ
// Vergleich auf ==
                JFALSE   __LBL00001
// wenn nicht true aus Schleife springen
                JMP     __LBL00000
// Schleife weiter verfolgen
__LBL00001   EQU $
                LDI       0 ; hex=00000000
// Wert 0 nach R0
                ST       @ __mail
// 0 aus R0 auf mail speichern
                GETLCL  2
// lokale Variable x ist 2 relativ vom Stack zsp=0
                OUTA     SigmaDeltaDacA
// Wert an DACa ausgeben
                GETLCL  1
// Wert y nach R0
                OUTA     SigmaDeltaDacB
// Wert an DACb ausgeben
                RET
// Funktionsreturn
```

De commando's LDPUSH en GETLCL zijn toegevoegd door de optimizer, en wel met het volgende idee:

van

```
LD      @ __mail
PUSH   R0
```

maakt hij

```
LDPUSH  @ __mail
```

en van

```
LDI    2
IADD  SP
LDIND @R0
```

maakt hij

```
GETLCL 2
```

Deze optimalisatie maakt het programma korter en sneller. En dan nu de interrupts.

## Interrupts

De sCCCP-CPU biedt plaats aan vier interrupts, waarvan er maar twee gebruikt worden (`RTCinterruption` en `uartRXready-interrupt`). De CPU doet de interruptverwerking in de fetch-fase. Normaal gesproken wordt daar het huidige commando opgehaald. Maar als er een interrupt is (`selectedIRQ != 0`), dan gebeurt er iets anders. De program counter wordt bewaard op de stack. Dan wordt de program counter op de waarde  $8 * \text{interrupt-nummer}$  gezet. Voor interrupt IRQ1 wordt dus naar 8 gesprongen, voor interrupt 2 naar 16, enzovoort. Het schrijven van de oude PC-waarde op de stack gebeurt tegelijk met de volgende fetch-fase, die meteen aansluit op de huidige (interrupt-)fetch-fase. In de Verilog-code van de CPU ziet dat er als volgt uit:

```
if(selectedIRQ) begin
    writeAddress <= sCregSP-1'b1 ;
    writeValue   <= sCregPC ;
    writePending <= 1 ;
    sCregSP <= sCregSP-1'b1 ;
    sCregPC <= selectedIRQ << 3 ;
    sCstate <= sCstateFetch ;

    if(selectedIRQ==1) begin irq0reg<=0 ; end
    ...
end
```

Om nu interruptgestuurd naar een C-routine te gaan, programmeren we voor de RTC-interrupt de volgende code:

```
#asm
RTCintVector      EQU  8

ORG      RTCintVector
PUSH    R0
PUSH    R1
CALL    RTCinterruption
POP     R1
POP     R0
RET
#endasm

...
int mail ;

RTCinterruption(){
    mail=1 ;
}
```

In de eigenlijke interruptroutine wordt gewoon de variabele `mail` op 1 gezet. Wat nu nog ontbreekt, is het volgende: in het hoofdprogramma wordt `RTCtimerTOP` op 50 gezet, dat wil zeggen om de 50 klokpulsen wordt een interrupt gegenereerd. En met `setCPUflags (RTCintEna)` wordt de RTC-interrupt vrijge-

geven. Dan wordt de functie `doSamples` aangeroepen, die achtereenvolgens 32 waarden naar de beide DAC-kanalen stuurt.

```
#asm
CPUflagsChannel    EQU  6
RTCtimerTopChannel EQU  7
#endasm

doSamples(){
// written by SinTabGen1v01.lua
// 128+127*sine(Xi)
while(1){
    outDACs( 128 , 0 ) ;
    outDACs( 153 , 8 ) ;
    ....
    outDACs( 103 , 248 ) ;
}
}

#define RTCintEna 1

main(){
    setRTCtimerTop(50) ;
    setCPUflags( RTCintEna ) ;
    doSamples() ;
}
```

Het eigenlijke experiment is weer vrij gemakkelijk uit te voeren: we synthetiseren het Quartus-project `experiment6.qpf` en laden de bitfile in de FPGA. Dan compileren we de C-code `experiment6.c` en laden die met de processing-uploader in de CPU. Daarna zou op PWM-uitgang 1 een sinus en op uitgang 2 een zaagtand te zien moeten zijn.

## Experiment 7: sampling-scope

Onze experimenten hebben tot nu toe vooral als demonstraties gediend. Maar nu gaan we iets proberen dat meer op een realistische toepassing lijkt. We willen data registreren met de A/D-converter in de MAX10-chip en die via de seriële interface naar de PC zenden. De PC moet de data dan weergeven (oscilloscoop-functie). Natuurlijk kunnen we ook met een FFT op de PC het spectrum van het signaal bepalen (spectrumanalyser-functie).

We gaan op drie niveaus werken om deze toepassing te realiseren. Op FPGA-niveau breiden we onze CPU uit, zodat de FPGA 8192 samples in een buffer schrijft, waarbij de ADC op maximale samplesnelheid van 1 MSamples/s draait. Daarna gaan we de buffer met een CPU-programma uitlezen en de data naar de PC sturen. En als derde stap laten we de PC de waarden weergeven en/of het spectrum berekenen.

Het inlezen van de waarden in de FPGA gaat als volgt: de schrijfpointer `traceWritePtr` speelt een centrale rol. In rust staat hij op 8191. Als hij kleiner is dan 8191, worden er waarden ingelezen, tot hij uiteindelijk weer op 8191 staat. Om het inlezen te starten, zetten we de pointer gewoon op nul, dan worden daarna 8192 waarden in de buffer `traceBuffer` geschreven. We stellen de schrijfpointer weer in met een `OUTA-commando`. De schrijfpointer kan ook gelezen worden via kanaalnummer 32,

zodat we kunnen testen of alle waarden al zijn ingelezen. Om data uit de buffer te lezen, gebruiken we een leespointer `traceReadPtr`, die we kunnen instellen met een `OUTA`-commando op kanaalnummer 33. De waarde in het buffergeheugen op dat adres wordt ingelezen en is beschikbaar via het commando `INPA`-commando op kanaal 33:

```
parameter traceLength = 8192 ;
parameter traceLengthWidth = 13 ;
reg [12-1:0] traceBuffer[0:traceLength-1] ;
reg [traceLengthWidth-1:0] traceWritePtr ;
reg [traceLengthWidth-1:0] traceReadPtr ;
reg trigger ;

// End Of Conversion pulse eocPulse kommt vom ADC

always @(posedge sCclk) begin
    if (outAstrobe) & (outAchannel==32) begin
        trigger<=1 ; end
    if (eocPulse) begin
        if(trigger) begin
            traceWritePtr<=0 ;
            trigger<=0 ;
        end
        if(traceWritePtr != traceLength-1) begin
            traceBuffer[traceWritePtr] <= ADCval ;
            traceWritePtr <= traceWritePtr+1 ;
        end
    end
end

always @(posedge sCclk) begin
    if(outAstrobe) begin
        if( outAchannel==33) begin traceReadPtr<=outA ;
        end
    end
end

assign inpA=
    (inpAchannel==0) ? { 32'h12341234 } :
    ....
    (inpAchannel==32) ? traceWritePtr :
    (inpAchannel==33) ? traceBuffer[traceReadPtr] :
    32'h12341111 ;
```

Dat zijn bij elkaar nog geen 40 regels in Verilog. Het komt bij het verwerken van signalen vaak voor, dat de FPGA-functionaliteit door een kort codefragment kan worden gerealiseerd. Nu moeten we de waarden uit de buffer naar de PC sturen. Het is nogal omslachtig om dat ook in Verilog op FPGA-niveau te programmeren. Dat gaat veel gemakkelijker met onze softcore-CPU.

Het seriële protocol tussen de MAX1000-kaart en de PC moet er als volgt uitzien: als de PC een x zendt, start van de FPGA het bemonsteringsproces en leest de ADC uit tot alle 8192 waarden in de `traceBuffer` staan. Dan haalt de sCCCP-CPU de data uit de FPGA-buffer en stuurt die serieel met 115200 bit/s naar de PC.

Daarna wordt nog een checksum gestuurd. In ons voorbeeldprogramma `experiment7.c` kunt u trouwens ook zien, hoe we data kunnen inlezen met de sCCCP-CPU. De code ziet er als volgt uit:

```
doCommand(){
    int c ;
    c=getCharPolling() ;
    if(c=='x') { execute() ; return ; }
    putChar('?') ;
}

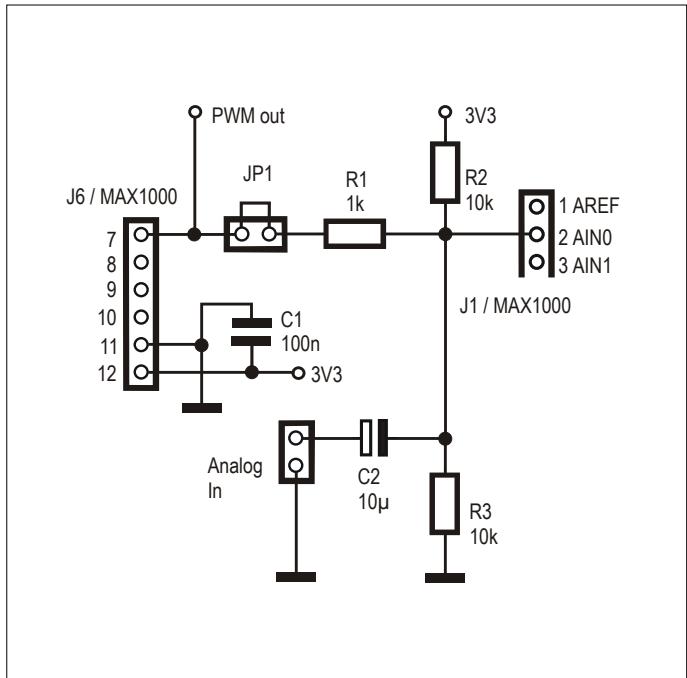
execute(){
    int data,k,i,chkSum ;
    traceTrigger() ;
    // reset traceWritePtr=0
    while( getTraceWritePtr() !=traceLength-1) { }
    // wait until end reached
    chkSum=0 ;
    putChar('[') ;
    // opens frame
    for(k=0 ; k<frameLength ; k++){
        // loop over the values
        setTraceReadPtr(k) ;
        data=getTraceData() ;
        // get traceData[k]
        word12Out(data) ;
        // output 3 nibbles hex=12Bit
        chkSum=chkSum+data ;
        // update checksum
        putChar('.') ;
        // . behind every value
        if( (k&0x1F)==0x1F ){ crlf() ; }
        // increase readability
    }
    word12Out(chkSum) ;
    // send chkSum to PC
    putChar(']') ;
    // closes frame
    crlf() ;
}
```

De analoge ingang AIN0 wordt aangesloten zoals in **figuur 2**. Eerst plaatsen we jumper JP1 en laten we de analoge ingang *Analog in* open. Zo bereikt het PWM-signalen de ADC. Later verwijderen we dan JP1 en sluiten het signaal aan op de AC-gekoppelde ingang *Analog in*.

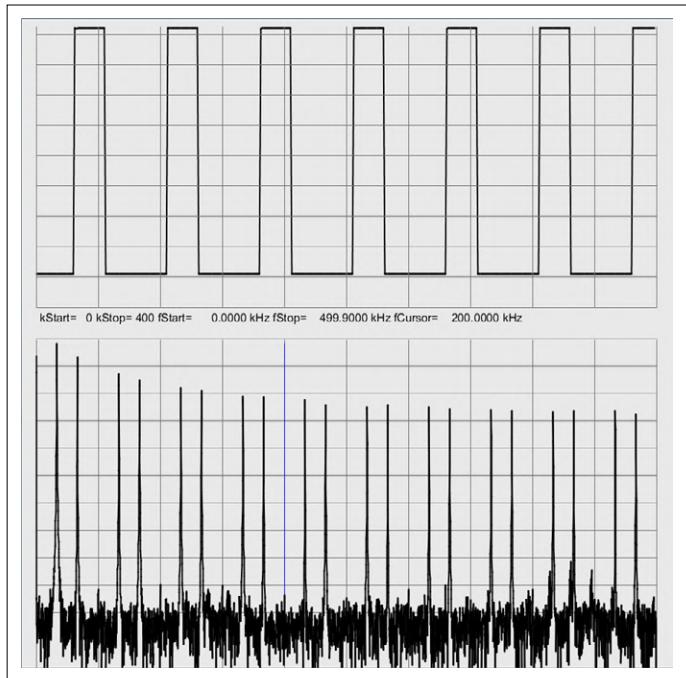
Het PWM-signaal wordt opgewekt met deze code:

```
setPWMtop(2000) ;
setPWMperiod(6000-1) ;
```

De periodeduur is 6000 maal de PWM-klokfrequentie van 100 MHz, dus 60  $\mu$ s. De *on-tijd* staat op 2000 cycli, dus 20  $\mu$ s. Nu hebben we nog een programma op de PC nodig om de gesampelde waarden op te halen en weer te geven. We gebruiken daarvoor we de Processing-sketch  
`C:/sCCCP/experiment7/Processing>ShowGraph1>ShowGraph1/`



Figuur 2. Schakeling aan de analoge ingang AIN0.



Figuur 3: Boven is de gesampelde data te zien, onder het spectrum.

*ShowGraph1.pde*. Voordat we die starten, moet u de derde regel (`String serialPort="COM2"`) nog aanpassen, zodat de COM-poort wordt gebruikt, die u met de user-poort van de MAX1000 hebt verbonden. We kunnen ook met een terminalprogramma testen of het goede programma op de MAX1000 draait. Als we een x (execute) naar de MAX1000 sturen, moet die antwoorden met een heleboel data in de vorm:

[804.842.....]

De processing-sketch voor het weergeven van de gemeten waarden genereert een plaatje zoals in **figuur 3**. De bovenste helft geeft de gesampelde data weer. De onderste helft toont het spectrum.

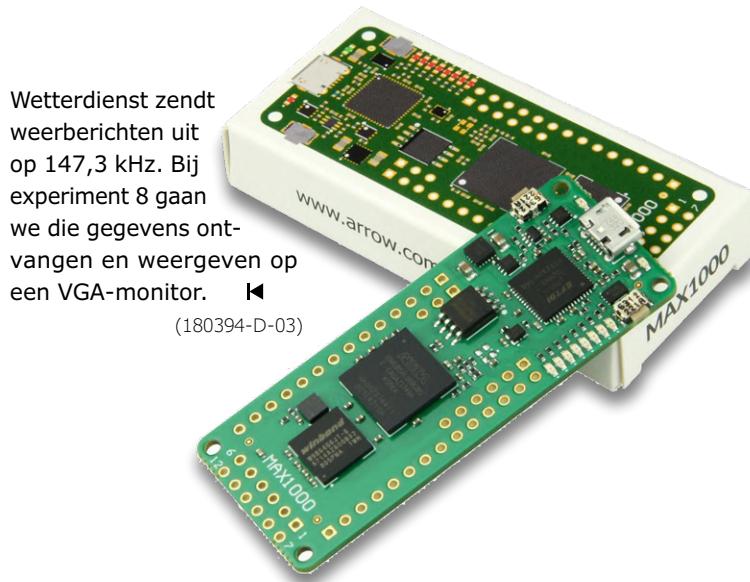
Met deze toepassing hebben we al bijna een digitale oscilloscoop of een spectrumanalyser gebouwd. Wat nog ontbreekt, is het starten van de registratie met een triggersignaal. En natuurlijk zouden we ook nog een fatsoenlijke GUI moeten programmeren. Om het helemaal mooi te maken, zouden we een veel snellere A/D-converter kunnen gebruiken om hoogfrequenter signalen te kunnen weergeven. En als we geen ADC-samples zouden registreren, maar de waarden op de digitale ingangen, dan hadden we een logic analyzer gerealiseerd.

In het volgende en laatste deel van de serie gaan we onze hardware nog verder opvoeren: de zender DDH47 van de Deutsche

■
IN DE STORE

→ MAX1000 FPGA-ontwikkelkaart  
[www.elektor.nl/max1000](http://www.elektor.nl/max1000)

→ E-Book: "Microprocessor Design Using Verilog HDL"  
[www.elektor.nl/verilog](http://www.elektor.nl/verilog)



## Weblinks

- [1] Het SCCC-project (1), Elektor maart/april-2019: [www.elektormagazine.nl/180394-03](http://www.elektormagazine.nl/180394-03)
- [2] Het SCCC-project (2), Elektor mei/juni-2019: [www.elektormagazine.nl/180394-B-03](http://www.elektormagazine.nl/180394-B-03)
- [3] Het SCCC-project (3), Elektor juli/augustus-2019: [www.elektormagazine.nl/180394-C-03](http://www.elektormagazine.nl/180394-C-03)
- [4] Projectpagina bij dit artikel: [www.elektormagazine.nl/180394-D-03](http://www.elektormagazine.nl/180394-D-03)

# LED-kweekbox

## assimilatieverlichting met 'horticulture' LED's van Würth Elektronik

Luc Lemmens, met medewerking van Würth Elektronik

Het gebruik van en onderzoek naar het stimuleren van plantgroei met kunstmatige lichtbronnen gaat terug tot de tweede helft van de 19e eeuw. De ontwikkeling liep gelijk op op met die van de gewone verlichting, die kan worden ingedeeld in drie hoofdstromen: gloeilampen, booglampen en gasontladingslampen.

En nu worden ook onze geliefde LED's voor tuinbouwdoel-einden toegepast. Hier presenteren we een plantvriendelijke LED-lichtbron die weinig te wensen overlaat.



De LED-theorie achter assimilatielampen is complexer dan je zou denken en valt buiten het bestek van dit artikel. Gelukkig is hierover alles te vinden in de *application notes* ANO002, ANO003, ANO004 op de website van Würth Elektronik [1]. We beperken ons hier tot de belangrijkste voordelen van LED's voor kweekdoeleinden, vergeleken met de conventionele voorgangers.

In essentie gaat het bij de groei en ontwikkeling van planten niet alleen om de intensiteit, maar ook om het spectrum van het gebruikte licht. We noemen enkele karakteristieke bandbreedtes en de bekende effecten daarvan:

- Rood licht (630...660 nm) is de belangrijkste stimulans van fotosynthese en voor de groei van stengels. Deze golflengte kan ook de bloei, de kiemrust en de zaadontkieming reguleren.
- Blauw licht (400...520 nm) is een voor fotosynthese belangrijke golflengte, maar moet zorgvuldig worden gestuurd en gemengd met andere frequenties, omdat een teveel van deze kleur de groei belemmert. Deze golflengte is ook werkzaam bij de regulering van de chlorofylconcentratie, laterale knopgroei en bladdikte.
- Ver rood licht (720...740 nm), dat al in het IR-spectrum ligt, beïnvloedt het kiemen en kan de bloeitijd van planten bekorten, maar ook de stengellengte vergroten als onderdeel van een effect dat 'schaduwvermijdingsreactie' wordt genoemd.
- Groen licht (500...600 nm) is lang

beschouwd als irrelevant voor plantengroei. Recent onderzoek laat echter zien dat planten die in de schaduw staan als onderdeel van hun schaduwvermijdingsreactie just voor deze golflengte gevoelig zijn.

- UV-licht (280...400 nm) bevindt zich nog in de experimentele fase bij plantcultivatie. Hoewel deze golflengte mutageen (schadelijk voor DNA) is, blijken sommige planten (bijvoorbeeld sla en tomaat) veel resistenter tegen deze golflengten. UV-licht zou verantwoordelijk kunnen zijn voor het genereren van bepaalde beschermende moleculen zoals antioxidanten en fenolen, die belangrijk zijn voor de menselijke voeding.

In tegenstelling tot langer bestaande lichtbronnen kunnen LED's voor een nauwkeurig gedefinieerde spectrale bandbreedte worden geproduceerd. Hierdoor zijn we in staat het spectrum van een assimilatielamp (die verschillende kleuren LED's bevat) af te stemmen op de behoeften van specifieke planten. Würth Elektronik levert de 'WL-SMDC SMD Mono-colour Ceramic LED Waterclear' serie LED-lampen. Deze serie bestrijkt ook golflengten van 450 nm (ver blauw), 660 nm (rood) en 730 nm (ver rood), die passen bij de absorptie-spectra van fotosynthetische pigmenten. Deze drie LED-varianten zijn onderdeel van onze kweekbox, samen met enkele koelwitte (6000 K) LED's. Om de intensiteit van elk van de vier kleurkanalen naar keuze onafhankelijk van elkaar of gezamenlijk vanaf smartphone, tablet of PC te kunnen regelen, beschikt onze LED-kweekbox over

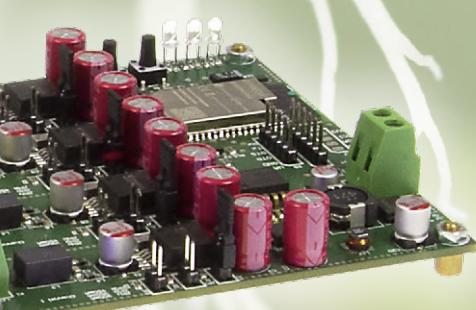
### PROJECT-INFO

	LED	PWM	ESP32
	groeilamp		
	assimilatielamp		
	kweken		
	beginners		
	gevorderden		
	experts		
	ca. 2 uur		
	Arduino IDE (optioneel)		
	ca. € 165		

een ESP32 met een speciaal ontworpen webinterface. Hierdoor kunt u naar hartlust experimenteren met verschillende lichtrecepten in uw kweekbox. Het gebruik van LED's als assimilatieverlichting biedt overigens nog veel meer voordelen, zoals afmetingen, rendement en levensduur.

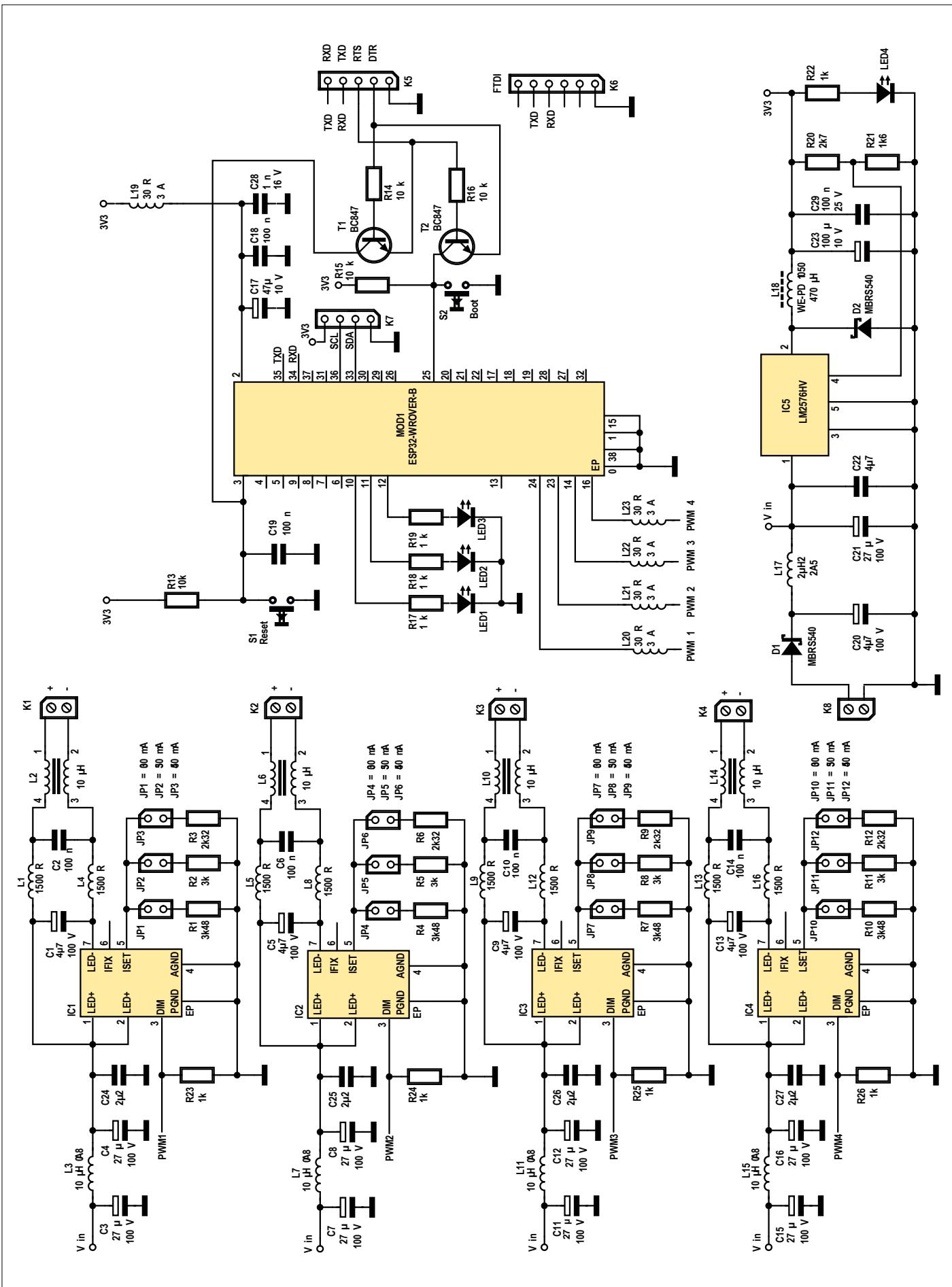
#### Driverprint

Voor het grootste deel van de schakeling hebben we dankbaar gebruik gemaakt van een referentie-ontwerp van Würth Elektronik met een vierkanaals PWM LED-driver. Luc Lemmens en Matthias

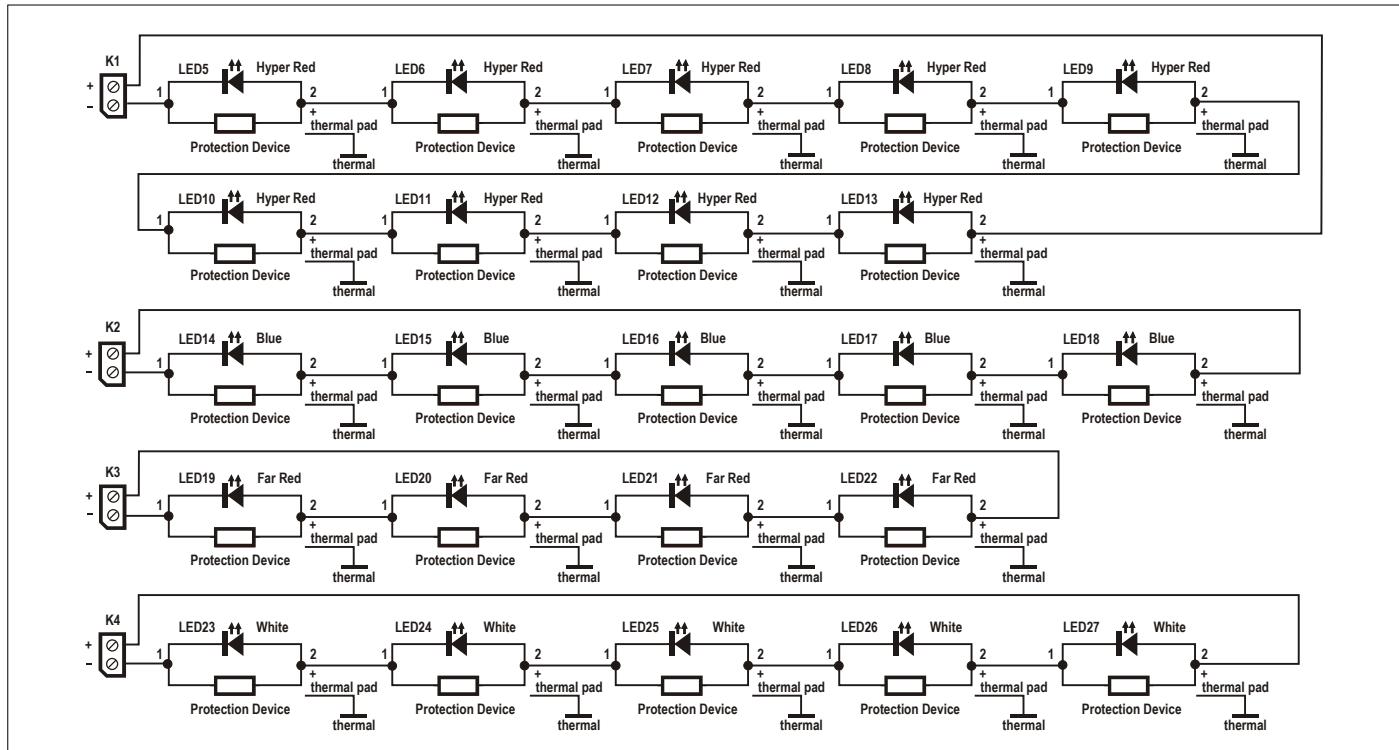


#### Eigenschappen

- 4 kleurkanalen: rood, ver rood, blauw, wit
- Würth Elektronik WL-SMDC high brightness LED's
- 300 mA / 350 mA / 450 mA LED-stroombegrenzing
- 4 PWM-drivers
- ESP32-bestuurd
- Volledig instelbare kleurmenging en helderheid
- Timer voor dag/nacht-cyclus
- Verlicht oppervlak met een straal van 55 cm
- Besturing via WiFi (webinterface / MQTT)
- Werkt met elke 24V/50W-spanningsbron
- Volledig bestukte driver- en LED-print verkrijgbaar



Figuur 1: Schema van de driverprint voor de kweekbox, met de ESP32-module als spin in het web.



Figuur 2: Schema van de LED-print.

Claussen van Elektor Labs vervangen de PIC-controller en Bluetooth-module door de populaire ESP32-module ESP32-WROVER-B van Elektor. Luc besloot een LM2576 DC/DC-converter te gebruiken in plaats van de voedingsmodule van Würth Elektronik uit het originele ontwerp. Het resultaat van deze revisie is te zien in het schema van figuur 1. Het schema bevat alle spoelen en condensatoren die nodig zijn om – zelfs tijdens dimmen – te voldoen aan de EMC-voorschriften uit de normen EN55015 en CISPR32. Het herziene ontwerp van de Elektor/Würth Elektronik-schakeling met de vier drivers kwam probleemloos door de pre-compliance tests. Het schema en de print van dit de schakeling hebben we nagenoeg ongewijzigd overgenomen. De ESP32 echter, zorgde voor behoorlijke problemen met harmonischen bij 80 MHz en hoger. Niet echt verbazingwekkend, want 80 MHz is de bus-klokfrequentie van de ESP32. Daarom is een deel van de print opnieuw ontworpen, met een wat andere opzet van de massavlakken en toevoeging van een aantal chip-ferrietkralen.

Verder heeft het schema niet veel om het lijf. De ESP32-ROVER module vormt natuurlijk de ‘hersenen’ van het driverboard en genereert de PWM-signalen die de lichtintensiteit bepalen. De microcontroller neemt ook de gebruikersinterface

voor zijn rekening, alsmede de verbinding met het lokale WiFi-netwerk. De LM2576 stepdown converter (IC5) verzorgt de 3,3V-voedingsspanning voor de ESP32-module (MOD1); LED4 (blauw) signaleert dat de voeding is ingeschakeld. De ingangsspanning van de kweekbox mag maximaal 45 V bedragen, maar om de vermogensdissipatie op de print bij ingeschakelde LED’s te beperken, kunnen we beter beneden 24 V blijven.

De UART van de ESP is op K5 of op K6 beschikbaar voor seriële communicatie of om de ESP32 in bootloader-mode te programmeren. De aansluitingen DTR en RTS op K5 stellen de Arduino-programmeeromgeving in staat de ESP32 automatisch in de bootloader-modus te zetten via T1 en T2. Maar daarvoor is wel een USB/UART-interface nodig die deze twee handshake-signalen levert – bijvoorbeeld de Elektor FT232R USB/Serial Bridge/BoB of de FT231X BoB. Op K6 kan een standaard FTDI-kabel worden aangesloten, maar in dat geval – zoals u wellicht weet van eerdere ESP32-projecten in Elektor – zult uzelf de drukknoppen S1 en S2 moeten bedienen om de ESP te resetten en in de bootloader-modus te brengen. Als u de kant-en-klaar opgebouwde printen uit de Elektor-shop aanschaft, hoeft u zich niet druk te maken over het programmeren van de firmware, hoewel dit ‘draadloos’ (OTA, Over The Air) kan

gebeuren. Inderdaad, er is geen kabel of interface nodig om een update te installeren!

De I<sup>2</sup>C bus-signalen SDA en SCL kunnen (met 3,3 V amplitude) op K7 worden aangesloten voor toekomstige uitbreidingen. Eventueel zijn pullup-weerstanden nodig om een apparaat aan te sluiten, evenals 5V/3,3V niveau-omzetters, want de ESP32 verdraagt geen 5 V!

De PWM-signalen van de ESP32 gaan naar de DIM-ingangen van LED-drivers IC1...IC4. De uitgangsstroom wordt beperkt tot 300 mA, 350 mA of 450 mA, afhankelijk van de bij elke driver afzonderlijk driver geplaatste jumpers.

LED1 licht op als de ESP32 met een WiFi-netwerk is verbonden en in Station-modus staat; deze LED knippert in Access Point-modus. LED2 (rood) licht op als een OTA-firmware-update wordt uitgevoerd. De gele LED3 tenslotte licht op als een MQTT-verbinding wordt gemaakt.

### LED-print

Dit lijkt misschien het minst gecompliceerde deel van dit project, maar de eenvoud van het schema van figuur 2 bedriegt! Om te beginnen moesten we beslissen hoeveel LED’s van elke kleur we in onze kweekbox wilden hebben. Elektor Labs heeft op dit gebied geen enkele ervaring, dus vertrouwden we op de kennis en expertise van Würth Elek-



Figuur 3. Mean Well  
24V/2,2A schakelende  
voedingsmodule  
LRS-50-24.

tronik om een idee te krijgen van een werkbare mix voor onze relatief kleine kweekbox. De op basis hiervan ontworpen print zou moeten volstaan om een cirkelvormig gebied met een straal van ca. 55 cm te dekken; de randgebieden worden daarbij iets minder belicht dan het centrum.

Ten tweede: LED's met een hoge lichtopbrengst worden warm. Würth Elektronik heeft de technologie en faciliteiten in huis om printen te ontwerpen waarop LED's zo koel mogelijk blijven en niet node-loos verouderen. De methode bestaat in essentie uit het lijmen van een dunne print op een koelplaat – meer hierover in het kader **Warmtehuishouding**. Op zich een prima oplossing, maar wel duurder dan een simpele print.

Om dit project betaalbaar te houden, kozen we voor een dunne (0,5 mm, oef!) print met grote kopervlakken aan beide zijden. De LED's uit de WL-SMDC-serie hebben elektrisch neutrale warmtepads die op het kopervlakken aan één kant van de print kunnen worden gesoldeerd. Deze kopervlakken zijn met thermische via's verbonden met de kopervlakken aan de andere zijde. Standaard staan alle jumpers op 300 mA voor alle vier kanalen. Bij deze waarde is nog geen koelplaat nodig, maar het licht is misschien een beetje weinig voor sommige planten.

## Bouwen en afregelen van de kweekbox

De kant-en-klare printen uit de Elektor-shop beperken uw eigen bijdrage tot het plaatsen van de jumpers, het maken van de onderlinge verbindingen en het aansluiten van een 24V/2,5A DC-voeding. Betaalbare kant-en-klare voedingen zijn volop verkrijgbaar. Wij werkten voor het prototype met een labvoeding, de EMC precompliance tests zijn uitgevoerd met een Egston-voeding type BI60-240250-E2. Mean Well heeft een kleine en elegante 24V/2,2A schakelende voeding (type LRS-50-24) die het prima doet (**figuur 3**).

Als u de print zelf volbouwt, dan hebt u minimaal soldeer pasta en een hete-lucht-soldeerstation nodig. Een kleine soldeerbout gaat eventueel ook nog, maar denk er dan aan de componenten in de juiste volgorde te monteren, omdat grotere componenten de toegang tot de pads van kleinere componenten kunnen belemmeren. Dus eerst de SMD-onderdelen. En monter de trough-hole onderdelen niet voordat de montage van de SMD's grondig is gecontroleerd!

## Warmtehuishouding

Omdat de prestatieverbetering en miniaturisering van vermogens-LED's nog steeds doorgaan, wordt de warmtedissipatie een steeds belangrijker aspect van LED-applicaties. Daarom moet een thermisch beheersconcept worden ontwikkeld om de warmtebelasting van onderdelen binnen de toegestane grenzen te houden. Het doel hiervan is de temperatuur van de LED-chip zo laag mogelijk te houden. Verticale warmtegeleiding door middel van zogenaamde thermische via's direct onder het te koelen oppervlak van de LED helpt in elk geval. Evenals horizontale warmtegeleiding via grote kopervlakken aan weerszijden van de print.

In de koelplaattechnologie van Würth Elektronik wordt de warmte via een zo groot mogelijk oppervlak overgebracht naar de aluminium koelplaat via een zelfklevende laag. Het bijzondere voordeel vergeleken met de IMS-techniek is de mogelijkheid multilayer-printen te maken.

Thermische via's zijn speciaal voor warmtegeleiding in de print gemaakte gaatjes. Idealiter worden deze direct onder de warmtebron geplaatst. De diameter is typisch 0,30 mm en de wanddikte van het koper in het gat is ten minste 25 µm. Maar ook grotere diameters komen voor.

Omdat thermische via's zijn gevuld en gesloten loopt het soldeer er niet doorheen, zodat onderdelen probleemloos op de kopereilandjes kunnen worden gesoldeerd. Een goede inschatting van de warmteweerstand en temperatuur in en op de print kunnen worden verkregen met simulaties of met een FEM-model.

Net als bij verticale warmtegeleiding wordt ook bij horizontale geleiding de warmte over een groter oppervlak van de print verdeeld. Een groot oppervlak is een belangrijke voorwaarde voor verticaal warmtetransport, vanwege minder goed warmtegeleidende materialen, bijvoorbeeld in de isolatielagen tussen print en koelplaat, of tussen print en omgeving.

Bekende methoden voor koeling door warmteverdeling zijn het verdelen van de warmte over koperlagen en montage van de print tegen een koellichaam.

Koelplaten van aluminium hebben hun diensten bewezen. Een gebruikelijke methode is deze samen te persen met een FR4-preparaat, of het gebruik van warmtegeleidende lijm. Het voordeel van die laatste manier is het dynamisch gedrag tijdens solderen. De verschillende uitzettingscoëfficiënten van aluminium en printplaat worden hierdoor beter gecompenseerd.

Met de technologie die door Würth Elektronik wordt gebruikt, wordt het koellichaam koud onder druk aan de printplaat gehecht. Deze hechting met behulp van warmtegeleidende lijm wordt TWINflex genoemd.

Idealiter bestaat de andere zijde van de print geheel uit koper. De warmte wordt eerst hierdoor verdeeld en vervolgens door de lijmplaat naar de koelplaat overgedragen. De thermische weerstand van deze combinatie wordt door de grotere koperlaag verkleind. Het doel is een minimale printdikte en een maximaal koelplaatoppervlak.

## Firmware programmeren en updaten

Op de kant-en-klare print uit de Elektor-shop is de ESP32 al geprogrammeerd en klaar voor gebruik. Als u de firmware van het systeem wilt aanpassen is installatie van de Arduino-IDE – met ondersteuning voor de ESP32 – vereist. Dit proces is uitvoerig beschreven in [2]. Vervolgens moet de firmware voor dit project worden gedownload van de Elektor Labs-website [3]. De commentaarregels in het begin van de sketch 'Firmware.ino' bevatten een lijst met de voor compilatie vereiste Arduino-libraries. Als u een USB-UART hebt met RTS- en

DTR-signalen, moet deze op K5 worden aangesloten. De Arduino-IDE is nu in staat de firmware- en sketchdata automatisch te uploaden. Als je een UART hebt zonder deze signalen (bijvoorbeeld een FTDI-kabel die op K6 is aangesloten), dan moet eerst met de drukknoppen S1 en S2 de ESP32 in bootloader-modus worden gebracht: houd S2 ingedrukt, druk S1 in en laat die weer los en laat tenslotte S2 los.

In de standaard-firmware is Arduino OTA geactiveerd, zodat flashen van het systeem mogelijk is via WiFi zonder USB-verbinding en zonder knoppen te hoeven indrukken. Maak verbinding met

de kweekbox in Access Point-modus of laat het apparaat verbinding maken met uw lokale netwerk. Start de Arduino IDE, open het Tools-menu en kies 'ESP32 ROVER module'. In het submenu 'Ports' ziet u een HC-LED-XX-XX-XX als één van de te kiezen poorten. Dat is de ESP32 van de kweeklamp. Nadat die is geselecteerd, kan de Upload-knop worden aangeklikt om de code naar de chip over te brengen. Als alles goed is gegaan, zal de chip na het uploaden opnieuw opstarten en is de nieuwe firmware operationeel. Maar let op: als je de firmware aanpast en de het Over The Air-updates onderbreekt, zul je een kabel moeten aansluiten om het



## ONDERDELENLIJST DRIVERPRINT

### Weerstanden:

(alle dikke film)

R1,R4,R7,R10 = 3k48, 1%, 0,125 W, 150 V

R2,R5,R8,R11 = 3 k, 1%, 0,125 W, 150 V

R3,R6,R9,R12 = 2k32, 1%, 0,1 W, 150 V

R13,R14,R15,R16 = 10 k, 5%, 0,1 W, 150 V

R17...R19,R22...R26 = 1 k, 5%, 0,1 W, 150 V

R20 = 2k7, 5%, 0,1 W, 150 V

R21 = 1k6, 1%, 0,125 W, 150 V

### Spoelen:

L1,L4,L5,L8,L9,L12,L13,L16 = EMI-onderdrukking,  
1500 Ω @ 100 MHz, 0805, WE-CBF

L2,L6,L10,L14 = SMD common mode line filter 10 μH,  
1,6 A, WE-SL2

L3,L7,L11,L15 = SMD vermogensspool, 10 μH, 800 mA,  
WE-PD2, 3521

L17 = SMD vermogensspool 2,2 μH, 2,5 A, WE-PD2, 4532

L18 = SMD vermogensspool, 470 μH, 600 mA, WE-PD 1050

L19,L20,L21,L22,L23 = ferrietkraal, 31 Ω, 3 A, 1206

### Condensatoren:

C1,C5,C9,C13,C20 = 4,7 μ, 100 V, 7,7 x 6,3 mm

C2,C6,C10,C14,C18,C19,C29 = 100 n, 100 V, X7R, 0805

C3,C4,C7,C8,C11,C12,C15,C16,C21 = 27 μ, 100 V, 20%,  
8 mm, radiaal

C17 = 47 μ, 10 V, 2312

C22 = 4,7 μ, 50 V, X7R, 1210

C23 = 100 μ, 10 V, 5,5 x 5,5 mm

C24,C25,C26,C27 = 2,2 μ, 100 V, X7R, 1210

C28 = 1 n, 16 V, X7R, 0603

### Halfgeleiders:

D1,D2 = MBRS540, 40V, VF = 550 mV @ If = 5 A

LED1 = LED, groen, 3 mm

LED2 = LED, rood, 3 mm

LED3 = LED, geel, 3 mm

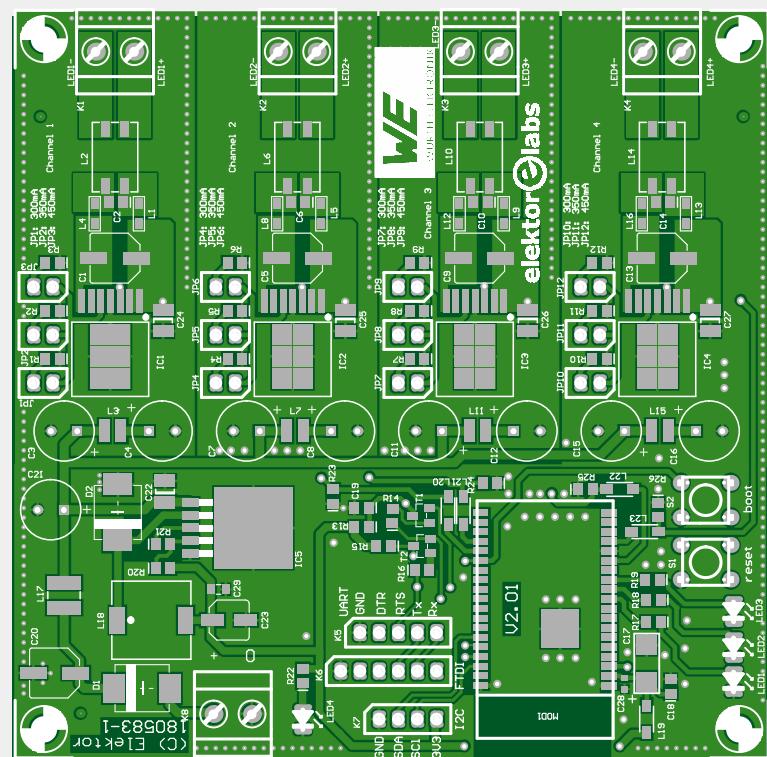
LED4 = LED, blauw, 3 mm

T1,T2 = BC847C, 45 V, 100 mA, 250 mW, hfe = 400

IC1...IC4 = MagI3C LED Step Down High Current, Würth 172946001

IC5 = LM2576HVS-ADJ, step-down regulator, 4-60 V, 3 A

MOD1 = ESP-32-WROVER-B



60% ware grootte

### Diversen:

S1,S2 = drukknop, 12 V, 5 mA, 6 x 6 mm

K1,K2,K3,K4,K8 = tweepolige printkroonsteem steek 0,2", 630 V

K5 = 5-polige pinheader, steek 0,1", verticaal

K6 = 6-polige pinheader, steek 0,1", verticaal

K7 = 4-polige pinheader, steek 0,1", verticaal

JP1...JP9 = jumper, 1x2, steek 0,1", verticaal

4 stuks 2-polige jumper, steek 0,1"

Print 180583-1 v2.01



## ONDERDELENLIJST LED-PRINT

### Halfgeleiders:

LED5...LED13 = high brightness LED, WL-SMDC serie, rood, 660 nm, 125°, 700 mA

LED14...LED18 = high brightness LED, WL-SMDC serie, blauw, 450 nm, 125°, 700 mA

LED19...LED22 = high brightness LED, WL-SMDC serie, rood, 730 nm, 125°, 700 mA

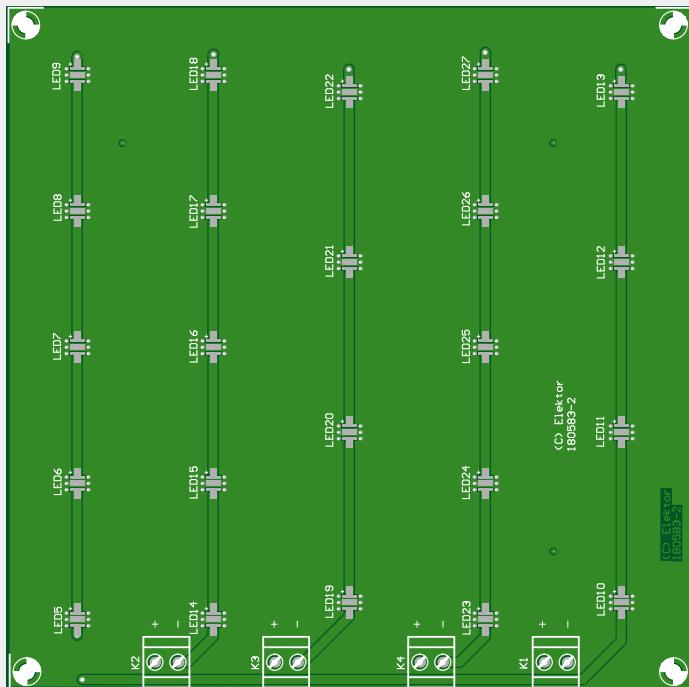
LED...LED27 = high brightness LED, WL-SWTC serie, wit, 120°, 121 lm, 6000 K, 700 mA

### Diversen:

K1...K4 = 2-polige printkroonsteen, steek 0,2", 630 V

Print 180583-2 v1.1

60% ware grootte



## Pas op uw ogen!

Voorkom blijvende schade aan uw ogen en kijk nooit rechtstreeks in de lichtbron van de kweekbox (of andere felle vermogens-LED's). De lichtsterkte hiervan is zeer hoog, zelfs als ze gedimd zijn.

En er is nog iets wat we ons moeten realiseren bij de omgang met PWM-bestuurde LED's zoals in dit project. In tegenstelling tot bijvoorbeeld gloeilampen kan de intensiteit van een LED zeer snel veranderen. Een LED die ogenschijnlijk uit is, kan extreem sterke lichtpulsen uitzenden die zo kort duren, dat het menselijk oog ze niet waarneemt. Hoewel de gevolgen hiervan niet helemaal duidelijk zijn, betekent dat niet dat ze ongevaarlijk zijn – gezond is het zeker niet.

Het veiligste is het daarom om nooit rechtstreeks in vermogens-LED's te kijken, ook al lijken ze uit te zijn.

systeem helemaal opnieuw te flashen. Het uploaden van een ESP-sketch kan eveneens draadloos worden uitgevoerd, klik daarop op 'ESP32 Sketch Data Upload' in het Tools-menu.

### Een beetje software

In dit project gingen hardware- en software-ontwikkeling hand in hand, waardoor een mooie interactie tussen beide ontstond, in het bijzonder bij de toewijzing van de I/O-pinnen. Tijdens het ontwerpproces veranderden enkele pinnen van functie om het printontwerp te vereenvoudigen, terwijl daarvoor slechts één regel code hoeft te worden aangepast.

Wie de laatste edities van Elektorlabs heeft gelezen, heeft vast wel gezien dat behoorlijk wat ESP32-projecten zijn gepubliceerd. Als ontwerper begint u nooit vanaf nul, maar maakt u gebruik van eerder ontwikkelde 'componenten' die u als puzzelstukjes in uw nieuwe project probeert in te passen. Voor de kweekbox-software wordt de basis gevormd door routines die vergelijkbaar zijn met die van de 'Gigantische LED-klok' [4] en het 'ESP32-weerstation' [5], namelijk een webserver en de basisroutines om een WiFi-verbinding op te zetten. Voor de kweekbox zijn nog wat modules toegevoegd. De eerste is 'ledc', een PWM-

unit met wat aardige extra's om LED's aan te sturen. Dit is makkelijk te doen door de gewenste PWM-frequentie en de resolutie op te geven. De LED-driver van Würth Elektronik is gebaseerd op de dim-verhouding, maar we moeten ook een minimum aan-tijd in acht nemen. Zoals bij alle PMW's willen we natuurlijk de resulterende resolutie kennen bij een gegeven PWM-frequentie. De PWM heeft 40 MHz als ingangs frequentie en we kunnen de maximale resolutie in bits uitrekenen met:

$$\log_2(40 \times 10^6 / 250) = 17,2$$

Dit resulteert in een bruikbare resolutie van 17 bits. Omdat we een minimum aan-tijd voor de PWM hebben, gebruiken we in de code slechts 8 bits. Om de PWM-unit te initialiseren gebruiken we deze regel code:

```
ledcSetup(CHANNEL, PWM_FREQ, PWM_RES);
```

Dit stelt één van acht beschikbare LED-kanalen in. Anders dan gebruikelijk is dit ledc-kanal niet gebonden aan een specifieke pin van de ESP32, maar kunnen we elke beschikbare pin gebruiken. Om dit in te stellen hebben we de volgende regel nodig:

```
ledcAttachPin(GPIO_PIN, CHANNEL);
```

Hieraan zijn we klaar om de PWM-pinnen te besturen. We kunnen als volgt een waarde instellen:

```
ledcWrite(CHANNEL, VALUE);
```

We kunnen ook de huidige waarde van een ledc-kanaal teruglezen:

```
ledcRead(CHANNEL)
```

Hiermee is de basis gelegd voor het besturen van de ‘ledc-unit’. Binnen de software-module hebben we de vier fysieke LED-kanalen en een vijfde die geheel softwarematig is, om de algemene helderheid van alle kanalen te kunnen instellen. In de web-interface heet dit ‘intensity’. Als deze waarde wordt gewijzigd heeft dat niet direct effect. In plaats daarvan wordt door een timer iedere 100 ms een functie aangeroepen die controleert of de ingestelde waarde overeenkomt met de huidige waarde. Als dat niet het geval is, wordt de waarde met een gegeven maximale stapgrootte verhoogd of verlaagd. Dit leidt tot een geleidelijke verandering van de helderheid. Dit wordt voor alle kanalen tegelijk gedaan; bij inschakelen neemt de intensiteit langzaam toe.

De module die in de ESP32-LED-klok de tijd bijhoudt is – met een paar aanpassingen – eveneens toegevoegd. Sommige planten hebben een dag- en nachtritme, zodat we een functie kunnen bouwen die de LED’s gedurende een bepaalde periode van de dag uitschakelt.

### Natuurlijk een webinterface

Zoals te zien in de screenshots van **figuur 4**, is de gebruikers-interface gerecycleerd uit eerdere Elektor-projecten. De hoofdpagina heeft vijf velden voor de PWM-instellingen. Kanalen 1 t/m 4 zijn de fysieke kanalen van de kweekbox, terwijl ‘intensity’ de algehele helderheid bepaalt. Plus en Min wijzigen de helderheid in stapjes van 1%, maar u kunt ook direct waarden invullen tussen 0 en 100%. In de communicatie op de achtergrond worden wijzigingen van de waarden verwerkt via web-sockets. Als meer dan één persoon de waarden aanpast, krijgen alle andere gebruikers die waarden ook naar hun browser gepushed. In de code is dit gedaan door een aparte web-socket server die de berichten niet alleen verzendt, maar ook verwerkt als JSON. Er is een apart deel om de LED’s gedurende bepaalde perioden te kunnen schakelen.

Binnen ‘Time settings’ kan worden gekozen tussen handmatige tijdinstelling, of automatisch via een NTP-server.

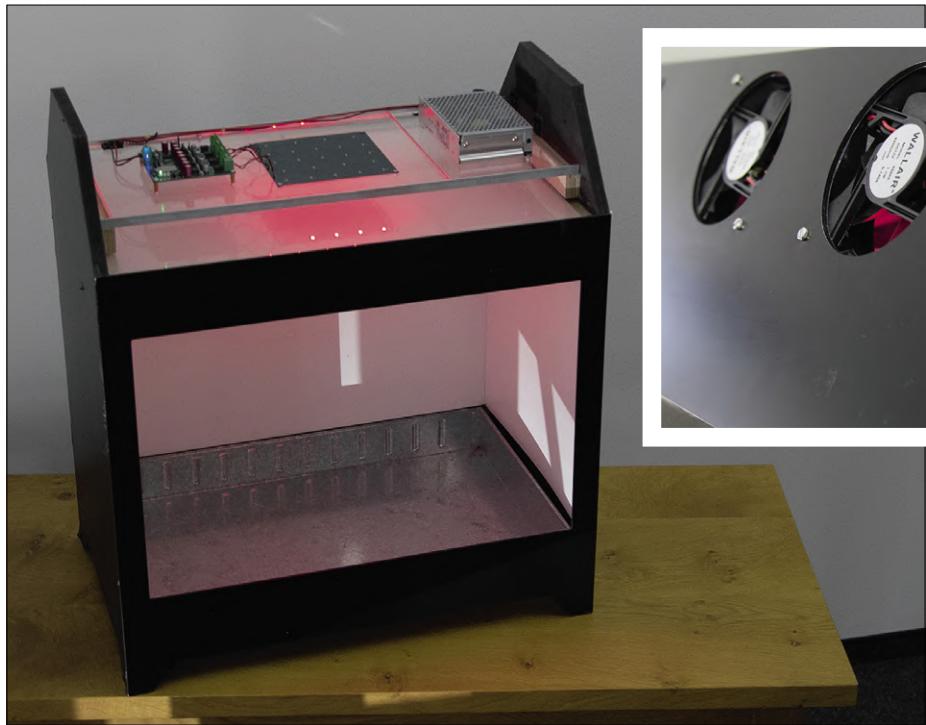
Bij WiFi-settings kunnen uw eigen WiFi-instellingen worden ingevuld. Hierdoor kan de kweekbox worden bestuurd door elk willekeurig op dit netwerk aangesloten apparaat.

Het MQTT-gedeelte komt u wellicht niet bekend voor. Dit hebben we overgenomen uit het weerstation-project; het bevat een manier om de LED-instellingen en de automatische aan/uit-timer te besturen vanuit een MQTT-broker. Dit gebeurt door de PubSubClient-library en ArduinoJson voor gegevensuitwisseling. Om nieuwe settings naar het systeem te sturen, moet een JSON-string worden aangemaakt die er als volgt uit ziet:

The figure displays five screenshots of the Elektor Horticulture box's web-based user interface. The left sidebar for all screens shows a menu with the following items: Main page, Notes, MQTT settings, Time settings, WiFi settings, and Restart. The 'Notes' item is highlighted in grey, indicating it is the active screen.

- Main page:** Shows a large text input field for notes with a character limit of 500. A 'Update Notes' button is at the bottom right.
- Time settings:** Allows setting local time, NTP settings (using an NTP server and updating every 0 to 60 minutes), and manual time zone adjustment (Europe/London) or manual daylight saving.
- Network settings:** Displays SSID (VGV7519A8D43) and password fields, along with 'Connect' and 'Clear WiFi Settings' buttons. It also indicates 'No network configured'.
- MQTT settings:** Includes options for enabling the MQTT client, setting an interval (10 seconds), specifying MQTT Hostname, Server (IP 0.0.0.0, Port 1833), Topic, User, and Password, and a 'Submit' button.
- LED settings:** Features four channels for individual LED control (0% to 100%) and an 'Intensity' slider. It also includes an 'Automatic Light' section with a checkbox for 'Timed LED ON/OFF Enable' and fields for 'LED ON Start' (00:00:00) and 'LED ON End' (00:00:00), with a 'Submit' button.

Figuur 4: Screenshots van de gebruikersinterface voor de LED-kweekbox: hoofdpagina, MQTT-pagina, notities, tijd-instellingen en WiFi. De interface is een gerecycleerde versie uit twee eerdere Elektorlabs-projecten.



Figuur 5: Van de 'HortiCooture' behuizing is door Würth Elektronik slechts één exemplaar gemaakt, speciaal voor promotiedoeleinden. Het frame is gemaakt van multiplex, de panelen van acryl, en de plantenbak van verzinkt staal. Het 'dak' is afneembaar en bevat de elektronica plus twee kleine ventilatoren.



Figuur 6. De jongste versie van de Horticulture Box komt uit een 3D-printer. (product en afbeelding: Würth Elektronik)

```
{
  "light": {
    "ch0": 6400,
    "ch1": 0,
    "ch2": 0,
    "ch3": 0,
    "intense": 32000
  },
  "timer": {
    "enable": false,
    "start": {
      "hour": 0,
      "minute": 0,
      "second": 0
    },
    "end": {
      "hour": 0,
      "minute": 0,
      "second": 0
    }
  }
}
```

Voor 'ch0' tot 'intensity' zijn 16-bit unsigned getallen toegestaan. Grote getallen worden automatisch begrensd op 65535, de ondergrens is 0. Zo mag bijvoorbeeld ook NodeRed de besturing overnemen, om de planten niet alleen te voorzien van een daglicht-cyclus maar ook van verschillende kleurnuances. Niet alle velden hoeven te zijn ingevuld om de waarden te wijzigen. Een voorbeeld

## Web Links

- [1] Würth Horticulture LEDs Application Notes: [https://katalog.we-online.de/en/led/WL-SMDC\\_HORTICULTURE](https://katalog.we-online.de/en/led/WL-SMDC_HORTICULTURE)
- [2] Arduino IDE voor de ESP32: [www.elektormagazine.com/labs/esp32-getting-started](http://www.elektormagazine.com/labs/esp32-getting-started)
- [3] Horticulture Box firmware download: [www.elektormagazine.nl/180583-04](http://www.elektormagazine.nl/180583-04)
- [4] Gigantische LED-klok: [www.elektormagazine.nl/180254-03](http://www.elektormagazine.nl/180254-03)
- [5] ESP32-weerstation: [www.elektormagazine.nl/180468-01](http://www.elektormagazine.nl/180468-01)

waarbij alleen de waarde van 'ch0' wordt gewijzigd:

```
{  
  "light": {  
    "ch0": 6400  
  }  
}
```

Dit zet alleen het eerste kanaal op 6400. Om dit mogelijk te maken, controleert de code die de MQTT-berichten verwerkt of het om individuele objecten gaat. Als we ArduinoJson gebruiken, gaat dat als volgt:

```
JsonObject light = doc["light"];  
JsonVariant light_ch0_var = light["ch0"];  
If (false == light_ch0_var.isNull() ){  
  ...  
}
```

In het if-statement controleren we of er een waarde voor een bepaald kanaal is. Als dat zo is wordt die verwerkt en anders genegeerd. Als u zelf JSON wilt verwerken, kunt u beter controleren of de waarden die u nodig hebt werkelijk binnen JSON voorhanden zijn, want anders gebruikt u wellicht onbedoelde standaardinstellingen.

### LED it grow

Onze kweekbox is geïnstalleerd in een behuizing die als *paludarium* (een kunstmatige leefomgeving met zowel een land- als een watergedeelte) kan worden gebruikt, maar elke behuizing met de LED-print boven uw geliefde planten is goed. **Figuur 5** toont de door Würth Elektronik speciaal eenmalig voor promotiedoelen gebouwde 'HortiCoolture' behuizing. Deze heeft bovenin twee kleine ventilatoren om de LED-print te koelen. **Figuur 6** toont een later ontwerp van Würth Elektronik met 3D-geprinte onderdelen.

Het zelf ontwikkelen van een dergelijke fraai uitziende behui-

zing ligt wellicht niet binnen uw mogelijkheden, maar er zijn natuurlijk altijd goedkopere alternatieven, zoals:

- zoek naar een aquarium of herbarium in uw lokale kringloopwinkel;
- breng een bezoekje aan Ikea;
- kijk eens rond bij de aquarium- en vijverwinkel.

Met name Nederlandse lezers zouden ook een bezoekje aan een 'Growshop' kunnen overwegen.

Hoewel er een groot aantal wetenschappelijke publicaties zijn over de optimale lichtcondities voor specifieke planten, dagen we u graag uit proefondervindelijk uw eigen 'lichtrecepten' te ontwikkelen. ▶

(180583-04)



### IN DE STORE

- Kweekbox driver- en LED-print, kant-en-klar geassembleerd, inclusief geprogrammeerde ESP-32-W-ROVER-B module [www.elektor.nl/180583-71](http://www.elektor.nl/180583-71)
- Kweekbox 'kale' driverprint 180593-1 v.2.01 [www.elektor.nl/180583-1](http://www.elektor.nl/180583-1)
- Kweekbox 'kale' LED-print 180593-2 v1.1 [www.elektor.nl/180583-2](http://www.elektor.nl/180583-2)

Advertentie



**HortiCoolture.  
Let it grow!**

**#LEDITGROW**  
**WE speed up  
the future**

**Electronic components  
and printed circuit boards  
for a successful growth!**

[www.wedirekt.com](http://www.wedirekt.com)  
[www.we-online.com](http://www.we-online.com)

# WSPR-berichten verzenden

## bereik heel Europa met het Elektor SDR-shield

**Burkhard Kainka** (Duitsland)

The screenshot shows the WSPRnet homepage. At the top, there's a banner with the text "WSPRnet" in large yellow letters over a background of colorful spectrograms. Below the banner, the word "WSPRnet" is displayed in a large, bold, black font. To its right is a navigation bar with links: Activity, Map, Database, Stats, Forum, and Downloads. The main content area has a blue header "WSPRnet" and "Welcome to the Weak Signal Propagation Reporter Network". On the left, there's a "User login" form with fields for Username and Password, and links for Create new account and Request new password. Below it is a "Frequencies" section listing various USB dial values. On the right is a world map showing propagation paths between stations like OH8GKP (Finland) and EA8BFK (Spain), with labels for countries and洋 (Oceans).

De WSPR-modulatietechniek is ontwikkeld om met gering vermogen en dito bandbreedte grote afstanden te overbruggen. We laten hier zien hoe u met een Elektor-SDR-shield en gratis software WSPR-berichten kunt verzenden.

De WSPR-modulatietechniek (*Weak Signal Propagation Reporter*, spreek uit als ‘whisper’) is ontwikkeld om met kleine vermogens en kleine bandbreedtes grote afstanden te overbruggen. Er zijn altijd veel stations actief en die melden hun ontvangstresultaten op de WSPR-website [1]. Hierdoor kunt u heel snel een overzicht te krijgen waar een signaal ontvangen kan worden ontvangen. 1000 kilometer met slechts 10 mW is haalbaar. Met 100 mW kan zonder enig probleem heel Europa worden bereikt. Wat we nodig hebben is het programma WSPR 2.0

[2] van Joe Tailor, die WSPR heeft ontwikkeld. Het zendsignaal is in essentie een constante toon van 1,5 kHz, die doorgaans door een SSB-transceiver wordt uitgezonden. Als je heel goed luistert, kun je zeer kleine verschillen in deze toon waarnemen. WSPR maakt gebruik van een 4FSK-signal met vier frequenties op een onderlinge afstand van 1,46 Hz. Elke frequentie blijft ongeveer één seconde stabiel. Daardoor kan de ontvanger-software extreem smalbandig filteren en zodoende een goede stooraafstand bereiken.

De WSPR-ontvangstechniek met het Elektor-SDR-shield is al eerder genoemd in Elektorlabs [3]. Nu laten we ook de zendtechniek zien. Daarvoor hebben we nog Arduino-software nodig, een kleine HF-eindtrap en de antenne-aanpassing. De hier getoonde zendexperimenten kunnen in deze vorm alleen legaal door gelicenseerde zendamateurs worden uitgevoerd. Geïnteresseerden zonder zo'n licentie kunnen wel op beperkte schaal experimenteren zonder aangesloten zendantenne, waarbij het eigen signaal weer ontvangen wordt.

### Aansturing van het SDR-shield

Het SDR-shield wordt doorgaans als kortegolfontvanger gebruikt. Maar er nog twee extra uitgangen van de PLL-generator Si5351 beschikbaar, die kunnen worden gebruikt voor meetdoeleinden of voor het aansturen van een zender. Hiermee kan ook een WSPR-signalen worden gegenereerd. Deze en andere experimenten worden beschreven in het (Engelstalige) SDR Hands-on Book [4].

Bij het schrijven van dat boek en van de eerste artikelen over het SDR-shield heeft de auteur de oudere Etherkit Si5351-library van Jason Milldrum gebruikt (versie 1.1.2). Later ingestapte gebruikers kwamen eerst de nieuwere versie 2.1.4 tegen. Maar de nieuwere library-versie leidde in combinatie met de oorspronkelijke code tot foutmeldingen, omdat enkele structuren enigszins waren gewijzigd.

De initialisatie vereist nu een extra parameter 0, waarbij de extra 0 aangeeft dat we de kristalfrequentie op deze plek niet (afwijkend van 25 MHz) nauwkeuriger willen opgeven.

```
si5351.init(SI5351_CRYSTAL_LOAD_8PF, 0, 0);
```

De eigenlijke kalibratie moet plaatsvinden na de initialisatie, waarbij `..INPUT_X0` aangeeft, dat deze betrekking heeft op de kristaloscillator;

```
si5351.set_correction(162100, SI5351_PLL_INPUT_X0);
```

De uitvoer van de frequentie werkt nu zonder de eerdere parameter `PLL_FIXED`, omdat de bibliotheek automatisch de PLL-instellingen vastlegt:

```
si5351.set_freq(freq*100ULL, SI5351_CLK0);
```

Intussen zijn alle voorbeeldprogramma's in het boek herzien, waaronder ook de eerder in de artikelserie over het SDR-shield

### Listing 1. De initialisatie in si5351vfo2\_1.ino.

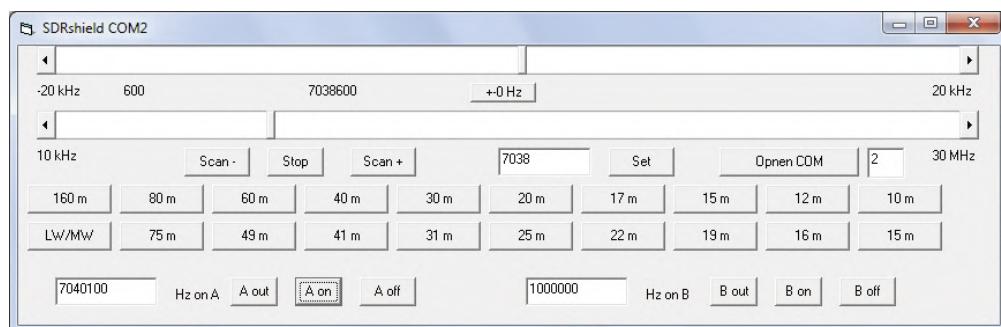
```
{
    Serial.begin(9600);
    Serial.println("Si5351 Clockgen"); Serial.
    println("");
    si5351.init(SI5351_CRYSTAL_LOAD_8PF, 0, 0);
    si5351.set_correction(162100,
    SI5351_PLL_INPUT_X0);
    Serial.println(10100);
    si5351.set_freq(404000000ULL, SI5351_CLK1);
    si5351.output_enable(SI5351_CLK1, 1);
    freq = 10100;
    lcd.begin(16, 2);
    lcd.print(freq);
}
```

voorgestelde toepassingen. Het bij het boek horende software-archief [4] bevat nu zowel de oorspronkelijke als de geactualiseerde versies.

Om te beginnen hebben we het afstemprogramma *si5351vfo2\_1* (**listing 1**) nodig. Hier wordt de VFO-frequentie van 7038,6 kHz ingesteld (**figuur 1**), waarmee WSPR-signalen in de 40m-band naar 1,5 kHz omhoog gemengd kunnen worden. Ter controle van een juiste instelling kan uitgang A worden geactiveerd. Een signaal bij 7040,1 kHz straalt dan zwak in op de ingang en kan in WSPR 2.0 precies in het midden van het 200 Hz brede ontvangstbereik worden gezien (**figuur 2**). De kalibratie van de ontvangerfrequentie is echter meestal niet nauwkeurig genoeg. We zien dat de draaggolf wordt ontvangen, omdat er een zwakke capacitive koppeling is. Maar de modulatie ontbreekt nog en dus ook de gewenste omschakeling tussen de vier naburige frequenties van het 4FSK-signalen.

### De JTEncode-library

Ook van Jason Milldrum komt de *JTEncode*-library [5] voor het genereren van WSPR- en andere digitale signalen met de Si5351. Op basis van zijn programmavoorbeeld is een eerste programma voor het SDR-shield ontwikkeld. Daartoe wordt de ontvanger vast afgestemd op de WSPR-frequentie in de 20m-band (VFO = 14095,600 kHz). De zender werkt in dezelfde band (14097,100 kHz) en start zodra pin 12 van de Arduino laag wordt gemaakt. Het eigen signaal is dan bij 1,5 kHz te horen.



Figuur 1. Instelling van de VFO-frequentie.

**Listing 2. Zenden van een WSPR-signaal (gedeelte van Si5351WSPR1.ino).**

```
// Simple JT65/JT9/WSPR/FSQ beacon for Arduino, with the Etherkit
// Si5351A Breakout Board, by Jason Milldrum NT7S.

#include
#include
#include
#include
#include "Wire.h"
#define BUTTON          12
#define LED_PIN         13

...
char message[] = "EL2SDR JO31";
char call[] = "EL2SDR"; //das Rufzeichen
char loc[] = "JO31";   //der Lacotor wie z.B. JO31";
uint8_t dbm = 10;      //10 mW, 30 für 1W, 37 für 5 W
uint8_t tx_buffer[255];
...

void encode()
{
    uint8_t i;
    // Clear out the old transmit buffer
    memset(tx_buffer, 0, 255);
    jtencode.wspr_encode(call, loc, dbm, tx_buffer);
    // Reset the tone to the base frequency and turn on the output
    si5351.output_enable(SI5351_CLK0, 1);
    digitalWrite(LED_PIN, HIGH);
    for(i = 0; i < symbol_count; i++)
    {
        si5351.set_freq((freq * 100) + (tx_buffer[i] * tone_spacing), SI5351_CLK0);
        Serial.print (tx_buffer[i]); Serial.print ",";
        proceed = false;
        while(!proceed);
    }
    // Turn off the output
    si5351.output_enable(SI5351_CLK0, 0);
    digitalWrite(LED_PIN, LOW);
}

void loop()
{
    if(digitalRead(BUTTON) == LOW)
    {
        delay(50); // delay to debounce
        if (digitalRead(BUTTON) == LOW)
        {
            encode();
            delay(50); //delay to avoid extra triggers
        }
    }
}
```

Voor het programma *Si5351WSPR1.ino* (**listing 2**) hebben we de (niet-bestante!) roepnaam EL2SDR verzonnen, die natuurlijk alleen voor interne experimenten zonder aangesloten antenne toegestaan is. Daarnaast moeten een QTH-locator (in het geval van de auteur is dat JO31) en het zendvermogen (10 dBm) worden opgegeven. Hieruit vormt de encode-functie een zendbuffer met de WSPR-symbolen 0, 1, 2 en 3, die staan voor vier dicht bij elkaar liggende frequenties op een onderlinge afstand van 1,46 Hz.

```
jtencode.wspr_encode(call, loc, dbm, tx_buffer);
```

Een typische bufferinhoud ziet er dan zo uit:

```
1,3,0,2,2,2,2,2,1,0,2,0,1,3,1,0 ...
```

Hiermee wordt nu tijdens het eigenlijke zenden de frequentie van het FSK-signal gestuurd (listing 2).

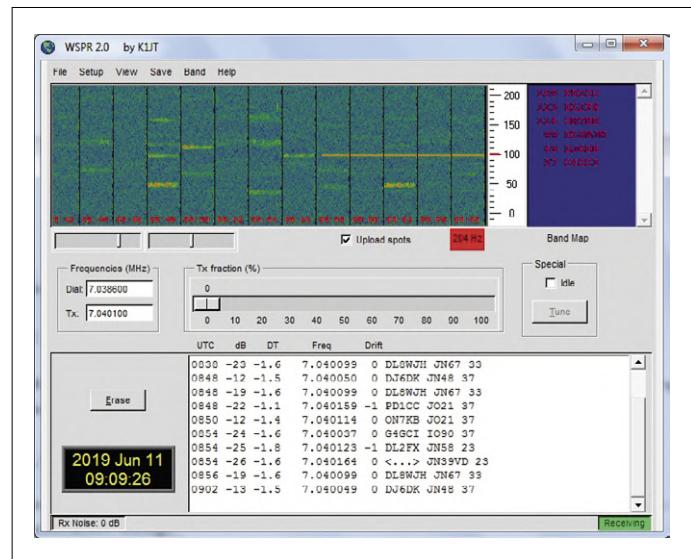
Een uitzending duurt twee minuten en begint op de hele minuut. Voor een goed verloop moet die tijd precies aangehouden worden. Twee seconden na het begin van een hele minuut wordt de eigen zender gestart door een drukknop aan pin 12.

De ontvanger kan met een antenne worden uitgerust om echte WSPR-stations te ontvangen. De zender-uitgang daarentegen blijft vrij en krijgt geen antenne. Het signaal wordt vanuit uitgang A voldoende sterk capacitief gekoppeld met de antenne-ingang en wordt door WSPR 2.0 correct ontvangen.

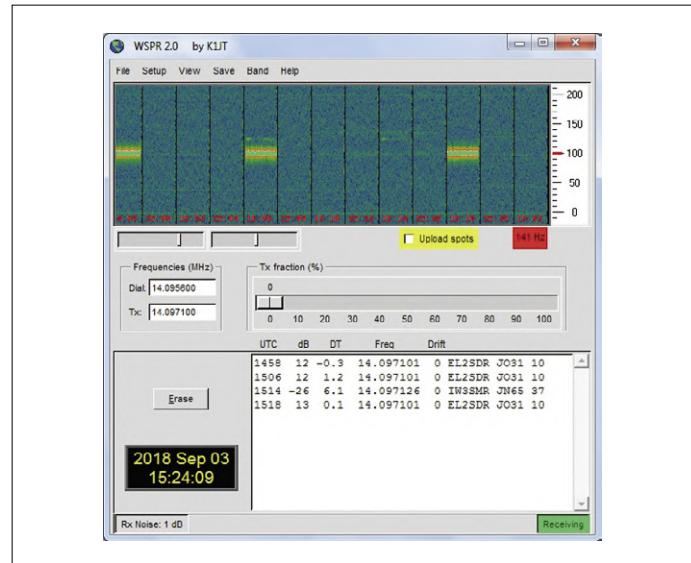
**Figuur 3** toont het resultaat van het experiment. Het eigen signaal ligt 12 dB boven de ruis. Daarnaast werd een Italiaans station (dat een zendvermogen van 37 dBm (5 W) had) met -26 dB gedecodeerd.

### Kalibratie en start-timing

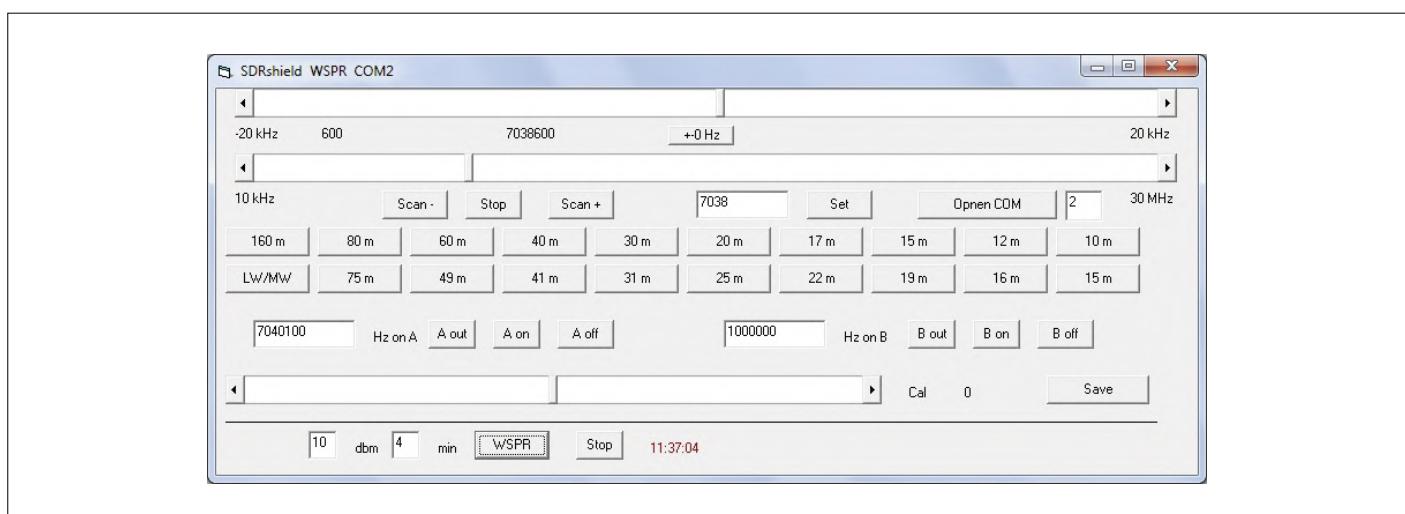
Om op alle banden met WSPR te kunnen werken, is een uitgebreide firmware *si5351vfo2\_WSPR.ino* en een geschikt PC-gebruikersprogramma *SDRshield\_WSPR* geschreven. Doorslaggevend is de precieze frequentie. De auteur heeft daarom een kalibratiefunctie geïntegreerd. Als frequentie-standaard zoekt u eerst een kortegolf-omroepstation op. Met de schuifregelaar wordt de ontvanger precies op deze frequentie ingesteld. In het spectrum van de software SDR# [6] moet de draaggolf nu precies op het 5 kHz-raster staan. De instelling kan worden



Figuur 2. WSPR-signalen en de hulpdraaggolf A.



Figuur 3. Ontvangst van het eigen signaal.



Figuur 4. WSPR-frequentie 7038,600 kHz in de 40m-band.

### Listing 3. De zenderbesturing.

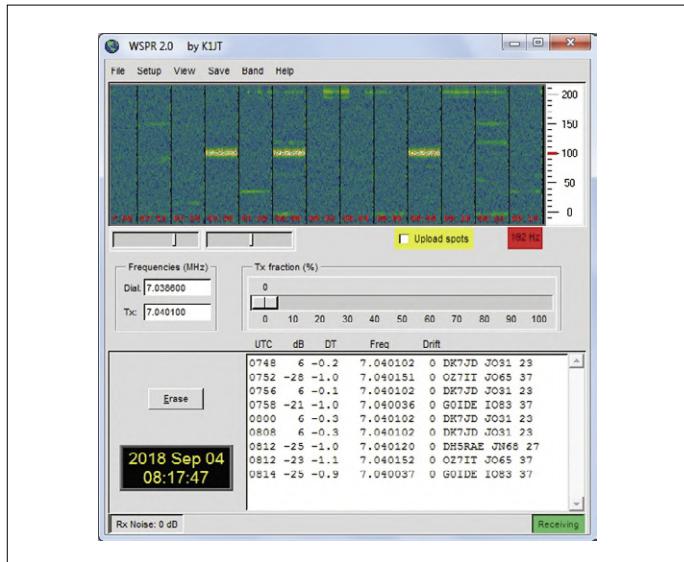
```

if (ch == 119) {           // w, Start wspr
    dbm = number;
    encode();
}

...
if (ch == 102) {           //f
    si5351.set_freq(freq*400ULL, SI5351_CLK1);
    wsprfreq = freq;
}

void encode()
{
...
    si5351.set_freq((wsprfreq * 100)+ 150000 + (tx_buffer[i] * tone_spacing),
                    SI5351_CLK0);
...
}

```



Figuur 5. Ontvangst van het eigen signaal.

getest met andere omroepzenders, die in de meeste gevallen hun frequentie zeer nauwkeurig aanhouden. Met Save wordt de correctie naar de Arduino overgebracht.

Het VB-gebruikersprogramma is zodanig gewijzigd dat de snel-toetsen voor de amateur-radiobanden niet langer het begin

van de band, maar de voor deze band vastgestelde WSPR-frequentie kiezen. De VFO wordt met een klik op de 40m-button op 7038,600 kHz ingesteld.

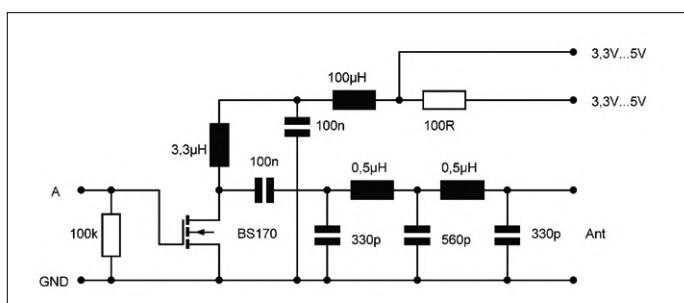
Met een klik op de WSPR-knop wordt de zendfunctie geactiveerd (**figuur 4**). De zender start daarna op de eerstvolgende hele minuut en vervolgens telkens na een vooraf ingestelde interval van 2 (default), 4, 6, 8 of 10 minuten. De tijd wordt tijdens de actieve fasen rood weergegeven.

De momentele ontvangstfrequentie wordt in de Arduino-sketch aan de zendfrequentie toegewezen. Met het *w*-commando start de uitzending, zodat niet meer op een startpuls op pin 12 hoeft te worden gewacht. De enige verzonden parameter is het momentele zendvermogen in dBm. Tijdens het zenden worden bij de frequentie in *wsprfreq* nog 1500 Hz en de betreffende modulatiefrequentie opgeteld (**listing 3**).

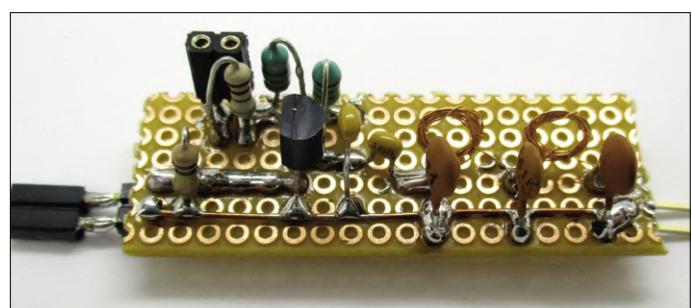
De eerste test in de 40m-band werd weer zonder antenne, met open aansluiting A uitgevoerd. **Figuur 5** toont enkele via de ontvangstantenne gedetecteerde stations, met daartussen het eigen signaal dat met de eigen stationsnaam DK7DJ van de auteur werd gegenereerd. Weliswaar is hier sprake van een draadloze verbinding, maar de reikwijdte is met 2 cm tamelijk bescheiden...

### 200mW-zendversterker

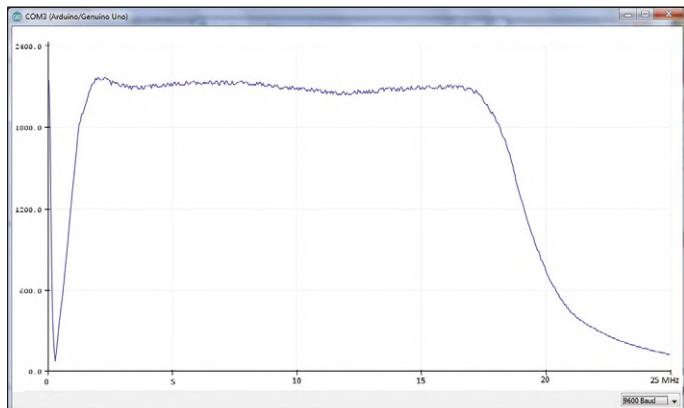
**Figuur 6** toont een kleine zend-eindtrap met een MOSFET BS170. Deze levert maximaal 200 mW (23 dBm) bij een voedingsspanning van 5 V. Naar keuze kan ook een spanning van 3,3 V worden gebruikt, waarmee een uitgangsvermogen van



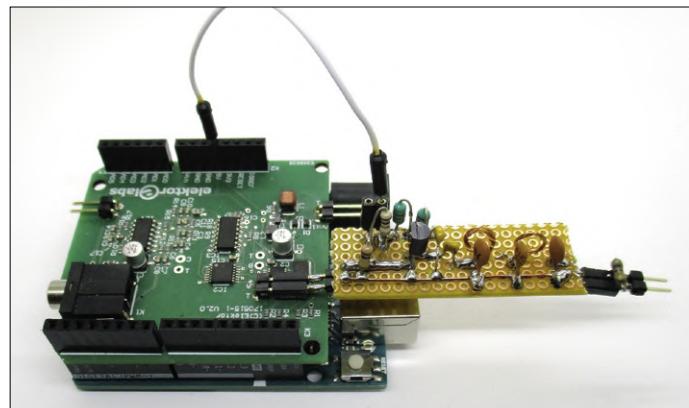
Figuur 6. 200mW-eindtrap met een BS170.



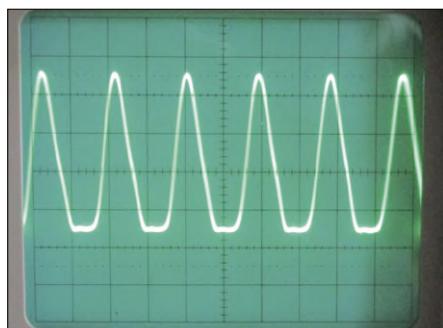
Figuur 7. FET-eindtrap met filter.



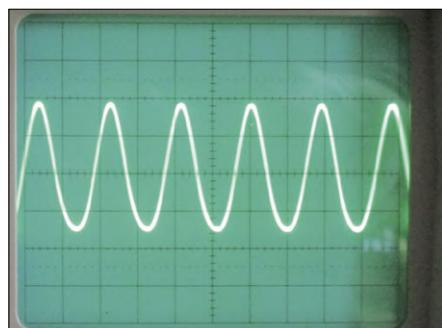
Figuur 8. De frequentiekarakteristiek van de eindtrap.



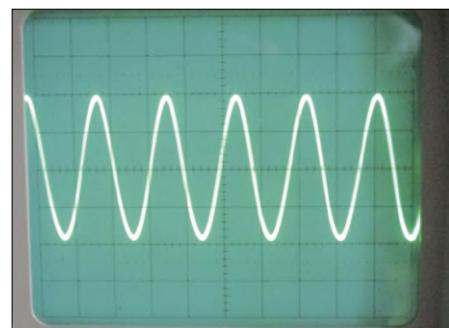
Figuur 9. Met voedingsspanning 5 V en afsluitweerstand.



Figuur 10. Signaal op de drain van de FET.



Figuur 11. In het midden van het filter.



Figuur 12. Aan de afsluitweerstand.

100 mW (20 dBm) kan worden gehaald. Met een serieweerstand van  $100 \Omega$  wordt het vermogen nogmaals met 10 dB gereduceerd, zodat nu 10 mW (10 dBm) respectievelijk 20 mW (13 dBm) aan de antenne wordt geleverd. Het uitgangsvermogen is op deze manier in een ruim bereik instelbaar:

- 5 V: 200 mW
- 3,3 V: 100 mW
- 5 V via  $100 \Omega$ : 20 mW
- 3,3 V via  $100 \Omega$ : 10 mW

Het laagdoorlaatfilter aan de uitgang van de eindtrap is geoptimaliseerd voor de 20m- en 30m-banden. De filterspoelen van  $0,5 \mu\text{H}$  worden gewikkeld als luchtspoelen van tien windingen met een diameter van 5 mm. **Figuur 7** toont de compacte opbouw van de eindtrap.

Het laagdoorlaatfilter is gedimensioneerd voor een typische werkfrequentie van 10 MHz. Om dit te testen is het SDR-shield met een vierpool-meetprogramma onderzocht.

De meetresultaten (**figuur 8**) tonen een bovenste grensfrequentie van 18 MHz. Hierdoor is ook de 20m-band rond 14 MHz nog te gebruiken en – binnen zekere grenzen – ook de 17m-band rond 18 MHz. De nog bruikbare ondergrens van de zendeindtrap ligt rond de 2 MHz. Vanaf 10 MHz is voldoende verzwakking van harmonischen te verwachten om een geschikte antenne direct te mogen aansluiten. De onderste grensfrequentie van 2 MHz is het gevolg van de relatief geringe zelfinductie van  $3,3 \mu\text{H}$  in de drain-kring van de eindtrap.

Nader onderzoek met de oscilloscoop toont de werking van het laagdoorlaatfilter en het haalbare uitgangsvermogen. Daartoe

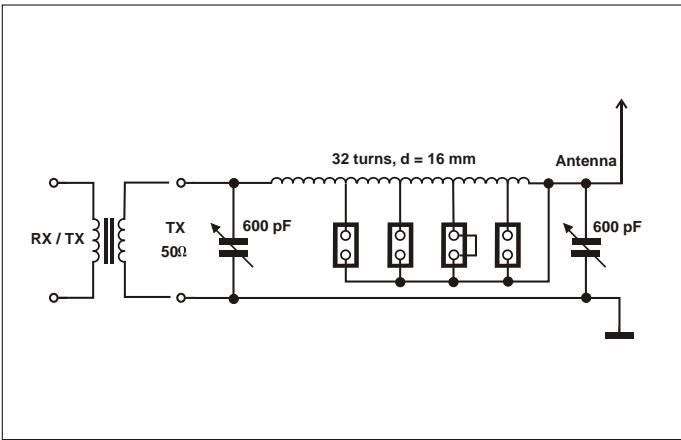
Advertentie

**Maker Faire Eindhoven**

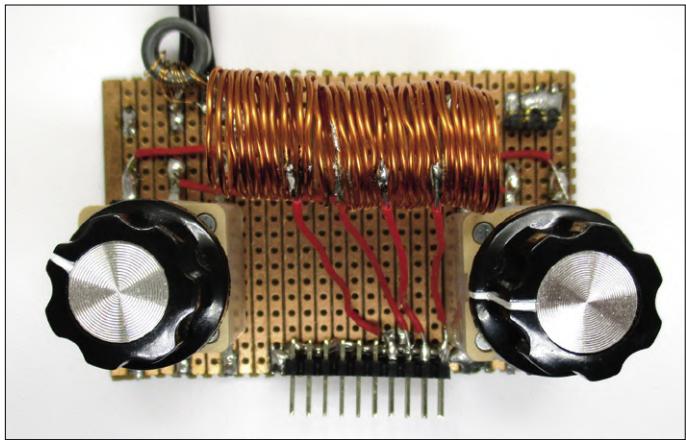
**28-29 september**

**tickets online**

Klokgebouw Strijp-S [eindhovenmakerfaire.nl](http://eindhovenmakerfaire.nl)



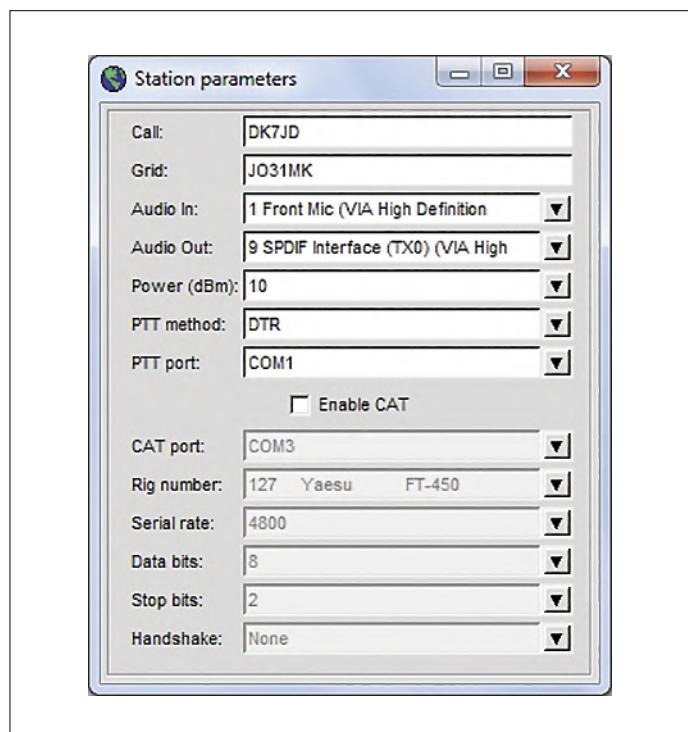
Figuur 13. Pi-filter met breedband-transformator.



Figuur 14. Opbouw met luchtpoelen en folie-draaicondensatoren.



Figuur 15. WSPR-spots bij 200 mW.



Figuur 16. Software-instellingen.

werd de eindtrap met 5 V gevoed en voorzien van een belastingsweerstand van  $51 \Omega$  (**figuur 9**). Over de belastingsweerstand staat een spanning van 8 Vpp, een amplitude dus van 4 V. Hieruit is het uitgangsvermogen te berekenen:  $P = U^2 / 2 R = (4 V)^2 / 100 \Omega = 0,16 W = 160 \text{ mW}$ . Experimenten lieten zien dat met een wat hogere voedingsspanning van 7 V nog iets meer vermogen haalbaar is. De scoopplaatjes in **figuur 10...12** tonen de steeds betere filtering tot aan de antenne-uitgang. Al met al zijn alle banden van 160 m tot 20 m met deze eindtrap te gebruiken. Maar dan is wel extra onderdrukking van harmonischen vereist. Dit lukt relatief eenvoudig met een antenne-adapter in de vorm van een pi-filter.

### Antenne-aanpassing

Met een antenne-aanpassing kunnen voor het zenden ook antennes gebruikt worden die niet de juiste lengte hebben en dus niet vanzelf resoneren. Alleen een optimale dipoolantenne heeft een reële voepuntsweerstand. In alle andere gevallen is sprake van een hogere impedantie met een extra capacieve of inductieve schijnimpedantie.

Met een pi-filter (**figuren 13 en 14**) kan vrijwel elke antenne worden aangepast. Het filter is opgebouwd uit twee draaicondensatoren en een spoel met aftakkingen, waarmee met een kortsluitjumper de zelfinductie kan worden gevarieerd. Afhankelijk van de instelling kan de spanning aan de uitgang hoger of lager zijn dan aan de ingang, waardoor zowel hoog- als laagohmige antennes aangepast kunnen worden. Het pi-filter transformeert de impedantie van de antenne en compenseert overblijvende blindweerstanden.

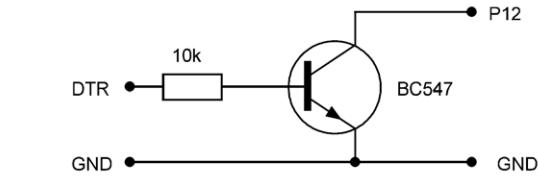
Tegelijkertijd fungeert het pi-filter ook als laagdoorlaatfilter en dempt het de harmonischen van het zendsignaal. Voor een optimale filtering moet worden gestreefd naar een zo hoog mogelijke kwaliteitsfactor. Dat is niet op alle frequenties haalbaar, omdat het filter direct aan de  $50\text{-}\Omega$ -uitgang van de eindtrap is aangesloten. Daarom is een breedband-transformator met een wikkelperhouding van 1:4 toegevoegd. De uitgangsspanning wordt omhoog getransformeerd en is dan optimaal aangepast aan het pi-filter. Hiermee kan de WSPR-zender ook op 80 m, 60 m en 40 m worden gebruikt. Als afstemhulp is een oscilloscoop gebruikt om het antennesignaal op de hoogste spanning in te stellen. Dat biedt tevens het voordeel dat grote fouten in de filtering direct worden onderkend. In dat geval namelijk is

een duidelijke afwijking van de sinusvorm waarneembaar, die duidt op de aanwezigheid van harmonischen.

Met deze kleine WSPR-zender is bij 200 mW met succes op alle banden gewerkt van 80 t/m 20 meter. De auteur kon hiermee bijna heel Europa bereiken en afstanden overbruggen tot 3000 km (**figuur 15**). Bij andere experimenten werden met net zulke kleine vermogens ook stations in de USA, Canada en Australië bereikt. Daartoe moet de zender lang genoeg op dezelfde frequentie blijven zenden. Bij steeds wijzigende propagatie-condities is er dan altijd wel een kort moment waarin een succesvolle verbinding kan worden gemaakt. Een test met slechts 10 mW op 40 m resulteerde in afstanden van 600 km.

### CAT-sturing

Het programma WSPR 2.0 is voorbereid voor de CAT-aansturing (Computer Aided Transceiver) van een hele reeks amateur-apparatuur met geschikte interfaces. Daarbij wordt eerst de gewenste frequentie ingesteld waarna de transceiver vervolgens naar zenden wordt omgeschakeld. De software genereert de modulatie in het bereik van 1400...1600 Hz en stuurt daarmee de microfooningang van de SSB-zender aan. In het menu *Setup/Station-Parameters* kunnen alle daarvoor benodigde instellingen en parameters worden ingevoerd (**figuur 16**). De volledig automatische besturing van het SDR-shield is niet zonder verdere uitbreidingen mogelijk, omdat het hier geen SSB-zender betreft. Maar een automatische start kan ook via de PTT-methode worden bereikt. De instelling *DTR* met COM-



Figuur 17. Start-interface.

poort COM1 bewerkstelligt dat tijdens elke zendactie de uitgang DTR op de seriële interface hoog wordt, waarmee een zender kan worden ingeschakeld. Daarvoor is wel een eenvoudige inverter vereist (**figuur 17**) die de start-ingang P12 van de Arduino aanstuurt.

Met deze verbinding neemt de WSPR-software de besturing van de zendfase over. Het eigen signaal wordt dan niet meer ontvangen. Bovendien zijn de intervallen tussen de zendfasen weliswaar instelbaar, maar tevens willekeurig gespreid. Hiermee wordt voorkomen dat tegenstations gedurende langere tijd steeds gelijktijdig zenden en dus hún tegenstation niet kunnen ontvangen. ▶

180709-04

### Weblinks

- [1] Weak Signal Propagation Reporter Network:  
<http://wsprnet.org/>
- [2] WSPR 2.0-software:  
<https://physics.princeton.edu/pulsar/K1JT/wspr.html>
- [3] Elektor SDR-Shield 2.0 (2), Elektorlabs september/oktober 2018: [www.electormagazine.nl/180284-03](http://www.electormagazine.nl/180284-03)
- [4] SDR Hands-on Book (met software-archief):  
[www.elektor.nl/sdr-hands-on-book](http://www.elektor.nl/sdr-hands-on-book)
- [5] JTEncode-library:  
<https://github.com/etherkit/JTEncode>
- [6] Software SDR#: <https://airspy.com>

**IM ELEKTOR-STORE**

→ Elektor SDR-Shield 2.0  
[www.elektor.nl/170515-91](http://www.elektor.nl/170515-91)

→ SDR Hands-on Book  
[www.elektor.nl/sdr-hands-on-book](http://www.elektor.nl/sdr-hands-on-book)

Advertentie



**ONLINE ASSEMBLAGE VAN  
ELEKTRONISCHE PRINTPLATEN**

[www.emsproto.com](http://www.emsproto.com)

**ONLINE**  
PRIJSOPGAVE  
VAN UW  
ELEKTRONISCHE  
PRINTPLAAT

**LEVERTIJD VAN  
2 tot 12  
DAGEN**

**HOEVEELHEID VAN  
1 tot 50  
PRINTPLATEN**

# Hover-catamaran met ingebouwde MQTT-broker



**Walter Trojan** (Duitsland)

Omgaan met IoT-hard- en software leer je het gemakkelijkst en intensiefst door er op een speelse manier mee aan de slag te gaan. De auteur had een idee, waarbij de kleinkinderen zelf voor de mechanische opbouw van het project konden zorgen: een catamaran met hover-aandrijving! Een WLAN en het MQTT-protocol zijn heel geschikt om de commando's van de afstandsbesturing over te dragen. We laten in dit artikel ook zien dat zelfs een vrij kleine controller zoals de ESP8266 de taak van MQTT-broker kan vervullen.

"Opa, jij hebt al zoveel gebouwd, bouw nog eens iets voor ons!" Deze oproep van mijn kleinkinderen onderbrak mijn mijmeringen over spraakbesturing, kunstmatige intelligentie en dergelijke zaken

en zette me weer met beide benen op de grond. Het was niet gemakkelijk om meteen met iets nieuws te komen: ik heb al robots, verkeerslichten en andere dingen voor mijn kleinkinderen gebouwd.

Ik wierp eens een blik in mijn knutselkist en daar vielen me enkele ventilatoren en een afgedankte afstandsbediening van een neergestorte helikopter op. Toen kwam ik op een idee waarbij de

kleinkinderen zelf voor de mechanische opbouw van het project konden zorgen: een catamaran met hover-aandrijving! En ik wilde er zelf ook iets aan overhouden, bijvoorbeeld nieuwe kennis en ervaring. Omdat ik pas iets had gelezen over een MQTT-broker op de ESP8266, leek dit me een leuk moment om dat eens uit te proberen. Ja, dat leest u goed: een MQTT-broker op de kleine ESP8266 met zijn uiterst beperkte geheugen. Zou dat kunnen? Leest u verder!

### Het eenvoudigste was de mechanische opbouw

De opbouw was werkelijk geen grote klus: we plakten twee middelgrote PET-flessen met smeltlijm ('hotmelt') aan een stukje triplex en vormden zo een romp met twee drijvers. Als aandrijving gebruikten we twee 5V-ventilatoren, die ook gewoon met het lijmpistool werden bevestigd. De voeding komt uit twee LiPo-accu's van het type 18650 met een capaciteit van elk 3400 mAh. Dat is voldoende energie om lang te kunnen spelen. Het kaartje met de ESP8266 is in het midden gemonteerd, naast de batterijhouder. In de koptoefoto ziet u het

### Eigenschappen

- Communicatie tussen catamaran en afstandsbediening voor het uitwisseling van commando's en statusinformatie met behulp van MQTT-protocol
- Eigen WLAN voor onafhankelijkheid van de locatie, eigen MQTT-broker
- Comfortabele besturing met een (omgebouwde) console
- Bewaking van de batterijspanning van catamaran en console
- Gebruik van de Arduino-IDE als eenvoudige ontwikkelomgeving voor de software
- Eenvoudige en goedkope mechanische opbouw

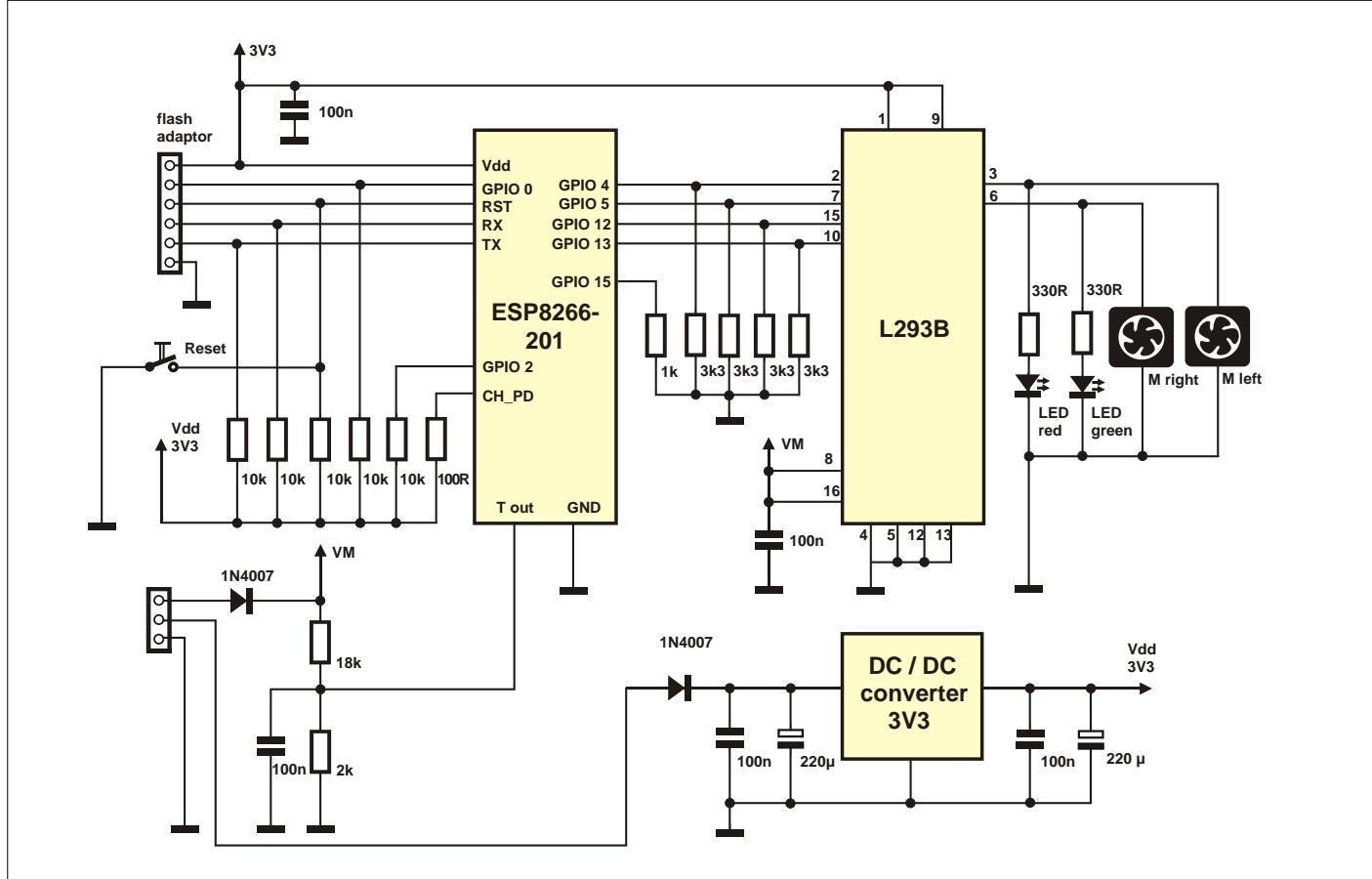
complete vaartuig en daar blijkt wel uit dat technisch geavanceerd speelgoed ook met eenvoudige middelen is te realiseren. Het vehikel drijft uitstekend en wordt met de krachtige ventilatoren behoorlijk vlot (alleen vooruit) aangedreven.

### Catamaranbesturing met het universele genie ESP8266

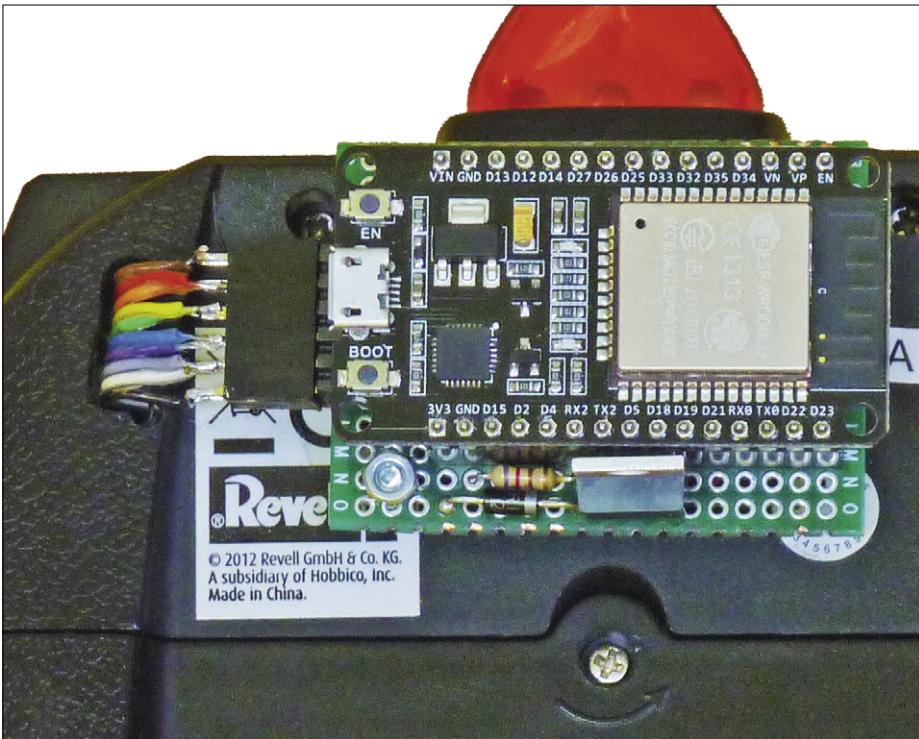
In dit project is de ESP8266 de aanvoerder in het geheel, terwijl zijn grote broer de ESP32 in de console alleen maar assisteert. Beide chips zijn afkomstig van de Chinese halfgeleiderfabrikant Espressif en hebben één of twee 32-bits processorkerken aan boord plus de nodige periferie.

Omdat beide µC's in Elektor al in veel projecten zijn gebruikt, hoef ik ze hier niet uitgebreid te beschrijven. Er is ook volop literatuur over beschikbaar op het internet [1]. De ESP8266 in de bouwvorm 201 bevindt zich op een klein break-out board waarop veel van de aansluitingen van de µC naar buiten zijn gevoerd. Dit board is dan weer op een kleine draagprint gemonteerd, samen met de regelaar en de driver.

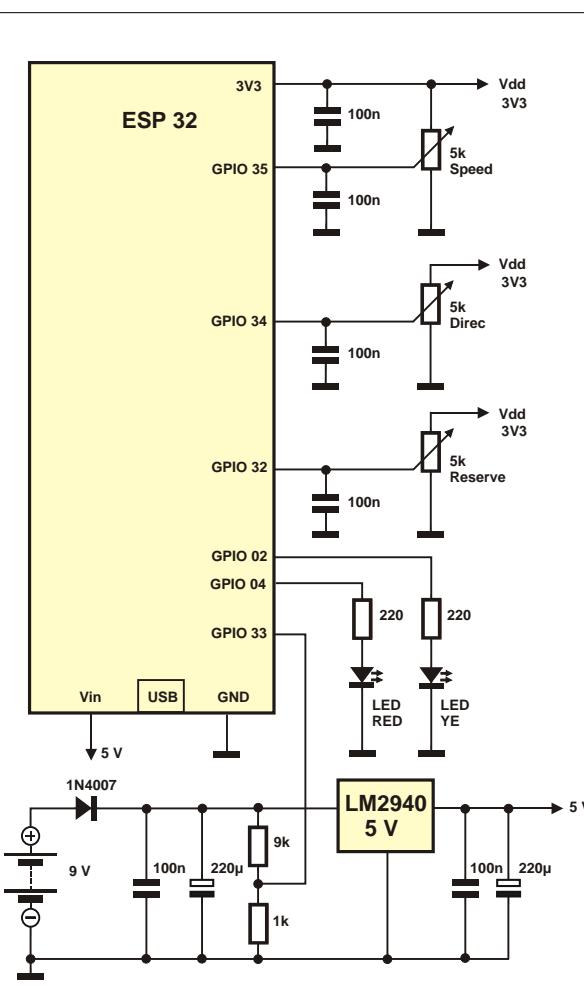
In **figuur 1** zien we de elektronica die op de catamaran is aangebracht. De ESP8266 bestuurt de catamaran-ventilatoren, zet een eigen WLAN op en herbergt ook nog een compacte MQTT-broker.



Figuur 1: De elektronica op de hover-catamaran.



Figuur 2: De ESP32-module is aan de buitenkant van de RC-afstandsbediening aangebracht.



Figuur 3: De catamaran wordt met twee potmeters bediend. De derde is gereserveerd voor toekomstige toepassingen.

Omdat de ventilatoren bij vollast tot wel 0,5 A stroom trekken, verzorgt een krachtige vermogensdriver van het type L293B de aansturing. Van de vier drivers in dit IC worden er hier maar twee gebruikt, de twee andere zijn beschikbaar als reserve (bijvoorbeeld voor een brugschakeling voor vooruit/achteruit). Parallel aan de ventilatoren zijn LED's aangesloten zodat we de aandrijving goed kunnen zien. De ventilatormotoren worden rechtstreeks gevoed uit de accuspanning VM van 7...8,4 V, verminderd met de verliezen in de beschermingsdiode en in de driver. Om te zorgen dat er bij het inschakelen gedefinieerde niveaus op de besturingslijnen staan, zijn de GPIO-besturingslijnen 4, 5, 12 en 13 uitgerust met pulldown-weerstanden van 3,3 kΩ.

De accuspanning wordt continu bewaakt. Hij gaat via een 10:1-spanningsdeeler naar de enige analoge ingang van de ESP8266. Omdat het meetbereik 0...1 V bedraagt, kunnen met deze deler spanningen tot 10 V worden gemeten. Ook andere aansluitingen van de µC zijn geconfigureerd voor onze toepassing: CH\_PD en GPIO 2 liggen aan Vdd, GPIO 15 aan GND. De aansluitingen die nodig zijn voor het programmeren, RST, GPIO 0, TX en RX, zijn beschikbaar op een zespolige flash-connector en zijn via pullup-weerstanden aan Vdd gelegd. De voedingsspanning voor de controller wordt tot 3,3 V teruggebracht en gestabiliseerd met een lineaire regelaar van het type LM2940. Ontkoppelcondensatoren voor en achter de regelaar dempen de storingen die worden veroorzaakt door het schakelen van de motoren. Beschermingsdiodes aan de batterijansluiting voorkomen dat de elektronica wordt verwoest bij een verkeerde polariteit.

### Comfortabele bediening met gemodificeerde console

De afgedankte helikopter-afstandsbediening bewijst in dit project nog goede diensten. De oorspronkelijke elektronica was bedoeld om te werken met infrarood licht en niet bruikbaar voor dit project, maar het batterijvak, de drie besturingspotmeters, een dubbele LED met de kleuren geel en rood en een aan/uitschakelaar vormen een uitstekende, comfortabele 'gebruikersinterface' voor de hover-catamaran.

Voor de communicatie met de catamaran hebben we een WLAN-verbinding en een MQTT-client nodig. De ESP32 is daar-

voor bij uitstek geschikt. Maar waarom is hier niet ook gekozen voor de kleinere broer, de ESP8266, die op de catamaran zelfs nog aanzienlijk meer te doen heeft? Dat ligt gewoon aan het aantal analoge ingangen: de ESP8286 heeft er maar één, maar de ESP32 heeft er meerdere. Omdat de meeste ruimte in de behuizing van de console wordt ingenomen door het batterijvak (6xAA), moest de ESP32 er aan de buitenkant tegenaan geflanst worden. Zoals u ziet in **figuur 2** is de kleine print van de ESP32-ontwikkelmodule met USB-programmeeraansluiting via een flatcable verbonden met de in de console ondergebrachte componenten. Het schema (**figuur 3**) is niet erg ingewikkeld. Omdat de console voorzien is van een snelheidsregelaar en een stuurnuppel met twee potentiometers, zijn om te beginnen de lopers op analoge ingangen aangesloten. Voor de besturing van de catamaran zijn alleen de snelheidsregelaar en de links/rechts-regelaar nodig, de niet gebruikte vooruit/achteruit-regelaar van de stuurnuppel dient als reserve en kan misschien nog nuttig zijn voor andere projecten. De analoge ingangen hebben een meetbereik van 0...3,3 V en dat is ook het bereik van de loperspanningen van de potmeters. De accuspanning gaat via een 10:1-deler naar de analoge ingang GPIO 33 (ADC1\_CH5).

De twee digitale uitgangen GPIO 02 en GPIO 4 sturen de dubbele LED aan. De gele dient als toestandsindicator en de rode dient als waarschuwing bij een zwakke batterij in de catamaran of in de console.

Voor de voeding dient ook hier de lineaire regelaar LM2950, maar nu met een uitgangsspanning van 5 V. Een tweede regelaar, op de ontwikkelmodule van de ESP32, maakt daar dan weer 3,3 V van. Dubbel geregeld is ook dubbel stabiel!

### Een goed begin is het halve werk!

Voor het ontwikkelen van de project-software is de bekende en alom geliefde Arduino-IDE gebruikt. Deze ontwikkelomgeving is niet alleen comfortabel, maar garandeert ook toegang tot veel bibliotheken en toepassingsvoorbellen. Gelukkig worden de µC's ESP8266 en ESP32 van Espressif dankzij additionele core-modules door deze IDE ondersteund. Voordat kan worden begonnen met het ontwikkelen van de software zijn dus de volgende voorbereidingen nodig:

- installatie van de Arduino-IDE: actuele versie downloaden van [2] en de IDE installeren;
- installatie van de ESP8266-core: onder [3] is gedocumenteerd, hoe de extra module in de IDE kan worden geïntegreerd;
- installatie van de ESP32-core: ook hiervoor bestaat een heel goede installatiehandleiding onder [4].

Als de IDE en beide cores geïnstalleerd zijn, moeten er nog bijkomende bibliotheken worden toegevoegd. Dit wordt beschreven in de volgende paragrafen over de besturingsprogramma's.

### MQTT in het kort

Natuurlijk had de communicatie tussen catamaran en console kunnen werken via eenvoudige TCP-pakketten, maar het was veel interessanter om een echte MQTT-broker te realiseren op de kleine ESP8266. En dat brengt nog een voordeel met zich mee: we kunnen een MQTT-client starten op een PC en de complete gegevensuitwisseling bekijken.

Als u helemaal niets weet over MQTT, kunt u het beste het **tekstkader** even doorlezen.

De broker is de besturingscentrale van het MQTT-berichtenprotocol. Er zijn heel veel MQTT-brokers, zowel commerciële als open source-oplossingen zoals Mosquitto. Maar om MQTT te kunnen draaien op zo'n 'ondermaatse' µC als de ESP8266, is een speciale broker nodig, die vooral vanwege het beperkte geheugen maar een deel van de mogelijkheden van MQTT kan bieden. Toch ondersteunt de hier toegepaste ESP8266-broker:

- protocolversies MQTT v3.1 en v3.1.1 tegelijk;
- maximaal acht aangesloten clients;
- kwaliteitsniveau 0;
- testament met laatste wil;
- klaarzetten van berichten ook voor nieuw aangesloten clients (*retained*);
- authenticatie met behulp van naam en wachtwoord.

Dat is al heel netjes en we treuren dus niet dat we:

### MQTT in het kort

Het door de firma IBM ontwikkelde, maar nu licentievrije protocol MQTT (*Message Queuing Telematic Transport*) heeft, in tegenstelling tot het bekende HTTP met request/response-methodiek, een publish/subscribe-architectuur, waarbij een centrale broker (server) wordt gebruikt. Apparaten die iets te melden hebben, zenden (publiceren) hun berichten naar de broker, die deze berichten doorstuurt aan andere apparaten die zich daarop hebben geabonneerd.

De communicatie kan in beide richtingen verlopen, een deelnemer kan dus publiceren en zich abonneren. Een knooppunt kan een eenvoudige microcontroller, een PC of een Linux-server zijn. Op het knooppunt moeten daarvoor een TCP-stack en het MQTT-protocol geïmplementeerd zijn.

De adressering bij het zenden en ontvangen van berichten werkt via zogenaamde topics. Dat zijn strings die een soort onderwerp van de berichten weergeven, maar zijn opgebouwd als een URL. Een temperatuursensor in de werkkamer zou bijvoorbeeld zijn huidige temperatuur kunnen publiceren op een topic als 'MijnHuis/Werkkamer/Temperatuur'. Naast het topic worden de nuttige data (payload) en verdere parameters overgedragen. De data kan worden overgebracht in de vorm van binaire waarden, teksten of zelfs XML- of JSON-structuren.

De kwaliteit van de overdracht van de berichten kan op drie niveaus gewaarborgd zijn:

Niveau 0: Geen kwaliteitsbewaking onder het motto: 'shoot and forget'.

Niveau 1: Gegarandeerde overdracht van minstens één bericht, daarbij zijn duplicaten niet uitgesloten.

Niveau 2: Gegarandeerde overdracht van exact één bericht zonder duplicaten.

Hebt u al eens gehoord van een protocol met een testament? Ook dat is met MQTT mogelijk. Voor het geval dat een knooppunt een keer uitvalt, kan het bij de broker een 'laatste wil' deponeren, die na zijn overlijden moet worden uitgevoerd. Dat kan bijvoorbeeld een melding aan de systeembeheerder zijn.

Er is nog veel meer te vertellen over MQTT: filtering, meertraps beveiligingsstrategie, bescherming tegen uitval enz. enz. U kunt het beste eens gaan kijken op internet en in het (Duitstalige) MQTT-praktijkboek van de auteur [10]!

### **Listing 1. Initialisering.**

```
#include
#include
#include "Ticker.h"
#include

#define speed_TOPIC      "speed"
#define direc_TOPIC     "direc"
#define katama_TOPIC    "katama"

const char* ssid = "ESP_HOST";
const char* password = "";
const char* mqtt_server = "192.168.4.1";
```

- overdrachtskwaliteiten 1 en 2 voor gegarandeerde levering;
- betrouwbare versleuteling met TLS (*Transport Layer Security*);
- en permanente opslag van publicaties (*non-clear sessions*)

moeten missen. De ontwikkelaar van deze gratis software [5], martin-ger, heeft de ESP8266 gekozen als platform omdat deze µC wordt gebruikt in de goedkope Sonoff-schakelaars en hij daarmee een extra PC of Raspberry Pi als host voor de broker kan uitsparen. Hij is heel efficiënt, volgens zijn metingen haalt de broker ongeveer 100 publicaties/subscripties per seconde. Chapeau Martin!

### **Listing 2. MQTT-relevante instructies.**

```
WiFiClient espClient; // Definities: WiFi- en MQTT-client
PubSubClient client(espClient);

client.setServer(mqtt_server, 1883);
  // Setup: Broker-adres en poort
client.setCallback(receivedCallback);
  // Setup: Start callback voor abonnement

if (!client.connected()) {
  // Setup: Verbinding met de Broker
  mqttconnect(); }

client.loop();      // Loop: aanroep van de MQTT-client en
send_MQTT();        // uitvoering van vereiste activiteiten

client.publish(speed_TOPIC, spdmqt);
  // Loop: publicatie van de potmeterdata
delay(100);
client.publish(direc_TOPIC, dirmqt);
delay(100);

// Callback-Funktion
void receivedCallback(char* topic, byte* payload, unsigned int
length)
```

Dit zijn de gebruikte MQTT-commando's:

De console publiceert de data van de stuurknuppels onder de topics 'speed' en 'direc'. Hij doet dat telkens als een stand verandert, of eenmaal per tien seconden. De payload bevat ook de topic-aanduiding, maar die dient alleen om de communicatie beter te kunnen volgen. De catamaran verzendt zelf herhaaldelijk zijn accuspanning. Die wordt in de console geëvalueerd en bij te lage spanning wordt een bescheiden alarm geslagen.

### **Software van de console**

De software voor de besturingsconsole is ook gerealiseerd in de vorm van een Arduino-sketch. Deze verzendt via MQTT de momentele positie van de stuurknuppels voor snelheid en richting naar de catamaran. Daarnaast controleert hij continu de batterijspanning van de console en de catamaran en meldt een te lage spanning door de rode LED te laten knipperen. De gele LED knippert één keer per seconde als statusindicator.

De software heet **Konsole.ino** en de benodigde bibliotheken zijn beschikbaar op de projectpagina bij dit artikel [6]. Kies in de Arduino-IDE onder 'Hulpmiddelen Board' uw ESP32-module (bijvoorbeeld 'ESP32 Dev Module') en stel het poortnummer en de overdrachtssnelheid in (die laatste moet 115200 bit/s bedragen). De MQTT-client bevindt zich in het bestand 'pubsubclient-master.zip' en kan via 'Schets Bibliotheek gebruiken Voeg .ZIP-bibliotheek toe...' worden geïnstalleerd. Verder moeten de bestanden 'Ticker.h' en 'Ticker.cpp' zich in uw projectmap bevinden.

De programmastructuur is weergegeven in **figuur 4** en komt overeen met het

### **Dies sind die verwendeten MQTT-Kommandos:**

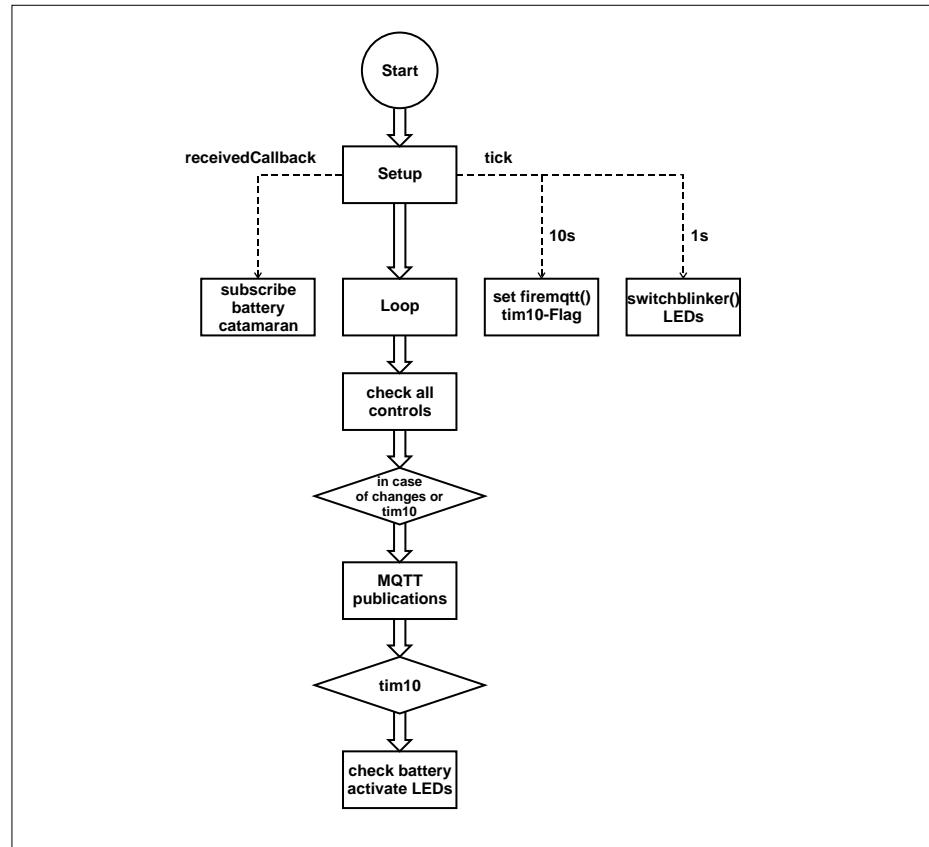
Topic	Payload	Werte
speed	speed_xxx	speed_000 = min, speed_100 = max
direc	direc_xxx	direc_000 = links, direc_100 = geradeaus, direc_200 = rechts
vbatt	vbatt_xxx	vbatt_999 = 9,99 V

normale Arduino-model met setup- en loop-routine. Om de loop-routine niet te blokkeren met het wachten op een inkomend MQTT-bericht, is het abonnement voor de catamaran-accuspanning ondergebracht in de callback-functie `receivedCallback()`. Als er een melding binnenkomt, wordt deze functie actief en evalueert hij het bericht. Er zijn ook twee routines die elke tien respectievelijk één seconde worden aangeroepen door timers van de ticker-bibliotheek. De functie `firemqtt()` stuurt minstens één keer per tien seconden een publicatie van de stuurknuppeldata uit, terwijl `blinker()` de LED's in een secondenritme aanstuurt. De initialisatie gebeurt, zoals gewoonlijk, in de setup-functie. Daarbij wordt onder meer de verbinding met het WLAN gemaakt, de MQTT-client met verbinding met de broker gestart en worden de callback-functies vastgelegd.

In de oneindige lus worden om te beginnen de analoge waarden van de stuurknuppels bepaald. Veranderingen worden meteen gepubliceerd naar de catamaran. Als de stuurknuppels niet zijn bewogen, bepaalt het tickerinterval wanneer de data wordt verstuurd. In deze lus worden ook de meting en verwerking van de batterijspanningen uitgevoerd. (En als het nodig is, wordt de rode waarschuwing-LED geactiveerd. Die knippert dan om-en-om met de gele statussindicator).

Aan het begin van het programma (**listing 1**) worden headerfiles van de bibliotheken voor WiFi, MQTT en de tickers voor de timer-interrupt meegenomen. Daarna volgen MQTT-topics en de inloggegevens voor het eigen WLAN van de catamaran en de broker. Er is geen actieve toegangscontrole: het netwerk heeft als SSID *ESP\_HOST* en gebruikt geen wachtwoord. Bij een professioneel project zouden we natuurlijk wel een wachtwoord gebruiken, letten op goede versleuteling en de toegang tot de broker beveiligen met een naam en een wachtwoord. Het IP-adres van de MQTT-server ligt vast op *192.168.4.1*.

Dan volgen de instructies voor MQTT (zie **listing 2**). De publicaties in de loop worden uitgevoerd met een tussentijd van 100 ms, zodat de software in de catamaran niet overbelast wordt. Zo krijgt hij maximaal tien berichten per seconde, wat de broker gemakkelijk aankan. De callback-functie levert alle nodige data over de ontvangen subscriptie zoals topic, payload en de lengte van de payload.



Figuur 4: Het console-programma met de typische Arduino-structuur.

### **Listing 3. De beide tickers.**

```

Ticker minmqtt; // Definitie: toewijzing van de tickers
Ticker ledblink;

ledblink.attach_ms(1000, blinker); // Setup: tijdinterval en
service-functie
minmqtt.attach(10, firemqtt);

void firemqtt(){ // Funktie: firemqtt()
tim10 = true;
}

```

### **Weblinks**

- [1] ESP32/ESP8266-compilatie: [www.elektor.de/esp32-esp8266-kompilation-de](http://www.elektor.de/esp32-esp8266-kompilation-de)
- [2] Arduino-IDE: [www.arduino.cc/en/Main/Software](http://www.arduino.cc/en/Main/Software)
- [3] Start ESP8266: [www.heise.de/ct/artikel/Arduino-IDE-installieren-und-fit-machen-fuer-ESP8266-und-ESP32-4130814.html](http://www.heise.de/ct/artikel/Arduino-IDE-installieren-und-fit-machen-fuer-ESP8266-und-ESP32-4130814.html)
- [4] Start ESP32: [www.elektrormagazine.com/labs/esp32-getting-started](http://www.elektrormagazine.com/labs/esp32-getting-started)
- [5] martin-ger: <https://github.com/martin-ger>
- [6] Projectpagina bij dit artikel: [www.elektrormagazine.nl/170198-04](http://www.elektrormagazine.nl/170198-04)
- [7] Broker Arduino: <https://github.com/martin-ger/uMQTTBroker>
- [8] Broker C: [https://github.com/martin-ger/esp\\_mqtt/](https://github.com/martin-ger/esp_mqtt/)
- [9] Broker op Youtube: [www.youtube.com/watch?v=0K9q4IuB\\_oA](http://www.youtube.com/watch?v=0K9q4IuB_oA)
- [10] MQTT-Praxisbuch: [www.elektor.de/das-mqtt-praxisbuch](http://www.elektor.de/das-mqtt-praxisbuch)

#### Listing 4. Details van de catamaran-software.

```
#include
#include "uMQTTBroker.h"      // MQTT-Broker met client
#include "Ticker.h"
#include

char ssid[] = "ESP_HOST";    // SSID (naam) van het catamaran-WLAN
char pass[] = "";           // WLAN-wachtwoord, hier geen
bool WiFiAP = true;         // moet een eigen WLAN ingericht worden?

#define speed_TOPIC "speed" // Definitie van de topics
#define direc_TOPIC "direc"
#define katama_TOPIC "katama"

myMQTTBroker myBroker;       // Definitie van de Broker-instantie
```

De beide tickers worden gedefinieerd bij het begin van het programma (**listing 3**). In de setup krijgen ze een tijdsinterval in milliseconden resp. seconden en een service-functie toegewezen. Deze interruptfuncties moeten zo kort mogelijk zijn en zo weinig mogelijk verwerkingsstijd kosten. In de functie `firemqtt()` wordt alleen een vlag gezet, die in de loop wordt geëvalueerd.

Alle verdere details van het programma zijn te vinden in de van commentaar voorziene broncode. Let op de `Serial.print`-uitvoer, waarmee u het verloop van het programma kunt volgen in de seriële monitor van de Arduino-IDE.

#### De software van de catamaran

Ook voor het catamaran-programma (**figuur 5**) zijn vooraf enkele instellingen in de Arduino-IDE nodig. Onder *Hulpmiddelen Board* kiezen we de juiste ESP8266-module uit (hier *Generic ESP8266 Module*) en stellen we het poortnummer en de overdrachtssnelheid (115200 Baud) in. Bij de reset-methode kunt u het beste *nodemcu* kiezen, zodat u bij het flashen geen knopjes hoeft in te drukken. Het is ook belangrijk om de parameter *LwIP Variant* in te stellen op *v1.4 Higher Bandwidth* om de TCP-stack zo 'krachtig' mogelijk te maken.

De broker-software voor de Arduino-IDE is beschikbaar onder [7], er is ook een C-versie te vinden onder [8]. Het is aan te raden om eerst het YouTube-filmpje [9] van de ontwikkelaar te bekijken. Daarin wordt een uitstekende beschrijving van de werking van de broker gegeven. De software zit al in de Elektor-map [6], die hoeft u alleen maar naar uw projectmap te halen; installeren in de Arduino-IDE is niet nodig. Het IP-adres van de MQTT-server is standaard ingesteld op 192.168.4.1.

Het programma voor het besturen van de catamaran staat in de projectmap onder de naam **Katamaran.ino**. De structuur lijkt op die van de console-software. Ook hier worden in de setup callback- en ticker-routines gestart. De functie `onData()` wordt steeds geactiveerd als er een bericht van het console binnenkomt of als het default-interval van één van de ticker-functies is afgelopen.

Omdat de toegepaste ventilatoren bij de gebruikelijke PWM-frequenties maar heel langzaam draaiden, is in dit project gekozen voor een besturing met een heel lage frequentie van 1 Hz. Ze worden dus

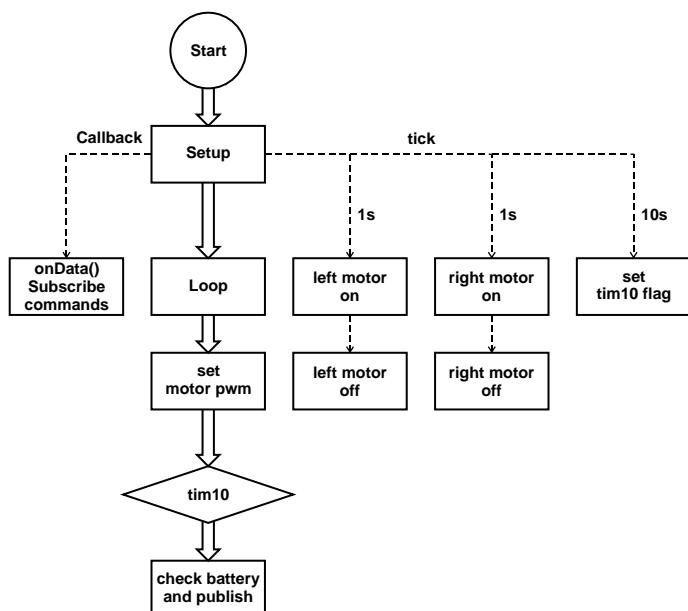
#### Listing 6. WiFi- en MQTT-instructies.

```
if (WiFiAP) // Eigen WLAN ...
startWiFiAP();
else
startWiFiClient(); // of deelname aan thuisnetwerk

myBroker.init(); // Setup: initialisering van de Broker

myBroker.subscribe(speed_Topic);
// Abonnement snelheid en richting
myBroker.subscribe(direc_Topic);

myBroker.publish(katama_TOPIC, batmqt);
// Loop: publicatie van de batterijspanning
```



Figuur 5: Het stroomdiagram voor de catamaran-besturing.

elke seconde ingeschakeld en na een van ingestelde snelheid afhankelijke tijd weer uitgeschakeld. De massatraagheid van de schoepen van de ventilator zorgt daarbij voor een vloeiend verloop.

Er is ook een ticker die elke tien seconden zorgt voor meting en publicatie van de accuspanning. In de loop worden de schakelintervallen berekend en doorgegeven aan de ticker-functies.

In **listing 4** zijn enkele interessante gedeelten van de software te zien. Bij de definities worden de *WiFiClient*-bibliotheek, de MQTT-broker en de ticker-functies aangeroepen. Als naam voor



## Kies het ruime sop met MQTT

het WLAN van de catamaran wordt *ESP\_HOST* opgegeven. Er wordt geen wachtwoord ingesteld, dus er is geen beveiliging. Door *WiFiAP* op *true* te zetten, geven we aan dat er een eigen access point moet worden opgezet.

De MQTT-broker heeft drie belangrijke callback-functies (**listing 5**):

- **onConnect**: wordt geactiveerd bij het aansluiten van een nieuwe client

en geeft het IP-adres en het aantal ingeschakelde clients. Daarmee kunnen we beslissen, of deze client welkom is en of we hem van dienst kunnen zijn.

- **onAuth**: geeft bij het aansluiten met een nieuwe client de naam en het wachtwoord, als dat is ingesteld. Als *true* wordt teruggegeven, wordt de client toegelaten. In dit project wor-

### **Listing 5. Callback-functies van de MQTT-broker.**

```
class myMQTTBroker: public uMQTTBroker
{
public: // Een client logt zich in
virtual bool onConnect(IPAddress addr, uint16_t client_count) {
Serial.println(addr.toString() + " connected");
return true; // true = toegestaan, false = afgewezen
}
// Authenticatie van de client
virtual bool onAuth(String username, String password) {
Serial.println("Username/Password: "+username+"/"+password);
return true; // true = toegestaan, false = afgewezen
}
// ontvangst van een abonnement
virtual void onData(String topic, const char *data, uint32_t length) {
char payload[length+1];
char payval[4] = "000";

os_memcpy(payload, data, length);
payload[length] = '\0';
if(topic == speed_TOPIC){ // Speed-commando ontvangen
payval[0] = payload[6];
payval[1] = payload[7];
payval[2] = payload[8];
spdval = atoi(payval);
if((spdval >= 0) && (spdval <= 100)) // toegestaan bereik 0..100
spdok = true;
}
if(topic == direc_TOPIC){ // richting-commando ontvangen
payval[0] = payload[6];
payval[1] = payload[7];
payval[2] = payload[8];
dirval = atoi(payval);
if((dirval >= 0) && (dirval <= 200)) // toegestaan bereik 0..200
dirok = true;
}
Serial.println("received topic '" +topic+ "' with data '" + (String)payload+ "'");
}
};
```

den alle clients, met of zonder inloggegevens, toegelaten.

- **onData:** ontvangt alle geabonneerde topics en geeft naast de naam van het topic ook de payload en de lengte van de data. Dan wordt het numerieke deel van de payload overgedragen en geconverteerd naar de integer-variabelen `spdval` en `dirval`. Daarna wordt gecontroleerd, of deze waarde in het geldige bereik ligt en met behulp van een vlag doorgeven aan de motorsturing.

In **listing 6** zijn de relevante WiFi- en MQTT-instructies op een rijtje gezet. De eigenlijke motorsturing in **listing 7** werkt met vier tickers. De start-functies `Tlion` en `Treon` krijgen vaste startintervallen van 1000 ms toegewezen. De service-routines `Molian` en `Moreon` schakelen de ventilatoren in en starten de routines `Molioff` en `Moreoff` die de motoren weer uitschakelen als de berekende impulsijd is verstreken. Een vijfde ticker schakelt elke tien seconden de vlag `tim10` in om de batterijmeting te triggeren.

Het is gebleken dat het ontwikkelen van de software met de Arduino-IDE probleemloos in zijn werk ging. We hoeven dus niet altijd naar ingewikkelde ontwikkelsystemen te grijpen om kleine of middelware projecten uit te voeren.

## En wat vonden de kinderen ervan?

De kleinkinderen waren dolblij met de apparaten en zijn ze meteen uitgebreid gaan testen in een vijver. De catamaran reageerde goed en was tot op een afstand van ruim 50 m prima te besturen. Vanwege de hoge opbouw, het lage gewicht en de geringe diepgang is de catamaran wel wat (zij)windgevoelig. Natuurlijk zouden we het vaartuig mechanisch kunnen verbeteren en bijvoorbeeld kunnen uitrusten met scheepsschroeven.

Voor mij heeft dit project aangetoond dat de MQTT-broker op de ESP8266 goed en inspirerend te gebruiken is in kleine projecten. Maar het belangrijkste was dat ik veel plezier heb gehad aan het samenwerken met de kleinkinderen. Dat wens ik u ook toe bij uw projecten! ▶

(170198-04)

## Listing 7. De motorsturing.

```
// Setup: Definition der Ticker
Ticker Tlion;                                // Motor links aan
Ticker Tloff;                                 // Motor links uit
Ticker Treon;                                // Motor rechts aan
Ticker Treoff;                               // Motor rechts uit
Ticker Tick10;                               // Ticker 10 sec.

Tlion.attach_ms(1000, Molion);                // Start motor links elke seconde
Treon.attach_ms(1000, Moreon);                // Start motor rechts elke seconde
Tick10.attach(10, SendBatt);                  // Ticker 10 sec.

void Molioff(){                                // schakel motor uit
digitalWrite(MOLI,LOW);
}

void Molion(){                                 // schakel motor in
digitalWrite(MOLI,HIGH);
Tloff.attach_ms(molipow, Molioff);           // Start motor uit met
                                                // variabele tijd 0..1000
}

void SendBatt(){ // Zet ticker-flag
tim10 = true;
}
```

## Installatie-tips

### Inhoud van de software-map

`Konsole.ino`: Besturingsprogramma voor de console op de ESP32

`Katamaran.ino`: Besturingsprogramma voor de catamaran op de ESP8266 incl. MQTT-broker

`uMQTTBroker-master.zip`: Bibliotheek met MQTT-broker-functies voor Katamaran.ino

`pubsubclient-master.zip`: Bibliotheek met MQTT-client voor Konsole.ino

`Ticker.h / Ticker.cpp`: Bibliotheek voor de tickers

`mqtt-brok-E.zip`: C-versie van de MQTT-broker voor gebruik in de Espressif-IDE (niet nodig bij de Arduino-IDE)

### Installatie van de software

Arduino-IDE: ESP8266- en ESP32-core installeren

Catamaran: `uMQTTBroker-master.zip` als zip-bestand installeren in de Arduino-IDE

Console: `pubsubclient-master.zip` als zip-bestand installeren in de Arduino-IDE

### IN DE STORE

→ **Das MQTT-Praxisbuch**

[www.elektor.nl/das-mqtt-praxisbuch](http://www.elektor.nl/das-mqtt-praxisbuch)

→ **Esp32/Esp8266-Kompilation**

[www.elektor.nl/esp32-esp8266-kompilation-de](http://www.elektor.nl/esp32-esp8266-kompilation-de)



# Eenvoudige A/D-converter met PLD

## bouw met weinig moeite een sigma-delta-ADC

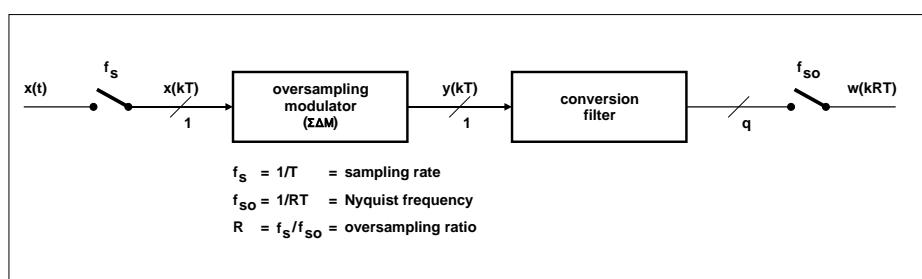
Guido Nopper (Duitsland)

Met PLD's is werkelijk van alles te maken. In dit artikel ziet u hoe u met slechts 26 macro-cellen een 8-bit A/D-converter kunt maken op basis van een sigma-delta-modulator ( $\Sigma\Delta M$ ), met daarachter een digitaal laagdoorlaat- en conversiefilter. Vervolgens beschrijven we de nauwkeurigheid, het oplossend vermogen en de hoeveelheid werk die ermee gemoeid is. Opmerkelijk is dat het digitale filter nog het meeste werk in beslag neemt.

Er zijn diverse manieren om een analoge signaal om te zetten in digitale waarden. De 'politiek correcte' benaming voor zo'n schakeling is analoog-digitale-omzetter, maar we komen hem meestal tegen onder de naam A/D-converter, van *analog/digital converter*. De ADC in dit artikel maakt gebruik van het principe van overbemonstering, meestal *oversampling* genoemd in technisch Engels (en Nederlands).

### Oversampling

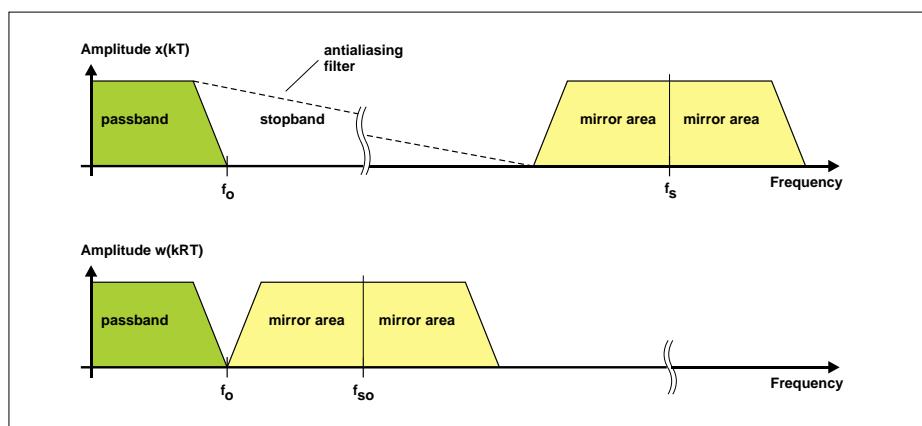
Oversampling houdt in dat het analoge ingangssignaal wordt afgetast met een aanmerkelijk hogere frequentie dan nodig zou zijn volgens het bemonsteringstheorema van Nyquist-Shannon, dat stelt dat de bemonsteringsfrequentie ten minste tweemaal die van de hoogste in het signaal aanwezige frequentiecomponent dient te zijn. In **figuur 1** ziet u het principe van zo'n ADC. De modulator levert een 1-bits-datastroombus met bitrate  $f_s$ . De amplituderesolutie is laag, slechts één bit. De frequentieresolutie is echter 'veel hoger dan nodig is'. De verhouding tussen de werkelijke oversampling-frequentie en de Nyquist-frequentie heet R of OSR, van *oversampling ratio*. Deze ligt vaak een factor ter waarde van meerdere machten van twee (32, 64, 128...) boven de hoogste te bemonsteren frequentie.



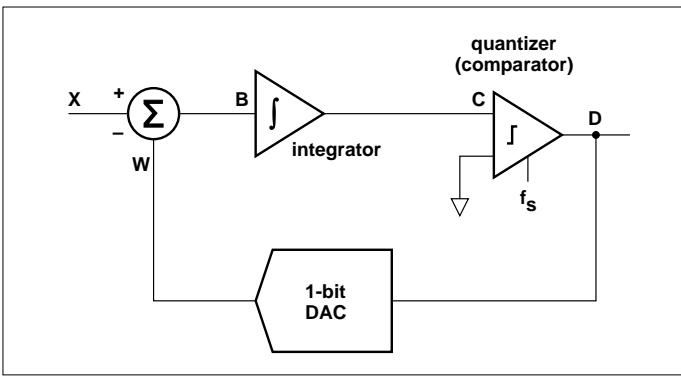
Figuur 1. ADC-principe met oversampling.

Het daaropvolgende conversiefilter moet de amplituderesolutie verhogen door een bepaald aantal bits in de datastroombus uit te middelen, wat dus neerkomt op de functie van een laagdoorlaatfilter. Daar-

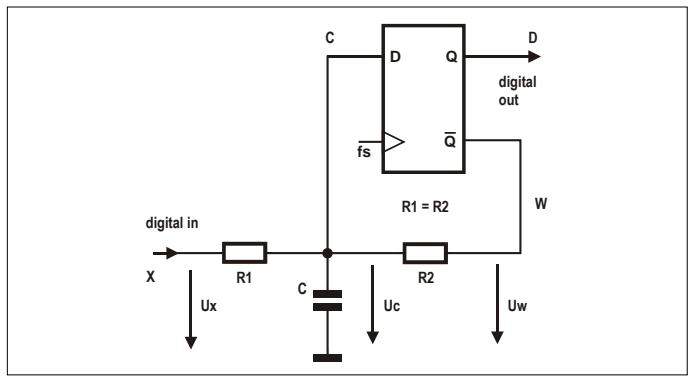
mee kan aansluitend de samplefrequentie verlaagd worden tot  $f_{so}$  zonder dat er al te veel ongewenste signaalcomponenten worden 'omgevouwen' naar binnen het gewenste frequentiebereik.



Figuur 2. Spectra van de signalen in figuur 1.



Figuur 3. Blokschema van een eenvoudige sigma-delta-modulator.



Figuur 4. De eenvoudigste sigma-delta-modulator.

In **figuur 2** ziet u de signaalspectra van figuur 1. Oversampling met de hogere  $f_s$  heeft het prettige neveneffect dat we het benodigde anti-aliasing-filter heel eenvoudig kunnen uitvoeren met een filter van lage orde, of dat filter zelfs helemaal weg kunnen laten. Doordat de samplefrequentie aan de uitgang wordt gereduceerd tot  $f_{so}$  kunnen er eventueel nog wel ongewenste signaalcomponenten vanuit de stopband in de doorkaatband omklossen.

### Sigma-delta-modulator

Een eerste-orde  $\Sigma\Delta$  (sigma-delta-modulator) bestaat uit een verschiltrap, een integrator, een quantiseerde en een 1-bit DAC. **Figuur 3** geeft het blokschema. Dit principe wordt ook wel delta-sigma-modulator genoemd, omdat het door de 1-bit DAC analog gemaakte uitgangssignaal op punt W wordt afgetrokken van het ingangssignaal op punt X; dus het verschil (de delta  $\Delta$ ), blijft over op punt B, en pas daarna volgt integratie (sommatie  $\Sigma$ ). De quantisator (com-

parator) krijgt naast het geïntegreerde verschilsignaal (punt C) ook nog de samplefrequentie  $f_s$ . Daardoor kan het uitgangssignaal alleen maar veranderen op elk sample-tijdstip en is het dus gequantiseerd met  $f_s$ . Dergelijke modulatoren bestaan al vele jaren [1].

Zo'n modulator is heel eenvoudig te realiseren met een D-flipflop, twee weerstanden en een condensator, zoals in **figuur 4**. Is de samplefrequentie  $f_s$  beduidend hoger dan de maximale signalfrequentie, dan spreken we van 'oversampling A/D-conversie'. De met hoofdletters aangeduide signalen van figuur 3 ziet u terug in figuur 4. Zo is ook te zien dat de eenvoud van de schakeling in wezen ten koste gaat van de nauwkeurigheid.

In de  $\Sigma\Delta$  in figuur 4 wordt het geïnverteerde digitale uitgangssignaal (punt W) met een RC-laagdoorlaatfilter van digitaal naar analog omgezet en opgeteld bij het ingangssignaal. Dankzij die inversie is dit een aftrekking, dus een vergelijking. De niveaus op punt W bepalen de

referentiespanningen voor de A/D-converters en moeten dus aan analoge eisen voldoen. Met een CMOS-flipflop halen we voor hoog en laag vrijwel de niveaus van VCC en GND.

De D-ingang van de flipflop fungeert als analoge comparotor. Het triggerniveau van de flipflop is daardoor maatgevend voor de nauwkeurigheid van het hele systeem. De spanning over de condensator ( $U_c$ , punt C) blijft altijd ongeveer op dat niveau. Zodra namelijk de ingangsspanning op punt X stijgt en  $U_c$  dus eveneens stijgt, geeft de flipflop naar verhouding meer laag-bits aan punt W, waardoor  $U_c$  weer daalt. Een  $\Sigma\Delta$  probeert dus met de gemiddelde waarde van zijn beide uitgangsniveaus de analoge ingangsspanning te volgen.

### Ingangsspanningsbereik en overdrachtsfunctie

Om ervoor te zorgen dat de  $\Sigma\Delta$  in figuur 4 de signaalamplitude niet afptoet, mag de ingangsspanning niet te hoog en niet te laag zijn. De flipflop-uitgang kan alleen maar hoog of laag zijn. Bij een maximale ingangsspanning  $U_x$  is flipflop-uitgang Q van de flipflop continu hoog en is zijn inverterende uitgang continu laag.

Als we de ingangsstroom aan de D-ingang van de flipflop en de reststroom van de keramische condensator buiten beschouwing laten, dan moeten de stromen door de beide weerstanden R1 en R2 gemiddeld even groot zijn. Dat betekent:

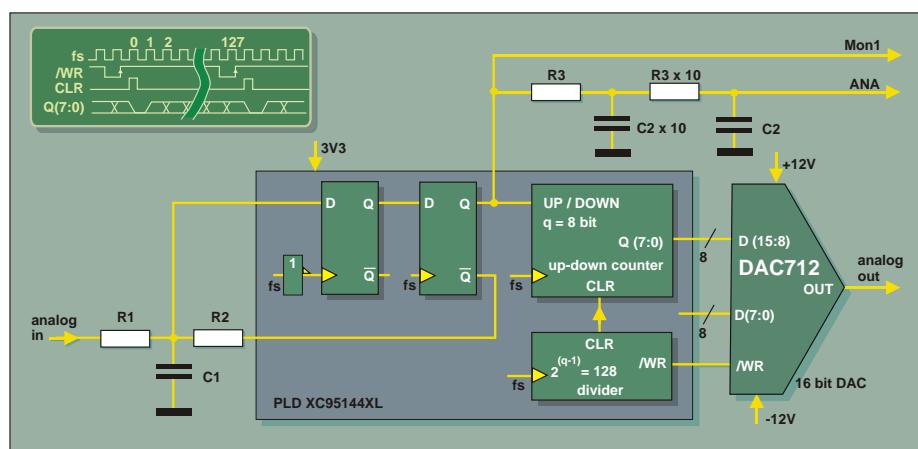
$$I_{R1} = I_{R2}$$

oftewel

$$\frac{U_X - U_C}{R1} = \frac{U_C - U_W}{R2} \quad \text{met } R1 = R2$$

$$U_X - U_C = U_C - U_W$$

$$U_X = 2U_C - U_W$$



Figuur 5. Blokschema van de  $\Sigma\Delta$ -ADC-testopstelling.

Voor de maximale ingangsspanning moet punt W dus continu laag zijn

$$U_{x_{\max}} = 2U_C - U_{W,\text{laag}}$$

En voor de minimale ingangsspanning moet punt W dus continu hoog zijn:

$$U_{x_{\min}} = 2U_C - U_{W,\text{hoog}}$$

Nemen we aan dat

$$U_{W,\text{laag}} = 0...5\% \text{VCC}$$

$$U_{W,\text{hoog}} = 95...100\% \text{VCC}$$

en dat het triggerniveau

$U_C = 45...55\% \text{VCC}$ , dan volgt daaruit:

$$U_{x_{\max}} = 85...110\% \text{VCC}$$

en

$$U_{x_{\min}} = -10...14\% \text{VCC}$$

De maximale zwaai van de ingangsspanning  $\Delta U_x$  is dus het verschil tussen  $U_{x_{\max}}$  en  $U_{x_{\min}}$ :

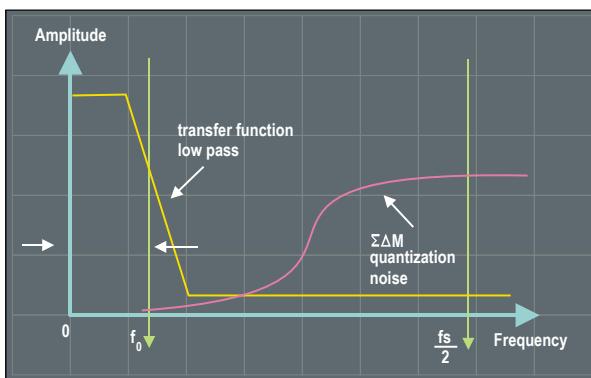
$$\Delta U_x = 90...100\% \text{VCC}$$

De maximale spanningszwaai aan de ingang gaat dus van bijna GND tot bijna VCC.

Om de dynamische ingangsweerstand te bepalen, kunnen we ervan uitgaan dat  $U_C$  nagenoeg constant gehouden wordt en dat daarom dit punt een heel lage impedantie heeft. Voor wisselspanningen levert dat dus een ingangsweerstand  $R_{INAC}$  ter grootte van:

$$R_{INAC} = R_1$$

Voor gelijkspanning geldt dat  $R_1$  aan de andere kant aan  $U_C$  ligt. Is  $U_{IN} > U_C$ , dan vloeit er stroom de ingang in, is  $U_{IN} < U_C$  dan loopt de stroom de andere kant op. De D-ingang wordt continu op triggerniveau gehouden en werkt dus steeds net buiten zijn specificaties. Met hoog- en laag-niveaus aan de ingang wordt geen rekening gehouden, noch met setup- en hold-tijden. Dat is een probleem, want daarop kan de uitgang van de flipflop in een metastabiele toestand raken, waarbij beide uitgangen onder omstandigheden niet meer exact met elkaar in tegenfase zijn. Dat is te compenseren door het uitgangssignaal met een extra D-flipflop af te tasten, zoals in **figuur 5**. De eerste flipflop pakt het signaal op de opgaande flank van de klok, de tweede neemt het over op de daaropvolgende neergaande flank. Dit om het signaal niet te veel vertraging te laten oplopen in de regelkring. Om bijkomende onnauwkeurigheden te voorkomen moeten we eventuele pullup- of pulldown-weerstanden van de PLD aan de ingang van de D-flipflop deactiveren. Een 'keeper'-functie in de vorm van een meekoppelweerstand in het  $k\Omega$ -bereik over de interne niet-inverte-



Figuur 6.  $\Sigma\Delta M$ -quantiseringsruis en frequentiebereik van het digitale filter.

rende ingangsversterker is aanvaardbaar, omdat die in verhouding tot de betrekkelijk grote ingangscapaciteit C binnen de samplefrequentie maar weinig invloed op de signaalspanning kan veroorzaken. De sigma-delta-modulator in figuur 4 heeft bij volledige uitsturing een signaal/ruisafstand S van:

$$S/N = \frac{3}{4\pi} \left( \frac{f_s}{f_o} \right)^{\frac{3}{2}}$$

Hierin is  $f_s$  de samplefrequentie en  $f_o$  de bovengrens van de signaalbandbreedte ( $0...f_o$ ). We zien direct dat een verdubbeling van de samplefrequentie de signaal/ruisverhouding verhoogt met een factor  $23/2 = 9$  dB. Hoe is dat mogelijk? Bij normale Nyquist-ADC's krijgen we door oversampling met een factor 2 immers slechts  $21/2 = 3$  dB verbetering.

De verklaring: bij gebruikelijke ADC's is de ruis gelijkmatig over de bandbreedte verdeeld. Met een  $\Sigma\Delta M$  echter wordt de ruis naar hogere frequenties verschoven, omdat de weerstanden en de condensator van de modulator als laagdoorlaatfilter werken voor het ingangssignaal, en als hoogdoorlaatfilter voor de ruis.

In **figuur 6** zien we dat de ruis vanaf  $f_o$  toeneemt en pas vanaf  $f_s/2$  constant blijft. De waarde voor  $f_o$  volgt uit de combinatie van  $R_1$ ,  $R_2$  en  $C$  (figuur 4), met  $R_1$  en  $R_2$  zo goed als parallel:

$$f_o = \frac{1}{2\pi \left( \frac{R_1 \cdot R_2}{R_1 + R_2} \right) C}$$

en met  $R_1 = R_2 = R$ :

$$f_o = \frac{1}{\pi R C}$$

Uit de formule voor S/N blijkt dat we met een  $f_s = 12$  MHz en een signaalbandbreedte  $f_o$  van 14,4 kHz een S/N-ratio van  $5.743 = 75,2$  dB kunnen behalen. Het heeft daarom zin om het uitgangssignaal van de  $\Sigma\Delta M$  met het daaropvolgende conversiefilter een resolutie van 8 bits of meer te geven.

## Het digitale laagdoorlaat-en conversiefilter

Dit filter heeft de taak om de quantiseringsruis te dempen voordat de samplefrequentie verlaagd wordt en deze ruis naar binnen de signaalbandbreedte omgevouwen wordt. Aangezien bij een eerste orde  $\Sigma\Delta M$  de quantiseringsruis vanaf  $f_o$  een frequentierespons vertoont die met 20 dB per decad toeeneemt, kunnen we voor eenvoudige toepassingen volstaan met een eerste orde digitaal laagdoorlaatfilter. Een extra functie van dit filter is dat het de woordbreedte verhoogt van 1 naar q bits, vandaar dat het ook conversiefilter heet.

Verscheidene typen filters zijn geschikt. Met een resetbare 8-bits up/down-counter zoals in figuur 5 maken we zo'n filter met wel heel weinig moeite. De overdrachtsfunctie komt overeen met een eerste orde sinc-filter (sinus cardinalis) met  $M = 2^{(q-1)}$  trappen. Er worden echter  $2^{(q-1)} - 1$  uitgangsbits uitgemiddeld en niet  $2^{(q-1)}$  uitgangsbits zoals bij het corresponderende sinc-filter van de  $\Sigma\Delta M$ . Door de reset van de teller na  $2^{(q-1)}$  klokcycli krijgen we een tweede tijdbasis ( $f_{so}$ ), waardoor de Z-transformatie niet meer gesloten kan worden toegepast [2]. De teller wordt een klokcyclus lang gereset, wat als voordeel heeft dat hij aan de uitgang maximaal/minimaal binnen de waarde  $\pm(2^{(q-1)} - 1)$  blijft. Hiermee is niet alleen de woordlengte van de teller goed benut maar voorkomen we bovendien tweecomplement-overflow.

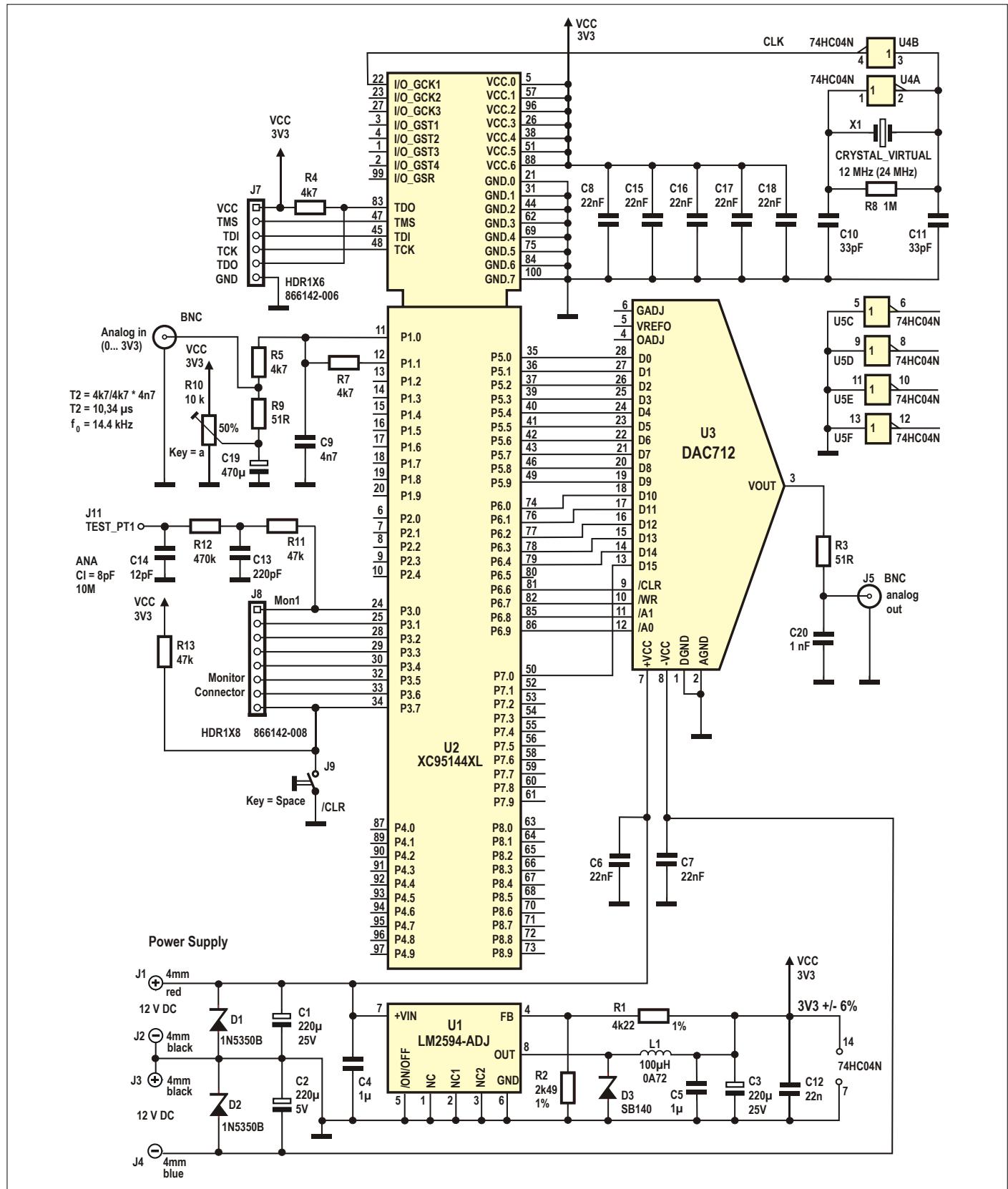
De complete ADC met zo'n filter en een up/down-teller voor  $q = 8$  bits is met slechts 26 macrocellen van de hier gebruikte PLD te realiseren. Voor ieder extra kanaal zijn telkens nog 11 macrocellen nodig, mits we de klokdelers gemeenschappelijk gebruiken. Gebruiken we een eerste orde sinc-filter met differentiatoren en integratoren, dan moeten we normaliter de ADC uitvoeren met filters alleen voor  $M = 112$ , met

alle beschikbare 144 macrocellen [3]. Hierbij is M het aantal uitgangsbits van de  $\Sigma\Delta$ M waarover doorlopend gemiddeld wordt. Als we de volgorde van integrator en differentiator omdraaien, dan kunnen we deze oplossing ook realiseren met

slechts 50 macrocellen.

Als we het digitale filter uitvoeren als niet-verliesvrije accumulator met  $k = 0,99609375$  ( $m = 8$ ) en een uitgangswoordbreedte van 8 bit, dan hebben we voor de hele ADC in totaal 61

macrocellen nodig. De factor k is de vermenigvuldigingscoëfficiënt, de factor waarmee we het resultaat van de accumulatie tot nu toe moeten vermenigvuldigen voordat er een nieuwe uitgangswaarde bij op wordt geteld. In



Figuur 7. Het complete schema van de  $\Sigma\Delta$ -ADC-testopstelling.

plaats daarvan kunnen we de factor  $k$  uitvoeren met een compleet vermenigvuldigingsnetwerk met schuiven en optellen. Dan komt  $m$  overeen met het aantal verschuivingen van de bits. Dus als we door  $2^m$  moeten delen, dan voegen we het binaire datawoord toe,  $m$  posities naar rechts verschoven.

### Praktische realisering

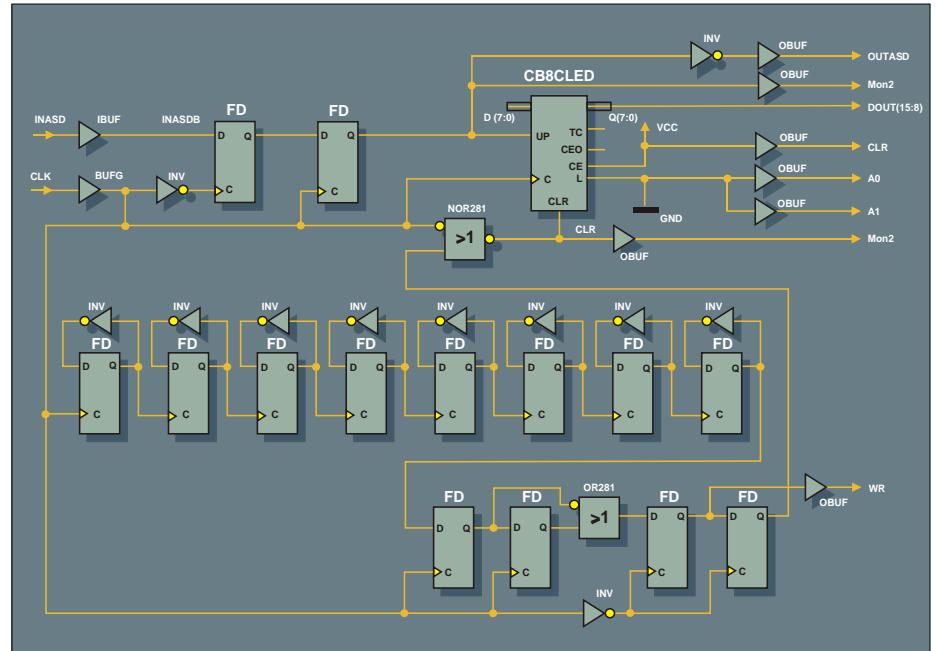
Figuur 5 geeft het blokschema van de  $\Sigma\Delta$ -ADC in een testopstelling. De  $\Sigma\Delta$ M bestaat hier uit twee D-flipflops die met verschillende flanken van de klok van  $f_s = 12$  MHz worden getriggerd, om metastabiele toestanden te onderdrukken. De combinatie  $R_1, R_2$  en  $C_1$  aan de ingang van de  $\Sigma\Delta$ M legt  $f_o$  vast op 14,4 kHz. Ter controle is de uitgang van de  $\Sigma\Delta$ M via Mon1 beschikbaar. Aan de ANA-uitgang verschijnt het analoge ingangssignaal van de  $\Sigma\Delta$ M, verkregen uit de uitgangspulsstroom via een derde orde laagdoorlaatfilter met  $f_c = 15,4$  kHz. De analoge ingangsspanning mag maximaal 3,3 V<sub>tt</sub> bedragen vanwege de 3,3V-voedingsspanning van de PLD.

De  $\Sigma\Delta$ M-uitgang gaat naar de up/down-ingang van de 8-bit-teller. Hoog betekent '+1', Laag betekent '-1'. De  $\Sigma\Delta$ -ADC uitgang Q(7:0) geeft data in tweecomplement-notatie.

De teller fungeert als conversiefilter. Hij wordt om de 128 klokcycli eerst uitgelezen en dan één klokcyclus lang gereset. Daardoor kan de telleruitgang maximaal/minimaal de decimale waarde  $\pm 127$  hebben, waarmee het bereik van 8 bits goed wordt benut en bovendien tweecomplement-overflow wordt voorkomen. De maximale resp. minimale uitgangswaarde krijgen we als het analoge ingangssignaal  $U_{Xmax}$  resp.  $U_{Xmin}$  is.

Als we de deler steeds 1 bit kleiner kiezen dan de up/down-counter, dan krijgen we een symmetrisch uitgangssignaal dat niet kan overlopen. Treedt tweecomplement-overflow of -underflow op, dan springt het signaal van maximaal naar minimaal of omgekeerd, wat heel onaangename harmonischen geeft. Verderop maken we dat met een screenshot duidelijk.

De  $\Sigma\Delta$ -ADC-uitgang Q(7:0) gaat naar de high-byte-ingang van een 16-bit tweecomplement-DAC. De low-byte-ingang gebruiken we niet, maar de PLD heeft hem wel, voor het geval een grotere woordbreedte nodig mocht zijn. De symmetrische voedingsspanning van  $\pm 12$  V voor de DAC geeft een spanningsbereik



Figuur 8. Het inwendige van de PLD van de  $\Sigma\Delta$ -ADC met slechts 26 macrocellen.

aan de uitgang van bijna  $\pm 10$  V. De maximale DAC-omzetfrequentie van 100 kHz is met  $f_{so} = 12$  MHz / 128 = 93,75 kHz bijna volledig benut.

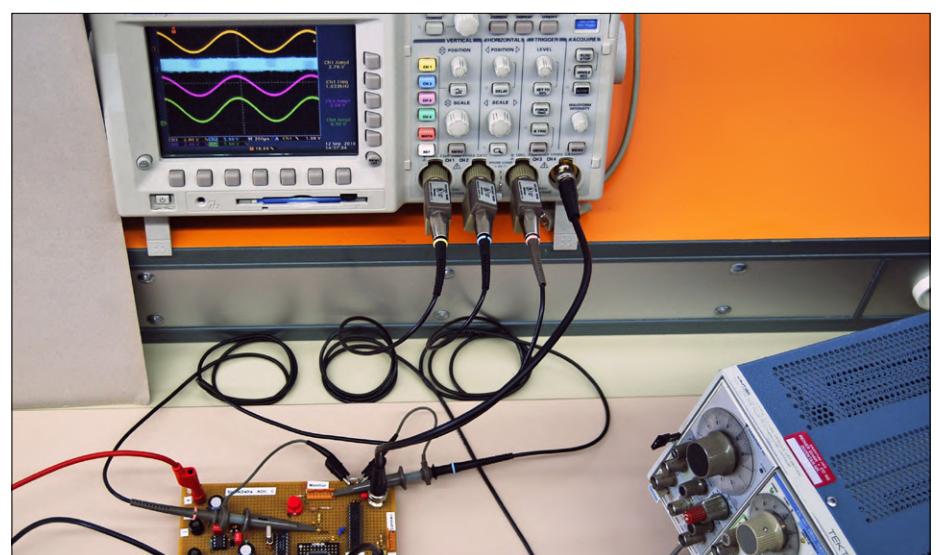
Als we de teller als pure 8-bit up-counter uitvoeren en voor zijn enable-ingang de  $\Sigma\Delta$ M-uitgang gebruiken, dan krijgen we in plaats van tweecomplement-weergave een binair offset. De bijbehorende deler moet dan op  $2^8$  worden ingesteld.

In **figuur 7** ziet u de schakeling van de testopstelling. U ziet ook functies voor bijvoorbeeld de voedingsspanning, de klok en de 16-bits-D/A-conversie, die niet per se bij de  $\Sigma\Delta$ -ADC horen. **Figuur 8** laat

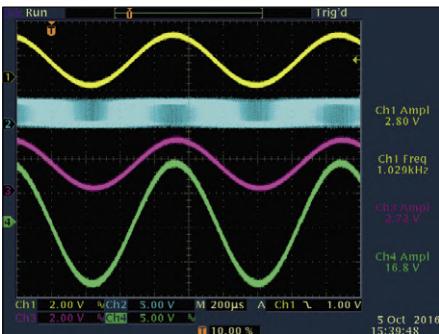
zien hoe de  $\Sigma\Delta$ -ADC is uitgevoerd in de PLD, een XC95144. We gebruiken slechts 26 van de 144 beschikbare macrocellen. Samen met de drie externe componenten  $R_1, R_2$  en  $C_1$  vormt dit de complete  $\Sigma\Delta$ -ADC met 8-bits resolutie. De uitgangsdata is echter alleen maar geldig tijdens de opgaande flank van /WR. Wie dat te kort vindt zal een 8-bits D-register moeten toevoegen, getriggerd op deze flank.

### Metingen in het tijddomein

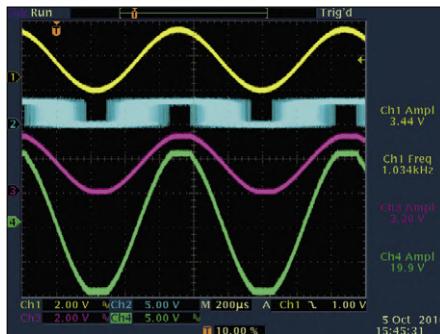
Om te laten zien wat er nu precies gebeurt in deze PLD-ADC en wat voor gedrag hij vertoont, hebben we een aan-



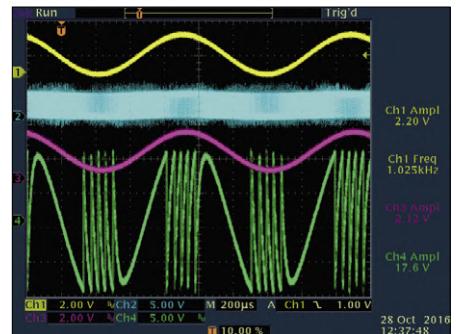
Figuur 9. Testopstelling voor metingen in het tijddomein.



Figuur 10. Signalen van de test in het tijddomein bij bijna maximale uitsturing. Ch1: analog in, Ch2: Mon1, Ch3: ANA en Ch4: analog out.



Figuur 11. Tijddomein-signalen bij oversturing: Ch1: analog in, Ch2: Mon1, Ch3: ANA en Ch4: analog out.



Figuur 12. Tijddomein-signalen van de testopstelling als de woordbreedte van het digitale filter te klein is. Ch1: analog in, Ch2: Mon1, Ch3: ANA en Ch4: analog out.

tal metingen gedaan met onze testopstelling, waarvan u een foto ziet in **figuur 9**, hier voor metingen in het tijddomein.

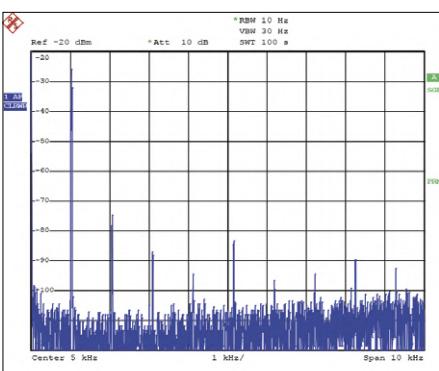
**Figuur 10** toont de signalen aan de ingang en aan de uitgang van de  $\Sigma\Delta$ -ADC bij een signalfrequentie van 1 kHz. Het sinusvormige analoge ingangssignaal (Ch1, geel) zit met  $2,8 \text{ V}_{\text{tt}}$  bijna op het maximale niveau. Aan de digitale uitgang (Ch2, blauw) ziet u de variatie in puls-

dichtheid, overeenkomstig de analoge spanning. Bij een lage ingangsspanning is de uitgang vaker laag en bij een hogere ingangsspanning vaker hoog. Aan het uitgangssignaal ANA (Ch3, roze) zien we dat het ingangssignaal tamelijk precies aan de digitale uitgang van de  $\Sigma\Delta$ -ADC wordt weergegeven. De lichte demping van de amplitude is te herleiden tot het passieve RC-laagdoorlaatfilter.

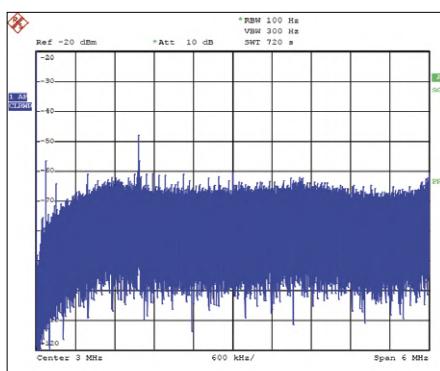
ter. De analoge DAC-uitgangsspanning, (Ch3, groen) heeft een amplitude van ca.  $\pm 8,4 \text{ V}$ , wat ook hier overeenkomt met nagenoeg het volledige bereik. De symmetrie bewijst dat het digitale DAC-ingangssignaal Q(7:0) in tweecomplement aanligt.

In **figuur 11** zien we hoe de  $\Sigma\Delta$ -ADC zich gedraagt bij oversturing. De  $\Sigma\Delta$ M-uitgang Mon1 geeft beurtelings alleen nog hoog of laag – daaraan zien we dat het oversturing betreft. Bijgevolg blijft dan ook de analoge DAC-uitgang bij deze pieken op zijn maximale of minimale waarde staan, maar er is geen tweecomplement-overflow. Dit exact begrensde ADC-uitgangssignaal bewijst dat de woordlengte van het digitale filter toereikend is.

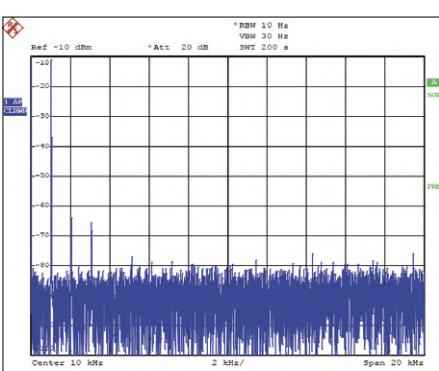
In **figuur 12** zien we wat er gebeurt als de woordbreedte van het digitale filter te klein is. We zien tweecomplement-overflow, en het uitgangssignaal van de DAC springt heen en weer tussen maximale en minimale waarde, met als gevolg veel sterke harmonischen in het uitgangssignaal.



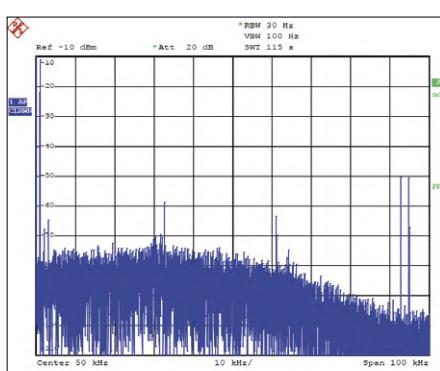
Figuur 13. Spectrum van figuur 10 met 10 kHz bandbreedte.



Figuur 14. Spectrum van figuur 10 met 6 MHz bandbreedte.



Figuur 15. Spectrum van figuur 10 met een bandbreedte van 20 kHz, maar op 'analog out'.



Figuur 16. Spectrum als bij figuur 15, maar nu met een bandbreedte van 100 kHz.

## Metingen in het frequentiedomein

Voor de metingen in het frequentiedomein passen we de meetopstelling van figuur 9 een beetje aan. Met een LC-laagdoorlaatfilter aan de uitgang van de generator dempen we de harmonischen in het testsignaal. De meetkop aan de ingang van de analyzer geeft een demping van 40 dB en zowel meetkop als ingang hebben een ingangsimpedantie van  $50 \Omega$ .

In **figuur 13** zien we het spectrum van de digitale uitgang Mon1 van 0 tot 10 kHz bij een sinus van 1 kHz en  $2,8 \text{ V}_{\text{tt}}$  aan de

analoge ingang. De ruis op  $-100$  dBm ligt ca.  $75$  dB onder het signaalniveau van  $-25$  dBm, wat goed overeenkomt met wat we theoretisch op dit gebied mogen verwachten. Het niveauverschil met de harmonischen van  $\geq 50$  dB geeft aan dat de  $\Sigma\Delta$ M een goede lineariteit heeft.

Tip: dit soort metingen met een FFT-analyzer kunnen meetfouten geven, omdat deze apparaten vaak een geringe samplefrequentie hebben in verhouding tot het meetbereik, waardoor ruis bij hogere frequenties in het meetbereik ‘gevouwen’ wordt.

**Figuur 14** is het spectrum aan de digitale uitgang Mon1, maar nu van  $0$  tot  $6$  MHz. Hier zien we duidelijk dat de ruis toeneemt met de frequentie. Vanaf ongeveer  $1,2$  MHz blijft de ruis ongeveer  $-60$  dBm. De afstand tot het nuttige signaal is hier nog maar  $35$  dB.

**Figuur 15** toont het spectrum van  $0$  tot  $20$  kHz aan de uitgang van de DAC ‘analog out’, ook weer met een sinus van  $1$  kHz en  $2,8$  V<sub>pp</sub> aan de analoge ingang. Het ruisniveau bij  $-75$  dBm ligt ongeveer  $65$  dB onder het nuttige signaalniveau van  $-10$  dBm.

De resolutie van de  $\Sigma\Delta$ -ADC is  $8$  bit, wat eigenlijk een S/N van  $49,92$  dB zou geven. Oversampling geeft echter een rekenkundige winst van ongeveer  $20$  dB, wat samen  $70$  dB is, en komt weer goed overeen met wat we meten. Het niveauverschil ten opzichte van de harmonischen van  $\geq 50$  dB geeft ook hier weer aan dat de lineariteit goed is.

**Figuur 16** geeft eveneens het spectrum van ‘analog out’, maar nu van  $0$  tot

## Over de auteur

Guido Nopper is afgestudeerd als ingenieur (Dipl.-Ing) aan de FH Furtwangen en werkte van 1981 tot 1984 als LSI-MOS-ontwikkelaar met het zwaartepunt op A/D-converters en digitale filters. Sinds 1984 is hij verantwoordelijk voor het complete hardware-ontwerp van papierverwerkende bureumachines, in het bijzonder chequeprinters en scanners.

## Weblinks

- [1] The Evolution of Oversampling Analog-to-Digital Converters (PDF): <https://bit.ly/2Z1q68G>
- [2] Do Multirate Systems Have Transfer Functions?: [www.dsprelated.com/showarticle/143.php](http://www.dsprelated.com/showarticle/143.php)
- [3] Datasheet CPLD XC95144XL: <https://bit.ly/2XvhtTy>

100 kHz. We zien nu duidelijk het digitale laagdoorlaatfilter in werking, want in tegenstelling tot figuur 13 neemt de ruis niet toe met toenemende frequentie, maar neemt die juist af. Rechts in beeld zien we de symmetrische ‘omklap’-producten van de ingangs frequentie naar de DAC-samplefrequentie  $f_{so}$  van  $93,75$  kHz. Conclusie

Wie geïnteresseerd is in PLD’s en het experiment niet schuwt, kan in dit artikel zien met hoe weinig externe onderdelen er relatief complexe schakelingen met goede technische prestaties te realiseren zijn. Niets verplicht u natuurlijk om het experiment in dit artikel na te doen. U kunt zich ook laten inspireren door de hier beschreven principes en bijvoorbeeld een ADC met betere resolutie (en geringere bandbreedte) realiseren. De beno-

digde wiskunde hebben we in dit artikel beperkt tot wat minimaal noodzakelijk is om deze materie te begrijpen. Het zou voldoende moeten zijn om de basisschakeling naar eigen ideeën te kunnen aanpassen en uitbreiden. ■

(170566-04)

## IN DE STORE



→ CPLD Breakout Board in

DIL-formaat

[www.elektor.nl/160425-91](http://www.elektor.nl/160425-91)

→ Boek: “Design Recipes for FPGAs”, Second Edition

[www.elektor.nl/18054](http://www.elektor.nl/18054)

Advertentie



**focus on cost drives out quality  
focus on quality drives out cost**

**precisiebeurs.nl | 13 & 14 nov | stand 216**

Vlakbodem 10  
+31 187 602 744

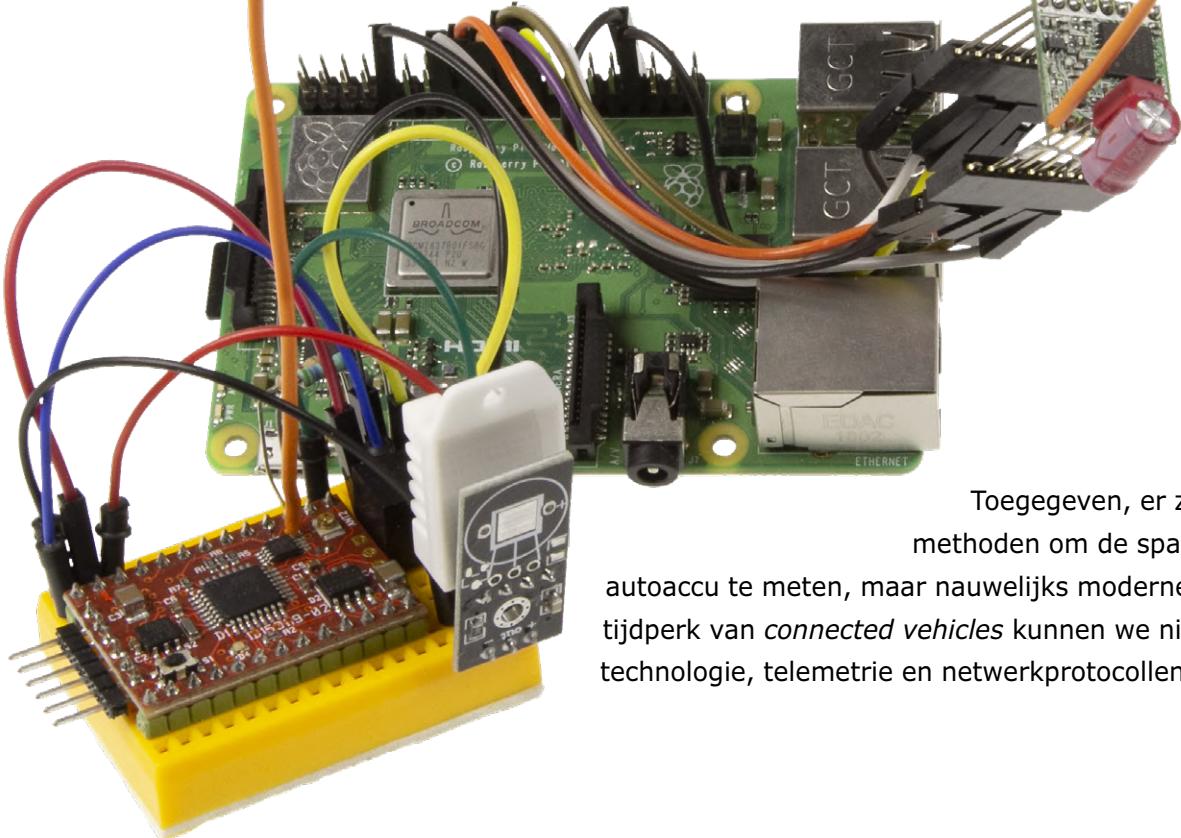
3247 CP Dirksland  
[www.tbp.nl](http://www.tbp.nl)

the Netherlands  
[info@tbp.nl](mailto:info@tbp.nl)

# Draadloze accu-voltmeter

## laadtoestand altijd in de peiling

Sven Bockstadt en Mathias Claußen (Elektor Labs)



Toegewezen, er zijn eenvoudiger methoden om de spanning van een autoaccu te meten, maar nauwelijks modernere. In het tijdperk van *connected vehicles* kunnen we niet meer om IoT-technologie, telemetrie en netwerkprotocollen heen.

Hoewel moderne voertuigen over steeds geraffineerdere laadelektronica beschikken, betekent dat nog steeds niet dat de autoaccu ook altijd 100% geladen is. Ten gevolge van moderne start/stop-mechanismen en alle huidige emissievoorschriften wordt de accu doorgaans alleen geladen gedurende de *bulk* en *absorption phase*. Bovendien kan in het koude seizoen spontane uitval of diepontlading van de accu optreden.

Om een constant gezonde ladingstoestand (*state of charge*) te bereiken, heeft auteur Sven Bockstadt een energiebesparende oplossing ontwikkeld, die op gezette tijden een korte spanningsmeting uitvoert. De gemeten spanning wordt vanaf de meetlocatie via een draadloze lange-afstandsverbinding conform de LoRa-standaard naar een verwerkings-

eenheid gestuurd, die de meetwaarden op een display toont en tevens opslaat voor latere verwerking.

Voor platforms zoals Arduino en Raspberry Pi is een scala aan compatibele shields voor draadloze overdracht beschikbaar, die data-overdracht betrekkelijk eenvoudig maken. Om het project niet nodeloos te complicerend, heeft de auteur voor zijn prototype zoveel mogelijk gebruik gemaakt van kant-en-klare boards voor het Arduino-platform. Daardoor hoeft geen print te worden ontwikkeld en heeft de externe bedrading ook niet veel om het lijf.

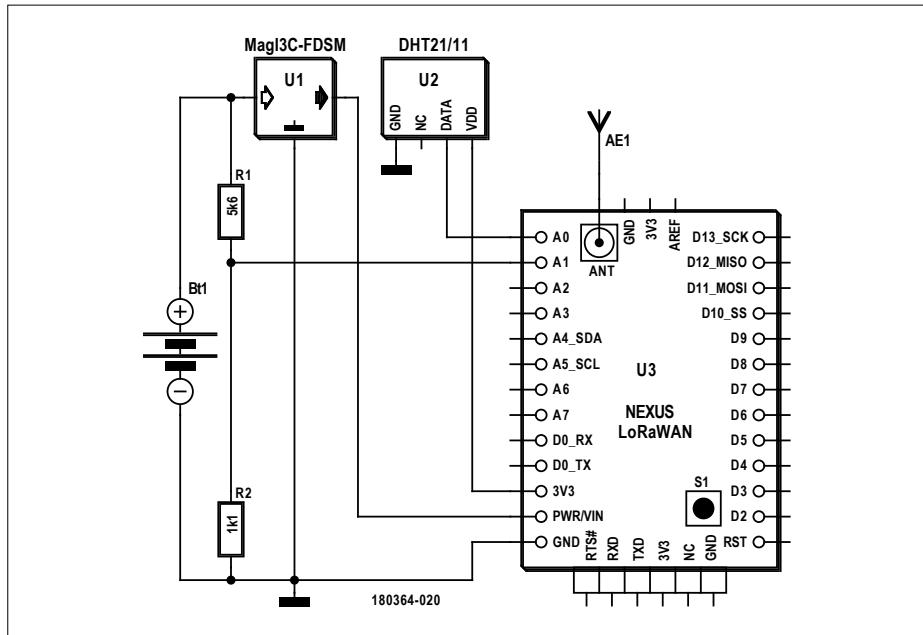
Voor de hardware van zender en ontvanger is monitorsoftware ontwikkeld om beide apparaten serieel te kunnen aansturen en debuggen. De log-functie genereert een zich steeds uitbreidend

CSV-bestand, dat op een micro-SD-card wordt opgeslagen en op een PC in Excel of vergelijkbaar spreadsheetprogramma eenvoudig kan worden bekeken en bewerkt.

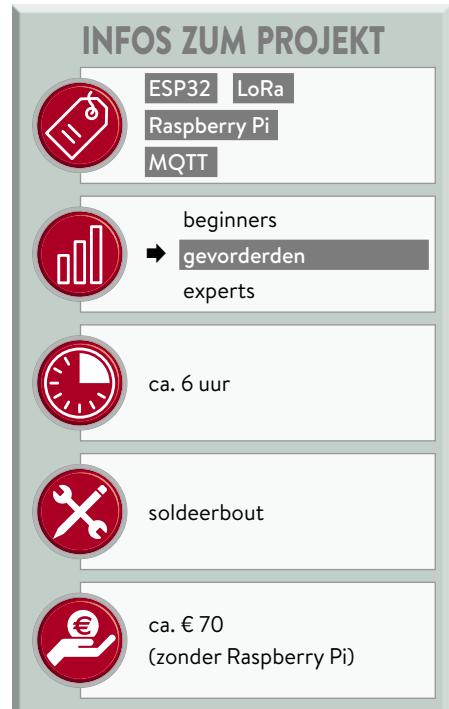
Deze draadloze spanningsmonitor voor motorvoertuigen is door de auteur enige tijd geleden op het Elektor Labs projectplatform [1] gepubliceerd. Hieraan voorafgaand doorliepen hard- en software de Elektor Labs-molen, waarbij enige veranderingen, verbeteringen, vereenvoudigingen en uitbreidingen werden doorgevoerd. We presenteren hier het resultaat.

### De zender

Als zender werd in het oorspronkelijke ontwerp een Feather-M0-board van Adafruit toegepast, dat een 433MHz-LoRa-radiomodule RFM96 van



Figuur 1. De zender bestaat uit een spanningsdeler en de LoRa-module. De optionele weersensor DHT21/11 gebruikt één van de vele vrije analoge ingangen.



HopeRF [2] aan boord heeft. Dit board vereist echter een externe analoog/digitaal-converter met log-functie en ook nog een externe timer-module, die het data-verkeer (energiezuinig) klokt. Omdat we dat nogal omslachtig vonden, hebben we het Feather-M0-board vervangen door een Arduino-compatibel LoRa-Nexus-board ([3], zie kader), dat niet alleen een op 868 MHz werkende RFM95-LoRa-module [2] bevat, maar ook nog eens wordt bestuurd door een ATmega328P, die een eigen 10-bits ADC heeft. **Figuur 1** laat zien dat er alleen nog een simpele weerstandsdivider met  $R_1 = 5\text{k}\Omega$  en  $R_2 = 1\text{k}\Omega$  voor de ADC-ingang hoeft te worden gezet om de accuspanning aan te passen aan het bereik van ADC-ingang A1. Het LoRa-Nexus-board bevat ook nog een flash-geheugen (W25X40CLSNIG) en een realtime-klok, zodat verder geen externe componenten nodig zijn. De timing van de dataoverdracht wordt door de controller-firmware verzorgd.

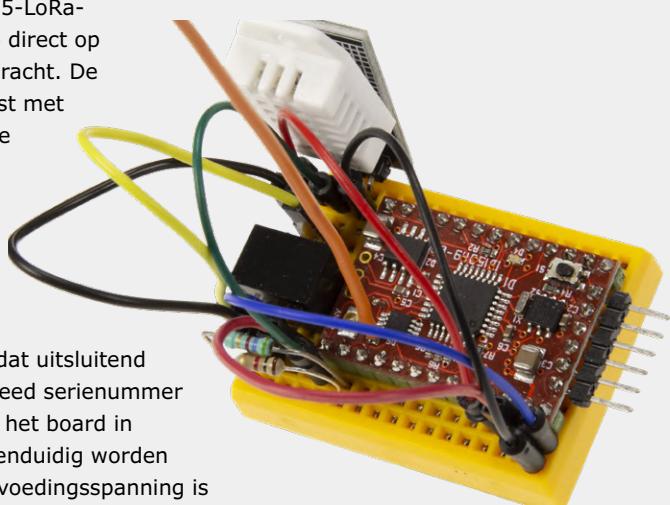
Voor de voeding van de module wordt een DC/DC-converter met vaste uitgangsspanning van 3...28 V een stabiele uitgangsspanning van +5 V bij een stroom van 1 A levert. Mocht u bij deze waarden aan de goede oude 7805 moeten denken, dan is dan nog niet zo gek: deze kleine schakelende converters worden door Würth als rechtstreekse vervanger voor de L7805 aangeprezen. Zelfs de gebruikelijke in- en uitgangscon-

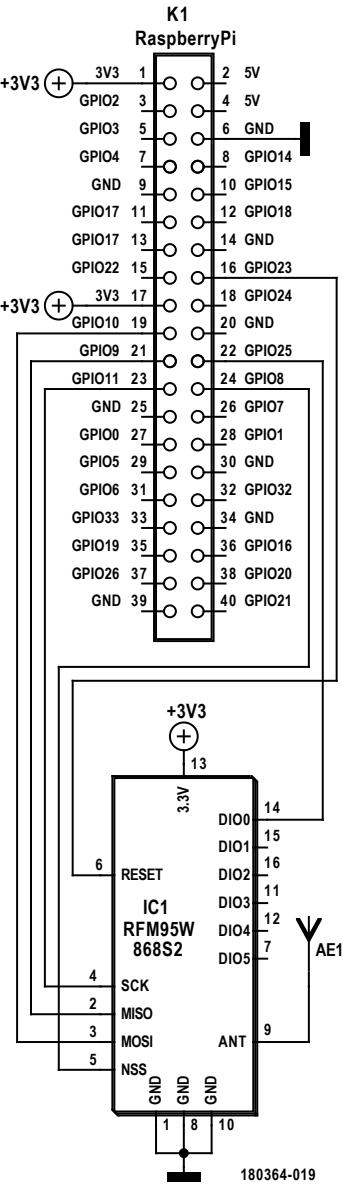
densatoren kunnen bij deze converters vervallen. De +5V-uitgangsspanning van de schakelende regelaar worden op het board door een tweede (lineaire) spanningsregulator teruggebracht tot 3,3 V. Hiermee worden niet alleen alle IC's op de module gevoed, maar hij dient ook als referentiespanning voor de ADC en zou zelfs als I2C-busspanning (hier niet van toepassing) kunnen worden gebruikt.

De 3,3V-spanning van de onboard-regulator voedt ook een 'weersensor' DHT21/11, die de temperatuur en de relatieve luchtvochtigheid meet. De software evalueert deze data en kan daardoor beter inschatten welke parameters van de accu te verwachten zijn. Deze sensor is aangesloten op ADC-ingang A0. Deze paar componenten hebben we op een breadboard opgebouwd en ze kun-

## LoRa Nexus Board

Het 23 x 33 mm kleine LoRa Nexus board is gebaseerd op de 8 bit MCU ATmega328P van Atmel, dezelfde controller als op de Arduino Nano. Daar komt nog een RFM95-LoRa-radiomodule bij, die direct op het board is aangebracht. De controller is uitgerust met een externe realtime klok met 64 bytes SRAM (MCP7940M) en een 4 Mbit-flashgeheugen (W25X40CL). Een bijzonderheid is het IC DS2401P, dat uitsluitend een uniek, 48 bit breed serienummer bevat. Hiermee kan het board in een radionetwerk eenduidig worden geïdentificeerd. De voedingsspanning is  $3,3 \text{ V}_{\text{DC}}$ , die uit de aangelegde ingangsspanning van  $V_{\text{raw}} = 5 \text{ V}_{\text{DC}}$  wordt afgeleid door een LDO-spanningsregelaar AZ1117CR-3.3TRG1.





Figuur 2. De schakeling bestaat slechts uit een RFM95-ontvanger, die met enkele kabeltjes is verbonden met de Raspberry Pi-GPIO.

**Tabel 1. Verbindingen tussen RFM95 en RPi.**

RFM95	Raspberry Pi
SCK	GPIO11
MISO	GPIO09
MOSI	GPIO10
NSS	GPIO08
RESET	GPIO22
DIO0	GPIO25

nen worden geprogrammeerd met een geschikte FTDI-kabel ( $V_{CC} = 3,3$  V) aan de FTDI-ingang van het Nexus-board.

### De ontvanger

Oorspronkelijk was de ontvanger eveneens met een Feather-M0-board opgebouwd, bijgestaan door een externe ADC-datalogger, een timer-module en elektronica voor de voeding.

Na bewerking in het Elektor-Lab is alleen de RFM95-module als LoRa-modem [5] overgebleven, die op enkele GPIO-pinnen van de Raspberry Pi-header wordt aangesloten (zie **figuur 2** en **tabel 1**). Voor de rekenkracht wordt in plaats van een ‘Arduinoïde’ controller een Raspberry Pi model 3+ ingezet, omdat deze veel meer mogelijkheden biedt voor dataverwerking dan een Arduino. Een andere reden om voor de Raspi te kiezen is gelegen in het feit dat hier heel eenvoudig een monitor op is aan te sluiten. En de Raspberry Pi levert ook nog eens een schone 3,3V-voedingsspanning voor de radiomodule.

Beide modules moeten worden uitgerust met een antenne. Twee stukjes draad met een lengte van elk 86 mm zijn daarvoor al voldoende. Tot zover de hardware van dit project, de rest is aan de software.

### Radioverbinding LoRa-PHY

De radioverbinding voldoet aan de LoRa-standaard, waarvan alleen de fysieke laag (LoRa RF/PHY) wordt gebruikt. Daarbij gaat het vooral om de CSS-modulatietechniek op de overdrachtsfrequentie in de ISM- of SRD-band. LoRa schrijft op zich geen complex hoger protocol voor. Alleen een header met zenderadres is vereist. Een ontvanger-adres kan worden meegegeven, maar dat hoeft niet: de ontvanger is (zoals in onze toepassing) zo in te stellen, dat hij alle datapakketten op een draaggolf van 869,5 MHz ontvangt, onafhankelijk van een adresheader.

We hoeven alleen op te geven hoeveel bytes we willen verzenden, waarna de communicatie zonder verder gedoe door de Semtech-chip wordt geregeld. De LoRa-module zet hier in het belang van de stoer vastheid nog een – zelf in te stellen – preamble voor. De ontvanger wordt natuurlijk op dezelfde preamble ingesteld. Bereikt een pakket met deze preamble de ontvanger, dan geeft deze de bytes van de *payload* door.

De parameters aan de Raspberry Pi-kant worden geconfigureerd met een kleine applicatie op basis van de Radiohead-library [6]. Voor de zender worden natuurlijk dezelfde standaardinstellingen gekozen.

De software op de Raspberry Pi leest de ontvangen LoRa-datapakketten en zoekt naar een bericht met als afzender 0x20, dat met een payload-lengte van 12 bytes aan 0x10 als ontvanger wordt overgedragen. Als zo’n bericht wordt gevonden, wordt een payload-pakket verwacht met een structuur zoals in **figuur 3**. Na een payload-ID volgen successievelijk de spanningswaarde (16 bit), de temperatuur (8 bit) en de luchtvochtigheid (8 bit).

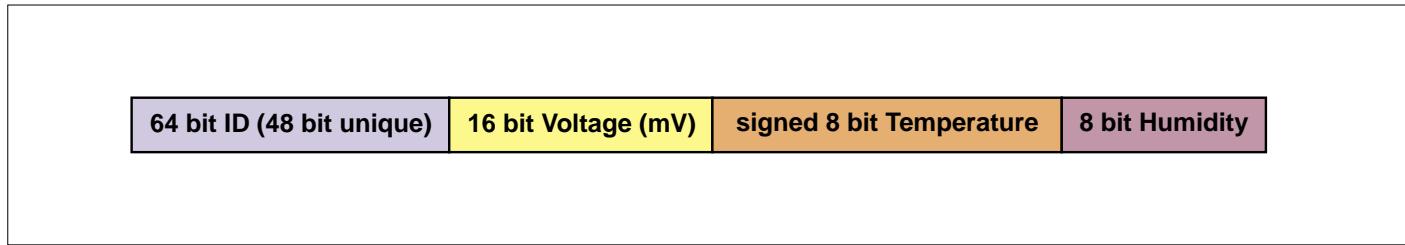
Het 64 bit brede ID wordt samengesteld uit het unieke 48-bit adres uit de authenticatie-chip DS2401P op het LoRa-Nexus-board (dit adres is te vinden op een sticker op de print, maar kan met een Arduino ook worden uitgelezen) en twee bytes die voor de accu-voltmeter-toepassing altijd hetzelfde zijn.

### MQTT en een Mosquitto-broker

De rest van de software verwerkt de binnenkomende data en zendt die met behulp van de *libmosquitto*-library naar een MQTT-broker. De library bepaalt aan de hand van het 64-bit ID of het bericht afkomstig is van een eigen zender. MQTT (zie **kader**) biedt een simpel, efficiënt protocol voor het uitwisselen van gegevens tussen verschillende (IoT-)apparaten. Meer over MQTT is te lezen in [8].

## ONDERDELENLIJST

- R2 = 1 k, 5 %, 0,25 W, 250 V
- R1 = 5k6, 5 %, 0,25 W, 250 V
- MOD1 = MagI<sup>3</sup>C-FDSM Fixed Step Down Regulator
- MOD2 = DHT11 uit de Elektor-ESP32-Smart Kit
- MOD3 = LoRa Nexus Board
- MOD4 = RFM95 Ultra LoRa Transceiver-module (868 MHz)
- Raspberry Pi 3 B+ Starter Kit + gepatchte RPi-compilatie
- J1,J2 = jumperdraad uit de Elektor ESP32-Smart-Kit
- (optioneel) groot breadboard
- (optioneel) FT232R-USB/Serial Bridge/BOB (Elektor 110553-91)



Figuur 3. Het eenvoudige payload-protocol bestaat uit een 64-bit payload-ID gevuld door drie meetwaarden.

Voor MQTT is altijd een soort server nodig, die bij MQTT *broker* wordt genoemd. Een heel bekende MQTT-open-source-broker voor diverse platforms is Mosquitto, die intussen ook in Raspbian is opgenomen. De broker geeft berichten van databronnen door aan dataconsumenten die zich op deze berichten hebben ‘geabonneerd’.

In dit geval publiceert de Lora-ontvanger de ontvangen Lora-pakketten op verschillende MQTT-topics: ‘car\battery\voltaage’, ‘car\interior\temperature’ en ‘car\interior\humidity’. Zo kunnen alle waarden individueel worden verwerkt. De berichten voor de MQTT-broker zijn verpakt in JSON, zodat de drie elementen heel eenvoudig en flexibel verder kunnen worden verwerkt.

Met de voor IoT-toepassingen gespecialiseerde grafische programmeertaal Node-RED is software te maken waar mee we ons vervolgens op MQTT-topics kunnen abonneren. Met de ontvangen berichten kunnen de meest uiteenlopende zaken worden gedaan, zoals het besturen van uw huis, e-mails versturen, of data presenteren in een gelikte grafiek (**figuur 4**). Dit soort toepassingen zijn met Node-RED heel eenvoudig te realiseren [9].

### Het installeren

Het installeren van Node-RED en MQTT op een Raspberry Pi is al beschreven in [19]. Als het installeren van Node-RED en Mosquitto is voltooid, moeten we nog enige voorbereidingen treffen om de code te kunnen compileren, zodat de software **op de RPi** functioneert.

Er hoeven maar een paar commando’s in de terminal te worden ingevoerd om een aantal afhankelijkheden van de software te installeren.

```
apt-get install libmosquitto-dev
apt-get install libmosquitto1
apt-get install libmosquitto1
```

Daarnaast moet de library *bcm2835* [10] op de Pi worden geïnstalleerd. Pak het bestand uit en ga naar de map. Voer daarin `./configure` uit, om de code op het compileren voor te bereiden. Nadat dit is voltooid en geen fouten zijn opgetreden, kunnen we het volgende invoeren:

```
make
sudo make check
sudo make install
```

Hiermee zou de library geïnstalleerd moeten zijn; nu zijn we klaar om de code voor de ontvanger te compileren. Haal de

code op van GitHub [11] en kopieer hem naar de Pi. Ga naar de map en voer `make` uit. Daarna is de ontvanger (hopelijk!) klaar voor gebruik. Helaas moet het programma met root-rechten worden uitgevoerd, om toegang te behouden tot enige Raspberry Pi-subsystemen. We starten de software dan ook met `./sudo main`. Om de data op te slaan, kunnen we *sqlite3* als databank-engine gebruiken. Installeer die met:

```
apt-get install sqlite3
```

Deze sqlite-databank maakt het mogelijk alle informatie op te slaan die door de

## MQTT en Mosquitto

Conform het MQTT-protocol (*Message Queuing Telemetry Transport*) worden data en commando’s tussen (IoT-)apparaten (devices) niet direct, maar via een centrale MQTT-server (broker) uitgewisseld. MQTT is efficiënt, betrouwbaar en vormt slechts een geringe belasting voor de processor. Het protocol is berichtgeoriënteerd: een databron zendt data alleen naar een broker (publish). Een ontvanger-client hoeft niet voortdurend bij de server op te vragen of er nieuwe data van de zender zijn, maar ‘abonneert’ (subscribe) zich op de berichten van de zender, zodat de broker nieuwe data van de geabonneerde zender automatisch naar de client doorstuurt. Zender en ontvanger worden zo door de broker volledig ontkoppeld, zodat iemand die de data beschikbaar stelt, zich niet hoeft te bekommeren om wie de data ontvangt.

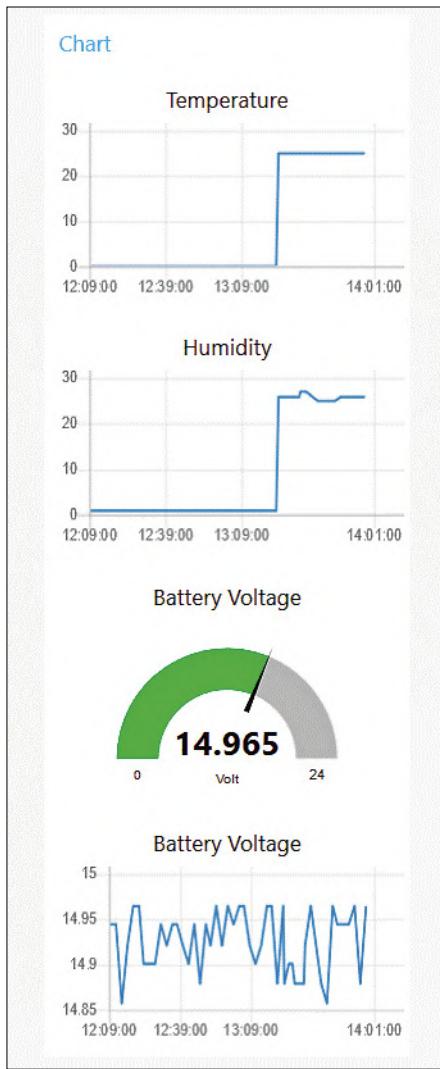
Een bericht bevat onder andere de elementen



- **Topic** is het thema van het bericht. Topics zijn eenvoudige strings, gescheiden door slashes. Een topic zoals **car/battery/voltage** omvat een hiërarchie van objecten, hier bevindt zich de zender in de auto, het gaat ons om de accu en daarvan zijn we geïnteresseerd in de spanning.
- **Payload** is de inhoud van het bericht, doorgaans commando’s of gegevens.

Om MQTT (versies 3.1, 3.1.1 en 5.0) te gebruiken, hebben we een broker nodig, zoals de geliefde Eclipse Mosquitto [17]. Deze slanke open-source broker kan uitstekend worden geïnstalleerd op alle low-power single-board-computers en dus ook op een Raspberry Pi (absoluut probleemloos via de hoofdrepository).

Bron: FHEM [7].



Figuur 4. Zo gelikt kan een GUI de meetwaarden presenteren.

accu-voltmeter wordt geleverd. Daarna gaan we naar de home-directory van de Pi-user en maken de databank aan. Open daartoe de terminal en tik in:

```
cd ~
sqlite3 carsensor.db
```

Dit genereert een nieuwe databank, die nog moet worden gevuld met drie tabellen:

```
sqlite> CREATE TABLE battery_
    voltage ( id INTEGER PRIMARY
    KEY AUTOINCREMENT, uuid
    NUMERIC, voltage NUMERIC,
    timestamp DATETIME DEFAULT
    CURRENT_TIMESTAMP);
sqlite> CREATE TABLE humidity
    ( id INTEGER PRIMARY
```

```
KEY AUTOINCREMENT, uuid
    NUMERIC, humidity NUMERIC,
    timestamp DATETIME DEFAULT
    CURRENT_TIMESTAMP);
sqlite> CREATE TABLE temperature
    ( id INTEGER PRIMARY KEY
    AUTOINCREMENT, uuid NUMERIC,
    temperature NUMERIC,
    timestamp DATETIME DEFAULT
    CURRENT_TIMESTAMP);
sqlite> COMMIT;
sqlite> .exit
```

Hiermee worden de voor de database benodigde tabellen gegenereerd. Nu moeten we alleen nog Node-RED installeren:

```
cd ~/.node-red
npm i --unsafe-perm
node-red-node-sqlite
```

Als dit is voltooid, wordt de Pi opnieuw gestart. Vrijwel alle componenten draaien nu. Nu moet de Flowcode [11] in Node-RED worden geïmporteerd, net zoals bij de Gigantische LED-klok uit Elektor mei/juni 2019.

Het enige wat ons aan de ontvanger-

kant nu nog te doen staat, is de gecompileerde software te starten. Ga naar *~/LoRa\_MQTT\_Voltmeter* en start het gecompileerde programma met *sudo ./main*. Dit is alleen voorlopig nodig, later kunnen we de software naar het RPi-autostartbereik verschuiven.

Nu is het tijd om onze aandacht op de **zenderkant** met het Arduino-compatibele LoRa-Nexus-board [3] te richten. De code heeft tot taak de hardware te initialiseren en elke 120 seconden een regel nieuwe meetwaarden te leveren. Hiervoor moeten we enkele libraries inrichten: een gepatchte versie van de *DS2401*-library die als download op GitHub is te vinden [12], en een op Paul Stoffregens *OneWire library* [13] gebaseerde library. Om stroombesparende maatregelen te kunnen nemen, moet ook de *LowPower*-library [14] worden geïnstalleerd. En om ook gebruik te kunnen maken van de luchtvochtigheids- en temperatuursensor, hebben we nog de library *DHT* van Adafruit [15] nodig. Tenslotte: om de LoRa-module aan de praat te krijgen hebben we nog de gepatchte versie van *RadioHead* nodig, die eveneens in het bij dit project horende download-pakket [11] te vinden is.

## LoRa-PHY

De communicatiestandaard LoRa (Long Range) is speciaal ontworpen om gegevens van IoT-sensoren over lange afstanden (10 km en meer, afhankelijk van de omstandigheden) naar een ontvanger te verzenden, waarbij moet worden opgemerkt dat dit niet sneller gaat dan 50 kbit/s. LoRa is vooral geschikt voor niet-realtime-dataoverdracht, waarin een vertraging geen probleem is. LoRa-apparaten zenden in Europa in de ISM-band (433,05..434,79 MHz) of (zoals in deze toepassing) in de SRD-band (863..870 MHz).



Bij Lora gaat het om de onderste, fysieke laag in het OSI-model, die kan worden gebruikt door hogere lagen zoals LoRaWAN (Long Range Wide Area Network), maar natuurlijk ook door andere hogere lagen.

In tegenstelling tot traditionele – op FSK-modulatie gebaseerde – draadloze systemen, gebruikt LoRa een soort *chirp spread spectrum*-

modulatie (CSS), uitgevonden door Semtech. Door variatie van de spreidingsfactor kan LoRa de datarate op de gevoeligheid bij een vaste bandbreedte afstemmen en haalt hierdoor een enorme reikwijdte met een extreem lage vermogens/overdracht-ratio van ca. 155...170 dB. Bovendien is LoRa dankzij Forward Error Correction Coding relatief ongevoelig voor storingen.

Nadere basisinformatie over LoRa is te vinden in het developers-portal van Semtech [18].

De firmware van de ontvanger wordt in de Arduino-IDE geladen, alle genoemde libraries ingesloten en het programma gecompileerd. Via de FTDI-kabel (3,3 V) wordt de gecompleteerde firmware naar het LoRa Nexus board overgedragen. Alle hier genoemde programmacodes kunnen van de GitHub-repository [11] van Elektor worden opgehaald.

### Inbouwen?

Iets wat auteur Mathias Claußen pas aan zijn werkbank in het Elektor Lab ontdekte, is dat de RFM95 niet met draadverbindingen mag worden aangesloten, omdat dit aanleiding kan geven tot instabiliteit en ongewenste resets. Daarom is de Radiohead-library enigszins gepatcht om de modem-configuratie te controleren en als tijdens bedrijf iets niet blijkt te kloppen, een re-initialisatie door te voeren.

Daarmee werd de laboratoriumtest van het op een klein breadboard opgebouwde prototype afgesloten. Als u nu zou vragen of de voltmeter daarmee klaar was om in een voertuig te worden ingebouwd, kunnen we niet anders dan met een gedecideerd **nee!** antwoorden. Als elektronica in een voertuig moet worden ingebouwd,

 **IN DE STORE**

- **Elektor ESP32 Smart Kit**  
[www.elektor.nl/elektor-esp32-smart-kit](http://www.elektor.nl/elektor-esp32-smart-kit)
- **LoRa Nexus Board**  
[www.elektor.nl/lora-nexus-board-arduino-mini-shape](http://www.elektor.nl/lora-nexus-board-arduino-mini-shape)
- **RFM95 Ultra LoRa Transceiver-module**  
[www.elektor.nl/rfm95-ultra-lora-transceiver-module-868-915-mhz](http://www.elektor.nl/rfm95-ultra-lora-transceiver-module-868-915-mhz)
- **Raspberry Pi 3 B+ Starter Kit**  
[www.elektor.nl/raspberry-pi-3-model-b-plus-starter-kit](http://www.elektor.nl/raspberry-pi-3-model-b-plus-starter-kit)
- **FT232R-USB/Serial Bridge/BOB**  
[www.elektor.nl/ft232r-usb-serial-bridge-bob-110553-91](http://www.elektor.nl/ft232r-usb-serial-bridge-bob-110553-91)
- **Walter Trojan, "Das MQTT-Praxisbuch"**  
[www.elektor.nl/das-mqtt-praxisbuch](http://www.elektor.nl/das-mqtt-praxisbuch)

is een geschikte protectieschakeling voor de hardware een absolute vereiste; deze moet enerzijds stoorsignalen uit het boordnet tegenhouden en anderzijds de elektronica beschermen.

Welke problemen u zoal kunt tegemoeten als u eigen componenten in de auto-elektronica wilt integreren, is uitgebreid beschreven in het artikel 'Automotive-stoorbronnen' in Elektorlabs maart/april 2019 [16]. Welke maatregelen moe-

ten worden getroffen om de voltmeter tegen stoorbronnen te beschermen, hangt echter zó sterk af van het voertuig en de inbouwomstandigheden, dat we daarvoor in dit artikel geen algemeen recept kunnen geven. ▶

(180364-04)

### Weblinks

- [1] Project op Elektor Labs: [www.elektormagazine.nl/labs/wireless-car-voltmeter](http://www.elektormagazine.nl/labs/wireless-car-voltmeter)
- [2] LoRa-module: [www.hoperf.com/modules/lora/index.html](http://www.hoperf.com/modules/lora/index.html)
- [3] LoRa Nexus Board: [www.elektor.nl/lora-nexus-board-arduino-mini-shape](http://www.elektor.nl/lora-nexus-board-arduino-mini-shape)
- [4] DC/DC-module Würth 173010578: [https://katalog.we-online.de/de/pm/MAGIC\\_FDSM\\_FIXED\\_OUTPUT\\_VOLTAGE](https://katalog.we-online.de/de/pm/MAGIC_FDSM_FIXED_OUTPUT_VOLTAGE)
- [5] RMF95 als LoRa-modem: [www.elektor.nl/rfm95-ultra-lora-transceiver-module-868-915-mhz](http://www.elektor.nl/rfm95-ultra-lora-transceiver-module-868-915-mhz)
- [6] Radiohead-bibliotheek: <https://github.com/hallard/RadioHead>
- [7] Korte inleiding tot MQTT: [https://wiki.fhem.de/wiki/MQTT\\_Einf%C3%BChrung](https://wiki.fhem.de/wiki/MQTT_Einf%C3%BChrung)
- [8] MQTT-praktijkboek (Duitstalig): [www.elektor.nl/das-mqtt-praxisbuch](http://www.elektor.nl/das-mqtt-praxisbuch)
- [9] Gigantische LED-klok: [www.elektormagazine.nl/180254-03](http://www.elektormagazine.nl/180254-03)
- [10] BCM2835: [www.airspayce.com/mikem/bcm2835/](http://www.airspayce.com/mikem/bcm2835/)
- [11] GitHub Repository van dit project: <https://github.com/ElektorLabs/180364-wireless-car-multimeter>
- [12] DS2401-bibliotheek: [https://github.com/sindrehal/Arduino\\_DS2401](https://github.com/sindrehal/Arduino_DS2401)
- [13] OneWire-bibliotheek: <https://github.com/PaulStoffregen/OneWire>
- [14] LowPower-bibliotheek: <https://github.com/rocketscream/Low-Power>
- [15] DHT-sensorbibliotheek: <https://github.com/adafruit/DHT-sensor-library>
- [16] Automotive-stoorbronnen: [www.elektormagazine.nl/180345-03](http://www.elektormagazine.nl/180345-03)
- [17] Mosquitto: <http://mosquitto.org/>
- [18] LoRa developers portal: <https://lora-developers.semtech.com/>
- [19] Gigantische LED-klok op Elektor Labs:  
[www.elektormagazine.com/labs/bedroom-clock-with-out-side-temperature-based-on-esp32](http://www.elektormagazine.com/labs/bedroom-clock-with-out-side-temperature-based-on-esp32)

# Tijdcorrecte basweergever voor het laagste laag

Jan Breemer (Nederland)



Hoewel de zelfbouw van audio-apparatuur (versterkers, luidsprekerboxen) tegenwoordig nauwelijks nog loont (tenminste wat het kostenplaatje betreft), is er gelukkig nog altijd een harde kern van zelfbouwers actief. Zo ook in dit geval. Het was de bedoeling van de auteur om een basweergever te ontwerpen die ook de allerlaagste frequenties zo goed mogelijk weergeeft. In dit artikel beschrijft de auteur de overwegingen die tot het gewenste resultaat hebben geleid.

Om teleurstellingen te voorkomen: in dit artikel wordt geen kant-en-klaar nabouw-ontwerp beschreven. Veeleer wil de auteur de geïnteresseerde lezer voldoende informatie aanreiken om zelf een basweergever te ontwikkelen en te bouwen.

## De problematiek

De meeste commerciële audiosystemen (en ook de meeste zelfbouwsystemen) zijn gebaseerd op een gewone stereoversterker, dus met een versterker voor het rechter en een voor het kanaal in dezelfde kast, die het gehele audiospectrum voor zijn rekening neemt. De luidsprekersystemen zijn in dat geval van passieve wisselfilters voorzien die het audiospectrum in twee of drie delen splitsen en naar de daarvoor geoptimaliseerde luidsprekers sturen.

En bij die wisselilters beginnen de problemen. In de meeste gevallen bederven ze de demping van de luidsprekers; ook maken ze de aanpassing aan de gevoeligheid van de afzonderlijke luidsprekers lastig. En tenslotte zijn de gebruikte onderdelen in vele gevallen behoorlijk 'aan de prijs'.

Een ander probleem is dat de meeste (lagetonen-)luidsprekerkasten volgens het basreflex-principe zijn opgebouwd, waarbij het inwendige van de kast via een reflecttunnel met de buitenwereld in verbinding staat. Deze reflexsystemen geven zeer lage frequenties (beneden ongeveer 30 Hz) niet of nauwelijks weer. Erger is dat deze systemen zich beneden de resonantiefrequentie als een steil (vierde-orde) hoogdoorlaatfilter gedragen. Die steile filtering veroorzaakt in- en uitslengverschijnselen die de impulsrespons

van het systeem niet ten goede komen. En tenslotte: als we zo'n box toch met extreem lage frequenties aansturen, zal de conus van de luidspreker weliswaar een zeer forse uitslag vertonen, maar hoorbaar geluid wordt daarbij niet geproduceerd ten gevolge van akoestische kortsluiting. Bovendien worden de hogere frequenties door deze extreme uitslagen vervormd.

## De aanpak

Een geheel gesloten box biedt een aantal voordelen, reden dat de auteur die als uitgangspunt voor zijn overwegingen heeft gekozen. Beneden de resonantiefrequentie valt de geluidsdruck minder steil af met 12 dB/octaaf, dus als een tweede-orde hoogdoorlaatfilter. Ter compensatie wordt een speciaal filter vóór de eindtrap opgenomen (terzijde: elke

luidspreker, dus ook de lagetonenweer-gever, heeft zijn eigen eindversterker); het bijzondere van dit filter is de correctie/compensatie van de eigenresonantie van het systeem.

En wat ook niet onbelangrijk is: het gedrag van een gesloten box kan relatief eenvoudig met goede nauwkeurigheid worden gemodelleerd.

Om het leven van de geïnteresseerde lezer te veraangenamen, heeft de auteur een computerprogramma ontwikkeld dat het moeilijke rekenwerk voor zijn rekening neemt en de resultaten grafisch weergeeft; dit programma kan gratis worden gedownload van de webpagina bij dit artikel (uitvoerbare bestanden voor Linux en Windows, en de broncode) [1].

## Twee heren

Het ontwikkelen van een goede luidsprekerbox is tegenwoordig geen extreem langdurig proces van proberen, aanpassen, weer proberen, weer aanpassen en zo vervolgens – en dat danken we vooral aan de twee heren Thiele en Small. De naar hen genoemde Thiele/Small-parameters kunnen prima worden gebruikt om het gedrag van een basweergave (we bedoelen dan de combinatie van luidspreker en behuizing) met een heel behoorlijke accuratesse te voorspellen. We hebben de belangrijkste TS-parameters in **tabel 1** samengevat. Dit zijn overigens niet alle TS-parameters – er zijn er nog meer, maar hun invloed op het uiteindelijke resultaat is betrekkelijk gering zodat we die hier buiten beschouwing laten.

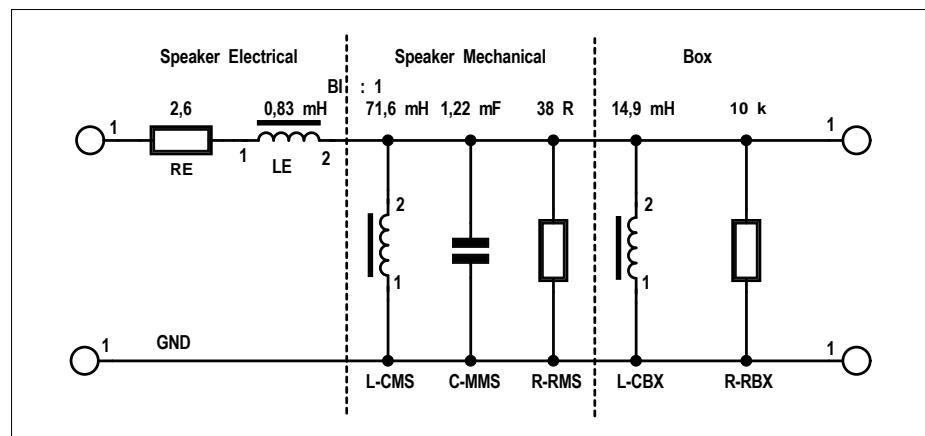
## Rekenmodel

We hebben nu een heleboel mechanische en akoestische parameters c.q. eigenschappen van de luidspreker en de beoogde box; om daarmee comfortabel te kunnen rekenen en op papier (dat was vroeger, tegenwoordig gebruiken we een simulator) te experimenteren, rekenen we deze om naar elektrische equivalen-ten. Dat maakt het mogelijk een willekeurige simulator te gebruiken.

Le en Re zijn al elektrische grootheden; die kunnen we meteen in het model stoppen. Voor de omrekening van de mechanische grootheden nemen we een logische benadering waarbij de stroom door de spreekspoel evenredig is met de kracht op de conus. Dan blijkt de massa van de conus gelijkwaardig te zijn aan een capaciteit, terwijl de veerkracht van de ophanging en van de lucht in de box als zelfinducties in het model terecht komen.

**Tabel 1.**

Parameter	Omschrijving	Eenheid
Mms	massa van de conus + meegesleepte lucht	kg
Cms	compliantie (beweeglijkheid) van de conus-ophanging	m/N
Rms	mechanische weerstand (demping) van de conus	kg/s
Re	ohmse weerstand van de spreekspoel	$\Omega$
Le	zelfinductie van de spreekspoel	H
Bl	'krachtfactor' van de spreekspoel	N/A of Tm
Sd	effectieve oppervlakte van de conus	$m^2$
Xmax	maximaal toelaatbare conusuitwijking +/-	mm
Cbx	compliantie van de lucht in de box	m/N
Rbx	demping door dempingsmateriaal (zoals steenwol) in de box	kg/s
Qtc	totale kwaliteitsfactor van het luidsprekerchassis	-



Figuur 1. Het rekenmodel voor de luidspreker/box-combinatie. De waarden gelden voor de ScanSpeak 30W4558T00 in een gesloten box met een volume van 41 l.

De omrekenfactor kunnen we ons voorstellen als een 'ideale' transformator met wikkelverhouding  $BI:1$ . Via deze 'trans-formator' wordt mechanische kracht vertaald naar elektrische stroom en snelheid naar elektrische spanning (en omgekeerd, natuurlijk). En net als bij een gewone transformator verschijnt bij de vertaling van impedanties de parameter  $Bl$  in het kwadraat.

In de vorm van een equivalent elektrisch schema gegoten ziet het rekenmodel er uit als geschetst in **figuur 1**.

Samenvattend zien de omrekeningen er als volgt uit:

$$\begin{aligned} L-CMS &= C_{MMS} * Bl^2 \\ C-MMS &= M_{MS} / Bl^2 \\ R-RMS &= Bl^2 / R_{MS} \end{aligned}$$

$$\begin{aligned} L-CBX &= C_{BX} * Bl^2 \\ R-RBX &= Bl^2 / R_{BX} \end{aligned}$$

De compliantie (meegevendheid) van de lucht in de box hangt – uiteraard – van het volume van de box af, en van het (kwadraat van het) effectieve oppervlak van de luidsprekerconus (met de afleid-ing van deze relatie zullen we u niet lastigvallen):

$$C_{BX} = 7,14 * 10^{-6} * V_{box} / S_d^2 \quad [\text{m}/\text{N}]$$

## Nu de praktijk

Tot dusver hebben we een aantal theoreti sche relaties op een rijtje gezet en een elektrisch rekenmodel opgesteld – allemaal mooi en prachtig maar nog niet echt bevredigend, want wat kunnen we daar nu eigenlijk mee?

Om het concreet te stellen: waar halen

Beschreibung	Technische Details	Diagramme + Abmessungen	PDF	Bewertungen	Garantie
<b>KURZDATEN</b>					
Einsatz als	Tiefotoner				
Belastbarkeit Sinus	350 Watt	Frequenz ab ca.	17 Hz		
Impedanz	4 Ohm	Max. Frequenz	1.5 kHz		
		Kennschalldruck	89 dB 1W/1m		
<b>CHASSIS-DATEN</b>					
Außenmaß	308 mm	Schw.-spulen Ø	51 mm		
Einbaumaß	280 mm	Schw.-spulen-Höhe	33 mm		
Einbautiefe (Gesamt)	144.3 mm	Luftspalt-Höhe	8 mm		
Korbmaterial	Aluminium-Druckguss	Magnet Ø	121 mm		
Membrantyp	Konus				
Membransicke	Gummi				
Membranmaterial	Aluminium				
Dustcap Material	Fiberglas				
<b>THIELE-SMALL-PARAMETER</b>					
Resonanzfrequenz	fs: 17 Hz	Nachgiebigkeit	Cms: 0.65 mm/N		
mech. Güte	Qms: 5.01	Bewegte Masse	Mms: 135 g		
elektr. Güte	Qes: 0.34	mech. Dämpfung	Rms: 2.88 kg/s <sup>-1</sup>		
Gesamtgüte	Qt: 0.32	Membranhub +/-	Xmax: 28 mm		
Äquivalent-Volumen	Vas: 197 Liter	Kraftfaktor	Bl: 10.5 N/A		
DC-Widerstand	Re: 2.6 Ohm	Membran Ø	Md: 244		
Induktivität	Le: 0.83 mH	Membran-Fläche	Sd: 466 cm <sup>2</sup>		
		Schalldruck	dB: 89 SPL		

Figuur 2. De TS-parameters van luidsprekers van gerenommeerde fabrikanten zijn niet moeilijk te vinden.

we de waarden vandaan die we in al die vergelijkingen en in het rekenmodel moeten invullen? En wat komt er dan uiteindelijk uit rollen?

Het eenvoudigste eerst: elke zichzelf respecterende fabrikant van kwaliteitsluidsprekers heeft de TS-parameters van zijn luidsprekerchassis naar eer en geweten gemeten, en stelt die gratis ter beschikking. De auteur heeft voor het luidsprekersysteem dat hij met behulp van bovenstaande overwegingen heeft ontwikkeld een ScanSpeak 30W4558T00 gebruikt. Een korte speurtocht op het internet levert probleemloos de benodigde waarden op (**figuur 2**).

Met de waarden uit figuur 2 zijn ook de ‘onderdelen’-waarden berekend die in figuur 1 zijn gegeven.

En nu wordt het leuk: we gaan de overdracht van het luidsprekersysteem bepalen – want daar gaat het ons uiteindelijk om: hoe de uitgangsspanning van de versterker wordt vertaald naar een geluidsdruck (SPL = Sound Pressure Level, meestal gespecificeerd op een afstand van 1 m van de luidspreker).

Daartoe bedenken we dat de elektrische spanning aan de uitgang van het rekenmodel (figuur 1) evenredig is met de snelheid waarmee de conus beweegt – en het zal (gevoelsmatig) niemand ver-

bazen dat de evenredigheidsfactor weer de krachtfactor Bl is. Er geldt:

$$v = U / Bl \quad [\text{m/s}]$$

met  $v$  = conussnelheid en  $U$  = uitgangsspanning van het elektrische rekenmodel.

Wanneer we dat combineren met het oppervlak van de conus, vinden we voor het volume  $V_{\text{lucht}}$  van de verplaatste lucht:

$$V_{\text{lucht}} = v * S_d \quad [\text{m}^3/\text{s}]$$

In 1954 heeft Leo Leroy Beranek afgeleid dat de geluidsdruck SPL op een afstand r van een bolvormige straler (of een puntvormige geluidsbron) in de vrije ruimte gelijk is aan

$$P_{\text{SPL}} = V_{\text{lucht}} * \rho * f / (2 * r) \quad [\text{N/m}^2]$$

Hierin is  $\rho$  de dichtheid van de lucht (1,2 kg/m<sup>3</sup>). In bovenstaande vergelijking is verondersteld dat de geluidsbron klein is ten opzichte van de golflengte van het geluid; als we een frequentie van 30 Hz aannemen (het hele verhaal gaat immers over lagetonenweergevers) bedraagt de golflengte 10 m zodat we de formule met een gerust hart mogen gebruiken.

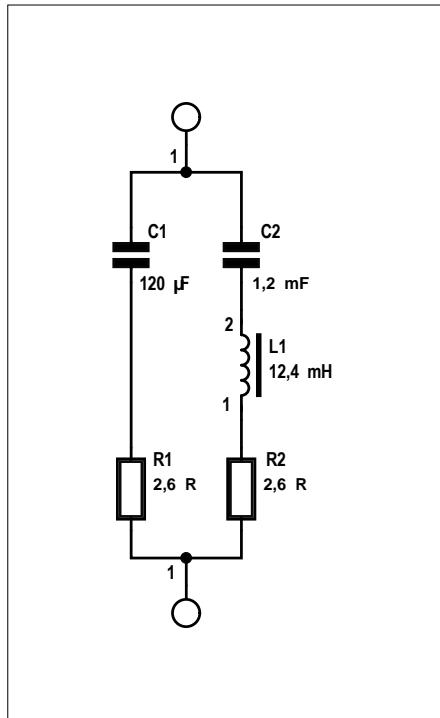
We kunnen de afstand r beschouwen als de straal van een bol met de geluidsbron in het midden. Nu komt een vrij in de ruimte hangende geluidsbron in de praktijk eigenlijk niet voor; we hebben te maken met een reële luidsprekerbox die op de vloer staat. Dat betekent dat de geluidsenergie zich niet over een hele maar over een halve bol verdeelt en dat levert een tweemaal zo grote geluidsdruck op.

Tenslotte rekenen we de geluidsdruk nog om naar dB's en daarvoor gebruiken we het referentieniveau:

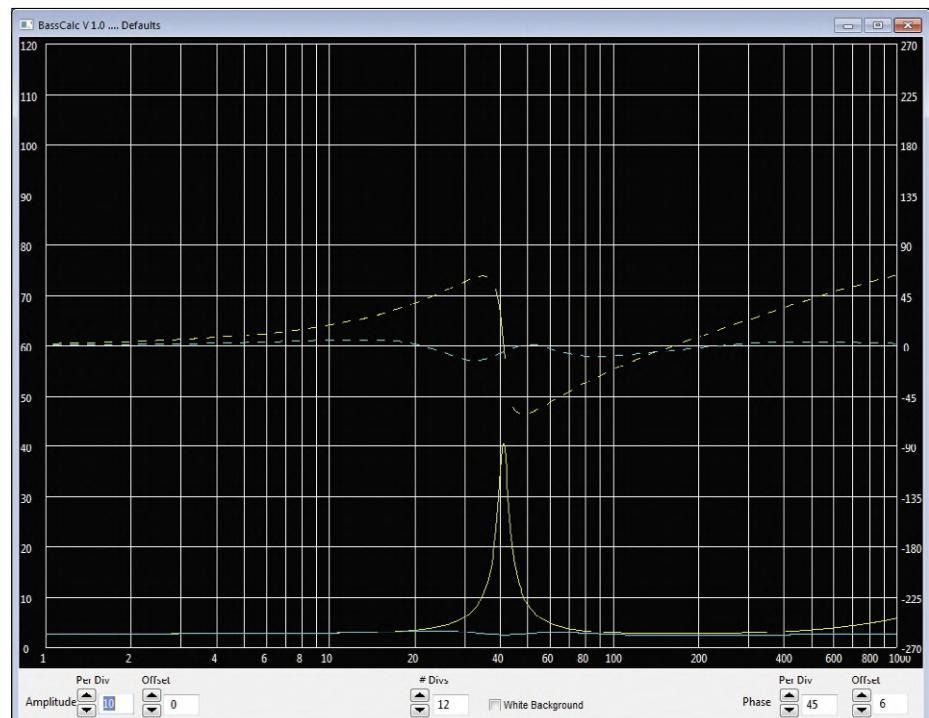
$$0 \text{ dB} = 2 * 10^{-5} \quad [\text{N/m}^2].$$

## Links en literatuur

- [1] Webpagina bij dit artikel: <http://www.elektormagazine.nl/180585-01>
- [2] Zobel-netwerk: [https://en.wikipedia.org/wiki/Zobel\\_network](https://en.wikipedia.org/wiki/Zobel_network)
- [3] Gyrator: <https://nl.wikipedia.org/wiki/Gyrator>
- [4] J. D'Appolito: Luidspreker-meettechniek (Uitgeverij Segment BV, 2000, ISBN 90 5381 116 8):
- [5] Vance Dickason: Luidsprekerkasten ontwerpen, herziene en uitgebreide editie (Uitgeverij Segment BV, 2001, ISBN 90 5381 137 0)
- [6] H.H. Klinger: Het luidspreker-bouwboek (Uitgeversmij. Elektuur BV, 1990, ISBN 90 70160 84 6):
- [7] Diamond-luidsprekersysteem: [www.temporalcoherence.nl](http://www.temporalcoherence.nl)
- [8] Een onconventioneel luidsprekersysteem: [www.breem.nl/lsp/LoudspeakerSystem.htm](http://www.breem.nl/lsp/LoudspeakerSystem.htm)



Figuur 3. Het Zobel-netwerk compenseert voor de resonantiepiek van een luidspreker.



Figuur 4. Hier is de invloed van het Zobel-netwerk duidelijk te zien: geel is het impedantieverloop zonder netwerk en blauw met Zobel-netwerk.

## Zobel

Voor audiofielen is altijd interessant te weten welke impedantie de uitgang van de versterker ‘ziet’. Verreweg de meeste mensen maken zich daar niet al te druk over, maar: een versterker vindt het doorgaans prettiger om een belasting aan te sturen die zich zo goed mogelijk ohms gedraagt. De vervorming zal wat geringer zijn als stroom en spanning ongeveer gelijke fase hebben.

En dan kan een Zobel-netwerkje goede diensten bewijzen.

Dit netwerkje wordt parallel aan de luidsprekerklemmen aangesloten. Op de totale overdracht heeft het nauwelijks invloed, maar het heeft wel invloed op de stroom die de versterker moet leveren. Het netwerk bestaat uit twee delen; het ene deel compenseert voor de zelfinductie  $Le$  van de spreekspoel, terwijl het andere deel compenseert voor de resonator die wordt gevormd door de massa en de veerkracht van de luidspreker in de box.

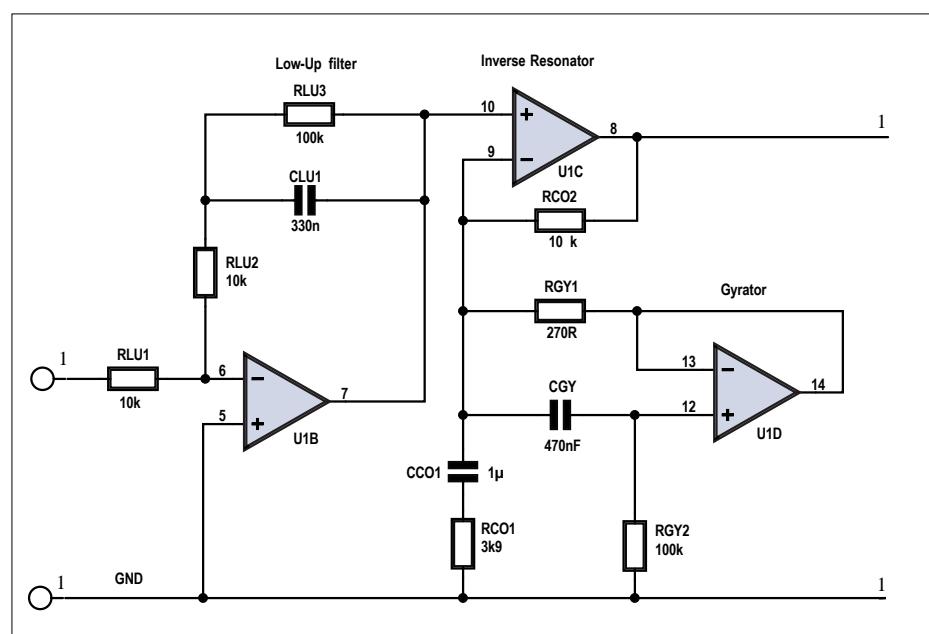
Het is niet bijzonder moeilijk de componentenwaarden te berekenen. Voor de beide weerstanden nemen we dezelfde waarde als de  $Re$  van de luidspreker (in het ScanSpeak-voorbeeld dus  $2,6 \Omega$ ).  $C1$  moet samen met  $R1$  dezelfde tijd-constante hebben als  $Re$  met  $Le$ . Dan vinden we:

$$C1 = Le / (Re * R1)$$

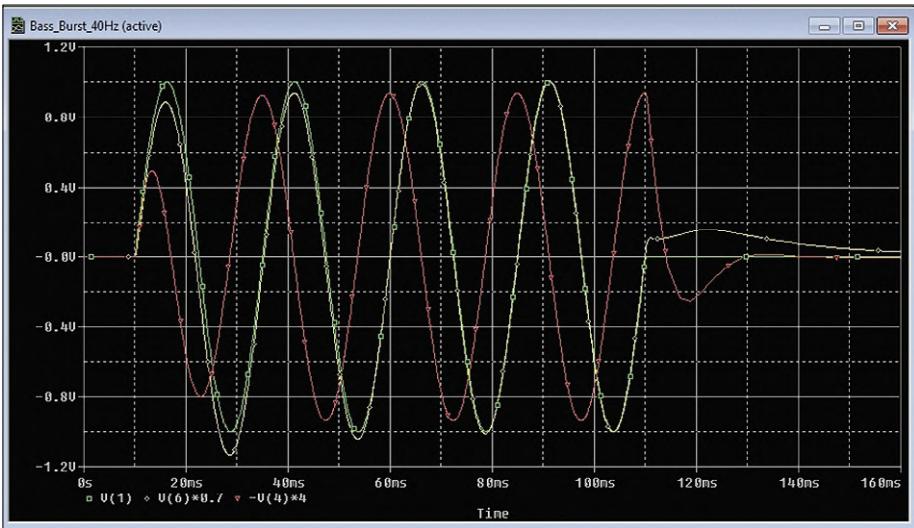
Voor de zuigkring nemen we  $C-MMS$  (zie figuur 1) als waarde voor  $C2$ , en voor  $L1$  nemen we de parallelschakeling van  $L-CMS$  en  $L-CBX$  (zie weer figuur 1).

Toegegeven:  $C2$  met een waarde van

$1,22 \text{ mF}$  is een ‘dikkerdje’ indien we daarvoor een foliecondensator zouden nemen. Het gaat echter ook met twee elco’s in anti-serie (waarbij het midden van die serieschakeling aan de voedingsspanning van de versterker wordt gelegd zodat de elco’s nooit verkeerd gepoold kunnen worden).



Figuur 5. Het schema van het compensatiefilter. De onderdelenwaarden gelden weer voor de ScanSpeak-luidspreker van de auteur; de benamingen van de onderdelen corresponderen met die in het BassCalc-programma.



Figuur 6. Spice-simulatie van het luidsprekersysteem met en zonder compensatiefilter. Groen: ingangssignaal (een burst van 4 perioden van een 40-Hz-sinus); rood: responsie zonder compensatie; geel: responsie met compensatie. Verder commentaar overbodig...

Ook spoel L1 is met 12,4 mH niet bepaald klein. In theorie is hier een gyrator mogelijk, maar dat heeft als nadeel dat dan een van de luidsprekeraansluitin-

gen aan massa moet liggen. Daarom heeft de auteur een spoel met (gesloten) ijzerkern gebruikt; dat is hier geen probleem omdat het Zobel-netwerk niet

in de signaalweg zit. We moeten alleen oppassen dat de kern niet volledig in verzagiding gaat.

In **figuur 4** is de zegenrijke invloed van het Zobel-netwerk duidelijk te zien.

### Compensatiefilter

Nu komen we bij het laatste deel van onze overwegingen: het compensatiefilter dat de afval van de lagetonen beneden de resonantiefrequentie van de box (12 dB/octaaf, zoals al eerder aangestipt) compenseert. Bovendien compenseert dit filter de eigenresonantie van het systeem. Het resultaat is een uitzonderlijk 'strakke' lagetonenweergave.

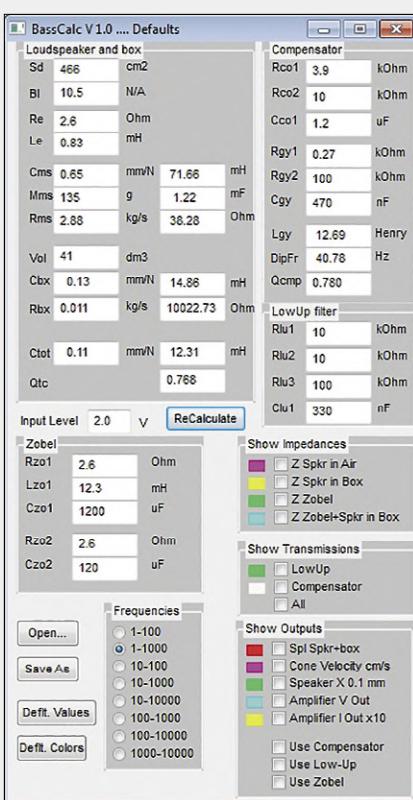
Het schema van dit compensatiefilter is getekend in **figuur 5**. Het bestaat weer uit twee delen: een ophaalversterker voor de lage tonen, en een 'antiresonator' om de resonantie van de luidspreker in de box te compenseren.

De ophaalversterker is opgebouwd rond U1B. RLU3 en CLU1 bepalen de frequentie waarbij het ophalen van de lage tonen begint; bij de hier gegeven dimensionering (Voor de ScanSpeak-luidspreker in

## De BassCalc-software

Dit programma (**figuur 7** toont de gebruikersinterface) is ontwikkeld met Free Pascal en de Lazarus IDE. Dit zijn tools die voor zo ongeveer elk platform (Windows, Linux, Mac, RPi, Android) gratis gedownload kunnen worden. De slogan is "Write Once, Compile Anywhere" en dat klopt in de praktijk heel behoorlijk, hoewel er altijd weer platformafhankelijke verrassingen optreden. Hoe het ook zij, het ermee gemaakte BassCalc-programma is getest op WinXP, Win7 en Linux Mint.

In het bedieningsvenster van figuur 7 worden in het veld linksboven de TS-parameters ingevuld (de default-waarden van het programma zijn de waarden van het door de auteur gebruikte Scan-Speak-chassis) en rechts daarnaast de equivalente waarden van het elektrische rekenmodel, plus de resulterende resonantiefrequentie en de totale Q-waarde. In het rechter veld staan de componentenwaarden voor het compensatiefilter en daaronder die voor het laag-ophaalfilter. Dan komen (onder de knop 'Recalculate') in het linker veld de waarden voor het Zobel-netwerk, en daaronder kunnen we aangeven over welk frequentiebereik de berekeningen moeten worden uitgevoerd en getoond. In de rechter velden kunnen



Figuur 7. De gebruikersinterface van het BassCalc-programma. De default-waarden zijn die van het door de auteur gebruikte ScanSpeak-chassis.

we aangeven welke grafieken we willen zien (impedanties, transmissies, outputs), en tenslotte kunnen we nog aangeven welke correcties moeten worden 'meegenomen'. Met de gekleurde vakjes kunnen desgewenst andere kleuren worden ingesteld. De knoppen 'Open' en 'Save' bieden de mogelijkheid een set gegevens op te slaan en weer op te halen. De knoppen 'Deflt. Values' en 'Deflt. Colors' tenslotte herstellen de standaardwaarden van het programma. Wanneer een componentwaarde is veranderd, moet de knop 'Recalculate' worden gedrukt.

Bij 'Input Level' wordt de ingangsspanning aangegeven die aan de luidspreker wordt toegevoerd. 2,38 V komt overeen met 1 W in 8 Ω. Voor een 4-Ω-exemplaar als de ScanSpeak van de auteur is 2 V nodig voor 1 W. Opmerking: waarden worden ingevoerd met decimale punt, niet met decimale komma.

Het programma hoeft niet te worden geïnstalleerd: kopieer het uitvoerbare bestand naar een map van uw keuze en voer het eenvoudig uit.

Opmerking: wanneer 'mallotige' waarden worden ingevuld, zal het programma ook mallotige waarden retourneren – of crashen.

een box met een volume van 41 liter) is dat 4,8 Hz. RLU2 en CLU1 bepalen de frequentie waarbij de extra versterking weer ophoudt; hier ongeveer 48,5 Hz. Deze frequentie-afhankelijke versterker wordt gevuld door een – nou ja, laten we het een inverse resonator noemen, opgebouwd rond U1C. De versterking van deze opamp is afhankelijk van de impedantie van de min-ingang naar GND, bestaande uit condensator CCO1, weerstand RCO1 en de gyrorond U1D. Zo'n gyrorond is een uiterst interessante schakeling waar we gemakkelijk een halve Elektorlabs over zouden kunnen volschrijven – maar dat doe we hier toch maar niet; we verwijzen naar de uitgebreide literatuur die op het internet te vinden is, onder andere [3].

Een gyrorond is simpel gesteld een halfgeleiderschakeling die zich als een spoel gedraagt, maar dan zonder spoel. Voor de equivalente zelfinductie van de gyrorond in de schakeling van figuur 5 geldt:

$$L = CGY * RGY1 * RGY2$$

[waarden in farad, ohm en henry]

Deze equivalente zelfinductie staat parallel aan CCO1 en RCO1, zodat in wezen een parallelring wordt gevormd, voor de resonantiefrequentie waarvan geldt:

$$f = 1 / (2 * \pi * \sqrt{L * C})$$

wat met de onderdelenwaarden van figuur 5 uitkomt op ongeveer 40 Hz. Wanneer u met dit schema wilt gaan experimenteren, zijn dit redelijke uitgangspunten: de resonantiefrequentie en Q van de inverse resonator kunnen het beste gelijk worden gemaakt aan die van het luidsprekermodel. Voor het instellen van de Q-factor kunt u de waarde van RCO1 variëren, want die heeft geen invloed op de resonantiefrequentie. De kantelfrequentie van het laag-op-filter kan het beste ook in de buurt van de resonantiefrequentie worden gekozen. In het BassCalc-programma (zie kader) kunt u met allerlei waarden en parameters stoeien zonder uw soldeerbout warm te stoken; mocht u besluiten uw eigen versie van deze schakeling te gaan bouwen, kunt u het beste opamps met een extreem geringe vervorming gebruiken, zoals de OPA134.

## Het resultaat

Na al deze theorie toont **figuur 6** wat het resultaat van onze overwegingen is:

## Een onconventioneel luidsprekersysteem

**Figuur 8** toont het luidsprekersysteem dat resulteerde uit de theoretische overwegingen in dit artikel. Het is een driewegsysteem waarbij elke luidspreker zijn eigen (klasse D-)versterker heeft gekregen. De onderste vijfhoekige behuizing herbergt de ScanSpeak-lagetonenweergever; deze reproduceert in deze configuratie frequenties tot ruim onder 20 Hz. Het systeem als geheel is geïnspireerd door het ‘Diamond’-systeem van Hans van Maanen [7].

De auteur heeft een uitgebreide (Engelstalige) beschrijving van zijn luidsprekersysteem op het internet beschikbaar gesteld [8]. Bij voldoende belangstelling kunnen we hier in een toekomstige Elektorlabs-uitgave op terugkomen.



Figuur 8. Het drieweg-luidsprekersysteem zoals de auteur het heeft opgebouwd, op basis van de overwegingen uit dit artikel.

een Spice-simulatie van de lagetonenweergever die de auteur heeft ontworpen (zie kader) en die de basis van dit artikel vormde.

In de figuur is het ingangssignaal (een burst van 4 perioden van een 40-Hz-sinus) groen weergegeven. In rood is de respons van de ScanSpeak-luidspreker in zijn 41-l-box zonder compensatie te zien. Het moge duidelijk zijn: de weergever heeft duidelijk moeite om op gang te komen, en blijft daarna duidelijk achter bij het ingangssignaal.

In geel is de responsie van het systeem te zien met compensatie: de weergever volgt het ingangssignaal vrijwel perfect. En nu zal ook duidelijk zijn wat wordt bedoeld met *tijdcorrecte* basweergever!

Natuurlijk is een simulatie niet anders dan dat: een simulatie. Niettemin klinkt het door de auteur gebouwde systeem opmerkelijk helder en gedetailleerd – de erin geïnvesteerde moeite (theoretisch en praktisch) bleek de moeite waard.

Zoals al eerder opgemerkt: dit is geen kant-en-klaar nabouwproject; de geïnteresseerde lezer kan gebruiken wat hij of zij wil en dat in een eigen systeem integreren. Het is altijd mogelijk om met de auteur contact op te nemen via de redactie [1].

(180585-01)



### IN DE STORE

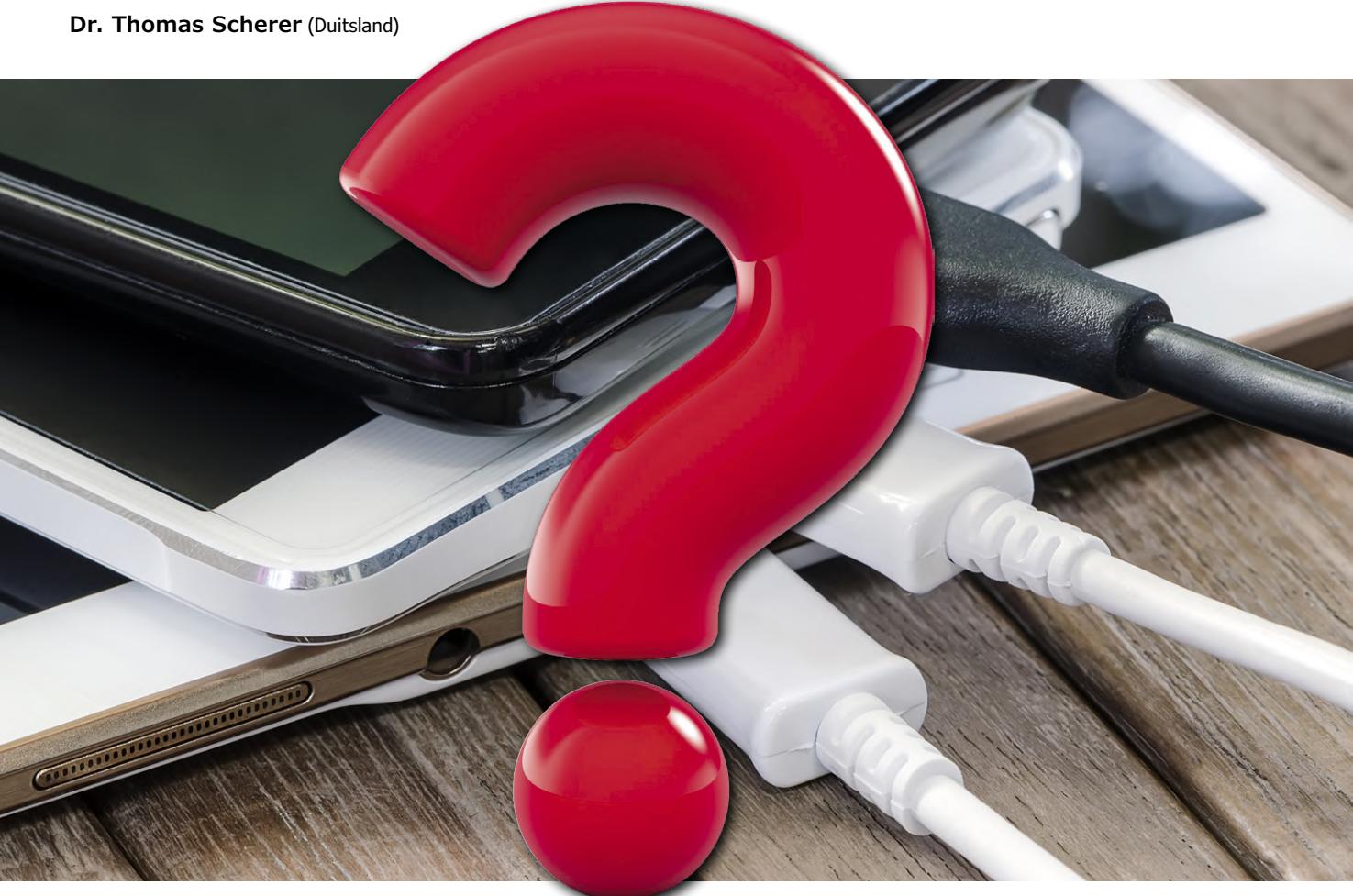
- E-boek: Introductie tot de elektro-akoestiek  
[www.elektor.nl/introductie-tot-de-elektro-akoestiek-e-book](http://www.elektor.nl/introductie-tot-de-elektro-akoestiek-e-book)
- Elektuur audio-special 1  
[www.elektor.nl/elektuur-audio-special-1-dutch](http://www.elektor.nl/elektuur-audio-special-1-dutch)
- Elektor audio-special 2  
[www.elektor.nl/elektor-audio-special-2-dutch](http://www.elektor.nl/elektor-audio-special-2-dutch)
- Elektor audio-special 3  
[www.elektor.nl/elektor-audio-special-3-dutch](http://www.elektor.nl/elektor-audio-special-3-dutch)
- Elektor audio-special 4  
[www.elektor.nl/elektor-audio-special-4-dutch](http://www.elektor.nl/elektor-audio-special-4-dutch)



# Qi-laders

## zo gaat inductief opladen van smartphones

Dr. Thomas Scherer (Duitsland)



Moderne smartphones kunnen zo ongeveer alles behalve lang op een acculading werken. Daarom verdient het aanbeveling om ze dagelijks op te laden. Als ze in de auto voor navigatie worden gebruikt, moet u vanwege het hoge stroomverbruik bijna altijd een laadkabel gebruiken. Het gevolg is: insteken, uittrekken, insteken, uittrekken... Dat is niet alleen lastig, maar ook zijn met name de mini-USB-connectors van Android-toestellen niet geschikt voor zoveel handelingen. Gelukkig kunnen moderne mobiele telefoons, dankzij hun Qi-functionaliteit, inductief en dus contactloos worden opgeladen. Dit wilde de auteur voor zijn nieuwe iPhone X ook hebben; hier volgen zijn ervaringen.

Het voortdurende insteken/uittrekken van de laadkabel bij mijn smartphone was mij allang een doorn in het oog. Maar je wilt natuurlijk ook niet bij een lange wandeling met een halflege accu van start gaan. Dus waren een laadkabel met 'Lightning'-connector (voor mijn iPhone) en een met micro-USB-connector (voor vrienden, familie en experimenten) permanent met mijn PC verbonden. Als ik aan mijn bureau zat werd mijn iPhone hier regelmatig op aangesloten. Toen steeds meer high-end smart-

phones in de Android-omgeving met Qi-laadtechniek werden uitgerust, was ik eerst jaloers, en later boos over het feit dat Apple hier als technologisch boegbeeld en uitvinder van de smartphone ver mee achterliep. "Not invented here" heeft het bedrijf van de appel met een hap eruit er waarschijnlijk lang van weerhouden om zich laadtechnisch bij Samsung en consorten aan te sluiten. Maar bij recente update-acties bleek er toch enige beweging in te komen: in 2017 werden de iPhone 8

en de iPhone X eindelijk met Qi-functionaliteit uitgerust. En dat wilde ik direct uitproberen.

### Qi-laders

Bij elektrische tandenborstels wordt de inductieve laadtechniek al heel lang toegepast. Ver voor de uitvinding van de smartphone in 2007 werden bijvoorbeeld Oral-B-tandenborstels al contactloos opgeladen. De daarbij gebruikte techniek is erg eenvoudig en lijkt veel op een echte transformator met een grote luchtspleet omdat de zender (= laadstation) met een geïsoleerde staafvormige kern is uitgerust die in de ontvangerspoel steekt (**figuur 1**). Dit is makkelijk voor de gebruiker, en in een vochtige badkamer worden op deze manier stekkers en stopcontacten vermeden die ook bij lage spanningen problemen kunnen opleveren.

Vele jaren later werd het *Wireless Power Consortium* opgericht, een nieuw gremium dat met Qi een industriestandaard voor het contactloos opladen van mobiele apparaten introduceerde (zie het kader **De Qi-standaard**). Anders dan bij elektrische tandenborstels wordt hier de nodige elektronica ingezet, zodat a) de zendspoel niet permanent met maximale amplitude hoeft te werken en daarbij door dissipatieverliezen energie verspilt, b) het laadapparaat kan detecteren dat er een smartphone op wordt gelegd en c) wordt vastgesteld of de smartphone energie nodig heeft. Om ervoor te zorgen dat het geheel weliswaar gestandaardiseerd is maar niet al te ingewikkeld wordt, communiceren de lader en het mobiele apparaat volgens een eenvoudig protocol via het magnetische wisselveld.

Als u in de onderliggende techniek bent geïnteresseerd: Elektor heeft in november 2016 een project met de titel '**Draadloze power-converter**' gepubliceerd [1] waarmee u zelf een systeem voor contactloze energieoverdracht tot 50 W kunt bouwen. In dat artikel worden zowel het principe als de technische realisatie uitvoerig beschreven.

Wie zijn smartphone inductief wil opladen ziet vaak door de bomen het bos niet meer, want op dit moment bestaat er een grote verscheidenheid aan Qi-compatibele laadapparaten met allerlei verschillende eigenschappen en in alle denkbare uitvoeringen; en dat voor prijzen die variëren van minder dan een tientje tot meer dan het tienvoudige.

### Mijn eerste Qi-lader

Welk laadapparaat moet ik nemen? Deze vraag stelde ik mijzelf toen mijn nieuwe iPhone X werd geleverd, want het werd meteen duidelijk dat deze — eindelijk — inductief oplaadbaar zou zijn en tot mijn grote verbazing zelfs Qi-compatibel was. Apple heeft dus bij hoge uitzondering besloten om af te zien van een volledig fabriekseigen inductieve laadtechniek en de fans van het merk met de bijbehorende schandalig dure accessoires op te zadelen. Voor mij kwam dit goed uit, want nu kon ik het laadapparaat uitzoeken dat mij het beste leek. Ik deed dus eerst wat iedere technisch-geïnteresseerde tijdgenoot doet: op het internet kijken wat er beschikbaar is, en wat de ervaringen van andere gebruikers zijn.

Voor de eerste bruikbaarheidstest wilde ik niet teveel geld uitgeven, en dus koos ik voor het mooi platte en met € 12,85 zeker niet dure model C310 van Pulesen uit het verre China (**figuur 2**). Het gaat hier om een 5-W Qi-lader waarvan in elk geval één eerdere gebruiker had gemeld dat deze met zijn iPhone 8 goed werkte.

Deze en bijna alle andere Qi-laders worden via een micro-USB-B-connector van stroom voorzien. U hebt alleen nog een net-



Figuur 1. Voorloper van Qi: inductief laden van een Oral-B-tandenborstel.

### De Qi-standaard



Het Wireless Power Consortium [3] heeft al in 2009 de zogenaamde 'Qi low power specification' aangenomen. Inductieve laadapparaten die aan deze fabriekseigen standaard voldoen, kunnen contactloos tot maximaal 5 W aan elektrische energie overbrengen. De Medium-Power-versie uit 2011 is bedoeld voor vermogens tot 120 W en gaat hiermee ver boven het laden van smartphones uit. In 2015 werd zelfs een High-Power-variant gedefinieerd die voorziet in het inductief overdragen van energie tot 1 kW en die ook voor bijvoorbeeld keukenapparatuur geschikt is.

Het duurde even totdat de marktintroductie van Qi een feit was. In 2012 introduceerde Nokia de Lumia-920-smartphone en Google (LG) de Nexus 4 met deze functionaliteit.

Kort daarna kwamen er vaste Qi-laders in auto's van Toyota en Ssangyong, en tegenwoordig zijn Qi-laders voor smartphones niets bijzonders meer — zelfs Apple volgde in 2017 met de iPhone-modellen 8 en X.

Wat betreft de techniek bestaan Qi-systeem uit een platte zendspoel met seriecondensator (voor de resonantie) en de bijbehorende elektronica voor de aansturing. Daarbij behoort ook een gestandaardiseerd protocol voor de communicatie tussen zender en ontvanger. Hierbij worden magnetische wisselvelden tussen 105 en 205 kHz (typisch 140 kHz) gebruikt. Qi-laadapparaten voor smartphones werken met vermogens van 5 tot 10 W.

Magnetische wisselvelden doordringen zonder problemen niet-magnetische materialen, en daarom is de achterkant van de voor Qi geschikte smartphones van kunststof of glas gemaakt. Ook de veelgebruikte telefoonhoesjes van kunststof vormen (tot een materiaaldikte van 1 mm) geen probleem voor het inductief laden. Dikkere lagen verminderen het rendement en het overgedragen nettovermogen.



Figuur 2. De Qi-lader C310 van Pulesen is geschikt voor een vermogen van 5 W.



Figuur 3. Een mooie afgeronde vorm: De 10-W Qi-Lader RC200 van Yootech.



Figuur 4. Ook 10 W maar iets hoekiger en dikker: de RP-PC014 van Ravpower.

adapter nodig. Toen de lader bij mij aankwam (er zat ook een micro-USB-kabel bij) heb ik hem eerst op een normale iPhone-netadapter van Apple aangesloten die 5,1 V bij maximaal 1 A levert. De lader knipperde even met zijn LED en vertoonde verder geen activiteit. Ietwat nerveus legde ik mijn nieuwe iPhone op het laadoppervlak, want per slot van rekening was de iPhone ongeveer 100.000% duurder dan de lader. Maar alles ging goed: de iPhone deed ‘ping’ om het begin van het laden aan te geven en de LED van de Qi-lader lichtte kort op. Na een kleine dertig minuten was de slechts gedeeltelijk ontlaadden accu weer vol. Na deze eerste test werd de lader op een

externe USB-3.0-hub aangesloten die was voorzien van een eigen netadapter die makkelijk 5 A kan leveren. Als er iets mis zou gaan, zou nu in ieder geval niet mijn PC in rook opgaan, maar alleen deze aanzienlijke goedkopere hub. Ook de volgende dagen waren er geen problemen — alles functioneerde betrouwbaar. Steeds wanneer ik aan mijn bureau ging zitten legde ik de iPhone op de lader. Ik was tevreden.

### Meer vermogen

Deze tevredenheid duurde totdat Apple iOS 11.2 uitbracht. Hierdoor werd namelijk de snelle oplaadfunctie (met 7,5 W) bij de iPhone 8 en de iPhone X geactiveerd. Mijn nieuwsgierigheid was gewekt. Een korte zoektocht resulteerde in het bestellen van een RC200-lader (**figuur 3**). Volgens fabrikant Yootech moest deze naast 5 W en 10 W ook de 7,5 W van Apple kunnen leveren, en bovendien compatibel zijn met de iPhone. Voor dit alles vond ik € 14,99 niet teveel gevraagd.

Het apparaat arriveerde, mijn USB-hub had nog een poort over en de lader bleek zonder verrassingen met laden te beginnen. Maar toen werd het interessant: ik had niet echt het gevoel dat deze powerlader sneller klaar was dan het 5-W-model. En daar kwam nog een merkwaardig verschijnsel bij: na een paar dagen wilde een van de laders niet meer laden. “Nu al kapot,” dacht ik — en met een gevoel van berusting laadde ik verder met de andere lader. Tot deze ook niet meer wilde laden. Kort loshalen en weer inpluggen van de ene of de andere lader zorgde soms, maar niet altijd, voor verbetering. Alleen als ik beide laders van de 5 V afhaalde en ze er daarna weer op aansloot, werkten ze allebei weer. Tot na een paar dagen... Het werd tijd om me verder in deze zaak te verdiepen. Uit het internetonderzoek dat ik vervolgens deed blijkt dat er redenen zijn om aan te nemen dat de chips in deze laders niet alleen maar passief aan de 5-V USB-voeding hangen, maar ook iets met de dataverbindingen uithalen en elkaar hierbij zo nu en dan in de war brengen. In ieder geval crashte geen van beide laders als ze in hun eentje op de hub of op een netadapter waren aangesloten. Om het zeker te weten — dat had u waarschijnlijk al verwacht — bestelde ik een derde Qi-lader. Weer een power-exemplaar; deze keer kocht ik een RP-PC014 van Ravpower voor slechts € 10,92 (**figuur 4**). Volgens de fabrikant moet deze ‘Fast Qi wireless’-lader 10 W kunnen leveren aan de Galaxy S8 en S8+ van Samsung, en aan Apple-apparaten altijd nog 5 W (het was dus een wat ouder model). Ook deze lader werkte — maar het liefst alleen.

### Troubleshooting

Na deze ervaringen vond ik dat ik nu maar eens grondig moest uitzoeken wat hier aan de hand was. Ik wilde beginnen met het meten van het stroomverbruik. Om geknussel aan USB-kabels te vermijden maakte ik gebruik van een eenvoudige USB-adapter die naast de spanning ook het stroomverbruik weergeeft. Resultaat: alledrie de laders hadden een stroomverbruik van ergens tussen 0,8 en 0,95 A. Geen van de laders wilde dus meer dan een krappe 5 W aan mijn iPhone leveren. En als we rekening houden met het beperkte rendement van 80...90% van deze laders, dan moet het laden langzamer verlopen dan direct via een kabel. Dit kwam overeen met een vergelijking tussen inductief en gewoon laden op de website MacTechNews [2]. Laden met een kabel en een gewone 5-W-netadapter gaat dus sneller dan met een inductieve lader — ook in het geval van de 7,5-W-variant.

## Quick Charge

Quick Charge is een fabiekseigen standaard van Qualcomm uit het jaar 2013 voor het met hogere vermogens via USB laden van mobiele apparaten. Bij de eerste versie QC 1.0 ging het nog om 5-V-techniek met stromen tot 2 A. Hiermee werd het mogelijk om smartphones met 5 W via een geschikte netadapter of een hiervoor bestemde interface van een PC te laden ('gewone' USB-2.0-interfaces leveren maximaal 2,5 W).

Met QC 2.0 werd in 2015 het maximale vermogen verhoogd tot 18 W. Omdat bij nog hogere stromen de spanningsval over de USB-contacten en de relatief dunne USB-kabels zou toenemen heeft men de spanning verhoogd. Een met QC 2.0 compatibele laaduitgang kan 5, 9 of 12 V leveren bij stromen van 3, 2 of 1,66 A.

Het laadproces wordt altijd begonnen met 5 V zodat niet-compatibele apparaten geen schade oplopen. Een eventuele spanningsverhoging wordt dan volgens een bepaald protocol

tussen bron en gebruiker (smartphone, tablet of Qi-lader) ingesteld.

Een nauwkeuriger aanpassing kwam in 2016 met QC 3.0 [4], waarbij de geleverde spanning tussen 3,6 en 20 V in stappen van 200 mV kan worden ingesteld. Tenslotte werden in 2017 QC4 en 4+ gedefinieerd waarmee vermogens tot 27 W en nog nauwkeuriger instelbare spanningen mogelijk zijn.

Hogere QC-niveau's zijn achterwaarts compatibel. Als u tegenwoordig smartphones met meer dan 5 W wilt laden, dan hebt u minstens een QC-2.0-compatibel laadapparaat respectievelijk netadapter nodig. Moderne USB-netadapters van goede kwaliteit zijn voordelig in QC-3.0-uitvoering verkrijgbaar. Dit geldt ook voor de huidige smartphones van goede kwaliteit. Apple levert zoals altijd ook eigen producten omdat er in Apple-telefoons geen Qualcomm-chips voorkomen. Maar er zijn veel Qi-laadapparaten die ook met Apple's 7,5-W-modus kunnen omgaan.

Maar hier maakte ik een denkfout. Weliswaar kon mijn hub gemakkelijk 5 A aan, en had ik ook een 5-V-netadapter die meer dan 2 A kon leveren, maar de laders wilden gewoon geen gebruik maken van het beschikbare extra vermogen. Dat is ook geen wonder, want zoals verder lezen aan het licht bracht is voor het extra vermogen een andere standaard met de naam **Quick Charge** verantwoordelijk (zie het betreffende **kader**). Ook hiervan bestaan verschillende varianten en daarin worden andere spanningen dan 5 V gedefinieerd.

De iPhone is weliswaar compatibel met Qi, maar niet met Quick Charge, want Apple wilde zich kennelijk toch niet volledig op het gebied van de 'gewone' smartphones begeven. Maar gelukkig maakt dat niets uit want voor vermogens boven de 5 W hoeft alleen de Qi-lader compatibel te zijn met Quick Charge — tenminste versie 2.0. Zoals ik kon vaststellen voldeden mijn twee powerladers hier aan. En als u de rechterhelft van de figuren 3 en 4 bekijkt, kunt u vaststellen dat ze in plaats van 5 V graag 9 V willen hebben als ze meer vermogen moeten leveren. Met 5 V leveren ze maximaal 5 W.

Alles wat nu nog aan mijn geluk ontbrak was een geschikte netadapter die Quick-Charge-2.0-laders zou kunnen voeden. Ik koos voor de 18-W-netadapter uit **figuur 5**, die zelfs compatibel is met Quick Charge 3.0 (zoals in **figuur 6** op de voor- kant te zien is). Deze netadapter heeft voldoende reserve en hiermee zou niets meer fout moeten gaan. En dat deed het ook niet zoals later is gebleken.

### Stroom

Om er achter te komen of de Qi-laders met deze netadapter goed werkten, had ik mijn iPhone drie keer (of vaker) volledig kunnen opladen, en de tijd kunnen meten die de verschillende laders voor het opladen nodig zouden hebben. Dit vond ik niet alleen omslachtig, maar ook onprettig omdat het diep ontladen van lithium-ion-accu's, ondanks veiligheidsmaatregelen, bij iedere elektronicus een slecht gevoel veroorzaakt. Het was eenvoudiger om direct de opgenomen stroom te meten en het corresponderende vermogen te berekenen.

Hoe voor de hand liggend dit idee ook lijkt: het bevat een val-

kuil! Mijn kleine USB-stroommeter zou gegrild worden als ik die simpelweg tussen de QC-3.0-netadapter en de Qi-powerla- der had aangesloten, omdat het daar niet bij de gebruikelijke



Bild 5. Steckernetzteil nach QC 3.0 mit aufgedruckten technischen Daten.



Figuur 6. Op de voorkant van deze netadapter staat duidelijk leesbaar: 'Quick Charge 3.0'.



Figuur 7. De 5-W-Lader verbruikt maar 0,85 A bij 5 V.



Figuur 8. De powerlader van Yootech schakelt bij 9 V regelmatig over naar de 5-W-modus.



Figuur 9. De lader van Ravpower neemt bijna 7 W op.

5 V blijft maar met hogere spanningen rekening moet worden gehouden. Ik had dus een meetadapter nodig die meer spanning zou kunnen verdragen. Ook hiervoor bestaat er een kant-en-klare oplossing. Ik bestelde voor € 9,99 de 'USB-multimeter' J7 van Muker, die volgens opgave van de fabrikant compatibel is met QC 2.0 en 3.0. Naast spanning en stroom kan deze ook direct de tijd (s), de belastingsweerstand ( $\Omega$ ), het vermogen (W), de energie (Wh) en de lading (mAh) weergeven. Dat zou toch voldoende moeten zijn. In **figuur 7** kunt u het apparaatje in volle glorie zien, waarbij het tussen de QC-netadapter en de 5-W-lader van Pulesen is geschakeld. Deze laatste werkt hier op 4,96 V (gemeten op de uitgang van de meetadapter) en er loopt een stroom van 0,85 A. Kennelijk is er ergens een shunt

## Weblinks

- [1] Draadloze power-converter: [www.elektrormagazine.nl/160119](http://www.elektrormagazine.nl/160119)
- [2] Vergelijking tussen inductief en gewoon laden: <https://bit.ly/2vzoGs4>
- [3] Wireless Power Consortium: [www.wirelesspowerconsortium.com](http://www.wirelesspowerconsortium.com)
- [4] Specificaties van Quick Charge 3.0: [www.qualcomm.com/products/quick-charge-3](http://www.qualcomm.com/products/quick-charge-3)

waar spanning over valt, want in niet-actieve toestand stijgt de spanning tot 5,03 V. Interessant is dat de 5-W-lader slechts 4,22 W opneemt. Als we voor inductief laden uitgaan van een reëel rendement van 80% wordt de accu van de iPhone dus met slechts ongeveer 3,4 W geladen en dit komt overeen met de resultaten van MacTechNews.

Er is nog meer interessants te melden: de lader van Yootech schakelt ondanks de voedingsspanning van 9 V regelmatig over op de 5-W-modus. In **figuur 8** is te zien dat er bij 8,94 V slechts 0,46 A loopt en er dus maar 4,11 W wordt opgenomen. Er is dus geen garantie dat alle onderdelen elkaar goed begrijpen. Bij zo'n twijfelgeval kunt u dan minder dan 5 W krijgen terwijl er 7,5 of zelfs 10 W op staat. Alleen bij de lader van RAV-Power wijst alles in de goede richting, en dit zelfs regelmatig en ook continu. **figuur 9** laat zien dat er bij 8,9 V in elk geval 0,75 A loopt en de lader zichzelf een opgenomen vermogen van 6,68 W gunt. Hier bestaat dus een reële kans dat het inductief laden sneller gaat dan het via de kabel laden met de 5-W-netadapter.

## Conclusie

Wat in dit artikel is beschreven, zou ook voor Android-smartphones moeten gelden. In het bijzonder bij de voor 10 W geschikte duurdere modellen van Samsung kunnen we ervan uitgaan dat de laders tussen 8 en 9 W opnemen en dat hierdoor het laden sneller gaat dan met de bijgeleverde netadapter van de fabrikant. Samsung gaat Apple hier moeiteeloos voorbij... Wat voor mij meer telt dan alleen de laadsnelheid is, naast de kennis die ik over dit thema in de praktijk heb verkregen, ook het gemak dat het met zich meebrengt (of nog belangrijker: de ergonomie). Het is echt geweldig wanneer je niet met connectors aan de gang moet, maar je mobiele telefoon eenvoudig op het laadoppervlak kunt leggen om deze op te laden. En als de telefoon dan overgaat hoeft je niet eerst de kabel los te halen, maar kan je gewoon de telefoon oppakken en het gesprek aannemen. Daardoor is ook de zwakste Qi-lader nog steeds een stap vooruit wat betreft gebruiksvriendelijkheid. En zelfs een combinatie van QC-netadapter en powerlader is al voor iets meer dan € 20 verkrijgbaar. Deze luxe is dus ook nog erg goedkoop. ▶

(160662)

## IN DE ELEKTOR-SHOP



→ Design Kit Wireless Power Demo 5 W  
[www.elektor.nl/design-kit-wireless-power-demo-5w](http://www.elektor.nl/design-kit-wireless-power-demo-5w)



## Wat wilde u worden toen u jong was?

Dankzij uw ambitie heeft u toegang tot de razendsnel veranderende wereld van engineering, waarin uw ideeën de toekomst vormen. Maar om te doen wat u het beste doet, heeft u tijd nodig om zich te concentreren op wat echt belangrijk is.

Wij helpen klanten al 80 jaar hun doelen te bereiken met ons uitgebreide aanbod procesautomatiserings-, onderhouds- en elektronica-producten, en op u afgestemde services.

We're here

for the **inspired**

Deskundige technische ondersteuning | Toonaangevende merken | Levering de volgende werkdag





# Project 2.0

## correcties, updates en brieven van lezers

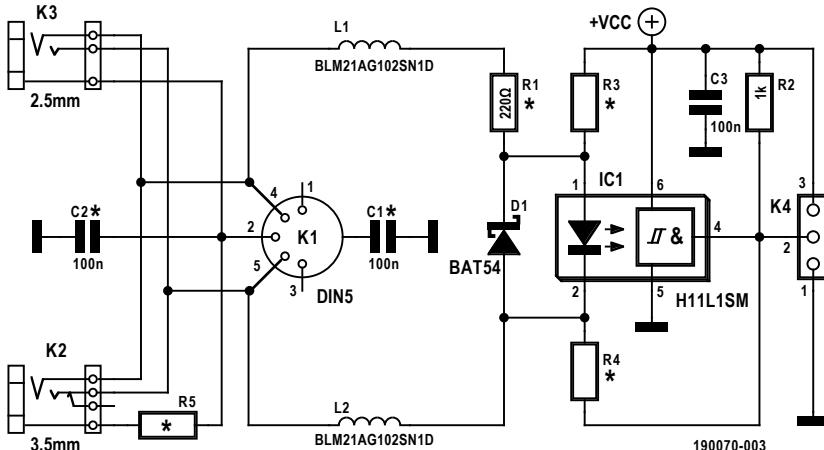
### MIDI I/O Break-out-Board

Elektor juli/augustus 2019, p. 26  
(190070)

In het schema is er bij de aansluitingen van de DIN-socket helaas wat mis gegaan. Juist is: tip DIN pin 5, ring DIN pin 4, shield DIN pin 2.

Ook is het typenummer van IC1 onjuist; dit moet zijn H11L1SM. Hiernaast ziet u het gecorrigeerde schema.

De printjes voor het Break-out-Board zijn correct.



### Luchtvervuilingsmonitor

Elektor maart/april 2019, p. 90 (170182)

In de download 170182-11.zip ([www.elektormagazine.nl/170182-02](http://www.elektormagazine.nl/170182-02)) bevinden zich meerdere sketches, een datemap en een PDF. Ik weet echter niet goed wat ik daarmee aan moet. Kunt u me een beetje op weg helpen? Ik heb Arduino 1.8.9 met de ESP32-libraries geïnstalleerd, alsmede de drivers voor de seriële interface. Ook voor de *ESP32 Sketch Data Upload* is alles voorbereid.

Frank van Elk

Bedankt voor uw feedback. We hebben de download inmiddels wat overzichtelijker gemaakt; deze staat op de projectpagina klaar.

Pak de map uit in de sketch-map van de Arduino IDE (onder

Windows 10 is dat doorgaans [...]\\Documents\\Arduino\\sketchbook).

Daarna moet die map er ongeveer

zo uit zien: [...]\\Documents\\Arduino\\

sketchbook\\tgs2600\\. De Arduino IDE

moet opnieuw worden gestart, anders

zal hij de sketch niet vinden. Nu kunt u

de sketch openen en deze in de ESP32

laden.

Na de installatie van de SPIFFS-uploa-

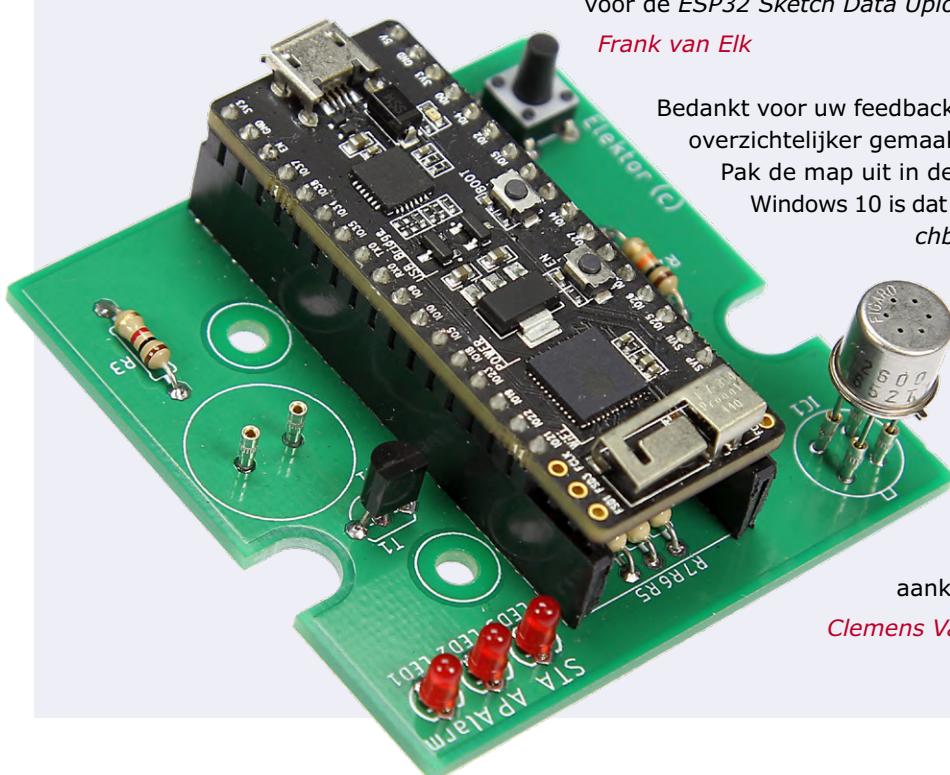
der, de herstart van de Arduino IDE

en het openen van de TGS2600-sketch

kunt u de optie *ESP Sketch Data Upload*

aanklikken.

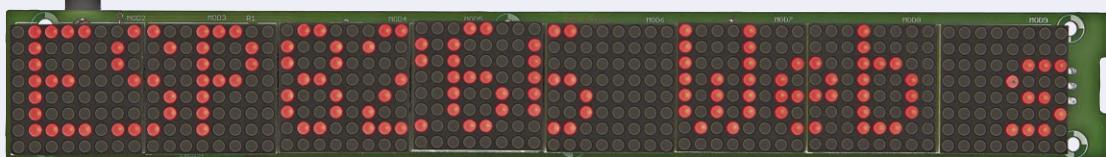
Clemens Valens (Elektor Labs)





## Lichtkrant met ESP-12F

Elektor mei/juni 2018, p. 90 (160491)



Voor dit project ([www.elektormagazine.nl/labs/scrolling-text-display-160491](http://www.elektormagazine.nl/labs/scrolling-text-display-160491)) heb ik een update gepost waarmee de mogelijkheden van het oorspronkelijke project worden vergroot en die op een ESP32 is gebaseerd.

Peter Tschulik

## ESP32 Arduino-library 1.0.2

Wie met de huidige versie van de ESP32 Arduino Board-software (versie 1.0.2 in de Board Manager) werkt, kan last krijgen van een vervelende bug. Wanneer in de code naar AsyncUDP wordt verwezen, kan dat tot een fout in de pakketbuffers leiden zodat de chip een exception genereert, wat op zijn beurt een herstart veroorzaakt.

Bibliotheken die deze functie gebruiken zijn bijvoorbeeld de *NtpClientLib* van <http://github.com/gmag11/ntpClient>. Wie met dit probleem wordt geconfronteerd, kan uitwijken naar versie 1.0.1 of versie 1.0.3.-rc1 in de Board Manager.

Mathias Claußen (Elektor Labs)



## Zelfbouw-soldeerstation met temperatuurregeling

Elektor januari/februari 2019, p. 14 (180348)

De nieuwe versie van het solderstation in het januari/februari-nummer is een uitstekend project. Ik had zijn Platino-voorganger al nagebouwd en gebruik die met veel plezier. Zijn opvolger is een flink stuk kleiner en beschikt over extra features. Hoewel – niet alles klopt wat in het artikel beschreven wordt. De uitvoer van de gewenste en werkelijke temperatuurwaarden alsmede de complete status van het solderstation via de USB-interface is helaas niet in de firmware geïmplementeerd. Een snelle blik op de Arduino-sketch leert om te beginnen dat de initialisering van de seriële interface ontbreekt en dat de waarden ook niet naar de interface worden gestuurd. Staat een uitbreiding van de firmware met deze functionaliteit op de planning?

Hans Schneider



In alle sinds 15 maart 2019 uitgeleverde solderstations is de update voor de seriële interface opgenomen (firmware vanaf versie 1.2).

Wanneer u het solderstation op uw computer aansluit, kunt u met de terminal van de Arduino IDE of een ander serieel terminalprogramma verbinding maken met de seriële interface. Wanneer daarbij wordt gevraagd een baudrate in te stellen, kies dan 115200. Omdat de communicatie via USB verloopt, is de baudrate voor het apparaat eigenlijk onbelangrijk. Belangrijk: gebruik geen 1200 baud, omdat hierdoor de bootloader gestart kan worden. Dit zijn de door de console ondersteunde commando's:

- `set/get set/get setpoint[xxx]` Hiermee wordt de gewenste waarde ingesteld of opgevraagd.
- `get temperature` Hiermee wordt de actuele temperatuur opgevraagd.
- `clear error` Wanneer een foutmelding wordt getoond, kunt u die hiermee wissen.
- `help` Hiermee wordt help-info getoond.

Tot slot nog een algemene tip: op Github (<https://github.com/ElektorLabs/>) bevinden zich het firmware-log en de softwareversies voor diverse Elektorlabs-projecten. Wanneer u een lab-project nabouwt, loont het de moeite daar een blik op te werpen. Voor het solderstation luidt de weblink <https://github.com/ElektorLabs/180348-DIY-Soldering-Station>.

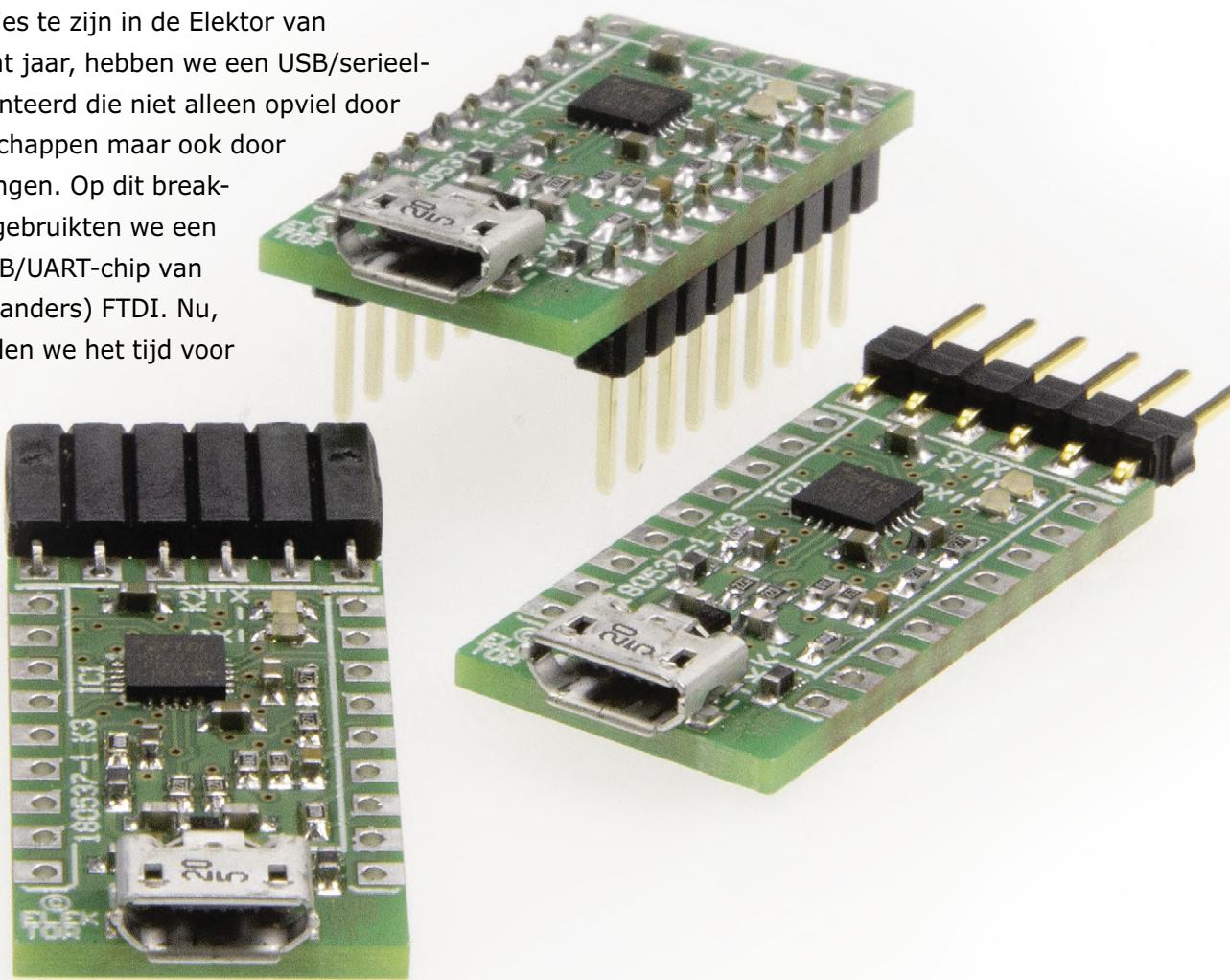
Mathias Claußen (Elektor Labs)

# USB/RS232 break-out-board

## zeg maar BoB

Ton Giesberts (Elektor Labs)

In 2011, om precies te zijn in de Elektor van september van dat jaar, hebben we een USB/serieel-omzetter gepresenteerd die niet alleen opviel door zijn goede eigenschappen maar ook door zijn kleine afmetingen. Op dit break-out-board (BoB) gebruikten we een FT232RQ, een USB/UART-chip van (hoe kan het ook anders) FTDI. Nu, na acht jaar, vonden we het tijd voor een update, die we u hieronder presenteren.



### Eigenschappen

- micro-USB-connector
- volledig USB 2.0-compatibel
- VCCIO +1,8...+3,3 V (absoluut maximum 4 V)
- geregelde 3,3-V-uitgang (50 mA max.)
- overdracht 300 baud... 3 Mbaud
- compatibel met RS232, RS485 en RS422
- full-handshake USB-naar-UART
- 4 configurerbare I/O-pennen

Ten opzichte van die eerdere versie [1] zijn er een paar veranderingen doorgevoerd. In de eerste plaats hebben we nu een goedkopere versie van de USD/UART-converter gebruikt: de FT231XQ. Ook is het bij deze versie mogelijk normale pinheaders te gebruiken, deze staan nu dichter bij elkaar. Dat heeft tot gevolg dat het nieuwe BoB met gemonteerde connectoren iets smaller is dan de oude versie; wel is de print iets langer geworden (29,2 mm in plaats van 27 mm). Met gemonteerde micro-USB-connector en een haakse header aan de tegenover-

liggende zijde (de RS232-aansluiting) komen we op een totale lengte van net geen 4 cm.

Ja, u hebt het goed gelezen: we gebruiken nu een micro-USB-connector in plaats van een mini-exemplaar omdat die inmiddels veel gebruikelijker zijn geworden. Daarnaast hebben we de volgorde van de signalen op de RS232-connector veranderd en daar het signaal RTS aan toegevoegd, dit om compatibel te zijn met de kant-en-klare USB/RS232-kabels van FTDI. En tenslotte hebben we nog

een ESD-bescherming in de vorm van een speciale suppressor toegevoegd.

Wie geïnteresseerd is in wat er zich in de kleine behuizing van de FTDI-chip bevindt, kan een blik werpen op de datasheet [2] en de bijbehorende technical note TN140 met errata [3].

### Het schema

Tijd om een blik op het schema van **figuur 1** te werpen. Dat ziet er misschien intimiderend uit, maar dat valt bij nader inzien reuze mee. Het hart wordt uiteraard gevormd door IC1, de eigenlijke omzetter-chip. De aansluitingen van dat IC worden op een viertal connectoren naar buiten gevoerd. K1 is de micro-USB-connector en K2 de RS232-aansluiting. De overige aansluitingen van het IC gaan naar de beide andere headers, waarbij K3 alle UART-signalen toegankelijk maakt en op K4 vier configurerbare I/O's van IC1 te vinden zijn (alsmede de voedingsspanningen).

De beide LED's geven bedrijvigheid op de datalijnen aan. Deze LED's zijn aangesloten op CBUS1 en CBUS2. Van huis uit zijn deze I/O-lijnen zo ingesteld dat het verzenden (LED1/TX) of het ontvangen (LED2/RX) via de USB-aansluiting wordt aangegeven. Het is echter ook mogelijk de chip zo te configureren dat het zenden en ontvangen van RS232-data wordt getoond. (Terzijde: op het kleine printje

was naast LED D1 geen plaats meer voor de opdruk 'TX', die staat daarom iets linksboven de LED. Deze opdruk heeft dus niets met connector K2 te maken!)

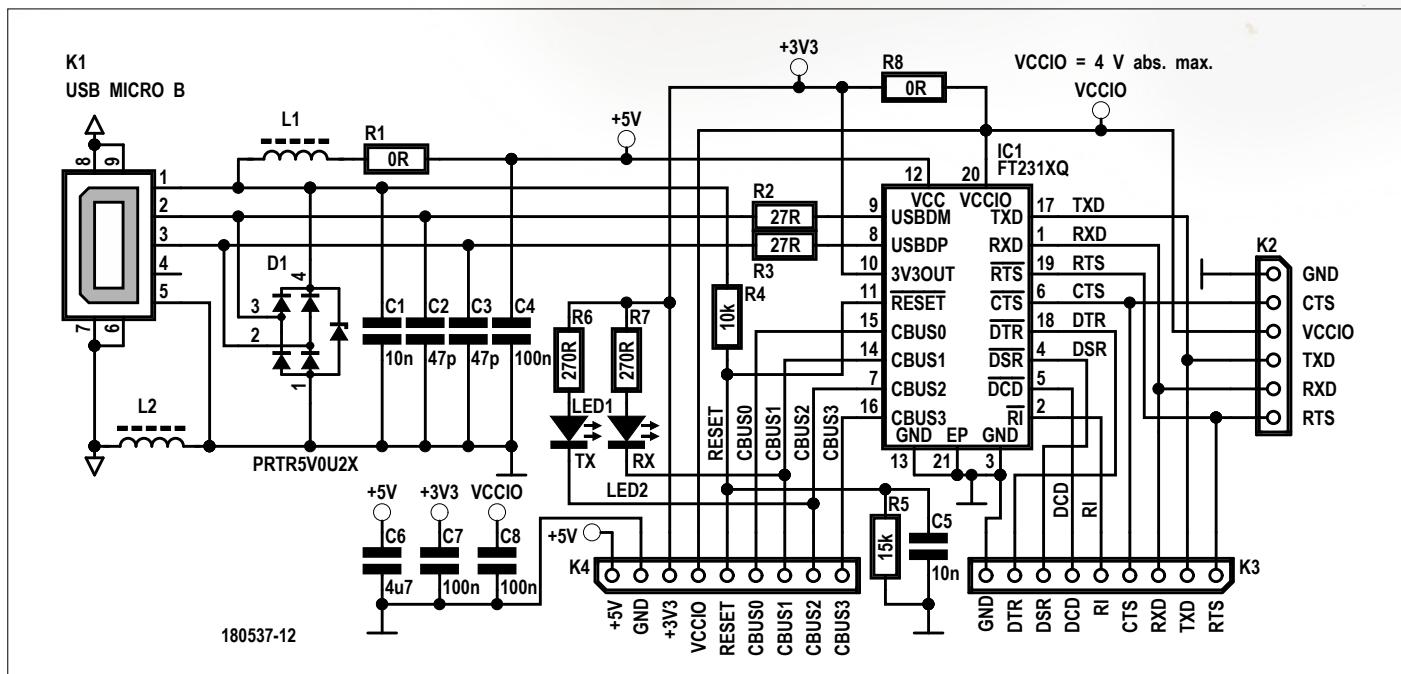
Een laatste onderdeel dat het vermelden waard is, is D1: een ESD-suppressor die de schakeling tegen elektrostatische ontlading beschermt. Het gekozen type munt uit door een uiterst geringe capaciteit.

Op het printje komen twee 0-Ω-weerstanden voor (R1 en R8). Deze hebben met de voeding te maken, en daar gaan we nu op in.

### Voeding

Hier valt wel het een en ander over te vertellen, vooral ook omdat vergissingen hier dodelijk kunnen zijn voor onze kleine BoB.

In de eerste plaats kan de chip via de USB-aansluiting (pin 1) worden gevoed. Ons BoB'je betreft dan +5 V van het apparaat waar het break-out-board via USB op is aangesloten. In dat geval moet R1 gemonteerd zijn.



Het spreekt voor zich dat in dat geval geen externe spanning van 5 V op pen 1 van K4 mag worden aangesloten.

Omgekeerd is het natuurlijk ook mogelijk de schakeling met een separate 5-V-spanning te voeden; die wordt dan op pen 1 van K4 aangesloten. Natuurlijk moet in dat geval R1 worden verwijderd. In dat geval neemt de schakeling vrijwel geen stroom op van het aangesloten apparaat – slechts 0,2 mA via spanningsdeler R4/R5; deze dient om te detecteren of een (ingeschakeld) apparaat op K1 is aangesloten of niet. Pas alstublieft wel op: pen 1 van K4 is direct met de chip verbonden!

Dan komen we bij VCCIO: dit is de voeding voor de I/O-cellens in de chip. Ook hier hebben we twee mogelijkheden. Wanneer 0- $\Omega$ -weerstand R8 is gemonteerd, is de VCCIO-pen (pen 20) verbonden met de interne 3,3-V-regulator in de chip (pen 10). Deze kan maximaal 50 mA leveren; deze spanning is extern beschikbaar op pen 3 van K4.

Wanneer R8 niet is gemonteerd, krijgt VCCIO zijn spanning via pen 4 van K4. Let hier heel goed op: VCCIO is gespecificeerd voor een spanning van +1,8...+3,3 V bij een absoluut maximum

van +4 V. Hogere spanningen blazen de chip op!

Derde: de RS232-aansluitingen zijn 5-V-resistant; externe UART-logica die op 5 V draait, kan probleemloos worden aangesloten.

### Configureren

Zoals we hierboven al zijdelings opmerkten, is het mogelijk de vier I/O-pennen van IC1 (CBUS1...CBUS4), die op K4 naar buiten zijn gevoerd, naar eigen behoeften te configureren. De maker van de chip stelt daarvoor het gratis tool FT2PROG [4] ter beschikking. Van dezelfde locatie kan ook een zeer uitgebreide (Engels-talige) handleiding voor dit tool worden gedownload, zodat we er hier niet nader op in gaan, temeer omdat de schakeling in de standaardconfiguratie precies doet wat ze moet doen. Een woord van waarschuwing: pas op wanneer u met het softwaretool gaat experimenteren: als u teveel rondmoddert is het mogelijk dat de chip helemaal nergens meer op reageert...

### Bouw

Voor de schakeling hebben we natuurlijk een (dubbelzijdig) printje ontworpen dat u in onze shop kunt bestellen – zie

 Zelfbouw eist een vaste hand!

**figuur 2.** Wat meteen zal opvallen is het grote aantal via's – bij zo'n klein printje met vier connectoren niet echt verwonderlijk.

Wie geen zin heeft in gepriegel met klein SMD-grut, kan in onze shop een gedeeltelijk bestukte print kopen — daar zijn alle ‘moeilijke’ onderdelen al op aangebracht. De verstokte doe-het-zelver heeft naast geschikt SMD-gereedschap (een hetelucht-soldeerbout, onder andere) vooral een vaste hand nodig.

Om goed te zien waar welke onderdelen thuisoren (om plaatsredenen hebben we de onderdelennummers binnen de omtrek van die onderdelen moeten zetten) verdient het aanbeveling een vergrote versie van de componentenopdruk te gebruiken. Die kunt u van de projectpagina bij dit artikel [5] downloaden. Daar is overigens ook duidelijk aangegeven welke signalen op welke pennen van de diverse connectoren te vinden zijn.



## ONDERDELENLIJST

### Weerstanden (alle SMD 0603):

R1,R8 = 0  $\Omega$ , 75 V/100 mW  
R2,R3 = 27  $\Omega$ , 75 V/100 mW  
R4 = 10 k, 50 V/100 mW  
R5 = 15 k, 50 V/100 mW  
R6,R7 = 270  $\Omega$ , 50 V/100 mW

### Condensatoren (alle SMD 0603):

C1,C5 = 10 n, 50 V, 20%, X7R  
C2,C3 = 47 p, 50 V, 2 %, C0G/NPO  
C4,C7,C8 = 100 n, 50 V, 10 %, X7R  
C6 = 4,7  $\mu$ , 6,3 V, 10 %, X5R

### Spoelen:

L1,L2 = 330  $\Omega$  @ 100 MHz, 1,7 A, 0,08  $\Omega$  (Murata BLM18KG331SN1D)

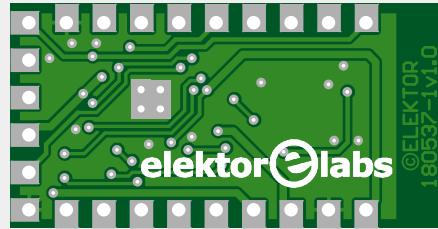
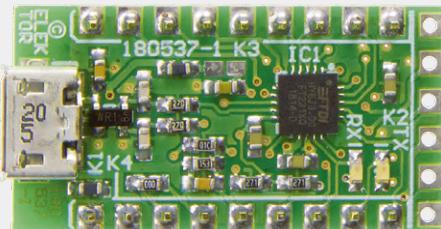
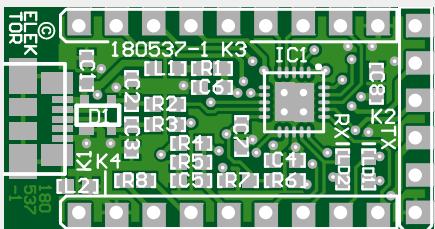
### Halfgeleiders:

D1 = PRTR5V0U2X, ESD-suppressor, SMD SOT-143B  
LED1 (TX) = HSMG-C190, groen, SMD 0603  
LED2 (RX) = HSMS-C190, rood, SMD 0603  
IC1 = FT231XQ, SMD QFN-20

### Diversen:

K1 = micro-USB type B female, SMD-printmontage  
K2 = 6-polige 1-rijige pinheader, raster 2,54 mm, haaks  
K3,K4 = 9-polige, 1-rijige pinheader, raster 2,54 mm, recht

Print 180537-1 v1.0



Figuur 2. Voor wie het gepriegel met SMD-onderdelen niet ziet zitten, is een gedeeltelijk bestukte print in onze shop verkrijgbaar.

Pas op bij de montage van de micro-USB-connector K1 en ESD-suppressordiode D1: soldeer eerst K1 en pas dan D1. Reden: wanneer D1 al is gemonteerd, is het vrijwel onmogelijk om eventuele kortsluitingen tussen de aansluitingen van K1 nog te repareren.

### Tot besluit

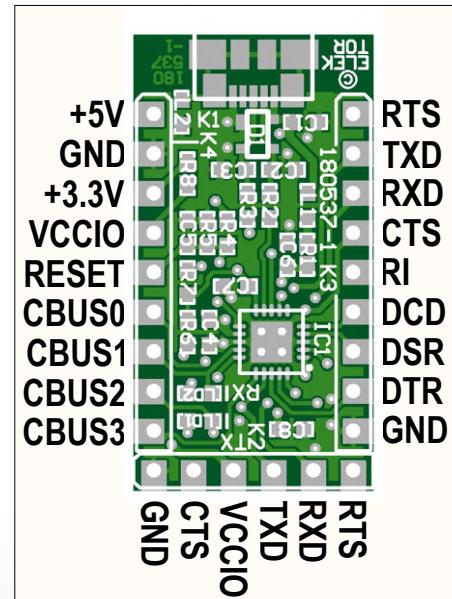
De simpelste RS232-verbinding tussen twee apparaten is de nul-modem-verbinding (ook wel kruislingse verbinding genoemd). Daarbij wordt TxD van het ene apparaat verbonden met RxD van het andere (en vice versa, en natuurlijk worden de massa's van beide apparaten verbonden). Naast 5...8 databits worden bij de communicatie een startbit, maximaal twee stopbits en eventueel een pariteitsbit gebruikt.

Een veiliger methode om data over te dragen is het gebruik van een buffer om de data tijdelijk op te slaan. Handshaking (hardware- of softwarematig) voorkomt dat de buffer 'overloopt'. Bij

signalen DTR, DSR, RTS en CTS gebruikt. CTS wordt kruislings verbonden met RTS, terwijl DTR kruislings met DSR wordt verbonden. In het geval dat een van de apparaten een modem is, worden ook DCD en RI gebruikt (DCD aan DCD en RI aan RI).

Bij software-handshaking worden twee speciale karakters gebruikt voor start en stop: XON en XOFF (decimaal 17 resp. 19). Dat wordt software flow control genoemd. Deze methode heeft echter een nadeel: bij de overdracht van binaire data mogen deze karakters niet voorkomen, anders gaat het mis.

Nadere informatie over RS232 is onder andere op [6] en [7] te vinden.



hardware-handschaking worden voor de besturing nog de

U kunt uw BoB na het opbouwen (en na een uiterst nauw-

kan de verbinding in HTerm worden gecontroleerd: groene indicatierrondjes geven aan dat DTR en/ of RTS aan de andere kant actief zijn. ▶

(180537-B-01)

gezette controle van het soldeerwerk!) testen met behulp van een terminalprogramma zoals HTerm. De eenvoudigste methode is natuurlijk het gebruik van twee computers en twee BoB's. Bij full handshaking

▶

**IN DE STORE**

- Kale print [www.elektor.nl/18878](http://www.elektor.nl/18878)
- Bestukte print   
(bijgeleverde headers zelf monteren) [www.elektor.nl/18895](http://www.elektor.nl/18895)

### Weblinks

- [1] USB-FT232R breakout-board: [www.elektrormagazine.nl/magazine/elektor-201109/16422](http://www.elektrormagazine.nl/magazine/elektor-201109/16422)
- [2] Datasheet FT232XQ: [www.ftdichip.com/Support/Documents/DataSheets/ICs/DS\\_FT231X.pdf](http://www.ftdichip.com/Support/Documents/DataSheets/ICs/DS_FT231X.pdf)
- [3] TN140: [www.ftdichip.com/Support/Documents/TechnicalNotes/TN\\_140\\_FT231X%20Errata%20Technical%20Note.pdf](http://www.ftdichip.com/Support/Documents/TechnicalNotes/TN_140_FT231X%20Errata%20Technical%20Note.pdf)
- [4] FTPROG-tool: [www.ftdichip.com/Support/Utilities.htm#FT\\_PROG](http://www.ftdichip.com/Support/Utilities.htm#FT_PROG)
- [5] Projectpagina bij dit artikel: [www.elektrormagazine.nl/180537-01](http://www.elektrormagazine.nl/180537-01)
- [6] Meer over RS232: [www.commfront.com/pages/3-easy-steps-to-understand-and-control-your-rs232-devices](http://www.commfront.com/pages/3-easy-steps-to-understand-and-control-your-rs232-devices)
- [7] Nog meer over RS232: [www.codrey.com/embedded-systems/rs232-serial-communication/](http://www.codrey.com/embedded-systems/rs232-serial-communication/)
- [8] FTDI-driver: [www.ftdichip.com/Drivers/VCP.htm](http://www.ftdichip.com/Drivers/VCP.htm)
- [9] FTDI-driver: [www.ftdichip.com/Drivers/D2XX.htm](http://www.ftdichip.com/Drivers/D2XX.htm)
- [10] Elektor Labs-pagina: [www.elektrormagazine.com/labs/remake-110553-usb-rs232-converter-ft232-bo](http://www.elektrormagazine.com/labs/remake-110553-usb-rs232-converter-ft232-bo)

www.elektrormagazine.nl september/oktober 2019 79



# HomeLab Helicopter

de wondere wereld van de elektronica in vogelvlucht

samengesteld door Clemens Valens (Elektor Labs)

## VAN IDEE TOT PRODUCT DEEL 1

Soms vragen mensen ons: *hoe maak je een "product"*? Voordat we kunnen ingaan op dit interessante maar gecompliceerde onderwerp, moet er eerst een andere vraag worden beantwoord: *waarom ontwikkelt u een product?* Het heeft geen zin om een product te ontwikkelen zonder reden. Zonder een realistisch doel is het meestal voldoende om een prototype in elkaar te zetten. Een product komt pas echt tot leven als het klaar is om te worden verkocht. Dat betekent dat bij het ontwikkelen van een product ook moet worden gezorgd voor één of meer verkoopkanalen, zonder details als verpakking, verzending, factureren, marketing en aftersales over het hoofd te zien. Een verkoopkanaal kan van

alles zijn, van verkopen op straat, via eBay of een eigen online webwinkel tot zorgen dat het in de schappen van een landelijke supermarktketen komt te staan of dat het verkocht wordt via Amazon en/of Alibaba. Maar voordat een product kan worden verkocht, moet het worden geproduceerd, getest en gecertificeerd. En er moet een gebruiksaanwijzing voor worden geschreven. De productie kan pas beginnen nadat er een volledig werkend en gevalideerd prototype is gebouwd dat is ontworpen met (massa) productie in gedachten. Het moet zijn *geindustrialiseerd*. U ziet wel, dat er bij het maken van een product veel meer komt kijken dan alleen een goed idee.

### HET BEGINT MET EEN GOED IDEE

U moet een idee voor een product (of dienst, maar we beperken ons hier even tot tastbare zaken) bedenken dat een voldoende groot publiek weet te verleiden. Dat kan natuurlijk een geheel nieuw product zijn, maar het



kan ook iets

zijn dat u al hebt ontwikkeld voor een ander doel. Veel projecten die voor een specifiek doel zijn ontwikkeld, kunnen gemakkelijk worden aangepast voor een veel breder of geheel ander toepassingsgebied. Dat maakt ze tot goede kandidaten voor het ontwikkelen van een product.



## STEL EEN BEDRIJFSPLAN OP

Als u eenmaal een interessant idee voor een product hebt, en u bent een ontwerper in hart en nieren, dan komt u gemakkelijk in de verleiding om maar meteen met de ontwikkeling te beginnen. Loop niet te hard van stape! Begin met een objectief marktonderzoek. U kunt zelf wel denken dat iets een geweldig idee is, maar wat vinden anderen er van? En hoeveel anderen hebben hetzelfde idee gehad? Bestaat het product al of niet? En als het niet bestaat, is dat dan omdat uw idee echt origineel is, of omdat alle anderen met hetzelfde idee de moed hebben opgegeven toen ze ontdekten dat er geen markt voor was? En als die markt bestaat, is die groot genoeg voor eventuele concurrenten? En wat hebt u nodig om uw product te ontwikkelen? Hoeveel tijd, geld, gereedschappen, mensen en andere middelen? Hoe gaat u uw product in de markt zetten? Kortom: u moet eerst een bedrijfsplan schrijven. En dat is niet zo gemakkelijk. Denk niet dat iedereen de wereld met dezelfde ogen beziet als u. En probeer een plan te ontwikkelen voor meerdere jaren. Waar wilt u bijvoorbeeld over vijf jaar staan? En hoe denkt u daar te komen? Er zijn op internet heel veel voorbeelden en templates voor bedrijfsplannen te vinden; die kunnen u goed op weg helpen. Ook de Kamer van Koophandel kan hulp bieden.



## WILT U ECHT EEN MANAGER ZIJN?

Stel dat u erin bent geslaagd om een bedrijfsplan op te stellen dat de toets der kritiek kan doorstaan. Dan is de volgende stap om het daadwerkelijk uit te voeren. Bent u bereid om ontslag te nemen, al uw spaargeld te investeren en in het ergste geval een echtscheiding te riskeren? Of volstaat het om al uw vrije avonden en de meeste weekends te besteden aan het project? Houd er ook rekening mee dat als uw bedrijf levensvatbaar wordt, uw takenpakket weleens zou kunnen veranderen: u gaat leiding geven, klanten binnenhalen, zoeken naar financiering en al die andere 'zaken' die zakenmensen doen. Het ontwerpen dat uw ziel en zaligheid is, moet u nu overlaten aan anderen. Is dat wat u wilt? En zo ja, denkt u dat u daar goed genoeg in zult zijn? Misschien hebt u wel één of meer partners nodig om u daarbij te helpen. Uw vrienden vragen of ze partner willen worden is een prima manier om de vriendschap kapot te maken, dus misschien moet u ze ergens anders zoeken. Vermijd als het even kan een samenwerking op 50/50-basis. Het is beter om zelf de meerderheid in handen te houden. Anders loopt u nog het risico dat u uit uw eigen bedrijf wordt geschoppt...



### Ideas

Any thoughts, opinions, creation, suggestions or conception that is existing in the mind as to a possible cause of action.

### Function

An action or activity proper to a person or thing the purpose for something which is designed for.

## WORDT VERVOLGD...

Als u na het lezen van dit artikel nog steeds gemotiveerd genoeg bent om een product te ontwikkelen, bereid u dan voor op de volgende afleveringen, waarin we u laten zien hoe u een idee kunt omzetten in een product dat ook werkelijk te produceren is. We zullen het hebben over elektronica-producten, want dat is waar het bij Elektor om draait, maar sommige aspecten zullen zeker ook van toepassing zijn op andere producten. Maar als u het gevoel hebt dat het ondernemerschap niets voor u is, zet dan lekker de soldeerbout aan en ga meteen aan de slag met uw idee. U weet immers nooit waar het toe kan leiden.



Alias gemonteerd bovenop de slimme assistent  
(foto: Bjørn Karmann).

## Uw slimme assistent te slim af

Persoonlijke of slimme assistenten zoals Google Home en Amazon Alexa luisteren permanent naar de geluiden uit de omgeving en reageren op sleutelwoorden en commando's. Omdat ze zijn verbonden met het internet, vormen ze een enorm risico voor de privacy. Als u zich daar zorgen over maakt, moet u ze niet installeren in uw huis en kunt u 'helaas' niet profiteren van de diensten die ze te bieden hebben. Zou het niet mooi zijn als u ze wat minder gevaarlijk zou kunnen maken? Dat is precies wat Project Alias van de Deense ontwerper Bjørn Karmann doet. Het is geïnspireerd door *cordyceps*-schimmels en virussen die de besturing van hun gastheer overnemen. Alias is een apparaat dat de oren van de slimme assistent vervangt door eigen oren. Hij doet dat door de microfoon van de assistent te storen met een constante ruis. Zelf luistert hij alleen naar zijn eigen trigger-woorden en hij is niet verbonden met het Internet. Als hij een geldige trigger detecteert, schakelt hij de ruis uit en geeft het commando door aan de slimme assistent, die dan gewoon zijn werk kan doen. Alias kan worden geprogrammeerd om allerhande gesproken commando's naar de slimme assistent te sturen, wat veel nieuwe mogelijkheden biedt. De broncode en bouwinstructies zijn te vinden op GitHub.

[http://bjoernkarmann.dk/project\\_alias](http://bjoernkarmann.dk/project_alias)

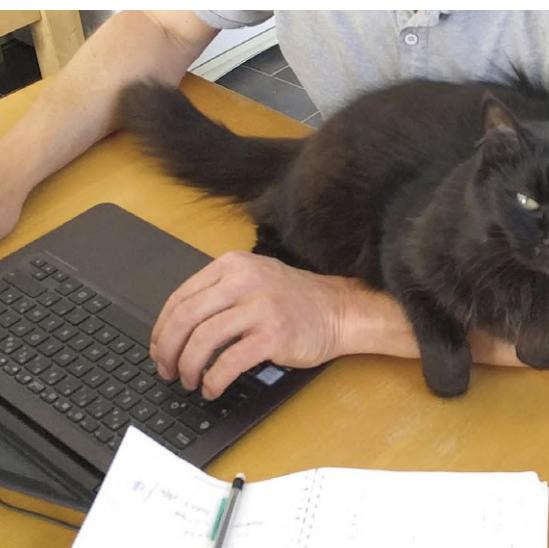
## LoRa zoekt het hogerop

Na de lancering van een satelliet van het Satish Dhawan Space Centre in India in april 2019 heeft Lacuna Space de eerste testfase van zijn LoRa Space Gateway volbracht. Testsystemen overal ter wereld hebben met goed gevolg gecommuniceerd met de LoRa Space Gateway. In de tweede helft van 2019 worden nog drie satellieten gelanceerd om het demonstratiennetwerk compleet te maken.

De LoRa Space Gateway kreeg een plaatsje op een 6U cubesat satelliet (foto: Nano Avionics).



## Homelab-hebbedier



Het is bekend dat katten stress- en angstverschijnselen bij mensen kunnen verminderen. Ze helpen de bloeddruk en de hartslag te verlagen, en dat vermindert weer de kans op hartaandoeningen. Dit is wetenschappelijk aangetoond. Uit onderzoek uit 2008 is zelfs gebleken dat mensen zonder katten 30% meer kans hebben op een hartaanval dan kittenbezitters. In een poging om de stress bij werknemers te reduceren en hun productiviteit te verbeteren, heeft een Japans IT-bedrijf een aantal katten geadopteerd en de medewerkers aangemoedigd om voor de dieren te zorgen. Ze krijgen zelfs een financiële tegemoetkoming als ze thuis ook een kat adopteren. Een ander interessant voorbeeld is een kantoorgebouw in Las Cruces in New Mexico (Verenigde Staten), waar een een soort kitten-o-theek is opgezet waar mensen (jonge) katten kunnen lenen om zich mee te ontspannen. Ik heb altijd al katten gehad en ik kan hun rustgevende kwaliteiten alleen maar bevestigen; daarom beveel ik ze aan voor elk laboratorium of kantoor. Het enige probleem is kittenpis. Die stinkt niet alleen afschuwelijk, maar is ook erg corrosief en maakt PCB's binnen de korste keren onbruikbaar. Natuurlijk bestaan er ook echte kattenhaters. Of ook bij hen positieve effecten te verwachten zijn, is zeer de vraag.

**Japanse IT-katten:** <https://youtu.be/vd21SH6zAX4>  
**Kitten-o-theek:** <https://youtu.be/OKq22oF4FiQ>



## SHOP-HIGHLIGHT

# De nieuwe Raspberry.. ...en meer lekkers uit onze shop

Of u nu het liefste programmeert, naar de radio luistert of op jacht gaat naar straling, wij hebben de juiste uitrusting voor u bijeengebracht.

## Raspberry Pi 4 – in drie uitvoeringen

Eindelijk is hij er: de nieuwe Raspberry Pi 4. De laatste hardware-updates brachten slechts kleine, zeer bescheiden verbeteringen, soms meer een facelift dan een substantiële verandering, maar de RPi4 heeft echt wat nieuws te bieden.

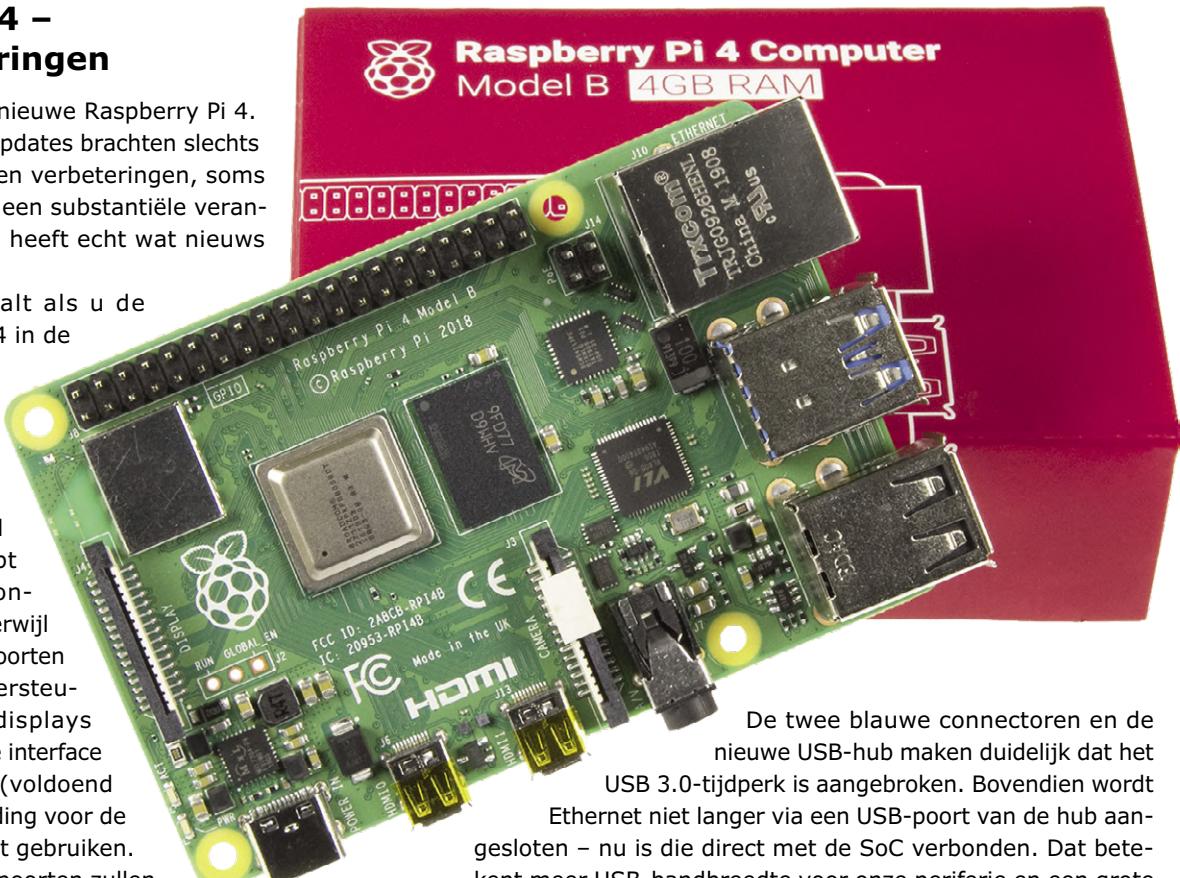
Het eerste wat opvalt als u de nieuwe Raspberry Pi 4 in de hand neemt, is dat die niet in bestaande behuizingen past.

Dat komt doordat de display- en voedings-aansluiting veranderd zijn. De voeding loopt nu via een USB-C connector en een PMIC, terwijl er twee micro-HDMI poorten zijn die high-res ondersteuning voor twee 4K-displays tegelijk via een digitale interface bieden, terwijl u een (voldoend krachtige) USB-C voeding voor de stroomverzorging kunt gebruiken.

De twee blauwe USB-poorten zullen u zeker opvallen. Inderdaad, we hebben eindelijk USB 3.0 aan boord, alsmede native Ethernet, waarmee problemen met de data-overdracht van een aangesloten massageheugen naar het LAN tot het verleden behoren. Dat danken we aan de nieuwe BMC2711 SoC die het hart van de RPi 4 vormt, met vier Cortex-A72 cores op maximaal 1,5 GHz en met ondersteuning voor maximaal 4 GB RAM.

Enkele van de vertrouwde features die de Raspberry Pi 4 heeft behouden, zijn de RCA-poort, de display-connector en de camera-interface. Ook onveranderd en wat de functionaliteit betreft achterwaarts compatibel is de 40-polige PCB-header. Dat betekent dat we een HAT op de vertrouwde manier kunnen aansluiten (tenminste wat de hardware betreft).

Omdat we nu over twee micro-HDMI poorten beschikken, kunnen we twee full HD-schermen op de Raspberry Pi 4 aansluiten. En eindelijk hebben we nu de beschikking over echte USB-poorten en niet slechts USB 2.0 met een trage uplink naar de SoC.



De twee blauwe connectoren en de nieuwe USB-hub maken duidelijk dat het USB 3.0-tijdperk is aangebroken. Bovendien wordt Ethernet niet langer via een USB-poort van de hub aangesloten – nu is die direct met de SoC verbonden. Dat betekent meer USB-bandbreedte voor onze periferie en een grote verbetering in de netwerk-doorvoersnelheid.

## IN DE STORE



→ [Raspberry Pi 4 met 1 GB RAM](http://www.elektor.nl/18966)

[www.elektor.nl/18965](http://www.elektor.nl/18965)

→ [Raspberry Pi 4 met 2 GB RAM](http://www.elektor.nl/18964)

[www.elektor.nl/18964](http://www.elektor.nl/18964)

→ [Raspberry Pi 4 met 4 GB RAM](http://www.elektor.nl/18963)

[www.elektor.nl/18963](http://www.elektor.nl/18963)

→ [Official case voor de RPi 4](http://www.elektor.nl/18962)

[www.elektor.nl/18962](http://www.elektor.nl/18962)

→ [Official Power Supply](http://www.elektor.nl/18961)

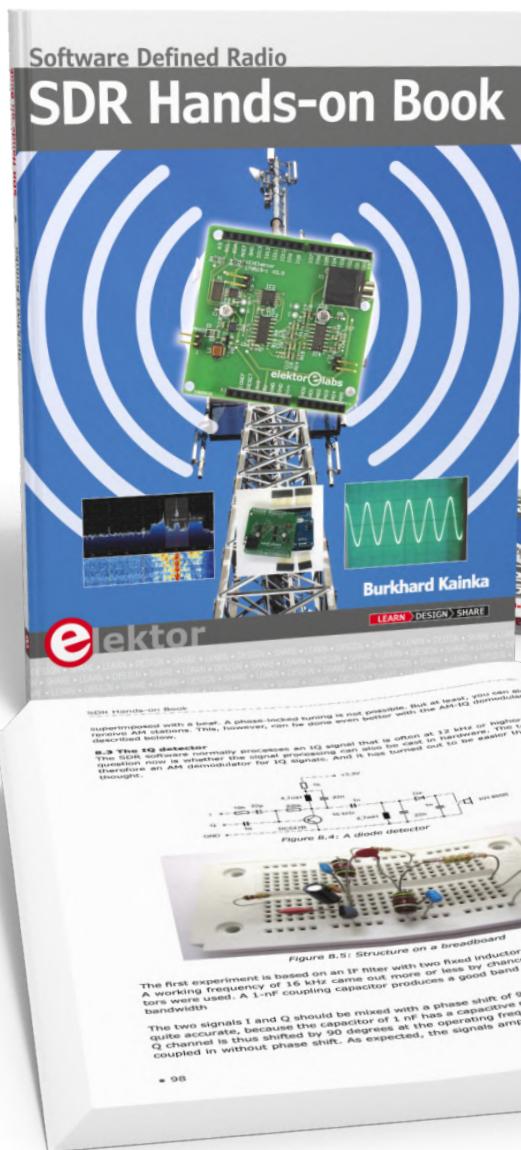
[www.elektor.nl/18961](http://www.elektor.nl/18961)

Iets wat de oudere Raspberry Pi-modellen moesten ontberen was de mogelijkheid 3D-versnelling op de desktop te gebruiken. Met de Raspberry Pi 4 en de driver voor het grafische deel van de VideoCore VI is die mogelijkheid eindelijk beschikbaar. Nu hebben we op de desktop 3D-versnelling in 'venster'-modus, en dat werkt prima!

Velen van u die al een oudere Raspberry Pi bezitten, of die willen beginnen met een of andere bestaande uitbreiding, zitten natuurlijk met de vraag of bestaande add-ons op de Raspberry Pi 4 kunnen worden aangesloten. Het antwoord is: jazeker, maar als u een van die LCD-displays gebruikt die op HDMI moeten worden aangesloten, zult u wel een nieuw kabeltje nodig hebben. Eindelijk hebben we de upgrades gekregen waar we zo lang op moesten wachten: sneller USB, sneller netwerk, een snellere CPU en ook meer RAM. Daar hangt een prijskaartje aan, niet alleen in euro's maar ook in stroomverbruik. Terwijl de instapmodellen met 1 GB RAM ongeveer hetzelfde kosten als de Raspberry Pi 3B+, nemen de prijzen toe wanneer u meer RAM wilt. Tegelijk nemen het stroomverbruik en de dissipatie toe.

## SDR Hands-on Book

Het inspirerende voorwoord van auteur Burkhard Kainka in dit (Engelstalige) boek maakt al meteen duidelijk waarom Software Defined Radio (SDR) zo populair is. SDR blijkt een stimulerende



→
**IN DE STORE**

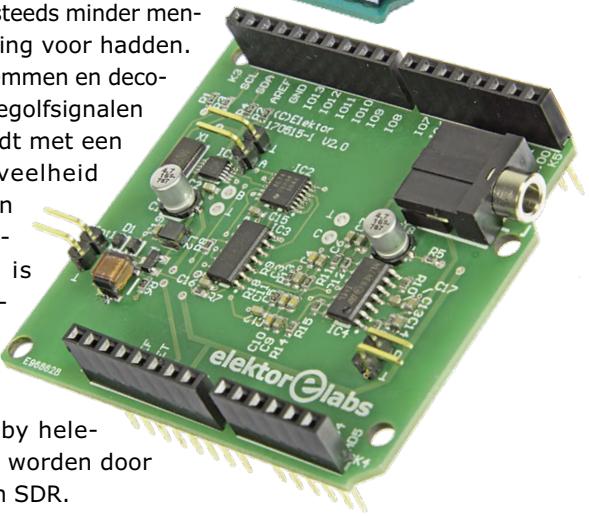
**SDR Hands-on Book (Burkhard Kainka, DK7JD)**  
Boek: [www.elektor.nl/18914](http://www.elektor.nl/18914)  
E-boek: [www.elektor.nl/18915](http://www.elektor.nl/18915)

**Arduino Uno R3**  
[www.elektor.nl/15877](http://www.elektor.nl/15877)

**SDR Shield 2.0**  
[www.elektor.nl/18515](http://www.elektor.nl/18515)



factor voor de radiohobby, waar lange tijd steeds minder mensen belangstelling voor hadden. Omdat het afstemmen en decoderen van kortegolfsignalen nu gedaan wordt met een minimale hoeveelheid hardware en krachtige (gratis) software, is het gemakkelijk te begrijpen waarom nieuwkomers in de radiohobby helemaal gegrepen worden door het concept van SDR.



De tijden dat het bouwen van een kortegolfradio zwarte magie was, waar ingewikkelde spoelen, HF-componenten en speciale constructietechnieken aan te pas kwamen, zijn voorbij. Vandaag de dag is alles wat er nodig is een eenvoudig HF-frontend-kaartje met een IQ-mixer en... software! Meestal draait die op een laptop met ingebouwde geluidskaart van goede kwaliteit. Alle 'intelligentie' van SDR zit in de software.

De juiste volgorde om met SDR aan de slag te gaan is: (1) verzamel de hardware en begrijp hoe die werkt, (2) draai de software. Het boek presenteert daarom eerst een shield voor de Arduino Uno.

De auteur bespreekt gedetailleerd het schema en de werking van het SDR-shield. Dat doet hij in de juiste volgorde, wat prettig is voor de lezer.

Optioneel kan het Elektor LCD-shield bovenop het SDR-shield worden gestoken. Deze configuratie kan worden gebruikt om de huidige SDR-frequentie weer te geven en kan ook dienst doen als een meetinstrument. Goedkoop en nuttig en het geeft de Arduino wat extra werk te doen! Daarna wordt de installatieprocedure voor één van de oudste en beste gratis SDR-softwarepakketten behandeld: G8JCF. Het boek bevat een stap-voor-stap-gids voor het koppelen van de G8JCF SDR-software met de Arduino met het SDR-shield erop en voor het draaien van de firmware van de auteur. De gids bestaat uit een oefening om op gang te komen en laat zien hoe alles moet worden ingesteld voor 7000 kHz in de 40-meterband. Dan volgt de installatie van SDRsharp (ook geschreven als SDR#) voor het frequentiebereik van 24 MHz tot 1800 MHz.

Het hoofdstuk over de antenne is een geslaagde manier om SDR te populariseren bij een nieuw publiek: begin bij het begin, met dat ene onderdeel dat gewoon niet kan worden vervangen door een stuk software: de antenne. Met zijn stap-voor-stap-bespreking van de eigenschappen van ontvangstantennes voor de korte golf en aanpassing aan de ontvanger, slaagt de auteur erin de verwachtingen te temperen van 'appartementsbewoners' die denken signalen van het andere einde van de wereld te kunnen oppikken met niet meer dan een stukje draad.

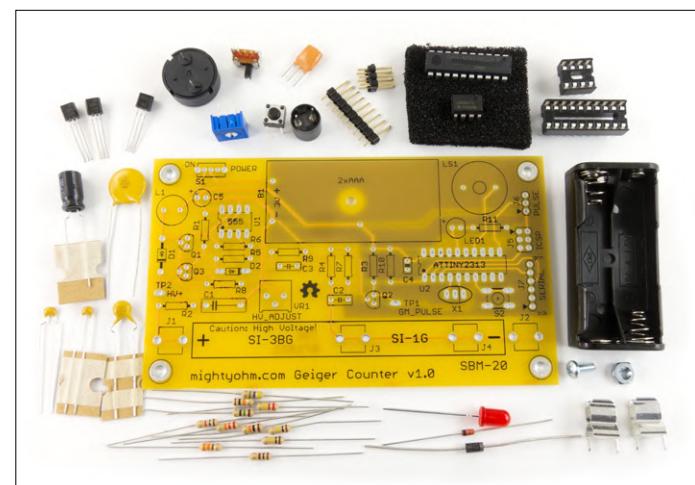
In elk geval is dit boek een must voor iedereen die met gangbare en goedkope hard- en software zoals Arduino en G8JCF de fascinerende wereld van de korte golf met behulp van SDR wil verkennen.

## Bouw uw eigen Geigerteller

De MightyOhm Geiger Counter is een apparaat voor het detecteren van bèta- en gammastraling, die afkomstig kunnen zijn van bijvoorbeeld cesium-134 ( $^{134}\text{Cs}$ ) of cesium-137 ( $^{137}\text{Cs}$ ), radioactieve isotopen van cesium die vrijkomen door nucleaire ongelukken.

Ook uranium (U) en thorium (Th), twee natuurlijk voorkomende radioactieve elementen op aarde produceren dit soort straling. Bij het verval van deze elementen komt onder meer radon (Rn) vrij, wat onder normale omstandigheden gasvormig is en gemakkelijk kan worden ingeademd. Daarom wordt het beschouwd als een gezondheidsrisico. Omdat straling zo schadelijk is, wilt u misschien uw leefomgeving in de gaten houden en investeren in een stralingsdetector zoals de MightyOhm Geigerteller.

De detector produceert een piep (een 'klik') en een lichtflits



telkens als het gas in de sensor, een Geiger-Müller-buis, wordt geïoniseerd door straling. Deze buis heeft een spanning van vele honderden volts nodig om te kunnen werken. Als er ionisatie plaatsvindt, ontstaat er een korte stroompuls die kan worden gemeten en hoorbaar gemaakt; dat zijn de beroemde tikken die u waarschijnlijk uit films kent.

Met deze kennis gewapend kunnen we er zelf een bouwen met behulp van de MightyOhm Geiger Counter kit. Het gaat om 100% open source software en hardware, dus u kunt de werking in detail bestuderen. Het apparaat is voorzien van een LED en een piëzospeaker, zodat u de radioactiviteitsniveaus zowel kunt zien als kunt horen.

Er zijn headers beschikbaar voor seriële communicatie (signaalniveau 3,3 V), in-circuit serieel programmeren van de AVR-microcontroller en een puls-uitgang.

De MightyOhm Geiger Counter is een gemakkelijk te bouwen kit die behoorlijk nauwkeurig straling detecteert. De loggingsmogelijkheden maken het een zeer bruikbaar instrument. ▶

180689-E-01

**IN DE STORE**

→ MightyOhm Geiger Counter Kit++ Bundle  
[www.elektor.nl/18509](http://www.elektor.nl/18509)

# ICL7660:

## reductie van de uitgangsruis

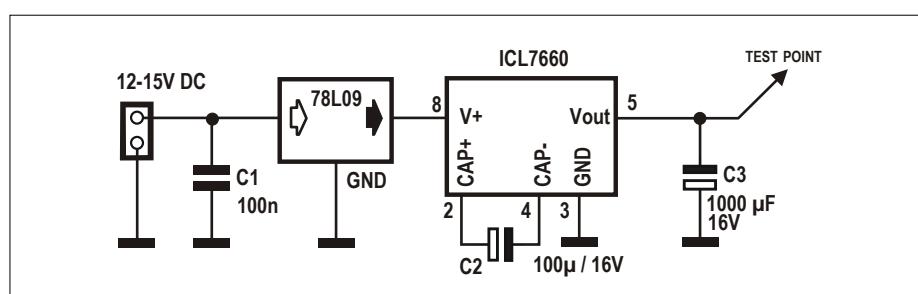
### met een tweede-orde pi-filter

Hesam Moshiri

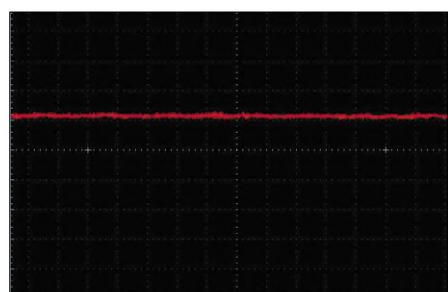
Sommige schakelingen, zoals versterkers met opamps, zijn gevoelig voor storingen op de voedingsrails, en dus moeten de ontwerpers de ruisniveaus daar tot het absolute minimum beperken. Dit artikel gaat over effectieve onderdrukking van de uitgangsruis van een ICL7660. Ook zien we hoe een spanningsconverter op een externe schakeling moet worden aangesloten om aan de EMC-normen te voldoen. Ons doel is te voorkomen dat de schakelstroom van de '7660 de te voeden schakeling beïnvloeden.

Het realiseren van een 'schone' negatieve voedingsrail vormt bij veel batterijgevoede toepassingen een onverwachte uitdaging. De ICL7660/MAX1044 is een bekende chip waarmee eenvoudig een negatieve voedingsspanning kan worden gemaakt. Het gebruik van deze chip en een paar passieve onderdelen lijkt

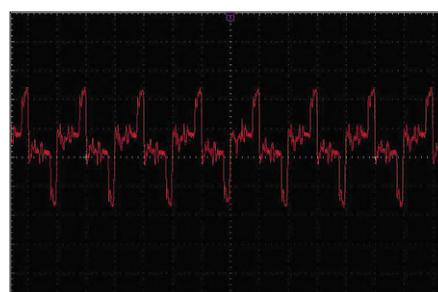
een goede optie, maar kan toch problemen opleveren. De chip bevat een monolithische spanningsconverter met geschakelde condensator die weliswaar efficiënt werkt maar helaas schakelruis op de uitgang produceert. We laten hier een manier zien om een echt 'schone' uitgangsspanning uit de 7660 te halen.



Figuur 1: Algemene opzet van een spanningsinverter met de ICL7660.



Figuur 2: Ruisvloer van de oscilloscoop (zonder probe gemeten, 2 mV/div).



Figuur 3: 50Hz-netspanningsbrom wordt altijd door de probe opgepikt.

#### Passieve analoge filters

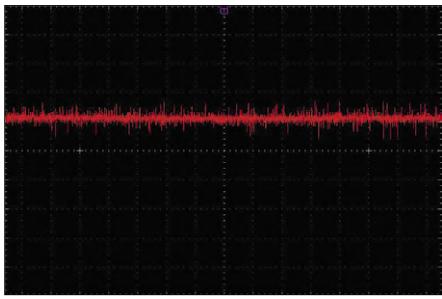
Een analoog filter is een belangrijke elektronische bouwsteen die veel wordt toegepast bij signaalverwerking. Ook vinden we zo'n filter vaak aan de ingang van analoog/digitaal-convertisers (ADC's). Analoge filters kunnen worden onderverdeeld in hoogdoorlaat-, laagdoorlaat- en banddoorlaat-/bandsper-filters. Voor onze toepassing moet het filter eenvoudig en goedkoop zijn, en mag het de uitgangsstroom op geen enkele manier beperken. Ook mag het geen instabiliteit of spanningsverlies aan de uitgang veroorzaken.

#### De ruis van een ICL7660

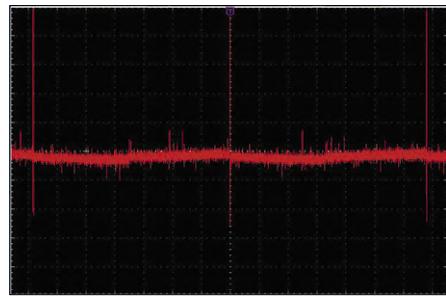
In **figuur 1** zien we de ICL7660 in een gebruikelijke 'inverter'-configuratie met een uitgangsspanning van -9 V. De waarden van C2 en C3 zijn geschikt voor de meeste praktische toepassingen.

Voor dat we de oscilloscoop met deze schakeling verbinden bepalen we eerst de interne ruis van de oscilloscoop. In **figuur 2** ziet u de interne ruisvloer van de oscilloscoop in eenkanaals-modus, een schaal van 2 mV/div en een bandbreedte van 20 MHz. We zien een DC-offset van ongeveer 2 mV. Met aangesloten probe pikte de oscilloscoop een flinke hoeveelheid 50Hz-brom op (**figuur 3**).

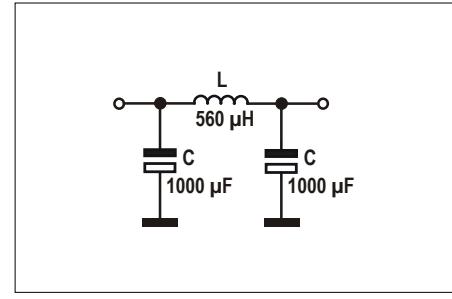
Een manier om bij een aangesloten probe bromgerelateerde ruis te elimineren en



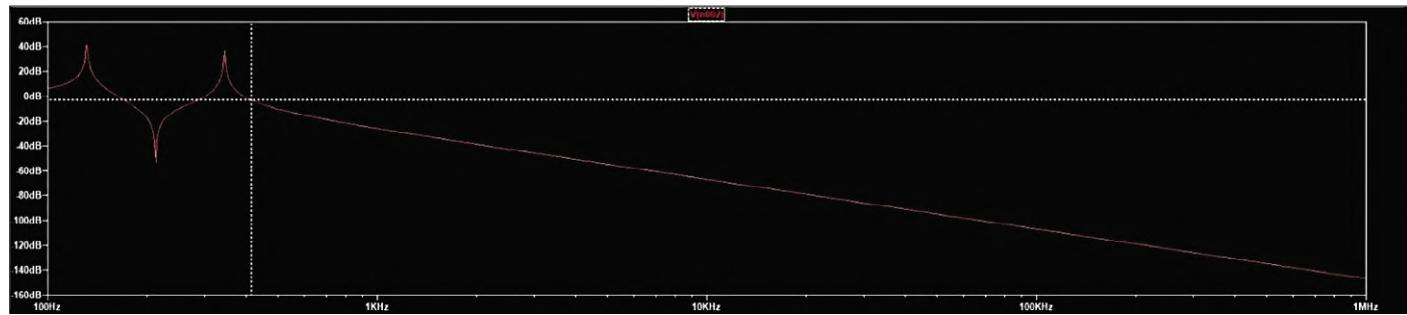
Figuur 4: Omgevingsruis met aangesloten batterij; probe: x1; oscilloscoop-ingang: AC.



Figuur 5: Uitgangsspanning ICL7660 zonder extra filtering (2 mV/div).



Figuur 6: Eerste-orde laagdoorlaat-pi-filter.



Figuur 7: Frequentiekarakteristiek van het eerste-orde filter.

alleen het niveau van de omgevingsruis te bepalen, is de probe te verbinden met een batterij. De batterij levert een ‘pure’ gelijkspanning, dus als de oscilloscoop-ingang in AC-modus staat wordt de waargenomen ruis alleen veroorzaakt door storingen in de testomgeving. **Figuur 4** toont de gemeten ruis als de batterij op de probe is aangesloten.

Nu kunnen we een eerste versie van de spanningsconverter met ICL7660 bouwen en de oscilloscoop met het testpunt verbinden. De schakeling moet uit een batterij worden gevoed (ruisvrije bron!) om er zeker van te zijn dat alles wat we ‘bovenop’ de ruis van figuur 4 meten, de schakelruis is die door de schakeling zelf wordt geproduceerd.

In **figuur 5** ziet u het signaal op het testpunt (2 mV/div). Hoewel een condensator van 1000  $\mu\text{F}$  aan de uitgang is opgenomen, die op zichzelf al voor een serieuze filterwerking zorgt, zit er nog steeds veel ruis op de voedingsrail. Uit de metingen blijkt dat de frequentie van de schakelruis rond 7,3 kHz ligt en dat deze een variabele amplitude heeft met pieken tot 15 mV<sub>pp</sub>. Om een zo zuiver mogelijke DC-spanning te maken moeten we een laagdoorlaatfilter gebruiken en alle ruis bij en boven de gemeten ruisfrequentie onderdrukken.

Hier voor zijn zowel RC- als LC-filters geschikt. RC-filters vallen echter af omdat het weerstandselement de stroom beperkt en een spanningsval veroorzaakt, met name in hogere-orde filters. Een LC-filter heeft dus de voorkeur. In de praktijk is een 3-element-filter een efficiënte oplossing, dus laten we condensator C3 zitten en voegen hier een LC-filtertrap aan toe zoals in **figuur 6** is te zien. We krijgen zo een CLC-filter of ‘pi’-filter.

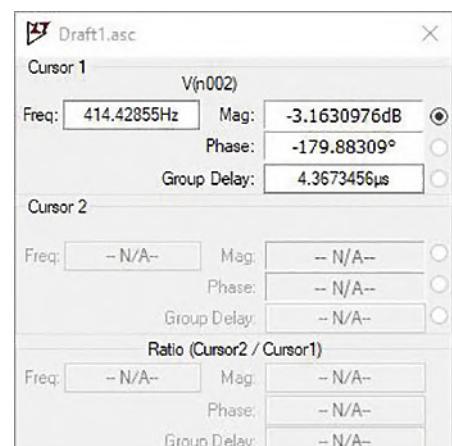
Een kenmerk van vrijwel ieder filter is de afsnijfrequentie. Een laagdoorlaatfilter onderdrukt signalen boven de afsnijfrequentie en laat de signalen onder deze ‘drempel’ door. Met een simulatieprogramma zoals LTspice kunnen we het gedrag van het filter voorspellen. Ook kan de effectiviteit van het filter worden bekeken door de waarden van de onderdelen L en C te variëren. We moeten hierbij waarden kiezen die een lage afsnijfrequentie opleveren en het ‘scherpe’ filteren vlak voor de ruis doen.

De uitgangscondensator (C3) heeft al een waarde van 1000  $\mu\text{F}$ , daar gaan we mee verder. U bent echter niet aan deze waarde gebonden. Met name als de afmetingen van de print van belang zijn, kunt u uw eigen passende waarde kiezen en de filterkarakteristiek simule-

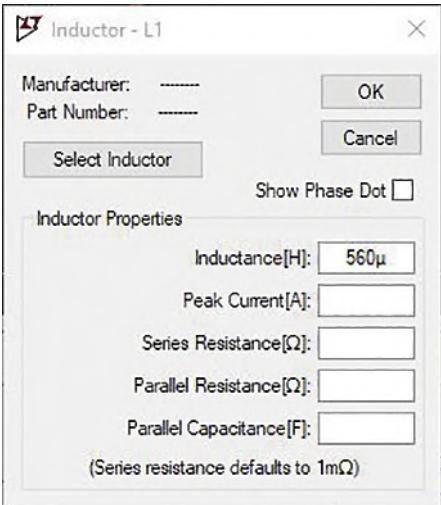
ren voordat u verder gaat met het printontwerp en de bouw.

**Figuur 7** toont de frequentiekarakteristiek van het filter. Voor de spoel is een waarde van 560  $\mu\text{H}$  gekozen. De AC-sweepfrequentie is ingesteld van 100 Hz tot 1 MHz.

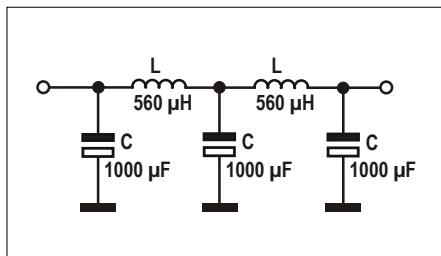
De afsnijfrequentie van het filter is de frequentie op de X-as waarbij de grafiek door het -3dB-niveau op de Y-as gaat. LTspice doet het rekenwerk voor u (zie **figuur 8**). De afsnijfrequentie ligt op ongeveer 414 Hz en de faseverschuiving



Figuur 8: Afsnijfrequentie en faseverschuiving van het filter (LTSpice).



Figuur 9: Instelbare spoelparameters in LTSpice.



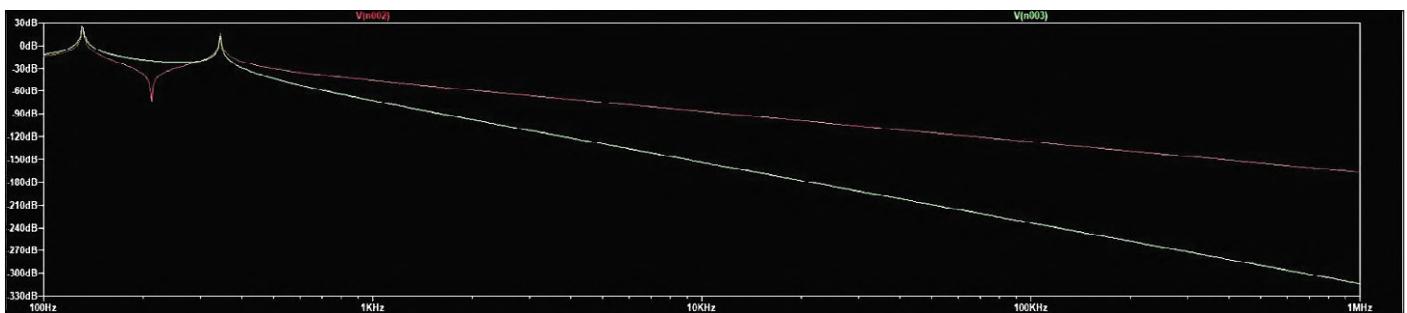
Figuur 10: Tweede-orde laagdoorlaat-pi-filter, met de geselecteerde waarden.

is bijna  $-180$  graden. Let op, bij deze berekeningen wordt uitgegaan van ideale (en dus theoretische) onderdelen. Voor een meer realistische simulatie kunt u de waarden van de onderdelen aanpas-

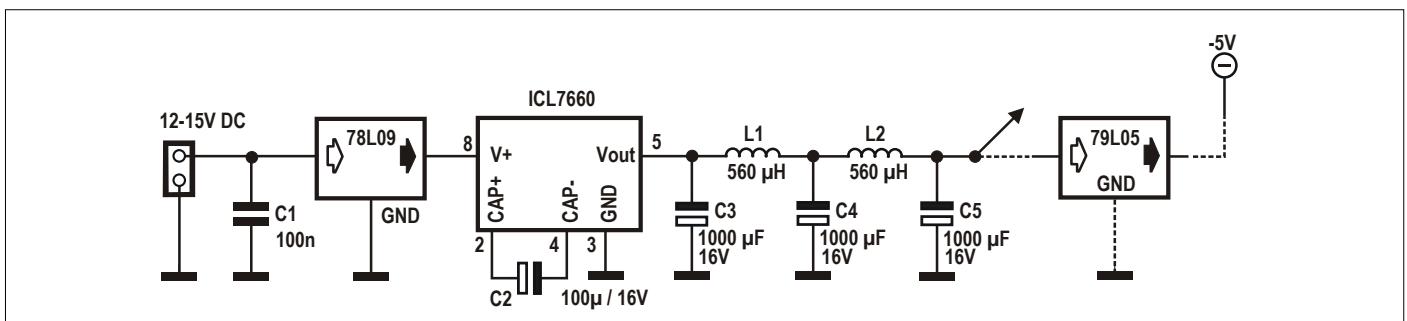
sen aan de waarden uit de datasheet. In **figuur 9** ziet u de voor de spoel opgegeven LTSpice-parametres zoals serierestand, piekstroom enzovoort.

Dit is een eerste-orde CLC- of pi-filter. We kunnen het filter ‘krachtiger’ (lees: effectiever) maken door de orde te verhogen. In **figuur 10** ziet u het schema van een tweede-orde pi-filter met dezelfde onderdelenwaarden.

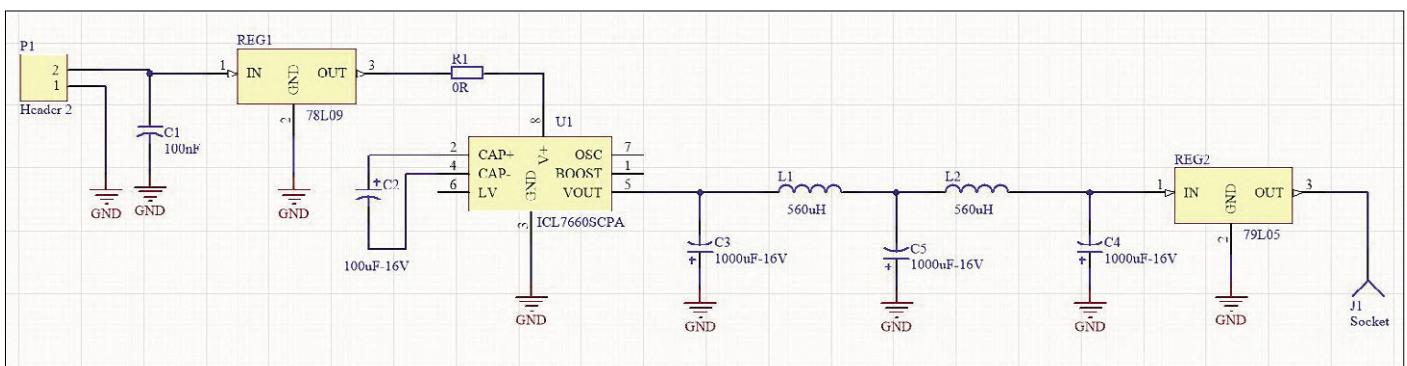
Als we het gedrag van het filter met LTSpice bekijken, is duidelijk te zien dat het tweede-orde filter een scherpere en stabielere karakteristiek heeft. Simpel gezegd: het filtert beter. In **figuur 11** ziet u de karakteristiek van het tweede-orde filter (groene lijn) in vergelijking met die van het eerste-orde filter (rood).



Figuur 11: Karakteristiek van het tweede-orde pi-filter (groene lijn).



Figuur 12: Volledige schakeling van de 7660-spanningsinverter, inclusief optionele negatieve spanningsregelaar.



Figuur 13: Schema van de low-noise voedingsconverterschakeling zoals gemaakt in Altium Designer.

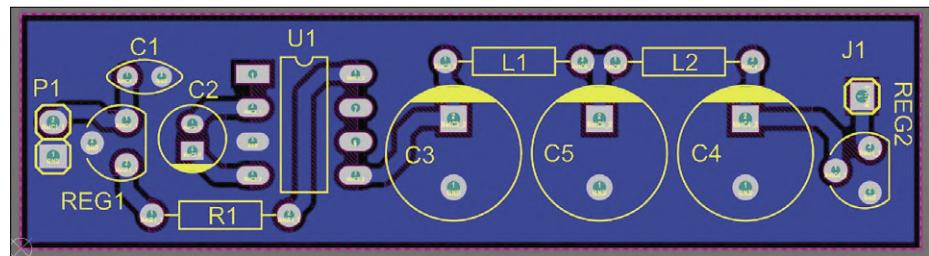
## En nu de praktijk

Nu wordt het tijd om de ICL7660-convertiserschakeling samen met het filter te testen. **Figuur 12** toont de volledige schakeling. Houd er rekening mee dat de uitgangsspanning van de ICL7660 niet is gestabiliseerd. Als u een gestabiliseerde spanning nodig hebt (bijvoorbeeld -5 V), kunt u zoals weergegeven een negatieve spanningsregelaar aan de uitgang van het filter zetten.

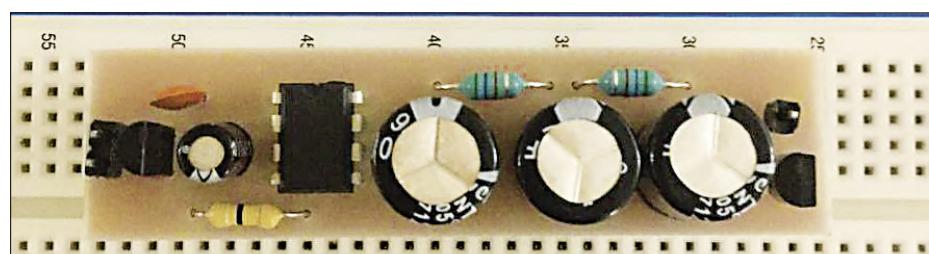
Voor het beste resultaat moet een passende print voor de schakeling worden ontworpen. Ik gebruikte hiervoor Altium Designer met de SamacSys onderdelen-zoekmachine-plugin voor Altium om de afmetingen van de onderdelen te bepalen. Het gebruik van deze plugin reduceert de ontwikkeltijd van de print aanzienlijk omdat het de herhalende schemabibliotheek/PCB-footprint acties elimineert.

In **figuur 13** en **figuur 14** ziet u het schema en de print, beide in Altium-stijl. De layout-bestanden kunnen gratis van de projectpagina bij dit artikel [1] worden gedownload. Een eerste ontwerptip is om een degelijk koper-massavlak te gebruiken, dus de ongebruikte ruimte zoveel mogelijk met koper op te vullen. Het massavlak helpt ruis en stoorpieken te onderdrukken. R1 is een nul-ohm-weerstand die voorkomt dat het kopervlak onder C2 nodoeloos wordt afgekapt. Het grote massavlak vormt een minimale impedantie voor de stroom in de schakeling.

Volgens de EMC-richtlijnen mag een geschakelde voeding (SMPS) geen gemeenschappelijk massavlak hebben met de rest van de schakeling. Als gevolg hiervan moet de verbinding tussen de SMPS-massa en de massa van de te voeden schakeling op één punt plaatsvinden. Voor de SMPS-schakeling is wel een massavlak gebruikt, maar de massaverbinding met het externe circuit loopt slechts via één headerpin.



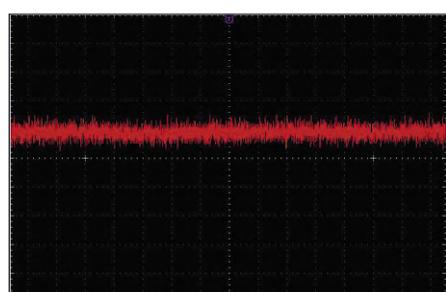
Figuur 14: De met Altium Designer ontworpen print voor de converterschakeling.



Figuur 15: De volledig gemonteerde print.

In **figuur 15** ziet u de volledig gemonterde schakeling. De aan de uitgang van de schakeling gemeten ruis is te zien in **figuur 16** (2 mV/div). In vergelijking met figuur 4 wordt duidelijk hoe dicht deze nu bij de ruisvloer van de oscilloscoop komt! ▶

(190127-04)



Figuur 16: Uitgangsruis van de converter met filter (2 mV/div; x1).

## Weblink

[1] PCB-layoutbestanden (Altium): [www.elektormagazine.nl/190127-04](http://www.elektormagazine.nl/190127-04)

**IN DE STORE**

→ Boek (Engelstalig): The LTspice IV Simulator  
[www.elektor.nl/the-ltspice-iv-simulator](http://www.elektor.nl/the-ltspice-iv-simulator)

## 'Best practice' voor oscilloscoopmetingen

Voor echt nauwkeurige metingen moet de massa-aansluitdraad van de probe worden verwijderd, en moet de verbinding met de afscherming van de probe worden gemaakt met een massaveer. In **figuur 17** ziet u een voorbeeld. Op deze manier kunt u er zeker van zijn dat de signaalruis op het oscilloscopescherm alleen afkomstig is van het testpunt in de schakeling, en dat de invloed van omgevingsruis zoveel mogelijk wordt beperkt.



17

# Een dag met LoRa

## conferentie van de LoRa Alliance in Berlijn

Jens Nickel (redactie Elektor)

Het Internet of Things moet ons leven comfortabeler, veiliger en bovenal duurzamer maken. Het IoT leeft van een enorme hoeveelheid sensoren die draadloos data versturen, jarenlang op één en dezelfde batterij. Toepassingen gaan van parkeerplaats- of magazijnbeheer via het brede terrein van milieusensoren tot en met *tracking* van goederen en dieren. Door de hoge eisen aan reikwijdte, doordringend vermogen in gebouwen en extreme energiezuinigheid komen klassieke draadloze technieken als WiFi en Bluetooth niet in aanmerking. Voor efficiënte draadloze data-overdracht met geringe bitsnelheden bestaan inmiddels nieuwe standaarden zoals LoRa, Sigfox en NB-IoT, die op een gestaag groeiende markt met elkaar concurreren. Hierbij ontvangen basisstations (ook gateways genoemd) de sensordata en geven die via Wifi of Ethernet naar het internet door. In het ecosysteem van deze radiotechnologie zien we *remote servers* die de data op afroep beschikbaar maken voor de eigenaar van de sensorknooppunten (via een Web-API, MQTT, e-mail enzovoort). Voor ontwikkelaars zijn goedkope ontwikkelboards, gratis gebruik van de infrastructuur voor evaluatiedoeleinden, softwarebibliotheeken en voorbeeldapplicaties beschikbaar.

LoRa heeft zich in deze markt een goede positie verworven. Fysiek is deze radiostandaard gebaseerd op CSS-modulatie zoals ontwikkeld door halfgeleiderfabrikant Semtech. Deze firma produceert en licentieert de benodigde transceiverchips en SoC's. In tegenstelling tot bij concurrent Sigfox bestaat hiervoor geen speciaal netwerk, maar wordt gebruik gemaakt van een bonte veelheid van kleinere en grotere private en openbare netwerken. Om sensorknooppunten flexibel in de diverse netwerken op te kunnen nemen is het natuurlijk niet alleen nodig dat transceivers elkaar op fysiek niveau begrijpen om 1:1 data heen en weer te sturen (hoewel dat op zich wel werkt), zoals we in deze Elektorlabs-uitgave bewijzen met het

project 'Draadloze accu-voltmeter'). Daarnaast moet er een aanmeldprotocol zijn, net als *collision management*, timing-afspraken en veiligheidsvoorzieningen. Deze aspecten worden allemaal gewaarborgd in de LoRaWAN-standaard, ondosteund door een steeds groter wordende alliantie van uiteenlopende firma's [1][2]. Behalve Semtech en verscheidene halfgeleiderfabrikanten hebben ook telecommunicatie-ondernemingen, softwarefirma's en allerlei gebruikers zich aangesloten bij de LoRa Alliance [3], die inmiddels bij elkaar meer dan 500 leden heeft. Elk jaar komt deze alliantie samen in een andere stad; in 2019 was Berlijn aan de beurt.

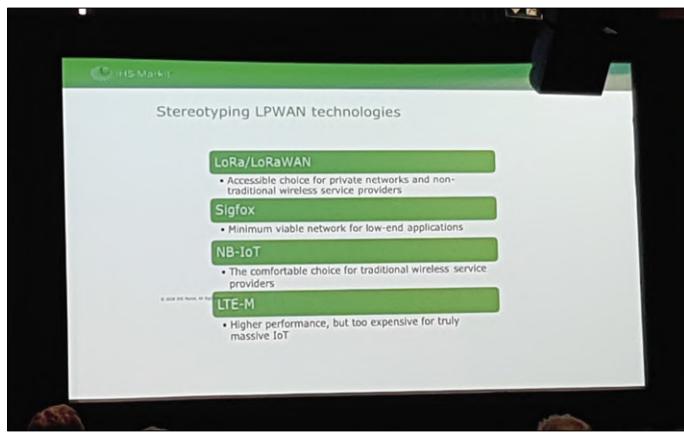
De eerste twee dagen werd in besloten kring vergaderd om verder te schaven aan het protocol en om ervaringen uit te wisselen. Op de derde dag waren ook pers en (potentiële) klanten uitgenodigd. Daarbij hoort uiteraard ook altijd dat men zichzelf trakteert op een feestje. Na de keynote van Alliance-chef Donna Moore, die er vooral de nadruk op legde dat het samenwerkingsverband in het afgelopen jaar volwassen geworden is, werden er prijzen uitgereikt aan experts die belangrijke bijdragen hebben geleverd aan de verdere ontwikkeling van het protocol en de ondersteuning daarvan. Daarna was het woord aan de sponsoren van de bijeenkomst. Hun betoog diende natuurlijk ook om zich te onderscheiden van hun concurrenten. Marktonderzoeksbedrijf ABI Research presenteerde een studie die vooral de verschillen tussen LoRa en NB-IoT liet zien (**figuur 1**) [5]. Het doet de LoRa-partners natuurlijk pijn dat veel Europese telecom-providers zich niet bij de alliantie hebben aangesloten en in plaats daarvan met NB-IoT hun eigen standaard promoten. Argumenten als een reeds bestaande infrastructuur en een eigen, gereserveerd frequentiebereik spreken natuurlijk in het voordeel van NB-IoT, terwijl LoRa in Europa opereert in de 868-MHz-band. Daartegenover wees men er in Berlijn op dat LoRa-sensorknooppunten energiezuiniger, goedkoper en compacter kunnen uitvallen. Op de bijbehorende productpresentatie kon men dat aan de hand van een



Figuur 1. In Berlijn werd een onderzoek gepresenteerd dat LoRa vergeleek met NB-IoT.



Figuur 2. Twee keer hetzelfde multisensor-knooppunt, de ene met LoRa- en de andere met NB-IoT-overdracht ([www.tektelic.com](http://www.tektelic.com)).



Figuur 3. LoRa vergeleken met NB-IoT en Sigfox.



Figuur 4. De startup Polygravity presenteerde een oplossing voor roaming en overkoepelende betaling van de communicatie.

### multi-sensor-knooppunt demonstreren (**figuur 2**).

Concurrent Sigfox werd in Berlijn aan het andere eind van het toepassingsspectrum ingedeeld: namelijk geschikt voor *low end*-applicaties waarbij heel weinig bytes per uur of per dag worden verzonden (**figuur 3**). Het aantrekkelijke van het Sigfox-ecosysteem is de eenvoud ervan: één en dezelfde firma licentieert de chips en neemt de basisstations en het serverplatform voor zijn rekening waarmee gebruikers de sensor-knooppunten beheren en meetgegevens kunnen ontvangen. LoRa is daarentegen veel heterogener. Gebruikers moeten ofwel hun eigen netwerk (gateways en server) inrichten ofwel de sensorknooppunten bij een commerciële netwerkleverancier onderbrengen. De uitbreidbaarheid van deze netwerken is meestal beperkt, maar voor toepassingen die een hoge graad van mobiliteit verlangen, zoals tracking van goederen over grote afstanden, bestaan er inmiddels ook al roaming-oplossingen (**figuur 4**).

Een vrij LoRa-netwerk dat naar mondiale dekking streeft is TheThingsNetwork (TTN) [6], opgezet door vrijwilligers die allemaal een of meer gateways in beheer hebben. Daarbij hoort ook een netwerkplatform waar men dan de data van eigen sensoren via een API kan oproepen. In Berlijn was ook TTN-oprichter Wienke Giezeman van de partij. Ook LoRaServer.io biedt een open netwerkserver [7]. Er was een workshop waar gedemonstreerd werd hoe met een Raspberry Pi, een aangesloten gateway-module en open source-software een LoRaWAN-gateway gebouwd kan worden die data kan doorsturen naar zowel TheThingsNetwork als naar een LoRa-server (**figuur 5**). Als we de tijd die de Raspberry Pi bezig was met de Github-download en de installatie niet meerekenen, dan kost dat niet meer dan een paar minuten.



Figuur 5. ICFOSS uit India had Raspberry Pi's en gateway-modules bij zich voor de workshop *LoRaWAN in a box*.

In dit artikel kunnen we maar enkele tipjes van de sluier(s) oplichten, maar er zijn al Elektorlabs-artikelen over dit onderwerp gepland, evenals LoRaWAN! ▶

(190340-04)

IN DE STORE

- Dragino LoRa IoT Development Kit  
[www.elektor.nl/18335](http://www.elektor.nl/18335)
- LoRa Nexus Board  
[www.elektor.nl/lora-nexus-board-arduino-mini-shape](http://www.elektor.nl/lora-nexus-board-arduino-mini-shape)

### Weblinks

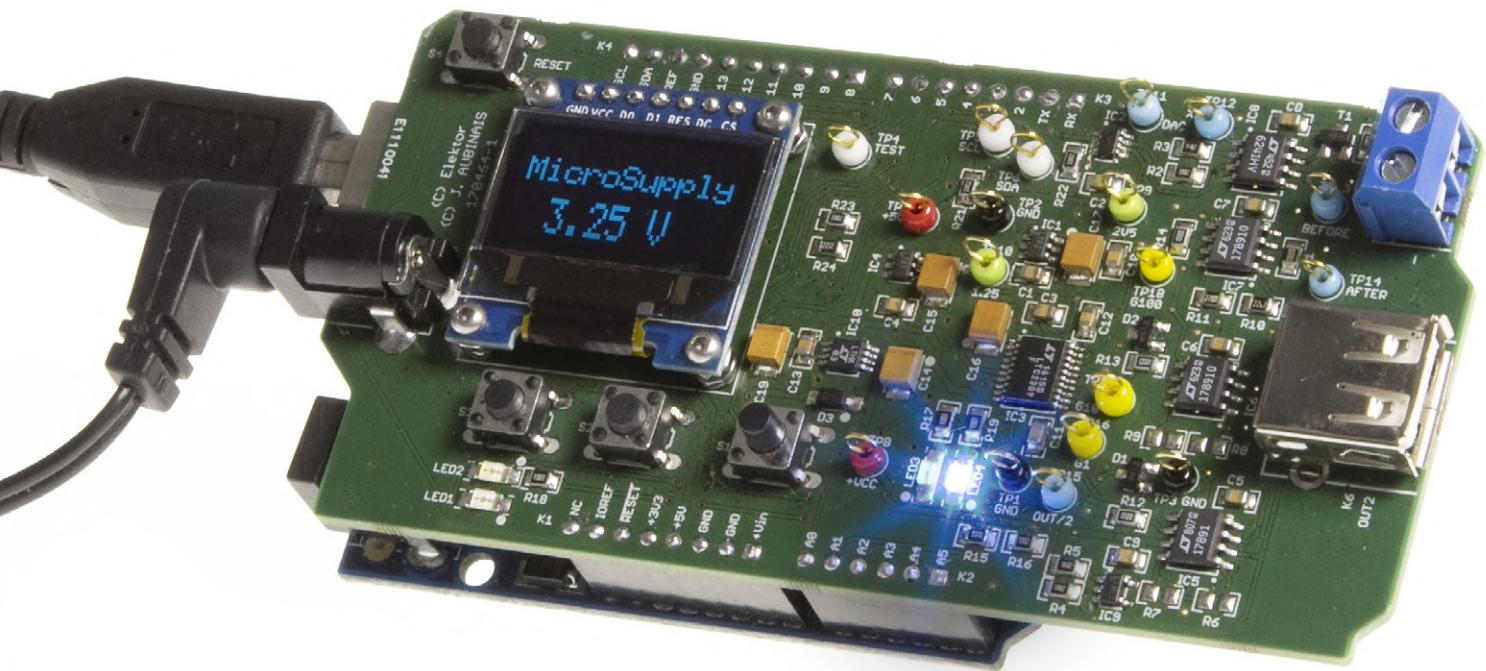
- [1] LoRaWAN-specificaties: <https://lora-alliance.org/sites/default/files/2018-07/lorawan1.0.3.pdf>
- [2] Software-'bouwdoos': <https://lora-developers.semtech.com/resources/tools/basic-mac/welcome-basic-mac/>
- [3] LoRa Alliance: <https://lora-alliance.org>
- [4] LoRa Alliance: meeting in Berlijn: <https://lora-alliance.org/events/berlin-annual-members-meeting-2019>
- [5] Onderzoek LoRaWAN versus NB-IoT: <https://lora-alliance.org/resource-hub/lorawan-and-nb-iot-competitors-or-complementary>
- [6] The Things Network: <https://www.thethingsnetwork.org/>
- [7] LoRa Server: [www.loraserver.io/loraserver/overview/](http://www.loraserver.io/loraserver/overview/)

# MicroSupply

## labvoeding voor 'connected' apparaten

Jennifer Aubinais (Frankrijk), met dank aan Jean-Luc en Philippe

Hoeveel stroom verbruikt mijn IoT-apparaat? Komt het verbruik in de slaapmodus overeen met de specificaties van de fabrikant? Hoelang gaat de CR2032-batterij van het apparaat mee? Als u dit soort vragen hebt, dan is deze MicroSupply (een regelbare voeding in de vorm van een Arduino-shield die zeer kleine stromen kan meten) iets voor u. Met de bijbehorende PC-software kunt u het verbruik van uw IoT-apparaat weergeven en opslaan.



### Eigenschappen

- Shield voor Arduino Uno.
- Uitgangsspanning regelbaar van 1,5 tot 5 V in stappen van 0,05 V.
- Uitgangsstroom wordt gemeten en weergegeven van 1 µA tot 40 mA.
- Meetfrequentie 1 kHz.
- PC-software voor weergave van de via een seriële poort ontvangen meetwaarden.
- Software-stroombegrenzing tot 40 mA.
- Software-spanningsonderbreking (uitgang lager dan 100 mV).
- Externe voeding nodig (10...12 V).

Dit is een project met een lange geschiedenis, want dit is al versie nummer vijf! De schakeling evolueerde niet alleen door technische problemen die bij het ontwerpen moesten worden overwonnen, maar ook door gedachtenwisselingen van de auteur tijdens seminars en beurzen. Er zullen altijd nog wel verbeteringen mogelijk zijn, maar er komt een moment waarop je er een punt achter moet zetten...

## Blokschema

Voordat we gedetailleerd ingaan op de elektronica van het project, laten we eerst de verschillende modules waar het uit bestaat, de revue passeren (**figuur 1**):

- een digitaal/analoog-converter (DAC) levert de regelbare voedingsspanning voor het IoT-apparaat;
- omdat deze converter niet erg veel stroom kan leveren, is er een kleine vermogensversterker aan toegevoegd die er maximaal 40 mA van maakt;
- door de spanningsval over de shuntweerstand te meten kan de stroom door de belasting (het IoT-apparaat) worden bepaald. Omdat de shuntweerstand in dit artikel vaak wordt genoemd, wordt deze als een aparte module beschouwd;
- de spanning over de shuntweerstand passeert eerst verschillende instrumentatieversterkers voordat deze in een digitaal signaal wordt omgezet. Waarom verschillende versterkers? De te meten stroom kan variëren van 1  $\mu$ A tot 40 mA, dus de grootste stroom is 40.000 maal de kleinste. Het is niet eenvoudig om dat met de gewenste precisie in één meetbereik te vangen. Daarom is het geheel opgedeeld in drie bereiken, met bijbehorende instrumentatieversterkers: x1, x10 en x100.
- met een analoog/digitaal-converter (ADC) worden de spanningen (stroommeetwaarden) aan de uitgangen van de drie instrumentatieversterkers in een digitaal signaal omgezet;

tot slot worden de uitgangsspanning en -stroom op een OLED-display weergegeven. Dit display wordt ook gebruikt voor het weergeven van meldingen zoals de verificatie van de goede werking tijdens de testfase van de voeding.

## De eerste versie...

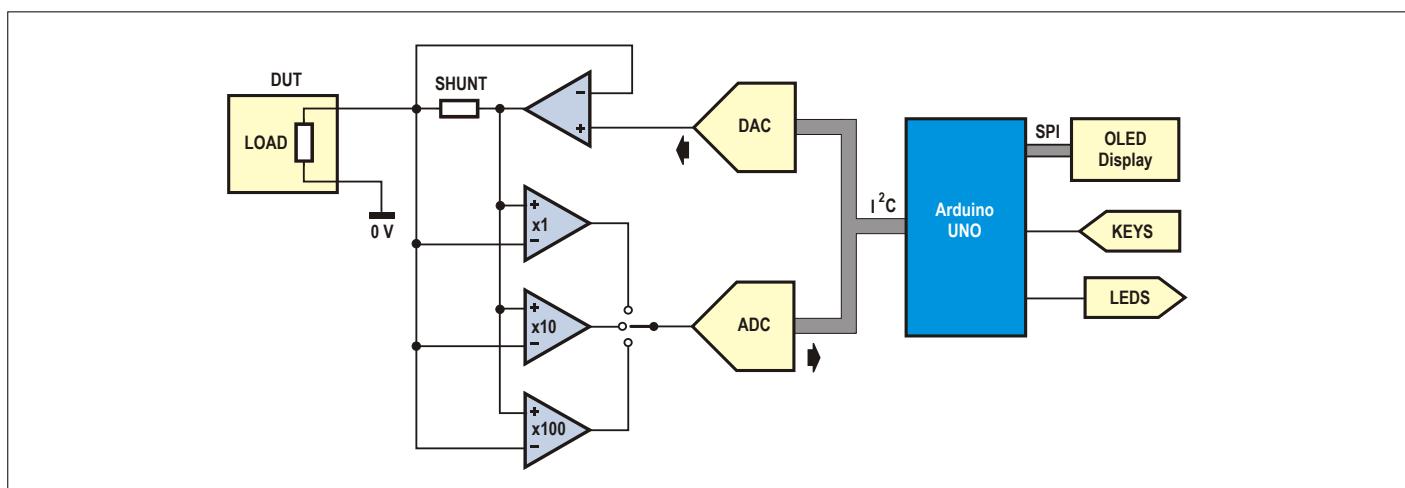
...was een eenvoudig Arduino-shield dat een uitgangsspanning tussen 0 en 5 V leverde en dat de geleverde stroom bepaalde door de spanning over een kleine shuntweerstand te meten. Op een OLED-scherm van 0,96 inch werden de meetwaarden periodiek weergegeven. Vervolgens leek het een goed idee om de meetwaarden naar een PC te verzenden om ze te kunnen weergeven en eventueel te bewerken of op te slaan. Het doel was niet om een digitale oscilloscoop te maken, maar om een interessant aantal metingen per tijdseenheid te kunnen weergeven. Een te langzame opeenvolging van metingen is niet nuttig omdat we het verbruik niet per uur of per dag willen weergeven, en een erg snel ritme, bijvoorbeeld iedere microseconde, vereist een complexe schakeling. De keuze is uiteindelijk gevallen op een meetfrequentie van 1 kHz, dat wil zeggen 1.000 metingen per seconde. De oorspronkelijk gekozen ADC was niet snel genoeg om elke milliseconde een conversie uit te voeren en moest dus worden vervangen.

## Shuntweerstand

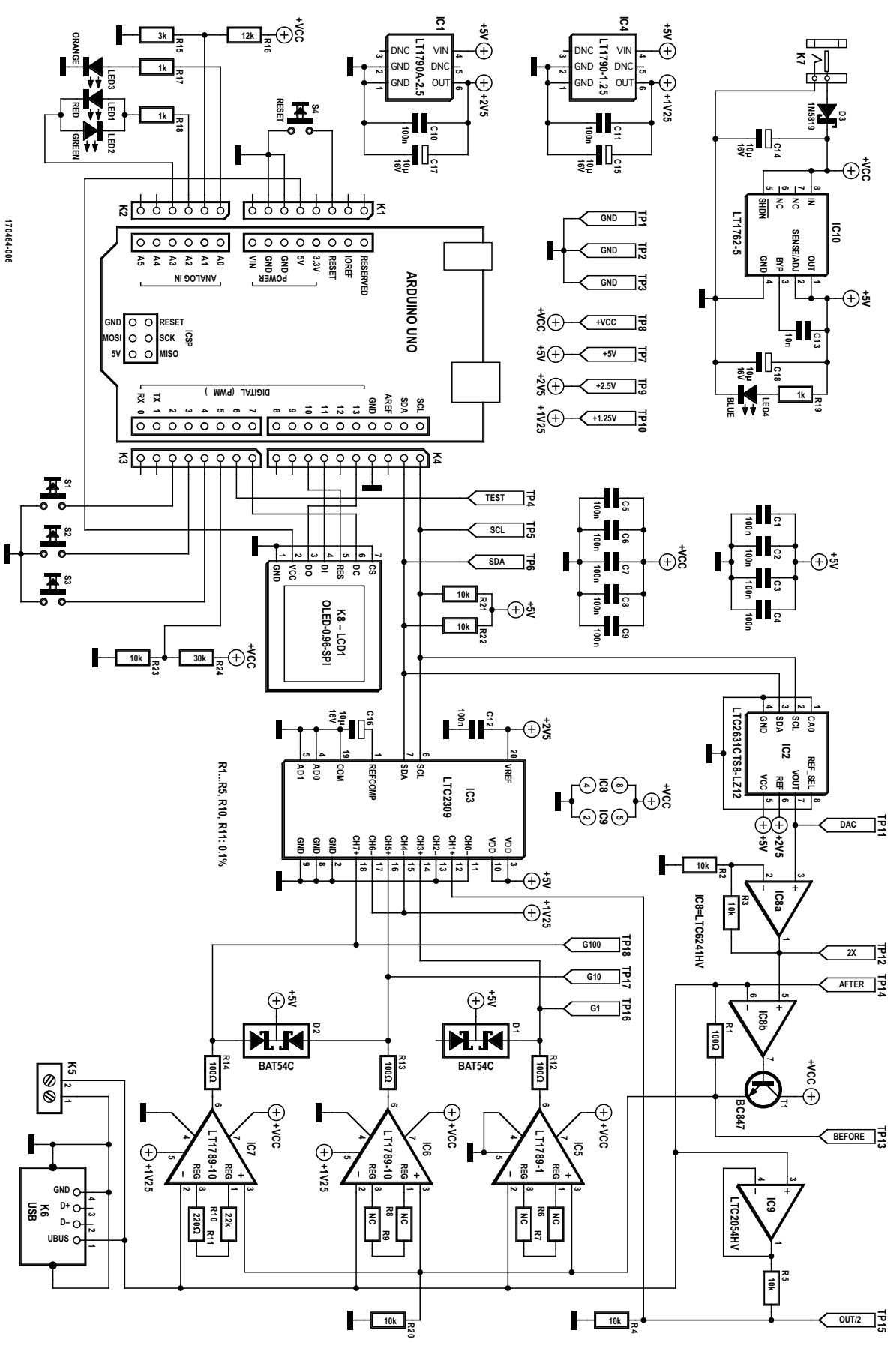
Een ander probleem leverde het gewenste meetbereik van 1  $\mu$ A tot 40 mA op. In de eerste versie had de shuntweerstand een waarde van 10  $\Omega$ . Een stroom van 40 mA geeft over deze weerstand



een spanningsval van 400 mV, bij 1  $\mu$ A is dit 10  $\mu$ V. Dat is erg weinig. Met een hogere waarde voor de shunt (bijvoorbeeld 100  $\Omega$ ), wordt dit probleem voor een deel opgelost, maar de maximale ingangsspanning van de ADC mag niet worden overschreden. Bovendien wordt de spanning over de shunt opgeteld bij de voedingsspanning van het IoT-apparaat. Als dit wordt gevoed met 5 V en een stroomverbruik van 40 mA heeft, dan is er met een shunt van 100  $\Omega$  een totale voedingsspanning nodig van 9 V (5 + 0,04 × 100) om het IoT-apparaat van de juiste spanning te voorzien. Kort gezegd: hoe hoger de waarde van de shuntweerstand, des te groter wordt de



Figuur 1. Blokschema van de MicroSupply.



Figuur 2. Het uitgewerkte schema van het shield.

spanning over de shunt bij kleine stromen, maar moet ook de totale voedingsspanning hoger worden gemaakt.

### Drie instrumentatieversterkers

Een versterker met regelbare versterking is de ideale oplossing. Versterkingsfactoren van bijvoorbeeld 1x voor de grote stromen, 10 x voor het middengebied en 100 x voor de kleine stromen maken het mogelijk om zowel kleine als grote stromen met dezelfde precisie te meten. Om de spanning over de shunt te meten hebben we een differentiële versterker nodig, bij voorkeur van goede kwaliteit, en daarom hebben we gekozen voor een instrumentatieversterker. Helaas heeft dit type versterker een vaste versterkingsfactor. Het is mogelijk om deze versterkingsfactor met omschakelbare weerstanden te wijzigen, maar hiervoor zijn meerdere schakelaars nodig plus de signalen om die aan te sturen (plus natuurlijk wat software). Er is uiteindelijk gekozen voor een parallelschakeling van drie versterkers met verschillende versterkingsfactoren.

### Uitgangsspanning

Er wordt een digitaal/analoog-converter (DAC) gebruikt om een regelbare spanning van 0 tot 5 V te leveren, in de definitieve versie loopt het bereik van 1,5 V (een hogere waarde dan de 1,25V-referentiespanning van de instrumentatieversterkers, zie verderop) tot 5 V. Met een referentiespanning van 2,5 V en een resolutie van 12 bits levert de converter dus een uitgangsspanning van 0 tot 2,5 V in stappen van 0,61 V ( $2,5 / (2^{12}-1) = 2,5 / 4.095$ ). Om een spanning tot 5 V te verkrijgen wordt een opamp met een versterkingsfactor 2 gebruikt. De stapgrootte wordt dan 1,22 mV.

### Verlagen van de uitgangsseerstand

Nu moet nog het probleem worden opgelost van de door de shunt verhoogde uitgangsseerstand van de voeding. Het doel is om een constante uitgangsspanning te leveren die onafhankelijk is van de stroom die door de belasting wordt opgesoupeerd, we moeten deze weerstand dus compenseren. Dit is mogelijk met een tegenkoppelplus rond de shunt. Er wordt een opamp met 'versterkte' uitgang (via T1 om de maximale stroom van 40 mA te kunnen leveren) gebruikt om de spanning op de uitgang constant te houden, zelfs als de uitgangsstroom

varieert tussen 1 µA en 40 mA. Naast het feit dat een maximale stroom van 40 mA voldoende is, komt deze ook overeen met de maximale stroom die door de shunt van  $100 \Omega$  mag lopen zonder dat de maximale spanning op de ingang van de ADC wordt overschreden (4,096 V).

### We hebben een externe voeding nodig

Hierboven zijn we tot een systeem gekomen dat een stabiele spanning van 1,5 tot 5 V kan leveren en het stroomverbruik kan meten. De prijs die we hiervoor naast de complexiteit van de schakeling moeten betalen is een voeding voor het geheel die een spanning van minstens 9 V levert (zie hierboven). De eerste versie beschikte hiervoor over een spanningsverdubbelaar, de LTC660, die de 5 V afkomstig van het Arduino-board opvoerde naar 10 V. Maar de op deze manier verkregen spanning bevatte veel ruis en we zijn er niet in geslaagd deze voldoende te onderdrukken. Uiteindelijk hebben we besloten de spanningsverdubbelaar weg te laten en deze te vervangen door een externe voeding. Dit besluit wordt gerechtvaardigd door het feit dat de MicroSupply een labinstrument is en dat ieder zichzelf respecterend lab wordt geacht over een degelijke labvoeding te beschikken.

Even een toelichting: in het begin zijn we uitgegaan van een externe voedingsspanning van 22 V die een shuntweerstand van  $400 \Omega$  mogelijk maakte, maar deze relatief hoge niet-standaard waarde werd als een bezwaar gezien, en daarom hebben we de spanning verlaagd naar minimaal 10 V (9 V plus de spanning over D3 en T1) en niet meer dan 12 V vanwege de maximale voedingsspanning van IC8.

Merk op dat de Arduino wordt gevoed via zijn USB-poort (onafhankelijk van het gebruik van het PC-programma) of door een externe voeding die dezelfde kan zijn als die van het shield.

### De details van de schakeling

Laten we eens met de voeding beginnen (**figuur 2**). Omdat deze verplicht wordt geleverd door een labvoeding van goede kwaliteit, blijft de verwerking ervan op het board beperkt tot een regelaar (IC10) en wat condensatoren waarmee de 5 V voor de DAC en de ACD wordt geleverd. De versterkers worden direct met de ingangsspanning gevoed (er is echter wel een bescherming tegen verkeerde polariteit: D3). Het OLED-scherm wordt

apart gevoed uit de 5 V van de Arduino. Hierdoor blijven de voedingen voor het analoge en het digitale gedeelte van elkaar gescheiden.

### Digitaal/analoog-converter

IC2 is de DAC die de uitgangsspanning van de MicroSupply levert. De schakeling heeft een resolutie van 12 bit en beschikt over een interne spanningsreferentie van 2,5 V. Deze levert dus een uitgangsspanning (meetbaar op TP11 met opschrift 'DAC') van 0 tot 2,5 V in stappen van 0,61 mV ( $2,5 / (2^{12}-1)$ ). Om de maximale uitgangsspanning te verhogen naar 5 V wordt een opamp (IC8) gebruikt met een versterkingsfactor 2 (TP2, '2x'). De stapgrootte wordt hierdoor 1,22 mV. Zoals hierboven aangegeven beschikt de DAC over een interne referentiebron, maar de datasheet bevat niet veel informatie over de kwaliteit van deze referentie. Daarom hebben we IC1 toegevoegd, een goed gedocumenteerde zeer nauwkeurige referentiebron. Zo weten we beter waar we aan toe zijn. De referentiespanning is beschikbaar op testpunt TP9 ('2,5 V').

### 'Vermogens'-versterker

Het andere deel van IC8 stuurt een transistor aan die de maximale stroom van 40 mA kan leveren. Omdat het hier gaat om een opamp, zal deze zijn uitgang zodanig proberen te regelen dat de spanningen op zijn twee ingangen (pen 5 en pen 6) aan elkaar gelijk zijn. Dankzij de verbinding van de inverteerende ingang van de opamp (pen 6) met de uitgang van de MicroSupply (TP14, 'AFTER'), zorgt dit mechanisme ervoor dat de uitgangsspanning identiek is aan de spanning op de niet-inverteerende ingang (pen 5), onafhankelijk van de stroom door R1 en T1. Als de opamp snel is in vergelijking met de stroomvariaties die door de belasting worden gevraagd, zal deze laatste de indruk krijgen te worden gevoed door een bron met een erg lage uitgangsseerstand, en dat was precies ons doel. IC9 deelt de uitgangsspanning door twee (TP15, 'OUT/2') voor de weergave en activering van de stroomonderbreker-software.

### Shuntseerstand plus versterkers

R1 is de hierboven al vaak genoemde shuntseerstand. De door de belasting afgenomen stroom loopt door R1 en veroorzaakt over zijn aansluitingen (TP13,



## ONDERDELENLIJST

### Weerstanden:

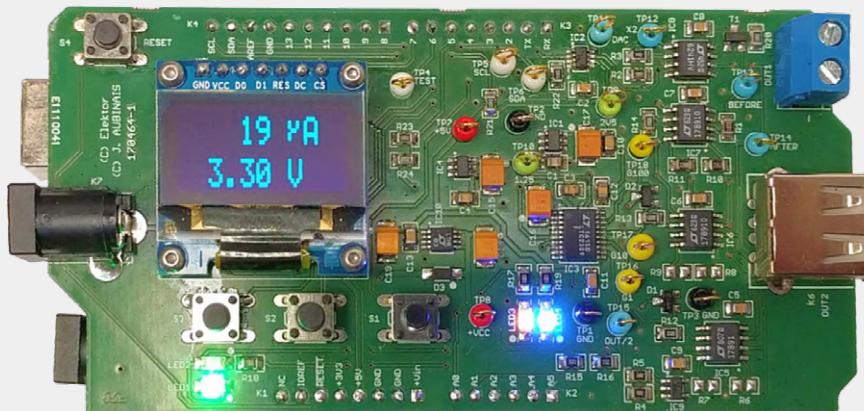
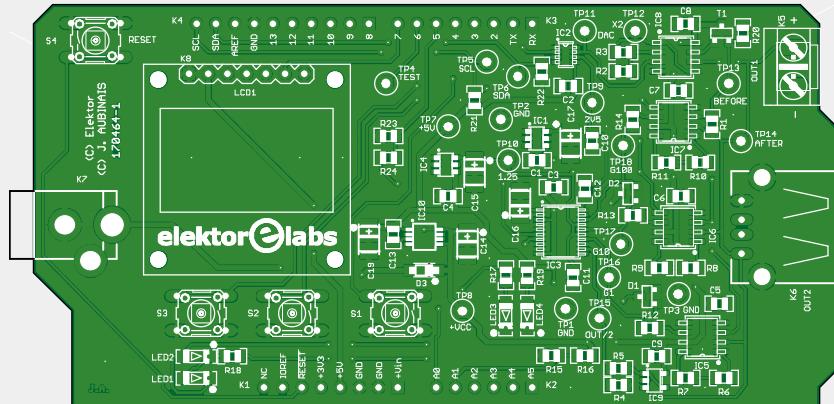
(alle SMD 0805)  
 R6,R7,R8,R9 = NC  
 R12,R13,R14 = 100 Ω  
 R1 = 100 Ω, 0,1%  
 R11 = 220 Ω, 0,1%  
 R17,R18,R19 = 1 k  
 R15 = 3 k  
 R20,R21,R22,R23 = 10 k  
 R2,R3,R4,R5 = 10 k, 0,1%  
 R16 = 12 k  
 R10 = 22 k, 0,1%  
 R24 = 22 k

### Condensatoren:

C13 = 10 n, SMD 0805  
 C1...C12 = 100 n, SMD 0805  
 C14...C18 = 10 μ/16 V, tantaal, SMD 1411

### Halfgeleiders:

D3 = 1N5819HW-7-F  
 D1,D2 = BAT54C  
 T1 = BC847C  
 LED4 = LED blauw, SMD 1206  
 LED2 = LED groen, SMD 1206  
 LED3 = LED geel, SMD 1206  
 LED1 = LED rood, SMD 1206  
 IC10 = LT1762EMS8-5  
 IC5 = LT1789CS8-1  
 IC6,IC7 = LT1789CS8-10  
 IC4 = LT1790BCS6-1.25  
 IC1 = LT1790ACS-2.5  
 IC9 = LTC2054HVCS5  
 IC3 = LTC2309CF  
 IC2 = LTC2631CTS8  
 IC8 = LTC6241HVCS8



### Diversen:

K8 = 7-polige busstrip, steek 2,54 mm voor LCD1  
 K2 = 6-polige pinheader, steek 2,54 mm  
 K1, K3 = 8-polige pinheader, steek 2,54 mm  
 K4 = 10-polige pinheader, steek 2,54 mm  
 K6 = USB-connector type A, haaks  
 K5 = 2-polige printkroonsteen, steek 5,08 mm  
 K7 = voedingsconnector,  
 centrale pen 1,95 mm

S1,S2,S3,S4 = druktoets, 6x6 mm  
 Print 170464-1 (zie kader 'In de store')  
 LCD1 = OLED-scherm, 0,96 inch, SPI,  
 7 aansluitingen  
 4x boutje, M2 x 8 mm  
 12x moertje, M2  
 Precisieweerstanden van 1 k, 100 k en 1 M,  
 0,1%, voor test en kalibratie.

'BEFORE' en TP14, 'AFTER') een spanning die we kunnen meten. Hiervoor worden drie parallelgeschakelde instrumentieversterkers gebruikt: een met een versterkingsfactor 1 (IC5), een tweede met een versterkingsfactor 10 (IC6) en de laatste met een versterkingsfactor 100 (IC7). Let bij het bestellen en monteren goed op de suffixen van de typenummers van deze onderdelen, want deze zijn niet hetzelfde. IC5 heeft suffix '-1', terwijl IC6 en IC7 '-10' als suffix hebben. Deze suffix geeft de voorgeprogrammeerde versterking aan. Met behulp van een weerstand is deze aan te passen, en dit hebben we bij IC7 met R10 en R11 gedaan om een versterking van 100x te krijgen. Het zou mogelijk zijn geweest om slechts één versterker te gebruiken en de weerstand omschakelbaar te maken voor de juiste verster-

kingsfactor, maar het was niet duidelijk of we hier uiteindelijk iets mee zouden winnen.

TP16 ('G1'), TP17 ('G10') en TP18 ('G100') zijn testpunten voor de drie versterkers. Let op, van de meetwaarden 'G10' en 'G100', moet de referentiespanning van 1,25 V (TP10) worden afgetrokken.

### Digitaliseren van de meetwaarden

De signalen aan de uitgangen van de versterkers, en het signaal dat afkomstig is van IC9, worden gedigitaliseerd door de analoog/digitaal-converter (ADC) IC3. De ingangen van deze schakeling kunnen worden gebruikt als differentiële paren of als enkelvoudige ingangen. We gebruiken ze als enkelvoudige ingangen voor de uitgangssignalen van IC9 en IC5. De

uitgangssignalen van IC6 en IC7 worden daarentegen differentieel gemeten ten opzichte van een referentiespanning van 1,25 V die afkomstig is van IC4 (TP10, '1.25'). Waarom? Omdat de uitgangen van IC6 en IC7 niet lager kunnen worden dan 110 mV, en dat is een beetje vervelend als je lage spanningen wilt meten die nauwelijks boven 0 V uitkomen. Gelukkig hebben deze schakelingen een referentie-ingang waarmee het uitgangssignaal kan worden verhoogd. Via een differentiële ingang van de ADC kunnen we vervolgens de referentiespanning van het signaal aftrekken en dit zo terugbrengen naar een meting ten opzichte van nul. We voeren deze truc niet uit bij het signaal van IC5 want dan overschrijden we bij een stroomverbruik van 40 mA de maximaal toelaatbare spanning van 4,096 V aan de uitgang van de versterker.

Net als de DAC heeft de ADC een resolutie van 12 bit. En deze heeft ook, net als de DAC, een interne referentie die we hebben vervangen door onze eigen referentie IC1. En tot slot communiceert de ADC, weer net als de DAC (ze zullen toch niet toevallig van dezelfde familie zijn?) via een I<sup>2</sup>C-bus (met de testpunten TP5, 'SCL' en TP6, 'SDA').

### Display met SPI-bus

We zijn nu aangekomen bij het OLED-scherm dat communiceert via de SPI-bus. Dit type display bestaat ook met een I<sup>2</sup>C-interface, maar we wilden de communicatiebussen gescheiden houden, temeer daar de SPI-bus meer geschikt is voor grafische toepassingen omdat deze hoge data-transmissiesnelheden mogelijk maakt.

### En de microcontroller?

Hier eindigt onze rondreis door het schema van de MicroSupply, we hebben alles gezien. Echt waar? Welnee, er ontbreekt een beschrijving van de microcontroller, het onderdeel dat onze schakeling tot leven wekt! U hebt het al geraden (en wij hebben het ook niet echt verborgen gehouden) het betreft een Arduino Uno-board waarop zich een ATmega328 bevindt.

Naast de besturing van ons shield, die onder andere bestaat uit het aansturen van de DAC, ophalen van de meetwaarden en het weergeven van de gegevens op het OLED-scherm, houdt de µC zich ook bezig met de communicatie met een PC via een seriële USB-poort.

Om er zeker van te zijn dat de voedingsspanning van het shield voldoende is om de belasting te voeden, meet de µC de ingangsspanning via zijn analoge ingang A1. De controller gaat niet verder met de initialisatie zolang de ingangsspanning lager is dan 10 V.

De ingangsspanning wordt ook gemeten via D5 van de Uno, maar minder nauwkeurig. Als de µC een laag logisch niveau meet op deze ingang, dan is de ingangsspanning lager dan ongeveer 9,6 V en stopt het systeem.

De uitgangsspanning wordt door de µC gemonitord (met behulp van IC9) om te controleren of deze niet lager wordt dan 100 mV ( ingesteld in het programma). Als dit gebeurt wordt de uitgang onmiddellijk afgeschakeld en wordt een foutmelding weergegeven.

### Een paar woorden over de software

De Arduino-sketch is op zich niet ingewikkeld, maar er zijn een paar coderegels die uw aandacht verdienen:

• **Ontdendering van de toetsen.** Met de druktoetsen S2 en S3 ('Down' en 'Up') kan de uitgangsspanning in stappen van 0,05 V worden ingesteld tussen 1,5 en 5 V. De toetsen staan onder controle van de (enigszins aangepaste) bibliotheek van Jack Christensen [3], die contactdender elimineert en waarmee de toename of afname van de spanning kan worden versneld door de toets ingedrukt te houden.

• **De 1ms-interrupts (1 kHz),** oftewel de tijd tussen de metingen. Bij de eerste tests werd de schakeling blootgesteld aan een 'foutieve' belasting, gerealiseerd met een sinusvormig signaal van 1 Hz, beschermd door weerstanden. Omdat 1 kHz een exact veelvoud is van 1 Hz, verwachtten we een perfect stabiel meetsignaal. Maar de weergave op de PC liet een sinus zien die langzaam verschafte. Na wat onderzoek bleek dat de periode van de interrupt in werkelijkheid 1,024 ms was. Dat komt door de Arduino die dezelfde teller gebruikt om pulsbreedtemodulatiesignalen (PWM) te produceren. Om dit te corrigeren moet het register OCROA op 249 worden ingesteld: 16.000.000 Hz (de kristalfrequentie) / 64 (interne deler) / (249+1) = 1.000 Hz.

• **De lettertypes.** Bij veel bibliotheken die op internet beschikbaar zijn hebben we een nieuw lettertype moeten creëren om de weergave op het OLED-scherm te verbeteren.

• **De I<sup>2</sup>C-bibliotheek.** Omdat bij dit project alleen IC's zijn gebruikt van Analog Devices (Linear Technology), hebben we gebruik gemaakt van de bibliotheek 'LT\_I2C' [5].

### De functie loop

Zoals zijn naam al aangeeft laat de functie *loop* de µC continu 'rondjes draaien'. De functie heeft drie modi:

• **Stand-by.** Deze modus wordt geactiveerd en gedeactiveerd met S1 (de meest rechtse toets). In deze modus kan de uitgangsspanning met S2 en S3

(onder het scherm) worden ingesteld.

• **Local.** De meetwaarden worden wel op het OLED-scherm weergegeven, maar niet verzonden via de seriële poort. Als het PC-programma het karakter 'E' over de seriële poort stuurt wordt deze modus geactiveerd.

• **Remote.** De meetwaarden worden wel via de seriële poort verzonden maar niet op het OLED-scherm weergegeven (Op het OLED-scherm staat: 'SERIAL'). Als het PC-programma het karakter 'B' via de seriële poort verstuurdt wordt deze modus geactiveerd.

### Geïntegreerde autodiagnose

Als tijdens het opstarten van de Arduino een van de toetsen wordt ingedrukt, wordt een klein testprogramma voor het shield gestart. Sluit nu na elkaar een weerstand van 100 kΩ (0,1%) en een weerstand van 1 kΩ (0,1%) aan op de uitgang, en volg de instructies op het OLED-scherm.

Het shield kan ook worden gekalibreerd. Om de procedure hiervoor te starten moeten tijdens het opstarten van de Arduino tegelijkertijd de toetsen S2 en S3 ('Down' en 'Up') worden ingedrukt. Sluit een weerstand van 1 MΩ (0,1%) aan op de uitgang, en volg de instructies op het display.

### PC-software

Omdat de MicroSupply de meetwaarden met een frequentie van 1 kHz verzendt, moet het PC-programma deze grafisch in realtime kunnen weergeven. De auteur, die voornamelijk in C# programmeert, heeft eerst geprobeerd om dit met Chart-Control te realiseren, maar heterversen van de scherminhoud en de automatische schaalaanpassing verliepen te langzaam. Na veel onderzoek heeft hij de Nebula-bibliotheek gevonden. Deze bibliotheek voldoet prima aan de eisen, maar vereist ontwikkeling in Eclipse/Java. De eerste tests waren overtuigend, behalve wat de seriële communicatie met de Arduino betreft. Hoe nu verder, zonder te kiezen voor een niet-gratis oplossing? Wel, met IKVM [6], een virtuele Java-machine (JVM) voor het .NET-framework, oftewel Visual Studio. Het tool IKVM.NET (ontwikkeld door Jeroen Frijters) maakt het mogelijk om een databasetoepassing van Java naar .NET te migreren. Omdat Elektor geen informatijschrift is, besparen we u de gedetailleerde beschrijving

van het gebruik van dit uitstekende tool, maar het komt er kort gezegd op neer dat dat dit een DLL produceert, geschreven in Java (*DLL MicroSupply\_JA.dll*) die eenvoudig met C# in Visual Studio kan worden beheerd. Onze PC-applicatie gebruikt dus twee verschillende programmeertalen.

De applicatie slaat 4000 meetwaarden op, dat is vier seconden aan metingen, die in delen van 1, 2 of 4 seconden kunnen worden weergegeven. Let op: het programma geeft niet meer dan 1000 punten weer. Bij de keuze voor 2 of 4 seconden worden er dus sommige waarden niet weergegeven en dit kan leiden tot fouten bij de beoordeling van de goede werking van uw IoT-apparaat.

#### 64-bit Windows 10 vereist

Om het programma op uw PC te draaien hebt u de 64-bits versie van Windows 10 nodig, en moet bovendien Java (64 bits) versie 8, update 191 of hoger zijn geïnstalleerd. Download het archiefbestand van [1] of [2] en pak het uit in een map naar keuze. U vindt daar vervolgens:

- het programma *MicroSupply.exe*;
- de twee DLL's *IKVM.OpenJDK.Core.dll* en *ikvm-native-win32-x64.dll*;
- de DLL's van de functie SWT (map: 'filesSWT' van de download [1]) met de mapnamen '.swt\lib\win32\x86\_64' moeten worden gekopieerd naar de map 'C:\Users\YourUserName';
- de DLL *MicroSupply\_JA.dll* (deze hoeft niet te worden gekopieerd want deze is samen met veel andere IKVM-DLL's geïntegreerd in het programma *MicroSupply.exe*).

Het verdient aanbeveling om het programma de eerste keer via een commandoregel met het argument 'debug' op te starten. Hierbij wordt namelijk veel informatie weergegeven die handig kan zijn voor het oplossen van problemen met de configuratie van uw PC bij het uitvoeren van het programma.

#### Gebruikshandleiding

Hier volgt een beknopt overzicht van de functies van de verschillende buttons en sliders in de PC-applicatie:

- 'Refresh': ververs de lijst met seriële poorten van de PC.
- 'Max Value' en 'Min Value': maximale en minimale gemeten waarde van de stroomsterkte gedurende de

## LED's en druktoetsen

- LED1 (groen): het IoT-apparaat krijgt voedingsspanning.
- LED2 (rood): de software-schakelaar is geactiveerd, het IoT-apparaat is afgeschakeld.
- LED3 (geel): knippert als het IoT-apparaat voedingsspanning krijgt. Bij een programmatoring is de LED continu aan of uit.
- LED4 (blauw): het shield krijgt externe voedingsspanning. (Dat wil niet zeggen dat deze spanning hoog genoeg is.)
- S1: activeren/deactiveren van de uitgang.
- S2: Up / Next / Yes
- S3: Down / No
- S4: Reset.

aangegeven seconde.

- '1 second', '2 seconds' en '4 seconds': weergavemodus met meetgegevens over 1, 2 of 4 seconden.
- Lange slider (van 0 tot 9): hiermee kan een meetwaarde-periode voor weergave worden gekozen (maximaal 4 s). In de modus '1 second' in stappen van 0,25 s; In de modus '2 seconds', is de stapgrootte 0,5 s.
- Korte slider (van 0 tot 1 of van 0 tot 3): selectie van de weer te geven opgeslagen meetwaarden. In de modus '2 seconds', kan met de slider één van twee metingen worden gekozen, de eerste of de tweede. In de modus '4 seconds', is het één van vier metingen: de eerste, de tweede, de derde of de vierde.
- 'Save': opslaan van alle meetwaar-

den in een CSV-bestand (waarden in  $\mu$ A), maximaal 4.000 waarden.

## Conclusie

De MicroSupply kan een IoT-apparaat met maximaal 5 V voeden en een maximale stroom van 40 mA leveren. Tegelijkertijd meet hij de stroom die de belasting verbruikt en geeft deze weer op het OLED-display. De gemeten waarden kunnen ook via een seriële poort worden verzonden, bijvoorbeeld naar een PC. Als de ontvanger een PC is die draait onder Windows 10, dan kan een speciaal hiervoor ontwikkelde applicatie de gegevens in de vorm van een grafiek zichtbaar maken. Dit is dus een klein tool dat erg handig is om het verbruik van uw (huidige of toekomstige) IoT-apparaten te controleren. ▶

(170464-04)

## Links en literatuur

- [1] Projectpagina bij dit artikel: [www.elektormagazine.nl/170464-04](http://www.elektormagazine.nl/170464-04)
- [2] Broncode: <https://github.com/jenniferaubinais/MicroSupply>
- [3] Button-library: <https://github.com/JChristensen/Button>
- [4] Boek (Engelstalig) 'Mastering microcontrollers helped by Arduino': [www.elektor.nl/mastering-microcontrollers-helped-by-arduino-edition-3](http://www.elektor.nl/mastering-microcontrollers-helped-by-arduino-edition-3)
- [5] LT\_I2C & Linduino : <https://github.com/analogdevicesinc/Linduino>
- [6] IKVM : <https://www.ikvm.net/>



### IN DE STORE

→ Kale print MicroSupply 170464-1  
[www.elektor.nl/18954](http://www.elektor.nl/18954)

→ Arduino Uno R3  
[www.elektor.nl/arduino-uno-r3](http://www.elektor.nl/arduino-uno-r3)

→ Boek (Engelstalig) 'Mastering microcontrollers helped by Arduino' (3e druk):  
[www.elektor.nl/mastering-microcontrollers-helped-by-arduino-edition-3](http://www.elektor.nl/mastering-microcontrollers-helped-by-arduino-edition-3)

# Motorola MC14500

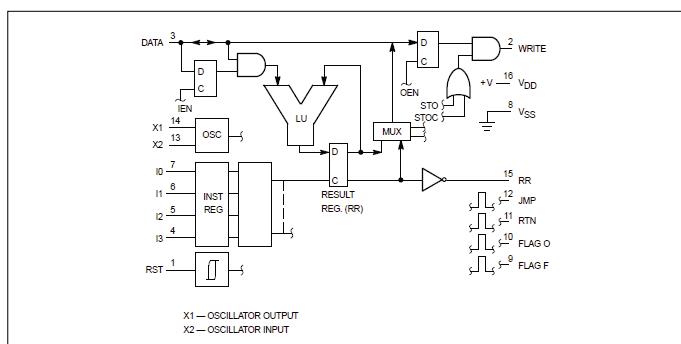
## Industrial Control Unit

### vreemd onderdeel

Neil Gruending (Canada)

De relaislogica die lange tijd in de industrie werd gebruikt voor het besturen van machines, is op een gegeven moment vervangen door discrete digitale logica, maar was nog steeds moeilijk aan te passen en te onderhouden. Er was behoefte aan een programmeerbare oplossing, maar computers waren aan het eind van de jaren '70 nog duur en ingewikkeld. Daarom introduceerde Motorola de industriële besturingseenheid (*Industrial Control Unit*) MC14500. Deze component heeft een interessante 1-bit-architectuur en kan met recht een *vreemd onderdeel* worden genoemd. De MC14500 kwam op de markt in 1977 en was ontworpen voor het implementeren van ladderlogica die werd gebruikt in *Programmable Logic Controllers* (PLC's). Het interessante ontwerp bevat niet meer dan nodig is voor het implementeren van 16 instructies met een 1-bit-register (zie **figuur 1**) [1]. De datalijn kan een ingang of een uitgang zijn, afhankelijk van de instructie die op dat moment wordt uitgevoerd; daarnaast zijn er speciale JMP/RTN-lijnen voor het overslaan van instructies. Er zijn ook instructies voor eenenvoudige logische operaties tussen het dataregister en de data in- en uitvoer. Er zijn geen numerieke instructies geïmplementeerd, want daar is meer dan 1 bit voor nodig. En hoe zit het met het ontbreken van een programmateller en een geheugen?

Dat werd opgelost met een externe programmateller en geheugens zoals in **figuur 2** [1]. De teller houdt het actieve geheugenadres bij dat de instructie en andere besturingssignalen bevat voor bijvoorbeeld data-latches (zoals afgebeeld), afhankelijk van wat de toepassing vereist. Met behulp van die latches kan de MC14500 zoveel in- en uitgangen besturen als nodig is, steeds met één bit tegelijk.



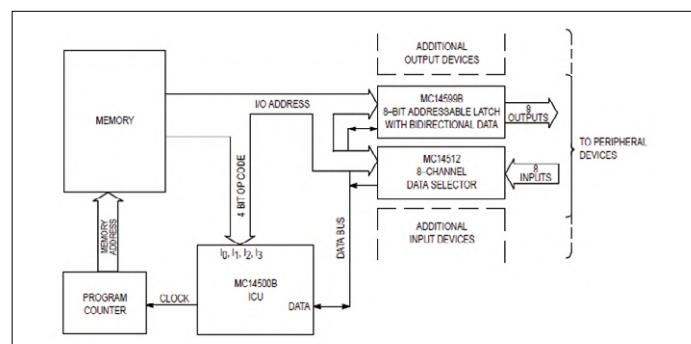
Figuur 1: Blokschema van de MC14500 [1].



Foto: JPL, eigen werk, CC BY-SA 4.0,  
<https://commons.wikimedia.org/w/index.php?curid=48560865>

Het blokschema van figuur 2 zou er bekend uit moeten zien, want het bevat alle basisbouwstenen voor een eenvoudige discreet opgebouwde computer. Het bedrijf DATANorf Hard and Software in Duitsland maakte en verkocht zelfs de WDR-1 training-kit voor educatief gebruik, die een MC14500 gebruikte als een eenvoudige CPU [2]. De kit werkte met LED's als uitgangen en had een toetsenbord voor het invoeren van instructies. Het is zelfs mogelijk om een MC14500 te implementeren met behulp van standaardlogica, zoals Dieter Mueller heeft gedaan [3]. De MC14500 werd gedurende ongeveer twintig jaar geproduceerd door Motorola en er is veel informatie over beschikbaar, inclusief de datasheet en het toepassingshandboek. Het zou niet zo moeilijk hoeven te zijn om er een te vinden op de tweedehandse markt, of misschien wilt u wel proberen om er zelf een te implementeren met programmeerbare logica. Hoe dan ook, een 1-bit-processor is beslist een *vreemd onderdeel!* ■

(180575-E-03)



Figuur 2: Blokschematische schakeling met een MC14500 [1].

#### Weblinks

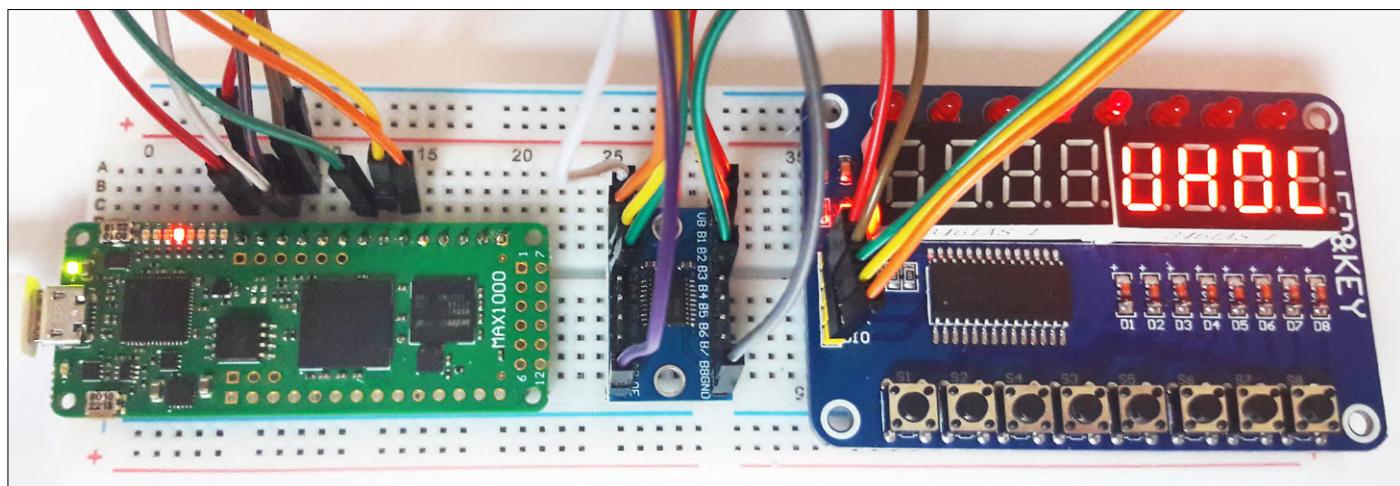
- [1] Schema's met de MC14500: [www.brouhaha.com/~eric/retrocomputing/motorola/mc14500b/mc14500brev3.pdf](http://www.brouhaha.com/~eric/retrocomputing/motorola/mc14500b/mc14500brev3.pdf)
- [2] WDR-1 training educational kit: [www.old-computers.com/museum/computer.asp?st=1&c=834](http://www.old-computers.com/museum/computer.asp?st=1&c=834)
- [3] Dieter Mueller's website: [www.6502.org/users/dieter/m14500/m14500.htm](http://www.6502.org/users/dieter/m14500/m14500.htm)

# Hardware-ontwerp met (V)HDL (4)

## reactietijdmeter

Jörg Zollmann (Duitsland)

In het vorige deel van deze serie [1] hebben we een 8x8 LED-matrix aangestuurd om de tijd weer te geven. Deze keer gebruiken we een 7-segment-module als display. Maar in plaats van een klok implementeren we een wat complexer algoritme en programmeren we een reactietijdmeter.



Figuur 1. Opbouw van het reactiespel op een breadboard.

Zoals gebruikelijk kan de broncode van dit project van de projectpagina bij dit artikel worden gedownload [2]. Als hardware-platform voor de VHDL-demo kan zowel het bekende CPLD-board worden gebruikt als het MAX1000-platform van het SCCC-project (zie het kader **In de store**). Daarnaast gebruiken we een uitbreidingsboard met een TM1638-chip die de aansturing van de acht LED's en drukknoppen en een achtvoedige 7-segment-module voor zijn rekening neemt [3] (**figuur 1**). De communicatie tussen de FPGA en de TM1638 verloopt via een serieel protocol. Omdat de FPGA op 3,3 V draait en de TM6138 op 5 V, is een niveau-omzetter (Txs0108E) nodig (**figuur 2**).

### Spelbeschrijving

We bespreken stap voor stap de voor het reactietijdspeel benodigde hardware-elementen en zullen kort stilstaan bij de – tot nog toe buiten beschouwing gebleven maar voor elke digitale schakeling noodzakelijke – timing-analyse.

Het spel bestaat uit meerdere functieblokken, die in verschillende *entities* en *packages* zijn onderverdeeld. Deze functies zijn: het spel-algoritme, een looplicht, de display-aansturing en bijbehorende display-driver (SPI-master), een toevalsgenerator, een tijdvertraging en een stopwatch. Het reactietijdspeel

wordt gestart met een druk op de knop KEY0 op het uitbreidingsboard. Dan wordt het looplicht op het board actief. Na een toevallige tijd stopt het looplicht en begint de reactietijdmeting. De speler moet nu zo snel mogelijk de knop indrukken die correspondeert met de plaats waar het looplicht is gestopt. Wanneer bijvoorbeeld LED3 continu oplicht, dan moet de speler om de stopwatch te stoppen KEY3 indrukken. Om een nieuwe poging te starten, moet opnieuw KEY0 worden ingedrukt.

### Stopwatch en looplicht

De stopwatch bestaat uit meerdere tellers die we al uit de vorige delen van deze VHDL-serie kennen. De configuratie van de tellers gebeurt in het package `count_pkg`. Eén teller wordt zo geconfigureerd dat hij als vórdeler fungiert. De in te stellen deelfactor is afhankelijk van het toegepaste board. Naast deze *prescaler* is er nog een teller voor de honderdsten seconden (`csec`), een voor de tienden seconden (`dsec`) alsmede twee secondentellers. Omdat de tellerwaarden van de stopwatch op het 7-segment-display moeten worden getoond, zijn de tellers als BCD-tellers ingericht. Elke teller bevat zijn voor de betreffende digit benodigde gewicht en telt steeds van 0 tot 9. Als bijvoorbeeld de honderdste-secondenteller

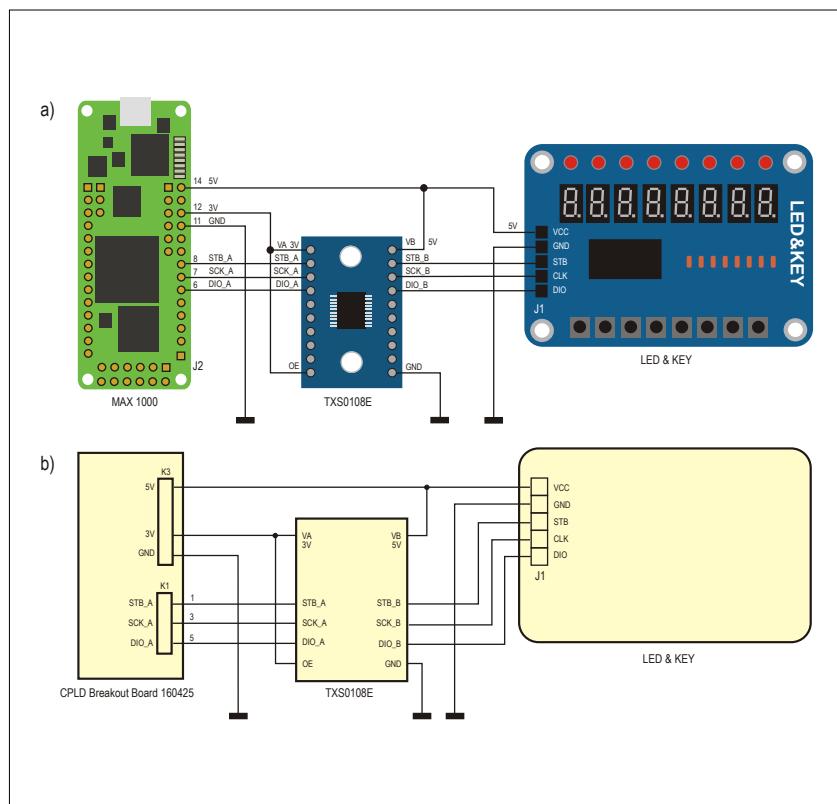
van 0 tot 9 heeft geteld, wordt de overlooppuls als klokvrijgavepuls (enable) gebruikt voor de tiende-secondenteller, zodat deze 1 verder telt. Om ervoor te zorgen dat de stopwatch als zodanig herkenbaar is, mag die noch te snel, noch te langzaam van toestand veranderen. Dit bereiken we door de toestand van het looplicht slechts elke 10 ms te wijzigen. De hiervoor benodigde time-tick (enable) wordt uit de stopwatch aan de daaropvolgende schakeling doorgegeven met het signaal `t10ms`. Het proces `led-chaser` geeft de functie van het looplicht weer (**listing 1**). Daarbij worden variabelen gebruikt.

In VHDL kunnen variabelen worden gebruikt zolang het signaal alleen binnen het eigen proces wordt gebruikt. Aan variabelen kan een waarde worden toegewezen met '`:=`' en niet zoals bij signalen met '`<=`'. Variabelen nemen de toegewezen waarde direct over, terwijl dit bij signalen (`signal`) alleen dan het geval is als met het signaal een combinatorisch deel wordt beschreven. Anders nemen signalen de toegewezen waarde pas een klokcyclus later over. In het looplicht-proces wordt de LED die direct moet worden aangestuurd, in een unsigned-signal `led_r` opgeslagen. De variabelen `dir` en `pos_led` bepalen of de volgende LED links of rechts van de huidige LED moet oplichten. Om het verschil tussen een variabele en een signaal te leren kennen, kunt u hier gewoon eens uitproberen wat er gebeurt als variabelen door signalen worden vervangen.

### Toevalsgenerator

Om te voorkomen dat het looplicht telkens op dezelfde plaats tot stilstand komt, wordt de duur van de looplicht-sequentie aan het toeval overgelaten. Daarvoor zijn twee componenten nodig. In de eerste plaats een tijdvertraging, die telkens als het spel opnieuw wordt gestart meegedeeld krijgt tot waar hij moet tellen, en in de tweede plaats een toevalsgenerator. Als een nieuwe reactietijdmeting wordt gestart, wordt het actuele toevalsgetal als preset-waarde aan de teller doorgegeven en wordt het reset-signaal vrijgegeven. In tegenstelling tot de tot nu toe gebruikte tellers heeft deze teller een synchrone reset en krijgt de entity de `reset_val` via een ingangspoort aangereikt. Het toevalsgetal wordt met een PRBS-generator (Pseudo Random Binary Sequence) gegenereerd. Het gaat hier om een teruggekoppelde 7-bit schuifregister (**listing 2**). Het schuifregister schuift de data aan de ingang (bit 0) steeds 1 bit verder. De data aan de ingang bestaat uit het resultaat van de terugkoppeling: een exclusive-OR van het MSB van het schuifregister (bit 6) met bit 5. De aftakkingen worden ook wel generatorpolynoom genoemd. De PRBS-generator kan maximaal  $2^n - 1$  verschillende bit-sequenties genereren. In onze reactietijdmeting zijn hiermee dus 127 verschillende vertragingen mogelijk.

Op deze plaats hebben we mooi de gelegenheid in te gaan op de implementatie van een schuifregister. Een variant daarvan is in twee opdrachten eerst de 'oude' waarde op te slaan en dan de nieuwe waarde aan het eind van het schuifregister te bepalen. In listing 2 en in SPI-master wordt deze methode



Figuur 2. De niveau-omzetter zet de 3,3V-signalen van de FPGA om naar 5 V.

### Listing 1. Looplicht.

```
led_chaser: process (rst, clk)
variable led_pos : unsigned(3 downto 0);
variable dir      : bit;
begin
  if (rst = '1') then
    led_r           <= 8B"0000_0001";
    led_pos         := (others => '0');
    dir             := '0';
  elsif rising_edge(clk) then
    if (t10ms) then
      if (run_chaser = '1') then
        if led_pos = 7 then
          dir := '1';
        elsif led_pos = 0 then
          dir := '0';
        end if;
        if dir = '0' then
          led_r   <= shift_left(unsigned(led_r), 1);
          led_pos := led_pos +1;
        else
          led_r   <= shift_right(unsigned(led_r), 1);
          led_pos := led_pos -1;
        end if;
      end if;
    end if;
  end process;
```

gebruikt. VHDL biedt ook de mogelijkheid, een schuifregister met behulp van functies uit de de *IEEE.NUMERIC.STD*-bibliotheek te realiseren. Hierin zijn de functies *shift\_left* en *shift\_right* gedefinieerd. Aan deze functies kan als eerste parameter een signed of een unsigned waarde worden doorgegeven; als

#### **Listing 2. PRBS-generator.**

```
prbs_lfsr: process (rst, clk)
begin
  if (rst = '1') then
    d(6 downto 1) <= (others => '0');
    d(0)           <= '1';
  elsif rising_edge(clk) then
    d(6 downto 1)  <= d(5 downto 0);
    d(0)           <= d(5) xor d(6);
  end if;
end process;
```

#### **Listing 3. De functie rdata2slv.**

```
function rdata2slv(
  rd : in std_logic_vector(31 downto 0)
)
return std_logic_vector is
  variable v_keys   : std_logic_vector(7 downto 0);
begin
  v_keys := rd(17)& rd(21) & rd(25)
& rd(29) & rd(19) & rd(23)& rd(27)
&           rd(31);
  return std_logic_vector(v_keys);
end;
```

#### **Listing 4. Genereren van SPI-Clock.**

```
-- f_ena_int = 2 x f_sck
-- use sck_r1 in non delayed version -> CPHA = 0
-- use sck_r2 to delay sck --> CPHA = 1
-- use CPOL=1 to invert sck (switchable inverter)

sck  <= ((sck_r1 and not CPHA) or (sck_r2 and
CPHA)) xor CPOL;
sck_proc : process (clk,rst) is
begin
  if (rst = '1') then
    sck_r1 <= '0';
    sck_r2 <= '0';
  elsif rising_edge(clk) then
    if (ena_int) then
      if (frame_enable)then
        sck_r1  <= (not sck_r1);
        sck_r2  <= (sck_r1);
      end if;
    end if;
  end if;
end process sck_proc;
```

tweede parameter wordt het (integer) aantal stappen opgegeven waarmee moet worden geschoven. Afhankelijk van de vraag of de eerste parameter een een signed of een unsigned waarde is, wordt met nullen aangevuld (logische shift) of is het resultaat een signed waarde (rekenkundige shift).

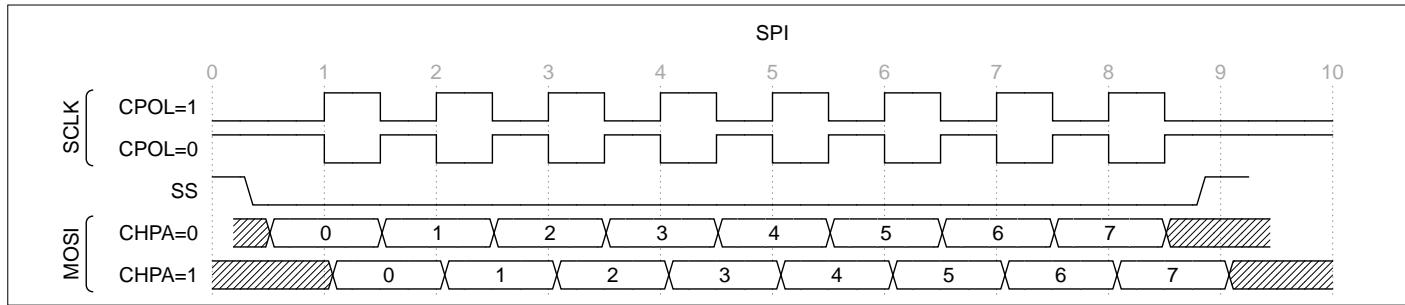
#### **LED-driver-IC**

Het toegepaste LED&KEY-board wordt aangestuurd door een TM1638-driver-IC en is bij Amazon of in China voor ca. € 5 verkrijgbaar. Het IC komt van de Chinese firma Titan Micro, die een hele serie LED/LCD-driver-IC's aanbiedt. De datasheet spreekt nou niet direct voor zich, wat ten dele te wijten is aan de slechte vertaling uit het Chinees. Omdat het board zeer populair is in het Arduino-wereldje, zijn er gelukkig op internet talrijke blog-bijdragen te vinden waarin de werking van het board veel beter wordt uitgelegd [4][5][6]. De TM1638 bestaat uit 16 interne 8 en 2 bit brede beschrijfbare registers. Daarnaast bezit de chip een 4 byte groot intern geheugen die kan worden uitgelezen. Om een LED of een 7-segment-display te laten oplichten moet eerst met een separaat commando de helderheid worden ingesteld en vervolgens de wijze waarop de interne registers zullen worden geadresseerd. Aansluitend moeten de gewenste waarden in de registers worden geschreven en moet tot slot het display worden ingeschakeld.

De waarden in de 16 beschrijfbare registers worden door de TM1638 continu door middel van tijdmultiplexen naar buiten gevoerd. Van de 16 registers worden altijd een 8-bits en een 2-bits register in een dubbelregister samengenomen. Tijdens de uitvoer worden de dubbele registers om beurten geadresseerd en worden de registergegevens van het op dat moment actieve 'raster' op de databus gezet. Daarbij wordt de betreffende grid-pin laag gemaakt. De duur van de grid-puls bepaalt zo de helderheid van de aangesloten LED's; dit kan zoals al opgemerkt separaat worden ingesteld. Aan het einde van de uitvoerfase volgt de keyscan-fase voor het afvragen van de drukknoppen. Daarbij wordt achtereenvolgens op de K-aansluitingen van de chip een korte hoog-puls gezet. De keyscan-lijnen zijn aan het andere uiteinde via de drukknoppen en een beschermingsdiode met de SEG-pinnen verbonden. Als nu tijdens het scannen een knop wordt ingedrukt, wordt in één van de vier leesbare registers een 1 opgeslagen.

#### **Functies**

Voor het aansturen van het LED&KEY-board hebben we een eenvoudige *toestandsmachine* geïmplementeerd in het *ctrl\_fsm*-proces. Deze geeft zijn commando's, die in het package *tm1638\_pkg* als constanten zijn gedeclareerd, door aan de onderliggende hardware-driver (SPI-master). Als eerste wordt de bedrijfsmodus van de TM1638 gewijzigd in directe adressering, waarbij tevens de helderheid wordt ingesteld. Vervolgens worden de afzonderlijke segmenten geadresseerd en wordt de waarde voor de weergave van het betreffende cijfer overgedragen, waarna de LED's beurtelings worden in- of uitgeschakeld. Dat gebeurt afhankelijk van de waarde van het signaal *led\_r*, waarvan de toestand in het looplicht-proces wordt gewijzigd. Tot slot worden nog de keyscan-registers uitgelezen en de gelezen waarden in het interne signaal *rdata* gekopieerd. Om de van de TM1638 teruggelezen 4 bytes weer terug te kunnen herleiden naar de knop die werd ingedrukt, wordt de functie (*function*) *rdata2slv* gebruikt. Om uit de BCD-getallen van de stopwatch de 7-segment-signalen te destilleren, wordt de



Figuur 3. SPI-timing.

functie `f_bcd_2_segment` aangeroepen. Zoals alle programmeertalen, maken ook VHDL-functies hergebruik van codedelen mogelijk, wat resulteert in compactere, leesbare code. Functies worden net als in C met hun naam aangeroepen, gevolgd door haakjes, waarbinnen de optionele parameters staan. Functies kunnen alleen een retourwaarde bevatten. De regel

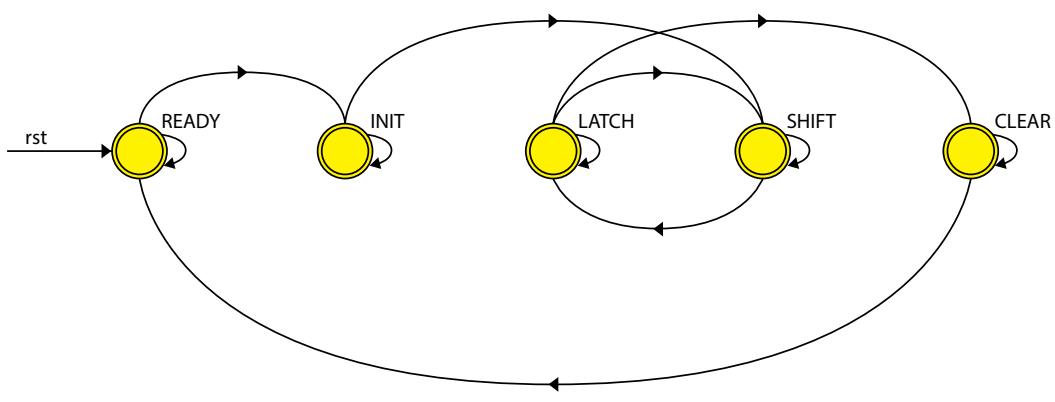
```
key_r <= unsigned(rdata2slv(rdata));
```

roept de functie `rdat2slv` (**listing 3**) aan, converteert de waarde van het resultaat (`std_logic_vector`) naar een `unsigned` waarde en wijst die toe aan `key_r`. De conversie naar een `unsigned` waarde is nodig om de vergelijking met de actuele LED-waarde, die is opgeslagen in het `unsigned` signaal `led_r`, mogelijk te maken. Beide typen `std_logic_vector` en `unsigned` zijn *unconstrained*-arrays van het type `std_logic`. Als alternatief hadden we voor de vergelijking van `led_r` en `key_r` ook een conversie van `led_r` naar een `std_logic_vector` met behulp van de functie `std_logic_vector` (`led_r`) kunnen uitvoeren.

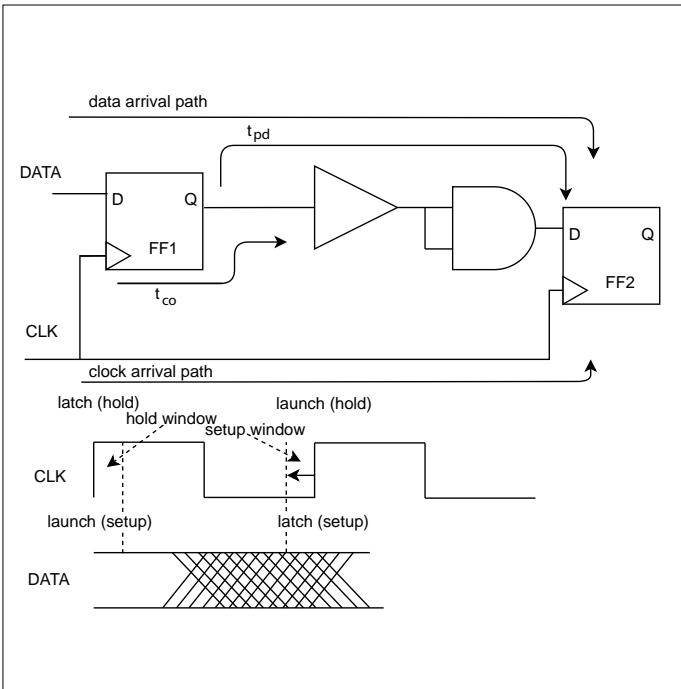
### SPI-master

De TM1638 wordt met drie signalen aangestuurd: een kloksignaal, een strobesignaal en een bidirectioneel datasignaal. Uit de datasheet van de TM1638 blijkt dat het protocol veel weg heeft van SPI, met het verschil dat de data via een bidirectionele lijn worden overgedragen. Technisch is de SPI-standaard heel eenvoudig gehouden. Die specificeert alleen de data-uitwisseling

op het laagste niveau, dus de uitwisseling van een byte tussen een master- en een slave-component. De SPI-standaard bevat geen elektrische specificatie (bijvoorbeeld van de spanningsbereiken). Normaal gesproken zijn er twee stuursignalen (SCK) en de twee datasignalen MOSI (Master Out Slave In) en MISO (Master In Slave Out). Maar er zijn ook componenten zoals de TM1638 die de datasignalen MOSI en MISO samenvoegen tot één bidirectioneel (open-drain) signaal. SPI is een synchroon serieel protocol, waarbij de data-uitwisseling wordt gestuurd via het door de master gegenereerde kloksignaal SCK. Vanwege de ruime specificatie zijn er vier ‘modi’ die verschillen in de rustpolariteit en fase van het SCK-singaal ten opzichte van de data (**figuur 3**). Uit de datasheet van de TM1638 volgt dat de communicatie moet plaatsvinden in de modus CPHA = 1 en CPOL = 0. De SPI-master is in dit project heel algemeen gehouden en kan daardoor ook in andere projecten worden gebruikt. Hij bestaat uit vier belangrijke functiegebieden. De frequentiedeler in het proces `ckdiv_proc` deelt de systeemklok en genereert twee enablepulsen per SPI-klokcyclus. Het genereren van SCK en het schuifregister FSM (`ctrl_fsm`) leiden hun acties uit dit signaal af. **Figuur 4** (`SPI_master_ctrl_fsm`) toont het toestandsdiagram van `ctrl_fsm`. De toestandsmachine stuurt in essentie twee schuifregisters aan. Bij elke enable-puls schuift de toestandsmachine één bit vanuit het TX-schuifregister naar de datalijn, of ontvangt hij een bit op de datalijn en schuift deze in het RX-schuifregister. Hoeveel bits in totaal moeten worden verzonden, wordt bepaald door het ingangssignaal



Figuur 4. Toestandsdiagram van de *toestandsmachine* voor de SPI-aansturing.



Figuur 5. Voorbeeld van een signaalpad in ons ontwerp en het daarbij horende timing-diagram.

### Listing 5. DIO.

```
-- Output enable generation for DIO
-----
din    <= dio  when d_oe = '0' else '0';
dio    <= dout when d_oe = '1' else 'Z';

doe_proc : process (clk,rst) is
variable rx_bit : integer := 0;
variable send_bits : integer := 0;
begin
if (rst = '1') then
  d_oe      <= '0';
elsif rising_edge(clk) then
  rx_bit    := to_integer(unsigned(first_rx_
byte)& "000");
  send_bits := to_integer(unsigned(frame_bytes)&
"000");
  if (ena_int) then
    if (fsm_state = READY) then
      d_oe      <= '0';
    elsif (fsm_state = INIT) then
      d_oe      <= '1';
    elsif (to_integer(send_bits-bit_cnt) = rx_
bit and (fsm_state = SHIFT)) then
      d_oe      <= '0';
    end if;
  end if;
end process doe_proc;
```

**frame\_bytes**. Het signaal `first_rx_byte` bepaalt na hoeveel bytes de datarichting omschakelt van verzenden naar ontvangen. De eigenlijke opwekking van SCK wordt verzorgd door het proces `sck_proc` (**listing 4**). Dit bestaat uit een toggle-flipflop (`sck_r1`), die bij elke enable-puls van toestand verandert, en een tweede flipflop (`sck_r2`) die de waarde van de toggle-flipflop opslaat en zo één klokcyclus vertraagt. Het SPI-kloksignaal aan de uitgang van de SPI-master (`sck`) is óf de uitgang van `sck_r1` ( $\text{CPHA}=0$ ) óf van `sck_r2` ( $\text{CPHA}=1$ ). Met het signaal `CPOL` kan de gekozen uitgang nog worden geïnverteerd. Het laatste belangrijke functieblok regelt de omschakeling van het data-signaal in het proces `doe_proc`. Dit proces zorgt ervoor, dat – afhankelijk van de toestand van de toestandsmachine en van het aantal bits dat moet worden verzonden – met behulp van het interne signaal `d_oe` de datarichting van de pin DIO wordt omgeschakeld (**listing 5**).

### TimeQuest timing-analyse

Voordat Quartus uit de VHDL-code een bedrijfsklare digitale schakeling kan ontwerpen, moet hij naast de functionele beschrijving die VHDL levert nog andere ontwerpdoelen meekrijgen. Die worden opgeslagen in een separaat \*.sdc-bestand. SDC staat voor *Synopsys Design Constraints* en is een TCL-gebaseerde industiestandaard. De belangrijkste constraints daarbij hebben betrekking op de timing (naast de pintoewijzing, die bij Quartus in een \*.qsf-bestand staat). **Figuur 5** toont een vereenvoudigd voorbeeld van het pad van een signaal binnen ons ontwerp en het bijhorende timing-diagram. FF1 en FF2 zouden bijvoorbeeld twee registers kunnen zijn die een deel van de toestanden van onze toestandsmachine voor het spelverloop representeren. Quartus moet met *Place and Route* de flipflops zodanig plaatsen en verbinden dat gegarandeerd dat de data die vanuit de eerste flipflop worden gestart (men spreekt wel van *launch-edge*), functioneel correct op de eerstvolgende systeem-klokflank (*latch-edge*) door de tweede flip-flop wordt overgenomen. Als in dit pad een setup-vertraging (*negative slack*, zie verderop) zou optreden, dan kan het zomaar zijn dat de data die in de tweede flipflop moet worden opgeslagen, vanwege de looptijd van het signaal nog niet is gearriveerd, waardoor oude en dus foute data wordt opgeslagen. Een schending van de *hold time* kan ertoe leiden dat de gegevens die moeten worden overgenomen in de volgende cyclus, al in de huidige cyclus worden overgenomen. Bovendien kan een *Setup and Hold Time Violation* ertoe leiden, dat de flipflop in een ongedefinieerde, metastabiele toestand terecht komt [8]. Een setup-schending treedt alleen dan niet op, als de klokfrequentie in vergelijking met de datavertraging laag genoeg is – dus als geldt:

$$\text{tclk} < \text{tsu} + \text{th} + \text{tpd} + \text{tco}$$

**tsu**: *setup time*; de tijd die de data stabiel moet zijn vóór de actieve flank,

**tpd**: *propagation* (vaak ook *path*) *delay*; de tijd die het signaal nodig heeft, om van A naar B te komen,

**tco**: *clock to out*; de tijd die het signaal nodig heeft om na de flank aan de uitgang van de poort (FF) te verschijnen,

**th**: *hold time*; de tijd gedurende welke het signaal na de actieve flank nog stabiel moet blijven.

Voor een *Setup Timing Check* moet het timing analyse-tool voor

elk intern pad de tijd berekenen die het duurt voordat het signaal van de ene flipflop bij de volgende aankomt (*data arrival path*). Daarbij worden de interne looptijden van de gekozen bedrading en de vertraging van het kloksignaal (*clock arrival path*) verrekend. De waarde die aangeeft of aan de hierboven beschreven voorwaarde voor een pad is voldaan, wordt *slack* genoemd (hier *Setup Slack*). Deze is positief als aan de voorwaarde is voldaan en negatief als sprake is van een incorrecte timing. Om Quartus de klokfrequentie van het ontwerp mee te delen, volstaat het volgende commando in het \*.sdc-bestand:

```
create_clock -name clk -period 83
[get_ports {clk12m}]
```

Dit creëert een *base clock* met een periodeduur van 83 ns en specificeert dat het kloksignaal in het ontwerp via het signaal `clk12m` beschikbaar is. Dit geldt alleen voor het MAX1000-board, voor het CPLD-board moet de `period`-waarde op 25 worden ingesteld. Voor de interne logica volstaat deze eerste stap al. Als tweede stap moeten we nog de SPI in- en uitgangen ‘constrainen’, dat wil zeggen: het tool laten weten dat de signalen op de STB-, DIO- en kloklijnen een bepaalde tijdsrelatie hebben, die door de routering in de FPGA/CPLD niet mag worden verstoord.

Daarom definiëren we nog een tweede klok en noemen deze `spi_clk`:

```
create_clock -name spi_clk -period $spi_per
[get_ports {sck}]
```

Omdat het .sdc-formaat is gebaseerd op TCL, kunnen variabelen daar met `set` gedefinieerd worden, en is de waarde toeganke-lijk met de `$`-referentie (het verdient aanbeveling ‘pakkende’ namen te gebruiken). Vervolgens missen we nog de tijdsrelaties tussen de externe schakeling en de in- en uitvoerpaden. Dit kan worden meegedeeld met de commando’s `set_output_delay` en `set_input_delay`. Of Quartus het met de ingegeven Timing Constraints eens is, kunnen we zien in het *Compilation Report Window* in de subdirectory *TimeQuest Timing Analyzer* of direct in het timing analyse-tool, dat met een klik op *Tools -> TimeQuest Timing Analyser* wordt gestart. Niet alle constraints kunnen hier in detail worden besproken. Alle voor dit project noodzakelijke

constraints zijn te vinden in het bestand *reactionTimer.sdc*. Van belang is vast te stellen dat een voltooide VHDL-code nog lang niet betekent dat we ook een compleet voltooide digitale schakeling hebben. Verdere informatie over timing constraints is te vinden op internet [7][9][10].

## Conclusie

De reactietijdmeting in het laatste deel van deze VHDL-serie is een goed voorbeeld om afsluitend de voor- en nadelen van een hardware-programmeertaal te ervaren. Om dit spel in een Arduino-sketch te implementeren zijn 50 regels code ruim voldoende. Bij VHDL komen we met 50 regels niet ver, het zijn er uiteindelijk meer dan 1200 geworden. Om even snel een MSR-schakeling in elkaar te knutselen zult u dan ook niet snel uw toevlucht nemen tot een CPLD of FPGA. Maar als een project hoge I/O-snelheden of deterministisch gedrag vereist (of korte vertragingen, of een groot aantal parallelle verwerkingsseenheden), dan kan het gebruik van FPGA’s technisch gewenst zijn. Een mooie mix van beide werelden bieden *soft processoren* zoals in het SCCC-project in dit nummer. Wie nog meer ideeën heeft voor het gebruik van VHDL, wordt hierbij uitgenodigd om die naar Elektor te sturen! ▶

(180285-D-03)


**IN DE STORE**

<a href="#">CPLD Breakout Board</a> <a href="http://www.elektor.nl/cpld-breakout-board-160425-91">www.elektor.nl/cpld-breakout-board-160425-91</a>
<a href="#">MAX1000 – IoT Maker Board</a> <a href="http://www.elektor.nl/max1000">www.elektor.nl/max1000</a>

## Weblinks

- [1] Hardware-ontwerp met (V)HDL (3), Elektor maart/april 2019: [www.elektrormagazine.nl/180285-C-03](http://www.elektrormagazine.nl/180285-C-03)
- [2] Projektpagina bij dit artikel: [www.elektrormagazine.nl/180285-D-03](http://www.elektrormagazine.nl/180285-D-03)
- [3] LED&KEY TM1638: [www.amazon.de/RoboMall-TM1638-Taster-8-stelligem-Display/dp/B0117BUAPU](http://www.amazon.de/RoboMall-TM1638-Taster-8-stelligem-Display/dp/B0117BUAPU)
- [4] LED&KEY TM1638 Demo 1: <https://draeger-it.blog/arduino-lektion-42-tm1638-led-taster-shield/>
- [5] LED&KEY TM1638 Demo 2: [www.forum.g-heinrichs.de/viewtopic.php?t=108](http://www.forum.g-heinrichs.de/viewtopic.php?t=108)
- [6] LED&KEY TM1638 Demo 3: [www.led-treiber.de/html/leds\\_display.html](http://www.led-treiber.de/html/leds_display.html)
- [7] TimeQuest User Guide: [https://fpgawiki.intel.com/wiki/File:TimeQuest\\_User\\_Guide.pdf](https://fpgawiki.intel.com/wiki/File:TimeQuest_User_Guide.pdf)
- [8] White Paper (Altera): Understanding Metastability in FPGAs:  
[www.intel.com/content/dam/www/programmable/us/en/pdfs/literature/wp/wp-01082-quartus-ii-metastability.pdf](http://www.intel.com/content/dam/www/programmable/us/en/pdfs/literature/wp/wp-01082-quartus-ii-metastability.pdf)
- [9] Eli Billauer: Quartus Timing Analysis on `set_input_delay` and `set_output_delay` constraints:  
<http://billauer.co.il/blog/2017/04/io-timing-quartus-calculation/>
- [10] Intel FPGA Wiki: Constrain SPI Core: [https://fpgawiki.intel.com/wiki/Constrain\\_SPI\\_Core](https://fpgawiki.intel.com/wiki/Constrain_SPI_Core)



# JOY-iT JDS6600 DDS-signalgenerator veel signaal voor weinig geld

Harry Baggen (Elektor Labs)



Zit er misschien een addertje onder het frequentiegras? We zullen maar meteen beginnen met het slechte nieuws. En dat is de behuizing van de JOY-iT JDS6600. Die ziet er nogal goedkoop uit, daar had de fabrikant wel een paar euro meer in mogen investeren. Maar het goede nieuws is dat het ding voor nog geen 140 euro een flink frequentiebereik biedt, best nauwkeurig is en een hele rits aan instelmogelijkheden biedt. Daarover zo meteen meer.

## Hardware

De generator van JOY-iT zit in een eenvoudig kunststof kastje van bescheiden afmetingen. Op het front zitten een klein maar duidelijk LC-display dat alle instellingen toont, een aantal bedieningsknoppen, een draaiknop en drie BNC-bussen, twee uitgangen en een ingang voor de eveneens ingebouwde frequentieteller. Het apparaatje kan dus twee uitgangssignalen leveren die onafhankelijk van elkaar of gekoppeld kunnen worden ingesteld. De voeding wordt verzorgd door een bijgeleverde netspanningsadapter. Verder zitten in de doos ook nog twee BNC-kabeltjes met krokodillenklemmen, een BNC/BNC-kabel en een USB-kabel, dat verwacht je niet bij een apparaat met zo'n prijs.

De belangrijkste eigenschappen van de JDS6600: frequentiebereik voor sinus tot 60 MHz, voor blok en driehoek tot 15 MHz en voor de overige golfvormen tot 6 MHz. Er kan worden gekozen uit circa 15 voorprogrammeerde en maximaal 60 zelf

Vroeger had elke elektronicus in zijn (hobby-)lab wel een functiegenerator met een XR2206 staan, maar die tijden zijn al lang voorbij. Tegenwoordig werkt zo'n generator met een DDS-chip die frequenties digitaal opwekt. Helaas is een fatsoenlijke DDS-functiegenerator niet echt goedkoop – of toch wel? De JOY-iT DDS-functiegenerator belooft een groot frequentiebereik en een heleboel mogelijkheden voor een bodemprijs.

geprogrammeerde golfvormen. De uitgangsspanning bedraagt maximaal 20 V<sub>pp</sub> onder 10 MHz, daarboven is nog 10 V<sub>pp</sub> (tot 30 MHz) resp. 5 V<sub>pp</sub> (tot 60 MHz) beschikbaar. De instelbare offsetspanning heeft vergelijkbare waarden. Verder kan de JDS6600 signaalbursts en frequentiesweeps opwekken. De ingebouwde frequentieteller is geschikt voor signalen tot 100 MHz en 2...20 V<sub>pp</sub>.

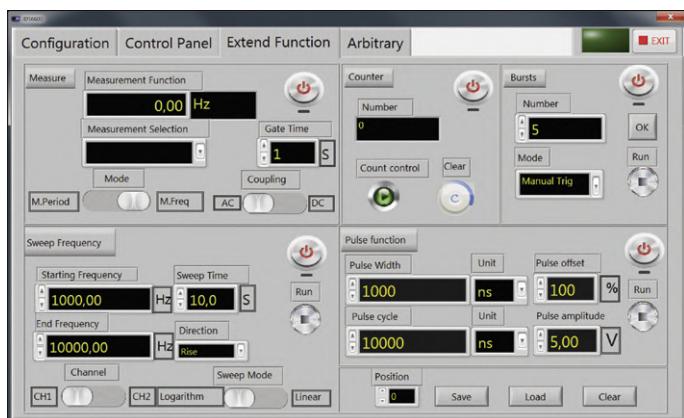
## Gebruik

De bediening van de JDS6600 is vrij overzichtelijk opgezet. Naast het display zitten vier functietoetsen waarvan de momentele functie afhankelijk is van de gekozen instellingen. Op het display worden de instellingen van beide kanalen weergegeven, bij het signaal dat bovenaan staat wordt ook de gekozen signaalvorm getoond. De meeste toetsen hebben meerdere functies, dat is in het begin wat verwarring. Zo kun je door éénmaal drukken op een CH-toets omschakelen naar het andere kanaal om dit in te stellen, terwijl éénmaal drukken op een reeds geselecteerd kanaal tot gevolg heeft dat dit kanaal wordt uitgeschakeld. Lang indrukken van de CH2-toets heeft tot gevolg dat kanaal 2 op het display bovenaan getoond wordt, met de golfvorm erbij. Door lang indrukken van CH1 verschijnt kanaal 1 weer bovenaan. Duidelijk? De toetsen werken weliswaar goed, maar bij het indrukken merk je dat de naastliggende toetsen ook mee bewegen, dat had mechanisch iets beter opgelost kunnen worden.

Het instellen van de waarden op het display gebeurt door met twee cursortoetsen eerst een cijfer te kiezen en dat dan met de draaiknop te veranderen. Bij de frequentie worden wel erg



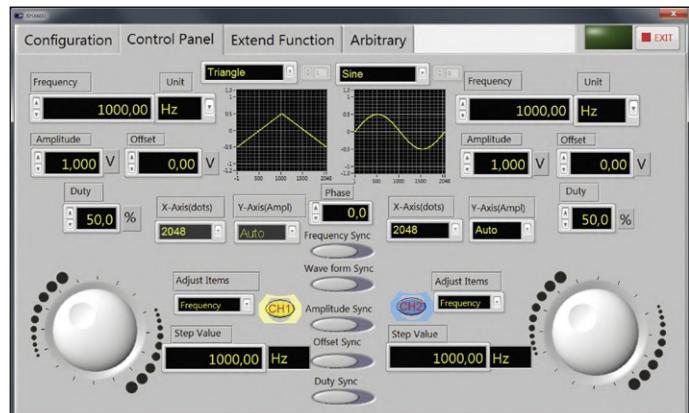
Figuur 1. Alle instellingen voor de twee uitgangssignalen zijn op het display te zien.



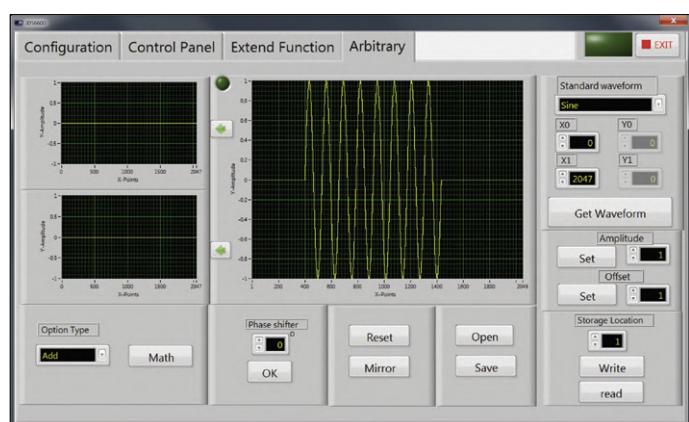
Figuur 3. Op deze tab zijn alle extra mogelijkheden beschikbaar, zoals frequentiesweeps.

veel cijfers weergegeven, dat maakt het instellen wat moeizaam. De OK-knop schakelt beide uitgangen in of uit, een wat vreemde benaming voor deze functie. En met de MOD-knop kun je sweeps, bursts en pulsbreedtes instellen, maar geen modulatie. Het zijn maar kleinigheden, al met al valt er toch wel aardig mee te werken als je deze eigenaardigheden door hebt. De geleverde signalen zien er op het oscilloscopscherm best aardig uit, hoewel ik afgaande op de specificaties (sampling-rate 266 Msamples/s, golfvorm-lengte 2048 punten, 14-bits resolutie) iets fraaiere golfvormen had verwacht. FFT-metingen aan een sinus in het audiogebied lieten vervormingsresten van iets meer dan 1% zien. De fabrikant vermeldt zelf minder dan 0,8%, dat wordt dus net niet gehaald bij mijn testexemplaar. Blokgolven zien er daarentegen behoorlijk goed uit, met vrij steile flanken en weinig overshoot.

Het frequentieverloop is vrij lineair: boven 30 MHz stijgt de uitgangsspanning zo'n 2 dB en daalt vervolgens tot circa -1 dB bij 60 MHz, helemaal niet gek! Op de website van JOY-iT kan een bijbehorend programma [1] worden gedownload waarmee de JDS6600 geheel vanaf de PC via een USB-verbinding kan worden bediend. Zoals veel Chinese software is de vormgeving niet geweldig, maar alles functioneert goed. Er zijn verschillende tabs voor de diverse functies, waaronder ook een voor zelf golfvormen samen te stellen. Na even uitproberen had ik daarmee in een mum van tijd een sinusburst gemaakt voor het testen van luidsprekers.



Figuur 2. Het instellen is ook via de PC mogelijk.



Figuur 4. Het zelf programmeren van een golfvorm gaat vrij gemakkelijk.

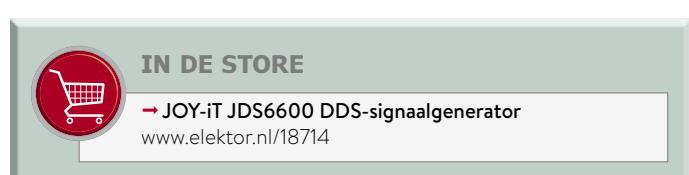
## Conclusie

De JOY-iT JDS6600 heeft een heleboel te bieden voor een prijs van nog geen 140 euro. Er zijn weliswaar enkele punten die voor verbetering in aanmerking komen, maar alles bij elkaar genomen krijg je ontzettend veel voor je geld. Het enige dat er niet op zit, is een modulatiemogelijkheid, maar verder vind je praktisch alles aan signaal mogelijkheden die je incidenteel voor een hobbylab nodig hebt. Daar hoeft je geen professioneel apparaat met een fraaie behuizing en een veel hogere prijs voor aan te schaffen. ◀

190310-01

## Weblinks

- [1] Software voor PC-bediening:  
<http://anleitung.joy-it.net/?goods=jds6600>



# Juweeltjes of e-schroot? u mag het zeggen...

**Jan Buiting** (redactie Retrotronica)

Met de onstuitbare omschakeling van elektronische meet-hardware naar digitaal/software/simulatie en veel plastic, zijn high-end instrumenten uit de oude analoge en ‘vroeg digitale’ periode afgeschreven en staan ze in kelders stof te verzamelen. Ik ontruimde onlangs zo’n locatie, en heb nu een vraag voor de lezers van Retrotronica: van welk exemplaar uit deze verzameling e-schroot zou u in de volgende aflevering een reparatieverslag willen zien?



De apparaten die u hier ziet, zijn afkomstig uit een grotere verzameling die bijna de hele kofferbak (1775 liter) van mijn Peugeot 308 SW in beslag nam. Hoewel bij het ophalen werd gezegd dat alle apparaten defect waren, bleek dat sommige alleen moesten worden gekalibreerd. In veel laboratoria verlopen de kalibratiecontracten na enkele jaren, daarna wordt de apparatuur afgekeurd voor wetenschappelijk onderzoek en vervolgens afgevoerd.

De hier besproken apparaten werkten na het inschakelen niet of niet goed. Tot nu toe kreeg Retrotronica positieve reacties op verhalen over reparatie en restauratie van oude apparaten, en daarom wil ik in deze aflevering uw interesse peilen voor een verhaal over een van de hieronder in het kort beschreven

instrumenten. Welk apparaat is uw favoriet voor een gedetailleerd en geïllustreerd reparatieverslag? Laat het me per e-mail weten op:

- jan.buiting@elektor.com;
- onderwerp: Juweeltjes of e-schroot?

Eventuele tips en ideeën voor het oplossen van de problemen zijn van harte welkom.

Van alle apparaten werd de buitenkant schoongemaakt en volgde een eerste inspectie met betrekking tot de elektrische veiligheid. Hier werden geen problemen gevonden. Voor zover mogelijk werd documentatie verzameld en werd aan

verschillende experts advies gevraagd. Hieronder ziet u de apparaatbeschrijvingen met het naar mijn mening meest verdachte (onder)deel. De stapels (zie de kopfoto) zijn genummerd van links naar rechts. De apparaten per stapel van onder naar boven.

### Stapel 1

#### Tektronix Type 11802 Digital Sampling Oscilloscope

Dit (ooit) extreem dure high-end apparaat stamt uit het midden van jaren '90. De laatste kalibratie vond plaats in 2008. Het heeft een 24-GHz plug-in type SD24.

Status: start op, scherm in orde, zelftest meldt: "error E1811, Subsys Executive".

Eerste inschatting: connectoren; SMD-elco's; NVRAM-batterij.

#### Philips PM 2436 DC-Micrometer

Dit is een instrument uit de beroemde 'grijze' reeks (PM 24xx) die door Philips in de vroege jaren '70 werden gemaakt. Fantastisch voor sub-milliampèremetingen en gedurende tientallen jaren een standaardapparaat in natuur- en scheikundelaboratoria.

Status: start op, meter slaat maximaal uit in alle bereiken en in alle modi.

Eerste inschatting: soldeerverbindingen; elco's; halfgeleiders.

### Stapel 2

#### Marconi Instruments TF2173 Digital Synchronizer en TF2016 10 kHz – 120 MHz AM/FM Signal Generator

Deze set vormt een breedband-PLL signaalgenerator (gestabiliseerd met een kristaloven) die in de jaren '80 voor grotere laboratoria net betaalbaar was.

Status: start op. Wordt eerst warm, dan ontsnapt lichte rook aan de achterzijde achterkant van de TF2016. Systeem lockt niet. Uitgangsfrequentie fluctueert op alle bereiken.

Eerste inschatting: interne voedingen.

#### Philips PM 2504 electronic V<sub>A</sub>Ω Meter

Uit de 'zwarte' PM 25xx-reeks. Draagbare batterijgevoede multimeter. De meegeleverde netadapter fungeert als voeding en als lader.

Status: start op. V- en A-meetwaarden zien er goed uit op DC en AC. Meternaald slaat helemaal uit in de weerstandsbereiken.

Eerste inschatting: geen idee.

### Stapel 3

#### Schomandl ND 100 M Frequency Decade 300 Hz – 100 MHz

Deze met Duitse *Gründlichkeit* geproduceerde zwaargewicht heeft een zeer stabiele kristaloven en is gespecificeerd voor een resolutie van 0,1 Hz.



Status: start op. Uitgangsfrequentie en -amplitude fluctueren. Waarschuwingsslampje blijft uit.

Eerste inschatting: interne voedingen; kristal overleden.

#### Tektronix TDS 520B Two-Channel Digitizing Oscilloscope

Status: start op. Zelftest geslaagd. Kanaal 2 heeft ernstige vervorming, AC-modus gelijk aan DC-modus.

Eerste inschatting: condensator in versterker voor kanaal 2; SMD-elco's.

### Stapel 4

#### Advantest R9211A Digital Spectrum Analyzer

Dit 'draagbare' instrument heeft een bereik van 10 mHz (geen tikfout!) tot 100 kHz voor real-time FFT-analyse. De -A versie heeft een 'lopende zoomfunctie' en is de op één na goedkoopste in de 9211-reeks.

Status: blaast de 2A-netzekering op.

Eerste inschatting: interne voedingen. Help – geen servicedocumentatie beschikbaar!

#### Keithley Instruments 410A Picoammeter

Een eenvoudig maar populair instrument dat met zijn picoampèrebereik (10-12 A) nog 'lager' gaat dan de Philips PM 2436, en dicht bij de 'magische' femto-ampèregrens komt.

Status: start op. Meteraanwijzing instabiel in alle bereiken.

Eerste inschatting: geen idee. ▶

(180574-03)

**ESTD 2004**

[www.elektor.tv](http://www.elektor.tv)



Retrotronica is een rubriek over oude elektronica en legendarische Elektro-ontwerpen. Bijdragen, suggesties en vragen zijn meer dan welkom; stuur uw telex of telegram naar [redactie@elektor.nl](mailto:redactie@elektor.nl)



# Fab Academy

## een crash course digitaal fabriceren

Tessel Renzenbrink (Nederland)

Een volautomatische cocktailmixer uit Frankrijk, een computergestuurde kalligrafeermachine uit Japan, een CNC-machine uit Peru. Het is week 17 van de Fab Academy en studenten van over de hele wereld tonen hun zelfgemaakte machines in het online-leslokaal. Elk team heeft twee minuten om zijn groepsproject te presenteren op de livestream. Daarna neemt prof. dr. Neil Gershenfeld, hoofddocent van de Fab Academy, een minuut of twee voor tips, complimenten en – heel af en toe – wat kritiek ("it clearly needs more work").

Alles van de Fab Academy is open en kan worden gevonden op [fabacademy.org](http://fabacademy.org) (zie kader). Van de videolessen van prof. Gershenfeld tot de documentatie van de projecten van alle studenten die ooit hebben meegedaan.

Ik ben op bezoek bij de Fab Academy van De Waag in Amsterdam [1]. Het snelkookpan-gehalte van de *lap around the world* zoals week 17 genoemd wordt, is tekenend voor de Fab Academy [2]. In 20 weken leren studenten hoe ze '(bijna) alles kunnen maken'. Elke week leren ze een nieuwe vaardigheid zoals lasercutten, elektronica ontwerpen, embedded programmeren en 3D-printen. En elke week moeten ze dat toepassen in een opdracht: ontwerpen, uitvoeren, testen, debuggen en documenteren. "De Fab Academy is heel erg intens", zegt Henk Buursen, mentor van de Amsterdamse Fab Academy. "Er staat 32 uur per week voor maar meestal is het meer. Je staat er mee op en je gaat er mee naar bed". Voor student Rutger gaat het nog verder: het lab spookt zelfs door zijn dromen.

### Van Fab Lab naar Fab Academy

De *Fab Academy* is voortgekomen uit het *Fab Lab (Fabrication Laboratory)*. De werkplaats met computergestuurde gereedschappen zoals lasercutters, CNC-machines en 3D-printers heeft als doel om digitaal fabriceren voor iedereen toegankelijk te maken. Het idee voor Fab Labs ontstond in 2001 aan het Massachusetts Institute of Technology in het Center for Bits and Atoms dat onder le-

ding staat van prof. Gershenfeld. Dit keer ontsnapte er geen creatie uit het laboratorium maar ontsnapte het lab zelf aan de muren van de universiteit. Inmiddels zijn er meer duizend Fab Labs wereldwijd. In 2003 startte prof. Gershenfeld de collegereeks 'How To Make (almost) Anything' op de MIT-campus, waarin studenten in één semester de instrumenten van het Fab Lab leren gebruiken. In 2009 konden mensen uit andere Fab Labs bij wijze van proef de colleges volgen via een livestream. Uit dat experiment ontstond de Fab Academy.

Gaandeweg werd het *remote learning*-model waarbij hoorcolleges worden gecombineerd met zelfstudie, vervangen door een nieuw leermodel: *distributed learning*. Hierbij wordt kennis uitgewisseld op mondial en lokaal niveau. In lokale werkgroepen leren studenten van hun mentor en elkaar. Buursen en zijn studenten komen elke donderdag bij elkaar in het Amsterdamse Fab Lab om het gebruik van de machines en de opdracht te bespreken. Op mondial niveau zijn de lokale academies verbonden via de interactieve online-les en doordat iedereen zijn projecten online documenteert en deelt.

### Open ontwerp

Het beschikbaar stellen van je ontwerp aan anderen is een belangrijke pijler van de Fab Academy. Dit *open design*-principe maakt dat iedereen in de wereld van elkaar kan leren. Net als *open*

*source-software* mogen open ontwerpen gebruikt, veranderd en verbeterd worden. Naast kennisuitwisseling wordt hier ook een ander doel mee gediend: werken naar een productiemodel waarbij steeds meer dingen lokaal geproduceerd kunnen worden. In plaats van containerschepen vol goederen de wereld over sturen, deel je ontwerpen via het internet die dan in lokale Fab Labs kunnen worden gefabriceerd.

### Machines die machines maken

"Het idee achter de Fab Academy is persoonlijk produceren", zegt Buursen. "We zijn consumenten geworden en produceren eigenlijk niets meer zelf. Het Fab Lab bevat een simpele verzameling machines waarmee je zelf bijna alles kan maken. Vroeger waren die machines groot en duur. De complete inventaris van een Fab Lab zou vroeger een miljoen dollar gekost hebben, inmiddels kom je met 100.000 dollar al een heel eind [3]. En in de toekomst zal dat richting de 10.000 gaan. Uiteindelijk wil je een Fab Lab dat een Fab Lab kan maken door met machines



nieuwe machines te fabriceren. Zoals de Fab Academy uit Peru die voor hun groepsopdracht een CNC-machine hebben gemaakt."

### Van consument naar producent

"Op de Fab Academy leer je met de hele inventaris werken in een lesprogramma van 20 modules", vervolgt Buursen. "Daarmee krijg je je autonomie terug. Je leert om datgene wat je bedenkt daadwerkelijk te maken. Daardoor leer je begrijpen hoe dingen werken. Ik vind het bijvoorbeeld belang-

rijk dat jongeren een computer uit elkaar halen zodat ze snappen hoe het werkt. Als het is ingepakt in een doosje heb je er geen zicht meer op, geen vat meer op. Maar technologie heeft een enorme invloed op de maatschappij. Als je niet weet hoe het werkt, heb je steeds minder grip op de wereld om je heen. Mijn drijfveer als Fab Academy-mentor is mensen leren hoe dingen echt werken. Het is *empowerment*: mensen de kennis en het gereedschap geven om de wereld om zich heen te kunnen snappen en naar hun hand te zetten."

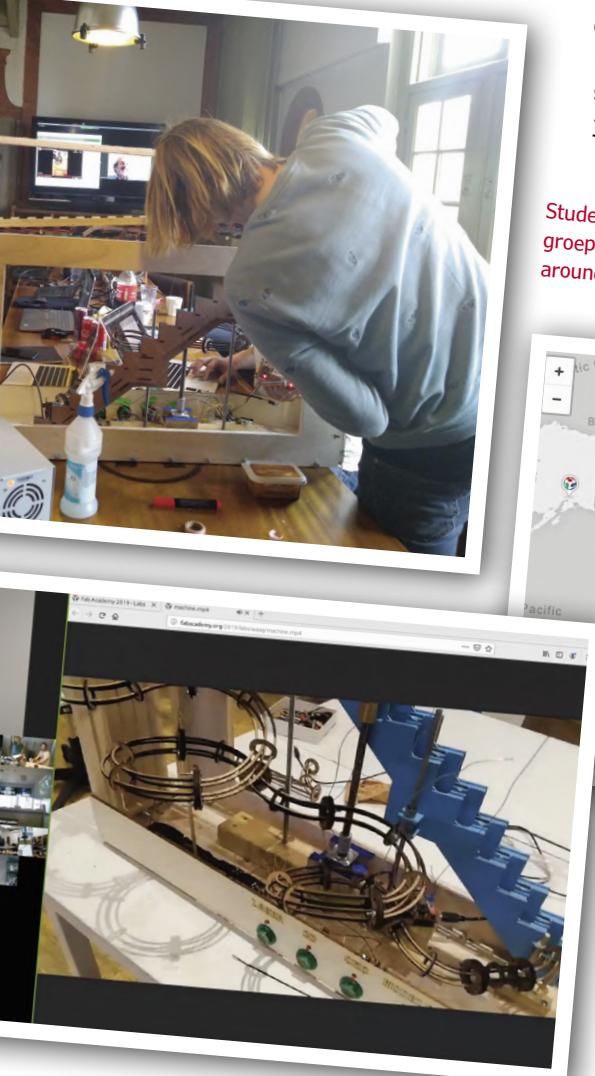
### Crash learning

De Fab Academy is intensief en veeleisend. Buursen zegt daarover: "Ik kan jou uitleggen hoe een machine werkt, of je kan er een boek over lezen maar

het opzij en vier maanden later ligt het er nog. Het maakt niet uit wat je niveau is als je begint aan de Fab Academy. Het belangrijkste is dat je nieuwe dingen leert op de top van je kunnen. En het liefst dat je zelfs ook daar overheen gaat. Iedereen die de Fab Academy doet, maakt tenminste één inzinking, huilbui of totale paniekaanval mee."

In het lokaal van de Amsterdamse Fab Academy zie ik op het scherm hoe prof. Gershenfeld stipt om drie uur begint aan de *lap around the world*: "Fab Lab Aachen, do you have a machine?" In het lokaal klinkt een zucht van verlichting. "Hij doet het op alfabetische volgorde!", roept Rutger opgelucht. Het geeft de studenten van Fab Academy de Waag nog bijna twee uur om hun groepsproject te debuggen voordat ze een live demo voor de hele wereld moeten geven. ■

180572-E-01



Screenshot van de interactieve les. Links boven prof. Gershenfeld. Rechts de presentatie van de machine van de Waag. Links onder de live streams van de deelnemende Fab Academies.



Kaart van alle Fab Labs in de wereld.

zodra je op de startknop drukt doet hij wat anders dan je verwacht had. En dan moet je op zoek naar de oorzaak. Je moet problemen oplossen, inventief zijn, beslissingen nemen. Op de Fab Academy leer je om snel te debuggen. Door de druk die er op staat, ga je dat ook daadwerkelijk doen. Want we kennen allemaal die projecten waar we aan zijn begonnen en dat iets niet werkt. Je legt

### Fab Academy

Alles van de Fab Academy is open en kan worden gevonden op <https://fabacademy.org/>. Van de videolessen van prof. Gershenfeld tot de documentatie van de projecten van alle studenten die ooit hebben meegedaan.

### Weblinks

- [1] De Waag: <https://waag.org/en/project/fab-academy>
- [2] Video van de lap around the world-les: <https://vimeo.com/academany/fab-2019/video/338245042>
- [3] Lijst met complete inventaris Fab Lab: <https://is.gd/XauCaK>

# welkom in je **WEB STORE**

TIP VAN DE REDACTIE



## Velleman Earth Listener

De Earth Listener is een praktisch en goed doordacht apparaat, uitstekend geschikt voor woonkamer of kantoor. Na de (eenvoudige) montage is het apparaat meteen gebruiksklaar; maar natuurlijk bestaat ook de mogelijkheid zelf te experimenteren en eigen uitbreidingen te ontwikkelen.

Naast de 'klassieke' meting van temperatuur, luchtdruk en luchtvochtigheid kan de sensor dankzij meting van CO<sub>2</sub> en stofdeeltjes de luchtkwaliteit

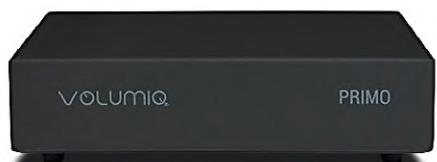
in het vertrek in de gaten houden en waarschuwen wanneer enige ventilatie op zijn plaats zou zijn.

Luc Lemmens  
(Elektor Labs)



[www.elektor.nl/velleman-earth-listener-kit](http://www.elektor.nl/velleman-earth-listener-kit)

## Volumio Primo



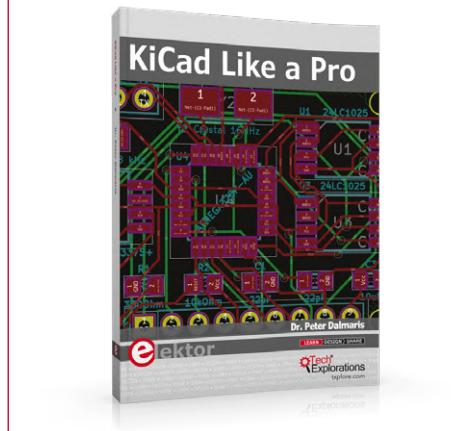
Volumio Primo is gebaseerd op software die door het bedrijf zelf is ontwikkeld. Met dit apparaat kunt u muziekbestanden afspeLEN die op een lokaal USB-medium staan, via het netwerk gedeeld worden of zich op netwerk-mediaservers bevinden. Ook is het mogelijk internet-radio te beluisteren en muziek te streamen. Afhankelijk van uw wensen kan Volumio de muziek in digitale of analoge vorm presenteren.



Ledenprijs: € 431,10

[www.elektor.nl/volumio-primo](http://www.elektor.nl/volumio-primo)

## KiCad Like a Pro



Dit (Engelstalige) boek leert u KiCad te gebruiken. Of nu een hobbyist bent of een professionele elektronicus, dit boek helpt u om snel productief te worden en te beginnen met het ontwerpen van eigen printen. Het boek volgt een praktische benadering om te leren: het bestaat uit vier projecten van oplopende moeilijkheidsgraad en verschillende recepten.



Ledenprijs: € 35,96

[www.elektor.nl/kicad-like-a-pro](http://www.elektor.nl/kicad-like-a-pro)

## Elektor Bestsellers

### 1. Raspberry Pi 4 B

[www.elektor.nl/rpi4](http://www.elektor.nl/rpi4)



### 2. SDR Hands-on Book

[www.elektor.nl/sdr-hands-on-book](http://www.elektor.nl/sdr-hands-on-book)

### 3. PCBite Kit

[www.elektor.nl/pcbite-kit](http://www.elektor.nl/pcbite-kit)

### 4. Elektor SDR Shield 2.0

[www.elektor.nl/170515-91](http://www.elektor.nl/170515-91)

### 5. Raspberry Pi Zero WH

[www.elektor.nl/rpi-zero-wh](http://www.elektor.nl/rpi-zero-wh)

### 6. SmartScope USB Oscilloscope

[www.elektor.nl/smartscope](http://www.elektor.nl/smartscope)

### 7. Camera Projects Book

[www.elektor.nl/camera-projects-book](http://www.elektor.nl/camera-projects-book)

### 8. Bakeey UM25C USB-Tester

[www.elektor.nl/bakeey-um25c](http://www.elektor.nl/bakeey-um25c)

## SDR Hands-on Book

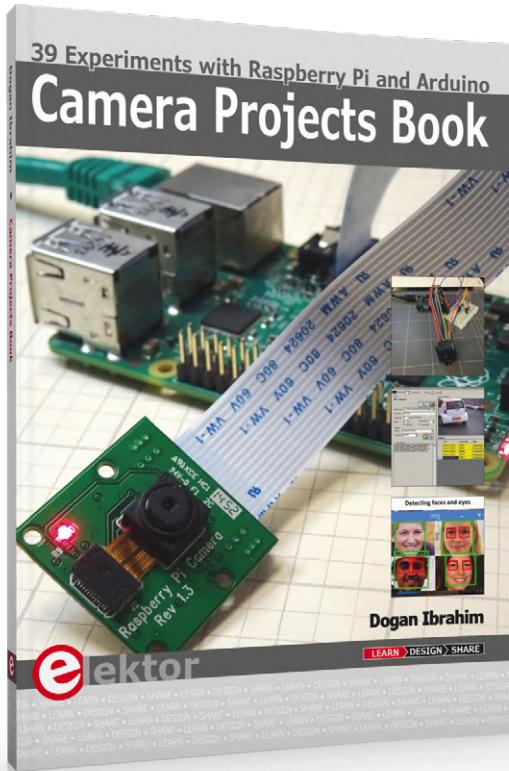


Het Elektor SDR Shield is een veelzijdige kortegolfontvanger tot 30 MHz. Met een Arduino en de passende software kunnen radiostations, morsesignalen, SSB-stations en digitale signalen worden ontvangen. In dit boek beschrijft de auteur de moderne praktijk van software-defined radio met behulp van het Elektor SDR Shield. Hij beperkt zich daarbij niet tot de theoretische achtergrond maar geeft ook tekst en uitleg over talloze open source softwaretools.



Ledenprijs: € 26,96

[www.elektor.nl/sdr-hands-on-book](http://www.elektor.nl/sdr-hands-on-book)

**nieuw**

## Camera Projects Book

In dit (Engelstalige) boek wordt op eenvoudige wijze en aan de hand van geteste en (dus) werkende voorbeeldprojecten uitgelegd hoe u een Raspberry Pi-camera en USB-webcams kunt configureren en gebruiken in cameraprojecten op basis van een Raspberry Pi.

Er worden voorbeeldprojecten behandeld om beelden vast te leggen, intervalopnames te maken, de camera en de RPi in beveiligings- en bewakingsapplicaties in te zetten, barcodes te lezen en veel meer.



**Ledenprijs: € 26,96**  
[www.elektor.nl/camera-projects-book](http://www.elektor.nl/camera-projects-book)

### LED-kweekbox



Dankzij deze kweekbox met speciale LED's groeien uw planten sneller en beter dan ooit – of nu groene vingers hebt of niet. De kit bestaat uit twee bestukte printen; u moet zelf zorgen voor een externe voeding (24 V/50 W) die de elektronica van stroom voorziet. Let op dat u niet direct in de LED's kijkt als die zijn ingeschakeld. Zelfs bij oogenschijnlijk geringe helderheid is de intensiteit van het gepulste licht zeer hoog waardoor u oogletsel kunt oplopen.



Ledenprijs € 148,46

[www.elektor.nl/horticulture-box](http://www.elektor.nl/horticulture-box)

### The Complete ESP32 Projects Guide



Dit boek wil u de Arduino IDE en MicroPython programmeertalen voor projecten op basis van de ESP32 bijbrengen, onder gebruikmaking van het bijzonder populaire ESP32 DevKitC ontwikkelboard. In het boek wordt een groot aantal projecten met de Arduino IDE en de ESP32 DevKitC gepresenteerd, variërend van simpel tot gevorderd. Alle projecten zijn getest en werken. Voor alle projecten worden de (blok)schema's en complete programmalistings inclusief commentaar gegeven.



Ledenprijs: € 35,96

[www.elektor.nl/esp32-projects-guide](http://www.elektor.nl/esp32-projects-guide)

### YDLIDAR X2 – 360-graden laser-afstandsmeter



De YDLIDAR X2 Lidar is een tweedimensionale afstandsmeter. De metingen zijn gebaseerd op het principe van driehoeksing meting; het instrument is voorzien van de benodigde optische componenten, elektrische aansluitingen en software. Tijdens de meting draait de motor de meetkop over een hoek van 360°; de combinatie van hoek- en afstandsing resulteert in een nauwkeurige en snelle rondom-scan.



Ledenprijs € 71,96

[www.elektor.nl/ydlidar-x2](http://www.elektor.nl/ydlidar-x2)

# 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 (gemarkerd door de dikdere 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 16 september 2019** naar: [hexadoku@elektor.nl](mailto:hexadoku@elektor.nl)

## De prijswinnaars

De juiste oplossing van de Hexadoku uit het juli/augustus-nummer is: **70DA8**.

De Elektor-waardebonnen van 50 Euro zijn gewonnen door Ulrich Schoor (Duitsland), Peter Maarse (Nederland), Michel Jamin (Frankrijk), Peter Wäckerle (Zwitserland) en Vladimir Saric (Servië).

Allemaal van harte gefeliciteerd!

8	3	0			E		6		B	F	4				
	5		2	3		D	1		8						
7	A		4					3			6	0			
9	2			7	D	C	E			3		A			
					6	B									
		9	8	F		A	E	6							
6	B	7	2	5			F	0	4	A	8				
E	0	2	7	A			6	4	F	1	D				
B	7	8							5	E	1				
C										A					
1				D	9	A	0				7				
3			F	E	1		4	8	6			2			
	8	4	A	C	2		5	0	7	9	B				
E			D				7			2					
		C	A	0			3	9	E						
9	D	F	6		2	B	A		4						

6	9	0	D	3	8	A	E	5	B	1	2	7	F	C	4
7	A	B	C	F	D	2	4	E	3	8	9	0	1	5	6
3	E	F	4	C	5	1	6	7	0	D	A	8	2	9	B
5	1	2	8	7	0	9	B	6	4	C	F	A	D	3	E
8	B	6	2	E	4	3	C	9	7	5	0	F	A	D	1
9	0	3	1	8	A	F	7	4	D	B	6	2	5	E	C
A	5	D	7	2	6	0	1	8	C	F	E	3	B	4	9
C	F	4	E	9	B	D	5	A	1	2	3	6	7	8	0
1	6	7	F	4	2	E	9	B	5	0	D	C	8	A	3
B	2	8	9	5	F	6	A	C	E	3	4	1	0	7	D
D	3	E	0	B	1	C	8	F	A	6	7	9	4	2	5
4	C	5	A	D	3	7	0	1	2	9	8	B	E	6	F
E	8	1	3	A	C	5	F	D	6	7	B	4	9	0	2
2	4	C	B	6	7	8	D	0	9	E	1	5	3	F	A
F	D	A	6	0	9	B	3	2	8	4	5	E	C	1	7
0	7	9	5	1	E	4	2	3	F	A	C	D	6	B	8

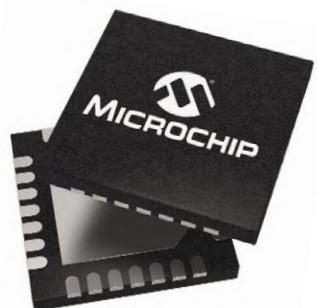
Medewerkers van Elektor International Media en hun familieleden zijn van deelname uitgesloten.



## Uw keus - elke kern, elk prestatieniveau, elke gewenste functie

Schaalbare prestaties bij veranderende eisen

Moet u een bepaald ontwerp ‘voor de zoveelste keer’ aanpassen? Laat Microchip u helpen en zet een punt achter de frustraties en verspilde tijd die het gevolg zijn van deze wijzigingen. Microchip is de enige halfgeleiderleverancier die innovaties mogelijk maakt over het totale aanbod van 8-, 16- en 32-bit microcontrollers, digitale signaalcontrollers en microprocessoren. Onze opwaarts-compatibele architecturen beschermen uw geïnvesteerde tijd bij het schrijven van programmacode. Bovendien kunt u met ons ontwikkelsysteem een gemeenschappelijk ecosysteem toepassen voor meerdere ontwerpen. Veranderende ontwerpeisen hoeven niet langer pijnlijk te zijn; kijk maar eens hoe dat met de producten van Microchip zonder al te veel inspanning mogelijk is.



**Maak het leven eenvoudiger op [www.microchip.com/Scalable](http://www.microchip.com/Scalable)**



productronica  
fast forward  
the start-up platform

powered by Elektor

COMPETE TO  
LAUNCH YOUR STARTUP  
@  
**PRODUCTRONICA 2019**

● p-ffwd 2019 – Doe mee!

12 - 15 november 2019

Messe München

Meer info:  
[www.elektormagazine.nl/p-ffwd](http://www.elektormagazine.nl/p-ffwd)

Productronica Fast Forward is brought to you by



Platinum Sponsor:



Silver Sponsor:



Bronze Sponsor:

